



UNIVERSIDAD NACIONAL DE LOJA

**ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y
LOS RECURSOS NATURALES NO
CARRERA DE INGENIERÍA EN SISTEMAS**

**Herramienta para el descubrimiento de
preferencias de uso de los cibernautas del sitio
Web de la Universidad Nacional de Loja aplicando
Redes Neuronales Artificiales SOM.**

Tesis de grado previa la obtención del título de Ingeniero
en Sistemas.

AUTORAS:

Andrea Jhoana Castillo Camacho
Herrera

Irlanda del Cisne Loján

DIRECTOR:

Ing. Wilman Patricio Chamba Zaragocín

CERTIFICACIÓN

Ing. Wilman Patricio Chamba Zaragocín

A petición de la parte interesada,

CERTIFICA:

Que la presente tesis titulada “**Herramienta para el descubrimiento de preferencias de uso de los cibernautas del sitio web de la Universidad Nacional de Loja aplicando Redes Neuronales Artificiales SOM**”, realizada por las egresadas Andrea Jhoana Castillo Camacho e Irlanda del Cisne Loján Herrera, ha sido orientada, revisada, corregida y aprobada a cabalidad, cumpliendo así con todos los requisitos y reglamentos establecidos por la carrera de Ingeniería en Sistemas; por lo que autorizo su presentación.

Loja, marzo del 2011

Ing. Wilman Patricio Chamba Zaragocín

DIRECTOR DE TESIS

AUTORÍA

Los conceptos, opiniones, conclusiones y recomendaciones emitidas en la presente tesis son de exclusiva responsabilidad de las autoras. Autorizamos a la Universidad Nacional de Loja hacer uso de la misma con fines

Andrea Jhoana Castillo Camacho

Irlanda del Cisne Loján Herrera

DEDICATORIA

A mi MADRE, quien con su abnegación y sacrificio ha sabido ayudarme haciendo posible la culminación de mis estudios universitarios.

A mis HERMANAS por su apoyo incondicional.

A mis SOBRINOS quienes con su amor me han brindado grandes momentos de alegría.

A todos mis AMIGOS por prestarme siempre su ayuda y comprensión.

Andrea

La concepción de este proyecto de tesis está dedicada a la tenacidad y lucha insaciable de mis padres, pilares fundamentales en mi vida.

A mi princesa, Sofía, mi compañera inseparable de cada jornada. Ella representa el esfuerzo y tesón en momentos de decline y cansancio.

A mis maestros de la Universidad por sus enseñanzas y ejemplo de profesionalidad que nunca olvidaré.

A ellos este proyecto, que sin ellos, no hubiese podido ser.

Irlanda

AGRADECIMIENTO

Este proyecto de tesis ha sido posible gracias al esfuerzo conjunto realizado con nuestro Director de Tesis el Ing. Wilman Chamba Zaragocín, de manera especial dejamos constancia de nuestra gratitud hacia él por su valiosa ayuda durante el desarrollo de este trabajo investigativo.

Así mismo presentamos nuestro más sincero agradecimiento al Ing. Jamil Ramón, Jefe de Informática de la Universidad Nacional de Loja, al Ing. Daniel Reyes, Responsable de Redes y Equipos Informáticos de la UNL, al Ing. Patricio Valarezo, Responsable de la sección de desarrollo de software de la UNL y a todas las autoridades de la Universidad que de una u otra manera nos permitieron tener acceso a la información necesaria para el desarrollo de esta tesis.

Finalmente un eterno agradecimiento a todos los docentes que nos brindaron sus sabias enseñanzas durante los años de estudio de la carrera universitaria, a quienes debemos gran parte de nuestros conocimientos.

Las Autoras

ÍNDICE

Portada.....	.i
Certificación.....	.ii
Autoría.....	.iii
Dedicatoria.....	.iv
Agradecimiento.....	v
Índice.....	.vi
Índice de Figuras.....	.xi
1. RESUMEN.....	1
2. INTRODUCCIÓN.....	5
3. METODOLOGÍA.....	6
3.1. MATERIALES.....	6
3.1.1. Hardware.....	6
3.1.2. Software.....	6
3.1.3. Materiales de Oficina.....	6
3.2. METODOLOGÍA PARA LA INVESTIGACIÓN.....	7
4. FUNDAMENTACIÓN TEÓRICA.....	10
4.1. SITIO WEB DE LA UNIVERSIDAD NACIONAL DE LOJA.....	10
4.1.1. Generalidades.....	10
4.1.2. Páginas del sitio web de la Universidad Nacional de Loja.....	10
4.2. ARCHIVOS LOG.....	17
4.2.1. Generalidades.....	17
4.2.2. Pre-procesamiento de Logs.....	18
4.2.3. Identificación de usuarios.....	20
4.2.4. Identificación de sesiones de usuario.....	21
4.3. REDES NEURONALES ARTIFICIALES (RNA).....	23
4.3.1. Definición.....	23
4.3.2. Elementos de una RNA.....	23
4.3.3. Tipos de RNA.....	25
4.3.4. Características de las RNA.....	26
4.3.5. Self Organizing Maps (SOM).....	27

4.3.5.1. Arquitectura de la Red SOM	29
4.3.5.2. Funcionamiento	31
4.3.5.3. Pre-procesamiento de los datos.....	32
4.3.5.4. Aprendizaje.....	32
4.4. EXPLOTACIÓN DE DATOS (DATA MINING).....	36
4.4.1. Generalidades	36
4.4.2. Explotación de datos de uso de la web	39
4.4.2.1. Pre-procesamiento.....	41
4.4.2.2. Descubrimiento de patrones	43
4.4.2.3. Análisis de patrones.....	45
4.5. HERRAMIENTAS PARA LA CONSTRUCCIÓN DEL SISTEMA	45
4.5.1. NetBeans 6.5	45
4.5.1.1. Generalidades	45
4.5.1.2. Estructura	46
4.5.1.3. Características	46
4.5.2. Java Server Faces (JSF)	47
4.5.2.1. Generalidades	47
4.5.2.2. Componentes de JSF	48
4.5.2.3. Características	49
4.5.2.4. Elementos principales de una aplicación JSF	50
4.5.2.5. Estructura de las páginas.....	50
4.5.2.6. Navegación entre páginas	51
4.5.2.7. Etapas de procesamiento de una petición.....	52
4.5.3. Hibernate	53
4.5.3.1. Generalidades	53
4.5.3.2. Características	55
4.5.3.3. Arquitectura	55
4.5.3.4. Hibernate Query Language HQL.....	57
4.5.3.5. Estructura del fichero XML.....	57
4.5.4. Joone.....	59
4.5.4.1. Descripción	59
4.5.4.2. Capas de neuronas.....	60
4.5.4.3. Sinapsis	62

4.5.4.4. El Monitor.....	63
4.5.4.5. Ventajas.....	64
4.5.5. ChartDirector	65
4.5.5.1. Definición.....	65
4.5.5.2. Características.....	65
3. EVALUACIÓN DEL OBJETO DE INVESTIGACIÓN	66
4. DESARROLLO DE LA PROPUESTA ALTERNATIVA	67
4.1. ANÁLISIS DE LA APLICACIÓN.....	67
4.1.1. Definición del problema.....	69
4.1.2. Glosario de términos.....	69
4.1.3. Documento de requerimientos	72
4.1.3.1. Requerimientos Funcionales.....	72
4.1.3.2. Requerimientos No Funcionales	75
4.2. MODELO DE LA APLICACIÓN.....	76
4.2.1. Modelo del dominio.....	76
4.3. MODELO DINÁMICO DE LA APLICACIÓN.....	77
4.3.1. Diagrama de Casos de Uso	77
4.3.1.1. Diagrama de Casos de Uso del Usuario Común	78
4.3.1.2. Diagrama de Casos de Uso del Usuario Administrador.....	79
4.2.1. Descripción de Casos de Uso.....	79
4.2.1.1. Caso de Uso: Registrar un usuario en el sistema.....	79
4.2.1.2. Caso de Uso: Iniciar sesión en el sistema.....	84
4.2.1.3. Caso de Uso: Crear una cuenta de usuario	88
4.2.1.4. Caso de Uso: Ver las cuentas de usuario	93
4.2.1.5. Caso de Uso: Eliminar las cuentas de usuario	97
3.1.1.3. Caso de Uso: Crear un proyecto.....	100
3.1.1.4. Caso de Uso: Mostrar todos los proyectos de un usuario	104
3.1.1.5. Caso de Uso: Eliminar un proyecto.....	108
3.1.1.4. Caso de Uso: Abrir un proyecto	110
3.1.1.5. Caso de Uso: Ver el resumen de un proyecto.....	112
3.1.1.6. Caso de Uso: Subir un archivo log.....	114
3.1.1.7. Caso de Uso: Cargar y Pre-procesar un archivo log	117
3.1.1.8. Caso de Uso: Ver los registros cargados de un archivo log	124

3.1.1.9. Caso de Uso: Construir las sesiones de usuario	127
3.1.1.10. Caso de Uso: Ver el detalle de las sesiones de usuario	131
3.1.1.11. Caso de Uso: Agrupar usuarios utilizando la Red Neuronal Artificial SOM.....	134
3.1.1.12. Caso de Uso: Ver la cantidad de sesiones de usuario en cada clúster	140
3.1.1.13. Caso de Uso: Ver el detalle de las páginas más visitadas por los usuarios de cada clúster	144
3.1.1.14. Caso de Uso: Ver el detalle de un archivo log	148
3.1.1.15. Caso de Uso: Eliminar un archivo log	151
3.1.1.16. Caso de Uso: Ver el detalle de un archivo de entrada.....	154
3.1.1.17. Caso de Uso: Ver el detalle de un archivo de salida	158
3.1.1.18. Caso de Uso: Cerrar un proyecto.....	162
3.2. MODELAMIENTO ESTÁTICO	164
3.2.1. Diagrama de Paquetes	164
3.2.2. Diagrama de Clases General	167
3.2.3. Diagrama de Clases por cada Caso de Uso	168
3.2.4. Modelo y Arquitectura	192
3.2.4.1. Diagrama de Componentes	192
3.2.4.2. Diagrama de base de datos	193
3.2.4.3. Diagrama de Despliegue.....	194
3.3. PRUEBAS Y VALIDACIÓN.....	195
3.3.1. Antecedentes.....	195
3.3.2. Descripción de las pruebas.....	195
3.3.2.1. Funcionalidades.....	196
3.3.2.1.1. Pre-procesamiento de archivos log	196
3.3.2.1.2. Identificación de usuarios.....	197
3.3.2.1.3. Identificación de sesiones de usuario.....	197
3.3.2.1.3.1. Identificación de preferencias de uso mediante Red Neuronal Mapa Auto-Organizativo (SOM)	197
3.3.2.1.4. Usuarios y proyectos.....	198
3.3.3. Dinámica de las Pruebas	198
3.3.4. Resultados obtenidos	199
6.5.4.1. Análisis de los archivos log del sitio web de la Universidad Nacional de Loja ..	200
3.3.4.1.1. Descripción del sitio web de la UNL	200

3.3.4.2. Páginas del sitio web de la Universidad Nacional de Loja	200
3.3.4.2.1. Análisis del log 09/01/2011 al 13/01/2011 del sitio web de la UNL.....	205
3.3.5. Valoración.....	211
4. VALORACIÓN TÉCNICA-ECONÓMICA.....	214
5. CONCLUSIONES	216
6. RECOMENDACIONES.....	218
7. BIBLIOGRAFÍA.....	219
8. ANEXOS.....	221

ÍNDICE DE FIGURAS

Figura 1: Arquitectura de mapa auto-organizado	29
Figura 2: Configuraciones más comunes en la retícula del SOM	30
Figura 3: Conexiones de una red SOM.....	34
Figura 4: Posible evolución de la vecindad en una red SOM	35
Figura 5: Modelo básico de la Minería de Datos	37
Figura 6: Web Usage Mining.....	41
Figura 7: Relaciones y responsabilidades de los componentes del patrón MVC.....	47
Figura 8: Desglose de la Arquitectura de JSF	48
Figura 9: Comparativa arquitectura en tres capas con patrón arquitectónico MVC	49
Figura 10. Ciclo de vida de JSF	52
Figura 11: Ejemplo de mapeo de una clase en Hibernate.....	54
Figura 12: Arquitectura de Hibernate	56
Figura 13: Estructura del fichero XML.....	57
Figura 14. Conexiones entre capas de neuronal.....	59
Figura 15: Modelo del dominio de la Aplicación	76
Figura 16: Diagrama de Casos de Uso	77
Figura 17: Diagrama de Casos de Uso del Usuario Común.....	78
Figura 18: Diagrama de Casos de Uso del Usuario Administrador	79
Figura 19: Página Principal del Sistema	79
Figura 20: Página “Registro de Usuario”	80
Figura 21: Mensaje de Registro de Usuario	80
Figura 22: Diagrama de secuencia Caso de Uso: Registrar un usuario en el sistema (Flujo normal)	83
Figura 23: Diagrama de secuencia Caso de Uso: Registrar un usuario en el sistema (Flujo alternativo A).....	83
Figura 24: Diagrama de secuencia Caso de Uso: Registrar un usuario en el sistema (Flujo alternativo B).....	84
Figura 25: Página “Iniciar Sesión”	84
Figura 26: Diagrama de secuencia Caso de Uso: Iniciar sesión en el Sistema (Flujo normal)	86

Figura 27: Diagrama de secuencia Caso de Uso: Iniciar sesión en el Sistema (Flujo alternativo A).....	87
Figura 28: Diagrama de secuencia Caso de Uso: Iniciar sesión en el Sistema (Flujo alternativo B).....	87
Figura 29: Diagrama de secuencia Caso de Uso: Iniciar sesión en el Sistema (Flujo alternativo C).....	88
Figura 30: Pantalla “Crear una cuenta de usuario”.....	88
Figura 31: Mensaje al Crear una cuenta de Usuario.....	89
Figura 32: Diagrama de secuencia Caso de Uso: Crear una cuenta de usuario (Flujo normal)	91
Figura 33: Diagrama de secuencia Caso de Uso: Crear una cuenta de usuario (Flujo alternativo A).....	92
Figura 34: Diagrama de secuencia Caso de Uso: Crear una cuenta de usuario (Flujo alternativo B).....	92
Figura 35: Página “Usuarios del Sistema”.....	93
Figura 36: Diagrama de secuencia Caso de Uso: Ver las cuentas de usuario (Flujo normal)	96
Figura 37: Diagrama de secuencia Caso de Uso: Ver las cuentas de usuario (Flujo alternativo A).....	96
Figura 38: Pantalla “Eliminar cuentas de Usuario”.....	97
Figura 39: Mensaje de Eliminación de una cuenta de usuario	97
Figura 40: Diagrama de secuencia Caso de Uso: Eliminar las cuentas de usuario	99
Figura 41: Página “Crear Nuevo Proyecto (Usuario Administrador)”	100
Figura 42: Página “Crear Nuevo Proyecto (Usuario Común)”	100
Figura 43: Mensaje al crear un proyecto.....	101
Figura 44: Diagrama de secuencia Caso de Uso: Crear un proyecto (Flujo normal)	103
Figura 45: Diagrama de secuencia Caso de Uso: Crear un proyecto (Flujo alternativo A)..	103
Figura 46: Diagrama de secuencia Caso de Uso: Crear un proyecto (Flujo alternativo B)..	104
Figura 47: Página “Proyectos”	104
Figura 48: Diagrama de secuencia Caso de Uso: Mostrar todos los proyectos de un usuario (Flujo normal).....	107
Figura 49: Diagrama de secuencia Caso de Uso: Mostrar todos los proyectos de un usuario (Flujo alternativo A).....	107

Figura 50: Página “Proyectos”	108
Figura 51: Mensaje de Eliminación de un Proyecto	108
Figura 52: Diagrama de secuencia Caso de Uso: Eliminar un proyecto.....	110
Figura 53: Pantalla “Abrir un Proyecto”	110
Figura 54: Diagrama de secuencia Caso de Uso: Abrir un proyecto	112
Figura 55: Página “Resumen del Proyecto”	112
Figura 56: Diagrama de secuencia Caso de Uso: Ver el resumen de un proyecto (Flujo normal)	114
Figura 57: Página “Subir Archivo Log”	114
Figura 58: Mensaje al Subir un archivo log	115
Figura 59: Página “Resultado de Subir el Archivo Log”	115
Figura 60: Diagrama de secuencia Caso de Uso: Subir un archivo log.....	117
Figura 61: Página “Cargar y Pre-procesar”	117
Figura 62: Mensaje al Cargar y Pre-procesar un archivo log	118
Figura 63: Página “Carga de Registros del Archivo Log”	118
Figura 64: Página “Resumen del Proyecto después de cargar y pre-procesar un archivo log”	118
Figura 65: Diagrama de secuencia Caso de Uso: Cargar y pre-procesar un archivo log (Flujo Normal).....	122
Figura 66: Diagrama de secuencia Caso de Uso: Cargar y pre-procesar un archivo log (Flujo alterno A)	123
Figura 67: Diagrama de secuencia Caso de Uso: Cargar y pre-procesar un archivo log (Flujo alterno B)	123
Figura 68: Página “Registros Cargados”	124
Figura 69: Diagrama de secuencia Caso de Uso: Ver los registros cargados de un archivo log (Flujo normal).....	126
Figura 70: Diagrama de secuencia Caso de Uso: Ver los registros cargados de un archivo log (Flujo alterno A)	126
Figura 71: Página “Construcción de Sesiones”	127
Figura 72: Página “Resumen de las Sesiones Cargadas”	127
Figura 73: Página “Resumen del Proyecto después de construir las sesiones de usuario”	128
Figura 74: Diagrama de secuencia Caso de Uso: Construir las sesiones de usuario....	130

Figura 75: Página “Detalle de Sesiones de Usuario”	131
Figura 76: Diagrama de secuencia Caso de Uso: Ver el detalle de las sesiones de usuario (Flujo normal)	133
Figura 77: Diagrama de secuencia Caso de Uso: Ver el detalle de las sesiones de usuario (Flujo alterno A)	134
Figura 78: Página “Construir Clústers o Grupos de usuarios”	134
Figura 79: Página “Resumen de la Construcción de Clústers”	135
Figura 80: Diagrama de secuencia Caso de Uso: Agrupar usuarios utilizando la Red Neuronal Artificial SOM (Flujo normal).....	138
Figura 81: Diagrama de secuencia Caso de Uso: Agrupar usuarios utilizando la Red Neuronal Artificial SOM (Flujo alterno A)	139
Figura 82: Diagrama de secuencia Caso de Uso: Agrupar usuarios utilizando la Red Neuronal Artificial SOM (Flujo alterno B)	139
Figura 83: Página “Cantidad de Sesiones de Usuario en cada Clúster”	140
Figura 84: Gráfico demostrativo de sesiones en clúster.....	140
Figura 85: Diagrama de secuencia Caso de Uso: Ver la cantidad de sesiones de usuario en cada clúster	143
Figura 86: Página “Páginas en cada Clúster”	144
Figura 87: Gráfico demostrativo de páginas en un clúster	144
Figura 88: Diagrama de secuencia Caso de Uso: Ver el detalle de las páginas más visitadas por los usuarios de cada clúster (Flujo normal)	147
Figura 89: Diagrama de secuencia Caso de Uso: Ver el detalle de las páginas más visitadas por los usuarios de cada clúster (Flujo alterno A).....	147
Figura 90: Página “Archivos Log”	148
Figura 91: Página “Detalle del Archivo Log”	148
Figura 92: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo log (Flujo normal)	150
Figura 93: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo log (Flujo alterno A).....	151
Figura 94: Pantalla “Lista de Archivos Log”	151
Figura 95: Mensaje de eliminación de un archivo log.....	152
Figura 96: Diagrama de secuencia Caso de Uso: Eliminar un archivo log	153
Figura 97: Página “Archivos de Entrada”	154

Figura 98: Página “Detalle de Archivo de Entrada”	154
Figura 99: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo de entrada (Flujo normal)	157
Figura 100: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo de entrada (Flujo alterno A)	158
Figura 101: Página “Archivos de Salida”	158
Figura 102: Página “Detalle de Archivo de Salida”	159
Figura 103: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo de salida (Flujo normal)	161
Figura 104: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo de salida (Flujo alterno A)	161
Figura 105: Pantalla “Cerrar Proyecto”	162
Figura 106: Diagrama de secuencia Caso de Uso: Cerrar un proyecto.....	163
Figura 107: Diagrama de paquetes preferenciasweb.....	164
Figura 108: Diagrama de paquetes preferenciasweb.fragmentos	164
Figura 109: Diagrama de paquetes preferenciasweb.control	165
Figura 110: Diagrama de paquetes preferenciasweb.dao	165
Figura 111: Diagrama de paquetes preferenciasweb.dao.hibernate	166
Figura 112: Diagrama de paquetes preferenciasweb.modelo	166
Figura 113: Diagrama de clases general de la aplicación	167
Figura 114: Diagrama de clases Caso de Uso: Registrar un usuario en el sistema	168
Figura 115: Diagrama de clases Caso de Uso: Iniciar sesión en el sistema.....	169
Figura 116: Diagrama de clases Caso de Uso: Crear una cuenta de usuario	170
Figura 117: Diagrama de clase Caso de Uso: Ver las cuentas de usuario	171
Figura 118: Diagrama de clase Caso de Uso: Eliminar las cuentas de usuario.....	172
Figura 119: Diagrama de clase Caso de Uso: Crear un proyecto (Usuario Administrador)	173
Figura 120: Diagrama de clase Caso de Uso: Crear un proyecto (Usuario Común).....	174
Figura 121: Diagrama de clase Caso de Uso: Mostrar todos los proyectos de un usuario	175
Figura 122: Diagrama de clase Caso de Uso: Eliminar un proyecto	176
Figura 123: Diagrama de clase Caso de Uso: Abrir un proyecto.....	177
Figura 124: Diagrama de clase Caso de Uso: Ver el resumen de un proyecto	178

Figura 125: Diagrama de clase Caso de Uso: Subir un archivo log	179
Figura 126: Diagrama de clase Caso de Uso: Cargar y pre-procesar un archivo log	180
Figura 127: Diagrama de clase Caso de Uso: Ver los registros cargados de un archivo log	181
Figura 128: Diagrama de clase Caso de Uso: Construir las sesiones de usuario.....	182
Figura 129: Diagrama de clase Caso de Uso: Ver el detalle de las sesiones de usuario	183
Figura 130: Diagrama de clase Caso de Uso: Agrupar usuarios utilizando la Red Neuronal Artificial SOM.....	184
Figura 131: Diagrama de clase Caso de Uso: Ver la cantidad de sesiones de usuario en cada clúster	185
Figura 132: Diagrama de clase Caso de Uso: Ver el porcentaje de acceso a cada página por los usuarios de cada clúster	186
Figura 133: Diagrama de clase Caso de Uso: Ver el detalle de un archivo log	187
Figura 134: Diagrama de clase Caso de Uso: Eliminar un archivo log	188
Figura 135: Diagrama de clase Caso de Uso: Ver el detalle de un archivo de entrada .	189
Figura 136: Diagrama de clase Caso de Uso: Ver el detalle de un archivo de salida	190
Figura 137: Diagrama de clase Caso de Uso: Cerrar un proyecto	191
Figura 138: Diagrama de componentes	192
Figura 139: Diagrama de base de datos	193
Figura 140: Diagrama de despliegue	194
Figura 141: Dinámica del Banco de Pruebas	199
Figura 142. Carga y Pre-procesamiento del archivo log.....	206
Figura 143. Resultado Carga y Pre-procesamiento del archivo log.....	206
Figura 144. Construcción de sesiones del archivo log	207
Figura 145. Resumen del Proyecto.....	207
Figura 146. Agrupar usuarios con la Red Neuronal Artificial SOM	208
Figura 147. Cantidad de sesiones en cada clúster.....	209
Figura 148. Porcentaje de sesiones en cada clúster con SOM	209
Figura 149. Páginas en un clúster	210
Figura 150. Tabulación de las encuestas.....	212
Figura 151. Resultados de las pruebas.....	212
Figura 152. Representación en porcentaje de los resultados de las pruebas.....	213

Figura 153. Representación gráfica de la tabulación de las encuestas	213
Figura 154. Detalle de costos	215

1. RESUMEN

La presente tesis cuyo tema versa: “**Herramienta para el descubrimiento de preferencias de uso de los navegadores del sitio web de la Universidad Nacional de Loja aplicando Redes Neuronales Artificiales SOM**”, tiene como objetivo principal realizar el análisis de los archivos log almacenados en el servidor web de la Universidad Nacional de Loja y con ello descubrir las páginas más visitadas por los usuarios en este sitio.

Para el diseño de la aplicación se realizaron: diagrama de casos de uso, diagramas de clases y diagramas de secuencia con la ayuda de la herramienta orientada a objetos StarUML, que es una plataforma para modelización de software que permite analizar, diseñar y documentar la aplicación. Debido a que esta herramienta no ofrece una amplia y detallada manera de generar el modelo físico de datos se optó por Enterprise Architect 3.60 para construir este modelo.

Como entorno de programación se utilizó NetBeans 6.5 el cual posee el Framework JSF que permite la utilización del patrón MVC (Modelo-Vista-Controlador) necesario para el diseño de las páginas web.

Para realizar la integración de los objetos con la base de datos se utilizó Hibernate ya que permite la persistencia de los mismos y el mapeo de las clases en una forma rápida y sencilla.

Con la implementación de la librería joone-engine.jar se logró construir la Red Neuronal Artificial SOM que permitió descubrir las páginas más visitadas por los usuarios del sitio web de la Universidad Nacional de Loja. De igual manera con la utilización de librería ChartDirector.jar se logró representar en forma gráfica los resultados obtenidos por la RNA.

En la fase de pruebas por motivos de seguridad no se pudo realizar el acceso remoto hacia el servidor web de la Universidad Nacional de Loja para obtener los archivos log objeto del análisis de la aplicación. Es por ello que el administrador del sitio web de la



UNL nos facilitó estos archivos en un medio extraíble.

Con todas estas herramientas se logró construir la aplicación web y su funcionamiento ha sido debidamente comprobado por el Ing. Patricio Valarezo Responsable de la sección de desarrollo de software de la Universidad Nacional de Loja.

Cabe indicar que para la correcta utilización de esta aplicación web se han construido los manuales de usuario y programador.



1. ABSTRACT

The present thesis which topic turns: “**Herramienta para el descubrimiento de preferencias de uso de los cibernautas del sitio web de la Universidad Nacional de Loja aplicando Redes Neuronales Artificiales SOM**”, has as principal objective realize the analysis of the files log stored in the webserver of Loja's National University and with it to discover the pages most visited by the users in this site.

For the design of the application they were realized: graph of use cases, graphs of classes, graphs of sequence and graphs of hardiness with the help of the tool orientated to objects StarUML, which is a platform for modeling software that it allows to analyze, to design and to document the application. Due to the fact that this tool does not offer a wide and detailed way of generating the physical model of information it was chosen by Enterprise Architect 3.60 to construct this model.

Since environment of programming was in use NetBeans 6.5 which possesses the Framework JSF that allows the utilization of the boss MVC (Model-View-Controller) necessary for the design of the web pages.

To realize the integration of the objects with the database it was used you Hibernate yourself since it allows the persistence of the same ones and the mapping of the classes in a quick and simple form.

With the implementation of the library joone-engine.jar it was achieved constructed the Network Neuronal Artificial SOM that allowed discovering the pages most visited by the users of the web site of Loja's National University. Of equal way with the utilization of library ChartDirector.jar it was achieved to represent in graphical form the results obtained by the RNA.

In the test phase for safety reasons were unable to perform remote access to the web server of Loja's National University to get the log files being analyzed application. That



is why the web site administrator of the UNL gave us these files on removable media.

With all these tools it was achieved to construct the web application and his functioning it has been due verified by the Engineer Patricio Valarezo Responsible for the section of development of software of Loja's National University.

It is necessary to indicate that for the correct utilization of this web application there have been constructed the manuals of user and programmer.



2. INTRODUCCIÓN

Actualmente el proceso de descubrimiento de hábitos de uso de sitios web ha sido obtenido utilizando métodos estadísticos, lo cual constituye un rezago en las nuevas técnicas (RNA), esto debido al escaso conocimiento de la aplicabilidad de las mismas para la creación de herramientas adaptativas e inteligentes que manejen este proceso de una manera más satisfactoria.

Una de las aplicaciones de las RNA en la gestión de la información es el Data Mining o Minería de Datos; la cual se refiere al estudio de la selección y recuperación de información especializada para descubrir y extraer información útil de documentos y servicios web como: texto, imágenes, video, hiperenlaces, archivos log, etc.

La identificación de hábitos o preferencias de uso de sitios web, WebUsageMining, consiste en el proceso de aplicar técnicas de minería de datos para el descubrimiento de patrones de uso en páginas web. Ésta utiliza los datos registrados en los archivos log de acceso de los servidores web, donde se registra el comportamiento de navegación de los usuarios.

Surgiendo así la necesidad de crear una herramienta que permita descubrir las preferencias de uso de los navegadores del sitio web de la Universidad Nacional de Loja, haciendo un análisis minucioso de los datos contenidos en los archivos log almacenados en el servidor web y de esta manera aportar con nuevos conocimientos que orienten al Administrador de este sitio para que pueda adaptarlo a los requerimientos de los usuarios.

Esta tesis fue desarrollada gracias a la colaboración especial del Ing. Wilman Chamba Director de este proyecto, Ing. Jamil Ramón, Jefe de Informática de la UNL, Ing. Daniel Reyes, Responsable de Redes y Equipos Informáticos de la UNL, Ing. Patricio Valarezo, Responsable de la sección de desarrollo de software de la UNL y demás autoridades de la Universidad Nacional de Loja.



3. METODOLOGÍA

3.1. MATERIALES

3.1.1. Hardware

- Computador Intel Pentium IV 2.8 Ghz, memoria RAM 1.5 GB, disco duro 80 GB.
- Computador portátil HP Pavilion dv4 Notebook PC, Intel CoreI3, memoria RAM 4 GB, disco duro 300 GB.
- Impresora HP Deskjet 3940
- Flash memory Imation 2 GB

3.1.2. Software

- Sistema Operativo: Microsoft Windows XP, Linux
- Lenguaje de programación: jdk 1.6.3
- Entorno de programación: Netbeans 6.5
- Servidor de Base de datos: MySQL Server 5.0
- Administrador de base de datos: MySQL-Front 5.1
- Herramienta de Mapeo objeto-relacional (ORM): hibernate3.jar
- Servidor web: Apache 6.0.18
- Herramienta de modelado lógico y dinámico: StarUml 5.0
- Herramienta de modelado físico de datos: Enterprise 3.60
- Librería para construir la Red Neuronal Artificial SOM: joone-engine.jar
- Librería para construir gráficos: ChartDirector.jar

3.1.3. Materiales de Oficina

- Esferográficos
- Lápices
- Borradores
- Carpetas
- Papel bond



- CD's
- Grapadora
- Perforadora
- Copias
- Anillados
- Empastados
- Cartuchos de tinta
- Internet

3.2. METODOLOGÍA PARA LA INVESTIGACIÓN

La metodología de desarrollo de software que se utilizó para la construcción de esta herramienta fue ICONIX debido a que está basada en un proceso que se utiliza para analizar, modelar y diseñar los casos de uso que son manejados dentro del contexto del problema.

En la fase de recolección de requerimientos se utilizó la investigación de campo ya que recurrimos al lugar donde se encuentra el objeto de investigación es decir a la Jefatura de Informática para obtener toda la información relacionada con los archivos log almacenados en el servidor web de la Universidad Nacional de Loja.

Luego se establecieron los requerimientos funcionales, no funcionales, el glosario de términos y el modelo del dominio de la aplicación.

Seguidamente y en base a los requerimientos se diseñaron los diagramas de caso de uso para el usuario y el administrador en base a los cuales se redactaron los mismos indicando el flujo normal y alterno de los eventos entre el sistema y los actores.

Además se realizaron los prototipos de pantallas, diagramas de clases y diagramas de secuencia; todos estos elementos sirvieron para comprobar los requerimientos de los casos de uso.



Para la codificación del diseño de la aplicación se utilizó NetBeans 6.5 con el Framework **JSF** el cual está basado en el patrón MVC (Modelo-Vista-Controlador) que sirve para crear aplicaciones basadas en la web.

Para realizar la integración de los objetos con la base de datos se utilizó **Hibernate** ya que permite la persistencia de los objetos y el mapeo de las clases en una forma rápida y sencilla. Además facilitó el manejo de consultas Java a la base de datos.

Para la construcción Red Neuronal Artificial SOM se añadió la librería **joone-engine.jar** la cual ofreció toda la documentación necesaria para ingresar los patrones, procesarlos y obtener las salidas de la red.

Para representar los resultados obtenidos a través de la Red Neuronal Artificial SOM se añadió la librería **ChartDirector.jar**.

Posteriormente se realizaron las pruebas de la herramienta con el Responsable de la sección de desarrollo de software de la Universidad Nacional de Loja y cinco estudiantes de la carrera de Ingeniería en Sistemas comprobando de esta manera su correcto funcionamiento.

Apoyándonos en el **método analítico** se pudo obtener cada uno de los elementos que intervinieron en el descubrimiento de preferencias de uso de los usuarios en el sitio web de la UNL y el **método sintético** el cual permitió unir todos estos elementos necesarios en el diseño y construcción de la aplicación.

Otro método que se utilizó fue el **método descriptivo** presente en las etapas de: formulación del problema, formulación de hipótesis, recolección de datos, organización de datos, clasificación de datos, análisis e interpretación de datos y conclusiones.

Finalmente cabe indicar que se utilizaron las técnicas de la observación directa, entrevista y encuesta.

La técnica de la observación directa fue utilizada para constatar la estructura de los archivos log del servidor del sitio web de la Universidad Nacional de Loja y para obtener las páginas del mismo.



La técnica de la entrevista fue aplicada al Ing. Patricio Valarezo, Responsable de la sección de desarrollo de software de la Universidad Nacional de Loja y la técnica de la encuesta fue aplicada a cinco estudiantes de la carrera de Ingeniería en Sistemas, quienes manifestaron un alto nivel de aceptación de esta aplicación web.



4. FUNDAMENTACIÓN TEÓRICA

4.1. SITIO WEB DE LA UNIVERSIDAD NACIONAL DE LOJA

4.1.1. Generalidades

El sitio web de la Universidad Nacional de Loja es de carácter básicamente educativo y permite obtener información general de la Institución tanto en la parte administrativa como académica. En él se exponen varios artículos relacionados con las carreras de las distintas Áreas que conforman la Universidad Nacional de Loja, permitiendo que los usuarios puedan enterarse de la oferta académica que ofrece así como también de los próximos eventos y últimas noticias correspondientes a cada Área.

Ofrece a los usuarios el servicio de *Biblioteca Virtual*, cuyo principal objetivo es proveer el acceso a documentos digitales y constituye una plataforma para que los miembros de la comunidad universitaria compartan sus estudios e investigaciones a través de artículos científicos, proyectos de investigación, tesis de grado, etc.

Además provee el servicio Libro de Visitas en el cual los usuarios del sitio pueden dejar sus comentarios y sugerencias luego de su visita. También pueden opinar sobre temas relacionados con la Universidad Nacional de Loja.

4.1.2. Páginas del sitio web de la Universidad Nacional de Loja

A continuación se detallan las principales páginas que componen el sitio web de la Universidad Nacional de Loja:

- **index.php:** Información general de la Universidad Nacional de Loja, servicios y novedades relacionadas con la misma.
- **quienes-somos.html:** Información general acerca de los servicios que ofrece la Universidad, la letra de su himno y el escudo.
- **mensaje-del-rector.html:** Mensaje del Rector a los miembros de la Institución.



- **misión-vision.html:** Página que hace referencia al compromiso de la Universidad Nacional de Loja con la formación académica, profesional, institucional y social.
- **historia.html:** Página que informa sobre la creación de la Universidad Nacional de Loja y los distintos eventos acontecidos hasta la fecha.
- **directorío.php:** Listado con los datos de los integrantes del Directorio de la Universidad Nacional de Loja.
- **egresado.php:** Listado con los datos de los Egresados de la Universidad Nacional de Loja.
- **transparencia.html:** Ofrece los siguientes links: Estructura Orgánica Funcional, Distributivo de personal, Escala de remuneraciones, Servicios, Contratos colectivos, Formatos para solicitudes, Presupuesto Anual, Liquidación de presupuesto, Auditorías internas y gubernamentales, Procesos precontractuales y contractuales, Incumplimiento de contratos, Planes y programas de la institución, Créditos internos y externos, Rendición de cuentas, Viáticos y movilizaciones, Responsable de la información y Consejo Académico Administrativo Superior (CAAS).
Todos los links indicados tienen la opción para descargar archivos en formato pdf.
- **bienestar/xmlrpc.php:** Página principal del Departamento de Bienestar Universitario que contiene los siguientes links: Presentación, Antecedentes, Misión, Visión, Líneas de Acción, Políticas, Objetivo, Servicios Universitarios, Servicio Psicopedagógico, Derechos y Obligaciones, Servicio de Becas, Beneficios, Servicio de Salud, Servicios Informáticos, Servicios Informáticos, Culturales y Bibliográficos para personas con capacidades especiales, ¿Quiénes somos? y Oferta Académica.
- **estadísticas.html:** Posee los siguientes links: Administrativos, Docentes, Trabajadores, Estudiantes, Jornadas de Estudiantes, Presupuesto y Resumen.



Los links indicados tienen la opción para descargar archivos en formato xls y doc.

- **procesos-evaluacion.html:** Ofrece los siguientes links: Proyecto de Autoevaluación de Bienestar Estudiantil, Informe de Autoevaluación Institucional, Plan de Mejoras para el Departamento de Bienestar Estudiantil, Proyecto Aprobado de Evaluación de Autoridades y Directivos, Proyecto de Evaluación del IV Encuentro Nacional de Culturas, Informe de Evaluación del Desempeño de Autoridades y Directivos (Rector, Vicerector y Directores de Área, Representantes de Nivel, Coordinadores de Carrera), Proyecto de Evaluación del desempeño Docente del Curso Preuniversitario, Evaluación del Desempeño Docente e Informe de Autoevaluación del Departamento de Bienestar de Estudiantil Universitario (2006).
- **sistema-bibliotecario.html:** Página que muestra una breve descripción del Sistema Bibliotecario de la Universidad Nacional de Loja.
- **acerca-sistema-bibliotecario-unl.html:** En esta página se describen las siguientes áreas: Quiénes Somos, Estructura, Filosofía, Campo de Acción, Instructivo e Información de Contacto.
- **biblioteca-virtual.html:** Describe el servicio de la biblioteca virtual de la Universidad Nacional de Loja a través de los siguientes ítems: Catálogo Virtual del Sistema Bibliotecario, Bases de Datos Científicas, Enlaces a Bibliotecas Virtuales y Sitios de Interés y Aula Virtual.
- **webmail/src/login.php:** Página de ingreso al correo electrónico para los usuarios del WebMail de la Universidad Nacional de Loja.
- **editorial/xmlrpc.php:** Página principal de la Editorial Universitaria que contiene los siguientes links: ¿Quiénes Somos? y Presentación.
- **miccambio/xmlrpc.php:** Página principal del Proyecto Miccambio que contiene los siguientes links: ¿Quiénes Somos?, Visión y Misión, Objetivos y Contáctanos.



- **cisaq/xmlrpc.php**: Página principal de Investigación-Cisaq que contiene los siguientes links: Presentación, Misión y Visión, Servicios, Espacios-Físicos, Políticas de Calidad, Manual de Gestión de Calidad y LABORATORIOS ANEXOS AL CISAQ-UNL.
- **component/banners/click/6.html**: Página relacionada con la Vinculación con la Colectividad.
- **component/banners/click/2.html**: Página relacionada con la Secretaría Nacional de Ciencia y Tecnología.
- **component/banners/click/9.html**: Página relacionada con el Campus Virtual de la Universidad Nacional de Loja.
- **component/banners/click/30.html**: Página relacionada con el Sistema Bibliotecario de la Universidad Nacional de Loja.
- **component/banners/click/36.html**: Página relacionada con el Centro de Investigación y Servicios de Análisis Químicos CISAQ-UNL.
- **component/banners/click/18.html**: Página relacionada con uni>ersia una red de Universidades.
- **component/banners/click/41.html**: Página relacionada con el Proyecto LOES para la discusión.
- **component/banners/click/33.html**: Página relacionada con la UNAMAZ Asociación de Universidades Amazónicas.
- **component/banners/click/1.html**: Página relacionada con el Proyecto WAFLA.
- **component/banners/click/28.html**: Página relacionada con el DFG de la Universidad Nacional de Loja.



- **component/banners/click/15.html:** Página relacionada con el Proyecto Binacional “Integración Regional para el Manejo Ambiental Sostenible y Control de la Desertificación de Ecuador y Perú”.
- **component/banners/click/31.html:** Página relacionada con la Corporación Catamayo Centro Binacional Ecuador - Perú.
- **component/banners/click/4.html:** Página relacionada a la UnINet.
- **component/banners/click/5.html:** Página relacionada a Idrisi Ecuador - CINFA-UNL, Sitio del Centro de Recursos Idrisi en Ecuador, con sede en el Centro Integrado de Geomática Ambiental de la Universidad Nacional de Loja.
- **component/banners/click/19.html:** Página relacionada al Quinto Plan de Desarrollo Institucional 2008 – 2013.
- **component/banners/click/16.html:** Página relacionada con la Revista Universitaria.
- **component/banners/click/35.html:** Página relacionada con el Consejo Nacional de Evaluación y Acreditación de la Educación Superior del Ecuador.
- **component/banners/click/42.html:** Página que muestra el Cronograma para la Admisión al Primer Módulo de las Carreras de la Universidad Nacional de Loja - Periodo 2010-2011.
- **component/banners/click/37.html:** Página relacionada con el Sistema de Gestión Académico UNL.
- **component/banners/click/10.html:** Página de ingreso al Sistema de Gestión Académica de los estudiantes de la Universidad Nacional de Loja.
- **component/banners/click/11.html:** Página de ingreso al Sistema de Gestión Académica de los docentes de la Universidad Nacional de Loja.



- **component/banners/click/29.html:** Página relacionada con la Evaluación del Desempeño Docente de la Universidad Nacional de Loja.
- **component/banners/click/40.html:** Página que muestra información relacionada con la Maestría en Administración Educativa Modalidad Semipresencial.
- **component/banners/click/32.html:** Página que informa sobre el Proyecto de Mejoramiento de las Capacidades Científico-Técnicas de los Docentes de la Universidad Nacional de Loja.
- **component/banners/click/38.html:** Página relacionada con el Sistema de Gestión Académico UNL.
- **component/banners/click/17.html:** Página que muestra la Postal de la Universidad Nacional de Loja.
- **educativa/xmlrpc.php:** Página que muestra toda la información relacionada con el Área de Educación, el Arte y la Comunicación de la Universidad Nacional de Loja.
- **juridica/xmlrpc.php:** Página que muestra toda la información relacionada con el Área Jurídica, Social y Administrativa de la Universidad Nacional de Loja.
- **agropecuaria/xmlrpc.php:** Página que muestra toda la información relacionada con el Área Agropecuaria y de Recursos Naturales Renovables de la Universidad Nacional de Loja.
- **salud/xmlrpc.php:** Página que muestra toda la información relacionada con el Área de la Salud Humana de la Universidad Nacional de Loja.
- **apul/PaginaPrincipal.htm:** Información general sobre APUL (Asociación de Profesores Universitarios).



- **apul/QuienesSomos.htm:** Información sobre los integrantes del Directorio del APUL.
- **apul/Proyecto%20Vivienda.htm:** Fotos y estadísticas del avance de las obras del Programa de Vivienda II Etapa.
- **apul/Noticias.htm:** Noticias relacionadas con el APUL en las distintas Áreas de la Universidad Nacional de Loja.
- **apul/Convenios.htm:** Describe los convenios a los cuales que pueden acogerse los socios de APUL.
- **apul/Eventos.htm:** Descripción de los eventos que realizarán los docentes universitarios.
- **apul/Contactos.htm:** Información sobre los medios para comunicarse con el APUL.
- **apul/Docentes.htm:** Links para acceder a los datos de los docentes de las diferentes Áreas de la Universidad Nacional de Loja.
- **apul/Internet.htm:** Descripción de los servicios de Internet que pueden adquirir los docentes universitarios.
- **apul/IESS.htm:** Acceso a los servicios del IESS.
- **ageul/xmlrpc.php:** Página que muestra toda la información relacionada con la Asociación General de Empleados Universitarios de la Universidad Nacional de Loja.
- **plan-gedes.html:** Presentación del PLAN GEDES 2008 – 2013.
- **autoridades.html:** Datos del Rector y Vicerector de la Universidad Nacional de Loja.



- **junta-universitaria.html:** Datos del Rector, Vice-Rector, Representantes de la Junta Universitaria y Representantes Estudiantiles de la Junta Universitaria.
- **representantes-universidad-nacional-de-loja.html:** Nombres de los Representantes de Empleados y Trabajadores de la Universidad Nacional de Loja.
- **consejo-academico-administrativo-superior.html:** Nombres de los Representantes del Consejo Académico Administrativo Superior y Representantes del Consejo Estudiantil Académico Administrativo Superior.
- **directores-de-las-areas.html:** Datos de los Directores de cada una de las Áreas de la Universidad Nacional de Loja.
- **estructura.html:** Describe el Sistema Modular, los Niveles de Formación y cada uno de los Elementos Académicos del Sistema Modular.
- **mapa-sitio.htm:** Muestra todas las páginas que conforman el sitio web de la Universidad Nacional de Loja.

4.2. ARCHIVOS LOG

4.2.1. Generalidades

Los archivos log de un sitio web contienen la información de acceso y operaciones de todos los usuarios en orden cronológico. En cada registro del log se encuentran: Dirección IP y agente, tiempo (timestamp), identificador de sesión, método de acceso, estados varios, agentes de software, byte transmitidos, objetos requeridos (páginas, imágenes, etc.). Basándose en esta información, se puede determinar la secuencia de páginas visitadas por el usuario durante su sesión.

De acuerdo a *Oxford Dictionary* la definición de log es:

”Registro oficial de eventos durante un periodo de tiempo en particular. Para los profesionales en seguridad informática un log es usado para registrar datos o



información sobre quién, qué, cuándo, dónde y por qué? (who, what, when, where y why) un evento ocurre para un dispositivo en particular o aplicación”.

La mayoría de log son almacenados o desplegados en el formato estándar ASCII, el cual es un conjunto de caracteres para dispositivos comunes y aplicaciones. De esta forma los log generados por un dispositivo en particular pueden ser leídos y desplegados en otro diferente.

Los campos del Formato Común de Log (CLF, Common Log Format) son:

- *IP-Cliente* es la dirección IP del usuario que realiza la petición o, el nombre de la computadora del mismo si el servidor web está configurado para resolver las direcciones IP al nombre.
- *Identd* es el nombre del usuario remoto, siguiendo la RFC 1413.
- *Usuario* es el nombre de usuario que se autenticó mediante http, en las páginas que utilicen esta autenticación.
- *Fecha/Hora* es la fecha y la hora en que fue realizada la petición al servidor web.
- *Pedido* es el pedido que el cliente realizó al servidor, que generalmente incluye el método http utilizado, el URL solicitado y el protocolo empleado.
- *Estado* es el código de estado http devuelto al cliente, indicando si el pedido pudo realizarse correctamente o no, y en este último caso, el error ocurrido; y
- *Bytes* es la cantidad de bytes devueltos al cliente a causa de la petición del mismo. Si algunos de los campos no pueden ser determinados, un guión (-) es colocado en su lugar.

4.2.2. Pre-procesamiento de Logs

Esta fase está integrada por las tareas de remover los datos que no necesitamos, identificar los visitantes, identificar las acciones de los visitantes y la extracción de características. Este proceso puede ser dividido en cuatro pasos:

- **Eliminación de robots de acceso web:** Los log, por lo regular, contienen entradas de robots web, como los: crawlers, spiders, índices y otros. La



decisión de eliminarlos depende de la meta de la que tengamos por llevar a cabo.

- **Filtrado de imágenes y datos ruidosos:** Las páginas web, además cargan archivos de imágenes, sonidos o video con los datos. Por consiguiente, el servidor web graba las entradas que fueron solicitadas de imágenes, sonidos o videos, así como las que fueron enviadas. Generalmente estas entradas son descartadas cuando hacemos un análisis de log. Algunas veces, los visitantes acceden a páginas que no existen, así como las veces que el servidor falló o los visitantes no lograron identificarse bien. En todos estos casos el servidor web hace un registro con un código apropiado.
- **Extraer transacciones:** Una transacción se puede considerar como sola entrada en el registro o un sistema de entradas alcanzadas por un visitante de la misma máquina en un lapso de tiempo definido o sesión. La transacción deseada puede ser el sistema de entradas de registro de un visitante en una sola visita. El uso de los servidores Proxy establecen un límite de la sesión y puede ser el máximo de tiempo para navegar en la web en una sola visita. El registro Referrer está almacenado en el log y denota el conjunto de las URLs de las páginas previas visitadas por el usuario desde las cuales el hipervínculo fue seguido.
- **Extracción y formato de las características:** El paso final es extraer características de las transacciones disponibles. La extracción de las características implica el identificar las cualidades relevantes y el reducir la dimensionalidad de los datos excluyendo cualidades inaplicables. La tarea es convertir transacciones de longitud variable en vectores de longitud fija de la característica seleccionada. Los visitantes de la web pasan generalmente más tiempo en las páginas que más les interesan.

Después de haber aplicado todo este proceso, podemos decir que tenemos unos datos limpios para ser preparados con técnicas de data mining y es así como empezaremos la preparación de los datos.



4.2.3. Identificación de usuarios

En este paso se lleva a cabo dos niveles de identificación, en el primer nivel se identifican las peticiones de páginas realizadas por el mismo usuario durante una visita. El segundo nivel radica en reconocer a un usuario dentro de sus múltiples visitas a un determinado sitio web, con la finalidad de poder analizar el comportamiento del usuario a lo largo de días, meses o años.

Existen distintos métodos para identificar a los usuarios; mediante las cookies, la utilización de Identd,¹ su dirección IP² y la registración explícita de ellos, cada uno de los cuales posee sus ventajas y desventajas.

- Mediante la utilización de cookies. La W3C define a una cookie como “datos enviados por el servidor web al cliente, el cual los almacena localmente y los envía nuevamente al servidor en los sucesivos pedidos”. Las cookies son utilizadas para identificar a los usuarios durante las interacciones dentro de un sitio web y contienen datos que permiten al servidor mantener el registro de las identidades de los usuarios y qué acciones realizan en el sitio web. Uno de los problemas del uso de las cookies para la identificación de los usuarios es que los mismos pueden deshabilitar el soporte para cookies en sus navegadores. Otro problema se debe al hecho que las cookies son almacenadas en la computadora del usuario. El usuario podría borrarla y cuando ingrese nuevamente al sitio será reconocido como un nuevo visitante.
- Mediante la utilización de Identd. Identd es un protocolo de identificación especificado en el RFC 1413 que permite identificar a los usuarios de una conexión TCP particular. Dado un par de números de puertos TCP, devuelve una cadena de texto, que identifica al dueño de esa conexión (el cliente) en el sistema del servidor web. El problema del uso de Identd para la identificación de los usuarios reside en que el cliente debe estar configurado para el soporte de Identd.

¹Eirinaki&Vazirgiannis. Pre-procesamiento de Logs, 2003, fecha consulta: 26-12-09

²Huysmans et al., Métodos para identificar usuarios, 2003, fecha consulta: 26-12-09



- Mediante su dirección IP, en cada línea del archivo de log se almacena la dirección IP del cliente que realizó el pedido.
- Mediante la registración explícita de ellos, necesitando que cada usuario inicie una sesión en el sitio web con un usuario previamente registrado por él. Este método tiene el claro problema que se necesita la intervención del usuario y para algunos tipos de sitios web es impracticable. Muy parecido al último método resulta la utilización del campo Usuario (authuser) de los archivos log, donde se almacena el nombre del usuario que inició sesión mediante la autenticación de http.

4.2.4. Identificación de sesiones de usuario

Luego de la identificación de los usuarios se deben identificar las sesiones de los mismos que son una serie de servicios solicitados por un mismo usuario a una única visita al sitio o portal web. Las sesiones son un factor importante ya que través de ellas se puede conocer la percepción del usuario con respecto a su visita al portal.

Para ello se necesita dividir las distintas peticiones realizadas por un mismo usuario en una o más sesiones. Debido a que las peticiones a los recursos de otros servidores web no están disponibles, es difícil saber cuándo un usuario abandona el sitio web.

Para la formación de sesiones se utiliza generalmente un tiempo máximo entre sucesivas peticiones, de modo que, si dos peticiones consecutivas de un usuario se realizan con un intervalo de tiempo menor al máximo, las dos peticiones son consideradas como parte de la misma sesión. Si dos peticiones consecutivas se realizan con un intervalo de tiempo mayor al máximo, las dos peticiones corresponden a sesiones distintas; la primera es la última petición de una sesión y la otra es la primera de una nueva sesión.

Se debe seleccionar un tiempo máximo entre peticiones para lo cual se han realizado investigaciones que buscan encontrar el valor que mejor divida las sesiones de los usuarios.



El trabajo de identificación de sesiones es especialmente difícil, debido a algunos inconvenientes como:

- **El orden en que los registros se encuentran almacenados:** Es incierta la secuencia en que los registros son consignados en el log, debido a múltiples factores, entre los que se incluyen la concurrencia del tráfico en la red y la reciente popularización de herramientas de navegación que permiten hacer múltiples peticiones web al servidor originadas por el mismo usuario.
- **La finalización de la sesión:** Normalmente, se asumiría que una sesión podría tomarse como la permanencia de un IP o usuario autenticado en el sistema, sin embargo, esta consideración dista de estar acertada, debido a situaciones en las que, por ejemplo, varios usuarios accedan al servicio web utilizando el mismo nombre de identificación de sesión en el sistema, o en las que algún programa envíe peticiones a la red automáticamente, la decisión de cómo poner fin a una sesión es algo de mucho cuidado.

Dentro de las estrategias utilizadas para identificación de sesiones de usuario se tienen:

- **Basadas en un umbral de tiempo:** Con esta aproximación, se fija un valor que determina que tanto tiempo dura una sesión. Superado este límite, se asume que la sesión ha expirado y se crea una nueva. Esta alternativa sufre del problema de terminar con sesiones de usuarios que probablemente tengan hábitos de navegación muchos mayores a ese valor. Sin embargo, tiene la ventaja de que tiene un alto desempeño.
- **Basadas en la estructura del sitio web:** Esta estrategia abstrae un sitio web específico como un grafo dirigido, donde los nodos son las páginas web y los arcos los vínculos que existen entre ellos; luego, se extrae del log del servidor web las entradas de un cliente y se van comparando con las de la estructura de la página web, para ver si coinciden, con lo cual se va asignando ese registro a la sesión. Las desventajas de esta aproximación radican en el comportamiento que el usuario tenga en la web: si presiona los botones del navegador (atrás,



adelante, recargar), o si navega con múltiples pestañas, la estructura del sitio se modifica y el algoritmo asumiría una nueva sesión.

- **Basadas en cookies:** esta estrategia es idéntica a la máquina con cookies.³

4.3. REDES NEURONALES ARTIFICIALES (RNA)

4.3.1. Definición

Las Redes Neuronales Artificiales (RNA) son modelos que intentan reproducir el comportamiento del cerebro humano. Como tal modelo, realiza una simplificación, averiguando cuáles son los elementos relevantes del sistema, bien porque la cantidad de información de que se dispone es excesiva o bien porque es redundante.⁴

También se puede definir la red neuronal artificial como una implementación, en hardware o software, de un sistema de procesamiento de datos que intenta emular las funciones computacionales elementales de la red nerviosa del cerebro humano. Su propiedad esencial es que implementan un nuevo paradigma de computación muy útil en problemas que no se adecuan bien a las estructuras convencionales de cálculo. Mediante este procedimiento las redes neuronales pueden extraer información estructural de masas de datos complicados o imprecisos, que sintetizan una descripción del fenómeno que ha generado estos datos, es decir, simulan las funciones de un experto en el tema, capaz de enfrentarse con cierto grado de eficacia a situaciones nuevas.⁵

4.3.2. Elementos de una RNA

Todas las RNA tienen elementos en común que son los siguientes:

- **Unidad de proceso:** Existen tres tipos de unidades en cualquier sistema; entradas, salidas y ocultas. Las unidades de entrada reciben señales desde el

³ P. Batista and M. J. Silva, Mining on-line newspaper web access logs. 12th International Meeting of the Euro Working Group on Decision Support Systems (EWG-DSS 2001), 2001, fecha consulta: 14-04-09

⁴ <http://personales.upv.es/ccarrasc/RedesNeuronalesExcaliburNativNoverges/redes.htm>, 2000, fecha consulta: 14-04-09

⁵ <http://monografias.com/redneur>, Redes Neuronales Artificiales, 2001, fecha consulta: 14-04-09



entorno; las de salida envían la señal fuera de la red y las unidades ocultas son aquellas cuyas entradas y salidas se encuentran dentro del sistema. Se conoce como capa o nivel a un conjunto de neuronas cuyas entradas provienen de la misma fuente y cuyas salidas se dirigen al mismo destino.

- **Estado de Activación:** Los valores de activación pueden ser continuos o discretos, limitados o ilimitados. Si son discretos, suelen tomar un conjunto discreto de valores binarios, así un estado activo se indicaría con un 1 y un estado pasivo se representaría por un cero.
- **Función de Salida o de Transferencia:** Asociada con cada unidad hay una función de salida, que transforma el estado actual de activación en una señal de salida. Existen cuatro funciones de transferencia típicas que determinan distintos tipos de neuronas:
 - Función Escalón
 - Función Lineal y Mixta
 - Función Sigmoidal
 - Función Gaussiana.
- **Conexiones entre neuronas:** Las conexiones que unen a las neuronas que forman una RNA tiene asociado un peso, que es el que hace que la red adquiera conocimiento. Se considera que el efecto de cada señal es aditivo, de tal forma que la entrada neta que recibe una neurona es la suma del producto de cada señal individual por el valor de la sinapsis que conecta ambas neuronas y es lo que se conoce como red de propagación.
- **Función o Regla de Activación:** Se requiere una regla que combine las entradas con el estado actual de la neurona para producir un nuevo estado de activación. Esta función F produce un nuevo estado de activación en una neurona a partir del estado que existía y la combinación de las entradas con los pesos de las conexiones. Esa F es denominada función de activación y las salidas que se obtienen en una neurona para las diferentes formas de F serán:



- Función de Activación Escalón
 - Función de Activación Identidad
 - Función de Activación Lineal - Mixta
 - Función de Activación Sigmoidal
-
- **Regla de Aprendizaje:** El aprendizaje puede ser comprendido como la modificación de comportamiento inducido por la interacción con el entorno y como resultado de experiencias conduce al establecimiento de nuevos modelos de respuesta a estímulos externos. En el cerebro humano el conocimiento se encuentra en la sinapsis. En el caso de las RNA el conocimiento se encuentra en los pesos de las conexiones entre neuronas. Todo proceso de aprendizaje implica cierto número de cambios en estas conexiones. En realidad puede decirse que se aprende modificando los valores de los pesos de la red.⁶

4.3.3. Tipos de RNA

Existen diversos tipos de redes neuronales artificiales:

- **Redes neuronales supervisadas:** Pueden emplearse como clasificadores de patrones, estimadores de funciones multi-variadas o memorias asociativas.
- **Redes auto-organizadas o no supervisadas:** En su entrenamiento no presentan las salidas objetivo que se desean asociar a cada patrón de entrada, será la red la que proporcione cierto resultado. La principal aplicación es la realización de agrupamiento de patrones (clustering), visualización de datos y representación de densidades de probabilidad.
- **Redes neuronales realimentadas:** Son más complicadas que las dos anteriores. En este caso la información se propaga tanto hacia delante como hacia atrás, comportándose como un sistema dinámico de difícil análisis y en el que deberá garantizarse la estabilidad de su respuesta.⁷

⁶<http://monografias.com/redneur>, Redes Neuronales Artificiales, 2001, fecha consulta: 14-04-09

⁷<http://personales.upv.es/ccarrasc/RedesNeuronalesExcaliburNativNoverges/redes.htm>, 2000, fecha consulta: 14-04-09



4.3.4. Características de las RNA

Existen cuatro aspectos que caracterizan una red neuronal artificial:

1. **Topología:** La arquitectura de las redes neuronales consiste en la organización y disposición de las neuronas formando capas más o menos alejadas de la entrada y salida de la red. En este sentido, los parámetros fundamentales de la red son: el número de capas, el número de neuronas por capa, el grado de conectividad y el tipo de conexiones entre neuronas.
 - **Redes Monocapa:** Se establecen conexiones laterales, cruzadas o autorrecurrentes entre las neuronas que pertenecen a la única capa que constituye la red. Se utilizan en tareas relacionadas con lo que se conoce como autoasociación; por ejemplo, para generar informaciones de entrada que se presentan a la red incompleta o distorsionada.
 - **Redes Multicapa:** Son aquellas que disponen de conjuntos de neuronas agrupadas en varios niveles o capas. Una forma de distinguir la capa a la que pertenece la neurona, consiste en fijarse en el origen de las señales que recibe a la entrada y el destino de la señal de salida.
2. **Mecanismo de Aprendizaje:** El aprendizaje es el proceso por el cual una red neuronal modifica sus pesos en respuesta a una información de entrada. Los cambios que se producen durante el proceso de aprendizaje se reducen a la destrucción, modificación y creación de conexiones entre las neuronas, la creación de una nueva conexión implica que el peso de la misma pasa a tener un valor distinto de cero, una conexión se destruye cuando su peso pasa a ser cero. Se puede afirmar que el proceso de aprendizaje ha finalizado cuando los valores de los pesos permanecen estables. Existen redes neuronales con **aprendizaje supervisado y redes neuronales con aprendizaje no supervisado:** La diferencia fundamental entre ambos tipos estriba en la existencia o no de un agente externo (supervisor) que controle el aprendizaje de la red.



3. **Tipo de Asociación entre las Informaciones de Entrada y Salida:** Las RNA son sistemas que almacenan cierta información aprendida; esta información se registra de forma distribuida en los pesos asociados a las conexiones entre neuronas de entrada y salida. Existen dos formas primarias de realizar esa asociación de entrada/salida; heteroasociación y autoasociación.
4. **Representación de la Información de Entrada y Salida:** En un gran número de redes, tanto los datos de entrada como de salida son de naturaleza analógica, cuando esto ocurre, las funciones de activación de las neuronas serán también continuas del tipo lineal o sigmoideal. Otras redes sólo admiten valores discretos o binarios a su entrada generando también unas respuestas en la salida de tipo binario. En este caso, las funciones de activación de las neuronas son de tipo escalón. Existe también un tipo de redes híbridas en las que las informaciones de entrada pueden ser valores continuos, aunque las salidas de la red son discretas.⁸

4.3.5. Self Organizing Maps (SOM)

Los mapas autoorganizados o *SOM (Self Organizing Map)*⁹ son un modelo de red neuronal que es capaz de proyectar datos de entrada de grandes dimensiones sobre un *espacio* de dos dimensiones. Esta proyección da como resultado un mapa que puede ser de utilidad para la detección y el análisis de características del espacio de los datos de entrada.

La principal razón de la popularidad del SOM es su capacidad de presentar de manera automática un mapa en el cual se puede observar una descripción intuitiva de la similitud entre los datos; el despliegue bidimensional tiene la propiedad de presentar la información contenida en los datos de manera ordenada y resaltando las relaciones mencionadas.

Dos son los tipos fundamentales de SOMs en función del tipo de entrenamiento que reciben:

⁸<http://monografias.com/redneur. Redes Neuronales Artificiales. 2001. fecha consulta: 14-04-09>

⁹T. Kohonen, Self-OrganizingMaps, 2001, fecha consulta: 08-09-09



- a) Clásica, también llamada En Línea (*On-Line*), y
- b) Por Lotes (*Batch*).

Si bien la convergencia, o velocidad de convergencia, a la solución de las redes *Batch* puede ser menos efectiva que en las *On-Line*, son las más eficientes cuando se usa el paralelismo.

Las principales propiedades que definen a un mapa auto-organizativo (SOM) son:

- **Aproximación del espacio de entrada:** Los mapas de características representados por el conjunto de vectores de pesos dan una buena aproximación al espacio de entrada.
- **Ordenación topológica:** La localización espacial de una neurona corresponde a un dominio particular o conjunto de patrones de entrada.
- **Densidad de Muestreo:** Reflejan variaciones en las estadísticas de las distribuciones de entrada.
- **Selección automática de la dimensionalidad de las características:** La distribución de los vectores de pesos tienden a aproximar una superficie suavizada.

En cuanto a la filosofía manejada en este tipo de redes se pueden comentar algunas de las características que la definen:

- Pesos: Se sigue la misma filosofía que en el aprendizaje supervisado.
- Región de interés: Los nodos pertenecientes a la región de interés del Peso ganador también son modificados. De esta forma se consigue que la red responda a valores muy próximos del vector de entrada. Esto demuestra la generalidad de esta red.
- La región de interés se reduce de manera lineal.¹⁰

¹⁰IA2007-Grupo01-TP, Arquitectura Redes de Kohonen, 2007, fecha consulta: 16-04-09



4.3.5.1. Arquitectura de la Red SOM

Las redes SOM poseen una arquitectura de dos capas (entrada-salida), funciones de activación lineales y flujo de información unidireccional (son redes en cascada). El objetivo de este tipo de redes es clasificar los patrones de entrada en grupos de características similares de manera que cada grupo activará siempre la(s) misma(s) salida(s). Cada grupo de entradas queda representado en los pesos de las conexiones de la unidad de salida triunfante. La unidad de salida ganadora para cada grupo de entradas no se conoce previamente, es necesario averiguarlo después de entrenar a la red.

El mapa auto-organizado trata de establecer una correspondencia entre los datos de entrada y un espacio bidimensional de salida, creando mapas topológicos de dos dimensiones de tal forma que ante datos de entrada con características comunes se deben activar neuronas situadas en zonas próximas de la capa de salida. Por esta razón, la representación habitual de esta red suele ser la mostrada en la Figura 1, donde las M neuronas de salida se disponen en forma bidimensional para representar los mapas de características.

La interacción lateral entre las neuronas de la capa de salida sigue existiendo aunque ahora hay que entender la distancia como una zona bidimensional que existe alrededor de cada neurona. Esta zona puede tomar la forma de cualquier polígono regular centrado en dicha neurona.

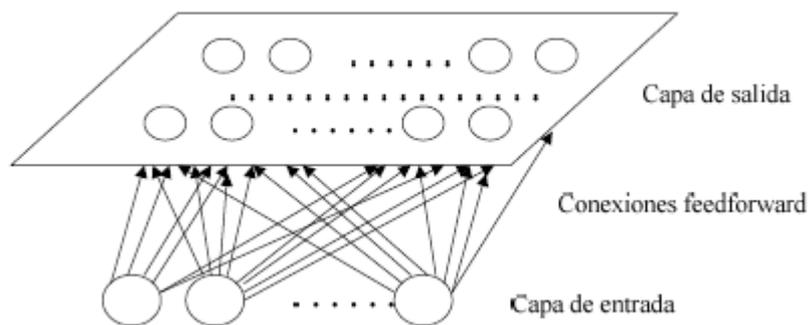
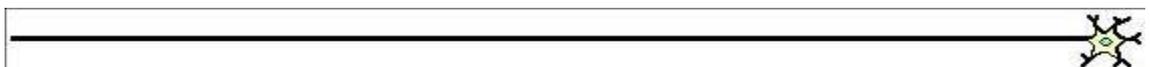


Figura 1: Arquitectura de mapa auto-organizado



La influencia que una neurona ejerce sobre las demás es función de la distancia entre ellas, siendo muy pequeñas cuando están muy alejadas. En el algoritmo básico del SOM, las relaciones topológicas y el número de neuronas son fijos desde el comienzo; este número de neuronas determina la escala o la granularidad del modelo resultante. La selección de la granularidad afecta la certeza y la capacidad de generalizar del modelo.

Normalmente, la arquitectura de la red tiene las siguientes características:

- Las neuronas se distribuyen a lo largo de una retícula bidimensional.
- Cada neurona constituye a un nodo de la retícula.
- La configuración o tipo de retícula puede ser definida como: rectangular, hexagonal o incluso irregular.
- La localización de la neurona sobre la retícula está representada por su vector de localización $r_i = (p_i, q_i) \in \mathbb{N}^2$.
- Cada neurona es asociada a un vector de pesos $w_i \in \mathbb{R}^n$. En el caso del SOM este vector es también llamado vector de referencia.

En la Figura 2 se muestran las configuraciones o tipo de retícula más usados con los correspondientes $r_i = (p_i, q_i)$ en cada nodo. Cabe señalar que la configuración hexagonal es más conveniente para efectos de visualización.

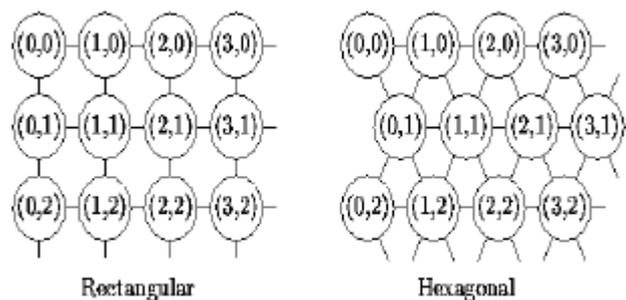


Figura 2: Configuraciones más comunes en la retícula del SOM.



4.3.5.2. Funcionamiento

El funcionamiento del mapa auto-organizado es relativamente simple. Cuando se

presenta a la entrada una información $\underline{E}_k = (e_1^{(k)}, \dots, e_N^{(k)})$ cada una de las M neuronas de la capa de salida la recibe a través de las conexiones feedforward con pesos \underline{W}_{ji} .

Estas neuronas también reciben las correspondientes entradas por medio de las conexiones laterales con el resto de las neuronas de salida y cuya influencia dependerá de la distancia a la que se encuentren.

Así, la salida generada por una neurona de salida j ante un vector de entrada \underline{E}_k

$$S_j(t+1) = f\left(\sum_{i=1}^N w_{ij} e_i^{(k)} + \sum_{p=1}^M Int_{pj} S_p(t)\right)$$

Donde Int_{pj} es una función del tipo sombrero mejicano que representa la influencia lateral de la neurona p sobre la neurona j. La función de activación de las neuronas de salida (f) será del tipo continuo, lineal o sigmoideal ya que esta red trabaja con valores reales.

La red SOM realiza una tarea de clasificación ya que la neurona de salida activada ante una entrada representa la clase a la que pertenece dicha información.

Además, como ante otra entrada parecida se activa la misma neurona o una cercana a la anterior, se garantiza que las neuronas topológicamente cercanas sean sensibles a entradas físicamente similares.

Un concepto muy importante en las redes SOM es la zona de vecindad o vecindario alrededor de la neurona vencedora i^* , los pesos de las neuronas que se encuentren en



esta zona a la que se le dará el nombre de $X(q)$, serán actualizados junto con el peso de la neurona ganadora, en un ejemplo de aprendizaje cooperativo.¹¹

4.3.5.3. Pre-procesamiento de los datos

Los datos que alimentan al SOM incluyen toda la información que toma la red. Si se le presenta información errónea, el resultado es erróneo o de mala calidad. Entonces, el SOM, tanto como los otros modelos de redes neuronales, deben eliminar la información “basura” para que no ingrese al sistema. Por lo cual se debe trabajar con un subconjunto de los datos; estos deben ser relevantes para el modelo a analizar. También se deben eliminar los errores en los datos; si los mismos se obtienen a través de una consulta a una base de datos, el resultado puede incluir datos erróneos debido a la falta de integridad de la base; entonces estos deben ser filtrados usando conocimientos previos del dominio del problema y el sentido común.

Comúnmente los componentes de los datos de entrada se normalizan para tener una escala de 0 a 1. Esto asegura que por cada componente, la diferencia entre dos muestras contribuye un valor igual a la distancia medida calculada entre una muestra de entrada y un patrón. Es decir que los datos deben previamente codificarse (normalizarse). De lo contrario no será posible usar la distancia como una medida de similitud. Esta medida debe ser cuantificable por lo que la codificación debe ser armónica con la medida de similitud utilizada. La medida mayormente utilizada es la distancia Euclídea. Los datos simbólicos no pueden ser procesados por un SOM como tales, por lo que deben ser transformados a una codificación adecuada.

4.3.5.4. Aprendizaje

El aprendizaje en las redes SOM es de tipo Off-line, por lo que se distingue una etapa de aprendizaje y otra de funcionamiento. En la etapa de aprendizaje se fijan los valores de las conexiones (feedforward) entre la capa de entrada y la salida. Esta red utiliza un aprendizaje no supervisado de tipo competitivo, las neuronas de la capa de salida compiten por activarse y sólo una de ellas permanece activa ante una determinada información de entrada a la red, los pesos de las conexiones se ajustan en función de la neurona que haya resultado vencedora.

¹¹IA2007-Grupo01-TP, Funcionamiento Redes de Kohonen, 2007, fecha consulta: 16-04-09



Durante la etapa de entrenamiento se presenta a la red un conjunto de informaciones de entrada (vectores de entrenamiento) para que ésta establezca en función de la semejanza entre los datos las diferentes categorías (una por neurona de salida) que servirían durante la fase de funcionamiento para realizar clasificaciones de nuevos datos que se presenten a la red. Los valores finales de los pesos de las conexiones entre cada neurona de la capa de salida con las de entrada se corresponderán con los valores de los componentes del vector de aprendizaje que consigue activar la neurona correspondiente. En el caso de existir más patrones de entrenamiento que neuronas de salida, más de uno deberá asociarse con la misma neurona, es decir pertenecerán a la misma clase.

En este modelo el aprendizaje no concluye después de presentarle una vez todos los patrones de entrada, sino que habrá que repetir el proceso varias veces para refinar el mapa topológico de salida, de tal forma que cuantas más veces se presenten los datos, tanto más se reducirán las zonas de neuronas que se deben activar ante entradas parecidas, consiguiendo que la red pueda realizar una clasificación más selectiva.

El algoritmo de aprendizaje de la red SOM, puede definirse como “un algoritmo para la visualización de datos multidimensionales que implementa un mapeo ordenado de una distribución multidimensional en una malla regular de menor dimensión, que usualmente consiste de una malla de dos dimensiones”.¹²

El algoritmo de aprendizaje utilizado para establecer los valores de los pesos de las conexiones entre las N neuronas de entrada y las M de salida es el siguiente:

1. En primer lugar se inicializan los pesos (w_{ij}) con valores aleatorios pequeños y se fija la zona inicial de vecindad entre las neuronas de salida.
2. A continuación se presenta a la red una información de entrada (la que debe aprender) en forma de vector $p = (p_1, p_2, \dots, p_n)$, cuyas componentes p_i serán valores continuos.

¹²Kohonen, Aprendizaje Redes SOM, 1995, fecha consulta: 05-09-09



3. Puesto que se trata de un aprendizaje competitivo, se determina la neurona vencedora de la capa de salida, esta será aquella i cuyo vector de pesos w_i sea el más parecido a la información de entrada p (patrón o vector de entrada). Para ello se calculan las distancias o diferencias entre ambos vectores, considerando una por una todas las neuronas de salida, suele utilizarse la distancia euclídea o la siguiente expresión que es similar a aquella, pero eliminando la raíz cuadrada:

$$d_i = \sum_{j=1}^N (p_j - w_{ij})^2 \quad 1 \leq i \leq M$$

p_j : Componente i -ésimo del vector de entrada w_{ij} : Peso de la conexión entre la neurona j de la capa de entrada y la neurona i de la capa de salida.

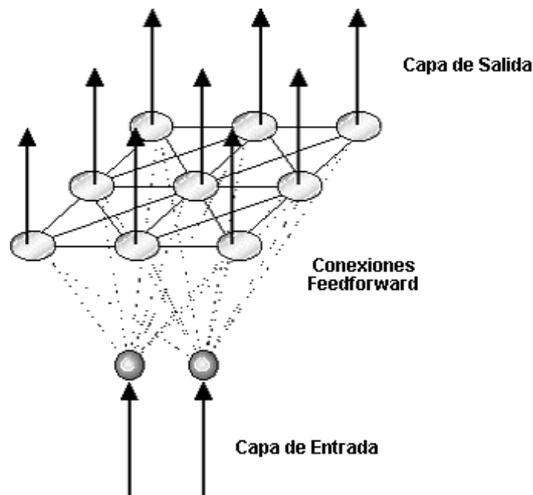


Figura 3: Conexiones de una red SOM

4. Una vez localizada la neurona vencedora (i^*), se actualizan los pesos de las conexiones entre las neuronas de entrada y dicha neurona, así como los de las conexiones entre las de entrada y las neuronas vecinas de la vencedora, en realidad lo que se consigue con esto es asociar la información de entrada con una cierta zona de la capa de salida. Esto se realiza mediante la siguiente ecuación:

$$w_i(q) = w_i(q-1) + a(q)(p(q) - w_i(q-1)) \quad \text{para } i \in X(q)$$



El tamaño de $X(q)$ se puede reducir en cada iteración del proceso de ajuste de los pesos, con lo que el conjunto de neuronas que pueden considerarse vecinas cada vez es menor como se observa en la Figura 4, sin embargo en la práctica es habitual considerar una zona fija en todo el proceso de entrenamiento de la red.

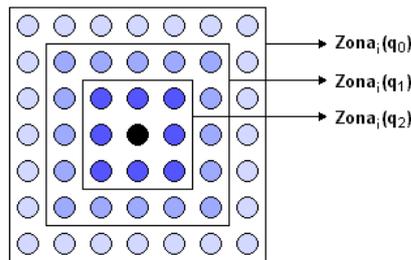


Figura 4: Posible evolución de la vecindad en una red SOM.

El término $\alpha(q)$ es el coeficiente de aprendizaje o parámetro de ganancia, con un valor entre 0 y 1 el cual decrece con el número de iteraciones (q) del proceso de entrenamiento, de tal forma que cuando se ha presentado un gran número de veces todo el juego de patrones de aprendizaje su valor es prácticamente nulo, con lo que la modificación de los pesos es insignificante.

5. El proceso debe repetirse, volviendo a presentar todo el juego de patrones de aprendizaje p_1, p_2, \dots, p_n hasta obtener la salida deseada.

En definitiva lo que hace una red SOM es realizar una tarea de clasificación, puesto que la neurona de salida activada ante una entrada representa la clase a la que pertenece dicha información de entrada, además ante otra entrada parecida se activa la misma neurona de salida u otra cercana a la anterior debido a la semejanza entre las clases, así se garantiza que las neuronas topológicamente próximas sean sensibles a entradas físicamente similares; por esta causa la red es especialmente útil para establecer relaciones desconocidas previamente entre conjuntos de datos.¹³

¹³<http://ohm.utp.edu.co/neuronales/Capitulo2/Competitivas/>, 2008, fecha consulta: 20-12-09



4.4. EXPLOTACIÓN DE DATOS (DATA MINING)

4.4.1. Generalidades

La minería de datos o Data Mining es un mecanismo de explotación, consistente en la búsqueda de información valiosa en grandes volúmenes de datos, con el fin de descubrir patrones, relaciones, reglas, asociaciones o incluso excepciones útiles para la toma de decisiones.

Lo que hace el Data Mining es reunir las ventajas de varias áreas como la Estadística, la Inteligencia Artificial, la Computación Gráfica, las Bases de Datos y el Procesamiento Masivo, principalmente usando como materia prima las bases de datos. Una definición tradicional es la siguiente: Un proceso no trivial de identificación válida, novedosa, potencialmente útil y entendible de patrones comprensibles que se encuentran ocultos en los datos¹⁴.

La Minería de Datos es un proceso completo de descubrimiento de conocimiento que involucra varios pasos¹⁵:

1. Entendimiento del dominio de aplicación, el conocimiento relevante a utilizar y las metas del usuario.
2. Seleccionar un conjunto de datos en donde realizar el proceso de descubrimiento.
3. Limpieza y pre-procesamiento de los datos, diseñando una estrategia adecuada para manejar ruido, valores incompletos, valores fuera de rango, valores inconsistentes, etc.
4. Selección de la tarea de descubrimiento a realizar, por ejemplo, clasificación, agrupamiento o clustering, reglas de asociación, etc.
5. Selección de los algoritmos a utilizar.
6. Transformación de los datos al formato requerido por el algoritmo específico de explotación de datos, hallando los atributos útiles, reduciendo las dimensiones de los datos, etc.

¹⁴Fayyad y otros. Data Mining, 1996. fecha consulta: 22-12-09

¹⁵Morales, Minería de datos, 2003, fecha consulta: 22-12-09



7. Llevar a cabo el proceso de minería de datos para encontrar patrones interesantes.
8. Evaluación de los patrones descubiertos y presentación de los mismos mediante técnicas de visualización. Quizás sea necesario eliminar patrones redundantes o no interesantes o se necesite repetir algún paso anterior con otros datos, con otros algoritmos, con otras metas o con otras estrategias.
9. Utilización del conocimiento descubierto ya sea incorporándolo dentro de un sistema o simplemente para almacenarlo y reportarlo a las personas interesadas.

Debido a que la minería de datos es un proceso que involucra varios pasos, enumerados anteriormente, se necesita un modelo que sirva de guía para la correcta ejecución de los mismos. La creación de un modelo de minería de datos es un proceso dinámico e iterativo, por esto, es importante comprender que la creación de un modelo de minería de datos es un proceso y que cada paso del proceso puede repetirse tantas veces como sea necesario para crear un modelo válido.

Un concepto importante reside en que la generación de un modelo de minería de datos forma parte de un proceso mayor que incluye desde la definición del problema básico que resolverá el modelo hasta la implementación del modelo en un entorno de trabajo. Este proceso se puede definir mediante los seis pasos básicos siguientes:¹⁶

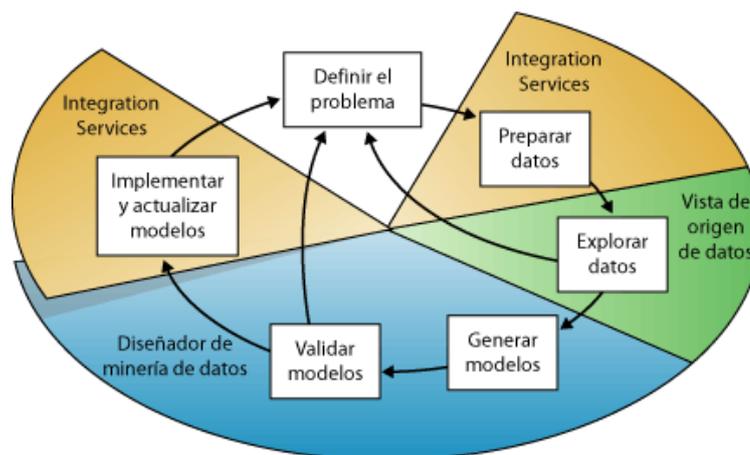


Figura 5: Modelo básico de la Minería de Datos

¹⁶<http://technet.microsoft.com/es-es/library/ms174949.aspx>, 2007, fecha consulta: 03-01-10



- **Definir el problema:** Consiste en definir claramente el problema empresarial. Este paso incluye analizar los requisitos empresariales, definir el ámbito del problema, definir las métricas por las que se evaluará el modelo y definir el objetivo final del proyecto de minería de datos.
- **Preparar los datos:** Consiste en consolidar y limpiar los datos identificados en el paso anterior.
- **Explorar los datos:** En esta fase se explora los datos preparados para tomar las decisiones adecuadas al crear los modelos. Entre las técnicas de exploración se incluyen calcular los valores mínimos y máximos, calcular la media y las desviaciones estándar y examinar la distribución de los datos.
- **Generar modelos:** Antes de generar un modelo se deben separar aleatoriamente los datos preparados en conjuntos de datos de entrenamiento y comprobación independientes. Una vez definida la estructura del modelo de minería de datos, la procesa relleno la estructura vacía con los patrones que describen el modelo. Esto se conoce como *entrenar* el modelo. El modelo de minería de datos se define mediante un objeto de estructura de minería de datos, un objeto de modelo de minería de datos y un algoritmo de minería de datos.
- **Explorar y validar los modelos:** Consiste en explorar los modelos que se ha generado y comprobar su eficacia. No se debe implementar un modelo en un entorno de producción sin comprobar primero si el modelo funciona correctamente.
- **Implementar y actualizar los modelos:** El último paso del proceso de minería de datos consiste en implementar los modelos que funcionan mejor en un entorno de producción. La actualización del modelo forma parte de la estrategia de implementación. A medida que la organización recibe más datos debe volver a procesar los modelos para mejorar así su eficacia.



Existen varios tipos de técnicas de explotación de datos, las cuales en función de su objetivo pueden clasificarse en:

- **Descubrimiento de Reglas de Asociación:** consiste en hallar patrones, asociaciones, correlaciones o estructuras causales frecuentes entre conjuntos de objetos almacenados en un repositorio de información.
- **Clasificación de Datos:** proceso que encuentra las propiedades comunes entre un conjunto de objetos y los clasifica en diferentes clases, de acuerdo a un modelo de clasificación.
- **Agrupamiento de Datos:** consiste en agrupar un conjunto de datos, sin tener clases predefinidas, basándose en la similitud de los valores de los atributos de los distintos datos.

4.4.2. Explotación de datos de uso de la web

El denominado Web Usage Mining (WUM), extrae información de datos secundarios generados por los usuarios en su interacción por un sitio web (navegación). WUM incluye datos provenientes de accesos a servidores web registrados en archivos de tipo log, logs de servidores proxy, logs de motores de búsqueda, perfiles de usuario, archivos de enrolamiento o registro, sesiones de usuarios y transacciones, consultas de usuarios, carpetas marcadas, clic del ratón, desplazamientos (scroll) y otros tipos de información.

La explotación de datos de uso de la web está orientada a descubrir el uso de las páginas web, direcciones IP, referencias a páginas, la fecha y la hora de accesos, datos que proporcionan la información demográfica sobre los usuarios del sitio web. Los servidores web guardan datos de las interacciones de sus usuarios en todo pedido recibido de algún recurso. Estos datos se almacenan en los log de acceso del servidor web, conteniendo información sobre la experiencia del usuario en el sitio. El análisis de los log de acceso de un servidor web puede ayudar a entender el comportamiento de sus usuarios.



Con la identificación de hábitos de uso de sitios web se busca:¹⁷

- Entender el comportamiento de navegación del usuario, permitiendo adaptar los sitios web a sus necesidades.
- Obtener la información para la personalización de los sitios
- Realizar mejoras en el sistema
- Modificar el sitio acorde a los patrones descubiertos
- Realizar inteligencia del negocio
- Ofrecer contenido a los visitantes de acuerdo a sus preferencias y
- Caracterizar el uso del sitio web por los usuarios.

Mediante estas acciones se busca:¹⁸

- a) Atraer nuevos clientes
- b) Retener a los clientes actuales
- c) Realizar campañas de promociones efectivas
- d) Determinar estrategias de marketing efectivas y
- e) Encontrar la mejor estructura lógica del espacio web.

La identificación de hábitos de uso de sitios web consiste de tres etapas:¹⁹

1. Pre-procesamiento
2. Descubrimiento de patrones y
3. Análisis de patrones.

¹⁷Borges, Levene et al., Web sites, 2000, fecha consulta: 12-01-10

¹⁸Abraham & Ramos, Creación de sitios web, 2003, fecha consulta: 12-01-10

¹⁹Srivastava et al., Etapas de la identificación de hábitos de uso en sitios web, 2000, fecha consulta: 12-01-10



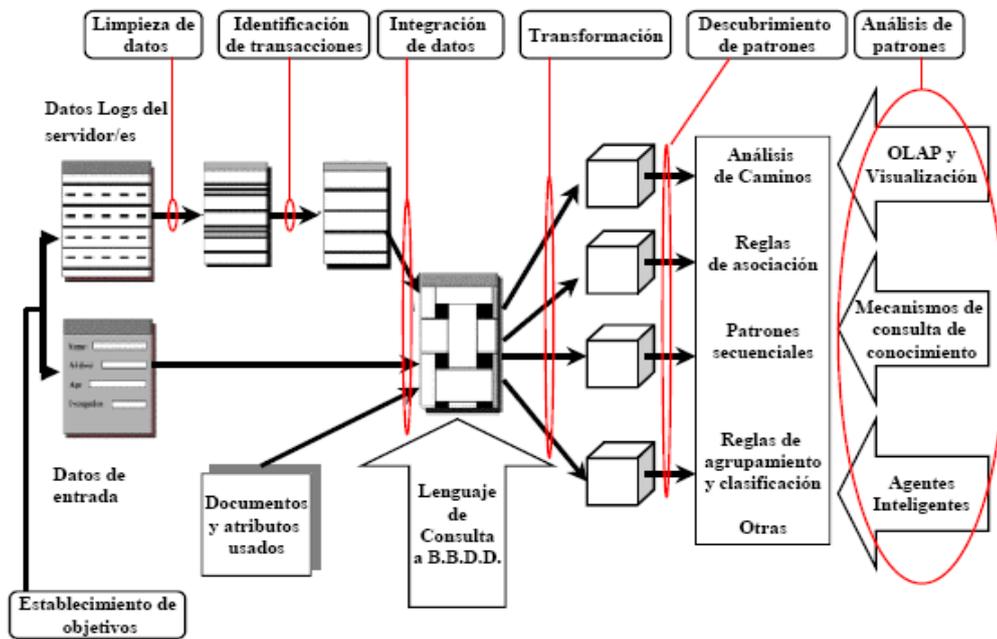


Figura 6: Web Usage Mining

4.4.2.1. Pre-procesamiento

El pre-procesamiento consiste en convertir la información de uso contenida en los log, realizando previamente una limpieza y selección de los mismos, en una abstracción de datos necesaria para el descubrimiento de patrones. En esta etapa se identifican las sesiones y transacciones de usuario.²⁰

La **limpieza** de los datos almacenados en los archivos log involucra el tratamiento de los valores fuera de rango, los errores en los datos y los datos incompletos que pueden ocurrir fácilmente debido a varias razones inherentes a la navegación web. Los campos más comunes de un archivo log de un servidor web incluyen: dirección IP del cliente, hora de acceso, método de petición, URL de la página accedida, protocolo de transmisión de datos, código de retorno, cantidad de bytes transmitidos por el servidor al cliente y el tipo de navegador utilizado.

²⁰Srivastava et al., Etapas de la identificación de hábitos de uso en sitios web, 2000, fecha consulta: 16-01-10



Una **sesión de usuario** está formada por todas las páginas consultadas por un usuario durante una sola visita al sitio, es una secuencia de vínculos (links) seguidos por el usuario.²¹

Los datos pueden ser recolectados en el lado del servidor, lado del cliente, servidores proxy u obtenidos de una base de datos. Para cada recolección de datos, la diferencia no solamente radica en la localización, sino también, en el tipo de datos disponible, el segmento de la población desde la cual los datos fueron recolectados y el método de implementación. Las fuentes de información de la minería pueden incluir logs de utilización de la web, descripción de páginas web, topología de sitios web, registros de usuario y cuestionarios.

Pre-procesamiento de contenido: Es el proceso de convertir texto, imágenes, scripts y otros archivos en formas que puedan ser usadas por el WUM. El contenido web puede ser usado para filtrar la entrada o la salida del algoritmo de descubrimiento de patrones.²² Para el desarrollo del pre-procesamiento, el contenido de cada página visitada debe ser ensamblado sea por un agente crawlero a través de una combinación de template, script y accesos a bases de datos. Baeza²³ define agentes crawlers (rastreadores), como agentes de software que recorren la web enviando las páginas nuevas o actualizadas para mantener un servidor donde ellas están indexadas.

Pre-procesamiento de estructuras: La estructura de un sitio web está formada por los hyperlinks entre las páginas visitadas. El pre-procesamiento de estructuras puede ser tratado de forma similar al pre-procesamiento de contenido. Sin embargo, cada sesión del servidor puede construir una estructura de sitios diferente cada vez.

Pre-procesamiento de utilización: Las entradas de esta fase de pre-procesamiento pueden incluir los log de servidores web, archivos log remitidos, registros de archivos, archivos log del índice del servidor y opcionalmente estadísticas de análisis previos. Las salidas son los archivos de las sesiones de usuario, transacciones de archivos,

²¹ Batista, Silva et al., Archivos Log, 2002, fecha consulta: 02-02-10

²² COOLEY, Robert. TAN, Pang-Ning. SRIVASTAVA, Jaideep, Discovery of interesting usage patterns from web data. M. Spiliopoulou and B. Masand editors, Advances in web usage analysis and user profiling, Lecture Notes in computer science, 2000, fecha consulta: 03-02-10

²³ BAEZA-YATES, Ricardo. RIBEIRO-NETO, Berthier. Modern Information Retrieval. Addison Wesley, 1999, fecha consulta: 03-02-10.



topología del sitio y clasificación de páginas. Es necesario adoptar siempre unas técnicas de datos limpios para eliminar el impacto de los ítems irrelevantes para el análisis de resultados. El pre-procesamiento de utilización es probablemente el que presenta mayores dificultades en las tareas del proceso de WUM debido a la falta de completitud de los datos.

4.4.2.2. Descubrimiento de patrones

Una vez identificadas las sesiones de usuario, es posible comenzar a buscar patrones de acceso y comportamiento de los usuarios de la web, dependiendo del problema a resolver y de los datos disponibles serán más adecuadas unas técnicas que otras.

Así pues, las técnicas que más se emplean para realizar minería de uso de la web son: el análisis estadístico, el descubrimiento de reglas de asociación, el agrupamiento y clasificación, análisis de caminos y los patrones secuenciales.²⁴

- **Análisis Estadístico:** Las herramientas estadísticas son herramientas poderosas en la extracción de conocimiento respecto a los visitantes de un sitio web. Los analistas pueden desarrollar análisis descriptivo basado en las diferentes variables cuando analizan los archivos de las sesiones. Es de gran utilidad analizar estadísticamente la información contenida en los reportes periódicos de los sistemas web. Ellos pueden mejorar el desempeño del sistema, ampliando su seguridad, facilitando de esta manera las tareas de modificación del sitio, y suministrando soporte para la toma de decisiones en las organizaciones. Sin embargo, los datos estadísticos disponibles solo pueden proveer la información explícita debido a la naturaleza y las limitaciones de la metodología.
- **Reglas de asociación:** Las reglas de asociación sirven en este caso para encontrar patrones, asociaciones y correlaciones frecuentes entre conjuntos de ítems. Las reglas de asociación son utilizadas para revelar la correlación entre páginas accedidas en forma conjunta durante una única sesión de un usuario. Tales reglas indican la posible relación entre páginas que son a menudo vistas juntas aunque no estén conectadas directamente y puede revelar asociaciones

²⁴Srivastava et al, Técnica de Minería de uso web, 1996, fecha consulta: 21-02-10



entre grupos de usuarios con intereses específicos. Además de poder ser explotadas para cuestiones de negocios, tales observaciones pueden ser utilizadas para una reestructuración del sitio web. A diferencia del análisis estadístico, que es corroborativo, el análisis de asociaciones descubre las relaciones sin que exista intervención alguna por parte del operador.

- **Descubrimiento de patrones secuenciales:** Es una extensión a las reglas de asociación que se centra en localizar la presencia de un conjunto de elementos seguida por otro elemento en un conjunto de transacciones o visitas ordenadas en el tiempo. En el dominio de la web, tales patrones podrían corresponder con una página web o un conjunto de páginas web que son accedidas inmediatamente después de otro conjunto de páginas. Utilizando esta información pueden ser descubiertas tendencias de los usuarios útiles y pueden realizarse predicciones de las visitas de los usuarios.
- **Agrupación y Clasificación:** La agrupación es utilizada para juntar ítems que poseen características similares. En el contexto de la minería de hábitos en web, se pueden distinguir dos casos; agrupamiento de usuarios y agrupamiento de páginas. El agrupamiento de usuarios produce conjuntos de usuarios que poseen un comportamiento similar cuando navegan el sitio web. Una vez descubiertos los prototipos o perfiles de cada grupo, se pueden usar las características de cada uno de ellos para realizar la clasificación. En minería de uso de la web, las técnicas de clasificación permiten desarrollar un perfil para clientes/usuarios que acceden a ficheros concretos del servidor, en función de sus patrones de acceso.
- **Análisis de Caminos:** Esta técnica supone la generación de algunas formas de grafos orientados que representan relaciones entre páginas web. Este grafo puede ser un esquema físico en el que las páginas web son los nodos del grafo y los hiper-enlaces entre las páginas son las flechas dirigidas entre nodos. Pueden formarse otros grafos a partir de los tipos de páginas web, con arcos que representen la similitud entre páginas o creando arcos que muestren el número de usuarios que van desde una página a otra. El análisis de caminos podría utilizarse para determinar los caminos más frecuentemente seguidos en



un sitio web.

4.4.2.3. Análisis de patrones

El análisis de patrones es el paso final del proceso del WUM. El objetivo de este paso es eliminar las reglas o patrones irrelevantes y extraer las reglas o patrones interesantes del descubrimiento de patrones utilizando métodos de visualización útiles para su análisis, realizando proyecciones dinámicas, filtros, zoom y distorsiones interactivas sobre los gráficos generados.²⁵

La salida de los algoritmos de webmining generalmente no se ajusta de forma directa a la concepción humana, es por ello que después se necesita hacer una transformación a un formato que sea fácilmente asimilable.²⁶

4.5. HERRAMIENTAS PARA LA CONSTRUCCIÓN DEL SISTEMA

4.5.1. NetBeans 6.5

4.5.1.1. Generalidades

NetBeans es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación.

El IDE NetBeans es un producto de código abierto y gratuito sin restricciones de uso, escrito completamente en Java usando la plataforma NetBeans. El NetBeans IDE soporta el desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles).

NetBeans IDE 6.5, fue lanzado el 19 de Noviembre del 2008, extiende las características existentes del Java EE (incluyendo Soporte a Persistencia, EJB 3 y JAX-WS). Adicionalmente, el NetBeans Enterprise Pack soporta el desarrollo de Aplicaciones empresariales con Java EE 5, incluyendo herramientas de desarrollo

²⁵Keim, Web Usage Mining, 2002, fecha consulta: 03-03-10

²⁶VILLENA, J. GONZÁLEZ, J. C. BARCELO, E. & VELASCO, J. R. Minería de uso de la Web mediante huellas y sesiones, 2002, fecha consulta: 03-03-10.



visuales de SOA, herramientas de esquemas XML, orientación a web servicios (forBPEL) y modelado UML. El NetBeans C/C++ Pack soporta proyectos de C/C++, mientras el PHP Pack, soporta PHP 5.

4.5.1.2. Estructura

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos.

Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregando nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

4.5.1.3. Características

Entre las características principales de la plataforma están:

- Administración de las interfaces de usuario (ej. menús y barras de herramientas).
- Administración de las configuraciones del usuario.
- Administración del almacenamiento (guardando y cargando cualquier tipo de dato).
- Administración de ventanas.
- Framework basado en asistentes (diálogos paso a paso).
- Mejoras en el editor de código.
- Permite enlazar datos con el Swing GUI.
- Características visuales para el desarrollo web.



4.5.2. Java Server Faces (JSF)

4.5.2.1. Generalidades

JSF es un framework para el desarrollo de aplicaciones web en Java basado en el patrón de arquitectura Modelo-Vista-Controlador (MVC) que se encarga de separar la parte de datos de la aplicación, de la interfaz de usuario y la lógica de control mediante tres componentes diferentes²⁷:

- **Modelo:** representación de la información que maneja la aplicación. La lógica de datos asegura la integridad de estos y permite derivar nuevos datos.
- **Vista:** representación del modelo en forma gráfica, disponible para la interacción con el usuario. En el caso de las aplicaciones web, la vista es la página HTML sobre la que el usuario puede realizar operaciones.
- **Controlador:** es la capa encargada de manejar y responder las solicitudes del usuario, procesando la información necesaria y modificando el modelo en caso de ser necesario.

La Figura 7 muestra de manera gráfica las diferentes relaciones entre los componentes de este patrón así como las responsabilidades de cada uno de ellos.

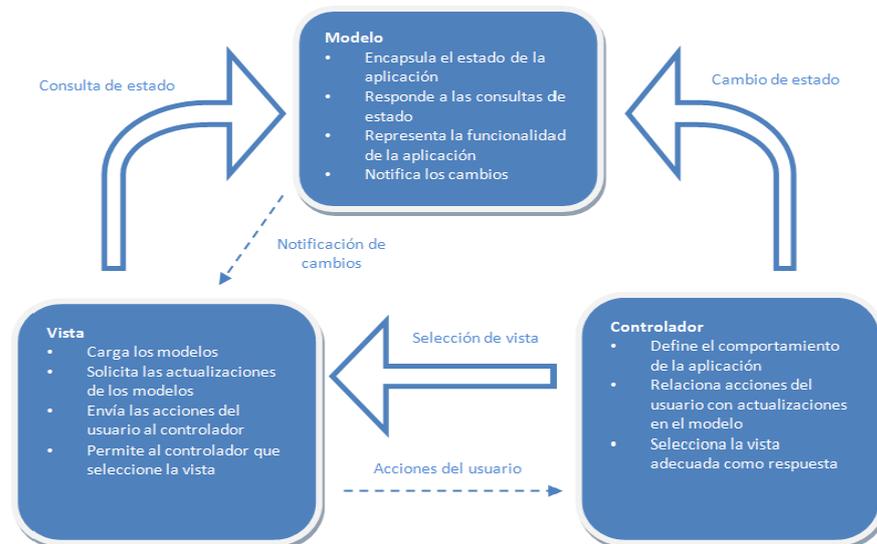


Figura 7: Relaciones y responsabilidades de los componentes del patrón MVC

²⁷ <http://www.proactiva-calidad.com/java/principal.html>, 2009, fecha consulta: 30-06-10



Este patrón de arquitectura cuenta con numerosas ventajas, de ahí que sea tan empleado para el desarrollo de aplicaciones web y que existan multitud de frameworks que faciliten su implementación. Algunas de sus ventajas son las siguientes:

- Separación del modelo de la vista, es decir, separación de los datos de la representación visual de los mismos.
- Simplifica la inserción de múltiples representaciones de los mismos datos o información.
- Facilita agregar nuevos tipos de datos según sea requerido por la aplicación debido a la independencia de capas.
- Permite la programación de las diferentes capas de manera paralela e independiente gracias a la autonomía de funcionamiento.
- Facilita el mantenimiento en caso de errores, así como las diferentes pruebas de funcionamiento del sistema.

4.5.2.2. Componentes de JSF

Los distintos componentes de JSF pueden agruparse siguiendo la estructura del patrón MVC, como puede verse en la Figura 8.

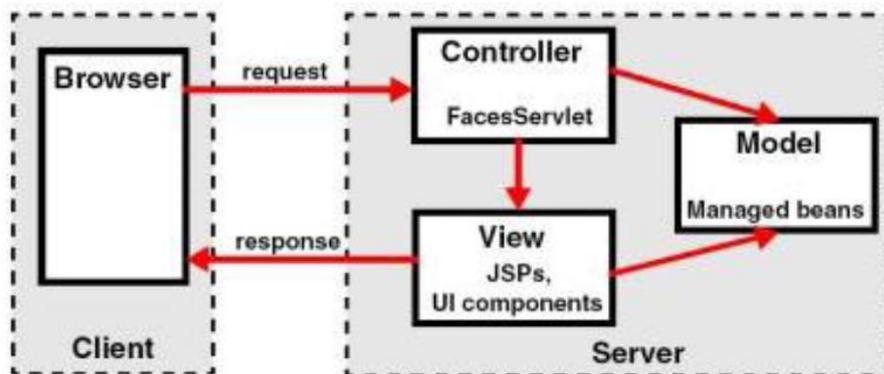


Figura 8: Desglose de la Arquitectura de JSF

En JSF la gestión de beans (managedbeans) representa el modelo de la aplicación. Estos Java Beans interactúan con la lógica de negocio o bien con sistemas externos como pueda ser una base de datos. La vista de la aplicación la componen distintos



JSPs que son creados mediante la combinación del modelo de datos con componentes de interfaz de usuario que pueden ser predefinidos o desarrollados a medida del cliente. El FacesServlet, que dirige la navegación y gestión de objetos representa el controlador de la aplicación. Los oyentes de eventos (eventlisteners) también contribuyen a la lógica de control.

En la Figura 9 se puede apreciar la comparación de la arquitectura en tres capas con el patrón arquitectónico MVC.

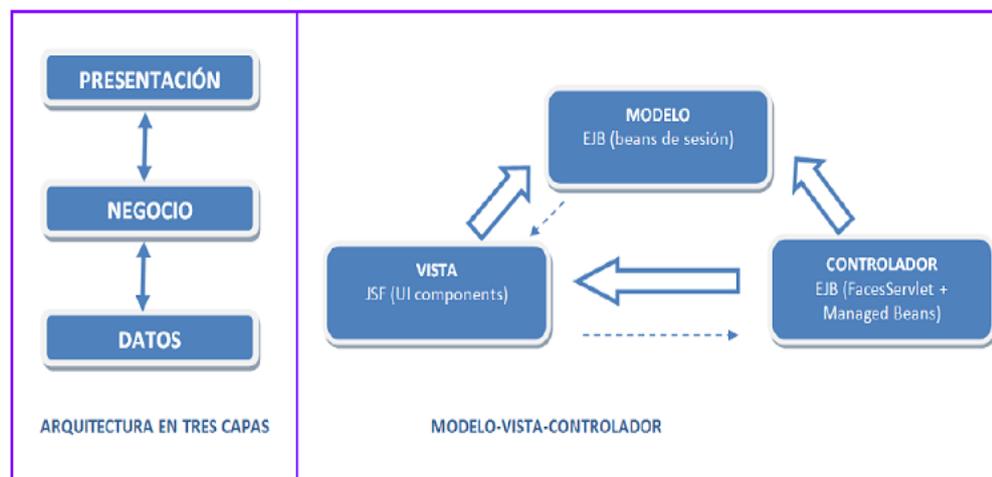


Figura 9: Comparativa arquitectura en tres capas con patrón arquitectónico MVC

4.5.2.3. Características

JSF tiene como características²⁸ principales:

- Utiliza páginas JSP para generar las vistas, añadiendo una biblioteca de etiquetas propia para crear los elementos de los formularios HTML.
- Asocia a cada vista con formularios un conjunto de objetos java manejados por el controlador (managedbeans) que facilitan la recogida, manipulación y visualización de los valores mostrados en los diferentes elementos de los formularios.

²⁸<http://www.exadel.com/web/portal/products/Tutorials>, 2008, fecha consulta: 20-07-10



- Introduce una serie de etapas en el procesamiento de la petición, como por ejemplo la de validación, reconstrucción de la vista, recuperación de los valores de los elementos, etc.
- Utiliza un sencillo fichero de configuración para el controlador en formato xml.
- Es extensible, pudiendo crearse nuevos elementos de la interfaz o modificar los ya existentes.
- Forma parte del estándar J2EE, permite crear la capa de presentación y control de una aplicación web java.

4.5.2.4. Elementos principales de una aplicación JSF

Las aplicaciones JSF están formadas por los siguientes elementos principales:

- Páginas JSP que incluyen los formularios JSF. Estas páginas generarán las vistas de la aplicación.
- Beans java que se conectan con los formularios JSF
- Clases java para la lógica de negocio y utilidades.
- Ficheros de configuración, componentes a medida y otros elementos del framework.
- Resto de recursos de la aplicación web: recursos estáticos, javascript y otros elementos.

4.5.2.5. Estructura de las páginas

Cada página JSF está formada por una página JSP que contiene un formulario (HTML FORM) y un backbean.

El controlador JSF registra en el servidor de aplicaciones un tipo especial de petición, típicamente *.jsf, que estará asociado a estas páginas.

El primer caso comienza cuando el usuario realiza en su navegador una petición de navegación a una url de tipo *.jsf. Cuando al servidor web llega una petición del tipo página JSF, el controlador JSF entra en funcionamiento.



- Primero comprueba si es la primera vez que se accede a dicha página. Si es así, carga la página jsp asociada y la procesa construyendo en memoria la representación de los controles de la página.
- El siguiente paso es asociarle los backbeans.
- El tercer paso es dar valores a las propiedades de los elementos JSF de la página.
- Finalmente el servidor devuelve al usuario una página creada a partir de una página JSP que incluye normalmente etiquetas JSF, cuyos valores se extraerán del backbean asociado, ahora ya actualizados.

4.5.2.6. Navegación entre páginas

Cuando se ejecuta una petición que incluye una acción se ejecuta el mecanismo de navegación de JSF. Tras la ejecución de la acción, el controlador determina cómo se debe mostrar al usuario el resultado de la petición. Hay varias posibilidades:

- Finalizar la petición mostrando la página jsp que originó la petición, que es la opción por defecto.
- Mostrando otra página jsp diferente.
- Enviando al usuario una petición de redirección, por lo que el navegador del usuario se dirigirá automáticamente a otra página cuando reciba la respuesta a su petición.

Este mecanismo de navegación se implementa de manera sencilla en la página JSF. Cuando el controlador JSF llama al método asociado a la acción, este devuelve un valor de tipo String. Este valor es utilizado junto con las reglas de navegación creadas en el fichero de configuración de JSF para determinar la página que se debe enviar como respuesta al usuario.

Las reglas de navegación definen:

- La página de origen. Indica el jsp que originó la petición.
- La etiqueta de destino. Es la cadena que identifica el destino. Esta cadena es



devuelta por el método del backbean que procesa la acción.

- La página de destino para cada etiqueta. Normalmente es el jsp el que procesará la petición de salida, utilizando los datos que hay en request y en sesión.
- Si es un envío directo interno o una redirección externa. En el primer caso la respuesta se generará en la misma petición mediante una redirección interna a otro jsp o servlet. En el segundo caso se enviará como respuesta al navegador una instrucción de redirección para que el navegador realice una nueva petición de otra página.

4.5.2.7. Etapas de procesamiento de una petición

Para entender el procesamiento de una página JSF hay que entender el ciclo de vida de la petición dentro del controlador JSF, esto se ilustra en la Figura 10.

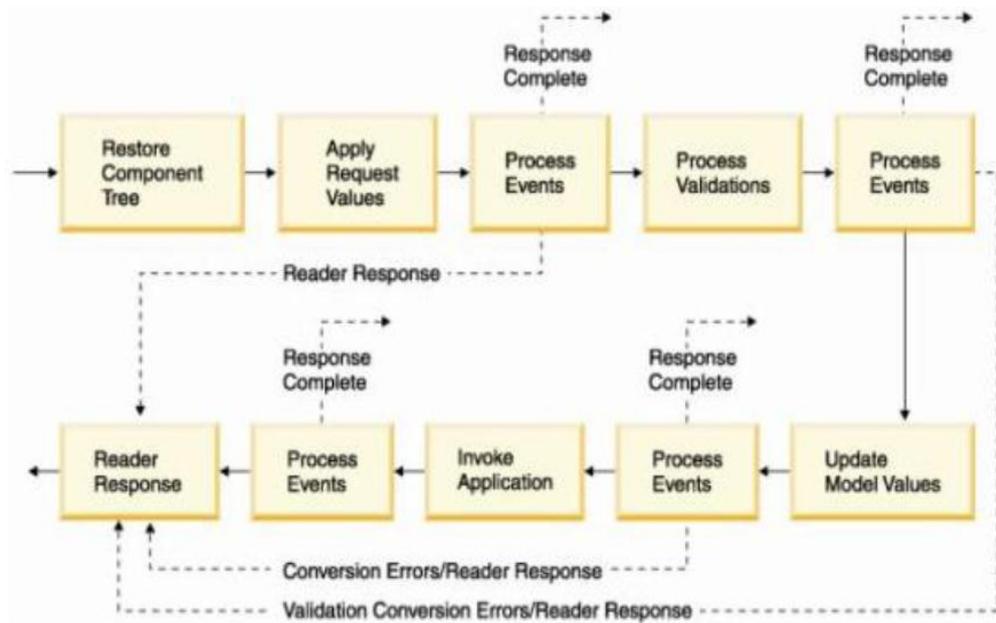
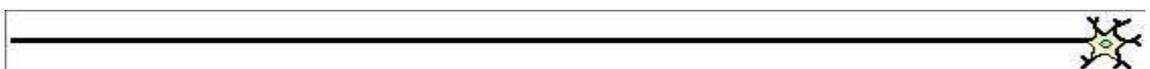


Figura 10. Ciclo de vida de JSF

Durante el procesamiento de una petición el controlador JSF realiza las siguientes etapas:



1. **Restaurar los componentes de la vista** (restoreview). En esta etapa el controlador construye en memoria la estructura de componentes de la página.
2. **Aplicar los valores de la petición.** (applyrequestvalues). En esta etapa se recuperan los valores de request y se asignan a los beans de la página.
3. **Procesamiento de las validaciones** (processvalidations). Se verifican los parámetros de entrada según un conjunto de reglas definidas en un fichero de configuración.
4. **Actualizar los valores del modelo** (updatemodelvalues). Los valores leídos y validados son cargados en los beans.
5. **Invocación a la aplicación** (invokeapplication). Se ejecutan las acciones y eventos solicitados para la página. Si es necesario se realiza la navegación.
6. **Generación de la página** (render response). En esta fase se genera la página que será enviada al usuario con todos sus elementos y valores actualizados.

4.5.3. Hibernate

4.5.3.1. Generalidades

Hibernate es una herramienta para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) que permiten establecer estas relaciones²⁹.

Hibernate funciona asociando a cada tabla de la base de datos un Plain Old Java Object (POJO, a veces llamado PlainOrdinary Java Object). Un POJO es similar a una Java Bean, con propiedades accesibles mediante métodos setter y getter.

²⁹http://www.hibernate.org/hib_docs/reference/en/html/index.x.html, 2007. fecha consulta: 28-08-10



Para poder asociar el POJO a su tabla correspondiente en la base de datos, Hibernate usa los ficheros hbm.xml.

En la Figura 11 para la clase Cat se usa el fichero Cat.hbm.xml para mapearlo con la base de datos. En este fichero se declaran las propiedades del POJO y sus correspondientes nombres de columna en la base de datos, asociación de tipos de datos, referencias, relaciones x a x con otras tablas etc.

```
<?xml version="1.0"?>
<!DOCTYPE hibernate-mapping
  PUBLIC "-//Hibernate/Hibernate Mapping DTD//EN"
  "http://hibernate.sourceforge.net/hibernate-mapping-2.0.dtd">
<hibernate-mapping>
  <class name="net.sf.hibernate.examples.quickstart.Cat" table="CAT">
    <!-- A 32 hex character is our surrogate key. It's automatically
      generated by Hibernate with the UUID pattern. -->
    <id name="id" type="string" unsaved-value="null" >
      <column name="CAT_ID" sql-type="char (32)" not-null="true"/>
      <generator class="uuid.hex"/>
    </id>
    <!-- A cat has to have a name, but it shouldn' be too long. -->
    <property name="name">
      <column name="NAME" length="16" not-null="true"/>
    </property>
    <property name="sex"/>
    <property name="weight"/>
  </class>
</hibernate-mapping>
```

Figura 11: Ejemplo de mapeo de una clase en Hibernate



4.5.3.2. Características

Las características principales de Hibernate son las siguientes:

- Hibernate es un ORM de libre distribución.
- Un framework ORM maduro y completo.
- Puede utilizarse en cualquier contexto de ejecución (no necesita un contenedor especial).
- Facilita el mapeo de atributos entre una base de datos relacional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML)
- No utiliza técnicas como generación de código a partir de descriptores de los modelos de datos o manipulación de bytecodes en tiempo de compilación ni obliga a implementar interfaces específicos.
- Basado en el mecanismo de reflexión de Java.
- No intrusivo (estilo POJO)
- Buena documentación y comunidad amplia y activa
- Transacciones, caché, asociaciones, polimorfismo, herencia, persistencia transitiva, estrategias de fetching.
- Potente lenguaje de consulta (HQL): subqueries, outerjoins, ordering, proyección (reportquery), paginación.
- Fácil testeo.
- Cada clase persistente necesita un fichero XML de mapeo.

4.5.3.3. Arquitectura

El API de Hibernate es una arquitectura de dos capas (Capa de persistencia y Capa de Negocio)³⁰.

En la Figura 12 se muestran los roles de las interfaces Hibernate más importantes en las capas de persistencia y de negocio de una aplicación J2EE. La capa de negocio está situada sobre la capa de persistencia, debido a que actúa como un cliente de la capa de persistencia.

³⁰<http://www.svsystemmobile.com/articles/IntroductionToHibernate.html>, 2006, fecha consulta: 30-08-10 

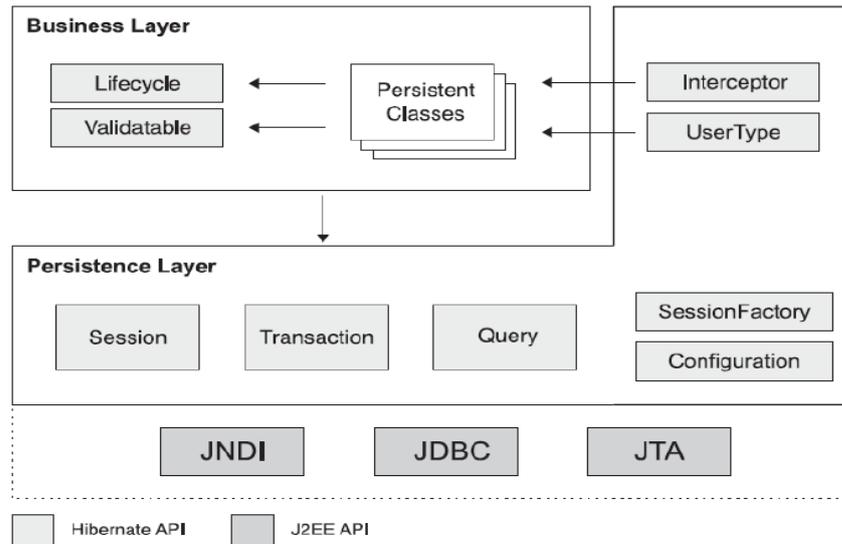


Figura 12: Arquitectura de Hibernate

Las interfaces mostradas se clasifican de la siguiente forma:

- Interfaces llamadas por la aplicación para realizar operaciones básicas:
 - **Session:** interfaz primaria utilizada en cualquier aplicación Hibernate (**SessionFactory**).
 - **Transaction**
 - **Query:** permite realizar peticiones a la base de datos y controla cómo se ejecuta dicha petición (query). Las peticiones se escriben en HQL o en el dialecto SQL nativo de la base de datos que estamos utilizando.
- Interfaces llamadas por el código de la infraestructura de la aplicación para configurar Hibernate. La más importante es la clase **Configuration** que se utiliza para configurar y "arrancar" Hibernate. La aplicación utiliza una instancia de **Configuration** para especificar la ubicación de los documentos que indican el mapeado de los objetos y propiedades específicas de Hibernate, y a continuación crea la **Session Factory**.
- Interfaces callback que permiten a la aplicación reaccionar ante determinados eventos que ocurren dentro de la aplicación, tales como **Interceptor**, **Lifecycle**



y **Validatable**.

- Interfaces que permiten extender las funcionalidades de mapeado de Hibernate, como por ejemplo **UserType**, **CompositeUserType**, e **IdentifierGenerator**.

Además, Hibernate hace uso de APIs de Java, tales como JDBC, JTA (Java Transaction Api) y JNDI (Java NamingDirectory Interface).

4.5.3.4. Hibernate Query Language HQL

Hibernate nos proporciona un lenguaje para el manejo de consultas a la base de datos. Este lenguaje es similar a SQL y es utilizado para obtener objetos de la base de datos según las condiciones especificadas en el HQL.

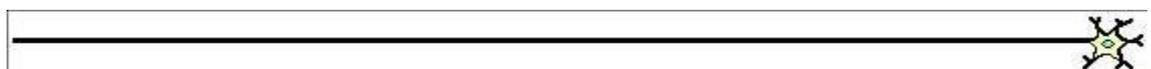
El uso de HQL nos permite usar un lenguaje intermedio que según la base de datos que usemos y el dialecto que especifiquemos será traducido al SQL dependiente de cada base de datos de forma automática y transparente.

4.5.3.5. Estructura del fichero XML

El esquema general de un fichero XML de mapeo es algo similar a lo siguiente:

```
<Encabezado XML>  
<Declaración de la DTD>  
<class - Definición de la clase persistente>  
<id - Identificador>  
<generator - Clase de apoyo a la generación automática de OID's>  
<component - Componentes, son las columnas de la tabla>  
<one-to-many / many-to-many –Posibles relaciones con otras entidades persistentes>
```

Figura 13: Estructura del fichero XML



A continuación se detallan las características, parámetros y definición de las etiquetas arriba utilizadas así como de algunas otras que nos serán de utilidad a la hora de pasar nuestro esquema relacional a ficheros de mapeo XML.

- **Declaración de la DTD.** El documento DTD de los ficheros XML se encuentra en cada distribución de Hibernate en el propio **.jar** o en el directorio src.
- Elemento Raíz **<hibernate-mapping>**. Dentro de él se declaran las clases de los objetos persistentes. Aunque es posible declarar más de un elemento **<class>** en un mismo fichero XML, no debería hacerse ya que aporta una mayor claridad a la aplicación realizar un documento XML por clase de objeto persistente.
- **<class>**. Este es el tag donde se declara la clase persistente. Una clase persistente equivale a una tabla en la base de datos y un registro o línea de esta tabla es un objeto persistente de esta clase.
- **<id>**. Permite definir el identificador del objeto (OID). Se corresponderá con la clave principal de la tabla en la BBDD.
- **<discriminator>**. Cuando una clase declara un discriminador es necesaria una columna en la tabla que contendrá el valor de la marca del discriminador. Los conjuntos de valores que puede tomar este campo son definidos en la cada una de las clases o subclases a través de la propiedad **<discriminator-value>**. Los tipos permitidos son: String, Character, Integer, Byte, Short, Boolean, yes_no, true_false.
- **<property>**. Declara una propiedad persistente de la clase que se corresponde con una columna.
- **Tipos de relaciones (Componentes y Colecciones).** En todo diseño relacional los objetos se referencian unos a otros a través de relaciones, las típicas son: uno a uno 1-1, uno a muchos 1-n, muchos a muchos n-m, muchos a uno n-1.



4.5.4. Joone

4.5.4.1. Descripción

Joone (<http://www.joone.org/>) es un framework en Java para construir y ejecutar aplicaciones de inteligencia artificial basadas en redes neuronales. Las aplicaciones realizadas con Joone pueden ser construidas en una máquina local, entrenadas en un ambiente distribuido y ejecutarse en cualquier dispositivo³¹.

Joone consiste de una arquitectura modular basada en componentes vinculados que pueden ser extendidos para construir nuevos algoritmos de aprendizaje y nuevas arquitecturas de redes neuronales.

Todos los componentes poseen algunas características básicas específicas, como persistencia, multihilos (multithreading), serialización y parametrización. Estas características garantizan escalabilidad, reliability y expansibilidad, todas características obligatorias para alcanzar el destino final de los estándares futuros en el mundo de la inteligencia artificial.

Cada red neuronal está compuesta de varias capas de neuronas conectadas mediante sinapsis (Figura 14). Pueden crearse variadas arquitecturas de redes neuronales dependiendo de cómo las neuronas son conectadas.

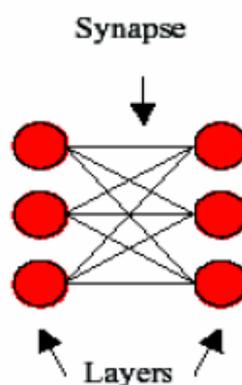


Figura 14. Conexiones entre capas de neuronal

³¹<http://www.joone.org/IooneCompleteGuide.pdf>, 2008, fecha consulta: 25-09-10



En la Figura 14 se observa un ejemplo de una red con dos capas y conectada mediante una sinapsis. Cada capa está compuesta de una cierta cantidad de neuronas, cada una de las cuales poseen las mismas características (función de transferencia, tasa de aprendizaje, etc.).

Una red neuronal construida con Joone puede estar compuesta de cualquier cantidad de capas de diferentes tipos. Cada capa procesa sus señales de entrada, aplica la función de transferencia y envía los patrones resultantes a la sinapsis que la conecta con la capa siguiente.

Las capas y las sinapsis poseen sus propios mecanismos para ajustar los pesos dependiendo del algoritmo de aprendizaje seleccionado.

4.5.4.2. Capas de neuronas

El objeto Capa (Layer) es el elemento básico que forma una red neuronal.

Cada capa está compuesta de neuronas con las mismas características, que transfieren los patrones de entrada a patrones de salida ejecutando una función de transferencia.

Existen diferentes tipos de capas, las cuales son detalladas a continuación.

- **Capa Lineal:** La Capa Lineal (LinearLayer) es el tipo de capa más simple y solamente transfiere los patrones de entrada a la salida aplicando una transformación lineal.
- **Capa Lineal Parcial:** La Capa Lineal Parcial (BiasedLinearLayer) utiliza como función de transferencia la adición de un parámetro ajustable al valor de entrada. El parámetro puede ser modificado durante la etapa de entrenamiento.
- **Capa Sigmoidal:** La Capa Sigmoidal (SigmoidLayer) aplica una función de transferencia sigmoidea a los patrones de entrada, limitando la salida al rango $[0,1]$.



- **Capa Tangente Hiperbólica:** La Capa Tangente Hiperbólica (TanhLayer) es similar a la Capa Sigmoidal excepto que aplica la función tangente hiperbólica, limitando la salida al rango $[-1,1]$.
- **Capa Logarítmica:** La Capa Logarítmica (LogarithmicLayer) aplica una función de transferencia logarítmica a los patrones de entrada, resultando en una salida cuyo rango es $[0, \infty]$.
- **Capa Senoidal:** La Capa Senoidal (SineLayer) utiliza como salida la función seno evaluada en la suma de los valores de las entradas ponderadas por sus pesos.
- **Capa con Retraso:** La Capa con Retraso (DelayLayer) aplica la suma de los valores de entrada a una línea de retraso, de modo que cada neurona es atrasada por la cantidad de iteraciones especificada como parámetro.
- **Capa con Contexto:** La Capa con Contexto (ContextLayer) es similar a la capa lineal excepto que posee una conexión auto recurrente entre su salida y su entrada, que propaga hacia atrás la salida de la neurona en el paso anterior.
- **Capa de Gauss:** La Capa de Gauss (GaussLayer) posee como salida la suma de las entradas ponderadas por los pesos, aplicada a una función gaussiana.
- **Capa RBF Gaussiana:** La Capa RBF Gaussiana (RBFGaussianLayer) posee como salida la suma de las entradas ponderadas por los pesos, aplicada a una función gaussiana, al igual que la Capa de Gauss, pero es posible ajustar el centro de la curva de gauss utilizada.
- **Capa “El Ganador Se Lleva Todo”:** La Capa “El Ganador Se Lleva Todo” (WinnerTakeAllLayer) recibe las distancias Euclídeas entre los pesos de la sinapsis anterior (una sinapsis de Kohonen) y sus entradas.
La capa calcula cual es el nodo que posee la menor distancia a la entrada y envía como salida el valor 1 para ese nodo y 0 para los demás.



- **Capa Gaussiana:** La Capa Gaussiana (GaussianLayer) es similar a la capa “El Ganador Se Lleva Todo”, pero las neuronas son activadas de acuerdo a una superficie gaussiana centrada en la neurona más activa (la ganadora).

4.5.4.3. Sinapsis

El objeto Sinapsis (Synapse) representa la conexión entre dos capas de neuronas, permitiendo que un patrón se transfiera de una capa a la otra.

Las sinapsis poseen los pesos de las conexiones, los cuales son modificados en la etapa de entrenamiento de acuerdo al algoritmo de aprendizaje seleccionado.

- **Sinapsis Directa:** La Sinapsis Directa (DirectSynapse) representa una conexión directa uno a uno entre los nodos de dos capas, las cuales poseen la misma cantidad de neuronas. Cada conexión posee un peso igual a 1 que no es modificado en la etapa de entrenamiento.
- **Sinapsis Completa:** La Sinapsis Completa (FullSynapse) conecta todos los nodos de una capa con todos los nodos de otra capa. Los pesos de sus conexiones se modifican en la etapa de entrenamiento de acuerdo al algoritmo de aprendizaje seleccionado.
- **Sinapsis con Retraso:** La Sinapsis con Retraso (DelayedSynapse) posee una arquitectura similar a la Sinapsis Completa, pero cada conexión es implementada usando una matriz de filtros FIR (Finite Impulse Response, Respuesta a Impulso Finito), que retrasa las conexiones permitiendo implementar un algoritmo de propagación hacia atrás temporal.
- **Sinapsis de Kohonen:** La Sinapsis de Kohonen (KohonenSynapse) pertenece a un tipo especial de componentes que permite construir redes neuronales no supervisadas. Durante la etapa de entrenamiento, los pesos de sus conexiones son ajustados para mapear los patrones de entrada N-dimensionales a un mapa de 2 dimensiones.



- **Sinapsis de Sanger:** La Sinapsis de Sanger (SangerSynapse) se utiliza para construir redes neuronales no supervisadas que implementan el algoritmo PCA (Principal Component Analysis, Análisis de Componentes Principales).
- **Sinapsis de Entrada:** Existen diversos tipos de sinapsis de entrada, dependiendo del origen de los patrones, las cuales se detallan a continuación.
 - Sinapsis de Entrada de Archivo
 - Sinapsis de Entrada de URL
 - Sinapsis de Entrada de Excel
 - Sinapsis de Entrada de JDBC
 - Sinapsis de Entrada de YahooFinance
 - Sinapsis de Entrada de Memoria
- **Sinapsis de Salida:** Joone también posee sinapsis de salida que permiten a la red neuronal escribir los patrones de salida en varios soportes de almacenamiento, como por ejemplo: archivos ASCII, hojas de cálculo de Excel, tablas de base de datos utilizando JDBC y vectores de Java.
- **Sinapsis Maestra:** La función de la Sinapsis Maestra (TeacherSynapse) es calcular la diferencia entre la salida de una red neuronal y el valor deseado obtenido de un origen externo de datos.

4.5.4.4. El Monitor

El objeto Monitor representa en punto central donde se encuentran todos los parámetros necesarios para que los componentes de la red neuronal funcionen correctamente, como la velocidad de aprendizaje, el momento, el algoritmo de aprendizaje, la cantidad de ciclos de entrenamiento, el ciclo actual, etc.

Cada capa de neuronas y cada sinapsis de una red neuronal recibe un puntero a una instancia de un objeto Monitor, de modo que, cuando el usuario desea cambiar algún parámetro, solo debe hacerlo en el objeto Monitor.



El Monitor también provee a la aplicación que lo utiliza de un mecanismo de notificaciones basado en varios eventos arrojados cuando una acción en particular sucede. Por ejemplo, una aplicación externa puede ser informada cuando la red neuronal comienza o termina los ciclos de entrenamiento, cuando finaliza un ciclo o cuando el valor del error global cambia durante la etapa de entrenamiento.

4.5.4.5. Ventajas

Joone ofrece las siguientes ventajas:

- **Clases en Java que permiten integrarlo en una aplicación:** debido a que Joone está compuesto de clases en java, se lo puede integrar como parte de otra aplicación, pudiendo generar un sistema completo que utilice las ventajas de las redes neuronales de Joone.
- **Manejo de componentes que permiten definir cualquier arquitectura de red:** ya que Joone está construido sobre la base de componentes modulares, que permiten conectar distintos tipos de componentes, formando cualquier arquitectura de red, pudiendo obtener exactamente la red necesaria.
- **Completa documentación:** documentación completa, explicando el funcionamiento de los componentes y como integrarlos para generar distintos tipos de redes neuronales.
- **Código fuente abierto:** se posee a disposición el código fuente, permitiendo, en caso que sea necesario, modificar alguno de los comportamientos de los componentes para adecuarlos al funcionamiento deseado.
- **Licencia LGPL:** distribuido bajo la licencia GNU Lesser General Public License, permitiendo su uso como parte de la aplicación de prueba a desarrollar.



4.5.5. ChartDirector

4.5.5.1. Definición

ChartDirector es un componente poderoso para crear gráficos profesionales y clickeables para ASP.NET y aplicaciones Windows Forms. Su arquitectura única de capas permite sintetizar el gráfico deseado usando capas estándar de gráfico.

4.5.5.2. Características

ChartDirector permite usar exhaustivos estilos entre los que se incluyen: tora, rosquilla, barras, líneas, curvas, discretos, líneas de tendencia, aproximados a curva, coloreado interlinea, área, dispersión, burbujas, caja flotante, caja de interruptores, cascada, financieros (soporta numerosos indicadores), gantt, vectorial, radar, polar, rosa, y medidores angulares, lineales y tipo reloj. Los gráficos ChartDirector son interactivos, con un juego completo de eventos de mouse, soporta consejos, objetos de gráfico clickeables, y capacidades "arrastrar para agrandar" y "arrastrar para desplazarse". Las capacidades AJAX habilitadas permiten actualizarse sin refrescar las páginas web³².

³²<http://www.advsofteng.com/product.html>, 2009, fecha consulta: 28-09-10



5. EVALUACIÓN DEL OBJETO DE INVESTIGACIÓN

La carrera de Ingeniería en Sistemas de la Universidad Nacional de Loja proporciona formación de pregrado de alta calidad en el campo de la Informática y la Computación, capaz de formar profesionales con fundamentos de: Software de Aplicación a través de Sistemas de Gestión de Bases de Datos, Educativo y Multimedia, Diseño Asistido por Computadora, Software para Automatización de Procesos al igual que las Telecomunicaciones y Redes; en este contexto desarrollamos una aplicación web destinada al análisis de los archivos log del sitio web de la Universidad Nacional de Loja y de esta manera poner en práctica los conocimientos adquiridos en los años de estudio universitarios.

Después de haber realizado un extenso análisis de los requerimientos el enunciado de la problemática quedó redactado de la siguiente manera: "Limitación en los resultados obtenidos en el proceso de descubrimiento de preferencias de uso en los sitios web debido a la aplicación de métodos estadísticos tradicionales y técnicas de análisis de datos inadecuadas", el propósito de este proyecto es proponer la utilización de Redes Neuronales Artificiales SOM en la construcción de una herramienta destinada a descubrir las preferencias de uso de los cibernautas del sitio web de la Universidad Nacional de Loja y de esta manera proveer al Administrador de este sitio, las exigencias de los usuarios.

El presente proyecto de tesis fue debidamente presentado, revisado y aprobado por los responsables de la línea de investigación de proyectos de tesis de la carrera de Ingeniería en Sistemas de la UNL. Así mismo como el proyecto está enfocado a dar solución a una problemática de esta Institución, las autoridades se comprometieron a proporcionar toda la información para culminar con éxito el presente proyecto investigativo.



6. DESARROLLO DE LA PROPUESTA ALTERNATIVA

6.1. ANÁLISIS DE LA APLICACIÓN

Para realizar el presente trabajo investigativo se plantearon los siguientes objetivos:

Objetivo General

- Construir e implementar la Herramienta para el descubrimiento de preferencias de uso de los cibernautas del sitio web de la Universidad Nacional de Loja aplicando Redes Neuronales Artificiales SOM, con la finalidad de identificar las preferencias de los usuarios de manera que a futuro el Administrador de éste sitio pueda adecuarlo basándose en resultados obtenidos a través de ésta herramienta.

Este objetivo se logró gracias al análisis minucioso de los archivos log almacenados en el servidor web de la Universidad Nacional de Loja proporcionados por el Ing. Jamil Ramón, Jefe de Informática de la Universidad Nacional de Loja, a las herramientas de desarrollo empleadas y a la orientación brindada por nuestro Director de tesis el Ing. Wilman Chamba.

Objetivos Específicos

- Analizar y determinar los requerimientos para el descubrimiento de preferencias de uso en el sitio web de la Universidad Nacional de Loja aplicando Redes Neuronales Artificiales SOM.

Para cumplir este objetivo acudimos a la Jefatura de Informática donde se pudo obtener toda la información relacionada con los archivos log almacenados en el servidor web de la Universidad Nacional de Loja; de esta manera se pudieron determinar los requerimientos funcionales y no funcionales de la aplicación.



- Realizar el diseño de la Herramienta para el descubrimiento de preferencias de uso en el sitio web de la Universidad Nacional de Loja aplicando Redes Neuronales Artificiales SOM en base a los requerimientos obtenidos utilizando las herramientas de modelado apropiadas.

La utilización de StarUML facilitó el cumplimiento de este objetivo, permitiéndonos realizar el diseño de los diagramas de casos de uso, diagramas de secuencia, diagramas de clases y diagramas de paquetes de la aplicación.

- Codificar los módulos de la aplicación en base a los diseños previamente realizados utilizando las herramientas: StarUML, Enterprise, MySQL y NetBeans.

La codificación de la aplicación fue posible gracias a la utilización del IDE NetBeans 6.5 el cual es un producto de código abierto y gratuito sin restricciones de uso, escrito completamente en Java.

- Llevar a cabo las respectivas pruebas lo que permitirá encontrar y corregir falencias y debilidades para asegurar el perfecto funcionamiento de la herramienta.

Este objetivo se cumplió mediante la aplicación de una encuesta a 5 usuarios (estudiantes de la Carrera de Ingeniería en Sistemas) y la demostración del funcionamiento de la aplicación al Ing. Patricio Valarezo, Responsable de la sección de desarrollo de software de la UNL.

- Implementar la herramienta y demostrar su funcionamiento con los archivos log del servidor web de la UNL.

Este objetivo se cumplió mediante el alquiler de un servidor particular para el alojamiento y administración de la aplicación a través de una ip pública, lo que además asegura su propagación y funcionamiento en la web.

- Presentar los resultados obtenidos en el análisis del sitio web de la Universidad Nacional de Loja a través de ésta herramienta.



Los resultados obtenidos en el análisis del sitio web de la UNL fueron presentados al Ing. Wilman Chamba, Director de tesis, usuarios de la aplicación y al Ing. Patricio Valarezo, Responsable de la sección de desarrollo de software de la UNL.

6.1.1. Definición del problema

La mayoría de herramientas diseñadas para descubrir hábitos de uso de sitios web se han basado en técnicas y métodos estadísticos por ello no han logrado obtener más información de la que se encuentra explícita en los registros de acceso de los servidores web; de aquí surge la necesidad de construir una “**Herramienta para el descubrimiento de preferencias de uso de los cibernautas del sitio web de la Universidad Nacional de Loja aplicando Redes Neuronales Artificiales SOM**”, que permita identificar los requerimientos más solicitados por los usuarios de este sitio.

Para evaluar la solución propuesta se realizó el estudio de los archivos log del servidor web de la Universidad Nacional de Loja.

Para el presente trabajo se decidió manejar el formato de log común (CLF) debido a la gran utilización del mismo. También se permite la utilización del formato de log común más los campos Referrer y Agent.

Se realizó la herramienta parametrizable, de modo que las tareas de limpieza puedan ser adaptadas a las necesidades de cada caso. Para ello, se utilizaron distintos filtros a los archivos log, los cuales permiten que el usuario defina su comportamiento.

Es posible acceder a información sobre la identificación de hábitos de uso realizada, como la cantidad de sesiones de usuario en cada grupo descubierto, el detalle de las sesiones en cada grupo y la frecuencia de acceso a cada página del sitio estudiado por parte de los usuarios de cada grupo.

6.1.2. Glosario de términos

- **Administrador:** Usuario que posee privilegios para ver y eliminar cuentas de usuario.
- **Aplicación web:** Es un sistema informático que los usuarios utilizan



accediendo a un servidor web a través de Internet o de una intranet.

- **Archivo log:** Archivo que contiene la información de acceso y operaciones de todos los usuarios del sitio web de la Universidad Nacional de Loja en orden cronológico. Su registro consta básicamente de: dirección IP, autor, fecha y hora, variable identificadora, url, código de error y número de bytes transmitidos.
- **Bean:** Es un componente de software reutilizable que permite acceder a sus atributos mediante los get y set correspondientes.
- **Clúster:** Número de grupo al que pertenece un conjunto de sesiones de usuario.
- **Clusterización:** Consiste en agrupar las sesiones de los usuarios a través de la Red Neuronal Artificial SOM.
- **DAO:** Data Access Object. Código de acceso a datos.
- **Fragmento de página:** Es una porción de código que debe ser incrustado en una página JSP.
- **HTML** (Hyper Text Markup Language): Es un "lenguaje de descripción de página" independiente de la plataforma en que se utilice. Define la estructura y contenido de los documentos insertando en el texto de los mismos una serie de marcas (tags o etiquetas) que delimitan elementos del documento como cabeceras, párrafos, etc., y se usan para dar un tratamiento diferente al texto que se encuentre entre ellas.
- **JSP:** Es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo. Las JSP's permiten la utilización de código Java mediante scripts. Además es posible utilizar algunas acciones JSP predefinidas mediante etiquetas.



- **MVC:** Patrón de arquitectura Modelo-Vista-Controlador.
- **Página en clúster:** Frecuencia de cada página del sitio en las sesiones de usuario pertenecientes a cada grupo descubierto mediante la RNA SOM.
- **Proveedor de datos:** Es un contenedor de datos el cual representa una capa entre los componentes de una aplicación web y su nivel de permanencia (tabla o colección de datos). El proveedor de datos más importante es el ObjectListDataProvider.
- **Proyecto:** Proyecto sobre el cual un usuario determinado realiza el análisis de uno o más archivos log del sitio web de la Universidad Nacional de Loja.
- **PSI:** Plan de Sistemas de Información.
- **RequestBean:** Es un objeto cuyo ciclo de vida es muy corto; inicia cuando se realiza la petición y muere cuando el servidor ha enviado la respuesta.
- **RNA:** Red Neuronal Artificial.
- **Sesión de Usuario:** Sesión iniciada por un usuario mediante un nombre y una contraseña.
- **Sesión de usuario log:** Indica las distintas sesiones de usuario basándose en la dirección IP y teniendo en cuenta el tiempo máximo entre peticiones consecutivas.
- **SessionBean:** Es un objeto conocido cookie que contiene la información de conexión de un usuario. Su ciclo de vida es mayor al de requestbean; inicia cuando un usuario hace una petición al servidor y muere cuando hay un largo tiempo de espera o cuando el usuario desea desaparecer.
- **SOM:** Self Organizing Maps. Red Neuronal Artificial que emplea la librería joone.engine.org la cual permite analizar un archivo que se construye con los patrones de entrada o sesiones de los usuarios de un archivo log.



- **UML:** Unified Modeling Language. Lenguaje estandarizado para especificar, visualizar, construir y documentar los artefactos de un sistema software.
- **XML:** Extensible Markup Language. Es un metalenguaje extensible de etiquetas. Permite definir la gramática de lenguajes específicos (de la misma manera que HTML). Por lo tanto XML no es realmente un lenguaje en particular sino una manera de definir lenguajes para diferentes necesidades.

6.1.3. Documento de requerimientos

6.1.3.1. Requerimientos Funcionales

El sistema permitirá:

RF 001	Ingresar usuarios mediante nombre y contraseña para su sesión en el sistema.	Evidente
RF 002	Crear cuentas de usuario con roles de Administrador o Común.	Evidente
RF 003	Generar nuevos usuarios al Administrador.	Evidente
RF 004	Listar todos los usuarios del sistema al Administrador.	Evidente
RF 005	Eliminar cuentas de usuario al Administrador.	Evidente
RF 006	Crear varios proyectos para cada uno de los usuarios registrados en el sistema donde se realizarán distintos análisis del sitio web de la Universidad Nacional de Loja. Debiéndoseles asignar un nombre y una descripción.	Evidente
RF 007	Listar todos los proyectos pertenecientes a un usuario.	Evidente
RF 008	Abrir proyectos del usuario.	Evidente



RF 009	Mostrar un resumen del proyecto seleccionado indicando su nombre, descripción, cantidad de archivos log cargados, cantidad de registros log cargados y la cantidad de sesiones cargadas.	Evidente
RF 010	Cerrar proyectos del usuario.	Evidente
RF 011	Eliminar proyectos del usuario.	Evidente
RF 012	Subir archivos log al directorio del usuario, los cuales estarán disponibles para su debido análisis.	Evidente
RF 013	Cargar un archivo log que se encuentra subido en el proyecto actual.	Evidente
RF 014	Pre-procesar los archivos log obteniendo los datos necesarios para el descubrimiento de patrones.	Oculto
RF 015	Soportar el formato CLF (Common Log Format) o CLF Extendido del archivo log seleccionado.	Evidente
RF 016	Utilizar tres filtros distintos para pre-procesar el archivo log: <ul style="list-style-type: none"> 1. Filtro de extensiones de recursos (htm, html, php, pdf, xls, doc, gif, jpg, png, asp) para el armado de las sesiones de los usuarios. 2. Filtro de códigos de error de http (200) para la tarea de la construcción de las sesiones de usuarios. 3. Filtro de distinción de páginas en base a variables GET y POST. 	Evidente
RF 017	Mostrar un resumen del archivo log cargado.	Evidente
RF 018	Listar todos los registros de un archivo log cargado y pre-procesado dentro del proyecto de un usuario.	Evidente
RF 019	Identificar los distintos usuarios del archivo log pre-procesado sobre la base de su dirección IP.	Oculto
RF 020	Identificar las sesiones de usuario en base a la cantidad mínima de páginas por sesión, tiempo máximo entre peticiones y la frecuencia mínima de cada página.	Oculto



RF 021	Listar todas las sesiones de usuario identificadas a partir de un archivo log cargado y pre-procesado.	Evidente
RF 022	Emplear una Red Neuronal Artificial SOM a través de la librería joone-engine.jar, que posee tantas entradas como páginas frecuentes posea un sitio web. La salida de la Red Neuronal SOM es un mapa de dos dimensiones de $N \times N$, donde N es configurable por el usuario.	Oculto
RF 023	Agrupar a los usuarios utilizando la Red Neuronal Artificial SOM.	Oculto
RF 024	Indicar a qué grupo de usuarios corresponde cada sesión de usuario identificada en la clusterización.	Evidente
RF 025	Mostrar a través de gráficos de pasteles utilizando la librería CharDirector.jar la cantidad de sesiones que existen en cada grupo de usuarios descubierto mediante la clusterización.	Evidente
RF 026	Listar las páginas a las que acceden frecuentemente los usuarios de cada grupo.	Evidente
RF 027	Mostrar a través de gráficos de barras utilizando la librería CharDirector.jar la frecuencia con la que son accedidas las páginas por los usuarios de cada grupo.	Evidente
RF 028	Listar todos los archivos log cargados y pre-procesados en su proyecto actual y todos los archivos de entrada y salida de la red SOM.	Evidente
RF 029	Obtener el detalle de los archivos log, los archivos de entrada y los archivos de salida de la red SOM.	Evidente
RF 030	Eliminar un archivo log.	Evidente
RF 031	Cerrar una sesión de usuario.	Evidente



6.1.3.2. Requerimientos No Funcionales

RNF 001	El sistema se podrá ejecutar bajo plataformas Unix y Windows 9x o superior.
RNF 002	El sistema debe ser accedido en un entorno web, de modo que el usuario pueda utilizarla directamente desde su navegador favorito.
RNF 003	El sistema está desarrollado en Java utilizando NetBeans 6.5 con el Framework JSF basado en el patrón de arquitectura Modelo-Vista-Controlador (MVC).
RNF 004	El sistema dispondrá del servidor web Apache Tomcat 6.0.18 y el servidor de base de datos MySQL con tablas InnoDB para almacenar los datos de la aplicación.
RNF 005	El sistema empleará la arquitectura cliente-servidor, con interfaz web, utilizando el protocolo HTTP.
RNF 006	El sistema emplea interfaces gráficas simples y amigables de administración en ambiente web, de modo que usuarios sin experiencia en el uso de herramientas de este tipo puedan trabajar fácilmente y realizar los análisis que necesiten.
RNF 007	El acceso al sistema estará restringido por el uso de claves asignadas a cada uno de los usuarios.
RNF 008	El sistema responderá al acceso de los usuarios y a sus procesos dependiendo de la velocidad de conexión.
RNF 009	El sistema empleará la RNA SOM la cual se construirá a través de la librería joone-engine.jar.
RNF 010	El sistema estará completamente documentado tanto en el código fuente como en los manuales del programador y del usuario.



6.2. MODELO DE LA APLICACIÓN

6.2.1. Modelo del dominio

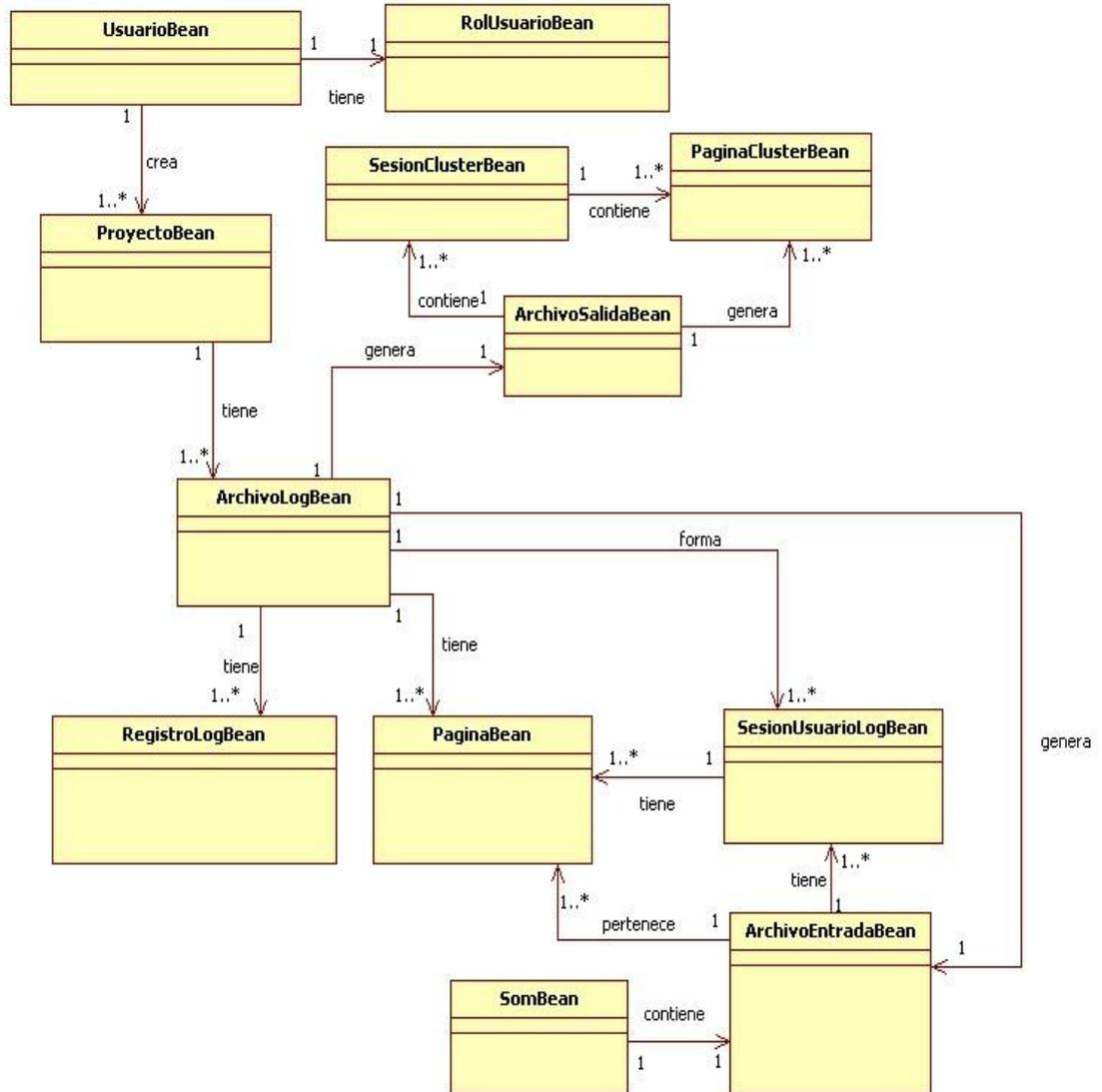


Figura 15: Modelo del dominio de la Aplicación



6.3. MODELO DINÁMICO DE LA APLICACIÓN

6.3.1. Diagrama de Casos de Uso

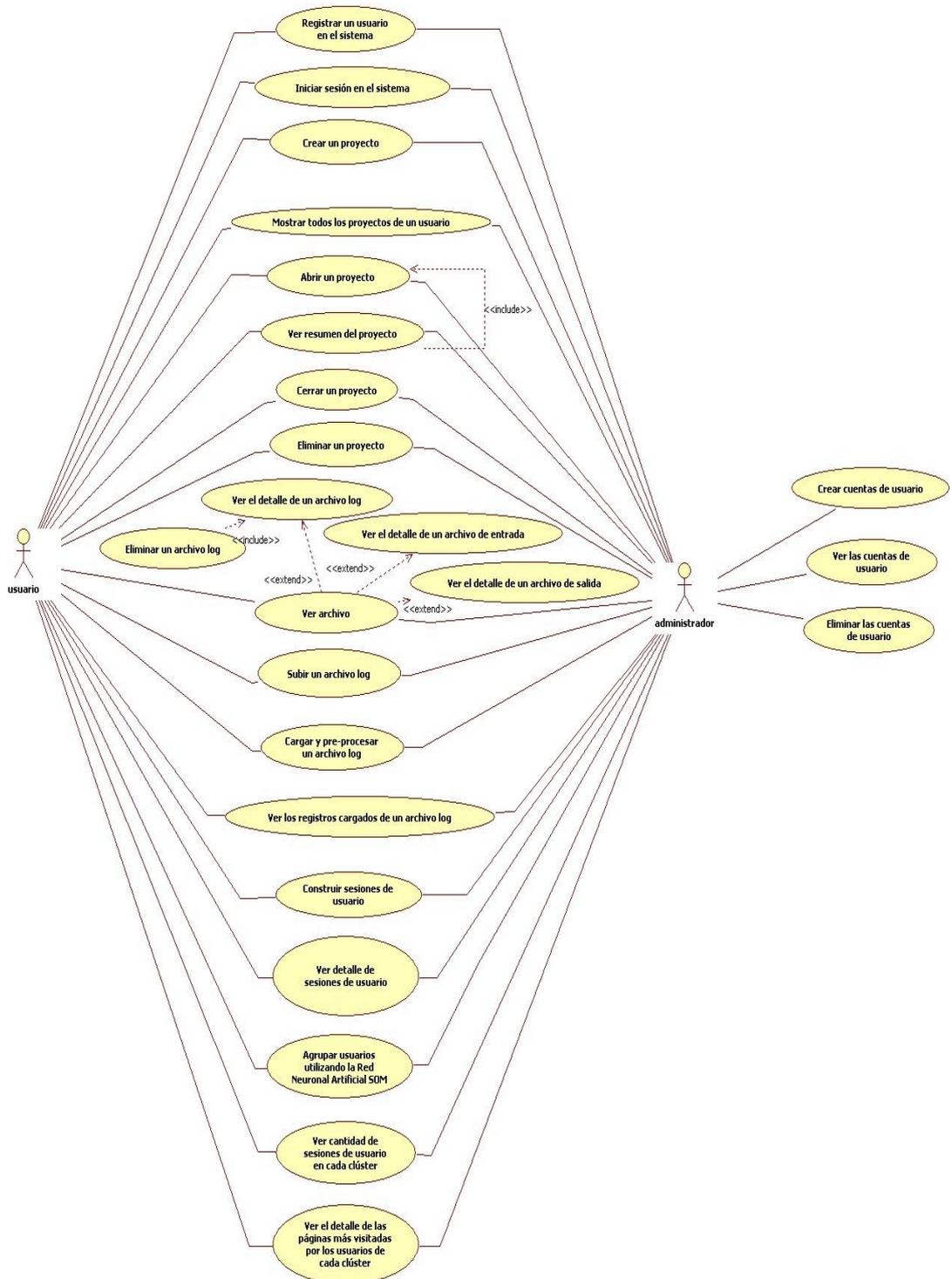


Figura 16: Diagrama de Casos de Uso



6.3.1.1. Diagrama de Casos de Uso del Usuario Común

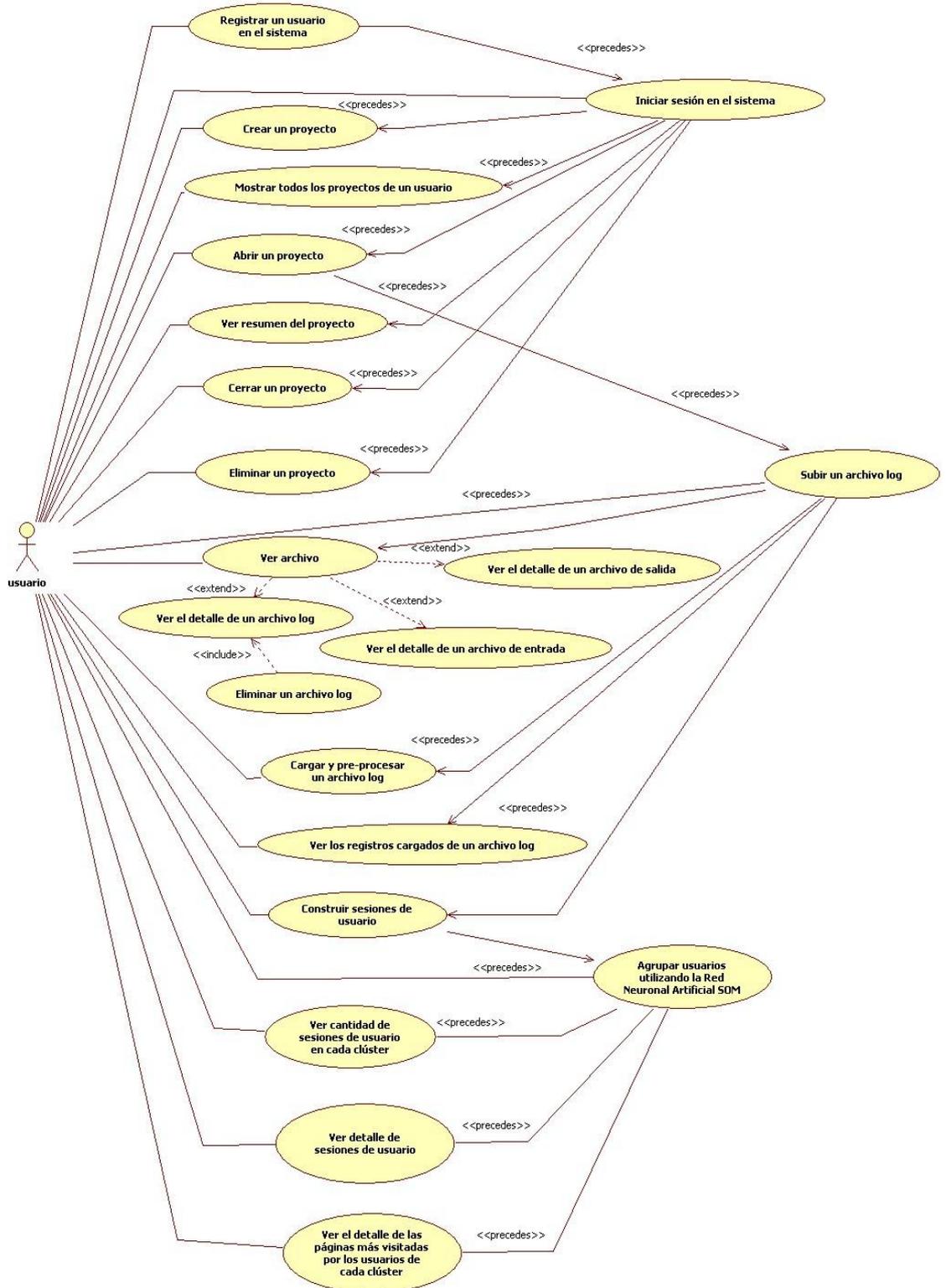


Figura 17: Diagrama de Casos de Uso del Usuario Común



6.3.1.2. Diagrama de Casos de Uso del Usuario Administrador

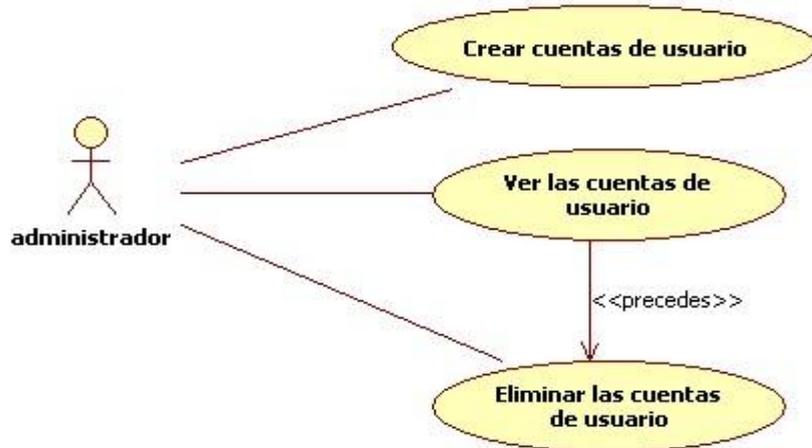


Figura 18: Diagrama de Casos de Uso del Usuario Administrador

6.3.2. Descripción de Casos de Uso

6.3.2.1. Caso de Uso: Registrar un usuario en el sistema

Pantallas



Figura 19: Página Principal del Sistema



Registro de Usuario

Ingrese sus datos para registrarse en el Sistema

Rol: * Administrador

Nombre: *

Apellido: *

Cargo: Administrativo

E-mail: *

Teléfono:

Nombre de usuario: *

Contraseña: *

Repetir la contraseña: *

(*) Campos obligatorios

Aceptar Cancelar

Figura 20: Página “Registro de Usuario”



Figura 21: Mensaje de Registro de Usuario



NOMBRE DEL CASO DE USO: REGISTRAR UN USUARIO EN EL SISTEMA		CÓDIGO: C.U.001
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 001 - RF 002	- Usuario Común - Usuario Administrador	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Capturar la información de un nuevo usuario.	El usuario se registra en el sistema ingresando sus datos personales, un nombre de usuario y contraseña para ser identificado en el mismo y en adelante poder iniciar una sesión.	
PRE-CONDICIONES	POST-CONDICIONES	
- El sistema permite el registro de un usuario. - El usuario debe estar visualizando la página principal del sistema la cual contiene el link de registro de usuario.	Se ha registrado un nuevo usuario en el sistema.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<p>1. El usuario hace clic en el link “regístrese aquí” de la página principal del sistema.</p> <p>3. El usuario ingresa en el formulario los siguientes datos: rol (Administrador o Común), nombre, apellido, cargo (Administrativo, Docente, Estudiante), e-mail, teléfono, nombre de usuario, contraseña, la repetición de la misma para chequear que haya sido ingresada correctamente.</p> <p>4. El usuario hace clic en el botón “Aceptar”.</p> <p>8. El usuario hace clic en “Aceptar”.</p>	<p>2. El sistema muestra el formulario “Registro de Usuario” para que el usuario ingrese sus datos.</p> <p>5. El sistema verifica que los campos obligatorios se hayan completado.</p> <p>6. El sistema crea el directorio del usuario con su nombre de usuario en C:\MiAplicacionWeb\Usuarios.</p> <p>7. El sistema muestra el mensaje “Se está registrando como nuevo usuario en el Sistema”.</p> <p>9. El sistema muestra la página “Iniciar Sesión”.</p>
FLUJO ALTERNO	
<p>A. EL USUARIO NO DESEA REGISTRARSE</p> <p>A4. El usuario hace clic en el botón “Cancelar”</p> <p>A5. El Caso de uso termina.</p> <p>B. CAMPOS OBLIGATORIOS INCOMPLETOS</p> <p>B6. El Caso de uso continúa en el paso 3 del curso normal.</p>	



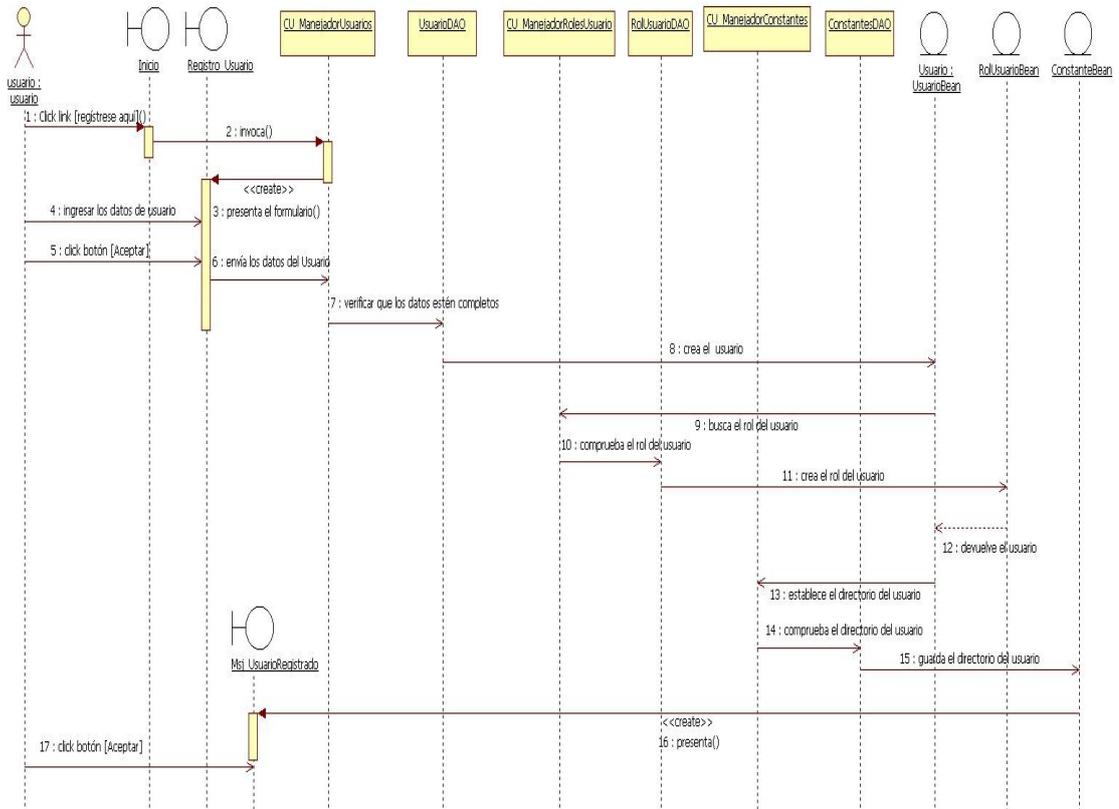


Figura 22: Diagrama de secuencia Caso de Uso: Registrar un usuario en el sistema (Flujo normal)

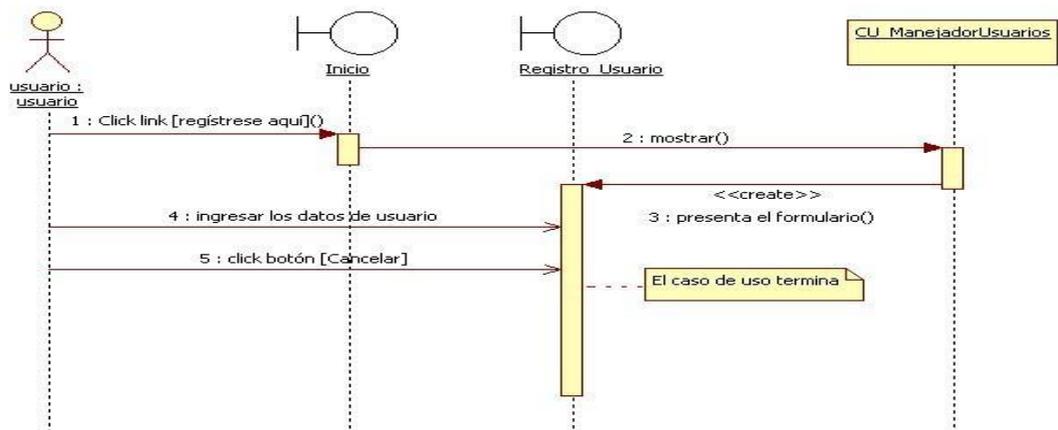


Figura 23: Diagrama de secuencia Caso de Uso: Registrar un usuario en el sistema (Flujo alternativo A)



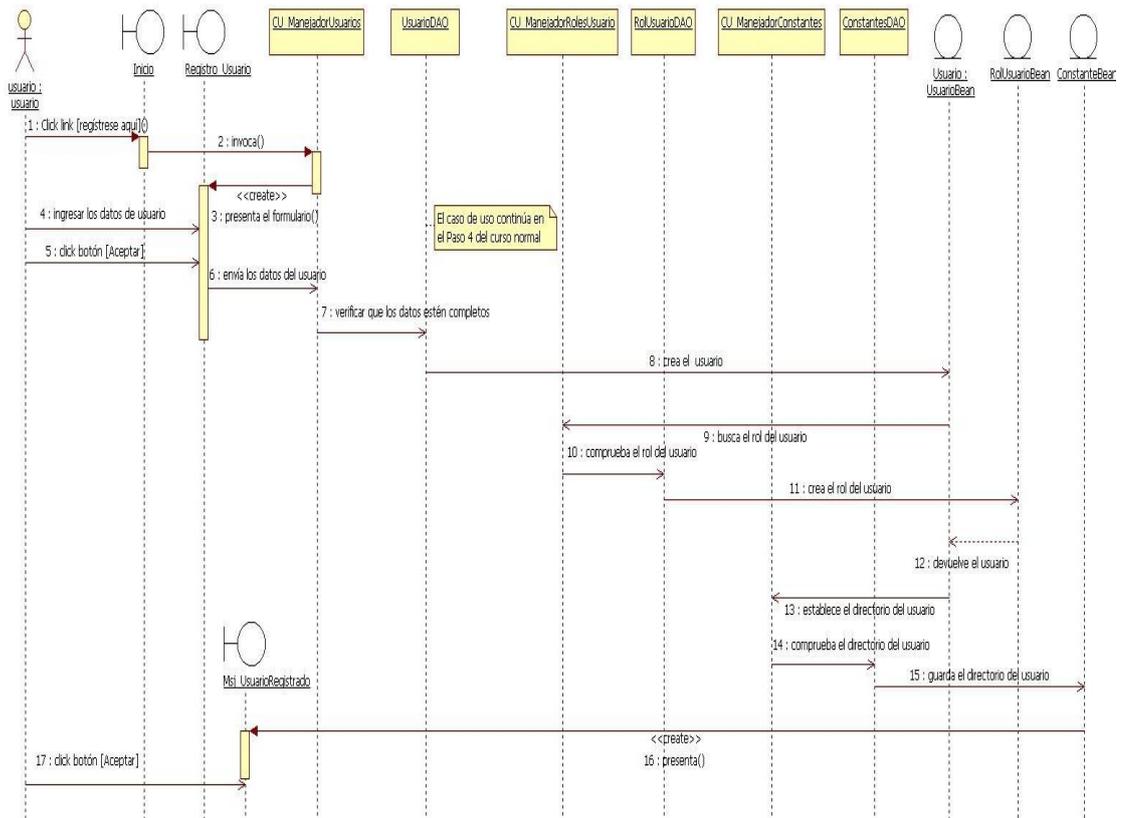


Figura 24: Diagrama de secuencia Caso de Uso: Registrar un usuario en el sistema (Flujo alternativo B)

6.3.2.2. Caso de Uso: Iniciar sesión en el sistema

Pantalla

Iniciar Sesión

Nombre de usuario:

Contraseña:

Figura 25: Página “Iniciar Sesión”



NOMBRE DEL CASO DE USO: INICIAR SESIÓN EN EL SISTEMA		CÓDIGO: C.U.002
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 001	- Usuario Común - Usuario Administrador	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Acceder al Sistema.	El usuario inicia sesión en el sistema previo su registro en el mismo.	
PRE-CONDICIONES	POST-CONDICIONES	
El usuario debe estar registrado en el sistema.	Se ha verificado el registro de usuario y se ha iniciado una nueva sesión.	
FLUJO NORMAL		
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA	
<ol style="list-style-type: none"> 1. El usuario ingresa en el formulario “Iniciar Sesión” su nombre de usuario y contraseña. 2. El usuario hace clic en el botón “Aceptar” para iniciar su sesión. 	<ol style="list-style-type: none"> 3. El sistema verifica que los campos obligatorios estén completos. 4. El sistema autentica al usuario comprobando que esté registrado. 5. El sistema crea la nueva sesión de usuario. 6. Si la autenticación es exitosa, el usuario inicia su sesión. 7. El sistema muestra la página principal del usuario. 	



FLUJO ALTERNO

A. CAMPOS OBLIGATORIOS INCOMPLETOS

A4. El Caso de uso continúa en el paso 1 del curso normal.

B. USUARIO NO ESTÁ REGISTRADO

B5. El sistema muestra la página principal del sistema para que el usuario pueda registrarse.

B6. El Caso de uso termina.

C. EL USUARIO NO DESEA INICIAR SESIÓN

C2. El usuario hace clic en el botón “**Cancelar**”

C3. El Caso de uso termina.

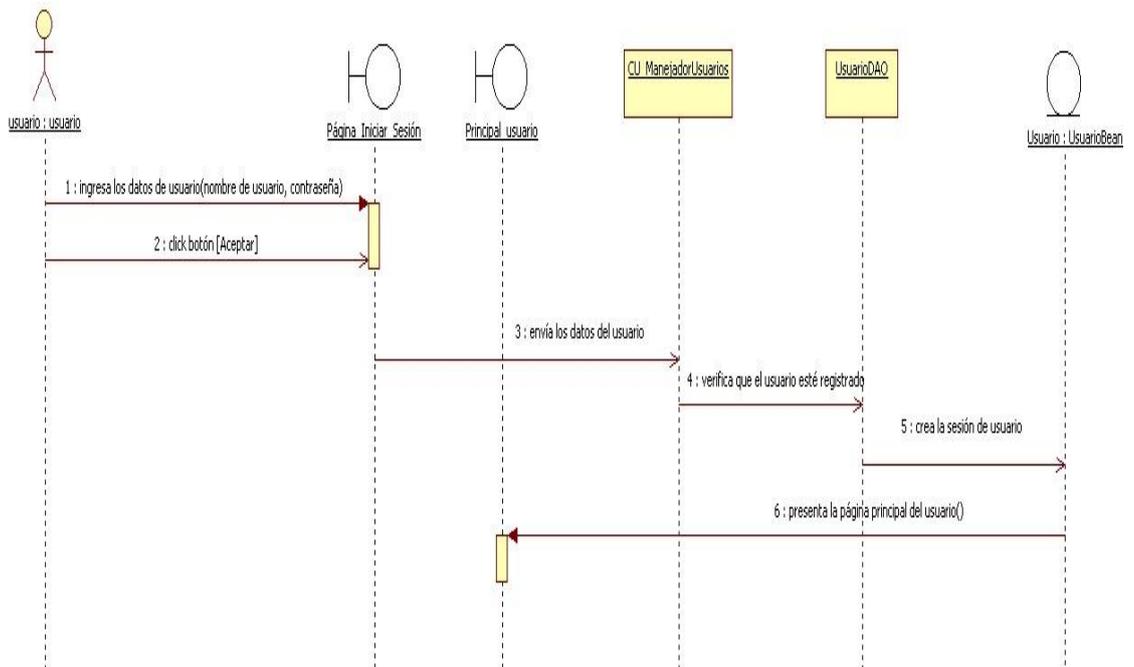


Figura 26: Diagrama de secuencia Caso de Uso: Iniciar sesión en el sistema (Flujo normal)



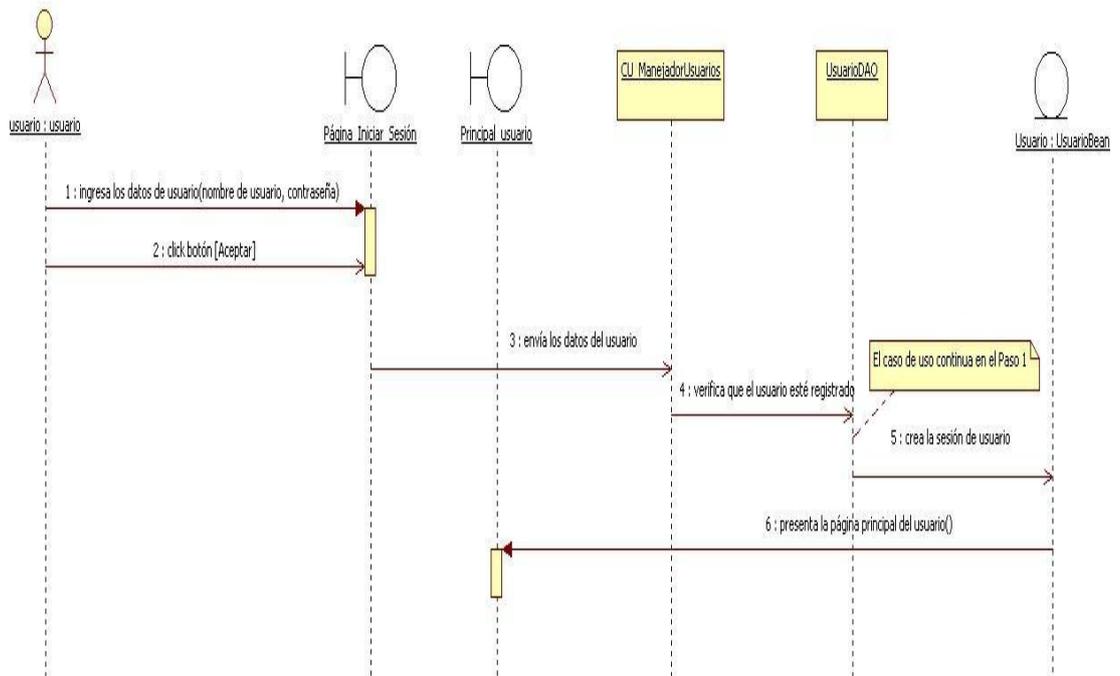


Figura 27: Diagrama de secuencia Caso de Uso: Iniciar sesión en el sistema (Flujo alternativo A)

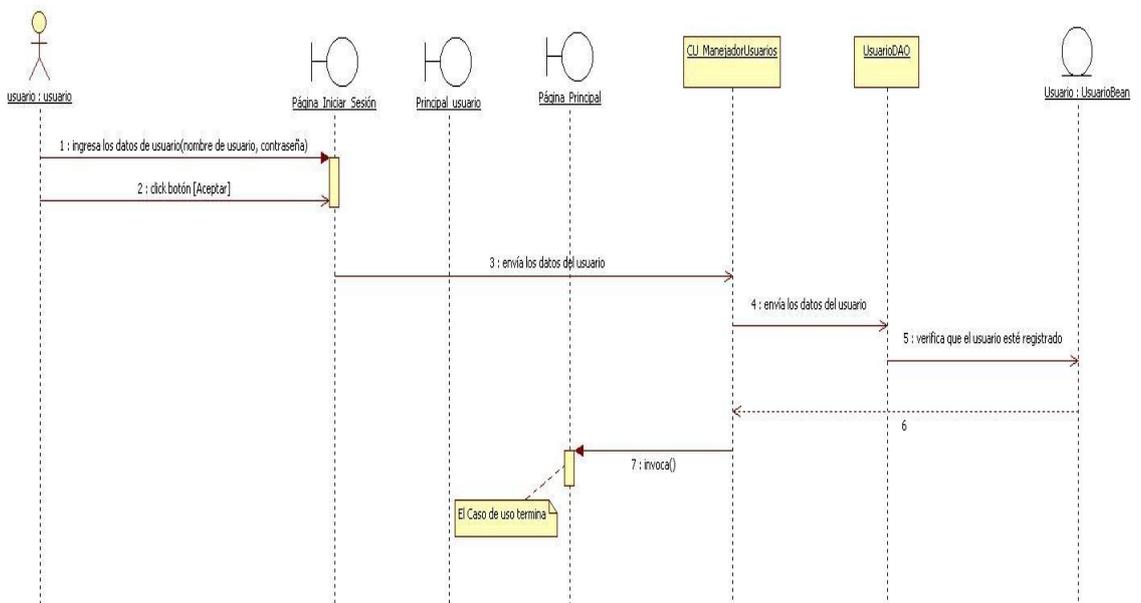


Figura 28: Diagrama de secuencia Caso de Uso: Iniciar sesión en el sistema (Flujo alternativo B)



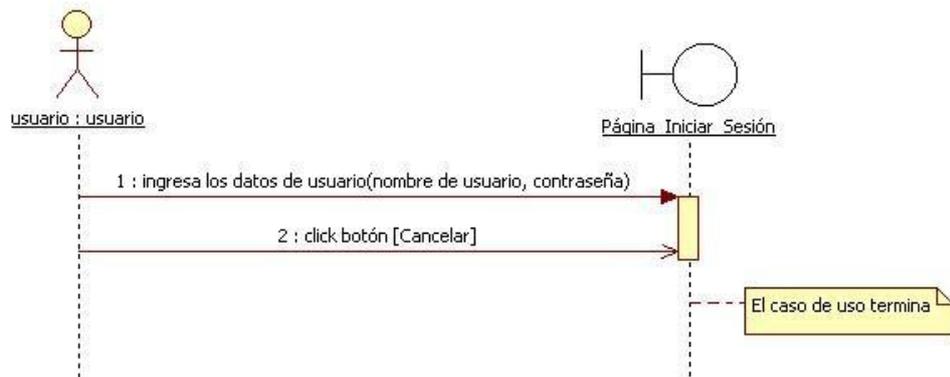


Figura 29: Diagrama de secuencia Caso de Uso: Iniciar sesión en el sistema (Flujo alterno C)

6.3.2.3. Caso de Uso: Crear una cuenta de usuario

Pantalla



Figura 30: Pantalla "Crear una cuenta de usuario"





Figura 31: Mensaje al Crear una cuenta de Usuario

NOMBRE DEL CASO DE USO: CREAR UNA CUENTA DE USUARIO		CÓDIGO: C.U.003
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
<ul style="list-style-type: none"> - RF 001 - RF 002 - RF 003 	- Usuario Administrador	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Capturar la información de un nuevo usuario.	El usuario Administrador tiene la opción de crear nuevas cuentas de usuario ingresando los datos indicados en el formulario “Registro de Usuario” del sistema.	
PRE-CONDICIONES	POST-CONDICIONES	
<ul style="list-style-type: none"> - El usuario debe estar registrado como Administrador. - El usuario debe haber iniciado una sesión en el sistema. - El usuario Administrador debe estar visualizando la página que contiene el menú “Cuentas de Usuario”. 	Se ha registrado un nuevo usuario en el sistema.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
1. El usuario Administrador hace clic en el elemento “Crear nueva” del menú “Cuentas de Usuario” .	2. El sistema muestra el formulario “Registro de Usuario” para que el usuario Administrador ingrese los datos del nuevo usuario.
3. El usuario Administrador ingresa en el formulario los datos correspondientes al nuevo usuario entre los cuales se tiene: rol (Administrador o Común), nombre, apellido, cargo (Administrativo, Docente, Estudiante), e-mail, teléfono, nombre de usuario, contraseña, la repetición de la misma para chequear que haya sido ingresada correctamente.	
4. El usuario Administrador hace clic en el botón “Aceptar” .	5. El sistema verifica que los campos obligatorios se hayan completado. 6. El sistema crea el directorio del usuario con su nombre de usuario en C:\MiAplicacionWeb\Usuarios. 7. El sistema muestra el mensaje “Se está registrando como nuevo usuario en el Sistema” .
8. El usuario Administrador hace clic en “Aceptar” .	



FLUJO ALTERNO

A. EL USUARIO ADMINISTRADOR NO DESEA REGISTRAR UN NUEVO USUARIO

A4. El usuario Administrador hace clic en el botón "Cancelar"

A5. El Caso de uso termina.

B. CAMPOS OBLIGATORIOS INCOMPLETOS

B6. El Caso de uso continúa en el paso 3 del curso normal.

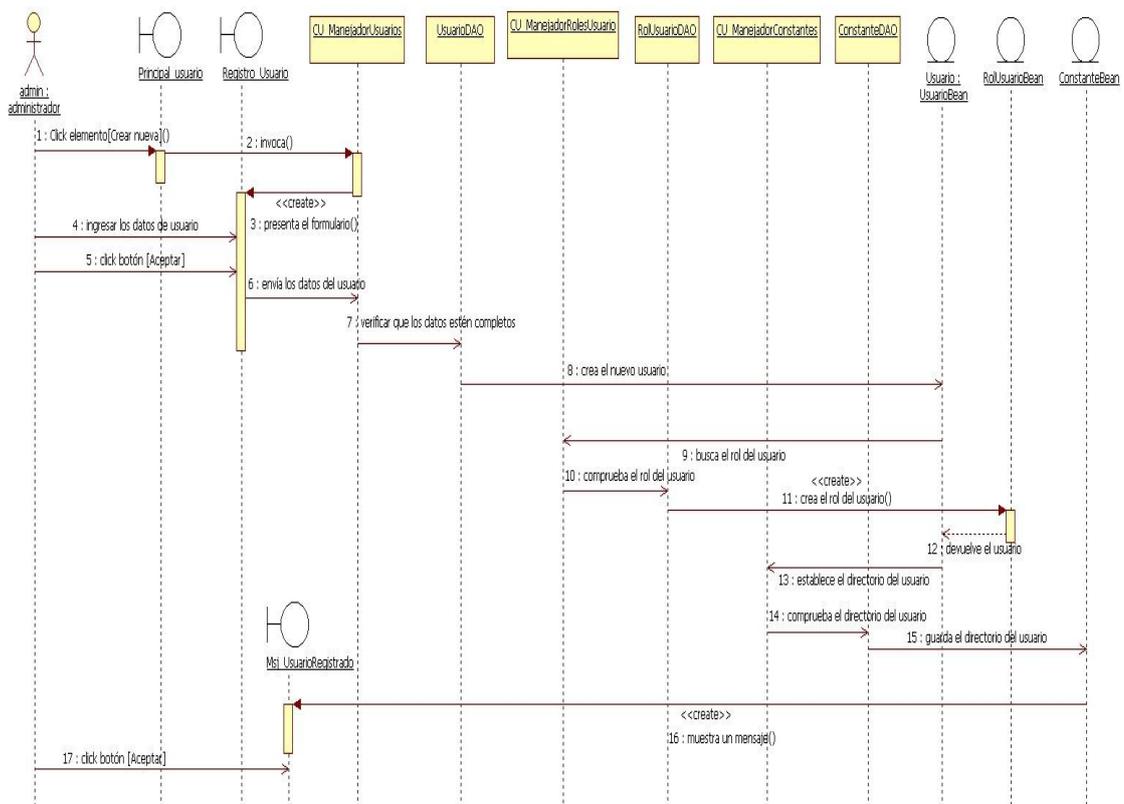


Figura 32: Diagrama de secuencia Caso de Uso: Crear una cuenta de usuario (Flujo normal)



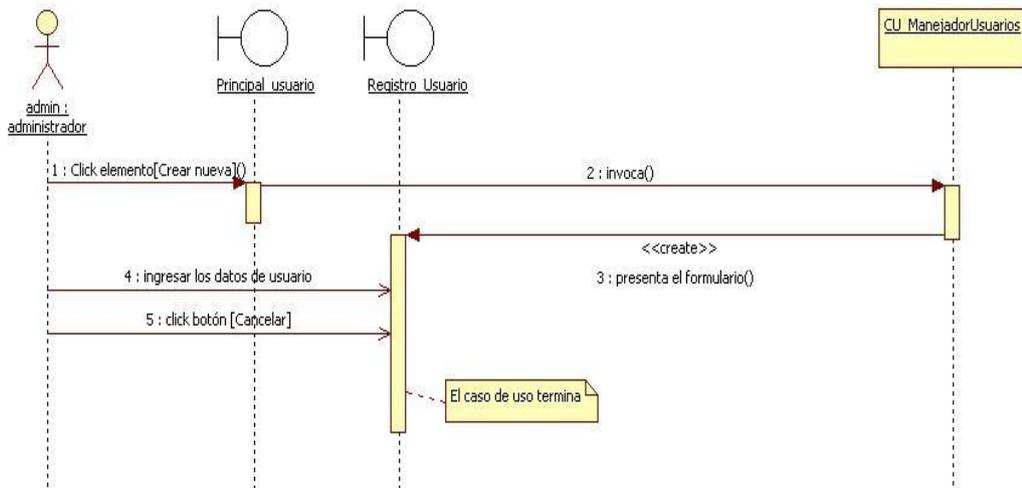


Figura 33: Diagrama de secuencia Caso de Uso: Crear una cuenta de usuario (Flujo alternativo A)

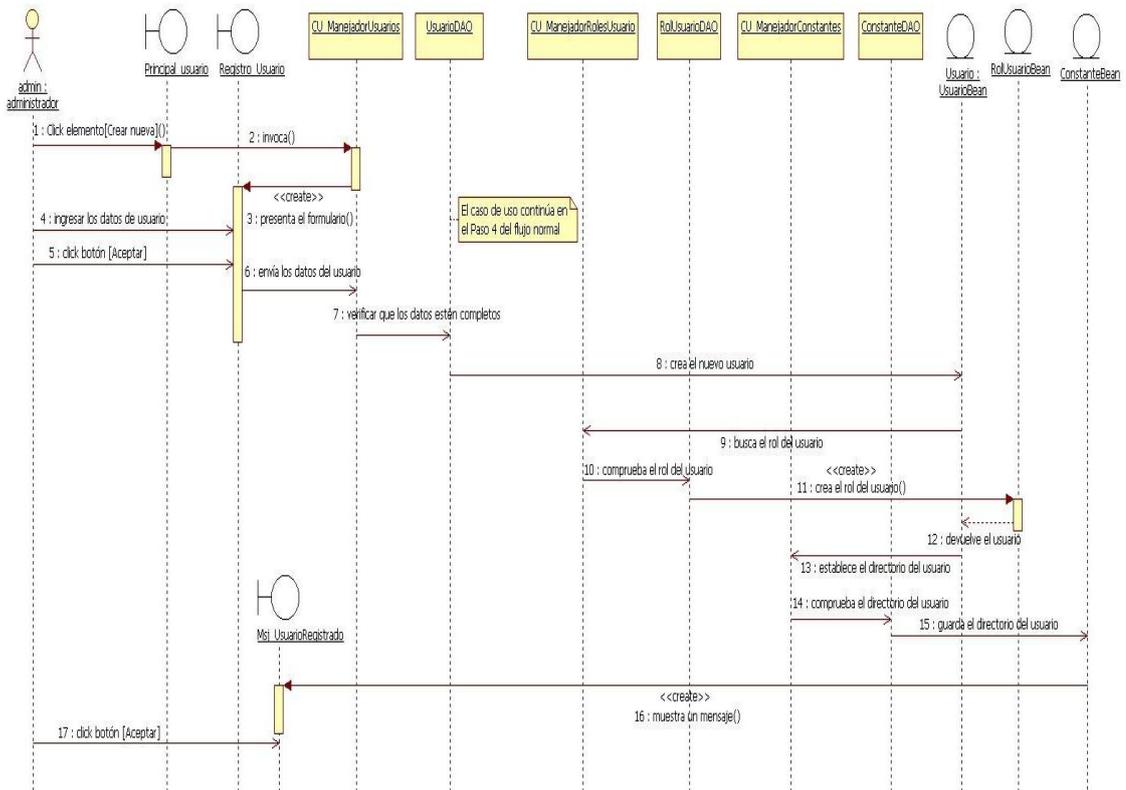


Figura 34: Diagrama de secuencia Caso de Uso: Crear una cuenta de usuario (Flujo alternativo B)



6.3.2.4. Caso de Uso: Ver las cuentas de usuario

Pantalla

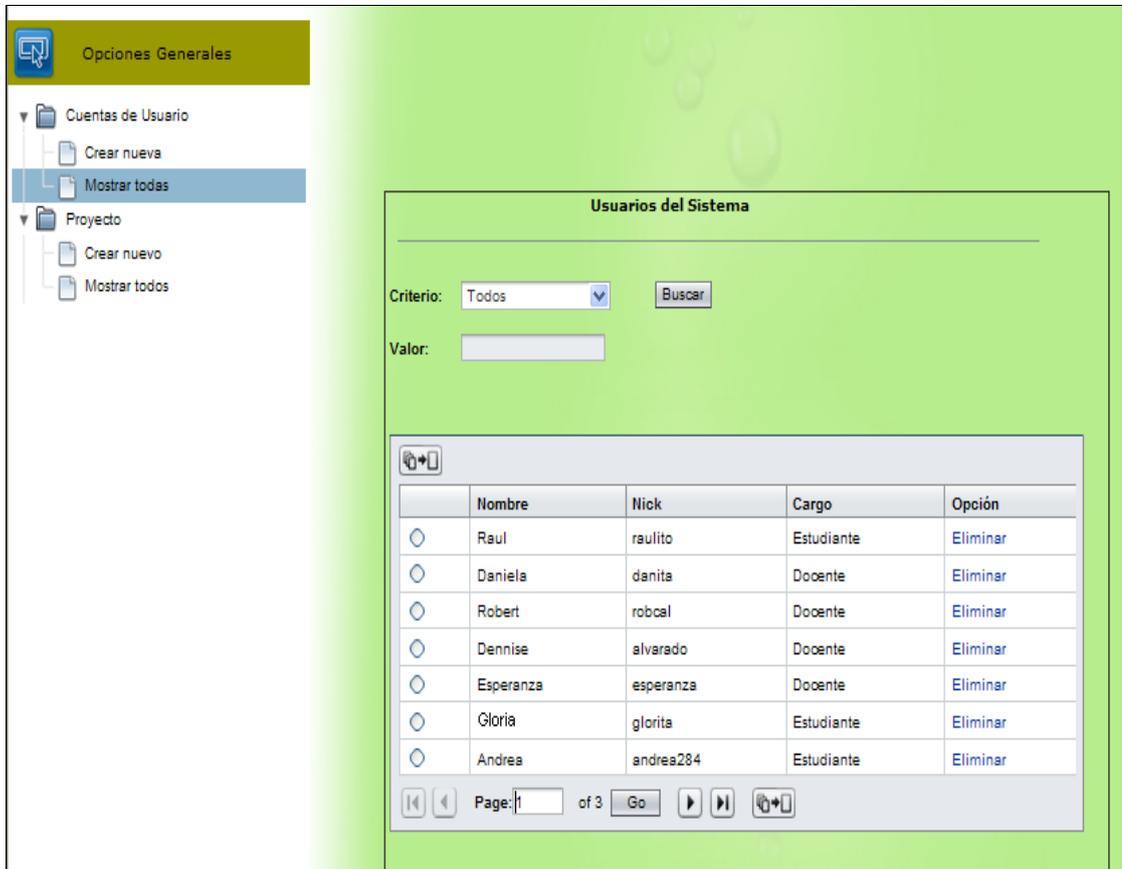


Figura 35: Página “Usuarios del Sistema”



NOMBRE DEL CASO DE USO VER LAS CUENTAS DE USUARIO		CÓDIGO: C.U.004
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 004	- Usuario Administrador	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Listar todos los usuarios registrados en el sistema.	El usuario Administrador registrado y que ha iniciado sesión obtiene una lista de todos los usuarios registrados en el sistema para poder administrar las cuentas de los mismos.	
PRE-CONDICIONES	POST-CONDICIONES	
<ul style="list-style-type: none"> - El usuario debe estar registrado como Administrador. - El usuario debe haber iniciado una sesión en el sistema. - El usuario Administrador debe estar visualizando la página que contiene el menú “Cuentas de Usuario”. 	Se ha obtenido un listado con todos los usuarios registrados en el sistema.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<p>1. El usuario Administrador hace clic en el elemento “Mostrar todas” del menú “Cuentas de Usuario”.</p> <p>4. El usuario Administrador puede seleccionar un valor de la opción “Criterio” (Todos, Nombre, Nombre de usuario, Cargo).</p> <p>5. El usuario ingresa en el campo “Valor” el texto que corresponda con la opción que seleccionó en “Criterio”.</p> <p>6. El usuario Administrador hace clic en el botón “Buscar”</p>	<p>2. El sistema muestra la página “Usuarios del Sistema” con la opción “Eliminar”.</p> <p>3. El sistema muestra en pantalla la opción “Criterio” para buscar cuentas.</p> <p>7. El sistema verifica que el campo “Valor” con el texto ingresado sea válido.</p> <p>8. El sistema realiza la búsqueda de la o las cuentas de usuario según los parámetros ingresados por el usuario Administrador.</p> <p>9. El sistema presenta en el listado la o las cuentas que cumplieron con los parámetros ingresados.</p>
FLUJO ALTERNO	
<p>A. ERROR AL INGRESAR LOS DATOS</p> <p>A9. El sistema muestra el mensaje de información "No existen usuarios".</p> <p>A10. El Caso de uso continúa en el paso 4 del curso normal.</p>	



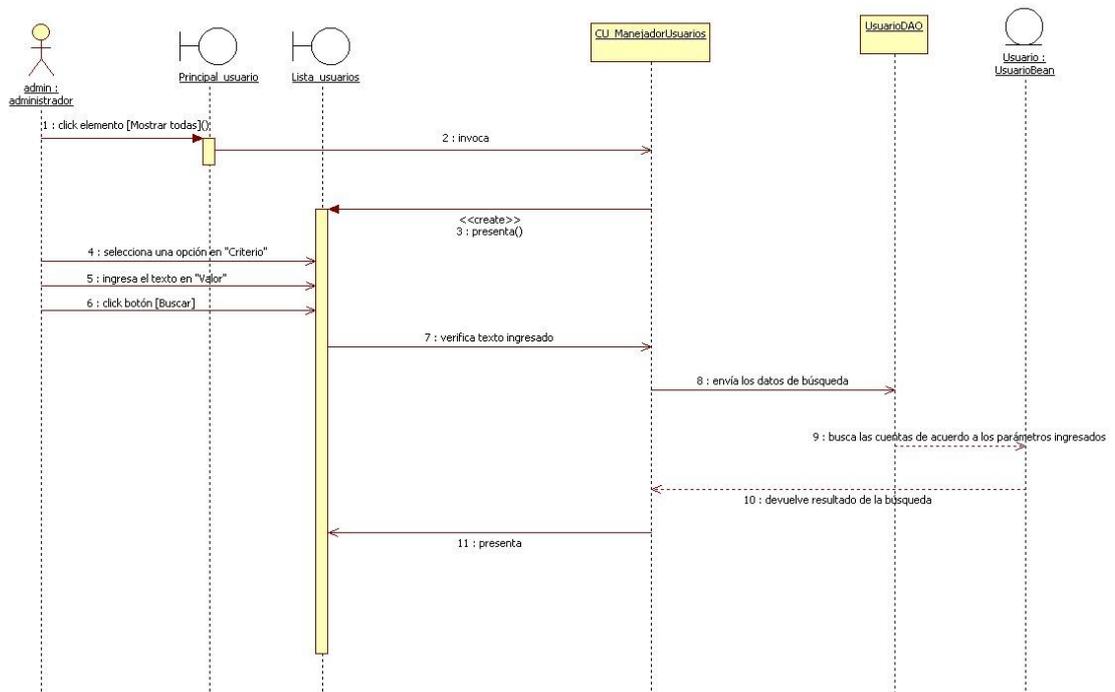


Figura 36: Diagrama de secuencia Caso de Uso: Ver las cuentas de usuario (Flujo normal)

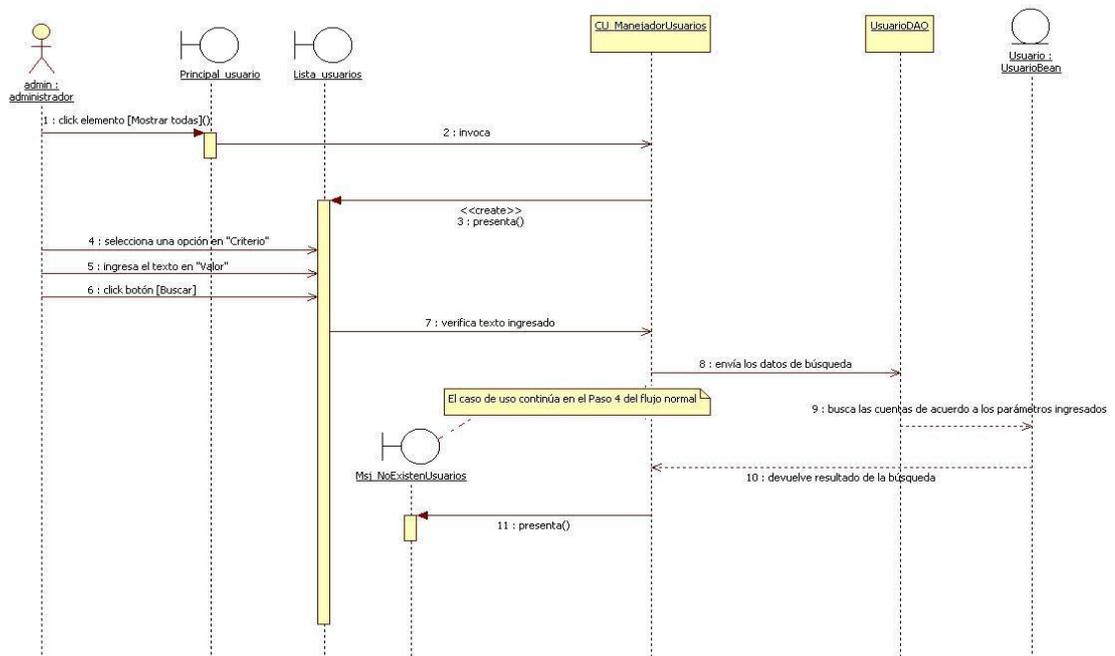


Figura 37: Diagrama de secuencia Caso de Uso: Ver las cuentas de usuario (Flujo alterno A)



6.3.2.5. Caso de Uso: Eliminar las cuentas de usuario

Pantalla

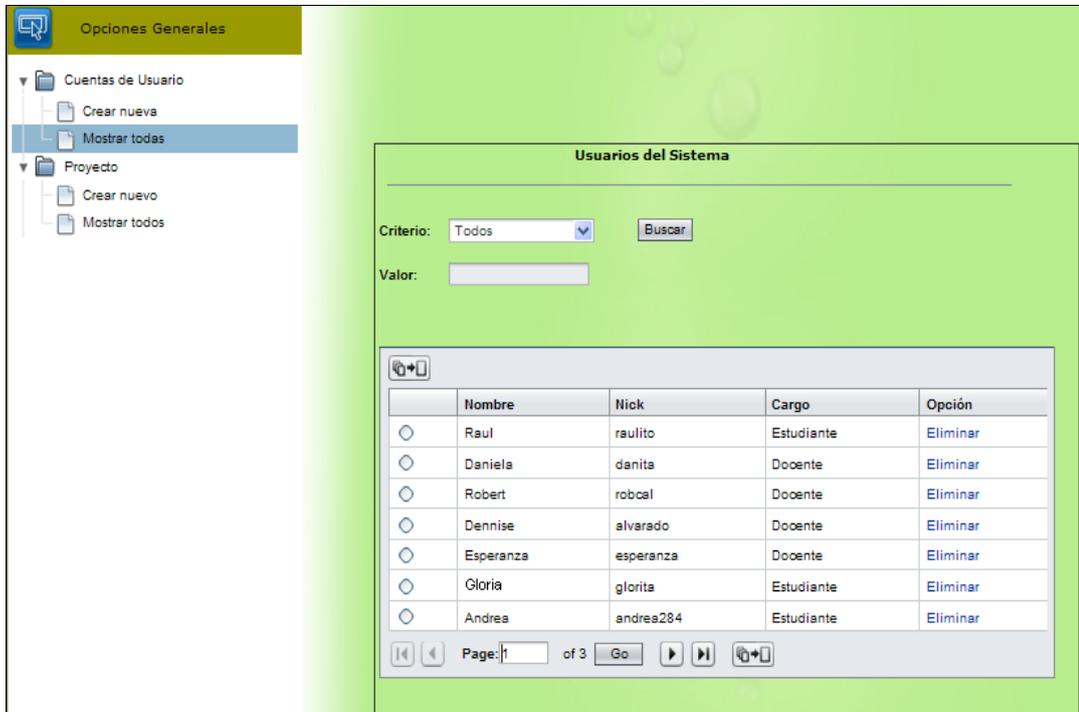


Figura 38: Pantalla “Eliminar cuentas de Usuario”

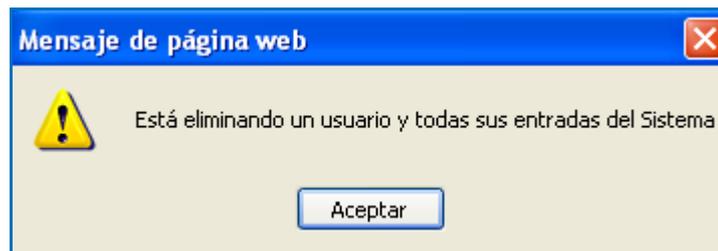


Figura 39: Mensaje de Eliminación de una cuenta de usuario



NOMBRE DEL CASO DE USO: ELIMINAR LAS CUENTAS DE USUARIO		CÓDIGO: C.U.005
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 005	- Usuario Administrador	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Borrar una o más cuentas de usuario del sistema.	El usuario Administrador registrado y que ha iniciado sesión en el sistema selecciona uno o más usuarios de la lista de cuentas de usuario y procede a su eliminación.	
PRE-CONDICIONES	POST-CONDICIONES	
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema como Administrador. - El usuario debe haber iniciado sesión. - El usuario debe estar visualizando la página que contiene la lista de cuentas de usuario del sistema. 	Se han eliminado una o más cuentas de usuario en el sistema.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<p>1. El usuario Administrador selecciona de la lista “Usuarios del Sistema” una o más cuentas para ser eliminadas.</p> <p>2. El usuario Administrador hace clic en el link “Eliminar”.</p> <p>4. El usuario Administrador hace clic en el botón “Aceptar”.</p>	<p>3. El sistema muestra el mensaje “Está eliminando un usuario y todas sus entradas del Sistema”.</p> <p>5. El sistema elimina la o las cuentas de usuario seleccionadas por el usuario Administrador borrando toda la información de las mismas.</p> <p>6. El sistema actualiza la vista de la lista de usuarios.</p>

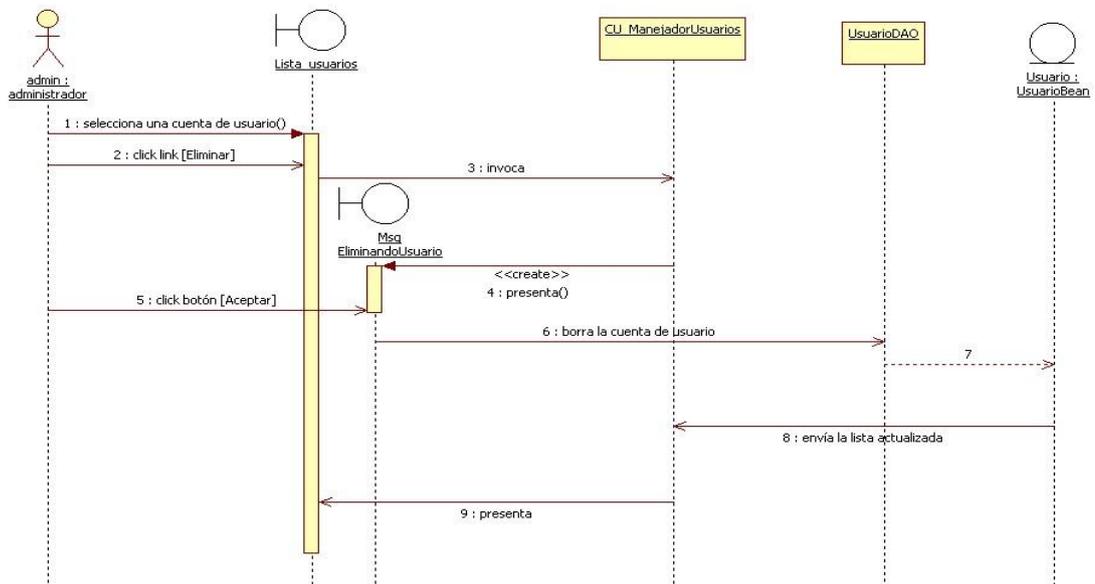


Figura 40: Diagrama de secuencia Caso de Uso: Eliminar las cuentas de usuario



6.3.2.6. Caso de Uso: Crear un proyecto

Pantallas



Figura 41: Página “Crear Nuevo Proyecto (Usuario Administrador)”

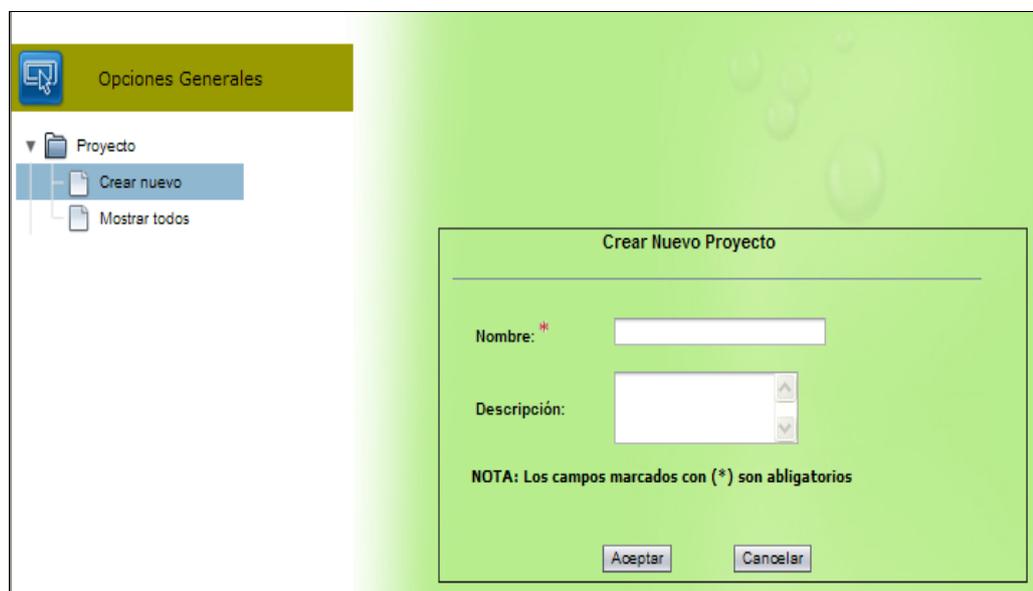


Figura 42: Página “Crear Nuevo Proyecto (Usuario Común)”





Figura 43: Mensaje al crear un proyecto

NOMBRE DEL CASO DE USO: CREAR UN PROYECTO		CÓDIGO: C.U.006
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 006	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Crear un nuevo proyecto para el usuario.	El usuario registrado y que ha iniciado sesión en el sistema crea un nuevo proyecto al cual le asigna un nombre y una descripción para posteriormente poder ejecutar el análisis minucioso de un archivo log del sitio web de la Universidad Nacional de Loja.	
PRE-CONDICIONES	POST-CONDICIONES	
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe estar visualizando la página que contiene el menú "Proyecto". 	Se ha creado un nuevo proyecto para el usuario del sistema.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<p>1. El usuario hace clic en el elemento “Crear nuevo” del menú “Proyecto”.</p> <p>3. El usuario ingresa en el formulario el nombre y la descripción del proyecto a crear.</p> <p>4. El usuario hace clic en el botón “Aceptar”.</p> <p>7. El usuario hace clic en el botón “Aceptar”</p>	<p>2. El sistema muestra el formulario “Crear Nuevo Proyecto” para ingresar los datos del proyecto.</p> <p>5. El sistema verifica que se hayan completado los campos obligatorios.</p> <p>6. El sistema muestra el mensaje “Está creando un nuevo Proyecto”.</p> <p>8. El sistema crea un nuevo proyecto para el usuario.</p> <p>9. El sistema actualiza la lista de “Proyectos”.</p>
FLUJO ALTERNO	
<p>A. CAMPOS OBLIGATORIOS INCOMPLETOS</p> <p>A6. El Caso de uso continúa en el paso 3 del curso normal.</p> <p>B. EL USUARIO NO DESEA CREAR UN NUEVO PROYECTO</p> <p>B4. El usuario hace clic en el botón “Cancelar”</p> <p>B5. El Caso de uso termina.</p>	



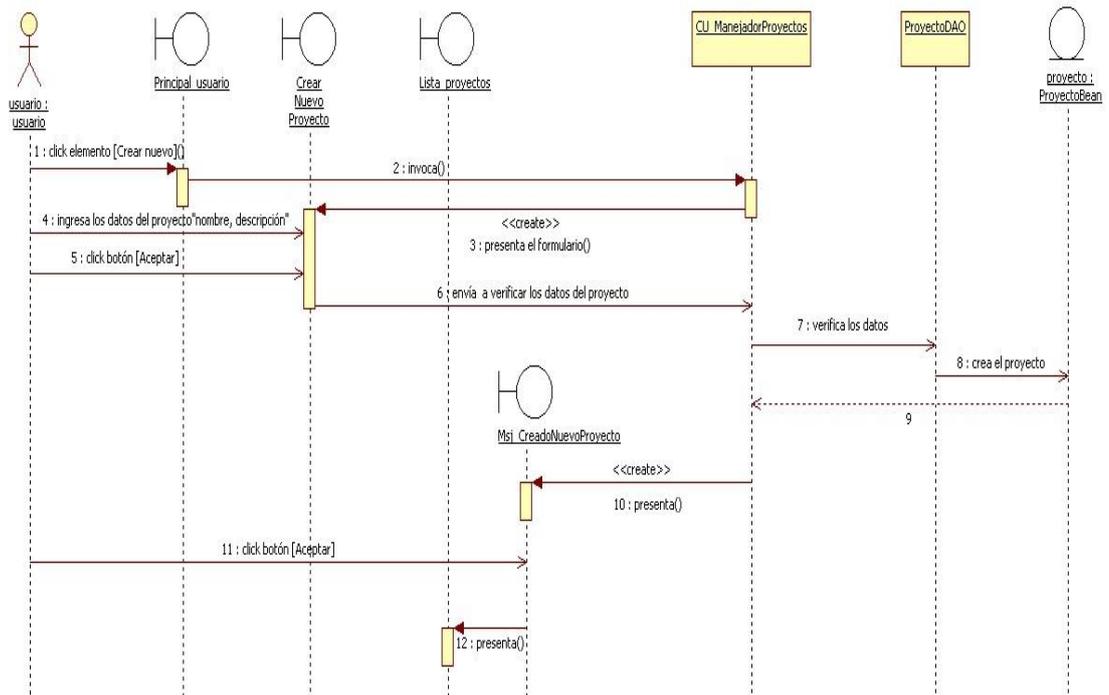


Figura 44: Diagrama de secuencia Caso de Uso: Crear un proyecto (Flujo normal)

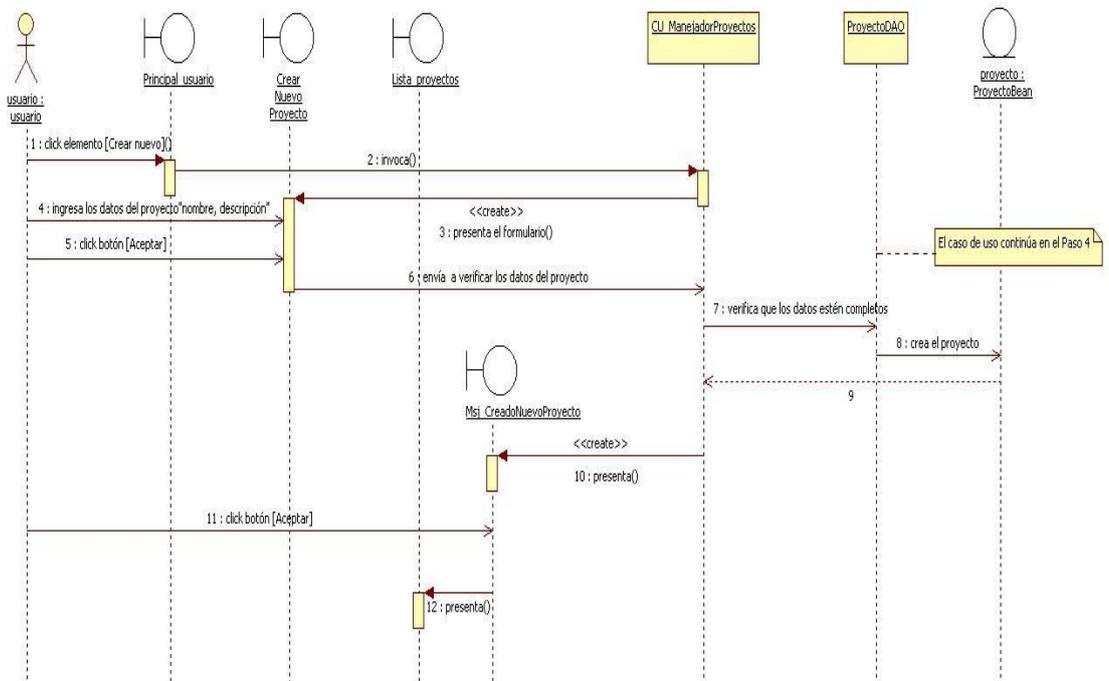


Figura 45: Diagrama de secuencia Caso de Uso: Crear un proyecto (Flujo alternativo A)



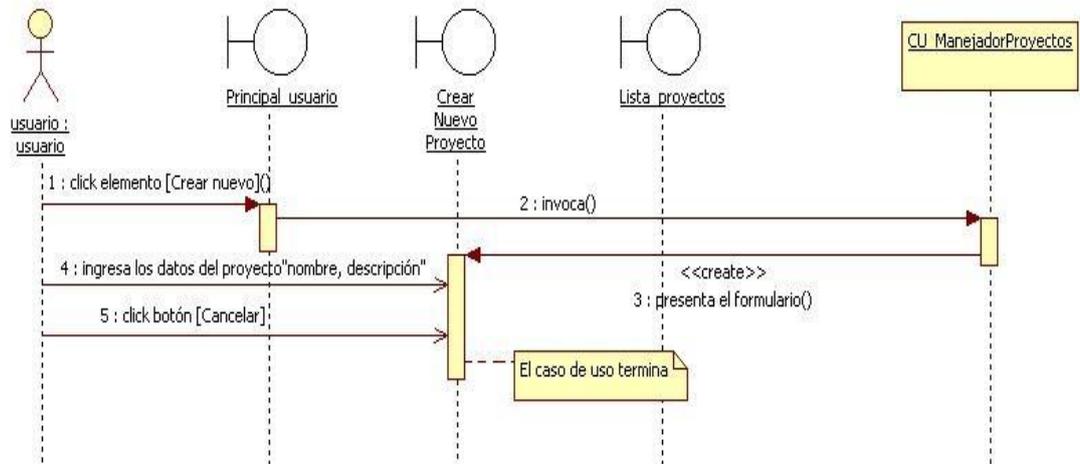


Figura 46: Diagrama de secuencia Caso de Uso: Crear un proyecto (Flujo alternativo B)

6.3.2.7. Caso de Uso: Mostrar todos los proyectos de un usuario

Pantalla

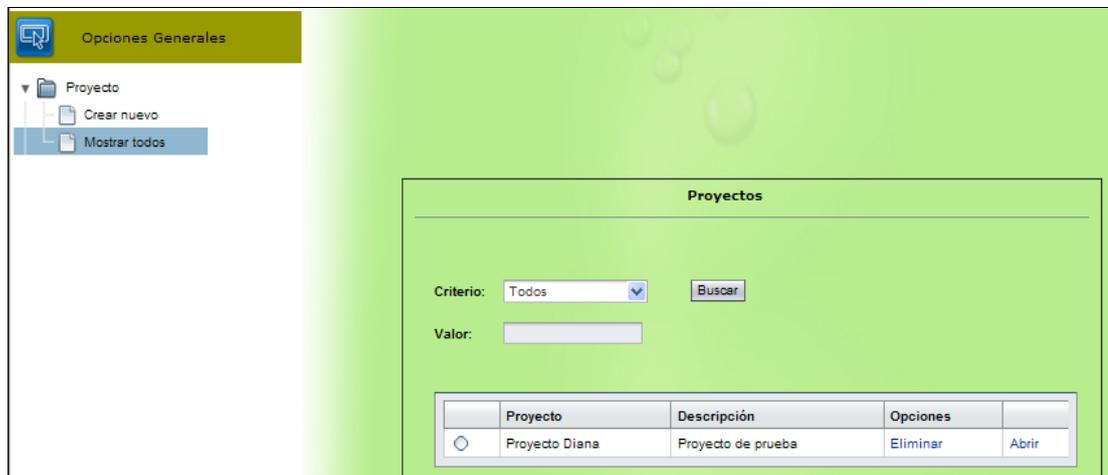


Figura 47: Página "Proyectos"



NOMBRE DEL CASO DE USO: MOSTRAR TODOS LOS PROYECTOS DE UN USUARIO		CÓDIGO: C.U.007
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 007	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Listar todos los proyectos de un usuario.	El usuario registrado que ha iniciado sesión en el sistema obtiene una lista de todos sus proyectos para poder acceder a la administración de los mismos.	
PRE-CONDICIONES	POST-CONDICIONES	
- El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe estar visualizando la página que contiene el menú " Proyecto ".	Se ha obtenido un listado con todos los proyectos de un usuario.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<p>1. El usuario hace clic en el elemento “Mostrar todos” del menú “Proyecto”.</p> <p>4. El usuario puede seleccionar un valor en la opción “Criterio”.</p> <p>5. El usuario ingresa en el campo “Valor” el texto que corresponda con la opción que seleccionó en “Criterio”.</p> <p>6. El usuario hace clic en el botón “Buscar”</p>	<p>2. El sistema muestra la página “Proyectos” con las opciones “Eliminar” y “Abrir”.</p> <p>3. El sistema muestra en pantalla la opción “Criterio” para buscar proyectos.</p> <p>7. El sistema verifica que el campo “Valor” con el texto ingresado sea válido.</p> <p>8. El sistema realiza la búsqueda de los proyectos según los parámetros ingresados por el usuario.</p> <p>9. El sistema presenta en el listado los proyectos que cumplieron con los parámetros ingresados.</p>
FLUJO ALTERNO	
<p>A. ERROR AL INGRESAR LOS DATOS</p> <p>A8. El sistema muestra el mensaje de información "No existen proyectos".</p> <p>A9. El Caso de uso continúa en el paso 3 del curso normal.</p>	



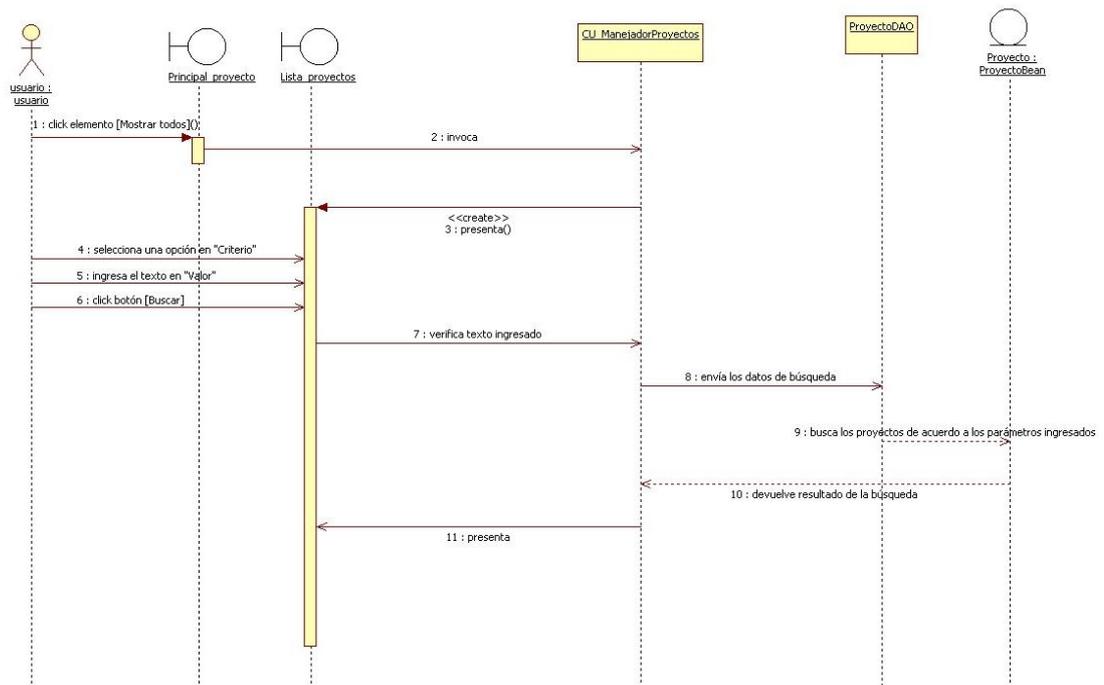


Figura 48: Diagrama de secuencia Caso de Uso: Mostrar todos los proyectos de un usuario (Flujo normal)

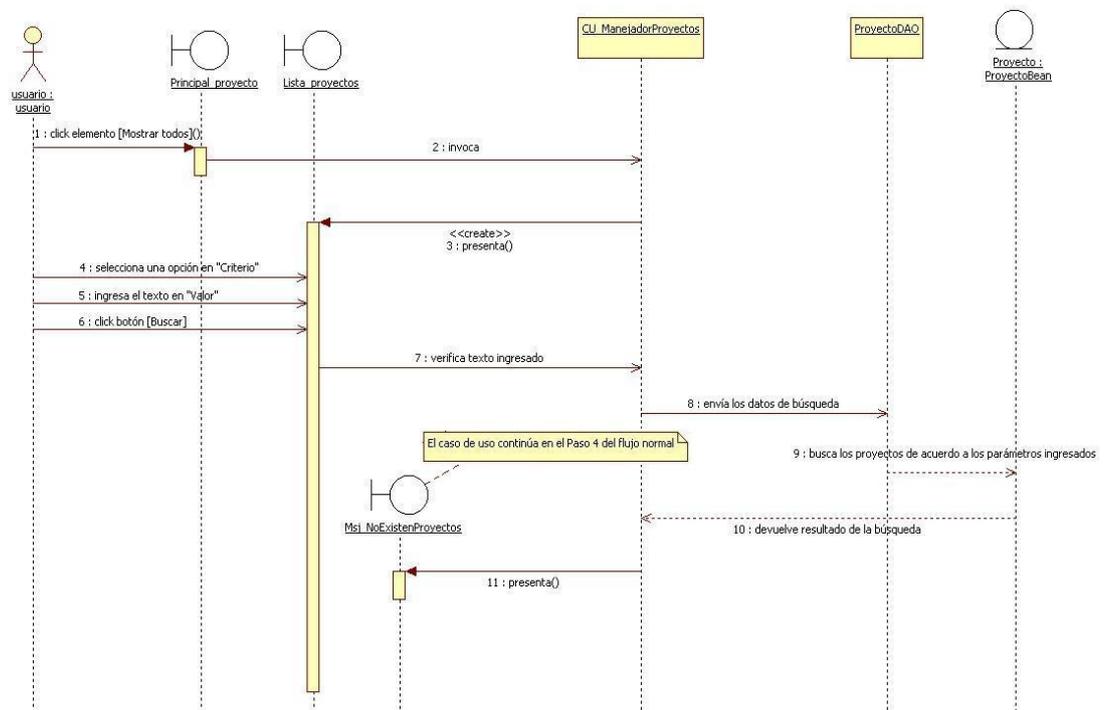


Figura 49: Diagrama de secuencia Caso de Uso: Mostrar todos los proyectos de un usuario



usuario (Flujo alterno A)

6.3.2.8. Caso de Uso: Eliminar un proyecto

Pantalla



Figura 50: Página "Proyectos"

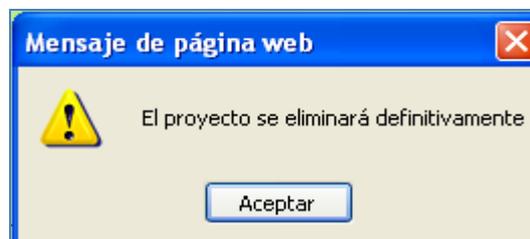


Figura 51: Mensaje de Eliminación de un Proyecto



NOMBRE DEL CASO DE USO: ELIMINAR UN PROYECTO		CÓDIGO: C.U.008
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 011	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Borrar uno o más proyectos.	El usuario registrado y que ha iniciado sesión en el sistema selecciona uno o más proyectos de la lista de proyectos para borrar toda la información relacionada a los mismos.	
PRE-CONDICIONES	POST-CONDICIONES	
- El usuario debe estar registrado. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe estar visualizando la página que contiene el listado de todos sus proyectos.	Se han eliminado uno o más proyectos del usuario.	
FLUJO NORMAL		
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA	
<ol style="list-style-type: none"> 1. El usuario selecciona de la lista "Proyectos" uno o más de sus proyectos para ser eliminados. 2. El usuario hace clic en el link "Eliminar". 4. El usuario hace clic en el botón "Aceptar". 	<ol style="list-style-type: none"> 3. El sistema muestra el mensaje "El proyecto se eliminará definitivamente" 5. El sistema elimina el o los proyectos seleccionados borrando toda la información de los mismos. 6. El sistema actualiza la vista de la lista de proyectos. 	



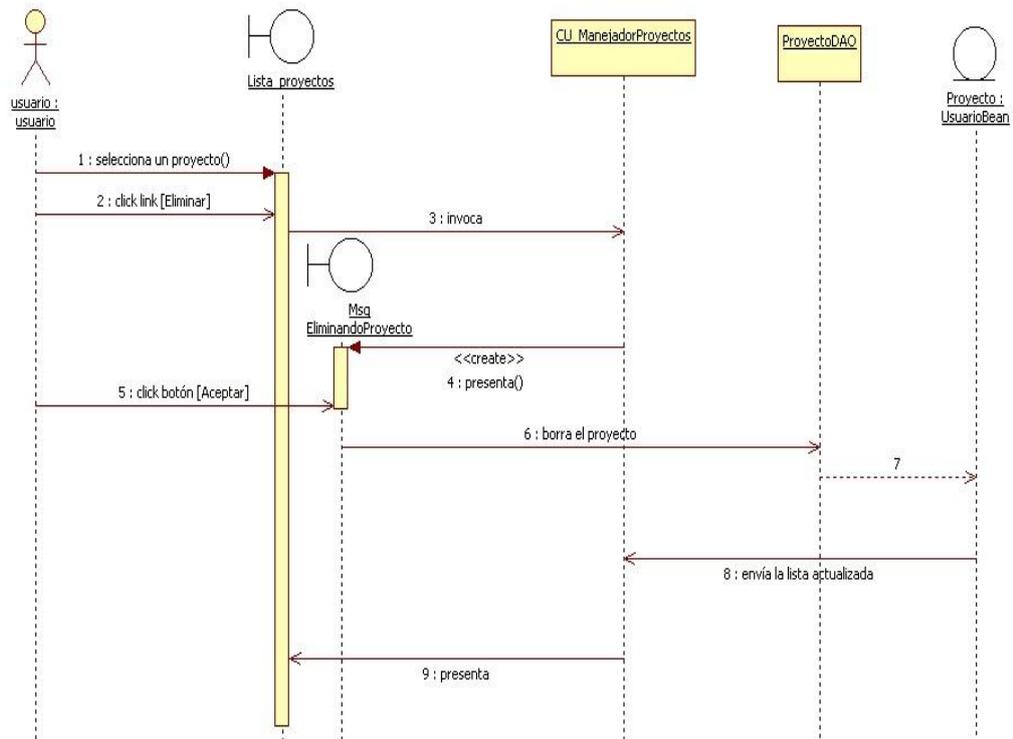


Figura 52: Diagrama de secuencia Caso de Uso: Eliminar un proyecto

6.3.2.9. Caso de Uso: Abrir un proyecto

Pantalla

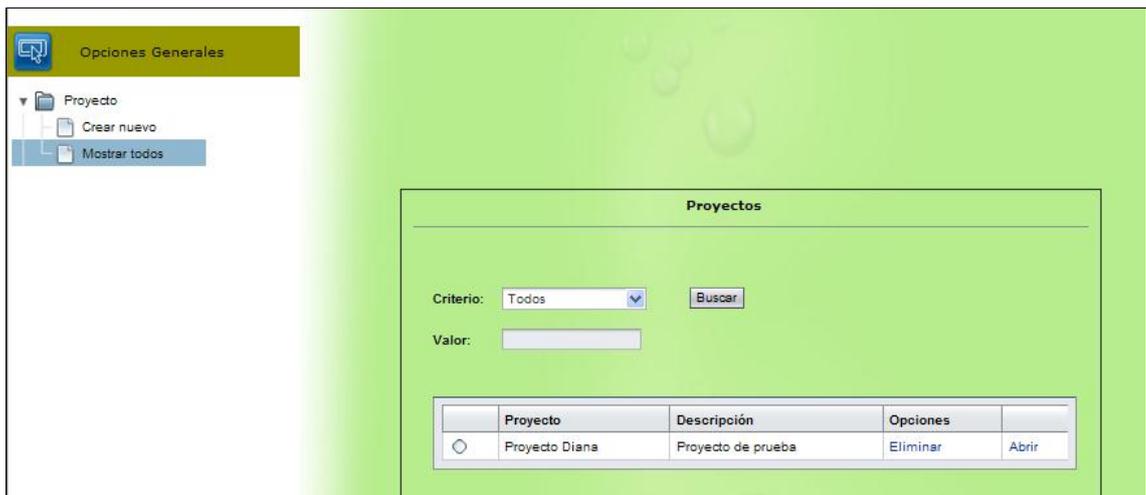


Figura 53: Pantalla “Abrir un Proyecto”



NOMBRE DEL CASO DE USO: ABRIR UN PROYECTO		CÓDIGO: C.U.009
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 008	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Abrir un proyecto.	El usuario registrado que ha iniciado sesión en el sistema selecciona de la lista de proyectos uno de ellos para abrirlo y acceder al menú que permite trabajar en el proyecto seleccionado.	
PRE-CONDICIONES	POST-CONDICIONES	
- El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe estar visualizando la página que contiene el listado de todos sus proyectos.	Se ha establecido un proyecto sobre el cual podrá trabajar el usuario.	
FLUJO NORMAL		
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA	
1. El usuario selecciona de la lista "Proyectos" uno de ellos. 2. El usuario hace clic en el link "Abrir".	3. El sistema establece el proyecto seleccionado como actual. 4. El sistema muestra la pantalla con el menú principal del proyecto.	



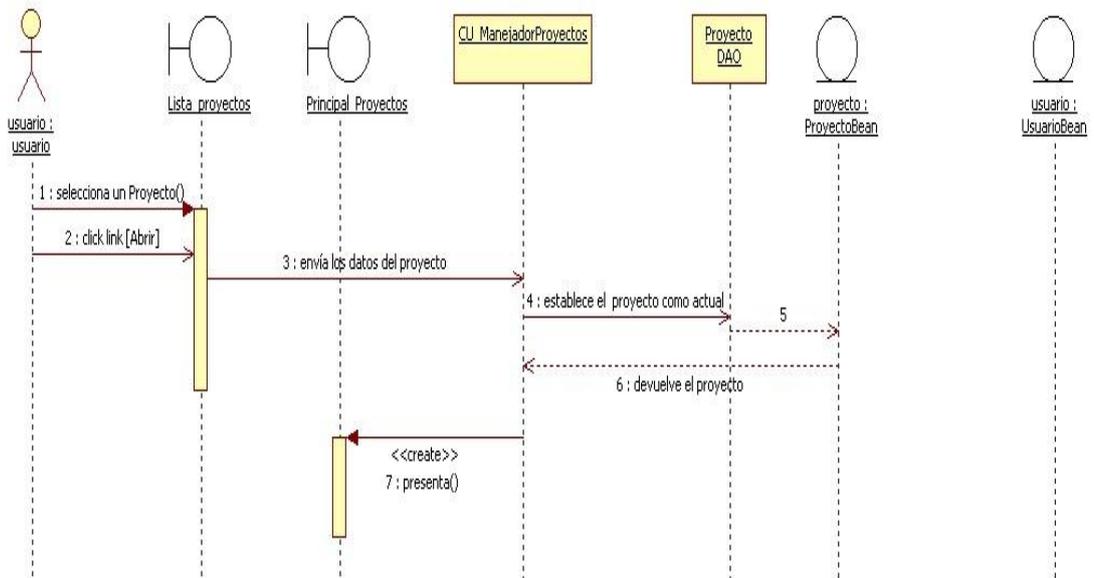


Figura 54: Diagrama de secuencia Caso de Uso: Abrir un proyecto

6.3.2.10. Caso de Uso: Ver el resumen de un proyecto

Pantalla

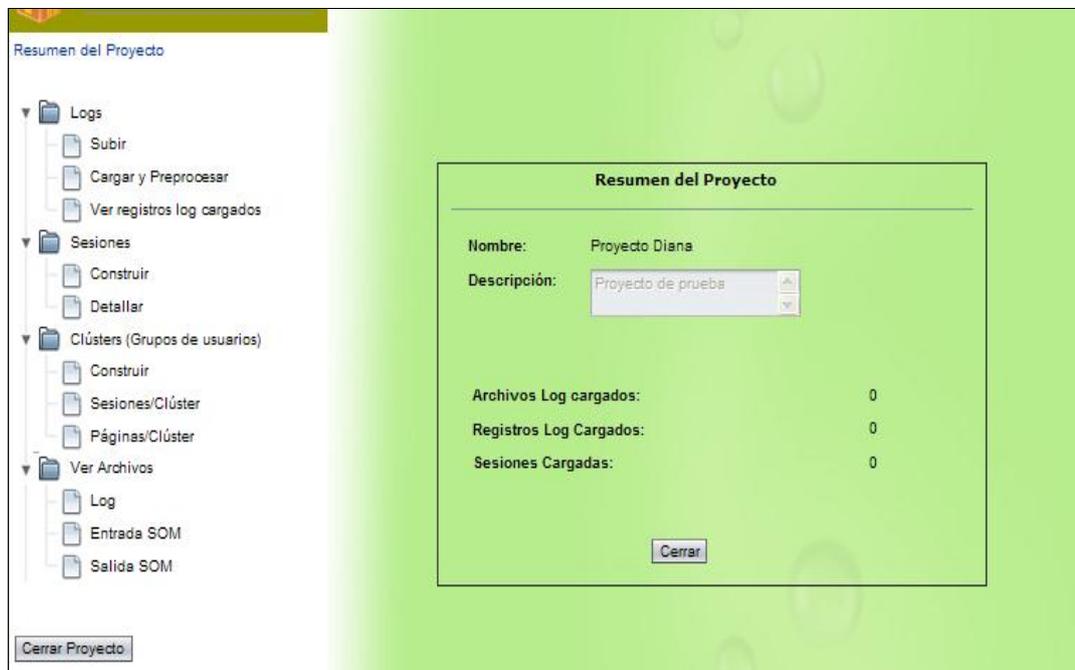


Figura 55: Página “Resumen del Proyecto”



NOMBRE DEL CASO DE USO: VER EL RESUMEN DE UN PROYECTO		CÓDIGO: C.U.010
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 009	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Ver el resumen del proyecto actual.	El usuario registrado y que ha iniciado sesión obtiene información relacionada de su proyecto ingresando al mismo desde la lista de Proyectos.	
PRE-CONDICIONES	POST-CONDICIONES	
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe abrir un proyecto. - El usuario debe estar visualizando la página principal del proyecto. 	Se ha recopilado información sobre el proyecto actual.	
FLUJO NORMAL		
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA	
<ol style="list-style-type: none"> 1. El usuario hace clic en el link “Resumen del Proyecto” ubicado en la parte superior del menú principal del proyecto. 3. El usuario hace clic en el botón “Cerrar”. 	<ol style="list-style-type: none"> 2. El sistema muestra el resumen de la información del proyecto establecido como actual indicando su nombre, descripción, cantidad de archivos log cargados, cantidad de registros log cargados y la cantidad de sesiones cargadas en dicho proyecto. 4. El sistema muestra la pantalla con el menú principal del proyecto. 	



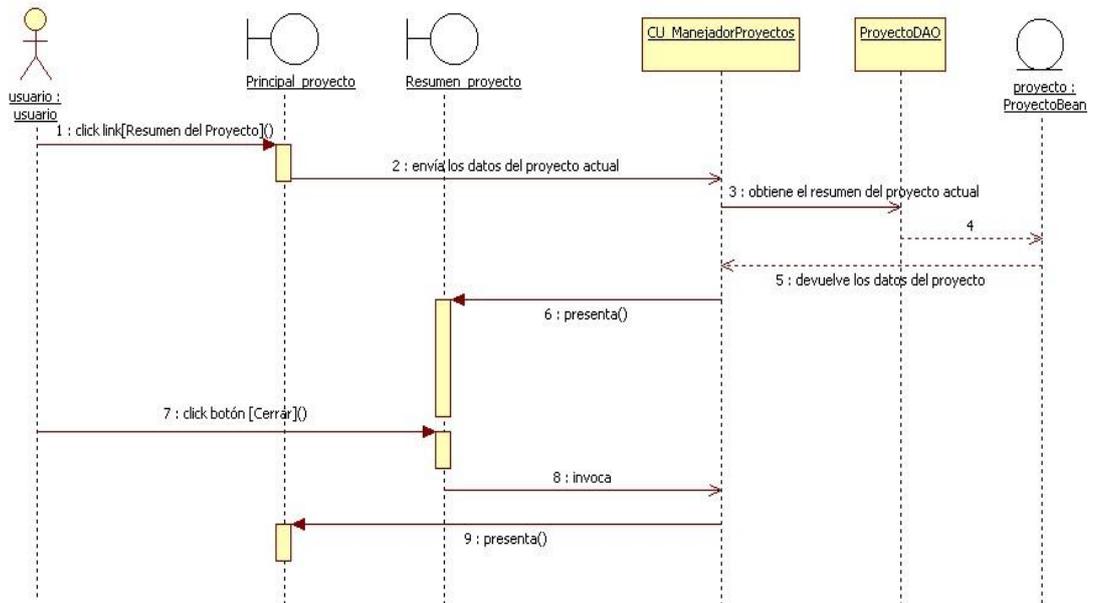


Figura 56: Diagrama de secuencia Caso de Uso: Ver el resumen de un proyecto (Flujo normal)

6.3.2.11. Caso de Uso: Subir un archivo log

Pantallas

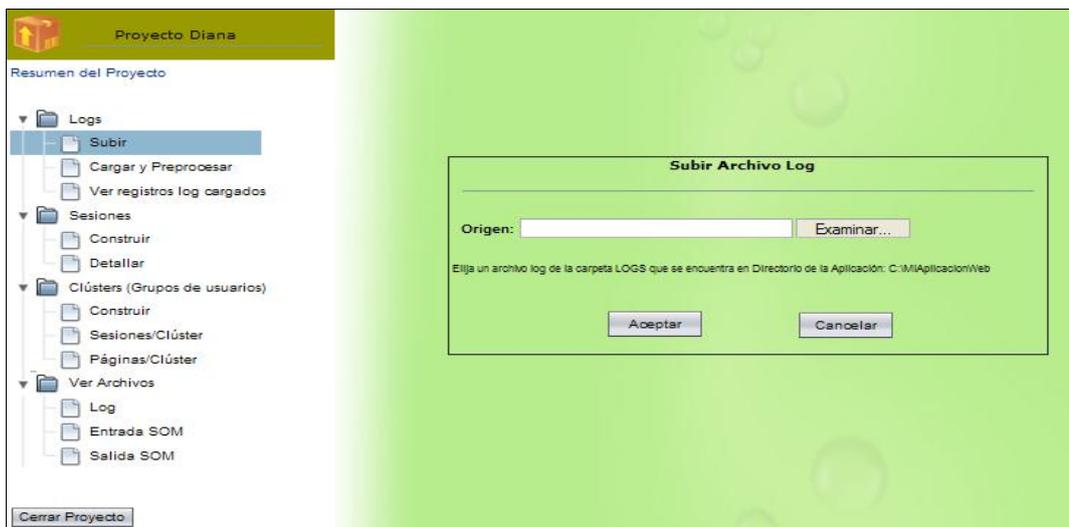


Figura 57: Página “Subir Archivo Log”



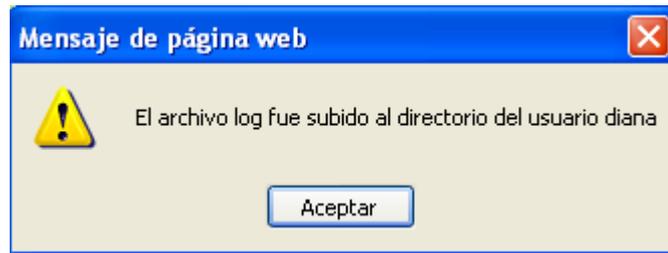


Figura 58: Mensaje al Subir un archivo log



Figura 59: Página “Resultado de Subir el Archivo Log”

NOMBRE DEL CASO DE USO: SUBIR UN ARCHIVO LOG		CÓDIGO: C.U.011
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 012	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Copiar en el directorio del usuario un archivo log.	El usuario registrado y que ha iniciado sesión en el sistema envía una petición a la aplicación para copiar un archivo log a su directorio y posteriormente realizar el análisis del mismo.	



PRE-CONDICIONES	POST-CONDICIONES
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe estar visualizando la página principal del proyecto. 	<p>Se ha copiado un archivo log al directorio del usuario.</p>
FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<ol style="list-style-type: none"> 1. El usuario hace clic en el elemento “Subir” del menú “Logs”. 3. El usuario hace clic en el botón “Examinar” del campo “Origen” para buscar el archivo log a copiar. 4. El usuario hace clic en el botón “Aceptar” 7. El usuario hace clic en el botón “Aceptar” 9. El usuario hace clic en “Cerrar”. 	<ol style="list-style-type: none"> 2. El sistema muestra el formulario “Subir Archivo Log”. 5. El sistema copia el archivo log al directorio del proyecto del usuario. 6. El sistema muestra el mensaje “El archivo log fue subido al directorio del usuario NN” 8. El sistema muestra la página “Resultado de Subir el Archivo Log” indicando su nombre, path de origen y path de destino. 10. El sistema muestra la pantalla con el menú principal del proyecto.



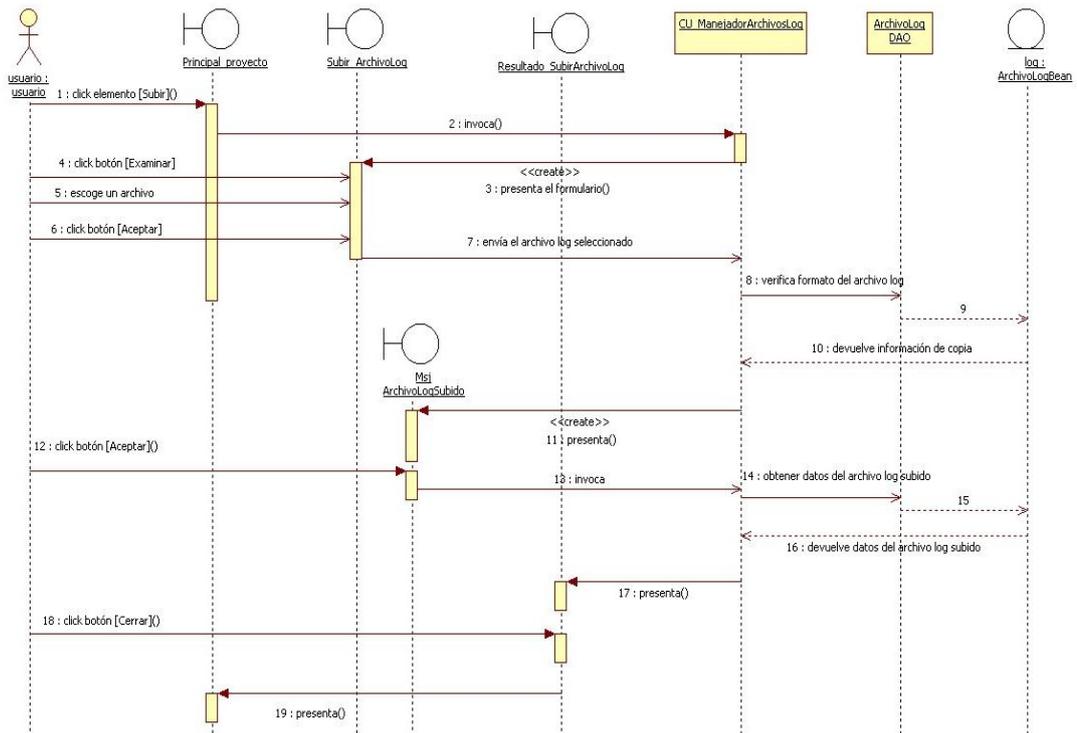


Figura 60: Diagrama de secuencia Caso de Uso: Subir un archivo log

6.3.2.12. Caso de Uso: Cargar y Pre-procesar un archivo log

Pantallas

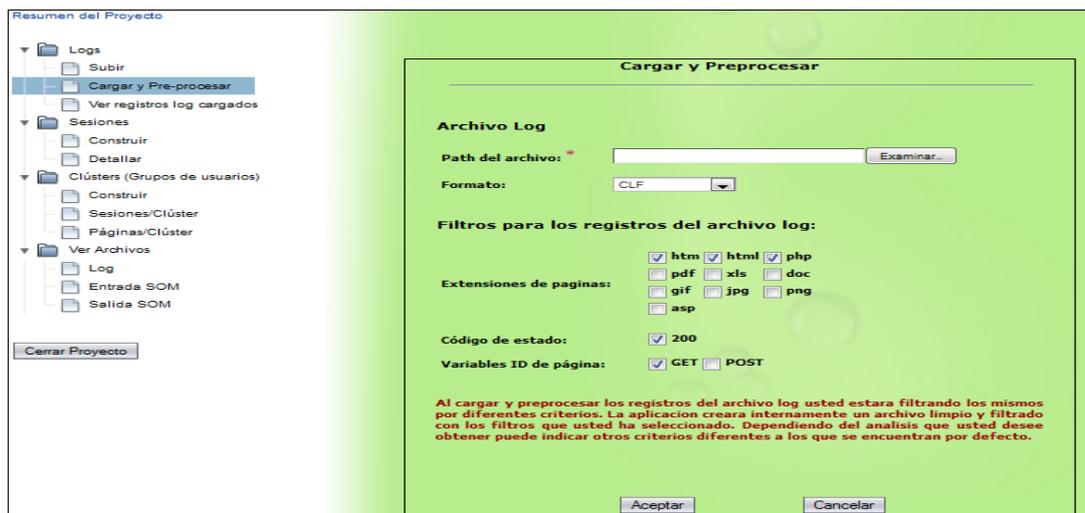


Figura 61: Página “Cargar y Pre-procesar”



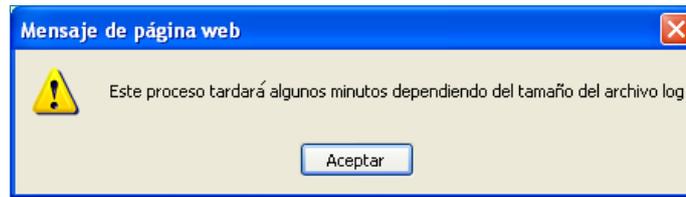


Figura 62: Mensaje al Cargar y Pre-procesar un archivo log



Figura 63: Página "Carga de Registros del Archivo Log"

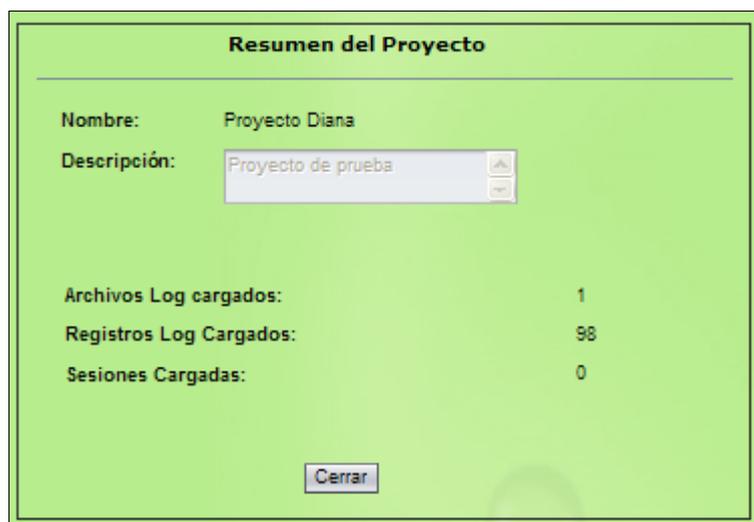


Figura 64: Página "Resumen del Proyecto después de cargar y pre-procesar un archivo log"



NOMBRE DEL CASO DE USO: CARGAR Y PRE-PROCESAR UN ARCHIVO LOG		CÓDIGO: C.U.012
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
<ul style="list-style-type: none"> - RF 009 - RF 013 - RF 014 - RF 015 - RF 016 - RF 017 	<ul style="list-style-type: none"> - Usuario Administrador - Usuario Común 	<ul style="list-style-type: none"> - Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Analizar toda la información contenida en un archivo log.	El usuario registrado y que ha iniciado sesión en el sistema selecciona uno de los archivos log subidos a su directorio para cargar los datos de sus registros y realizar el pre-procesamiento es decir remover del archivo log datos innecesarios quedando listo para construir las sesiones de usuario del mismo.	
PRE-CONDICIONES	POST-CONDICIONES	
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe estar visualizando la página principal del proyecto. 	Se han cargado y pre-procesado los registros de un archivo log.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<p>1. El usuario hace clic en el elemento “Cargar y Pre-procesar” del menú “Logs”.</p> <p>3. El usuario hace clic en el botón “Examinar” correspondiente al campo “Archivo log” para buscar un archivo log subido a su directorio.</p> <p>5. El usuario selecciona uno de los archivos log que subió.</p> <p>6. El usuario indica el “Formato” del archivo log (CLF o CLF Extendido).</p> <p>7. El usuario indica que “Extensiones” de páginas serán utilizadas (php, html, htm, doc, xls, pdf, png, jpg, gif, asp).</p> <p>8. El usuario indica el “Código de Estado” (200).</p> <p>9. El usuario indica las “Variables ID de Página” (GET y POST).</p> <p>10. El usuario hace clic en el botón “Aceptar”.</p>	<p>2. El sistema muestra el formulario “Cargar y Pre-procesar” para que el usuario ingrese los datos requeridos.</p> <p>4. El sistema muestra todos los archivos log subidos al proyecto del usuario.</p> <p>11. El sistema verifica que todos los campos se hayan completado correctamente.</p> <p>12. El sistema muestra el mensaje “Este proceso tardará algunos minutos dependiendo del tamaño del archivo log”</p>



<p>13. El usuario hace clic en el botón "Aceptar".</p> <p>17. El usuario hace clic en el botón "Cerrar".</p> <p>El usuario hace clic en el botón "Cerrar".</p>	<p>14. El sistema realiza la carga y pre-procesamiento del archivo log, seleccionando los registros que cumplen con las restricciones indicadas por el usuario.</p> <p>15. El sistema crea en el directorio del proyecto del usuario el archivo log limpio y el archivo log filtrado.</p> <p>16. Al finalizar la tarea el sistema muestra en pantalla la página "Carga de Registros del Archivo Log: NN" indicando la cantidad de registros y páginas guardadas, el path del archivo log filtrado, extensiones, código de estado e id's de página seleccionadas.</p> <p>18. El sistema muestra la página "Resumen del Proyecto" indicando la cantidad de archivos log, registros y sesiones de usuario cargadas en el proyecto.</p> <p>20. El sistema muestra la pantalla con el menú principal del proyecto.</p>
FLUJO ALTERNO	
<p>A. CAMPOS OBLIGATORIOS INCOMPLETOS</p> <p>A12. El Caso de uso continúa en el paso 3 del curso normal.</p> <p>B. EL USUARIO NO DESEA CARGAR Y PRE-PROCESAR UN ARCHIVO LOG</p> <p>B10. El usuario hace clic en el botón "Cancelar"</p> <p>B11. El Caso de uso termina.</p>	



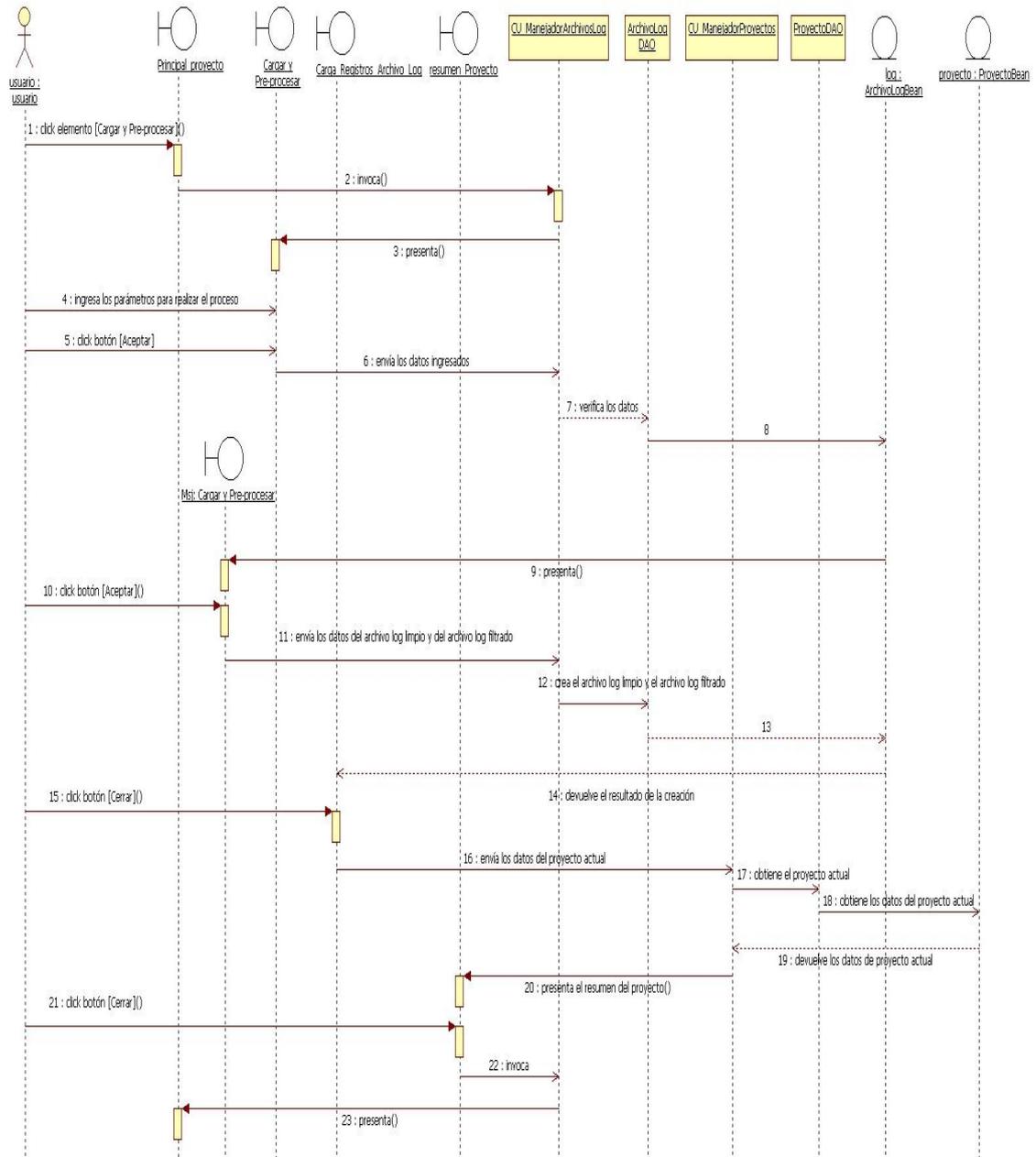


Figura 65: Diagrama de secuencia Caso de Uso: Cargar y pre-procesar un archivo log (Flujo Normal)



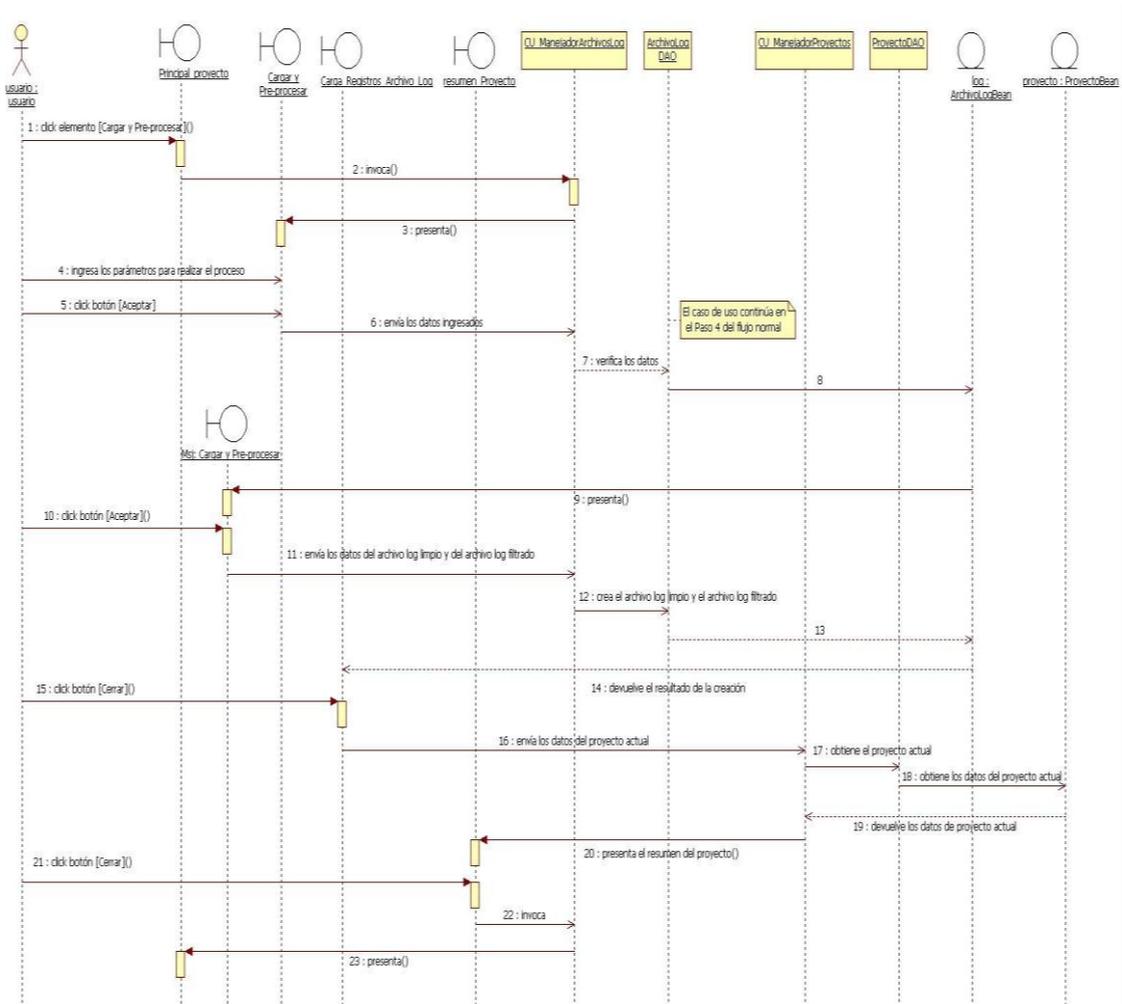


Figura 66: Diagrama de secuencia Caso de Uso: Cargar y pre-procesar un archivo log (Flujo alternativo A)

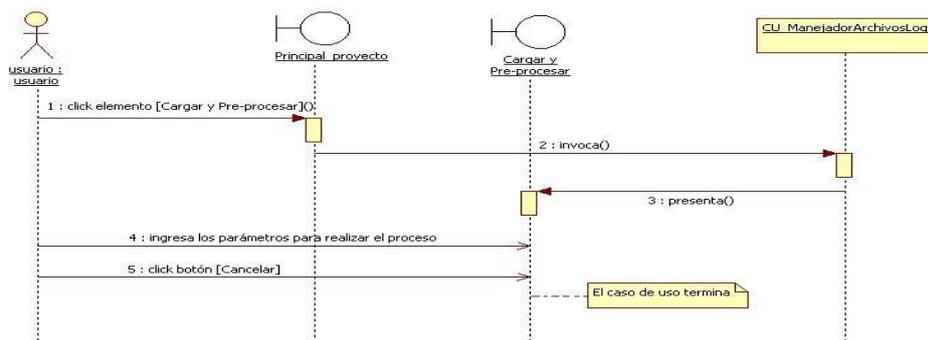


Figura 67: Diagrama de secuencia Caso de Uso: Cargar y pre-procesar un archivo log (Flujo alternativo B)



6.3.2.13. Caso de Uso: Ver los registros cargados de un archivo log

Pantalla

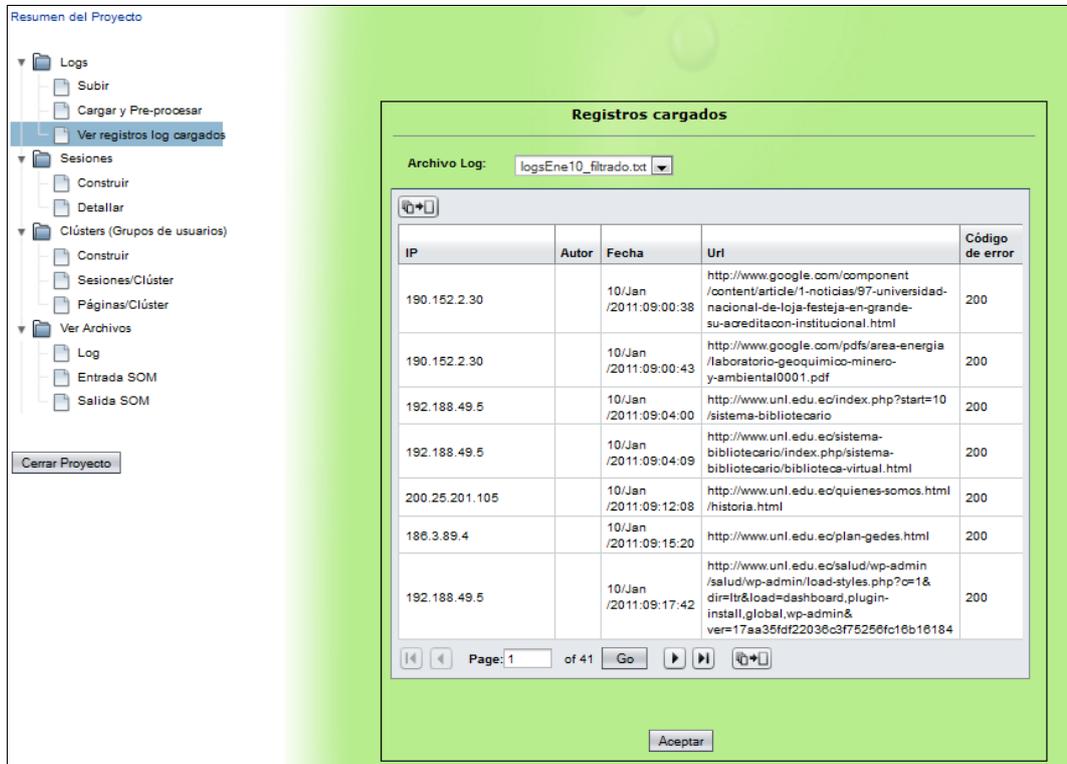


Figura 68: Página “Registros cargados”

NOMBRE DEL CASO DE USO: VER LOS REGISTROS CARGADOS DE UN ARCHIVO LOG		CÓDIGO: C.U.013
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 018	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Ver los registros de un archivo log cargado.	El usuario registrado y que ha iniciado sesión en el sistema selecciona un archivo log filtrado en su proyecto actual para visualizar la información de todos sus registros.	



PRE-CONDICIONES	POST-CONDICIONES
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe cargar y pre-procesar uno o más archivos log. - El usuario debe estar visualizando la página principal del proyecto. 	<p>Se ha recopilado información sobre los registros de un archivo log cargado y pre-procesado.</p>
FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<ol style="list-style-type: none"> 1. El usuario hace clic en el elemento “Ver registros log cargados” del menú “Logs”. 2. El usuario selecciona de la lista desplegable “Archivo Log” un archivo log filtrado del cual desea obtener información sobre sus registros. 4. El usuario hace clic en el botón “Aceptar”. 	<ol style="list-style-type: none"> 3. El sistema muestra la lista “Registros cargados” con todos los registros del archivo log filtrado indicando: ip, autor, fecha, url y código de error. 5. El sistema muestra la pantalla con el menú principal del proyecto.
FLUJO ALTERNO	
<p>A. NO SE HA CARGADO Y PRE-PROCESADO UN ARCHIVO LOG</p> <p>A3. El sistema muestra el mensaje de información "No existen registros en el archivo log".</p> <p>A4. El Caso de uso termina.</p>	



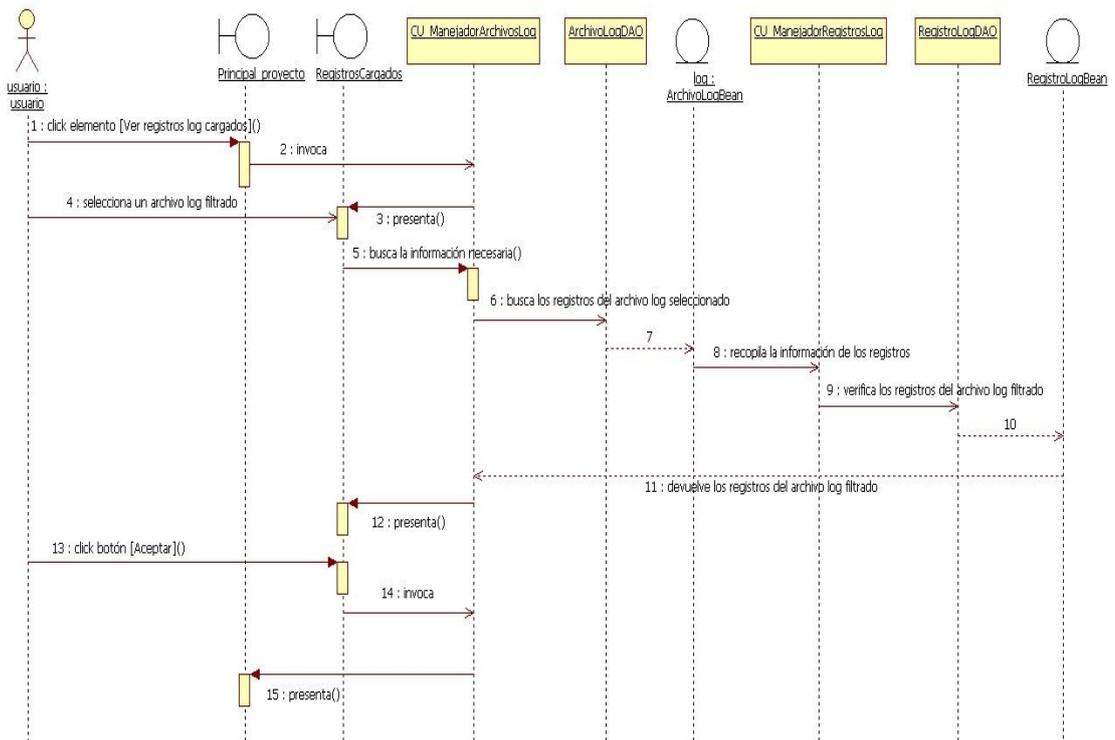


Figura 69: Diagrama de secuencia Caso de Uso: Ver los registros cargados de un archivo log (Flujo normal)

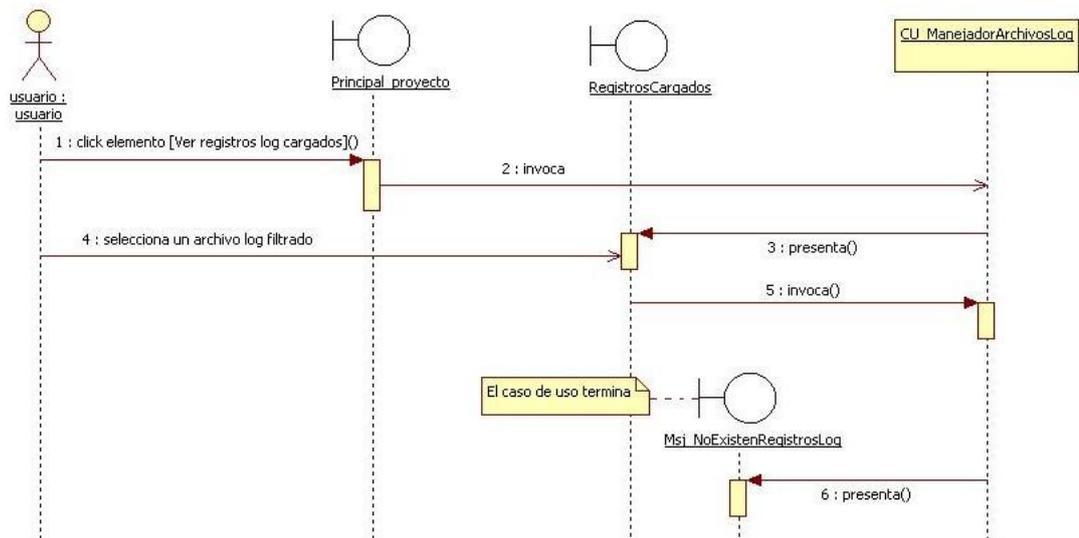


Figura 70: Diagrama de secuencia Caso de Uso: Ver los registros cargados de un archivo log (Flujo alternativo A)



6.3.2.14. Caso de Uso: Construir las sesiones de usuario

Pantallas



Figura 71: Página “Construcción de Sesiones”

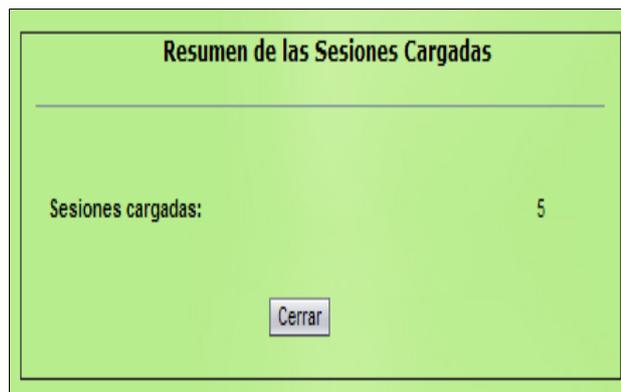


Figura 72: Página “Resumen de las Sesiones Cargadas”



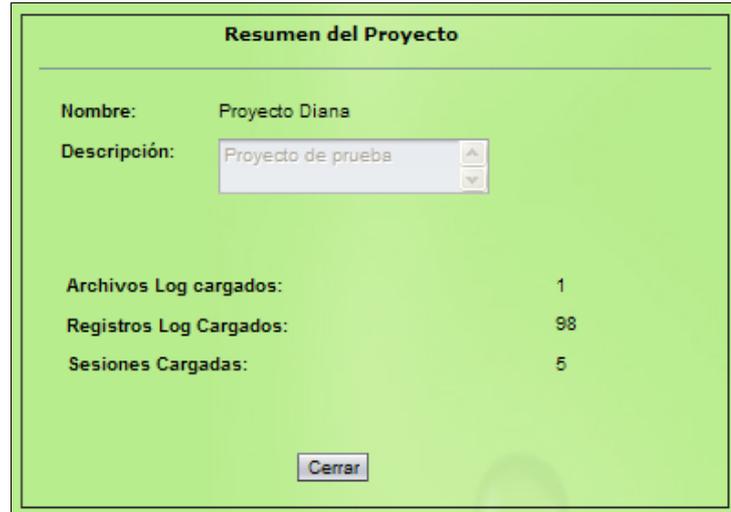


Figura 73: Página “Resumen del Proyecto después de construir las sesiones de usuario”

NOMBRE DEL CASO DE USO: CONSTRUIR LAS SESIONES DE USUARIO		CÓDIGO: C.U.014
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 009 - RF 019 - RF 020	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Obtener las sesiones de usuario de un archivo log pre-procesado o filtrado.	El usuario registrado y que ha iniciado sesión en el sistema selecciona un archivo log filtrado en su proyecto actual para construir las sesiones de usuario.	



PRE-CONDICIONES	POST-CONDICIONES
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe cargar y pre-procesar uno o más archivos log. - El usuario debe estar visualizando la página principal del proyecto. 	<p>Se han creado las sesiones de usuario del archivo log.</p>
FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<ol style="list-style-type: none"> 1. El usuario hace clic en el elemento “Construir” del menú “Sesiones”. 3. El usuario selecciona de la lista desplegable “Archivo Log” un archivo log filtrado para construir las sesiones de usuario. 4. El usuario indica la cantidad mínima de páginas en una sesión, el tiempo máximo de sesión y la frecuencia mínima de páginas; los valores establecidos por defecto pueden ser modificados. 5. El usuario hace clic en el botón “Aceptar”. 	<ol style="list-style-type: none"> 2. El sistema despliega el formulario “Construcción de Sesiones” que permite configurar las opciones para la construcción de las sesiones de usuarios. 6. El sistema muestra la página “Resumen de las Sesiones Cargadas” que indica el número de sesiones cargadas.



<p>7. El usuario hace clic en el botón “Cerrar”.</p> <p>9. El usuario hace clic en el botón “Cerrar”.</p>	<p>8. El sistema muestra la página “Resumen del Proyecto” indicando la cantidad de archivos log, registros y sesiones de usuario cargadas en el proyecto.</p> <p>10. El sistema muestra la pantalla con el menú principal del proyecto.</p>
---	---

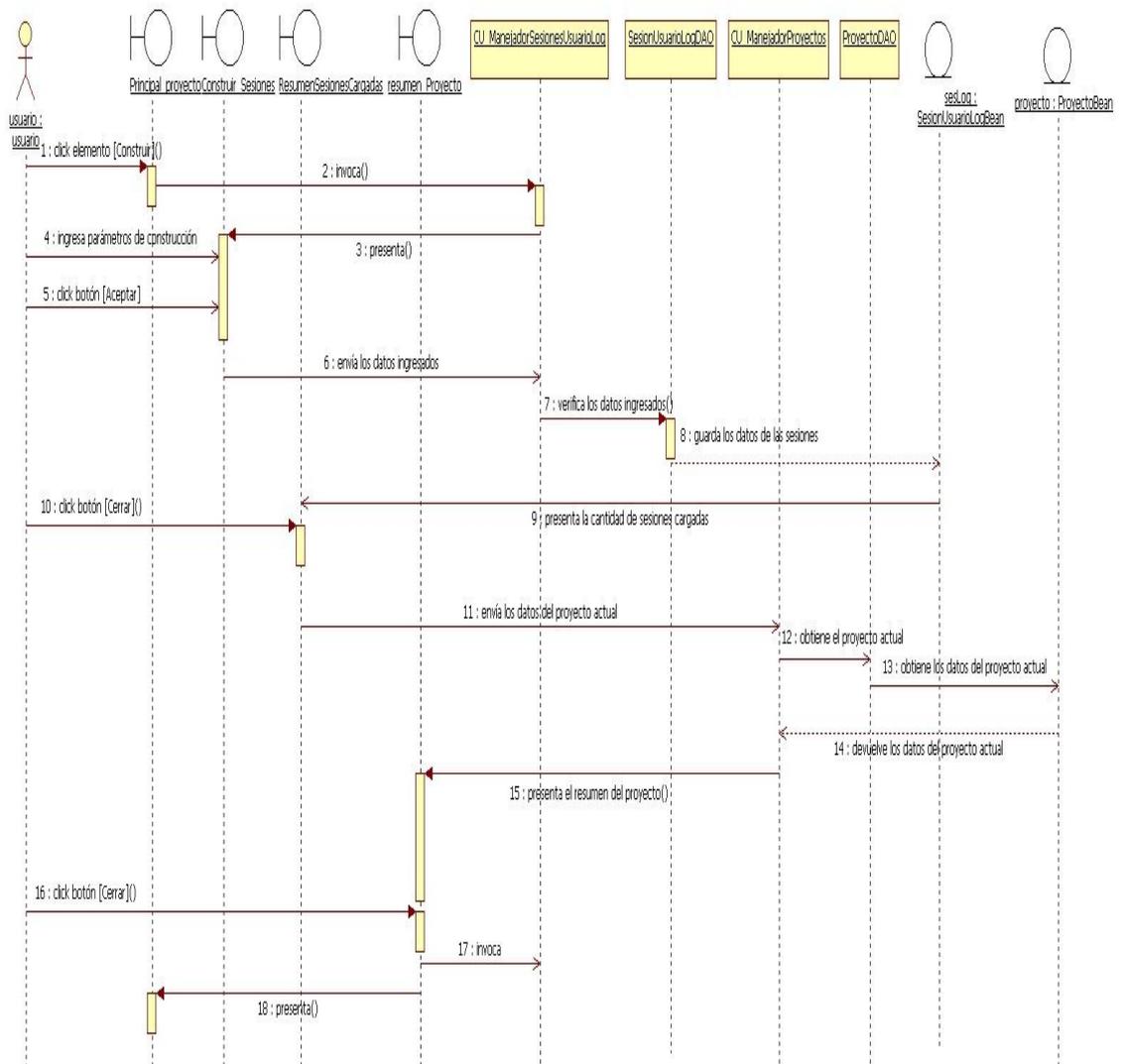


Figura 74: Diagrama de secuencia Caso de Uso: Construir las sesiones de usuario



6.3.2.15. Caso de Uso: Ver el detalle de las sesiones de usuario

Pantalla

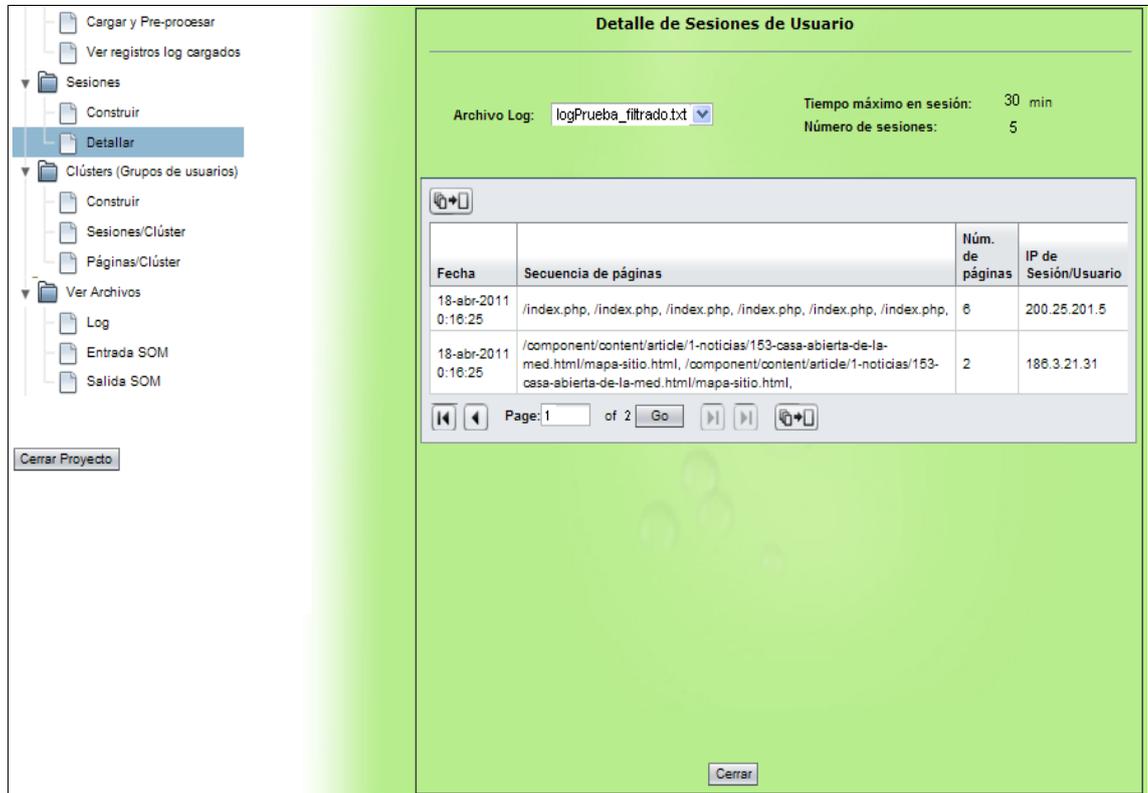


Figura 75: Página “Detalle de Sesiones de Usuario”

NOMBRE DEL CASO DE USO: VER EL DETALLE DE LAS SESIONES DE USUARIO		CÓDIGO: C.U.015
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 021	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Ver la información de todas las sesiones del usuario.	El usuario registrado y que ha iniciado sesión en el sistema selecciona un archivo log filtrado en su proyecto actual para conocer el detalle de las sesiones de usuario.	



PRE-CONDICIONES	POST-CONDICIONES
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe cargar y pre-procesar uno o más archivos log. - El usuario debe realizar la construcción de sesiones de uno o más archivos log pre-procesados o filtrados. - El usuario debe estar visualizando la página principal del proyecto. 	<p>Se ha recopilado información de las sesiones de usuario de un archivo log.</p>
FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<ol style="list-style-type: none"> 1. El usuario hace clic en el elemento “Detallar” del menú “Sesiones”. 3. El usuario selecciona de la lista desplegable “Archivo Log” un archivo log filtrado del cual desea obtener el detalle de sesiones de usuario. 5. El usuario hace clic en el botón “Cerrar”. 	<ol style="list-style-type: none"> 2. El sistema muestra la página “Detalle de Sesiones de Usuario”. 4. El sistema muestra una tabla con todas las sesiones de usuario del archivo log seleccionado indicando: fecha, secuencia de páginas, número de páginas e IP de sesión de usuario. 6. El sistema muestra la pantalla con el menú principal del proyecto.



FLUJO ALTERNO

A. NO SE HAN CONSTRUIDO LAS SESIONES DE USUARIO DE UN ARCHIVO LOG

A4. El sistema muestra el mensaje de información "No existen sesiones para este archivo log".

A5. El Caso de uso termina.

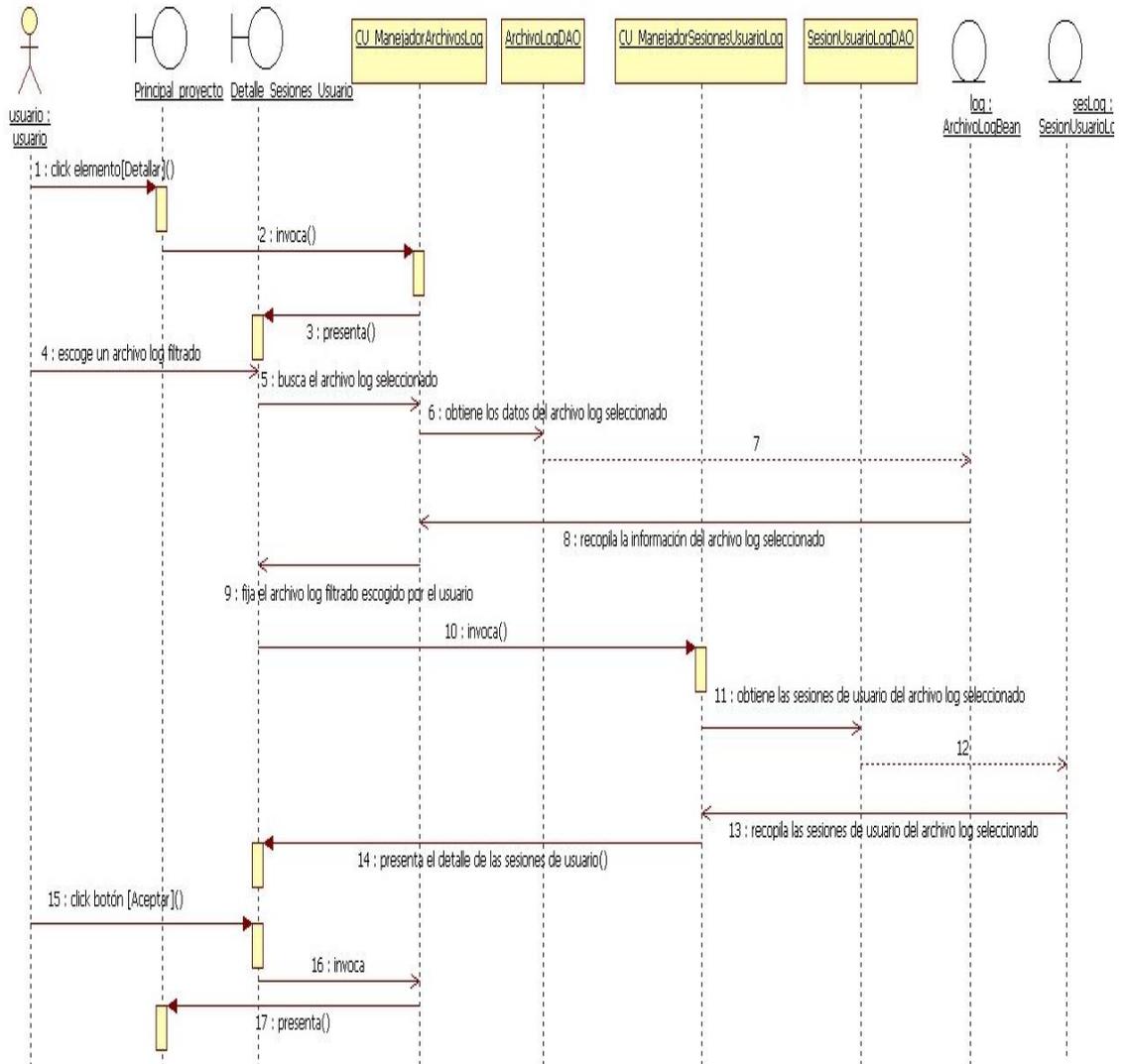


Figura 76: Diagrama de secuencia Caso de Uso: Ver el detalle de las sesiones de usuario (Flujo normal)



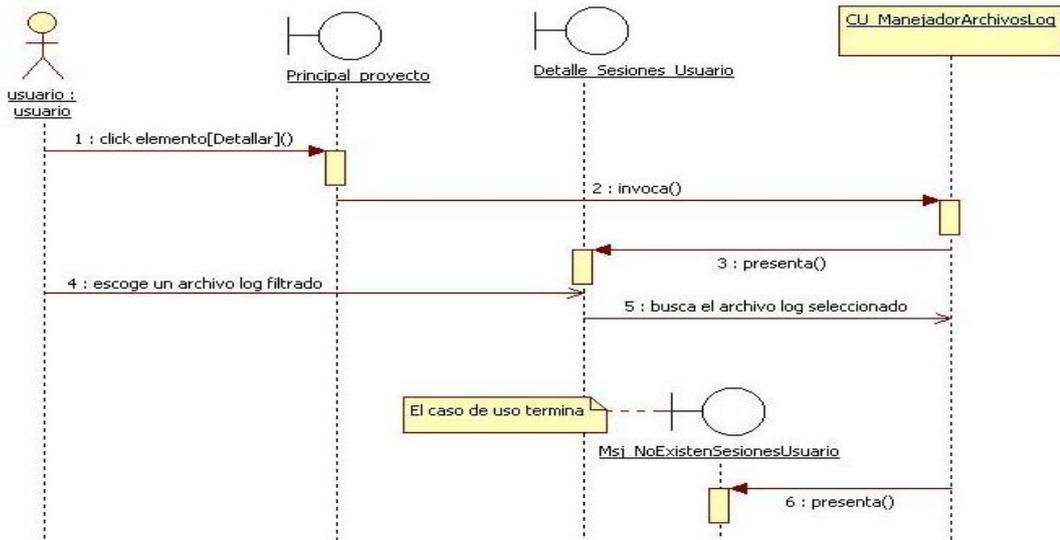


Figura 77: Diagrama de secuencia Caso de Uso: Ver el detalle de las sesiones de usuario (Flujo alternativo A)

6.3.2.16. Caso de Uso: Agrupar usuarios utilizando la Red Neuronal Artificial SOM

Pantallas

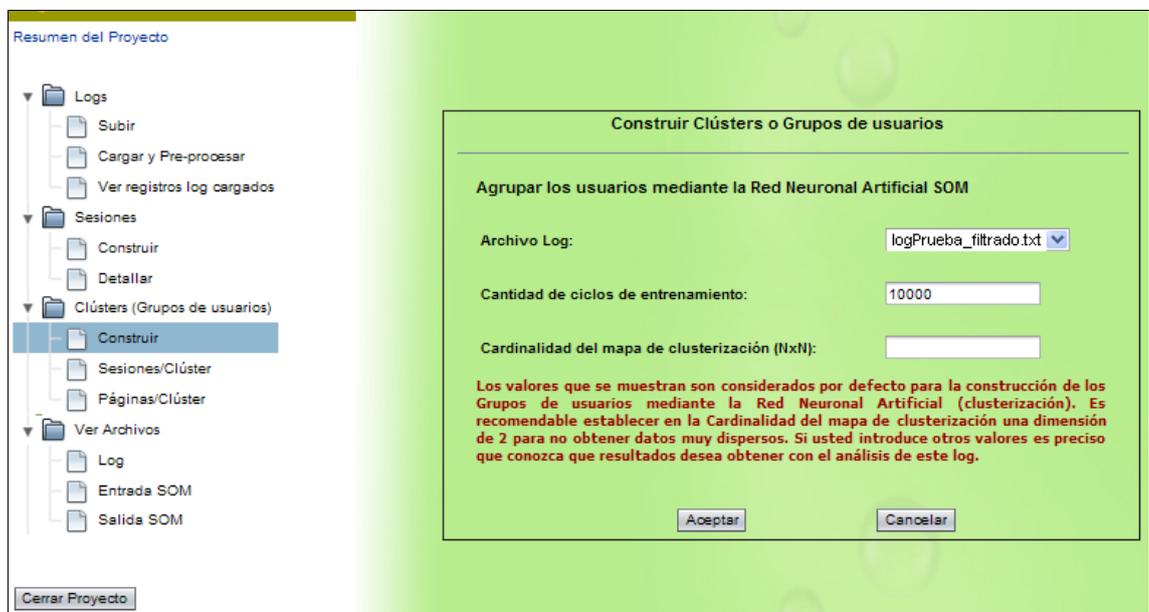


Figura 78: Página “Construir Clústers o Grupos de usuarios”





Figura 79: Página “Resumen de la Construcción de Clústers”

NOMBRE DEL CASO DE USO: AGRUPAR USUARIOS UTILIZANDO LA RED NEURONAL ARTIFICIAL SOM		CÓDIGO: C.U.016
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 022 - RF 023	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Obtener los grupos de usuarios.	El usuario registrado y que ha iniciado sesión en el sistema selecciona un archivo log filtrado en su proyecto actual para generar a través de la Red Neuronal Artificial SOM los distintos grupos de usuarios.	



PRE-CONDICIONES	POST-CONDICIONES
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe realizar la construcción de sesiones de uno o más archivos log pre-procesados o filtrados. - El usuario debe estar visualizando la página principal del proyecto. 	<p>Se han agrupado los usuarios de las diferentes sesiones.</p>
FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<ol style="list-style-type: none"> 1. El usuario hace clic en el elemento "Construir" del menú "Clústers (Grupos de usuarios)". 3. El usuario selecciona de la lista desplegable "Archivo Log" un archivo log filtrado para realizar el agrupamiento de usuarios. 4. El usuario indica la cantidad de ciclos de entrenamiento a realizar con los patrones de entrada a la Red Neuronal Artificial SOM. 5. El usuario indica la cardinalidad del mapa de clusterización (NxN), indicando el valor N (número entero) que resultará en un mapa de NxN. 	<ol style="list-style-type: none"> 2. El sistema muestra el formulario "Construir Clústers o Grupos de usuarios".





<p>6. El usuario hace clic en el botón “Aceptar”.</p>	<p>7. El sistema verifica que todos los Autoras: Irlanda Loján & Andrea Castillo campos se hayan completado correctamente.</p> <p>8. El sistema crea el archivo con los</p>
--	--

FLUJO ALTERNO

A. CAMPOS OBLIGATORIOS INCOMPLETOS

A8. El Caso de uso continúa en el Paso 3 del curso normal.

B. EL USUARIO NO DESEA CONSTRUIR CLÚSTERS

B6. El usuario hace clic en el botón “Cancelar”

B7. El Caso de uso termina.

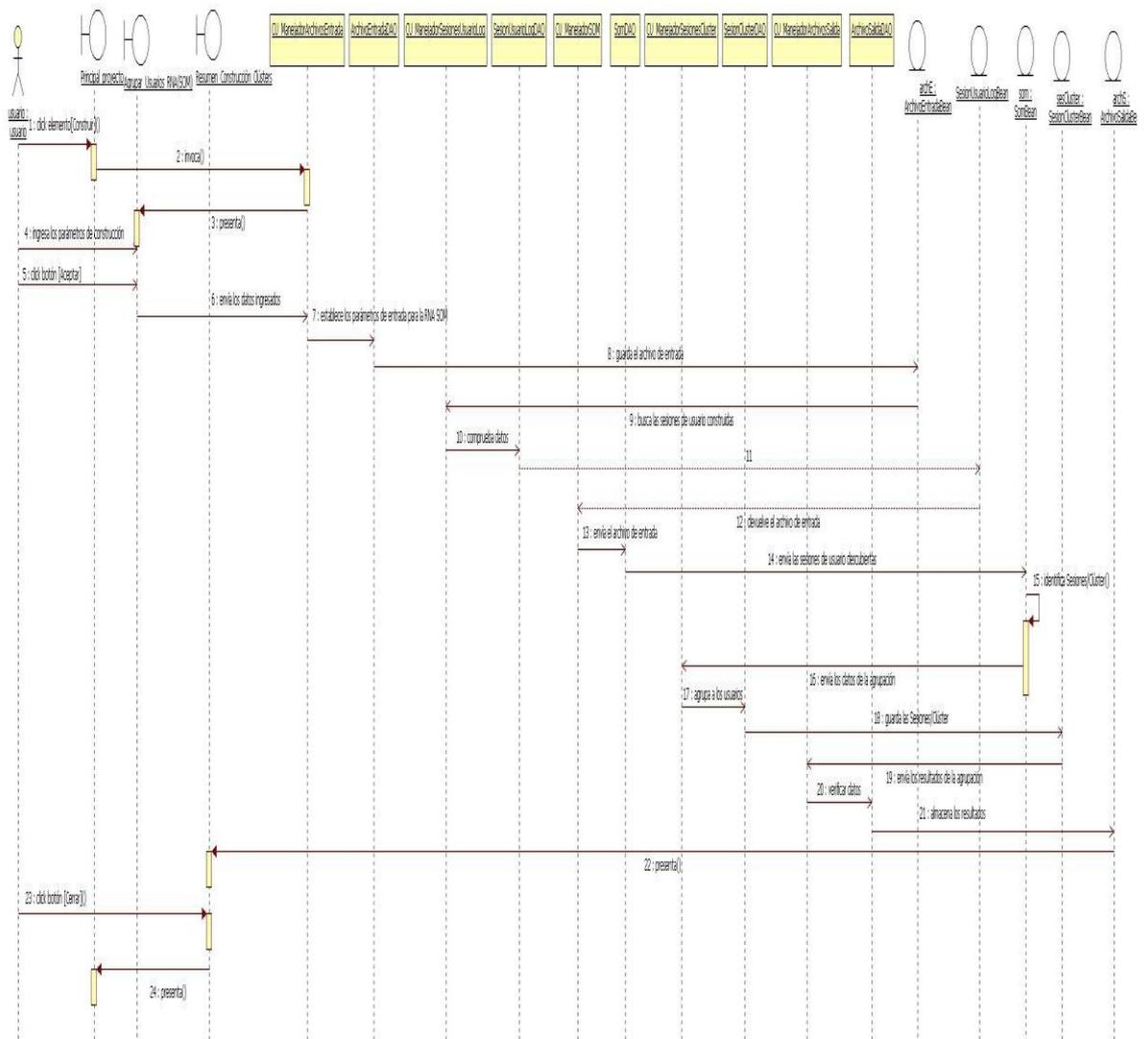


Figura 80: Diagrama de secuencia Caso de Uso: Agrupar usuarios utilizando la Red Neuronal Artificial SOM (Flujo normal)



6.3.2.17. Caso de Uso: Ver la cantidad de sesiones de usuario en cada clúster

Pantallas

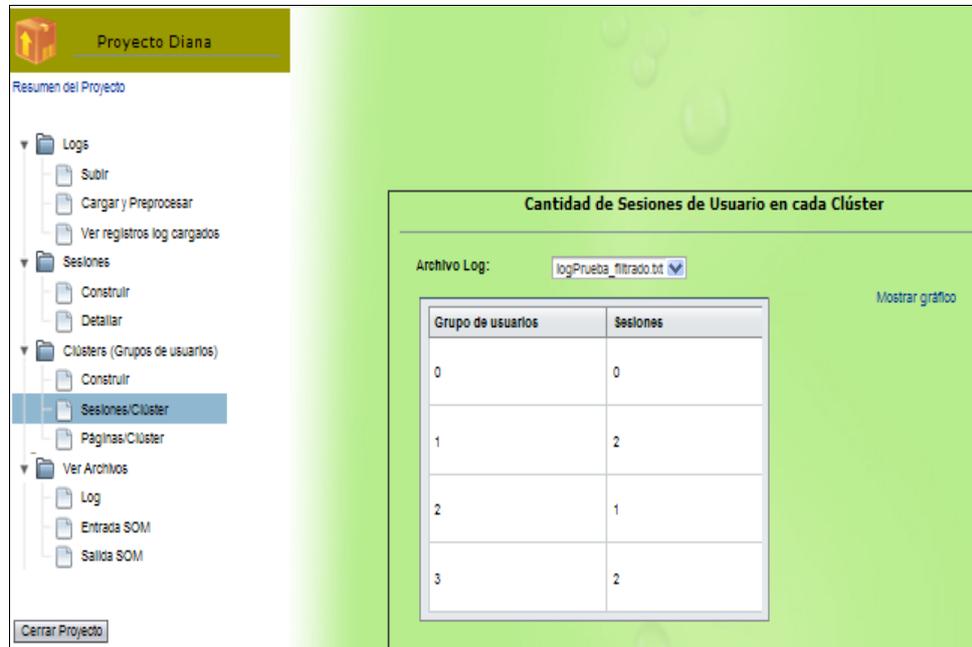


Figura 83: Página “Cantidad de Sesiones de Usuario en cada Clúster”



Figura 84: Gráfico demostrativo de sesiones en clúster.



NOMBRE DEL CASO DE USO: VER LA CANTIDAD DE SESIONES DE USUARIO EN CADA CLÚSTER		CÓDIGO: C.U.017
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
<ul style="list-style-type: none"> - RF 024 - RF 025 	<ul style="list-style-type: none"> - Usuario Administrador - Usuario Común 	<ul style="list-style-type: none"> - Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Obtener la cantidad de sesiones de usuario de cada grupo.	El usuario registrado y que ha iniciado sesión en el sistema selecciona un archivo log filtrado en su proyecto actual para ver la cantidad de sesiones de usuario en cada grupo descubierto.	
PRE-CONDICIONES	POST-CONDICIONES	
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe realizar el agrupamiento de usuarios utilizando la Red Neuronal Artificial SOM. - El usuario debe estar visualizando la página principal del proyecto. 	Se ha calculado y graficado el porcentaje de sesiones por clúster, esto es posible mediante la utilización de la librería ChartDirector.jar.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<p>1. El usuario hace clic en el elemento "Sesiones/Clúster" del menú "Clústers (Grupos de Usuarios)".</p> <p>3. El usuario selecciona de la lista desplegable "Archivo Log" un archivo log filtrado para ver la cantidad de sesiones de usuario en cada clúster.</p> <p>7. El usuario hace clic en el link "Mostrar gráfico"</p> <p>10. El usuario cierra la ventana.</p> <p>11. El usuario hace clic en el botón "Cerrar".</p>	<p>2. El sistema muestra la página "Cantidad de Sesiones de Usuario en cada Clúster".</p> <p>4. El sistema analizará el archivo de salida correspondiente al archivo log seleccionado por el usuario.</p> <p>5. El sistema calcula la cantidad de sesiones en cada clúster descubierto.</p> <p>6. El sistema muestra la cantidad de sesiones de usuario en cada clúster descubierto mediante la clusterización SOM.</p> <p>8. El sistema calcula el porcentaje de sesiones de usuario en cada clúster descubierto.</p> <p>9. El sistema genera un gráfico circular con el nombre del archivo log analizado y el porcentaje de las sesiones de usuario descubierto en cada clúster.</p> <p>12. El sistema muestra la pantalla con el menú principal del proyecto.</p>



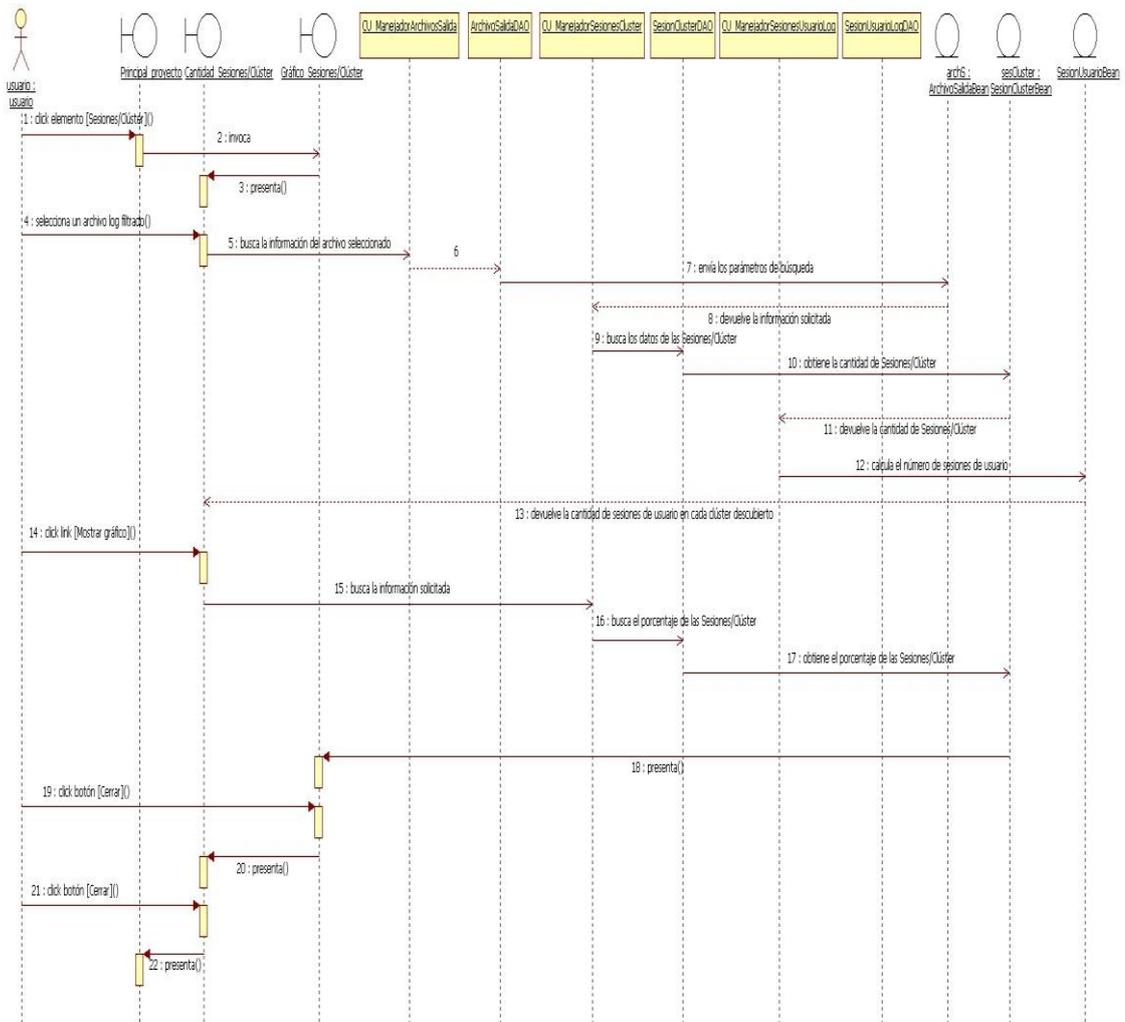


Figura 85: Diagrama de secuencia Caso de Uso: Ver la cantidad de sesiones de usuario en cada clúster



6.3.2.18. Caso de Uso: Ver el detalle de las páginas más visitadas por los usuarios de cada clúster

Pantallas

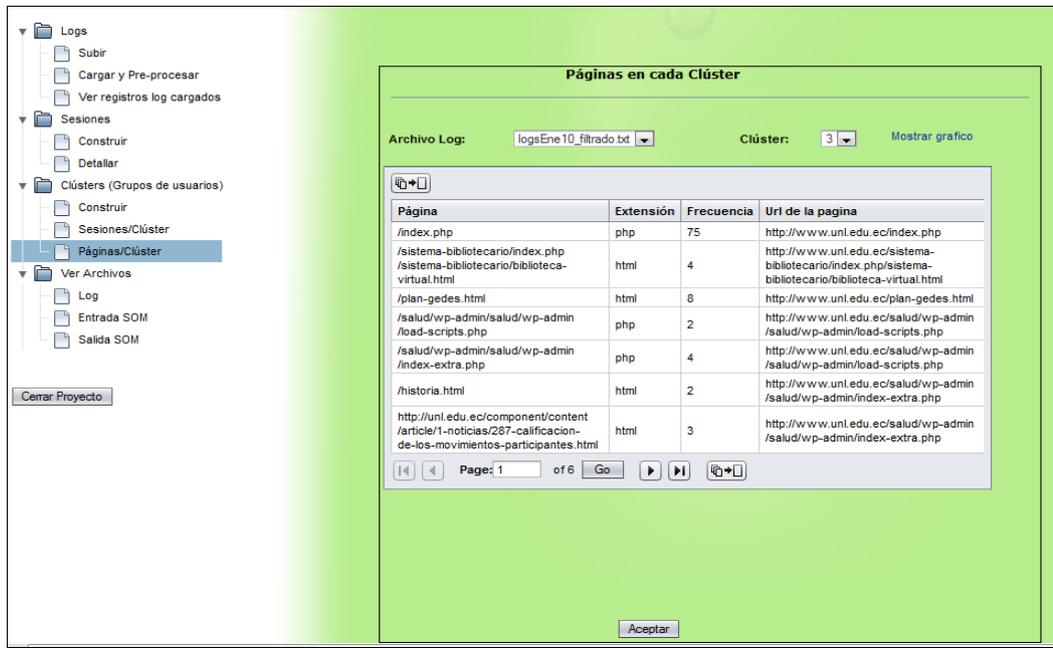


Figura 86: Página “Páginas en cada Clúster”

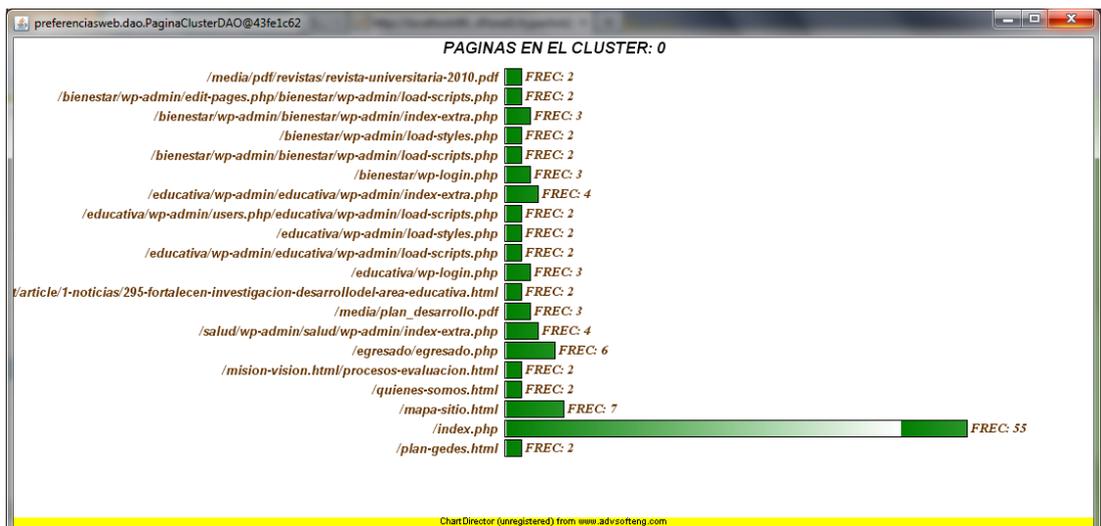


Figura 87: Gráfico demostrativo de páginas en un clúster.



NOMBRE DEL CASO DE USO: VER EL DETALLE DE LAS PÁGINAS MÁS VISITADAS POR LOS USUARIOS DE CADA CLÚSTER		CÓDIGO: C.U.018
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
<ul style="list-style-type: none"> - RF 026 - RF 027 	<ul style="list-style-type: none"> - Usuario Administrador - Usuario Común 	<ul style="list-style-type: none"> - Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Ver el detalle de las páginas más visitadas por los usuarios de cada clúster.	El usuario registrado y que ha iniciado sesión en el sistema selecciona un archivo log filtrado en su proyecto actual para ver el detalle de las páginas más visitadas por los usuarios de cada clúster.	
PRE-CONDICIONES	POST-CONDICIONES	
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe realizar el agrupamiento de usuarios utilizando la Red Neuronal Artificial SOM. - El usuario debe estar visualizando la página principal del proyecto. 	Se han obtenido las páginas más visitadas por los usuarios de cada clúster.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<ol style="list-style-type: none"> 1. El usuario hace clic en el elemento "Páginas/Clúster" del menú "Clústers (Grupos de usuarios)". 3. El usuario selecciona de la lista desplegable "Archivo Log" un archivo log filtrado. 4. El usuario selecciona de la lista desplegable "Clúster" uno de ellos para descubrir las páginas más visitadas por los usuarios en el clúster indicado. 6. El usuario hace clic en el link "Mostrar gráfico". 8. El usuario cierra la ventana. 9. El usuario hace click en el botón "Aceptar" 	<ol style="list-style-type: none"> 2. El sistema muestra la página "Páginas en cada Clúster". 5. El sistema muestra en una tabla: la página, su extensión, la frecuencia de acceso y url. 7. El sistema muestra una ventana con el grafico de barras que indica las páginas con la frecuencia correspondiente con la que han sido visitadas.
FLUJO ALTERNO	
<p>A. NO EXISTEN PÁGINAS EN EL CLÚSTER</p> <p>A5. El sistema muestra el mensaje de información "No existen páginas en este clúster".</p> <p>A6. El caso de uso continúa en el paso 4 del curso normal.</p>	



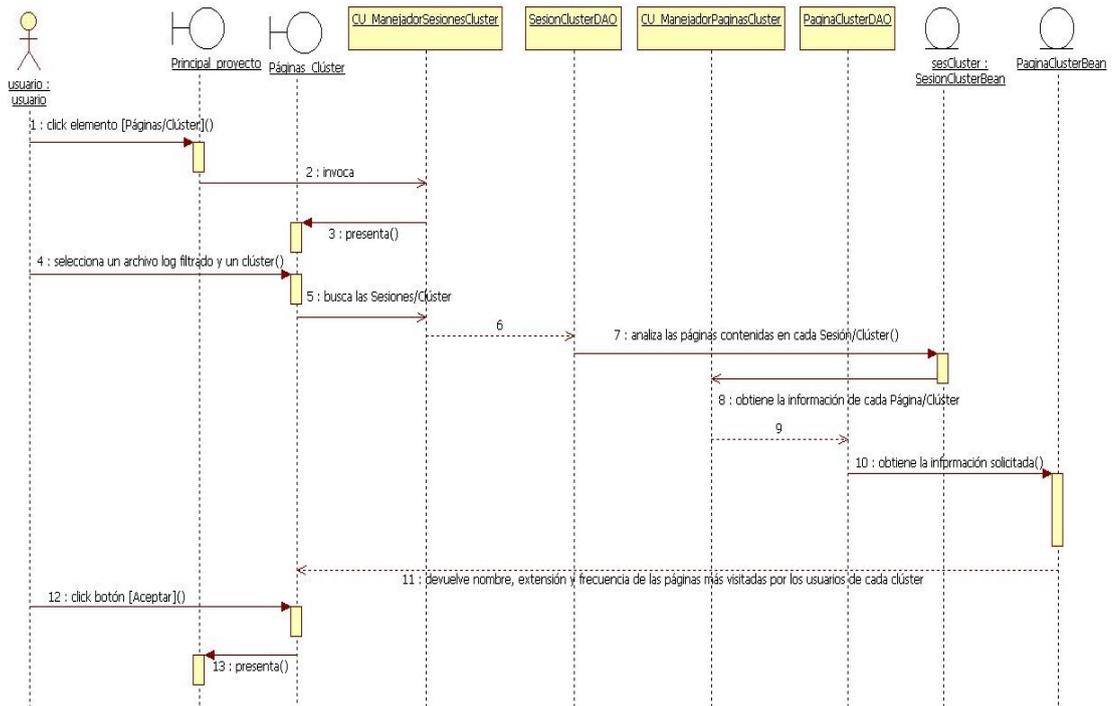


Figura 88: Diagrama de secuencia Caso de Uso: Ver el detalle de las páginas más visitadas por los usuarios de cada clúster (Flujo normal)

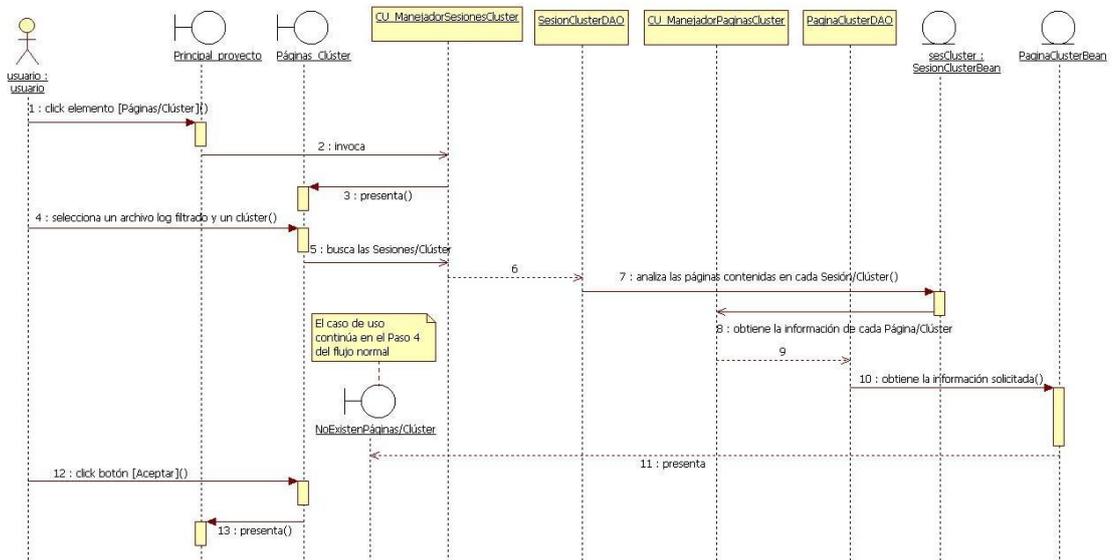


Figura 89: Diagrama de secuencia Caso de Uso: Ver el detalle de las páginas más visitadas por los usuarios de cada clúster (Flujo alterno A)



6.3.2.19. Caso de Uso: Ver el detalle de un archivo log

Pantallas

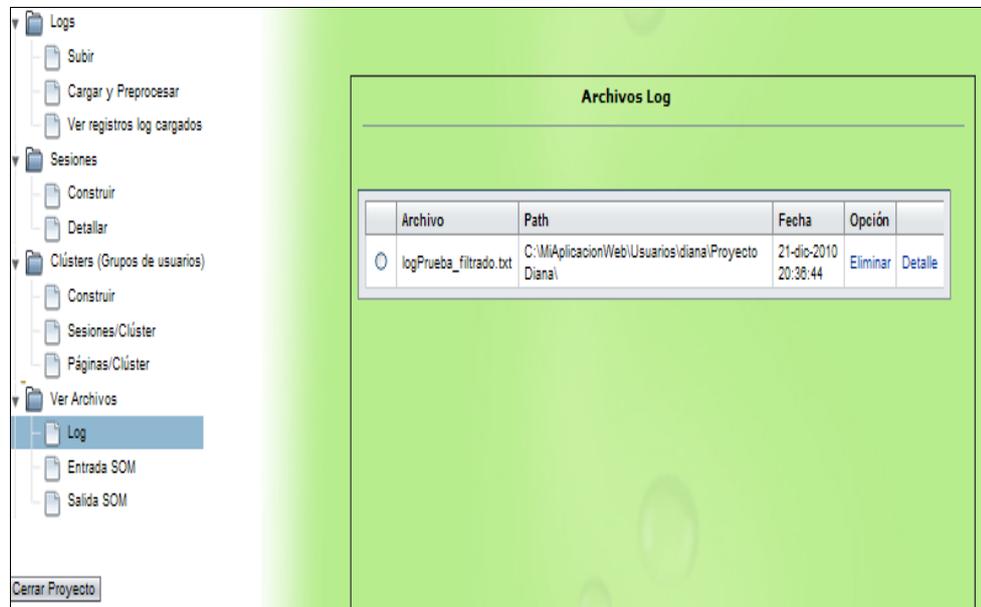


Figura 90: Página “Archivos Log”

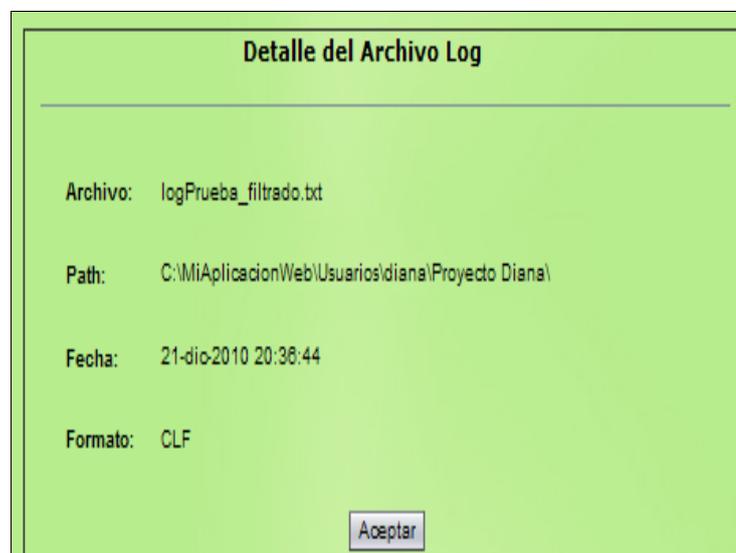


Figura 91: Página “Detalle del Archivo Log”



NOMBRE DEL CASO DE USO: VER EL DETALLE DE UN ARCHIVO LOG		CÓDIGO: C.U.019
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 028 - RF 029	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Listar y ver el detalle de los archivos log cargados y pre-procesados dentro de un proyecto.	El usuario registrado y que ha iniciado sesión en el sistema selecciona de la lista "Archivos Log" un archivo log para obtener el detalle del mismo.	
PRE-CONDICIONES	POST-CONDICIONES	
- El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe cargar y pre-procesar un archivo log. - El usuario debe estar visualizando la página principal del proyecto.	Se ha recopilado información sobre el archivo log seleccionado.	
FLUJO NORMAL		
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA	
1. El usuario hace clic en el elemento "Log" del menú "Ver Archivos". 4. El usuario selecciona el archivo log del cual desea ver el detalle haciendo clic en el link "Detalle" del listado "Archivos Log".	2. El sistema recopila todos los archivos del tipo seleccionado. 3. El sistema muestra la lista "Archivos Log" con todos los archivos log que fueron recopilados. 5. El sistema muestra la página "Detalle del Archivo Log" indicando el nombre, path, fecha y formato el archivo log.	



<p>6. El usuario hace clic en el botón “Aceptar”.</p> <p>8. El usuario hace clic en el botón “Cerrar”.</p>	<p>7. El sistema muestra la lista “Archivos Log”.</p> <p>9. El sistema muestra la pantalla con el menú principal del proyecto.</p>
FLUJO ALTERNO	
<p>A. NO EXISTEN ARCHIVOS LOG</p> <p>A3. El sistema muestra el mensaje de información “No existen archivos log”.</p> <p>A4. El usuario hace clic en el botón “Cerrar”</p> <p>A5. El caso de uso termina.</p>	

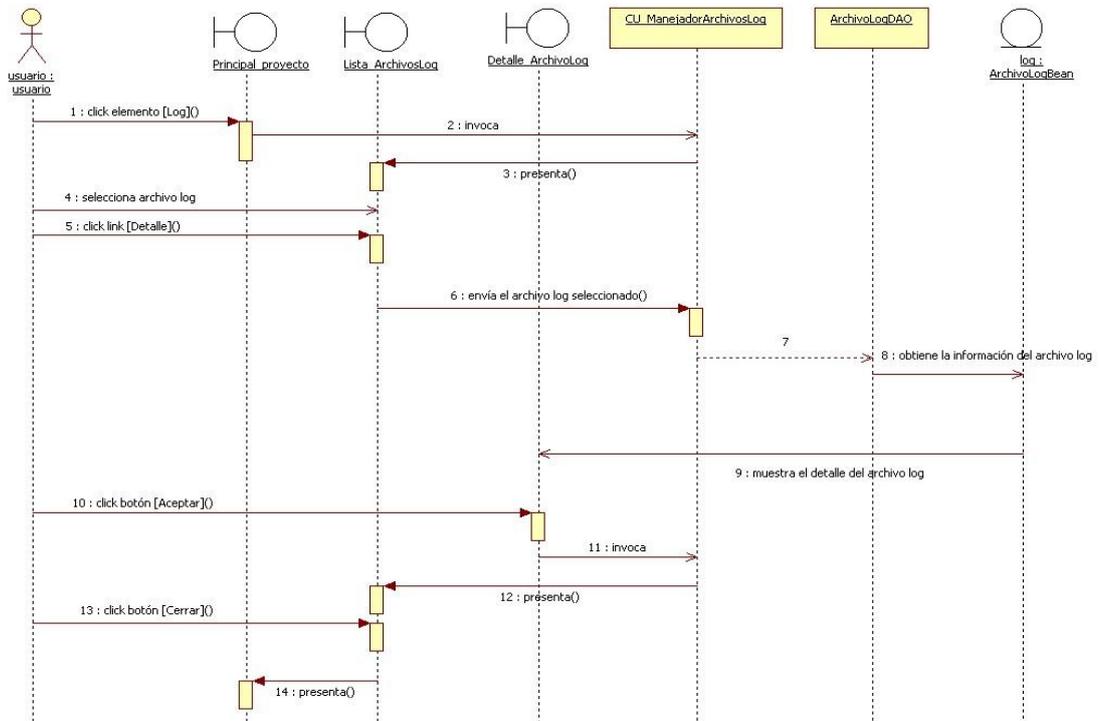


Figura 92: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo log (Flujo normal)



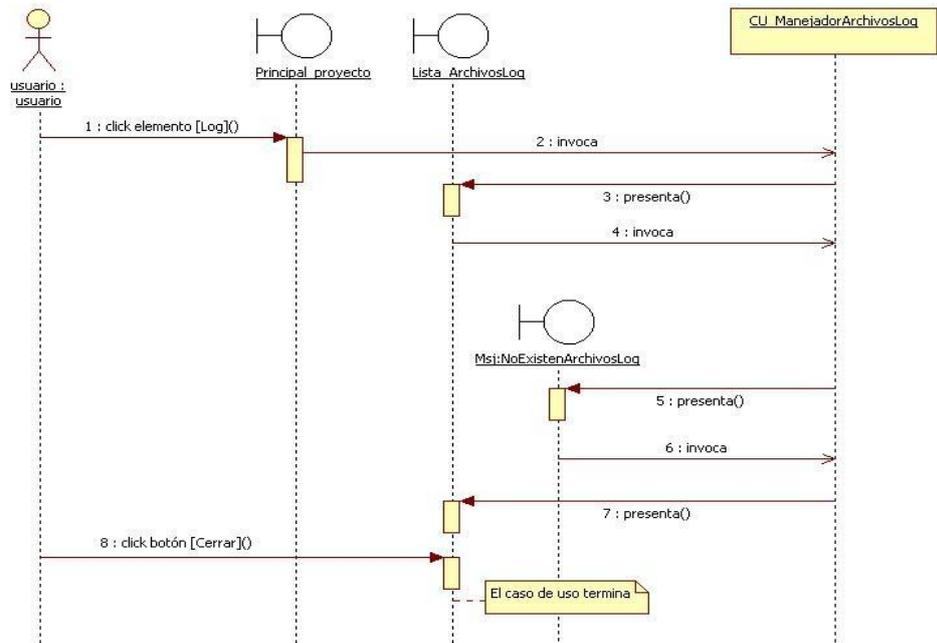


Figura 93: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo log (Flujo alternativo A)

6.3.2.20. Caso de Uso: Eliminar un archivo log

Pantalla

Archivos Log					
	Archivo	Path	Fecha	Opción	
	log02-Ago-2009_filtrado.txt	C:\MiAplicacionWeb\Usuarios\danita\Proy_Danita\	26/08/2010 03:11:43 PM	Eliminar	Detalle

Figura 94: Pantalla “Lista de Archivos Log”





Figura 95: Mensaje de eliminación de un archivo log

NOMBRE DEL CASO DE USO: ELIMINAR UN ARCHIVO LOG		CÓDIGO: C.U.020
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
<ul style="list-style-type: none"> - RF 028 - RF 030 	<ul style="list-style-type: none"> - Usuario Administrador - Usuario Común 	<ul style="list-style-type: none"> - Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Listar los archivos log cargados y pre-procesados para borrar uno o más de ellos.	El usuario registrado y que ha iniciado sesión en el sistema selecciona de la lista "Archivos Log" un archivo log para borrar toda la información relacionada con el mismo.	
PRE-CONDICIONES	POST-CONDICIONES	
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe estar visualizando la página principal del proyecto. 	Se ha eliminado un archivo log filtrado.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
1. El usuario hace clic en el elemento “Log” del menú “Ver Archivos”.	2. El sistema recopila todos los archivos del tipo seleccionado.
4. El usuario selecciona de la lista “Archivos Log” un archivo log para ser eliminado.	3. El sistema muestra la lista “Archivos Log” con todos los archivos que fueron recopilados.
5. El usuario hace clic en el link “Eliminar”.	6. El sistema muestra un mensaje “Está eliminando un archivo log, todos sus registros y páginas también serán eliminados”
7. El usuario hace clic en el botón “Aceptar”.	8. El sistema actualiza la vista de la lista de archivos log.
9. El usuario hace clic en el botón “Cerrar”.	10. El sistema muestra la pantalla con el menú principal del proyecto.

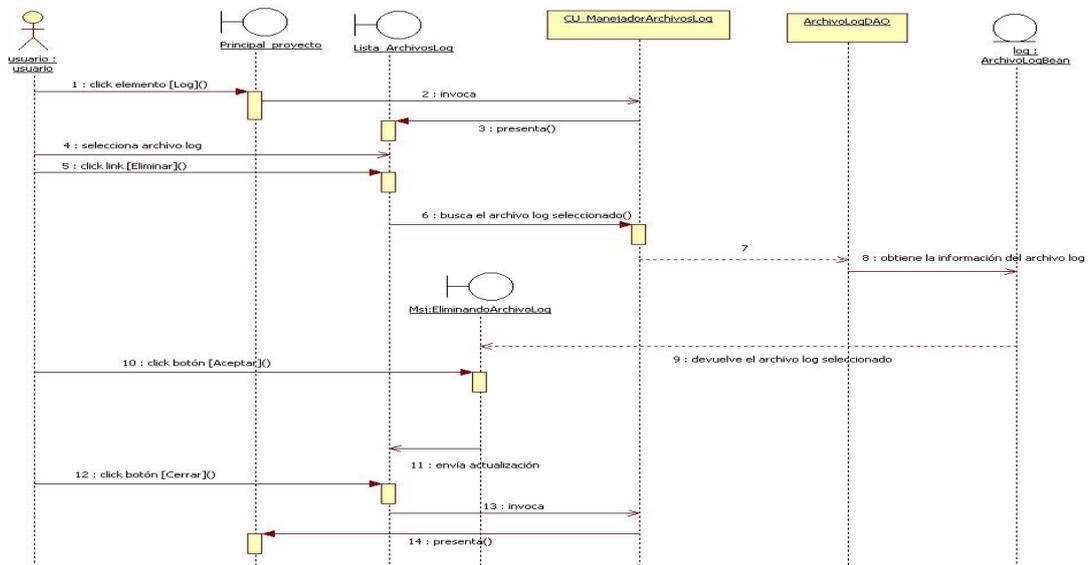


Figura 96: Diagrama de secuencia Caso de Uso: Eliminar un archivo log



6.3.2.21. Caso de Uso: Ver el detalle de un archivo de entrada

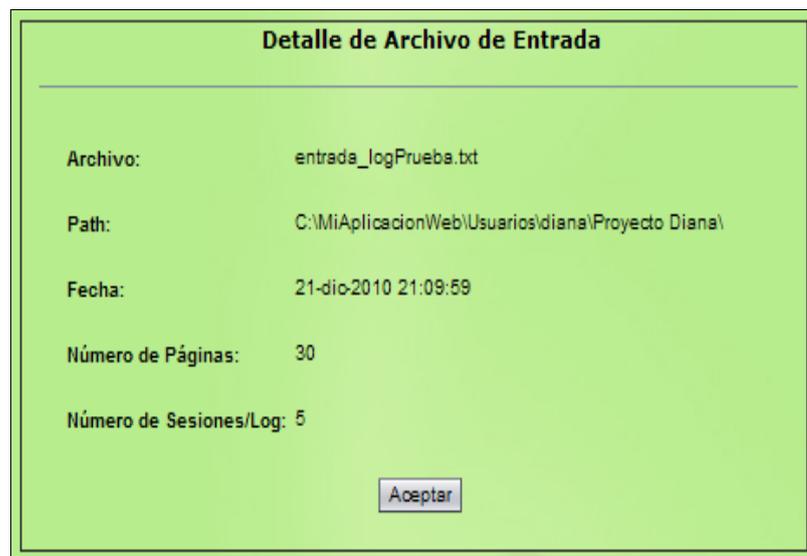
Pantallas



The screenshot shows a web interface with a light green background. At the top, there is a header "Archivos de Entrada". Below the header is a table with the following data:

Archivo	Path	Fecha	Opción
<input type="radio"/> entrada_logPrueba.txt	C:\MiAplicacionWeb\Usuarios\diana\Proyecto Diana\	21-dic-2010 21:09:59	Detalle

Figura 97: Página "Archivos de Entrada"

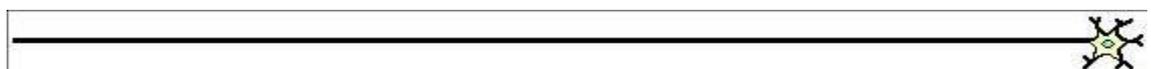


The screenshot shows a web interface with a light green background. At the top, there is a header "Detalle de Archivo de Entrada". Below the header, the following information is displayed:

Archivo: entrada_logPrueba.txt
Path: C:\MiAplicacionWeb\Usuarios\diana\Proyecto Diana\
Fecha: 21-dic-2010 21:09:59
Número de Páginas: 30
Número de Sesiones/Log: 5

At the bottom of the form, there is a button labeled "Aceptar".

Figura 98: Página "Detalle de Archivo de Entrada"



NOMBRE DEL CASO DE USO: VER EL DETALLE DE UN ARCHIVO DE ENTRADA		CÓDIGO: C.U.021
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 028 - RF 029	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Listar y ver el detalle de los archivos de entrada generados dentro de un proyecto.	El usuario registrado y que ha iniciado sesión en el sistema selecciona de la lista "Archivos de Entrada" un archivo de entrada para obtener el detalle del mismo.	
PRE-CONDICIONES	POST-CONDICIONES	
- El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe construir un archivo de entrada a la Red Neuronal Artificial SOM. - El usuario debe estar visualizando la página principal del proyecto.	Se ha recopilado información sobre el archivo de entrada seleccionado.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<p>1. El usuario hace clic en el elemento “Entrada SOM” del menú “Ver Archivos”.</p> <p>4. El usuario selecciona un archivo de entrada del cual desea ver el detalle haciendo clic en el link “Detalle” del listado “Archivos de Entrada”.</p> <p>6. El usuario hace clic en el botón “Aceptar”.</p> <p>8. El usuario hace clic en el botón “Cerrar”.</p>	<p>2. El sistema recopila todos los archivos del tipo seleccionado.</p> <p>3. El sistema muestra en la lista “Archivos de Entrada” con todos los archivos que fueron recopilados.</p> <p>5. El sistema muestra la página “Detalle de Archivo de Entrada” indicando el nombre, path, fecha, número de páginas y número de sesiones/log del archivo de entrada.</p> <p>7. El sistema muestra la lista “Archivos de Entrada”.</p> <p>9. El sistema muestra la pantalla con el menú principal del proyecto.</p>
FLUJO ALTERNO	
<p>A. NO EXISTEN ARCHIVOS DE ENTRADA</p> <p>A3. El sistema muestra el mensaje de información “No existen archivos de entrada”.</p> <p>A4. El usuario hace clic en el botón “Cerrar”</p> <p>A5. El caso de uso termina.</p>	



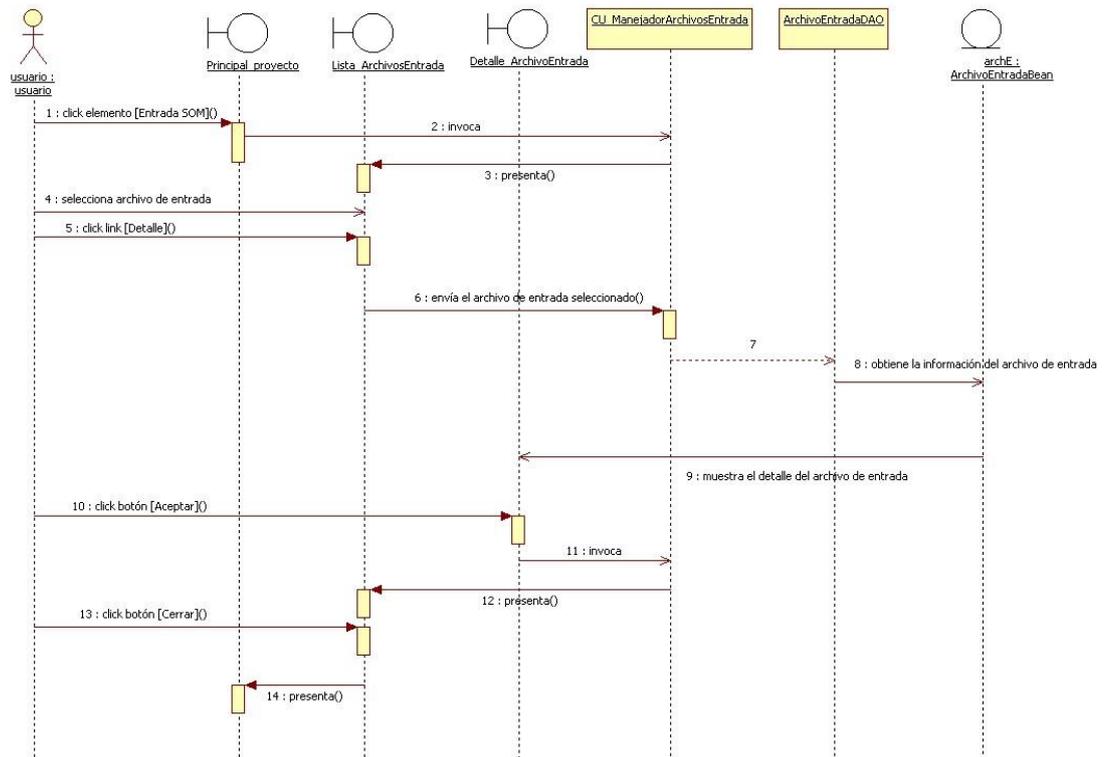


Figura 99: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo de entrada (Flujo normal)



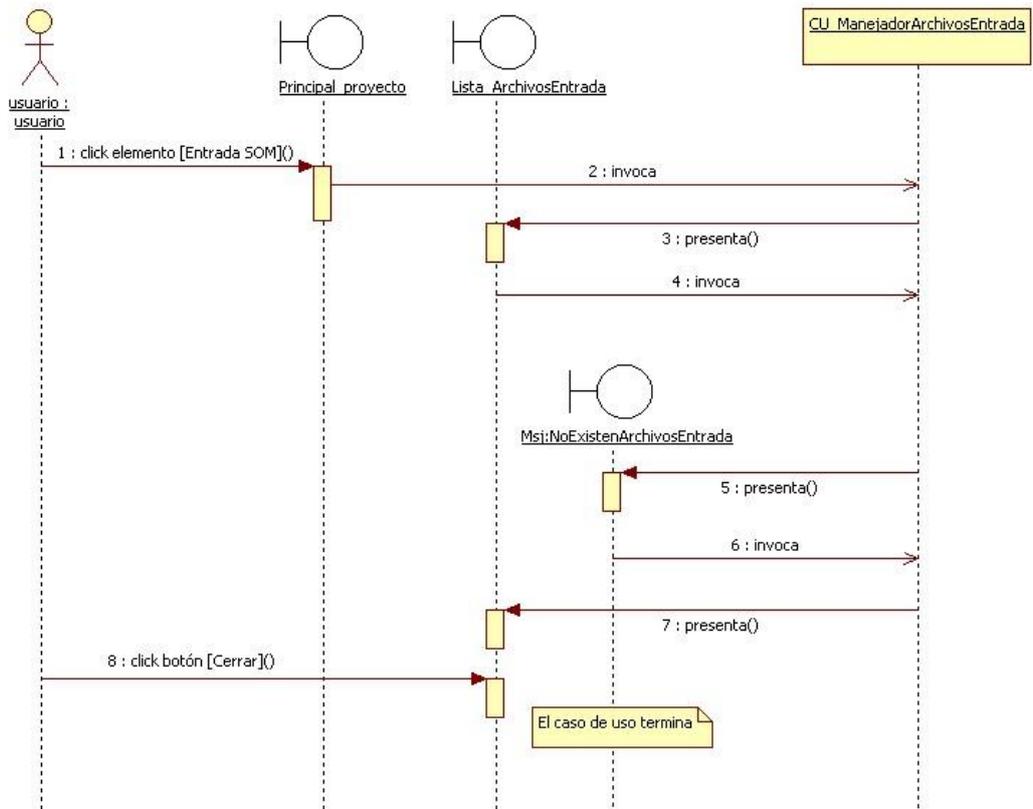


Figura 100: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo de entrada (Flujo alternativo A)

6.3.2.22. Caso de Uso: Ver el detalle de un archivo de salida

Pantallas

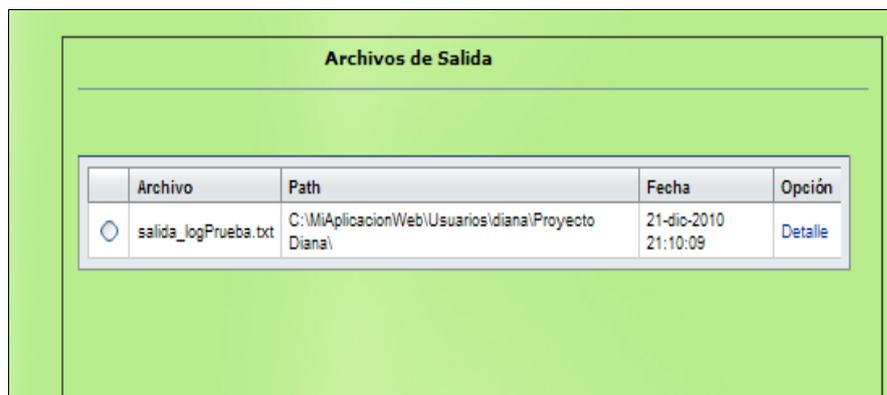


Figura 101: Página “Archivos de Salida”



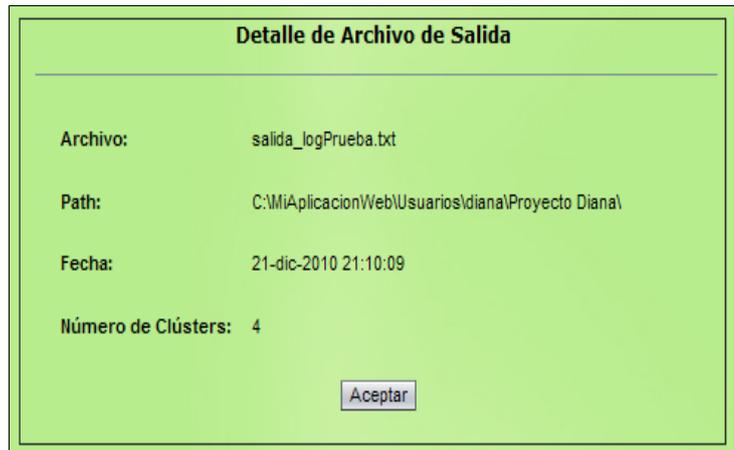


Figura 102: Página “Detalle de Archivo de Salida”

NOMBRE DEL CASO DE USO: VER EL DETALLE DE UN ARCHIVO DE SALIDA		CÓDIGO: C.U.022
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
<ul style="list-style-type: none"> - RF 028 - RF 029 	<ul style="list-style-type: none"> - Usuario Administrador - Usuario Común 	<ul style="list-style-type: none"> - Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Listar y ver el detalle de los archivos de salida generados dentro de un proyecto.	El usuario registrado y que ha iniciado sesión en el sistema selecciona de la lista “Archivos de Salida” un archivo de salida para obtener el detalle del mismo.	
PRE-CONDICIONES	POST-CONDICIONES	
<ul style="list-style-type: none"> - El usuario debe estar registrado en el sistema. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe construir un archivo de salida de la Red Neuronal Artificial SOM. - El usuario debe estar visualizando la página principal del proyecto. - 	Se ha recopilado información sobre el archivo de salida seleccionado.	



FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<p>1. El usuario hace clic en el elemento “Salida SOM” del menú “Ver archivos”.</p> <p>4. El usuario selecciona el archivo de salida del cual desea ver el detalle haciendo clic en el link “Detalle” del listado “Archivos de Salida”.</p> <p>6. El usuario hace clic en el botón “Aceptar”.</p> <p>8. El usuario hace clic en el botón “Cerrar”.</p>	<p>2. El sistema recopila todos los archivos del tipo seleccionado.</p> <p>3. El sistema muestra la lista “Archivos de Salida” con todos los archivos que fueron recopilados.</p> <p>5. El sistema muestra la página “Detalle de Archivo de Salida” indicando el nombre, path, fecha y número de clústers del archivo de salida.</p> <p>7. El sistema muestra la lista “Archivos de Salida”.</p> <p>9. El sistema muestra la pantalla con el menú principal del proyecto.</p>
FLUJO ALTERNO	
<p>A. NO EXISTEN ARCHIVOS DE SALIDA</p> <p>A3. El sistema muestra el mensaje de información “No existen archivos de salida”.</p> <p>A4. El usuario hace clic en el botón “Cerrar”</p> <p>A5. El caso de uso termina.</p>	



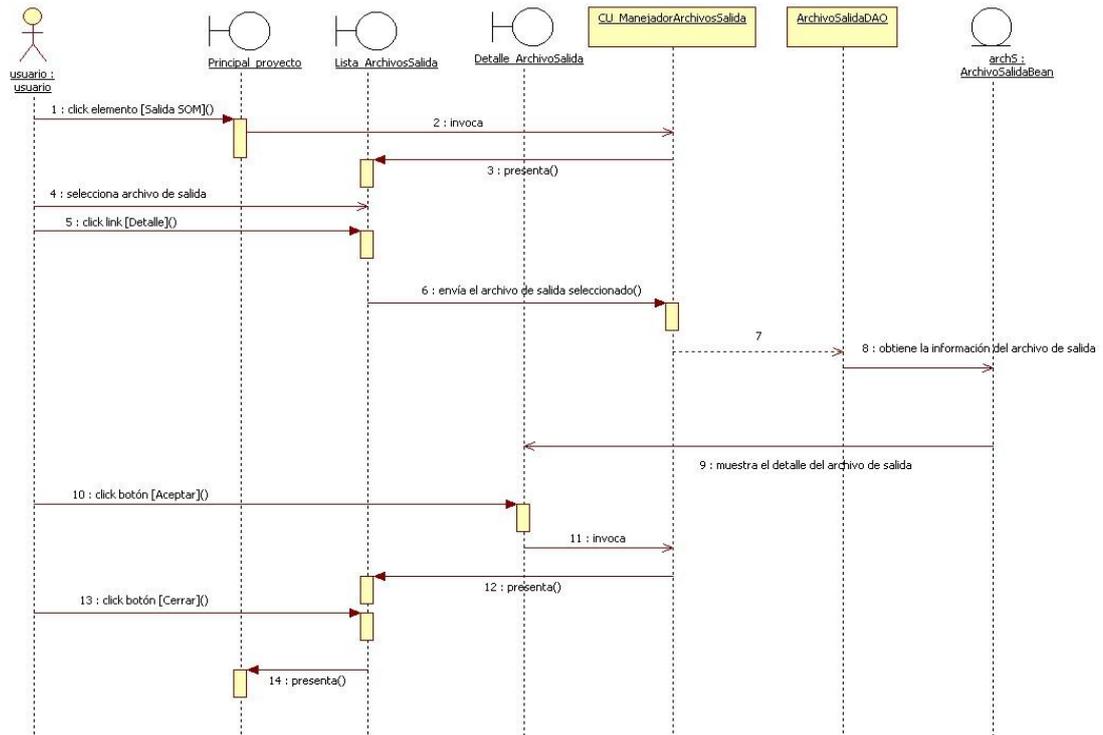


Figura 103: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo de salida (Flujo normal)

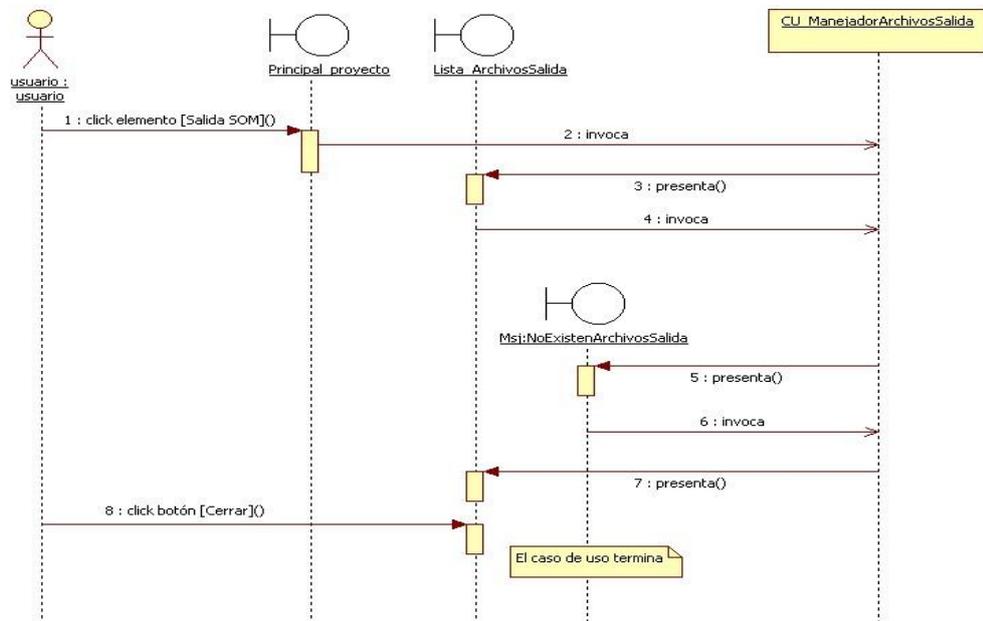


Figura 104: Diagrama de secuencia Caso de Uso: Ver el detalle de un archivo de salida (Flujo alterno A)



6.3.2.23. Caso de Uso: Cerrar un proyecto

Pantalla

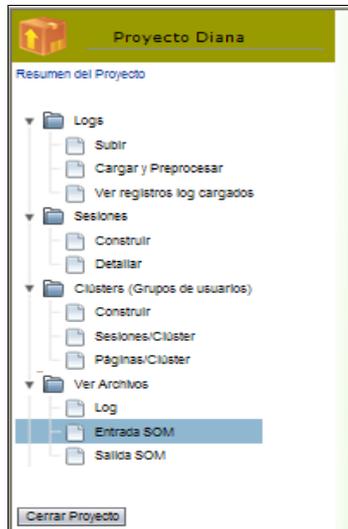


Figura 105: Pantalla “Cerrar Proyecto”

NOMBRE DEL CASO DE USO: CERRAR UN PROYECTO		CÓDIGO: C.U.023
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U.
- RF 010	- Usuario Administrador - Usuario Común	- Sistema
OBJETIVO (S)	DESCRIPCIÓN	
Cerrar un proyecto.	Cuando el usuario registrado y que ha iniciado sesión en el sistema ya no desea trabajar en su proyecto actual tiene la opción de cerrarlo.	



PRE-CONDICIONES	POST-CONDICIONES
<ul style="list-style-type: none"> - El usuario debe estar registrado. - El usuario debe haber iniciado sesión en el sistema. - El usuario debe abrir un proyecto. - El usuario debe estar visualizando la página que contiene el menú principal del proyecto. 	Se ha cerrado el proyecto actual del usuario.
FLUJO NORMAL	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
<p>1. El usuario hace clic en el botón “Cerrar proyecto” que se encuentra ubicado en la parte inferior del menú principal del proyecto establecido como actual.</p>	<p>2. El sistema guarda los cambios realizados en el proyecto.</p> <p>3. El sistema muestra la pantalla que contiene el listado de los proyectos.</p>

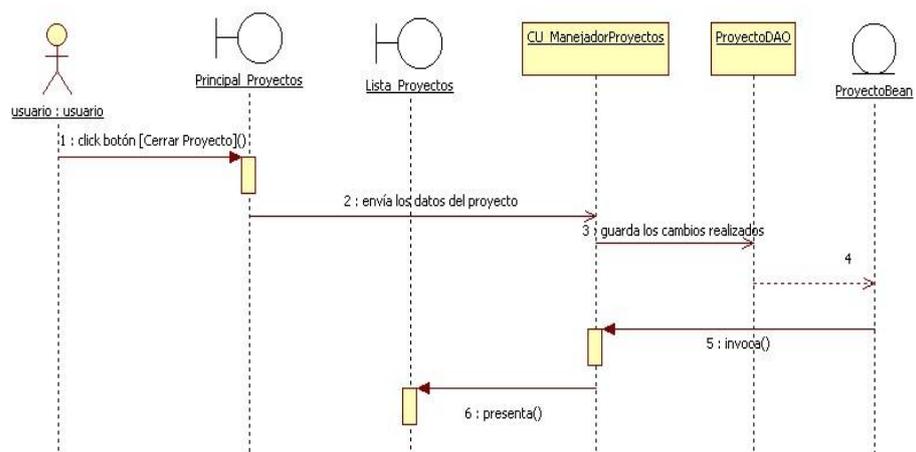


Figura 106: Diagrama de secuencia Caso de Uso: Cerrar un proyecto



6.4. MODELAMIENTO ESTÁTICO

6.4.1. Diagrama de Paquetes

preferenciasweb

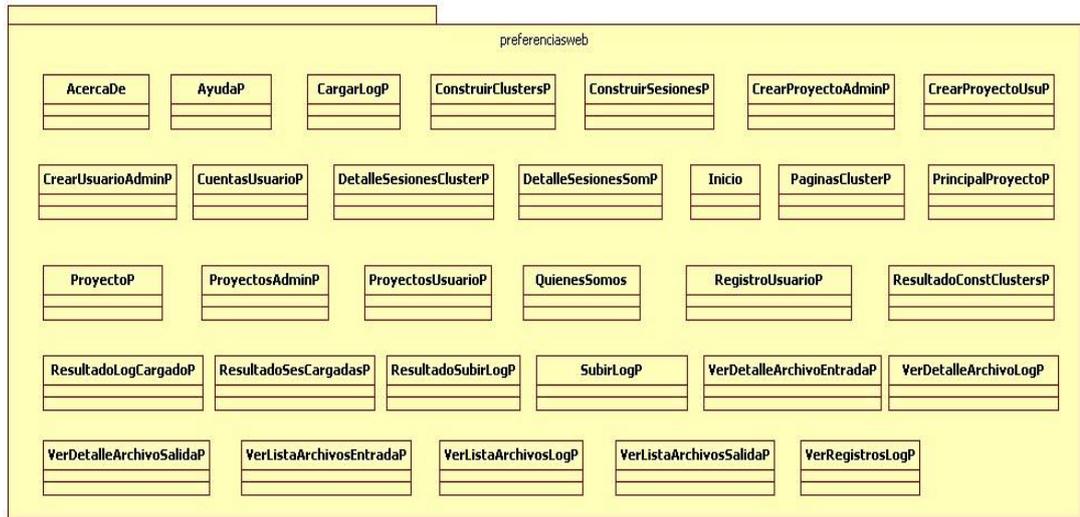


Figura 107: Diagrama de paquetes preferenciasweb

preferenciasweb.fragmentos

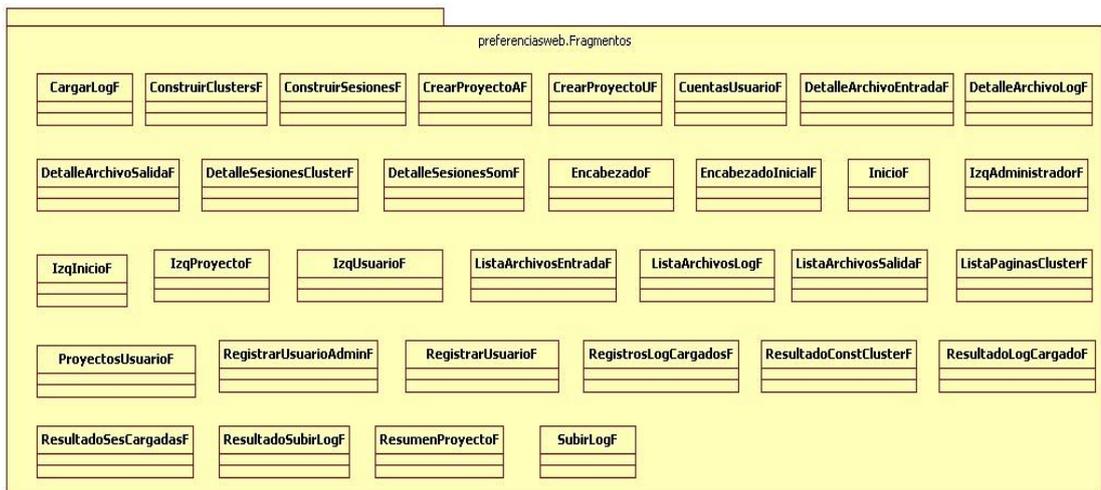


Figura 108: Diagrama de paquetes preferenciasweb.fragmentos



preferenciasweb.control

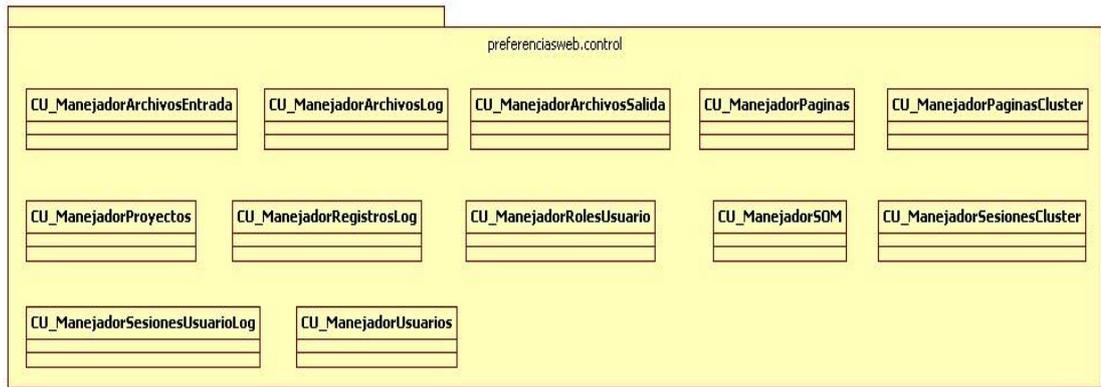


Figura 109: Diagrama de paquetes preferenciasweb.control

preferenciasweb.dao

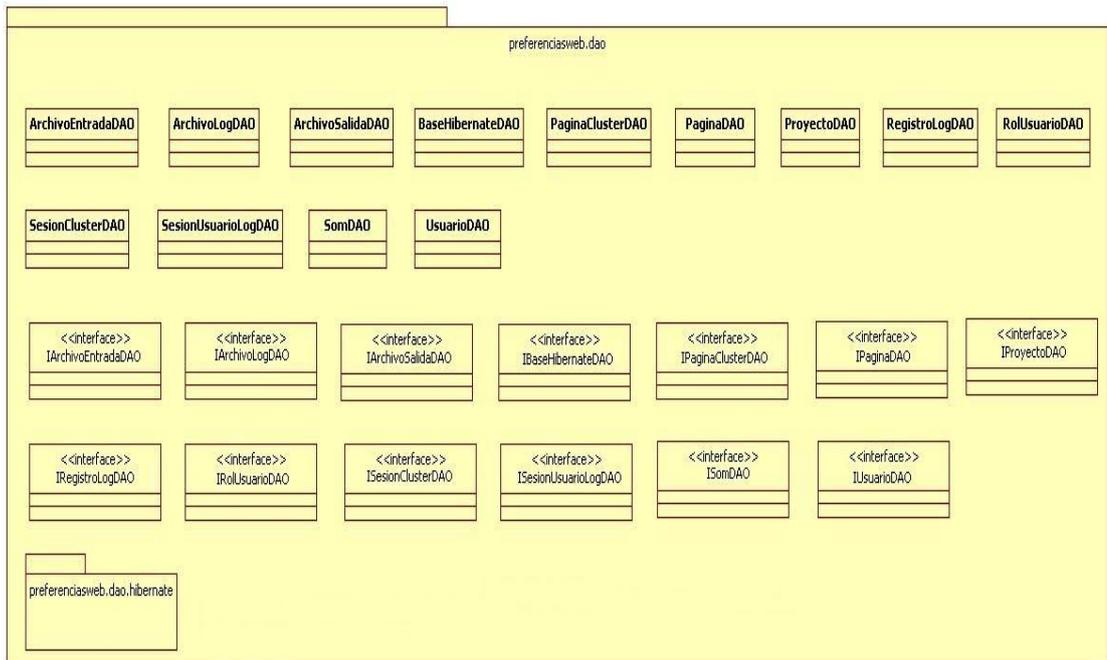


Figura 110: Diagrama de paquetes preferenciasweb.dao



preferenciasweb.dao.hibernate

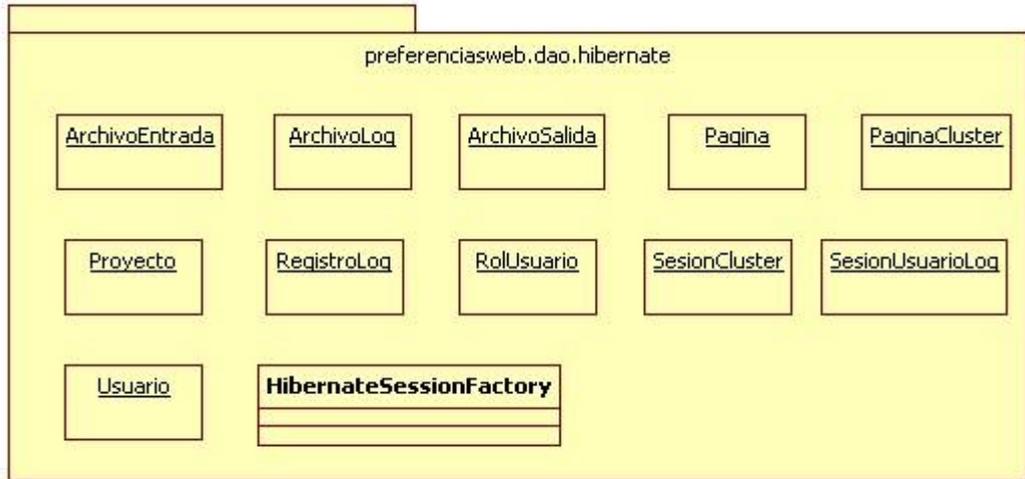


Figura 111: Diagrama de paquetes preferenciasweb.dao.hibernate

preferenciasweb.modelo

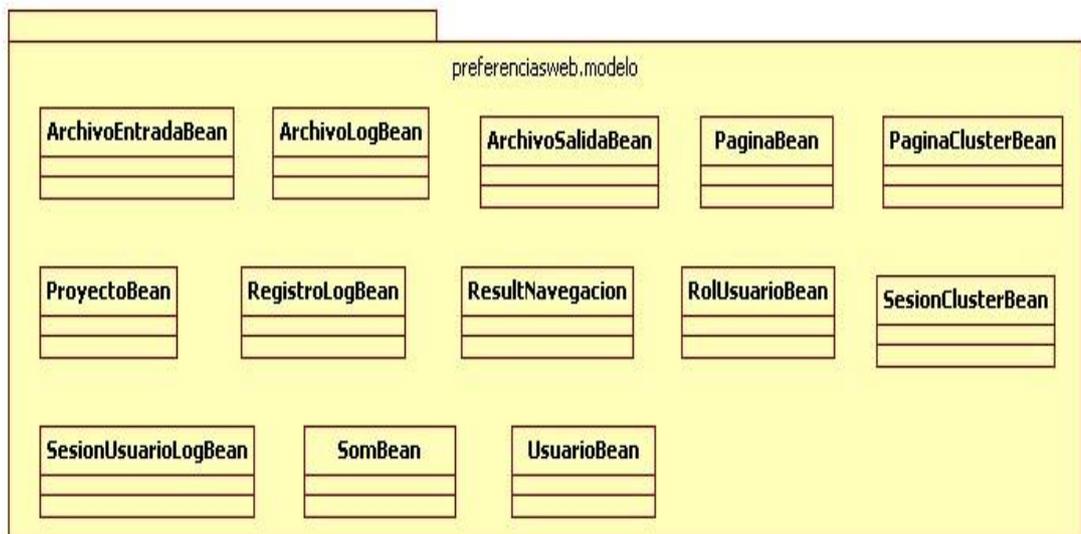


Figura 112: Diagrama de paquetes preferenciasweb.modelo



• Caso de Uso: Iniciar sesión en el sistema

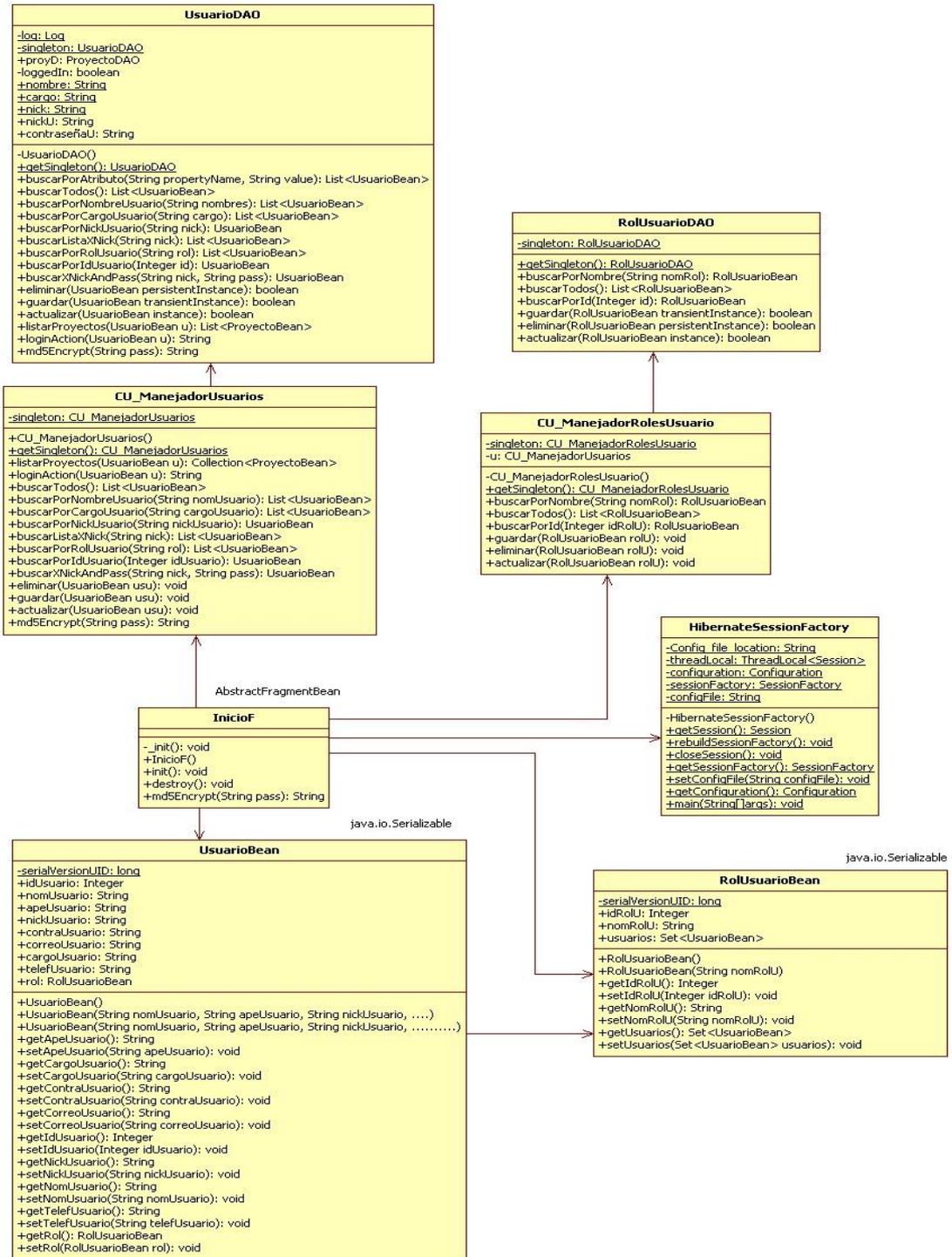


Figura 115: Diagrama de clases Caso de Uso: Iniciar sesión en el sistema



Caso de Uso: Crear una cuenta de usuario



Figura 116: Diagrama de clases Caso de Uso: Crear una cuenta de usuario



• Caso de Uso: Ver las cuentas de usuario

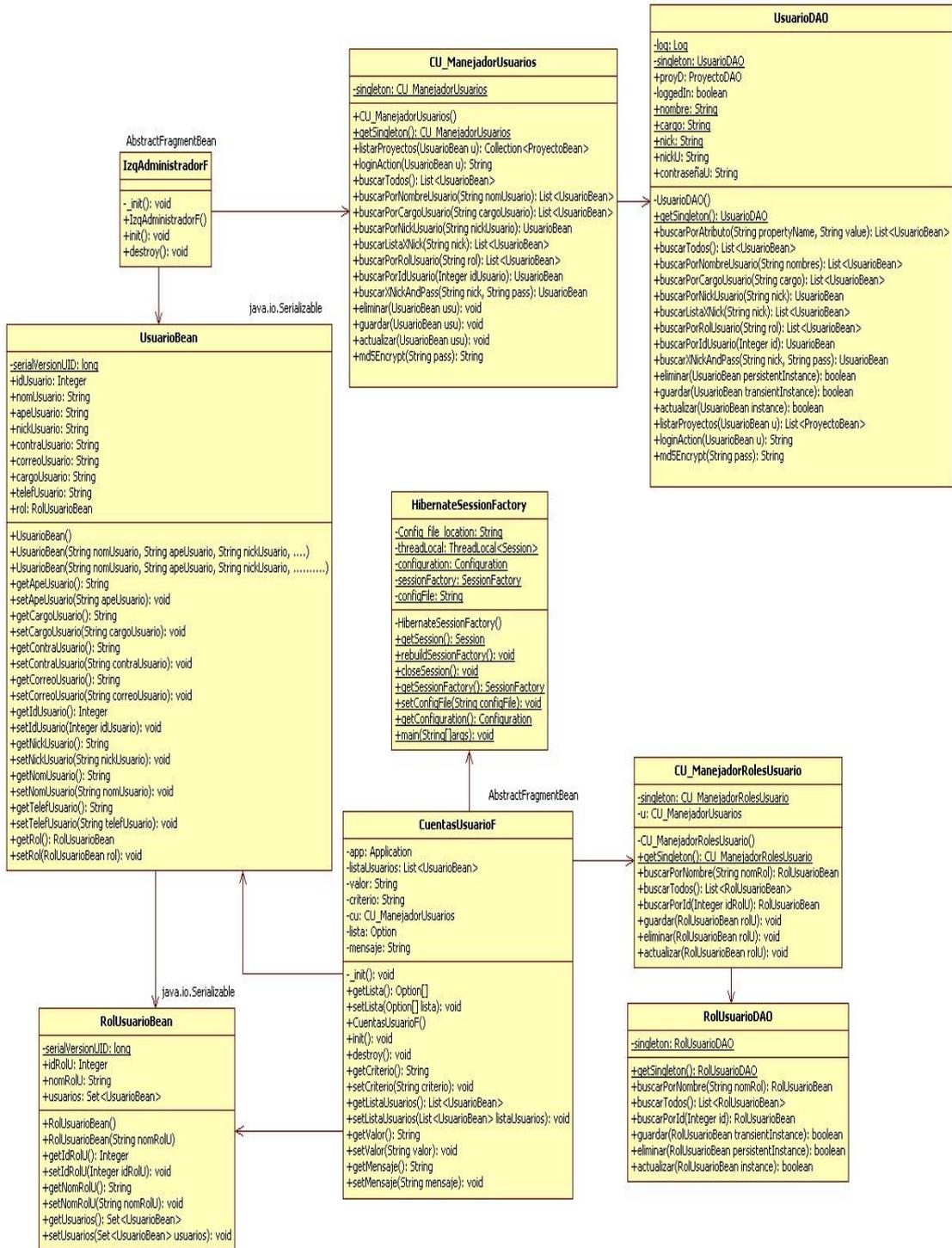


Figura 117: Diagrama de clases Caso de Uso: Ver las cuentas de usuario

• Caso de Uso: Eliminar las cuentas de usuario

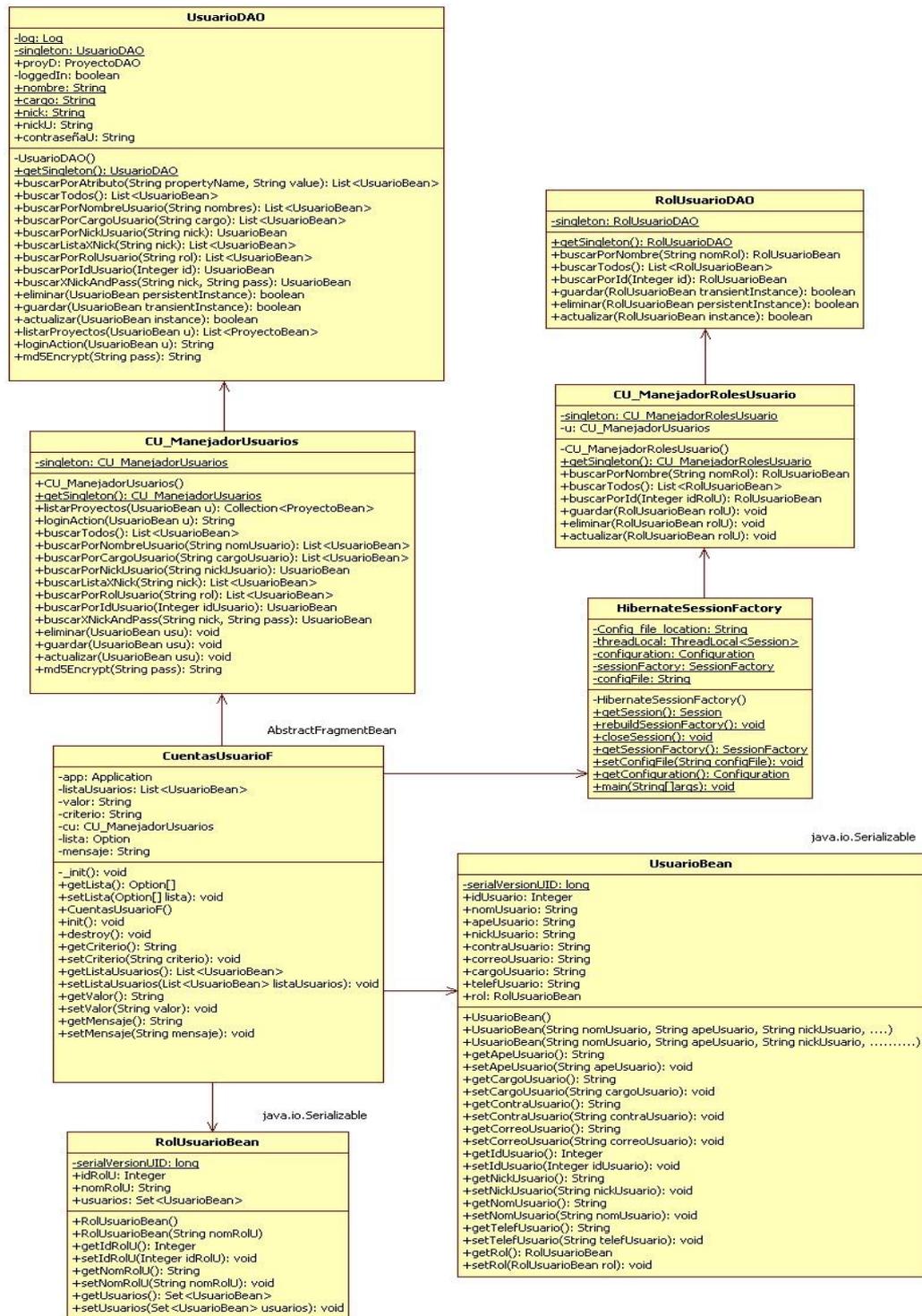


Figura 118: Diagrama de clases Caso de Uso: Eliminar las cuentas de usuario



• Caso de Uso: Crear un proyecto (Usuario Administrador)

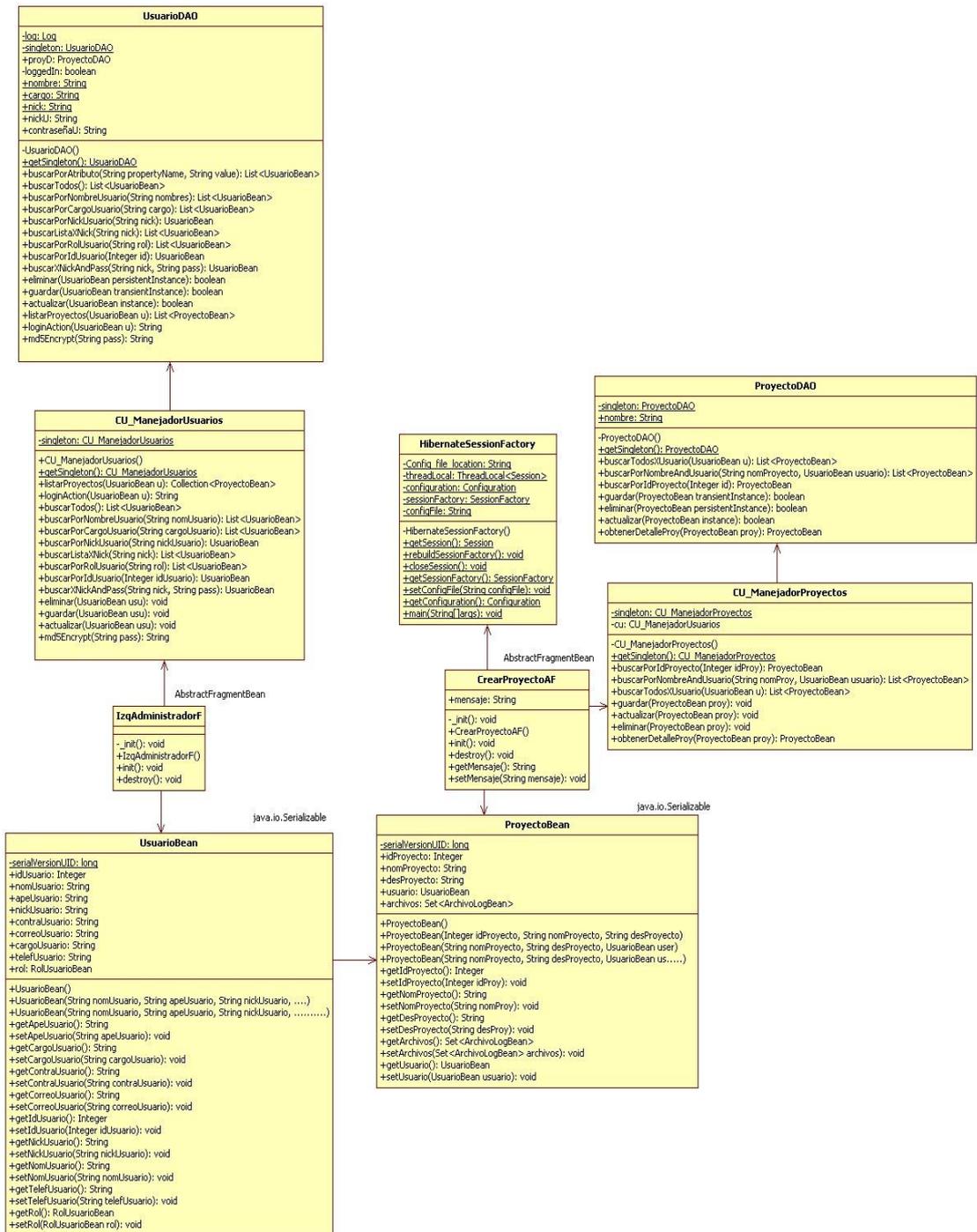


Figura 119: Diagrama de clases Caso de Uso: Crear un proyecto (Usuario Administrador)



• **Caso de Uso: Crear un proyecto (Usuario Común)**

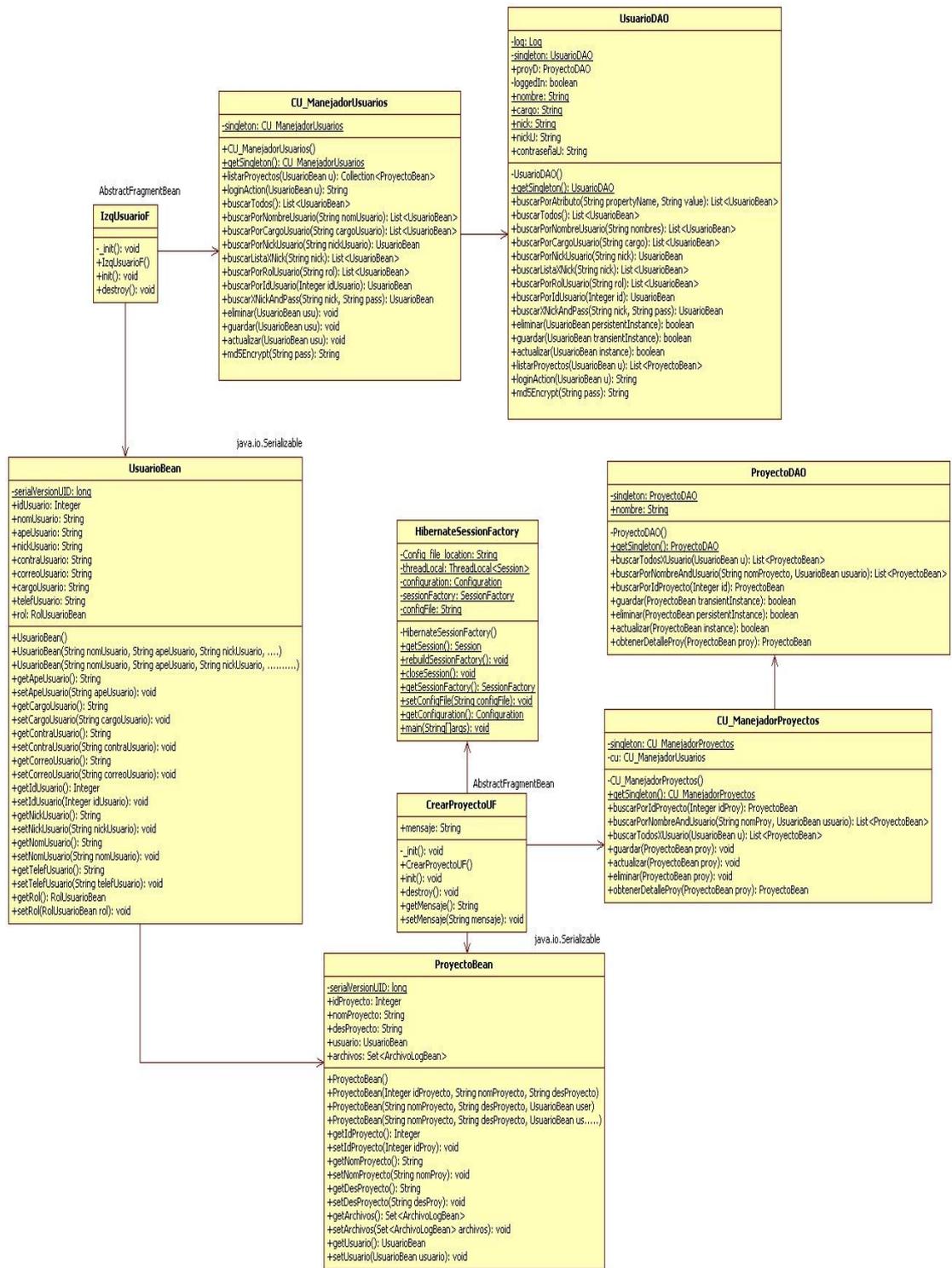


Figura 120: Diagrama de clases- Caso de Uso: Crear un proyecto (Usuario Común)



• Caso de Uso: Mostrar todos los proyectos de un usuario

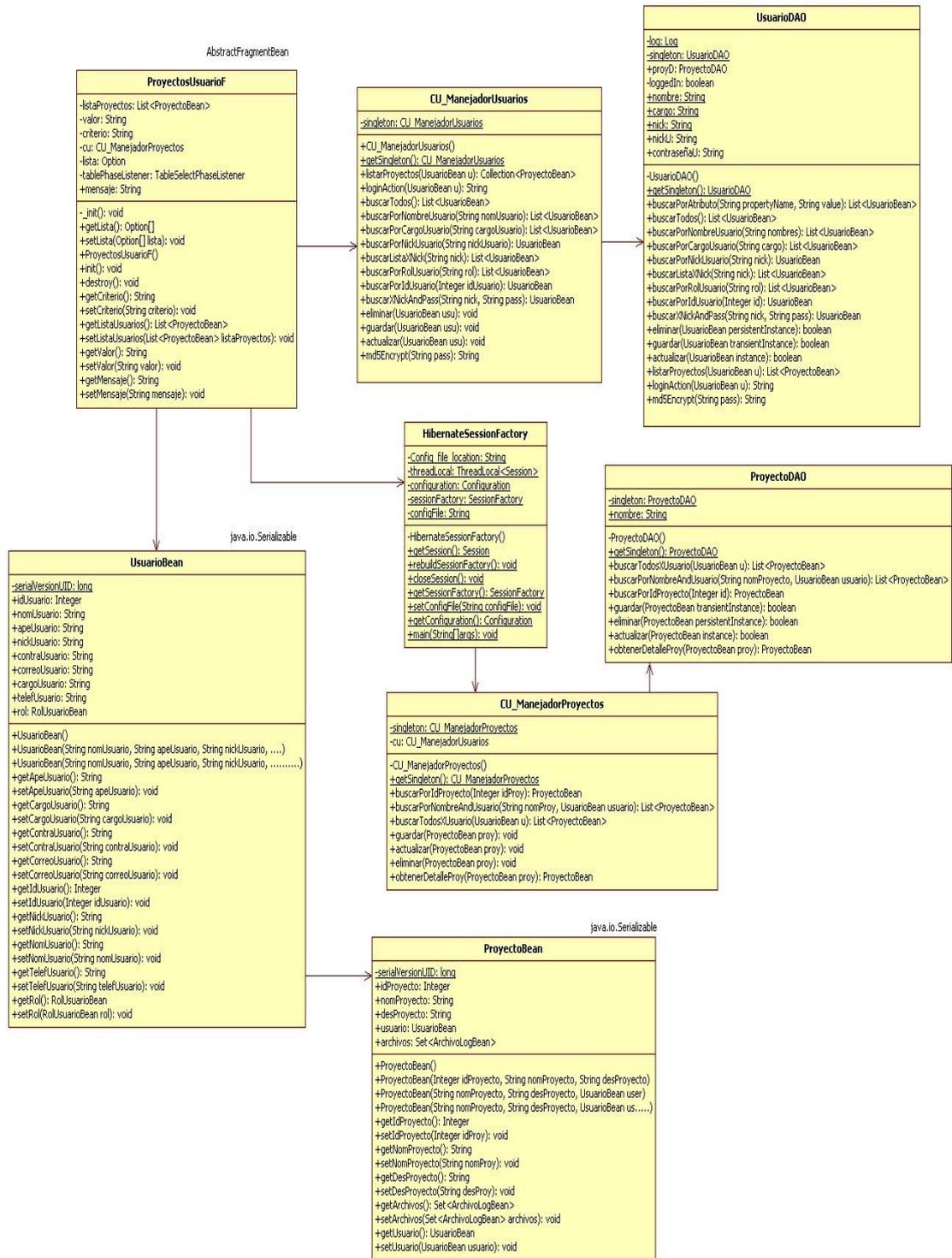


Figura 121: Diagrama de clases Caso de Uso: Mostrar todos los proyectos de un usuario



• Caso de Uso: Eliminar un proyecto

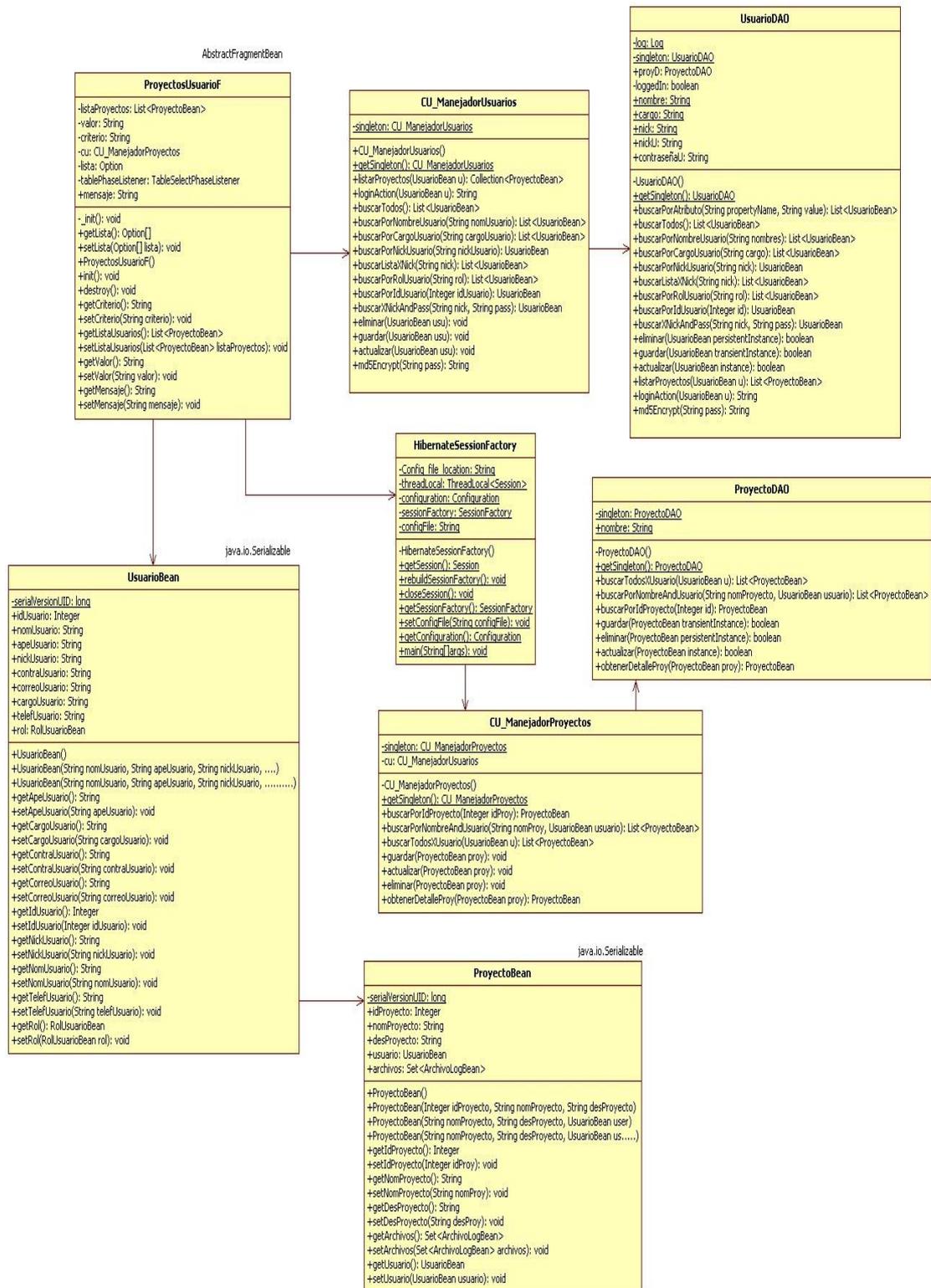


Figura 122: Diagrama de clases Caso de Uso: Eliminar un proyecto



• Caso de Uso: Abrir un proyecto



Figura 123: Diagrama de clases Caso de Uso: Abrir un proyecto



página por los usuarios de cada clúster

• Caso de Uso: Ver el detalle de un archivo log

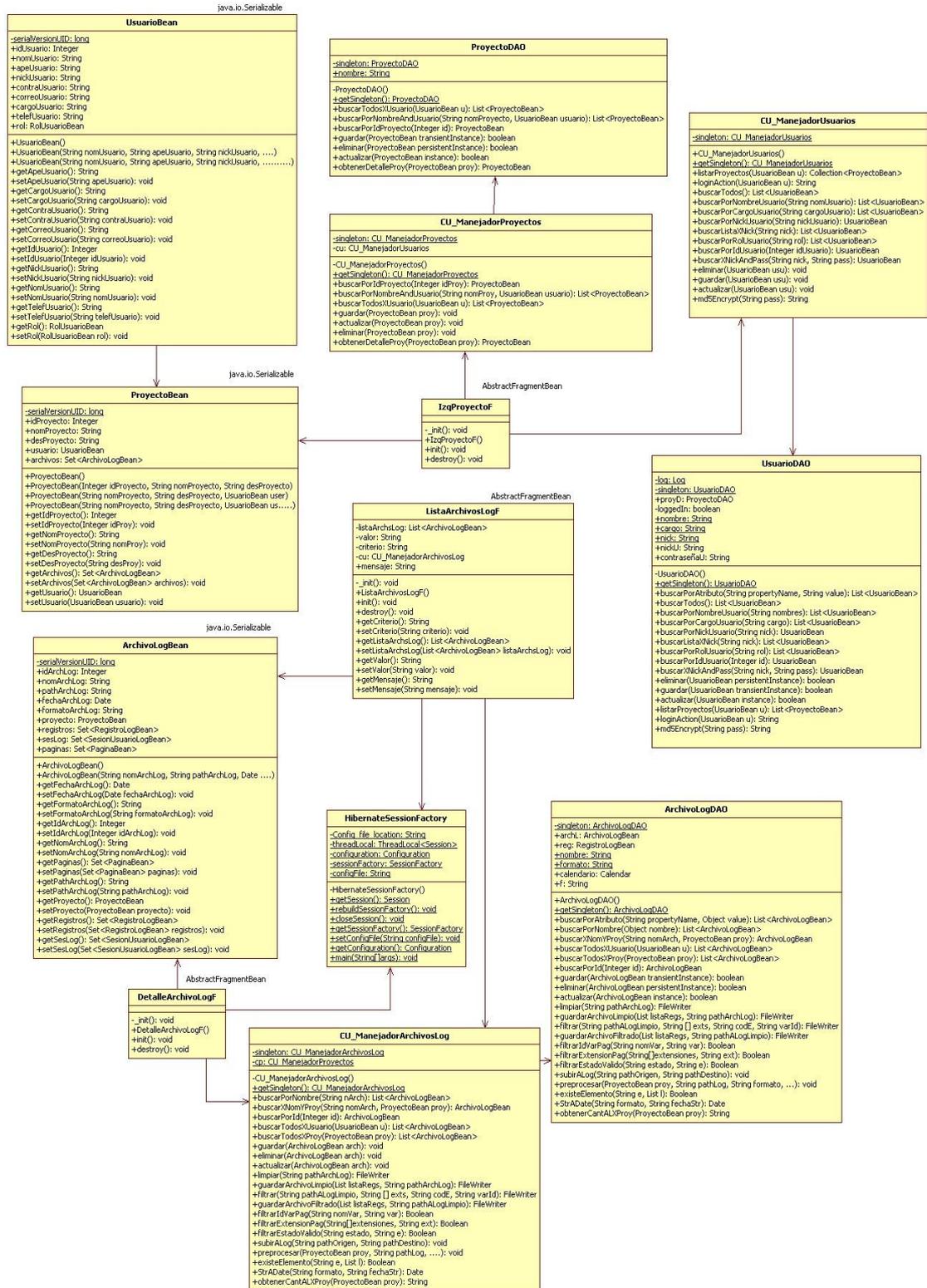


Figura 133: Diagrama de clases Caso de Uso: Ver el detalle de un archivo log

• Caso de Uso: Eliminar un archivo log

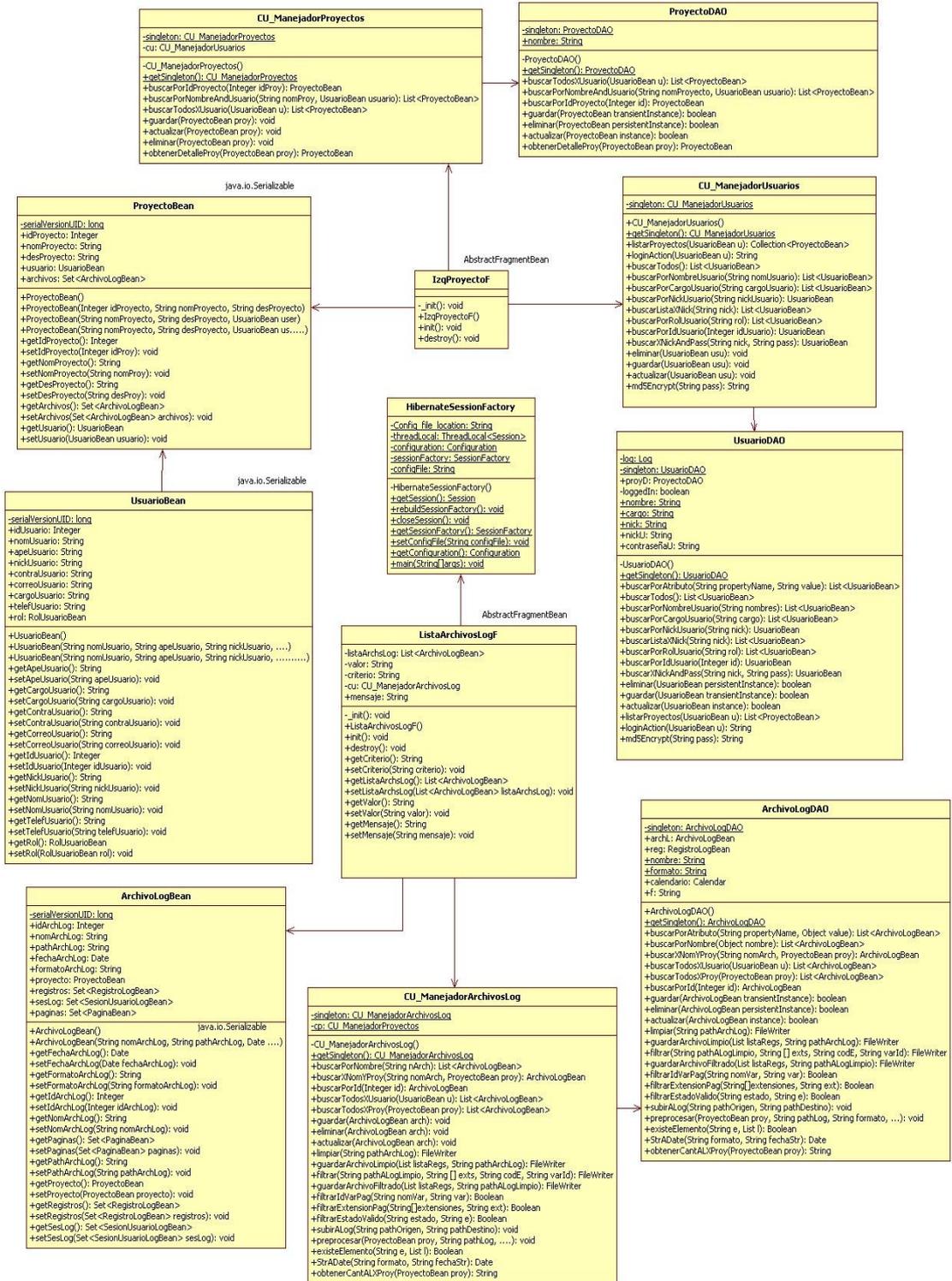


Figura 136: Diagrama de clases Caso de Uso: Ver el detalle de un archivo de salida

• Caso de Uso: Cerrar un proyecto



Figura 137: Diagrama de clases Caso de Uso: Cerrar un proyecto



6.4.4. Modelo y Arquitectura

6.4.4.1. Diagrama de Componentes

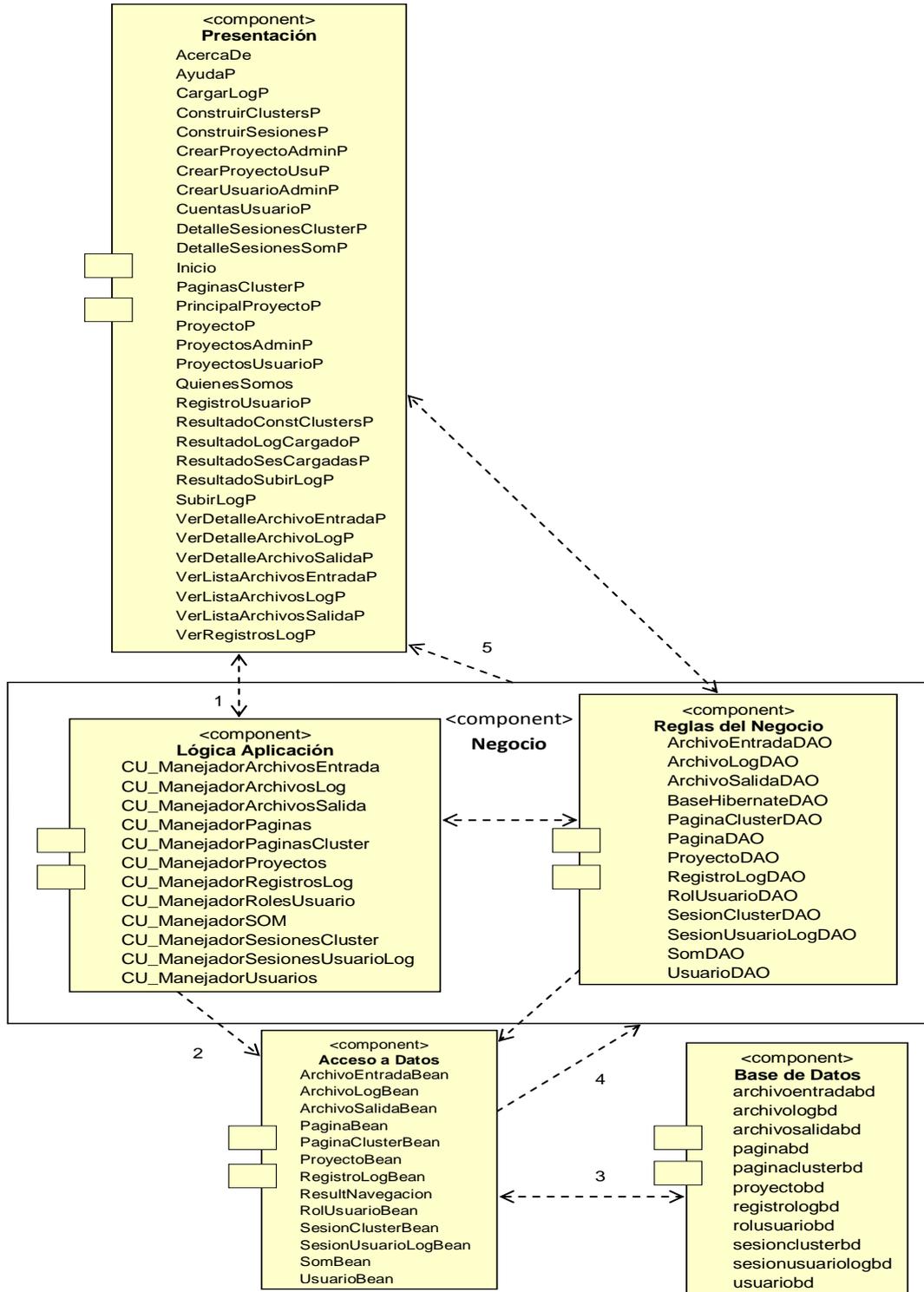


Figura 138: Diagrama de componentes

6.4.4.2. Diagrama de base de datos

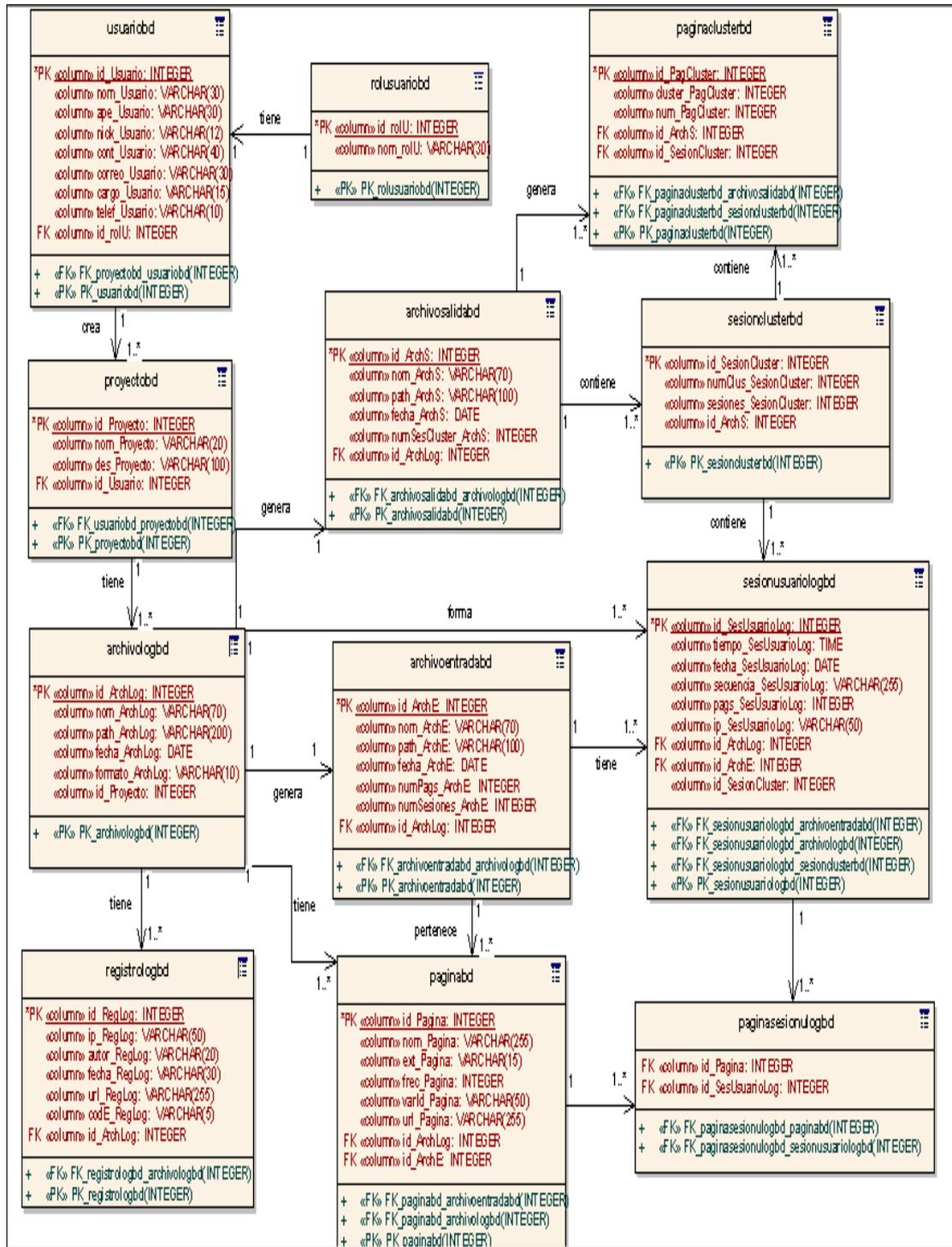


Figura 139: Diagrama de base de datos



6.4.4.3. Diagrama de Despliegue

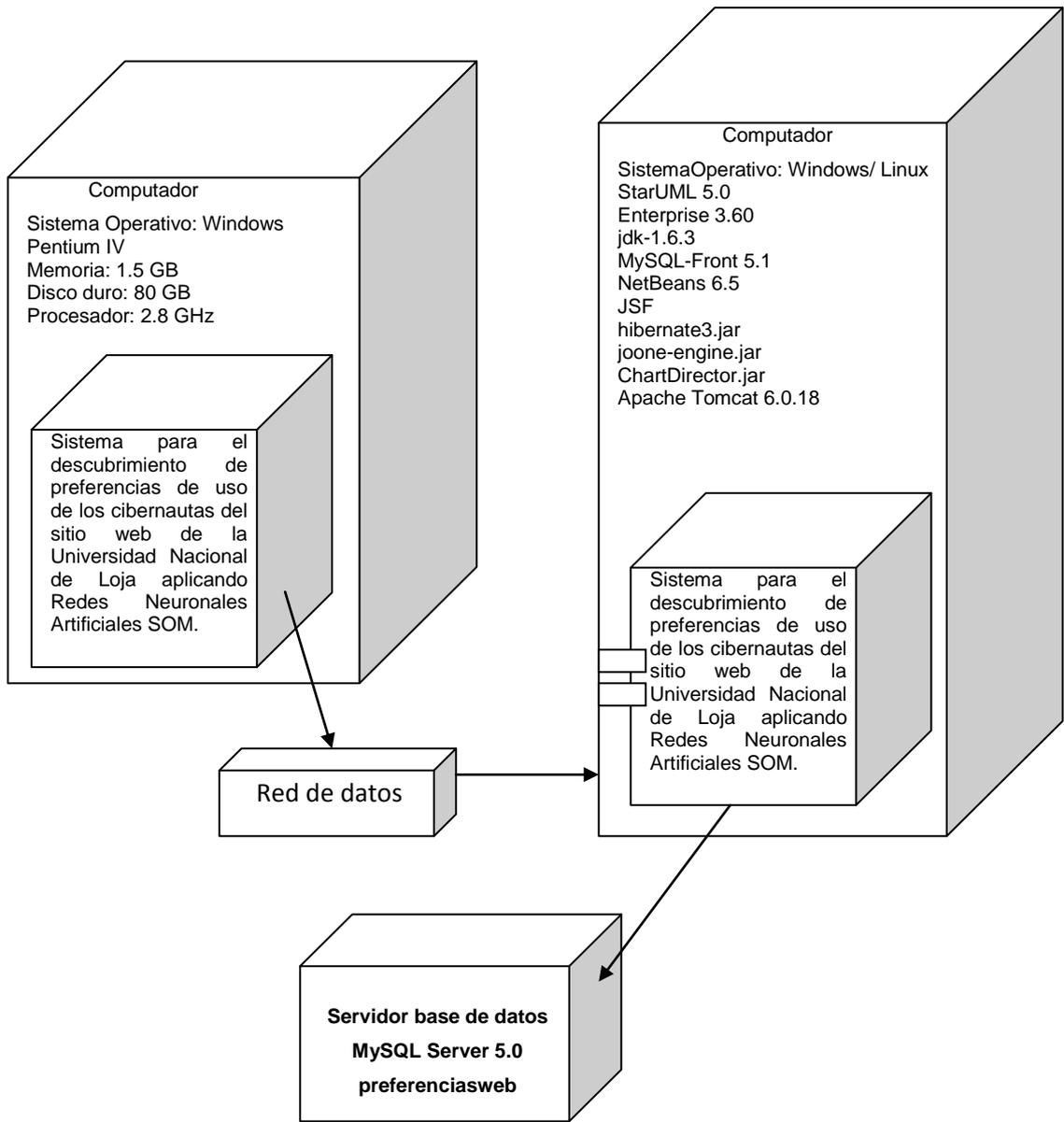


Figura 140: Diagrama de despliegue



6.5. PRUEBAS Y VALIDACIÓN

6.5.1. Antecedentes

Para las pruebas de la aplicación primeramente se solicitó mediante un oficio el 06 de enero del 2011 al Director del área de la Energía, las Industrias y los Recursos Naturales No Renovables designe a quien corresponda nos facilite para el 13 de enero del 2011 una clave de súper usuario para realizar un acceso remoto con la herramienta FreeNX al servidor web de la UNL con la finalidad de extraer los archivos log objeto de análisis de la presente tesis. La responsabilidad de esta tarea fue designada al Ing. Patricio Villamarín, quien a su vez delegó la responsabilidad al Ing. Jamil Ramón Director del Departamento de Informática de la UNL.

Es importante indicar que fue el Ing. Jamil Ramón él que en primera instancia al proponer el tema de tesis quién sugirió pedir esta clave para realizar las futuras pruebas de la aplicación en caso de aprobación del tema de tesis propuesto y además se comprometió a ayudarnos con los archivos log del servidor web para poder desarrollar la aplicación, lo cual si fue cumplido.

Después de que se delegó la responsabilidad al Ing. Jamil Ramón, el 12 de enero del 2011, nos manifestó que por motivos de seguridad del sitio web no podía conceder una clave de súper usuario para realizar el acceso remoto, a su vez se comprometió a seguir facilitando los archivos log del servidor web para llevar a cabo las pruebas y que hagamos conocer del particular al Ing. Wilman Chamba Director de Tesis.

6.5.2. Descripción de las pruebas

Para el desarrollo de las pruebas se utilizó la metodología de desarrollo de sistemas denominada Metodología Métrica Versión 3.

A continuación se detallan todas las actividades de la metodología realizadas para el desarrollo de la herramienta que implementa la solución propuesta en esta tesis.



6.5.2.1. Funcionalidades

El Banco de Pruebas provee las siguientes funcionalidades:

6.5.2.1.1. Pre-procesamiento de archivos log

Para el presente trabajo se decidió utilizar el formato de log común (CLF) debido a la gran utilización del mismo. También se permite la utilización del formato de log común más los campos Referrer y Agent.

Se realizó la herramienta parametrizable, de modo que las tareas de limpieza puedan ser adaptadas a las necesidades de cada caso. Para ello, se utilizaron distintos filtros a los archivos log, los cuales permiten que el usuario defina su comportamiento.

El primer filtro que se implementó es un filtro de extensiones de recursos. El usuario puede indicar cuáles extensiones de los recursos deben ser incluidas en el armado de las sesiones de los usuarios.

El segundo filtro disponible es sobre la base de los códigos de error de http. El usuario indica el código de error para la tarea de la construcción de las sesiones de usuarios. Solo las líneas del archivo de log que posean el código de error especificado por el usuario serán incluidas en el proceso de construcción de sesiones. En forma predeterminada, solo el código de error 200 es considerado.

El tercer filtro disponible viene dado por el amplio empleo de páginas generadas dinámicamente. Muchas de estas páginas poseen una plantilla o un controlador, donde son incluidas las distintas páginas. Por ende, en los archivos de log, queda registrada la petición solamente a la página correspondiente a la plantilla o al controlador. En varias ocasiones, se puede diferenciar a las distintas páginas reales incluidas en un sitio debido al uso de una variable que identifica la página a ser incluida. Esta variable es comunicada generalmente mediante el método GET al servidor, por lo cual, el valor de esta variable aparece en la URL de la petición. No es el caso si el método utilizado es POST. El filtro permite indicar qué variables identifican a páginas distintas dentro de una misma plantilla o controlador, de modo que puedan ser reconocidas las distintas páginas reales visitadas por el usuario.



6.5.2.1.2. Identificación de usuarios

Para la identificación de los usuarios a partir del archivo log pre-procesado, se utilizó la dirección IP. Todas las peticiones originadas desde una misma dirección IP son consideradas pertenecientes al mismo usuario.

6.5.2.1.3. Identificación de sesiones de usuario

Consiste en la construcción de las sesiones de usuarios. Las sesiones fueron divididas teniendo en cuenta un tiempo máximo entre peticiones consecutivas consideradas pertenecientes a la misma sesión de usuario. Este tiempo es configurable por el usuario, teniendo el valor por defecto de treinta minutos. También es posible indicar la cantidad mínima de páginas en una sesión y la frecuencia mínima de cada página.

6.5.2.1.3.1. Identificación de preferencias de uso mediante Red Neuronal Mapa Auto-Organizativo (SOM)

La solución propuesta consiste en la utilización de una Red Neuronal Artificial SOM para la identificación de las preferencias de los usuarios. Esta red neuronal debe agrupar a los usuarios del sitio web de la UNL sobre la base de las páginas accedidas por los mismos. Para ello, se debe pre-procesar el archivo de log, para identificar a los usuarios y a las sesiones de los mismos. Luego, con estas sesiones de usuarios es entrenada la red, para agrupar a los usuarios de forma automática. La elección de la Red Neuronal Artificial SOM se debe a que posee un entrenamiento no supervisado, permitiendo realizar el agrupamiento de los usuarios en forma automática, sin intervención del usuario más que para configurar algunos pocos parámetros para el análisis.

Luego de identificar todas las sesiones de los usuarios, se debe generar el formato adecuado para poder ingresar las sesiones como patrones a la Red Neuronal. Para cada sesión, se indica la presencia (1) o la ausencia (0) de cada página perteneciente al sitio web, resultando en un vector, de tamaño igual a la cantidad de páginas del sitio, para cada sesión. Cada elemento del vector es un número binario, indicando la presencia o no de la página representada por esa posición. Estos vectores corresponden con los patrones de entrada de la Red Neuronal utilizada para la



identificación de preferencias de los usuarios.

La Red Neuronal empleada es un mapa auto-organizativo (SOM), que posee tantas entradas como páginas frecuentes posea el sitio web.

Las páginas frecuentes son las que poseen una frecuencia mayor a la especificada por el usuario en la tarea de pre-procesamiento. La Red Neuronal se construye en forma dinámica para permitir las entradas necesarias para el sitio web.

La salida de la Red Neuronal SOM es un mapa de dos dimensiones de $N \times N$, donde N es configurable por el usuario y depende de la cantidad de clústers que el mismo desee obtener. En el mapa de salida, solo una salida será activada, indicado por el valor binario uno. Todas las demás salidas tendrán un valor nulo.

La salida activada representa el clúster (grupo de usuarios) al cual pertenece el patrón ingresado. Patrones similares pertenecerán al mismo clúster o a clústers cercanos en el mapa de salida.

6.5.2.1.4. Usuarios y proyectos

Permite la creación de usuarios y proyectos, de modo que se puedan realizar varios análisis independientes, controlando el acceso a los proyectos de cada usuario mediante una contraseña.

6.5.3. Dinámica de las Pruebas

La dinámica de las pruebas consiste en realizar los siguientes pasos:

- 1. Pre-procesamiento del archivo de log del servidor web:** indicando el archivo log que se utilizará, el formato del mismo y los parámetros deseados para el pre-procesamiento.
- 2. Identificación de usuarios y sus sesiones:** utilizando el archivo log pre-procesado de un proyecto, se procede a la identificación de los usuarios y sus sesiones. Se indican los parámetros necesarios para este paso.



3. Agrupamiento de usuarios utilizando Red Neuronal Artificial SOM o clusterización: se procede al agrupamiento de usuarios utilizando la Red Neuronal Artificial SOM, indicando los parámetros a utilizarse.

A continuación se indica la dinámica del banco de pruebas.

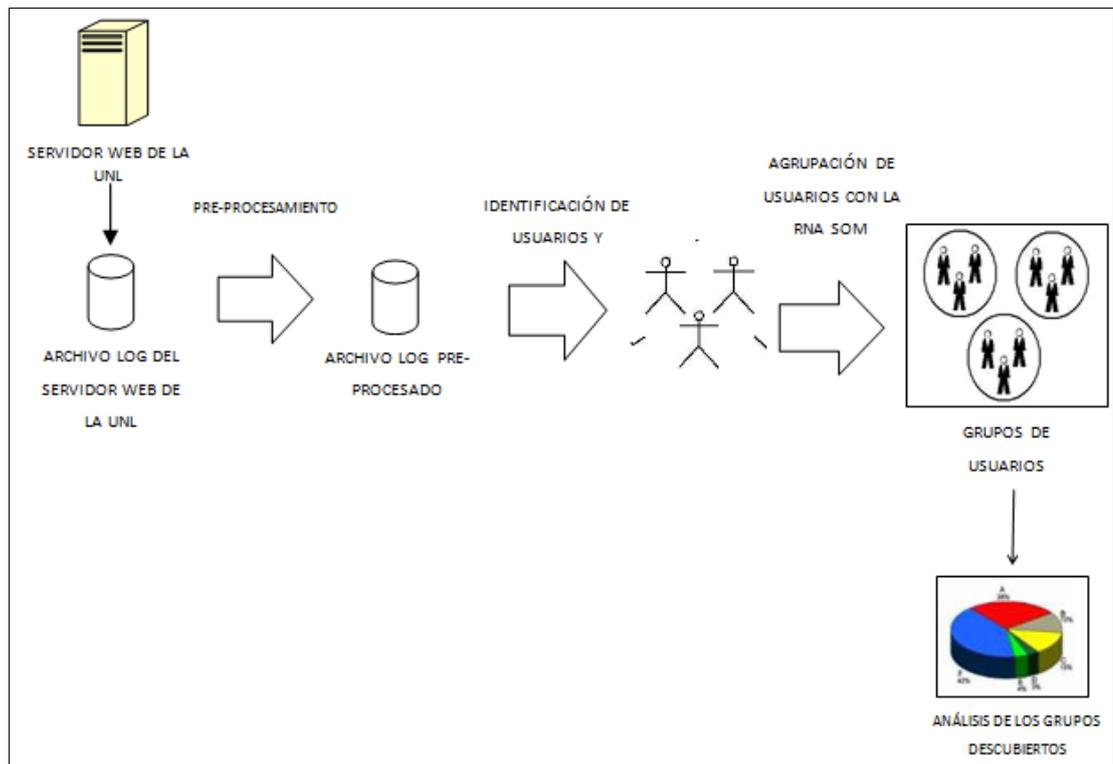


Figura 141: Dinámica del Banco de Pruebas

Se extrae el archivo log del servidor web del sitio a analizar, realizando su pre-procesamiento y la posterior identificación de usuarios y sus sesiones. El agrupamiento de los usuarios es realizado utilizando la Red Neuronal SOM, permitiendo el análisis de los grupos descubiertos.

6.5.4. Resultados obtenidos

Aquí se exponen los resultados obtenidos de los análisis realizados. Se analizaron los archivos log del sitio web de la Universidad Nacional de Loja, por ende para este sitio se realizaron todos los pasos involucrados en la identificación de las preferencias de



los usuarios.

6.5.4.1. Análisis de los archivos log del sitio web de la Universidad Nacional de Loja

6.5.4.1.1. Descripción del sitio web de la UNL

El sitio web de la Universidad Nacional de Loja es de carácter educativo y permite el acceso a información general de la Universidad, de las diferentes áreas, carreras, asociaciones, autoridades y estructura de la misma. Así como también brinda servicios de información de admisiones, SGA Preparación Académica Básica, SGA Estudiantes, SGA Profesores, servicios en línea como radio universitaria, sistema bibliotecario, correo electrónico, estudios de modalidad a distancia, editorial universitaria, cursos de regionalización, centro de recursos Idrisi, Proyecto Micambio e Investigación – Cisaq, entre otros.

6.5.4.2. Páginas del sitio web de la Universidad Nacional de Loja

A continuación se detalla la lista de páginas que componen al sitio:

- **index.php:** Información general de la Universidad Nacional de Loja, servicios y novedades relacionadas con la misma.
- **quienes-somos.html:** Información general acerca de los servicios que ofrece la Universidad, la letra de su himno y el escudo.
- **mensaje-del-rector.html:** Mensaje del Rector a los miembros de la Institución.
- **mision-vision.html:** Página que hace referencia al compromiso de la Universidad Nacional de Loja con la formación académica, profesional, institucional y social.
- **historia.html:** Página que informa sobre la creación de la Universidad Nacional de Loja y los distintos eventos acontecidos hasta la fecha.
- **directorio.php:** Listado con los datos de los integrantes del Directorio de la Universidad Nacional de Loja.
- **egresado.php:** Listado con los datos de los Egresados de la Universidad Nacional de Loja.
- **transparencia.html:** Ofrece los siguientes links: Estructura Orgánica



Funcional, Distributivo de personal, Escala de remuneraciones, Servicios, Contratos colectivos, Formatos para solicitudes, Presupuesto Anual, Liquidación de presupuesto, Auditorías internas y gubernamentales, Procesos precontractuales y contractuales, Incumplimiento de contratos, Planes y programas de la institución, Créditos internos y externos, Rendición de cuentas, Viáticos y movilizaciones, Responsable de la información y Consejo Académico Administrativo Superior (CAAS).

Todos los links indicados tienen la opción para descargar archivos en formato pdf.

- **bienestar/xmlrpc.php:** Página principal del Departamento de Bienestar Universitario que contiene los siguientes links: Presentación, Antecedentes, Misión, Visión, Líneas de Acción, Políticas, Objetivo, Servicios Universitarios, Servicio Psicopedagógico, Derechos y Obligaciones, Servicio de Becas, Beneficios, Servicio de Salud, Servicios Informáticos, Servicios Informáticos, Culturales y Bibliográficos para personas con capacidades especiales, ¿Quiénes somos? y Oferta Académica.
- **estadisticas.html:** Posee los siguientes links: Administrativos, Docentes, Trabajadores, Estudiantes, Jornadas de Estudiantes, Presupuesto y Resumen. Los links indicados tienen la opción para descargar archivos en formato xls y doc.
- **procesos-evaluacion.html:** Ofrece los siguientes links: Proyecto de Autoevaluación de Bienestar Estudiantil, Informe de Autoevaluación Institucional, Plan de Mejoras para el Departamento de Bienestar Estudiantil, Proyecto Aprobado de Evaluación de Autoridades y Directivos, Proyecto de Evaluación del IV Encuentro Nacional de Culturas, Informe de Evaluación del Desempeño de Autoridades y Directivos (Rector, Vicerector y Directores de Área, Representantes de Nivel, Coordinadores de Carrera), Proyecto de Evaluación del desempeño Docente del Curso Preuniversitario, Evaluación del Desempeño Docente e Informe de Autoevaluación del Departamento de Bienestar de Estudiantil Universitario (2006).
- **sistema-bibliotecario.html:** Página que muestra una breve descripción del Sistema Bibliotecario de la Universidad Nacional de Loja.
- **acerca-sistema-bibliotecario-unl.html:** En esta página se describen las siguientes áreas: Quiénes Somos, Estructura, Filosofía, Campo de Acción, Instructivo e Información de Contacto.



- **biblioteca-virtual.html:** Describe el servicio de la biblioteca virtual de la Universidad Nacional de Loja a través de los siguientes ítems: Catálogo Virtual del Sistema Bibliotecario, Bases de Datos Científicas, Enlaces a Bibliotecas Virtuales y Sitios de Interés y Aula Virtual.
- **webmail/src/login.php:** Página de ingreso al correo electrónico para los usuarios del WebMail de la Universidad Nacional de Loja.
- **editorial/xmlrpc.php:** Página principal de la Editorial Universitaria que contiene los siguientes links: ¿Quiénes Somos? y Presentación.
- **miccambio/xmlrpc.php:** Página principal del Proyecto Miccambio que contiene los siguientes links: ¿Quiénes Somos?, Visión y Misión, Objetivos y Contáctanos.
- **cisaq/xmlrpc.php:** Página principal de Investigación - Cisaq que contiene los siguientes links: Presentación, Misión y Visión, Servicios, Espacios-Físicos, Políticas de Calidad, Manual de Gestión de Calidad y LABORATORIOS ANEXOS AL CISAQ-UNL.
- **component/banners/clic/6.html:** Página relacionada con la Vinculación con la Colectividad.
- **component/banners/clic/2.html:** Página relacionada con la Secretaría Nacional de Ciencia y Tecnología.
- **component/banners/clic/9.html:** Página relacionada con el Campus Virtual de la Universidad Nacional de Loja.
- **component/banners/clic/30.html:** Página relacionada con el Sistema Bibliotecario de la Universidad Nacional de Loja.
- **component/banners/clic/36.html:** Página relacionada con el Centro de Investigación y Servicios de Análisis Químicos CISAQ-UNL.
- **component/banners/clic/18.html:** Página relacionada con una red de Universidades.
- **component/banners/clic/41.html:** Página relacionada con el Proyecto LOES para la discusión.
- **component/banners/clic/33.html:** Página relacionada con la UNAMAZ Asociación de Universidades Amazónicas.
- **component/banners/clic/1.html:** Página relacionada con el Proyecto WAFLA.
- **component/banners/clic/28.html:** Página relacionada con el DFG de la Universidad Nacional de Loja.
- **component/banners/clic/15.html:** Página relacionada con el Proyecto



Binacional “Integración Regional para el Manejo Ambiental Sostenible y Control de la Desertificación de Ecuador y Perú”.

- **component/banners/clic/31.html**: Página relacionada con la Corporación Catamayo Centro Binacional Ecuador – Perú.
- **component/banners/clic/4.html**: Página relacionada a la UnINet.
- **component/banners/clic/5.html**: Página relacionada a Idrisi Ecuador - CINFA-UNL, Sitio del Centro de Recursos Idrisi en Ecuador, con sede en el Centro Integrado de Geomática Ambiental de la Universidad Nacional de Loja.
- **component/banners/clic/19.html**: Página relacionada al Quinto Plan de Desarrollo Institucional 2008 – 2013.
- **component/banners/clic/16.html**: Página relacionada con la Revista Universitaria.
- **component/banners/clic/35.html**: Página relacionada con el Consejo Nacional de Evaluación y Acreditación de la Educación Superior del Ecuador.
- **component/banners/clic/42.html**: Página que muestra el Cronograma para la Admisión al Primer Módulo de las Carreras de la Universidad Nacional de Loja Periodo 2010-2011.
- **component/banners/clic/37.html**: Página relacionada con el Sistema de Gestión Académico UNL.
- **component/banners/clic/10.html**: Página de ingreso al Sistema de Gestión Académica de los estudiantes de la Universidad Nacional de Loja.
- **component/banners/clic/11.html**: Página de ingreso al Sistema de Gestión Académica de los docentes de la Universidad Nacional de Loja.
- **component/banners/clic/29.html**: Página relacionada con la Evaluación del Desempeño Docente de la Universidad Nacional de Loja.
- **component/banners/clic/40.html**: Página que muestra información relacionada con la Maestría en Administración Educativa Modalidad Semipresencial.
- **component/banners/clic/32.html**: Página que informa sobre el Proyecto de Mejoramiento de las Capacidades Científico-Técnicas de los Docentes de la Universidad Nacional de Loja.
- **component/banners/clic/38.html**: Página relacionada con el Sistema de Gestión Académico UNL.
- **component/banners/clic/17.html**: Página que muestra la Postal de la Universidad Nacional de Loja.



- **educativa/xmlrpc.php:** Página que muestra toda la información relacionada con el Área de Educación, el Arte y la Comunicación de la Universidad Nacional de Loja.
- **juridica/xmlrpc.php:** Página que muestra toda la información relacionada con el Área Jurídica, Social y Administrativa de la Universidad Nacional de Loja.
- **agropecuaria/xmlrpc.php:** Página que muestra toda la información relacionada con el Área Agropecuaria y de Recursos Naturales Renovables de la Universidad Nacional de Loja.
- **salud/xmlrpc.php:** Página que muestra toda la información relacionada con el Área de la Salud Humana de la Universidad Nacional de Loja.
- **apul/PaginaPrincipal.htm:** Información general sobre APUL (Asociación de Profesores Universitarios).
- **apul/QuienesSomos.htm:** Información sobre los integrantes del Directorio del APUL.
- **apul/Proyecto%20Vivienda.htm:** Fotos y estadísticas del avance de las obras del Programa de Vivienda II Etapa.
- **apul/Noticias.htm:** Noticias relacionadas con el APUL en las distintas Áreas de la Universidad Nacional de Loja.
- **apul/Convenios.htm:** Describe los convenios a los cuales que pueden acogerse los socios de APUL.
- **apul/Eventos.htm:** Descripción de los eventos que realizarán los docentes universitarios.
- **apul/Contactos.htm:** Información sobre los medios para comunicarse con el APUL.
- **apul/Docentes.htm:** Links para acceder a los datos de los docentes de las diferentes Áreas de la Universidad Nacional de Loja.
- **apul/Internet.htm:** Descripción de los servicios de Internet que pueden adquirir los docentes universitarios.
- **apul/IESS.htm:** Acceso a los servicios del IESS.
- **ageul/xmlrpc.php:** Página que muestra toda la información relacionada con la Asociación General de Empleados Universitarios de la Universidad Nacional de Loja.
- **plan-gedes.html:** Presentación del PLAN GEDES 2008 – 2013.
- **autoridades.html:** Datos del Rector y Vicerector de la Universidad Nacional de Loja.



- **junta-universitaria.html:** Datos del Rector, Vicerector, Representantes de la Junta Universitaria y Representantes Estudiantiles de la Junta Universitaria.
- **representantes-universidad-nacional-de-loja.html:** Nombres de los Representantes de Empleados y Trabajadores de la Universidad Nacional de Loja.
- **consejo-academico-administrativo-superior.html:** Nombres de los Representantes del Consejo Académico Administrativo Superior y Representantes del Consejo Estudiantil Académico Administrativo Superior.
- **directores-de-las-areas.html:** Datos de los Directores de cada una de las Áreas de la Universidad Nacional de Loja.
- **estructura.html:** Describe el Sistema Modular, los Niveles de Formación y cada uno de los Elementos Académicos del Sistema Modular.
- **mapa-sitio.htm:** Muestra todas las páginas que conforman el sitio web de la Universidad Nacional de Loja.

6.5.4.2.1. Análisis del log 09/01/2011 al 13/01/2011 del sitio web de la UNL

A continuación el análisis del archivo log:

- **Registros en el archivo de log original**

916462 Registros (231 MB).

- **Formato log**

CLF+REFERRER+AGENT.

- **Carga log**

A continuación se muestra los valores utilizados para la carga y pre-procesamiento en el sistema del archivo log.



Cargar y Preprocesar

Archivo Log

Path del archivo: *

Formato:

Filtros para los registros del archivo log:

Extensiones de paginas: htm html php
 pdf xls doc
 gif jpg png
 asp

Código de estado: 200

Variables ID de página: GET POST

Al cargar y preprocesar los registros del archivo log usted estara filtrando los mismos por diferentes criterios. La aplicacion creara internamente un archivo limpio y filtrado con los filtros que usted ha seleccionado. Dependiendo del analisis que usted desee obtener puede indicar otros criterios diferentes a los que se encuentran por defecto.

Figura 142. Carga y Pre-procesamiento del archivo log

Tiempo de Carga y Pre-procesamiento del log: 22 minutos.

A continuación se muestra el resultado de la carga y pre-procesamiento del archivo log:

Carga de Registros del Archivo Log: logs9-13ene_filtrado.txt

Registros guardados: 487127

Páginas guardadas: 48

Path del Log: C:\MiAplicacionWeb\Usuarios\ldanirobles\Proy-9-13Enero\

Extensiones: htm,html,php,gif,jpg,pngcss,js

Código de estado: 200

Id's de Páginas: GET

Figura 143. Resultado Carga y Pre-procesamiento del archivo log



Tiempo de Construcción de Sesiones: 1 minuto 15 segundos.

Construcción de sesiones con valores por defecto (Cantidad mínima de páginas en una sesión = 3, Tiempo máximo de sesión = 30 minutos, Frecuencia mínima de páginas en el log = 3).

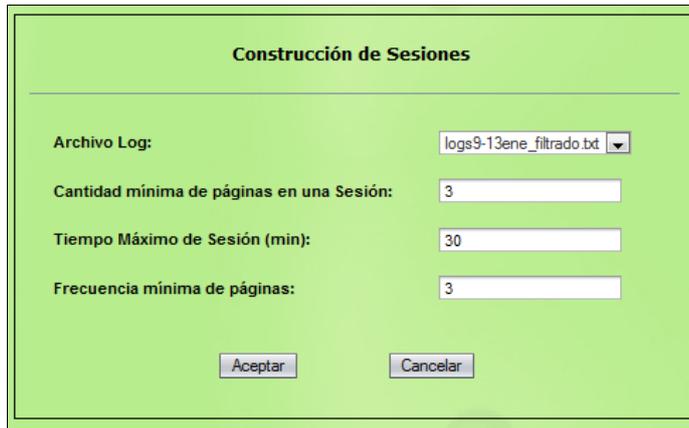


Figura 144. Construcción de sesiones del archivo log

A continuación se muestra el **Resumen del Proyecto** luego de la carga y pre-procesamiento del log y de la construcción de las sesiones de usuario.



Archivos Log cargados:	1
Registros Log Cargados:	487127
Sesiones Cargadas:	71548

Figura 145. Resumen del Proyecto



• Clusterización con SOM

Clusterización SOM o construcción de Grupos de Usuarios con los valores por defecto para proceder a clusterizar:

- Cantidad de ciclos de entrenamiento: 10000
- Cardinalidad del mapa de clusterización (NxN): 2

Construir Clústers o Grupos de usuarios

Agrupar los usuarios mediante la Red Neuronal Artificial SOM

Archivo Log: logs913ene_filtrado.txt

Cantidad de ciclos de entrenamiento: 10000

Cardinalidad del mapa de clusterización (NxN): 2

Los valores que se muestran son considerados por defecto para la construcción de los Grupos de usuarios mediante la Red Neuronal Artificial (clusterización). Es recomendable establecer en la Cardinalidad del mapa de clusterización una dimensión de 2 para no obtener datos muy dispersos. Si usted introduce otros valores es preciso que conozca que resultados desea obtener con el análisis de este log.

Aceptar Cancelar

Figura 146. Agrupar usuarios con la Red Neuronal Artificial SOM

Tiempo Clusterización con SOM: 4 minutos 20 segundos.

A continuación se muestra la cantidad de sesiones en cada clúster y el porcentaje correspondiente a cada uno:



Archivo Log: logs9-13ene_filtrado.txt

Grupo de usuarios	Sesiones
0	1 295
1	39916
2	20 399
3	9938

Figura 147. Cantidad de sesiones en cada clúster

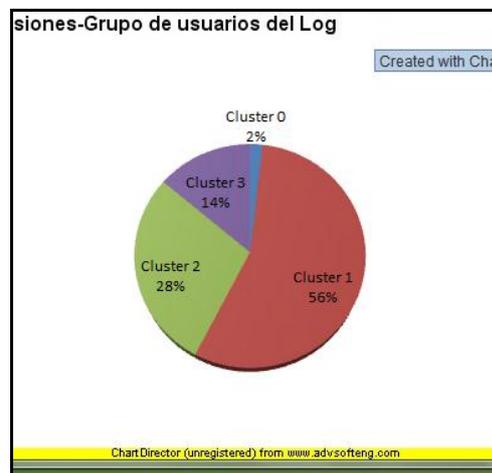


Figura 148. Porcentaje de sesiones en cada clúster con SOM

• **Representación de los clústers mediante las páginas de sus sesiones**

A continuación se muestran las páginas que accedieron los usuarios de cada clúster:



Páginas en cada Clúster

Archivo Log: Clúster: [Mostrar grafico](#)

Página	Extensión	Frecuencia	Url de la pagina
/index.php	php	75	http://www.unl.edu.ec/index.php
/sistema-bibliotecario/index.php /sistema-bibliotecario/biblioteca-virtual.html	html	4	http://www.unl.edu.ec/sistema-bibliotecario/index.php/sistema-bibliotecario/biblioteca-virtual.html
/plan-gedes.html	html	8	http://www.unl.edu.ec/plan-gedes.html
/salud/wp-admin/salud/wp-admin /load-scripts.php	php	2	http://www.unl.edu.ec/salud/wp-admin/salud/wp-admin/load-scripts.php
/salud/wp-admin/salud/wp-admin /index-extra.php	php	4	http://www.unl.edu.ec/salud/wp-admin/salud/wp-admin/load-scripts.php
/historia.html	html	2	http://www.unl.edu.ec/salud/wp-admin/salud/wp-admin/index-extra.php
http://unl.edu.ec/component/content /article/1-noticias/287-calificacion-de-los-movimientos-participantes.html	html	3	http://www.unl.edu.ec/salud/wp-admin/salud/wp-admin/index-extra.php

Page: 1 of 6

Figura 149. Páginas en un clúster

• **Análisis de los clústers de usuarios descubiertos**

- **Clúster 0:** Usuarios que en su mayoría sólo acceden a la página principal (index.php).
- **Clúster 1:** Usuarios que acceden a la información y pensum de estudios de las carreras, al proceso de evaluación de estudiantes para el ingreso a la UNL e información de los servicios que presta el departamento de Bienestar Estudiantil.
- **Clúster 2:** Usuarios que acceden al sistema e inician una sesión, acceden al sistema bibliotecario y otros links que ofrece el sitio pero no están dentro del mismo.
- **Clúster 3:** Usuarios que recorren algunas páginas del sitio, principalmente información general de la universidad, misión, visión, noticias de eventos, realizan búsquedas y descargan algunos documentos.



6.5.5. Valoración

Con todos estos antecedentes las pruebas se llevaron a cabo el día 13 de enero del 2011; para la realización de dichas pruebas primeramente se instruyó a 5 usuarios (estudiantes de la Carrera de Ingeniería en Sistemas) en el manejo de la aplicación y luego se aplicó una encuesta (ver Anexo 1).

Durante las pruebas con los usuarios no se presentaron errores de ejecución comprobándose así su correcto funcionamiento y la exactitud de los procesos involucrados.

El día 20 de enero del 2011 se demostró el funcionamiento de la aplicación desarrollada al Ing. Patricio Valarezo, Responsable de la sección de desarrollo de software de la UNL comprobándose que todos los procesos se realizan con exactitud; además se le aplicó una entrevista (ver Anexo2).

El Ing. Patricio Valarezo nos dio algunas sugerencias, las cuales tomamos en cuenta para mejorar la seguridad de la aplicación en el momento de registrar un usuario.

Finalmente se pudo demostrar que la **“Herramienta para el descubrimiento de preferencias de uso de los cibernautas del sitio web de la Universidad Nacional de Loja aplicando Redes Neuronales Artificiales SOM”** cumple con todos los requerimientos especificados en la etapa del análisis y diseño del sistema.

A continuación se presentan los resultados de la tabulación de las encuestas aplicadas a 5 usuarios de la aplicación.

Los indicadores de evaluación son los siguientes: MS (Muy Satisfactorio), S (Satisfactorio), A (Aceptable) y R (Regular).



	FUNCIONALIDAD	MS	S	A	R
1	Manipulación y acceso de los links (Inicio, ¿Quiénes Somos?, Ayuda y Acerca de.	3	2	0	0
2	Proceso de Registro del Usuario	3	2	0	0
3	Manipulación y acceso del menú principal del usuario	2	2	1	0
4	Administración de los usuarios	3	1	1	0
5	Administración de los proyectos	3	2	0	0
6	Manipulación y acceso del menú principal de un proyecto	4	1	0	0
7	Procesos de subir, cargar y pre-procesar un archivo log	4	1	0	0
8	Construcción de sesiones de usuario de un archivo log filtrado	1	3	1	0
9	Agrupación de los usuarios utilizando la Red Neuronal Artificial SOM (Mapa Auto-organizativo)	2	3	0	0
10	Representación de las sesiones de usuario en cada clúster (Grupo de usuario)	3	2	0	0
11	Resultados obtenidos de las páginas más visitadas por los usuarios de cada clúster en el sitio web de la UNL	3	2	0	0
12	Almacenamiento de la información en la base de datos	2	2	1	0
TOTAL		33	23	4	0

Figura 150. Tabulación de las encuestas

Los resultados de las pruebas realizadas son las siguientes:

INDICADOR	VALOR
MS (Muy Satisfactorio)	33
S (Satisfactorio)	23
A (Aceptable)	4
R (Regular)	0
TOTAL	60

Figura 151. Resultados de las pruebas



Representando estos valores en porcentaje se tiene:

INDICADOR	PORCENTAJE
MS (Muy Satisfactorio)	55%
S (Satisfactorio)	38%
A (Aceptable)	7%
R (Regular)	0%
TOTAL	100%

Figura 152. Representación en porcentaje de los resultados de las pruebas

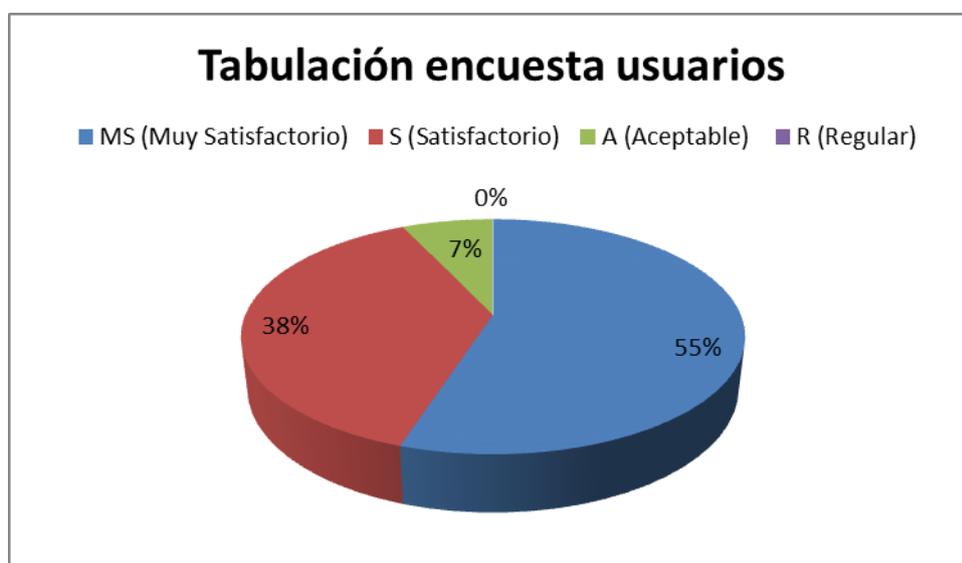


Figura 153. Representación gráfica de la tabulación de las encuestas

A través de la representación se puede evidenciar que el 55% de los usuarios consideran que los procesos involucrados en la aplicación son Muy Satisfactorios, el 38% que son Satisfactorios, el 7% que son Aceptables.



7. VALORACIÓN TÉCNICA-ECONÓMICA

Al finalizar el presente proyecto investigativo se ha podido comprobar la eficiencia del funcionamiento de la aplicación para descubrir las preferencias de uso de los cibernautas del sitio web de la Universidad Nacional de Loja, cumpliendo con todos los requerimientos especificados en las etapas de análisis y diseño.

Los costos aproximados asumidos para la construcción de la aplicación web se muestran a continuación:

Recursos Humanos

Descripción	Costo
Asesoría	\$ 250,00
Subtotal 1	\$ 250,00

Recursos Técnicos

Descripción	Costo
Computador Intel Pentium IV 2.8 Ghz, 512 de RAM, HD 160 Gigas	\$ 1.200,00
Impresora HP all in one 1410	\$ 60,00
Flash memory de 2 GB	\$ 30,00
Internet	\$ 720,00
Subtotal 2	\$ 2.010,00

Recursos Materiales

Descripción	Costo
Materiales de oficina	\$ 25,00
Cartuchos de tinta	\$ 80,00
Papel bond	\$ 16,00
Discos ópticos	\$ 10,00
Copias	\$ 5,00



Anillados	\$ 30,00
Empastados	\$ 80,00
Transporte	\$ 180,00
Servicios básicos (luz, agua, teléfono)	\$ 500,00
Subtotal 3	\$ 926,00
Subtotal de Recursos	\$ 3.186,00
Trámites de tesis	\$ 200,00
TOTAL	\$ 3.386,00

Figura 154. Detalle de costos



8. CONCLUSIONES

Al finalizar el presente proyecto se ha llegado a las siguientes conclusiones:

- Se logró el desarrollo de la **“Herramienta para el descubrimiento de preferencias de uso de los cibernautas del sitio web de la Universidad Nacional de Loja aplicando Redes Neuronales Artificiales SOM”**, la cual puede ser accedida desde una interfaz web, de modo que el usuario pueda utilizarla directamente desde su navegador favorito.
- La aplicación desarrollada emplea interfaces gráficas simples y amigables de modo que usuarios sin experiencia en el uso de herramientas de este tipo pueden trabajar fácilmente.
- La metodología ICONIX por ser reiterativa e incremental permitió corregir errores de las etapas iniciales del desarrollo de la aplicación sin afectar las etapas actuales.
- La utilización de StarUML facilitó el diseño de los diagramas de casos de uso, diagramas de secuencia, diagramas de clases y diagramas de paquetes debido a que es una herramienta totalmente sustentada en UML.
- La herramienta se codificó utilizando el IDE NetBeans 6.5 el cual es un producto de código abierto y gratuito sin restricciones de uso, escrito completamente en Java.
- La utilización del framework JSF facilitó el desarrollo de la aplicación web ya que al estar basado en el patrón de arquitectura Modelo-Vista-Controlador (MVC) se logró separar los datos de la interfaz de usuario y la lógica de control.
- Se acopló un método de seguridad para que los datos de los usuarios de la aplicación sean confidenciales; esto se logró mediante el código de encriptación MD5.
- Hibernate nos permitió mapear las clases y las relaciones entre ellas a través



de archivos XML. Además facilitó el manejo de consultas Java a la base de datos.

- Se aplicó la técnica de minería de datos Web Usage Mining (WUM) para extraer la información de los archivos log como: páginas web, direcciones IP, referencias a páginas, fecha y hora de acceso, cantidad de bytes transmitidos, protocolo utilizado, códigos de respuesta de las páginas solicitadas entre otros, lo que nos permitió entender el comportamiento de los usuarios del sitio web de la Universidad Nacional de Loja.
- Con la implementación de la librería `joone-engine.jar` se logró construir la Red Neuronal Artificial SOM, la misma que permitió ingresar los patrones obtenidos en el archivo de entrada que está conformado por unos y ceros que representan la presencia o ausencia de una página en una determinada sesión de usuario, procesar dichos patrones y descubrir los grupos de usuarios y las páginas más visitadas por los usuarios del sitio web de la Universidad Nacional de Loja.
- Mediante la utilización de librería `ChartDirector.jar` se logró representar gráficamente los resultados obtenidos por la RNA.
- Se logró demostrar el análisis de los archivos log almacenados en el servidor web de la Universidad Nacional de Loja a través de cada uno de los procesos involucrados en la aplicación desarrollada.
- Los resultados obtenidos de las encuestas aplicadas a 5 usuarios de la aplicación y al Ing. Patricio Valarezo (Responsable de la sección de desarrollo de software de la UNL) demuestran su aceptación por la herramienta desarrollada.
- La realización de este proyecto nos ha permitido adquirir nuevos conocimientos en cuanto al tratamiento de las Redes Neuronales Artificiales y a su aplicabilidad en las distintas áreas de la ciencia particularmente en la Informática.



9. RECOMENDACIONES

- A los futuros tesisistas, obtener por escrito el apoyo para la realización del proyecto de tesis a las diferentes autoridades de la Universidad y/o empresa en la que se desarrolle el proyecto para que se puedan cumplir todos los objetivos planificados en el proyecto.
- A las autoridades de la Universidad Nacional de Loja, otorgar los permisos necesarios para realizar las pruebas de las aplicaciones desarrolladas.
- Al Administrador del sitio web de la Universidad Nacional de Loja, implementar la herramienta desarrollada para descubrir las preferencias de los usuarios y a futuro acoplar el sitio de acuerdo a sus necesidades.
- A los estudiantes y futuros tesisistas, investigar continuamente sobre nuevas herramientas para el desarrollo de aplicaciones web.
- Investigar la factibilidad de la adaptación de otros tipos de Redes Neuronales Artificiales en la identificación de preferencias de uso de sitios web.
- Implementar nuevas funcionalidades a la herramienta desarrollada para mejorar los procesos de conexión y carga de información en la base de datos.
- Utilizar la solución propuesta como base para realizar nuevos proyectos orientados a descubrir las preferencias de los usuarios en aplicaciones de escritorio o motores de base de datos.



10. BIBLIOGRAFÍA

Libros:

1. A. K. JAIN, M.N. MURTY and P. J. FLYNN. Data clustering: a review. ACM Computing Surveys, 31(3):264–323, 1999.
2. BAEZA-YATES, Ricardo. RIBEIRO-NETO, Berthier. (2000), Modern Information Retrieval, Addison-Wesley.
3. BAEZA-YATES, Ricardo. RIBEIRO-NETO, Berthier. Modern Information Retrieval. Addison Wesley. ISBN 0-201-39829-X. ACM. New York. 1999.
4. BROWN, C.M. DANZING, P.B. HARDY, D. MAMBER, U. y SCHWARTZ, F. (1994), The Harvest information discovery and access system, in Proceedings of the Second International World Wide Web Conference, pp. 763-771.
5. COOLEY, Robert. TAN, Pang-Ning. SRIVASTAVA, Jaideep. "Discovery of interesting usage patterns from web data".
6. COOLEY, Robert. TAN, Pang-Ning. SRIVASTAVA, Jaideep. "Discovery of interesting usage patterns from web data", M. Spiliopoulou and B. Masand editors, Advances in web usage analysis and user profiling, Lecture Notes in computer science, ISBN: 3-540-67818-2, pp. 163-182, 2000.
7. GUTIERREZ, M.A. Métodos y Técnicas de Investigación, Texto guía para el estudiante, Quito, Editorial Época, 1984.
8. HAMMOND, K. BURKE, R. MARTIN, C. LYTINEN, S. (1995), FAQ finder: A case - based approach to knowledge navigation, in Working Notes of the AAAI Spring Symposium: Information Gathering from Heterogeneous, Distributed Environment, pp. 69-73, Stanford University, AA AI Press.
9. HIBERA, J.R. MARTINEZ, V.J. "Redes Neuronales Artificiales", AlfaOmega, 2000.
10. HUNTER, James. "Central Logging Security", 25 November 2000.
11. IA2007-Grupo01-TP. Redes de Kohonen.doc. Arquitectura
12. J, HAN and M, KAMBER. Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers Inc, 2000.
13. K, LAGUS. J, HONKELA. S, KASKI. T, KOHONEN. Self-Organizing Maps of Document Collections: A New Approach to Interactive Exploration. En



Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (1996).

14. Lineamientos para el desarrollo de las investigaciones de tesis de grado en la Carrera de Ingeniería en Sistemas, Universidad Nacional de Loja, 2006.
15. MORTON, Matt. "Logging and critical logs files: the Decision to effectively and proactively Manage System logging and Log Files", 9 December 2000.
16. P, BATISTA and M. J. SILVA. Mining on-line newspaper web access logs, 12th International Meeting of the Euro Working Group on Decision Support Systems (EWG-DSS 2001), 2001.
17. SANKAR K, Pal. VARUN, Talwar. PABITRA, M. "Web Mining in Soft Computing Framework".
18. SRIVASTAVA, J. COOLEY, R. DESHPANDE, M. TAN, P-N. Web Usage Mining.
19. T, KOHONEN: Self-Organizing Maps. *Springer-Verlag* (2001)
20. VILLENA, J. GONZÁLEZ. J. C, BARCELO, E. & VELASCO. J. R. Minería de uso de la Web mediante huellas y sesiones. IBERAMIA, 2002.

Sitios web:

1. <http://www.cert.org/securityimprovement/practices/p092.html>
2. <http://monografias.com/redneur>
3. <http://ohm.utp.edu.co/neuronales/Capitulo2/Competitivas/>
4. <http://personales.upv.es/ccarrasc/doc/2001/RedesNeuronalesExcaliburNativNoverges/redes.htm>
5. <http://technet.microsoft.com/es-es/library/ms174949.aspx>



11. ANEXOS

