



Universidad
Carlos III de Madrid

Ingeniería informática

PROYECTO FIN DE CARRERA

ANÁLISIS Y DISEÑO DE UNA PLATAFORMA WEB PARA UN SISTEMA DE GESTIÓN DE USUARIOS

Autor: Vanesa Moreno Boiza

Tutor: Luis Miguel Sánchez García

Leganés, octubre de 2012

Título: ANÁLISIS Y DISEÑO DE UNA PLATAFORMA WEB PARA UN SISTEMA DE GESTIÓN DE USUARIOS

Autor:

Director:

EL TRIBUNAL

Presidente: _____

Vocal: _____

Secretario: _____

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día ____ de ____ de 20____ en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE

*“Trabajar no es realizar lo que uno imaginaba,
sino descubrir lo que uno tiene dentro.”*

Boris Pasternak

Agradecimientos

A mis padres Carlos y Marisa.
A quien les debo todo.

A David.
Por su larga espera y paciencia.

A mis amigos y compañeros.
Por el interés, apoyo y ayuda recibidos.

Resumen

El presente proyecto denominado *Análisis y diseño de una plataforma Web para un sistema de gestión de usuarios*, consiste en una plataforma Web que implementa un sistema de gestión de usuarios (SUM - System Users Management) y se encuentra integrada en un sistema global corporativo para la gestión de expedientes de telecomunicaciones denominado GEXTEL.

La plataforma desarrollada tiene como objetivo gestionar los diferentes usuarios y permisos de acceso de los usuarios al sistema de gestión de expedientes de telecomunicaciones organizados de forma departamental.

Los usuarios, en función de los permisos de acceso que posean, podrán consultar, crear, eliminar o tramitar expedientes así como acceder a determinado tipo de expedientes, procesos, procedimientos o áreas.

Cada departamento dispondrá de al menos un usuario con permiso de administración de usuarios que le permitirá mantener los usuarios y los permisos de acceso de los usuarios de su departamento haciendo uso del sistema SUM.

Los siguientes módulos componen el sistema SUM:

- **SUM Web:** módulo Web principal que permite realizar diferentes acciones de configuración de usuarios y permisos de acceso de los mismos.

Acciones on line: bloquear y desbloquear el acceso al sistema de un usuario, modificar contraseña de usuario, asignar permisos de portafirmas, consultar permisos de usuario.

Acciones diferidas: crear usuarios, eliminar usuarios y asignar perfiles de acceso específicos a usuarios.

- **SUM Server:** módulo que realiza las operaciones de configuración de permisos de usuario de acciones diferidas solicitadas desde el módulo SUM Web (crear usuarios, eliminar usuarios y asignar perfiles de acceso específicos a usuarios).

Palabras clave: Gestión, usuarios, sistema, módulo, permisos, roles, perfiles, departamentos, SUM, SUM Web, SUM Server, GEXTEL.

Abstract

The present project, titled *Analysis and design of a Web platform for a system users management* consists on a Web platform that implements a system users management (SUM) and is fully integrated in GEXTEL, a global corporative system for managing telecommunications files.

The platform has been developed in order to manage users and users' access permissions to the telecommunications files management system which is organized in different departments.

Users, depending on access permissions, will be able to consult, create, delete or deal with files along with having access to certain type of expedients, procedures or areas.

In each department there will be at least one user with a user administration permission that will allow him to maintain users and access permissions for users in his department through the SUM system.

The SUM System is based on the following modules:

- **SUM Web:** main Web module that allows several users configuration actions to be performed and subsequent access permits for users.

Online actions: block and unblock user's access to the system, modify user passwords, check user's permissions.

Deferred Actions: create users, delete user accounts and assign specific access profiles to users.
- **SUM Server:** this module performs user's permissions configuration operations from deferred actions that are requested through the SUM web module (creating and deleting users and assigning specific access profiles to users).

Key Words: Management, users, system, module, permits, roles, profiles, departments, SUM, SUM Web, SUM Server, GEXTEL.

Índice general

CAPÍTULO 1. INTRODUCCIÓN.....	24
1.1 MARCO DEL PROYECTO.....	25
1.2 OBJETIVOS GENERALES.....	26
1.3 ESTRUCTURA DEL DOCUMENTO.....	27
CAPÍTULO 2. ESTADO DE LA CUESTIÓN	28
2.1 ARQUITECTURA WEB	29
2.1.1 Introducción.....	29
2.1.2 Descripción arquitectura Web	30
2.1.3 Aspectos de seguridad en Web.....	34
2.1.4 Ventajas \ desventajas de aplicaciones Web	35
2.1.5 Patrón de diseño MVC	37
2.1.6 Formularios Web	40
2.2 CRIPTOGRAFÍA: FUNCIÓN RESUMEN	43
2.3 TECNOLOGÍAS UTILIZADAS	45
2.3.1 Microsoft .NET	46
2.3.2 Microsoft SQL Server	50
2.3.3 Microsoft Visual Studio	51
2.3.4 Visual Basic.NET	51
2.3.5 JavaScript	51
2.3.6 HTML.....	52
2.4 ESTÁNDAR ESA	52
CAPÍTULO 3. ORGANIZACIÓN DEL PROYECTO.....	54
3.1 ROLES ORGANIZATIVOS	55
3.2 PLANIFICACIÓN.....	57
CAPÍTULO 4. ANÁLISIS Y DISEÑO	60
4.1 REQUISITOS DE USUARIO.....	61
4.1.1 Requisitos de capacidad	61
4.1.2 Requisitos de restricción.....	63
4.2 REQUISITOS SOFTWARE	64
4.2.1 Requisitos funcionales.....	64

4.2.3 Requisitos no funcionales de interfaz.....	68
4.2.4 Requisitos no funcionales de operación	68
4.2.5 Requisitos no funcionales de recursos.....	69
4.2.8 Requisitos no funcionales de seguridad	69
4.3 MATRIZ DE TRAZABILIDAD UR/SR	70
4.4 MATRIZ DE TRAZABILIDAD SR/UC	74
4.5 ARQUITECTURA DEL SISTEMA	77
4.6 DESCRIPCIÓN DEL SISTEMA	78
4.6.1 Módulo SUM Web	80
4.6.1.1 Módulo SUM Web: diagrama de casos de uso.....	80
4.6.1.2 Módulo SUM Web: descripción de casos de uso	82
4.6.1.3 Módulo SUM Web: diagramas de secuencia	93
4.6.1.4 Módulo SUM Web: diagrama de clases	107
4.6.1.5 Módulo SUM Web: descripción de clases	108
4.6.1.6 Módulo SUM Web: diagramas de actividad	111
4.6.1.7 Módulo SUM Web: mapa Web	124
4.6.1.8 Módulo SUM Web: diseño de interfaces	125
4.6.2 Módulo SUM Server	130
4.6.2.1 Módulo SUM Server: diagrama de casos de uso.....	132
4.6.2.2 Módulo SUM Server: descripción de casos de uso	133
4.6.2.3 Módulo SUM Server: diagramas de secuencia	136
4.6.2.4 Módulo SUM Server: diagrama de clases	140
4.6.2.5 Módulo SUM Server: descripción de clases	141
4.6.2.6 Módulo SUM Server: diagramas de actividad	144
4.7 MODELO DE DATOS	148
4.7.1 Modelo E/R	148
4.7.2 Modelo Relacional.....	149
4.7.3 Descripción de jerarquías	151
4.7.4 Descripción de entidades del sistema	151
4.7.5. Descripción de entidades provenientes de campos multivaluados.....	160
4.7.6 Descripción de entidades provenientes de interrelaciones	162
4.7.7 Modelo físico de datos.....	164
CAPÍTULO 5. IMPLEMENTACIÓN.....	165
5.1 IMPLEMENTACIÓN DEL MÓDULO SUM Web.....	172

5.2 IMPLEMENTACIÓN DEL MÓDULO SUM Server.....	177
CAPÍTULO 6. VERIFICACIÓN.....	181
6.1 VERIFICACIÓN DEL SISTEMA SUM Web.....	182
6.2 VERIFICACIÓN DEL SISTEMA SUM Server.....	194
6.3 MATRIZ DE TRAZABILIDAD SR/PR.....	198
CAPÍTULO 7. CONCLUSIONES Y LÍNEAS FUTURAS.....	202
7.1 CONCLUSIONES.....	203
7.2 LÍNEAS FUTURAS.....	204
ANEXOS.....	206
ANEXO I. FORMATO XML DE DESCRIPCIÓN DE PERFILES.....	207
ANEXO II. INSTALACIÓN Y CONFIGURACIÓN DE LAS HERRAMIENTAS DE DESARROLLO	208
ANEXO III. CONFIGURACIÓN DEL SISTEMA SUM.....	228
ANEXO IV. MANUAL DE USUARIO	235
ANEXO V. PRESUPUESTO.....	258
ANEXO VI. GLOSARIO.....	260
ANEXO VII. REFERENCIAS.....	264

Índice de figuras

Figura 1. Arquitectura Web	30
Figura 2. Comunicación Web.....	31
Figura 3. Arquitectura tres capas.....	32
Figura 4. Arquitectura n capas.....	33
Figura 5. Arquitectura MVC	38
Figura 6. Estructura de archivos de <i>Web Forms</i>	41
Figura 7. Mensaje cifrado MD5	44
Figura 8. Funcionamiento MD5	45
Figura 9. Arquitecta de .Net <i>Framework</i>	46
Figura 10. Estructura interna del entorno de ejecución en lenguaje común	47
Figura 11. Diagrama básico de la biblioteca de clases de .Net <i>Framework</i>	48
Figura 12. Diagrama detallado de la biblioteca de clases de .Net <i>Framework</i>	48
Figura 13. Diagrama interno de un ensamblado .Net	50
Figura 15. Estructura organizativa.....	55
Figura 16. Planificación I	57
Figura 17. Planificación II	58
Figura 18. Planificación III.....	58
Figura 19. Planificación IV	59
Figura 20. Arquitectura sistema SUM.....	78
Figura 21. Sistema SUM	79
Figura 22. Módulo SUM Web: diagrama de casos de uso	81
Figura 23. Módulo SUM Web: diagrama de secuencia bloquear usuario	94
Figura 24. Módulo SUM Web: diagrama de secuencia desbloquear usuario	95
Figura 25. Módulo SUM Web: diagrama de secuencia modificar contraseña.....	96
Figura 26. Módulo SUM Web: diagrama de secuencia asignar permisos portafirmas ..	97
Figura 27. Módulo SUM Web: diagrama de secuencia consultar permisos usuario.....	98
Figura 28. Módulo SUM Web: diagrama de secuencia consultar perfil	99
Figura 29. Módulo SUM Web: diagrama de secuencia solicitar alta usuario	100
Figura 30. Módulo SUM Web: diagrama de secuencia solicitar baja usuario	101
Figura 31. Módulo SUM Web: diagrama de secuencia solicitar asignar perfil usuario	102

Figura 32. Módulo SUM Web: diagrama de secuencia consultar error petición	103
Figura 33. Módulo SUM Web: diagrama de secuencia eliminar petición	104
Figura 34. Módulo SUM Web: diagrama de secuencia solicitar acceso	105
Figura 35. Módulo SUM Web: diagrama de secuencia consultar datos usuario.....	106
Figura 36. Módulo SUM Web: diagrama de clases	107
Figura 37. Módulo SUM Web: diagrama de actividad bloquear usuario.....	112
Figura 38. Módulo SUM Web: diagrama de actividad desbloquear usuario	113
Figura 39. Módulo SUM Web: diagrama de actividad modificar contraseña.....	114
Figura 40. Módulo SUM Web: diagrama de actividad asignar permisos portafirmas .	115
Figura 41. Módulo SUM Web: diagrama de actividad consultar permisos usuario....	116
Figura 42. Módulo SUM Web: diagrama de actividad consultar perfil	117
Figura 43. Módulo SUM Web: diagrama de actividad solicitar alta usuario	118
Figura 44. Módulo SUM Web: diagrama de actividad solicitar baja usuario	119
Figura 45. Módulo SUM Web: diagrama de actividad solicitar asignar perfil usuario	120
Figura 46. Módulo SUM Web: diagrama de actividad consultar error petición	121
Figura 47. Módulo SUM Web: diagrama de actividad eliminar petición	122
Figura 48. Módulo SUM Web: diagrama de actividad solicitar acceso	123
Figura 49. Módulo SUM Web: diagrama de actividad consultar datos usuario	123
Figura 50. Módulo SUM Web: mapa Web	124
Figura 56. Menú principal gestión de usuarios.....	128
Figura 57. Menú principal gestión de usuarios.....	128
Figura 58. Menú principal gestión de usuarios.....	129
Figura 59. Menú principal gestión de usuarios.....	129
Figura 60. Módulo SUM Server: diagrama de casos de uso	132
Figura 61. Módulo SUM Server: diagrama de secuencia crear usuario	137
Figura 62. Módulo SUM Server: diagrama de secuencia eliminar usuario	138
Figura 63. Módulo SUM Server: diagrama de secuencia asignar perfil	139
Figura 64. Módulo SUM Server: diagrama de clases.....	140
Figura 65. Módulo SUM Server: diagrama de actividad crear usuario	145
Figura 66. Módulo SUM Server: diagrama de secuencia eliminar usuario	146
Figura 67. Módulo SUM Server: diagrama de secuencia asignar perfil	147
Figura 68. Modelo E/R	148
Figura 69. Jerarquía de usuarios	151
Figura 70. Entidad SYSTEMUSER	152

Figura 71. Entidad REMUSER	152
Figura 72. Entidad IUSER.....	153
Figura 73. Entidad DEPARTMENT	154
Figura 74. Entidad EXTINF	155
Figura 75. Entidad IGROUP	155
Figura 76. Entidad PROFILE	156
Figura 77. Entidad SIGNROLE	157
Figura 78. Entidad TASK.....	158
Figura 79. Entidad REQUEST	159
Figura 80. Entidad SIGNROLEVALUE.....	160
Figura 81. Entidad TASKVALUE	161
Figura 82. Entidad REQUESTVALUE.....	161
Figura 83. Entidad USERGROUP	162
Figura 84. Entidad SIGNPERM	163
Figura 85. Modelo físico de datos	164
Figura 86. Ficheros módulo SUM Web.....	166
Figura 87. Ficheros módulo SUM Server.....	166
Figura 88. Diagrama paquetes	167
Figura 89. Entornos de prueba.....	182
Figura 90. Instalación IIS: componentes Windows.....	208
Figura 91. Instalación MVS: inicio	209
Figura 92. Instalación MVS: asistente.....	210
Figura 93. Instalación MVS: página principal	210
Figura 94. Instalación MVS: características.....	211
Figura 95. Instalación MVS: opciones	212
Figura 96. Instalación MVS: componentes	212
Figura 97. Instalación MVS: fin	213
Figura 98. Instalación MVS: reiniciar	213
Figura 99. Instalación MVS: configuración	214
Figura 100. Instalación MVS: aplicación	214
Figura 101. Instalación MSQLS: licencia	215
Figura 102. Instalación MSQLS: requisitos previos	216
Figura 103. Instalación MSQLS: requisitos instalando.....	216
Figura 104. Instalación MSQLS: requisitos fin.....	217

Figura 105. Instalación MSQLS: configuración sistema I	217
Figura 106. Instalación MSQLS: configuración sistema II.....	218
Figura 107. Instalación MSQLS: información de registro	218
Figura 108. Instalación MSQLS: componentes.....	219
Figura 109. Instalación MSQLS: características I.....	219
Figura 110. Instalación MSQLS: características II	220
Figura 111. Instalación MSQLS: características III	220
Figura 112. Instalación MSQLS: espacio en disco	221
Figura 113. Instalación MSQLS: instancia	221
Figura 114. Instalación MSQLS: cuenta de servicio I.....	222
Figura 115. Instalación MSQLS: cuenta de servicio II	222
Figura 116. Instalación MSQLS: autenticación	223
Figura 117. Instalación MSQLS: intercalación	223
Figura 118. Instalación MSQLS: informe errores	224
Figura 119. Instalación MSQLS: instalación I	224
Figura 120. Instalación MSQLS: instalación II.....	225
Figura 121. Instalación MSQLS: progreso instalación I	225
Figura 122. Instalación MSQLS: progreso instalación II.....	226
Figura 123. Instalación MSQLS: progreso instalación III	226
Figura 124. Instalación MSQLS: fin instalación	227
Figura 125. Configuración SUM Web: directorio virtual	228
Figura 126. Configuración SUM Web: asistente	229
Figura 127. Configuración SUM Web: alias directorio	229
Figura 128. Configuración SUM Web: path directorio.....	230
Figura 129. Configuración SUM Web: permisos acceso directorio.....	230
Figura 130. Configuración SUM Web: fin directorio virtual	231
Figura 131. Configuración SUM Web: sitio Web predeterminado.....	231
Figura 132. Configuración SUM Web: propiedades directorio	232
Figura 133. Acceso GEXTEL	235
Figura 134. Acceso erróneo GEXTEL	235
Figura 135. Acceso aplicación gestión de usuarios.....	236
Figura 136. Acceso erróneo aplicación gestión de usuarios.....	236
Figura 137. Ventana inicio gestión de usuarios.....	237
Figura 138. Información de usuario	237

Figura 139. Lista de peticiones.....	238
Figura 140. Acciones con lista de peticiones	238
Figura 141. Mensaje confirmación eliminar petición.....	239
Figura 142. Mensaje informativo error tramitación	239
Figura 143. Inicio nuevo usuario.....	240
Figura 144. Nuevo usuario	240
Figura 145. Usuario no existente en dominio.....	241
Figura 146. Usuario reconocido en dominio	241
Figura 147. Usuario existe.....	242
Figura 148. Longitud de contraseña inválida	242
Figura 149. Aviso de ausencia de datos	243
Figura 150. Confirmación contraseña errónea	243
Figura 151. Alta de usuario correcta	244
Figura 152. Baja usuario.....	245
Figura 153. Confirmación baja usuario	246
Figura 154. Indicadores tramitación baja usuario	246
Figura 155. Asignar perfil	248
Figura 156. Confirmación asignar perfil especial	248
Figura 157. Confirmación asignar perfil básico	248
Figura 158. Confirmación eliminar permisos asignar perfil especial.....	249
Figura 159. Tabla de peticiones asignar perfil	249
Figura 160. Bloqueo usuario	250
Figura 161. Confirmación bloqueo usuario	250
Figura 162. Indicador usuario bloqueado	250
Figura 163. Desbloquear usuario.....	251
Figura 164. Confirmación desbloqueo usuario	251
Figura 165. Cambio contraseña usuario	252
Figura 166. Formulario cambio contraseña.....	252
Figura 167. Error cambio contraseña	253
Figura 168. Confirmación cambio contraseña.....	253
Figura 169. Permisos portafirmas.....	254
Figura 170. Cargos firmantes	254
Figura 171. Confirmación permisos de firma modificados.....	255
Figura 172. Permisos de usuario.....	256

Figura 173. Mensaje de usuario sin permisos.....	256
Figura 174. Información de perfil.....	257
Figura 175. Presupuesto	258

Índice de tablas

Tabla 1. Requisitos de capacidad	63
Tabla 2. Requisitos de restricción.....	63
Tabla 3. Requisitos funcionales.....	67
Tabla 4. Requisitos no funcionales de interfaz.....	68
Tabla 5. Requisitos no funcionales de operación	68
Tabla 6. Requisitos no funcionales de recursos.....	69
Tabla 7. Requisitos no funcionales de seguridad	70
Tabla 8. Matriz de trazabilidad UR/SR	73
Tabla 9. Matriz de trazabilidad SR/UC	76
Tabla 10. SUM Web. Caso de uso: bloquear usuario.....	82
Tabla 11. SUM Web. Caso de uso: desbloquear usuario	83
Tabla 12. SUM Web. Caso de uso: modificar contraseña.....	84
Tabla 13. SUM Web. Caso de uso: asignar permisos portafirmas	84
Tabla 14. SUM Web. Caso de uso: consultar permisos usuario.....	85
Tabla 15. SUM Web. Caso de uso: consultar perfil	85
Tabla 16. SUM Web. Caso de uso: solicitar alta usuario	87
Tabla 17. SUM Web. Caso de uso: solicitar baja usuario	88
Tabla 18. SUM Web. Caso de uso: solicitar asignar perfil usuario.....	90
Tabla 19. SUM Web. Caso de uso: consultar error petición	90
Tabla 20. SUM Web. Caso de uso: eliminar petición	91
Tabla 21. SUM Web. Caso de uso: solicitar acceso.....	92
Tabla 22. SUM Web. Caso de uso: consultar datos usuario.....	93
Tabla 23. SUM Web. Clase IUSER	108
Tabla 24. SUM Web. Clase DEPARTMENT	109
Tabla 25. SUM Web. Clase DEPARTMENT	109
Tabla 26. SUM Web. Clase IGROUP	109
Tabla 27. SUM Web. Clase IREQUEST.....	110
Tabla 28. SUM Web. Clase TASK.....	110
Tabla 29. SUM Web. Clase ENCRYPTION.....	111
Tabla 30. SUM Web. Clase ROLE.....	111

Tabla 31. SUM Server. Caso de uso: crear usuario.....	134
Tabla 32. SUM Server. Caso de uso: eliminar usuario	135
Tabla 33. SUM Server. Caso de uso: asignar perfil	136
Tabla 34. SUM Server. Clase IUSER.....	141
Tabla 35. SUM Server. Clase REMUSER	142
Tabla 36. SUM Server. Clase PROFILE.....	142
Tabla 37. SUM Server. Clase IREQUEST.....	142
Tabla 38. SUM Server. Clase TASK.....	143
Tabla 39. SUM Server. Clase DEPARTMENT	143
Tabla 40. SUM Server. Clase IGROUP	143
Tabla 41. SUM Server. Clase ROLE.....	144
Tabla 42. Modelo relacional: notación	149
Tabla 43. Modelo E/R: notación columna tipo.....	151
Tabla 44. Entidad SYSTEMUSER.....	152
Tabla 45. Entidad REMUSER.....	153
Tabla 46. Entidad IUSER	154
Tabla 47. Entidad DEPARTMENT	154
Tabla 48. Entidad EXTINF	155
Tabla 49. Entidad GROUP	156
Tabla 50. Entidad PROFILE	157
Tabla 51. Entidad SIGNROLE	158
Tabla 52. Entidad TASK	159
Tabla 53. Entidad REQUEST	160
Tabla 54. Entidad SIGNROLEVALUE	160
Tabla 55. Entidad TASKVALUE	161
Tabla 56. Entidad REQUESTVALUE	162
Tabla 57. Entidad USERGROUP	162
Tabla 58. Entidad SIGNPERM	163
Tabla 59. Líneas de código.....	166
Tabla 60. PRFSW001	183
Tabla 61. PRFSW002	183
Tabla 62. PRFSW003	184
Tabla 63. PRFSW004	184
Tabla 64. PRFSW005	184

Tabla 65. PRFSW006	184
Tabla 66. PRFSW007	185
Tabla 67. PRFSW008	185
Tabla 68. PRFSW009	185
Tabla 69. PRFSW010	185
Tabla 70. PRFSW011	186
Tabla 71. PRFSW012	186
Tabla 72. PRFSW013	186
Tabla 73. PRFSW014	186
Tabla 74. PRFSW015	187
Tabla 75. PRFSW016	187
Tabla 76. PRFSW017	187
Tabla 77. PRFSW018	188
Tabla 78. PRFSW019	188
Tabla 79. PRFSW020	188
Tabla 80. PRFSW021	189
Tabla 81. PRFSW022	189
Tabla 82. PRFSW023	189
Tabla 83. PRFSW024	189
Tabla 84. PRFSW025	190
Tabla 85. PRFSW026	190
Tabla 86. PRFSW027	190
Tabla 87. PRFSW028	191
Tabla 88. PRFSW029	191
Tabla 89. PRFSW030	191
Tabla 90. PRFSW031	191
Tabla 91. PRFSW032	192
Tabla 92. PRFSW033	192
Tabla 93. PRFSW034	192
Tabla 94. PRFSW035	193
Tabla 95. PRFSW036	193
Tabla 96. PRFSW037	193
Tabla 97. PRFSW038	193
Tabla 98. PRFSW039	194

Tabla 99. PRFSS001	194
Tabla 100. PRFSS002	195
Tabla 101. PRFSS003	195
Tabla 102. PRFSS004	195
Tabla 103. PRFSS005	196
Tabla 104. PRFSS006	196
Tabla 105. PRFSS007	196
Tabla 106. PRFSS008	197
Tabla 107. PRFSS009	197
Tabla 108. Matriz trazabilidad SR/PR.....	201
Tabla 109. Glosario de términos	263

CAPÍTULO 1.

INTRODUCCIÓN

1.1 MARCO DEL PROYECTO

El presente proyecto *Análisis y diseño de una plataforma Web para un sistema de gestión de usuarios* se encuentra integrado en un sistema corporativo de gestión de expedientes de telecomunicaciones.

Los usuarios del sistema se agrupan por departamentos y son tanto personal de Servicios Centrales como de Jefaturas Provinciales de Inspección de Telecomunicaciones (JPITs).

La gestión de los usuarios es, hoy en día, un área de entidad propia, en la medida en que la relación de éstos con los sistemas es crucial a la hora de asegurar un correcto funcionamiento de los servicios puestos a su disposición.

La seguridad de los datos de un sistema de gestión pasa por mantener un riguroso control de acceso a la información. Mediante la concesión de permisos se puede garantizar esa seguridad para controlar qué usuarios deben tener acceso a parte o a toda la información y quienes no. Se pueden asignar permisos sólo para acceder a la información o permisos para acceder a la información y modificarla.

Existen por tanto diferentes niveles de acceso a la información definida como grupos de acceso y perfiles departamentales. Los perfiles departamentales son un conjunto de permisos o grupos de acceso. Mediante la asignación de perfiles a usuarios se establecen los permisos de acceso al sistema.

En la actualidad, la gestión de las cuentas y permisos de usuario del sistema de gestión de expedientes de telecomunicaciones, la realiza el área de desarrollo informático de la propia empresa.

Gestionar toda la información sobre los usuarios internos a la empresa y que tienen acceso al sistema llega a convertirse en una tarea muy costosa en tiempo. Multitud de peticiones de alta, baja, o modificación de usuarios y permisos son generadas. Existe por tanto un mantenimiento continuo de los usuarios internos que tienen acceso a los diferentes servicios que ofrece el sistema de gestión de expedientes. El mantenimiento incluye asignación y modificación de permisos así como la creación o eliminación de los posibles usuarios del sistema.

El aumento de tiempo dedicado a esta labor hace que surja la necesidad de independizar en gran medida el mantenimiento de usuarios de los diferentes departamentos. Se desea por tanto que los usuarios finales del sistema sean capaces de autogestionar este aspecto, para lo cual se desarrollará una aplicación que permitirá a determinados usuarios de cada departamento dar de alta o baja usuarios y asignarles permisos a las cuentas adscritas a su mismo departamento.

El proyecto realizado *Análisis y diseño de una plataforma Web para un sistema de gestión de usuarios* proporciona una herramienta de trabajo a los usuarios finales y se desarrolla para cumplir con la necesidad establecida de autogestión y mantenimiento de usuarios por departamentos.

1.2 OBJETIVOS GENERALES

El objetivo general del proyecto es desarrollar un sistema para dotar a determinados usuarios de los diferentes departamentos de la empresa de una herramienta que facilite la gestión y el mantenimiento de los usuarios adscritos a su mismo departamento, permitiendo establecer de esta manera un control de los permisos de acceso de los usuarios a los diferentes servicios que ofrece el sistema de gestión de expedientes de telecomunicaciones e independizando en gran medida, esta labor del área de desarrollo informático y en consecuencia reduciendo ampliamente los tiempos empleados en dicho mantenimiento.

Además de crear dicha herramienta se generará la documentación específica del proyecto completo utilizando la metodología de desarrollo ESA.

La principal funcionalidad que el sistema debe de cumplir es permitir a determinados usuarios con permiso de administración para la gestión de usuarios, realizar las siguientes funciones:

- **Gestión de usuarios:** permitir el mantenimiento de usuarios del ámbito de la empresa dentro un departamento específico.
- **Gestión de permisos de usuarios:** permitir el mantenimiento de los permisos de acceso al sistema de gestión de expedientes de telecomunicaciones de los usuarios de un departamento específico.

La principal función de los usuarios del sistema de gestión de expedientes de telecomunicaciones es la tramitación de expedientes. Para evitar que el mantenimiento de usuarios y/o mantenimiento de permisos de un usuario pueda afectar a los procesos en curso de dicha tramitación, este tipo de acciones se realizarán en diferido. Por tanto, el sistema desarrollado se divide en dos módulos bien diferenciados.

- **Módulo Web de gestión de usuarios (SUM Web)** que permite ejecutar acciones online sobre los usuarios del departamento y establecer solicitudes de acciones diferidas. Este módulo será al que tengan acceso los usuarios finales.
- **Módulo Servidor de gestión de usuarios (SUM Server)** que se encarga de ejecutar las acciones diferidas solicitadas.

Los apartados [4.5 Arquitectura del sistema](#) y [4.6 Descripción del sistema](#) amplían la información aquí presente y detallan el funcionamiento de los diferentes módulos que componen el sistema de gestión de usuarios (SUM).

1.3 ESTRUCTURA DEL DOCUMENTO

El presente documento se estructura por capítulos. A continuación se describe el contenido de cada uno de ellos:

- **Capítulo 1. Introducción:** Este capítulo presenta una explicación genérica del proyecto desarrollado. Partiendo de la necesidad de generación del proyecto y estableciendo unos objetivos aporta una visión global en la que situar al lector.
- **Capítulo 2. Estado de la cuestión:** Expone una visión de conocimientos relacionados con el mismo espacio de trabajo en el que se ha desarrollado el proyecto, arquitectura Web, criptografía... Además incluye información de las herramientas utilizadas para el desarrollo del proyecto.
- **Capítulo 3. Organización del proyecto:** En este capítulo se presenta la organización y planificación del proyecto. Se citan todas las actividades y tareas necesarias para la consecución de objetivos, su distribución a lo largo del tiempo de desarrollo del proyecto y la asignación de recursos necesarios para cada una de ellas.
- **Capítulo 4. Análisis y Diseño:** En este capítulo se realiza una descripción detallada de la solución implementada. Contiene información de análisis y diseño de la solución que proporcionan la información necesaria para la compresión e implementación del proyecto.
- **Capítulo 5. Implementación:** En este capítulo se especifican detalles de implementación y construcción del sistema, ejemplos de código fuente, ficheros generados, extensión del sistema medido en líneas de código, etc.
- **Capítulo 6. Verificación:** Capítulo dedicado a la verificación del sistema implementado. Se presentan diferentes escenarios de prueba y se especifican los resultados obtenidos tras su realización así como la forma en la que han sido llevadas a cabo.
- **Capítulo 7. Conclusiones y líneas futuras:** En este capítulo se especifican las conclusiones obtenidas de la realización del proyecto y se presentan opciones como futuras líneas de desarrollo o mejora del proyecto.

CAPÍTULO 2. ESTADO DE LA CUESTIÓN

2.1 ARQUITECTURA WEB

Este apartado ofrece información detallada acerca del tipo de arquitectura Web.

2.1.1 Introducción

En los últimos años, la rápida expansión de Internet y del uso de intranets corporativas ha supuesto una transformación en las necesidades de información de las organizaciones. En particular esto afecta a la necesidad de que:

- La información sea accesible desde cualquier lugar dentro de la organización e incluso desde el exterior.
- Esta información sea compartida entre todas las partes interesadas, de manera que todas tengan acceso a la información completa (o a aquella parte que les corresponda según su función) en cada momento.

Estas necesidades han provocado un movimiento creciente de cambio de las aplicaciones tradicionales de escritorio hacia las aplicaciones Web, que cumplen a la perfección con las necesidades mencionadas anteriormente. Por tanto, los sitios Web tradicionales que se limitaban a mostrar información se han convertido en aplicaciones capaces de una interacción más o menos sofisticada con el usuario. Inevitablemente, esto ha provocado un aumento progresivo de la complejidad de estos sistemas y la necesidad de buscar nuevas opciones de diseño que permitan dar con la arquitectura óptima que facilite la construcción de los mismos.

No debemos olvidar que para que una aplicación Web sea eficaz, debemos exigirle algunas cualidades indispensables: seguridad, escalabilidad, portabilidad y un diseño eficiente de la interfaz y el programa.

Una de las primeras cuestiones que pueden surgir al plantearnos migrar nuestros datos a Internet es el riesgo que ello conlleva en cuanto a seguridad y disponibilidad. Este dilema nos sirve bien para introducir la distinción entre dos tipos de aplicaciones Web según su entorno de funcionamiento. Este puede ser una intranet o por el contrario una extranet. En el primer caso, la aplicación se aloja y está sólo disponible dentro de la red interna de la empresa y en el segundo, lo está en cualquier punto con acceso a Internet. Entonces si no disponemos de una buena conexión y/o preferimos tener nuestras aplicaciones dentro de la organización lo podremos hacer sin renunciar a las ventajas de las tecnologías de Internet. Si necesitamos llegar a usuarios fuera de nuestra red, optaremos por una extranet y colocaremos la aplicación en un servidor con buena conexión a Internet que suele contratarse a una compañía especializada.

2.1.2 Descripción arquitectura Web

Una aplicación Web es una aplicación informática que se ejecuta en un entorno Web. La arquitectura Web permite a múltiples clientes o usuarios establecer una comunicación a través de Internet con un servidor Web.

Los clientes mediante un navegador emiten peticiones vía HTTP a un servidor Web que responde a cada una de las solicitudes.

El servidor aloja una serie de aplicaciones Web que ofrecen diferentes servicios a los usuarios conectados. Es en el servidor Web donde se centraliza la carga de trabajo. En la mayoría de los casos, el navegador suele ser un mero presentador de información (modelo de cliente ligero), y no lleva a cabo ningún procesamiento relacionado con la lógica de negocio.

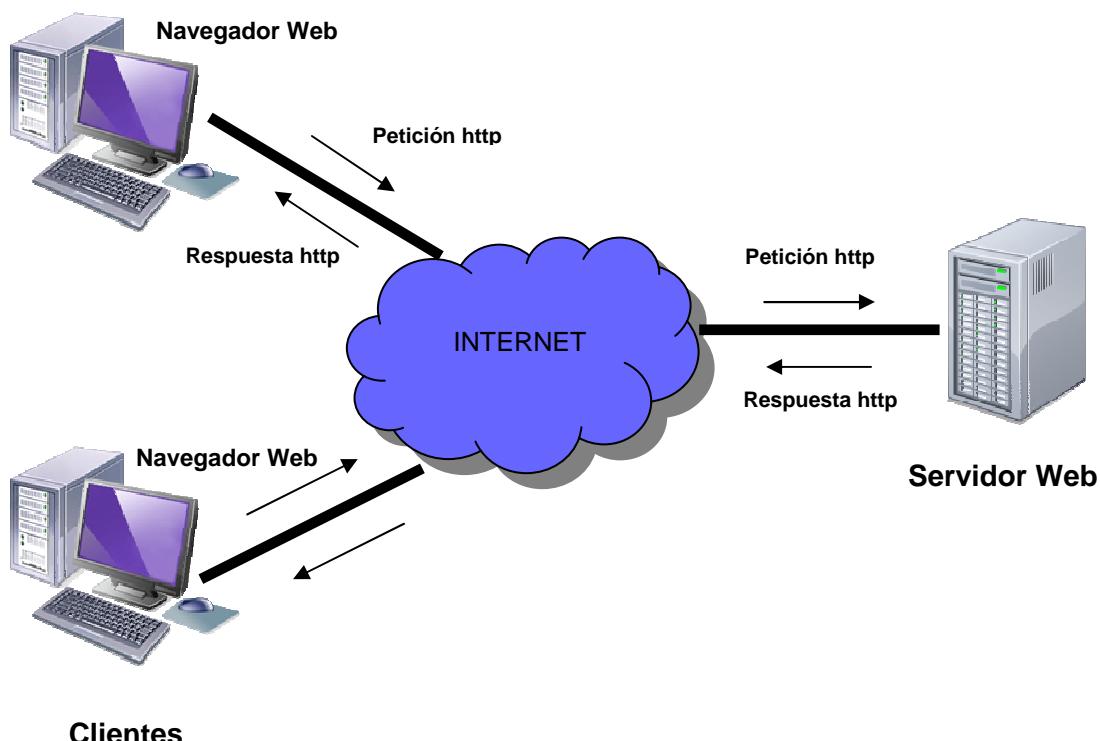
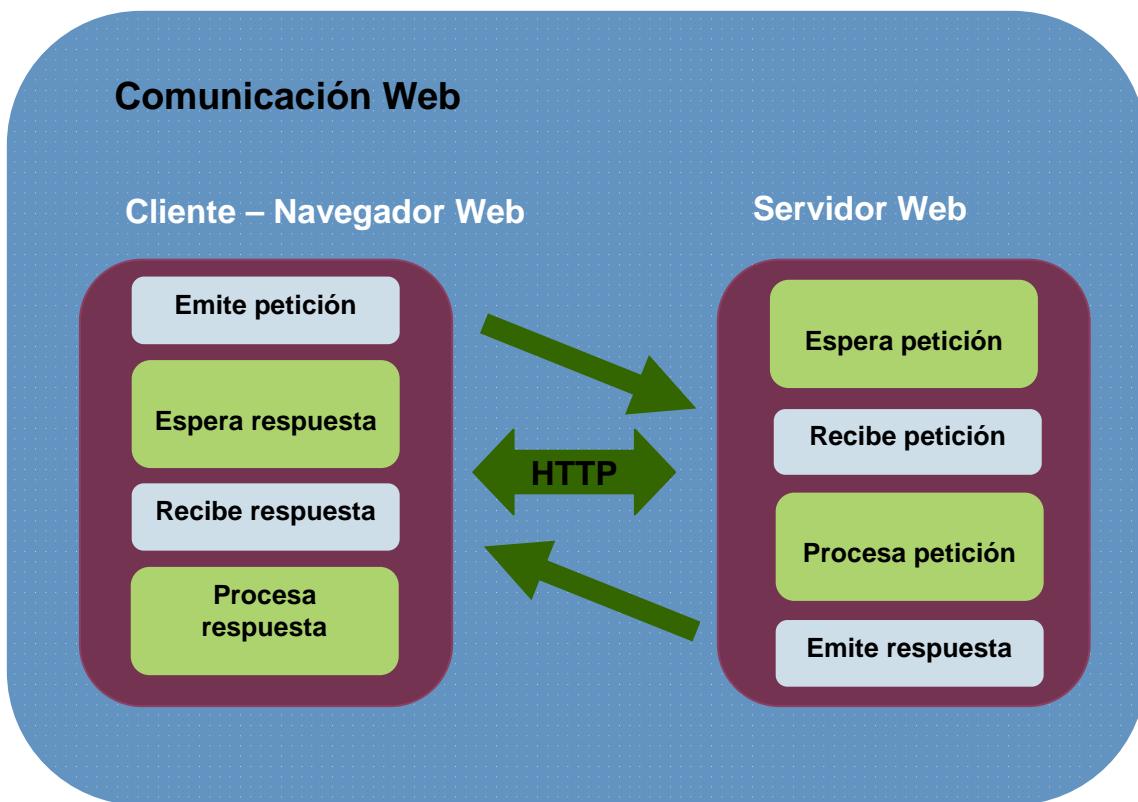


Figura 1. Arquitectura Web

En la arquitectura Web el modelo de comunicación entre cliente y servidor se establece mediante una conexión a Internet o intranet. El intercambio de información desde el navegador al Servidor se efectúa mediante el uso del protocolo HTTP (*Hipertext Transfer Protocol*) protocolo para el intercambio de información en Web.

El navegador Web interpreta las páginas dinámicas generadas por el servidor Web. El esquema de comunicaciones sigue el siguiente formato:

**Figura 2. Comunicación Web**

El usuario interacciona con las aplicaciones Web a través del navegador. Como consecuencia de la actividad del usuario, se envían peticiones al servidor, donde se aloja la aplicación Web y que normalmente hace uso de una base de datos que almacena toda la información relacionada con la misma. El servidor procesa la petición y devuelve la respuesta al navegador que la presenta al usuario.

Por tanto, el sistema se distribuye en tres componentes:

- el **navegador**, que presenta la interfaz al usuario
- la **aplicación**, que se encarga de realizar las operaciones necesarias según las acciones llevadas a cabo por éste
- la **base de datos**, donde la información relacionada con la aplicación se hace persistente.

Esta distribución se conoce como el modelo o arquitectura de tres capas.

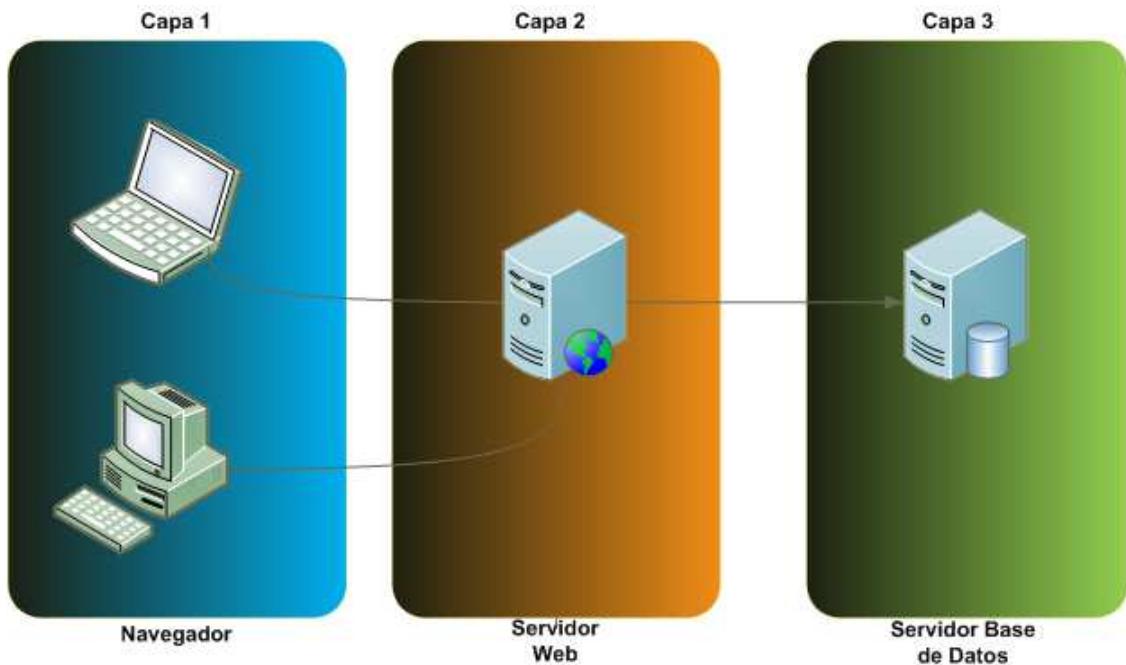


Figura 3. Arquitectura tres capas

La estrategia tradicional de utilizar aplicaciones compactas causa gran cantidad de problemas de integración en sistemas software complejos como pueden ser los sistemas de gestión de una empresa o los sistemas de información integrados consistentes en más de una aplicación. Estas aplicaciones suelen encontrarse con importantes problemas de escalabilidad, disponibilidad, seguridad, integración...

Para solventar estos problemas se ha generalizado la división de las aplicaciones Web en capas que normalmente serán tres donde:

- **Nivel de presentación (capa cliente):** se encarga de generar la interfaz de usuario. Captura datos de usuario, envía los datos a la capa intermedia y presenta los resultados que procedan de ésta.
- **Nivel de negocio (capa intermedia):** capa que constituye el verdadero núcleo de la aplicación Web. Contiene toda la lógica que modela los procesos de negocio y es donde se realiza todo el procesamiento necesario para atender a las peticiones del usuario. Se encarga del procesamiento de los datos del usuario y de la generación y envío de las respuestas a la capa cliente.
- **Nivel de administración de datos:** se encarga de hacer persistente toda la información, suministra y almacena información para el nivel de negocio.

El hecho de que se dividan las capas para organizar el código, no significa que las capas obligatoriamente deban correr en diferentes máquinas o que deben estrictamente correr en una sola máquina o en un único proceso.

Convertir un sistema de tres niveles a otro multinivel es fácil ya que consiste en extender la capa intermedia permitiendo que convivan múltiples aplicaciones en lugar de una sola.

A continuación se presenta un modelo genérico de n-capas:



Figura 4. Arquitectura n capas

Cada capa solamente interactúa con sus capas adyacentes lo que permite abstraer funcionalidades de las capas superiores e inferiores. Por ejemplo, la capa de presentación no se da cuenta de qué tipo de base de datos o qué repositorio de datos se utiliza porque ésta solamente se comunica con la capa de negocios, y el repositorio de datos no se da cuenta de dónde se está utilizando o desplegando la información ya que éste interactúa con la capa de acceso a datos.

La división entre la capa de presentación y la de la lógica de negocio permite una gran flexibilidad a la hora de construir aplicaciones, ya que se pueden tener múltiples interfaces sin cambiar la lógica de la aplicación.

Una arquitectura por capas nos facilita las siguientes ventajas:

- **Facilidad de desarrollo:** el desarrollo se puede llevar a cabo en varios niveles. Es posible realizar desarrollos paralelos (en cada capa).

- **Aplicaciones robustas:** se consigue gracias al encapsulamiento.
- **Facilidad de mantenimiento:** la modularidad facilita el mantenimiento y soporte.
- **Flexibilidad:** la modularidad facilita añadir o modificar módulos de una capa sin afectar a las restantes. Por ejemplo, se pueden tener múltiples interfaces sin cambiar la lógica de la aplicación o se puede modificar un módulo de lógica de negocio sin afectar a la interfaz.
- **Possibilidad de reutilización:** el código de la capa intermedia puede ser reutilizado por múltiples aplicaciones si esta diseñado en formato modular.
- **Alta escalabilidad:** se pueden manejar muchas peticiones con el mismo rendimiento simplemente añadiendo más hardware. El crecimiento es casi lineal y no es necesario añadir nuevo código para conseguirlo.

En contraposición a las ventajas citadas, las aplicaciones por capas pueden incrementar el tráfico en la red y requieren más balance de carga o tolerancia frente a posibles fallos.

2.1.3 Aspectos de seguridad en Web

Las aplicaciones Web están más expuestas a ataques. Se pueden tener ataques en tres niveles:

- Al cliente.
- Al servidor.
- A la información en tránsito.

La seguridad en Web tiene tres etapas primarias:

- Seguridad de la computadora del usuario.
- Seguridad del servidor Web y los datos.
- Seguridad de la información en tránsito.

Seguridad al cliente

Los usuarios deben contar con navegadores y plataformas seguras, libres de virus y vulnerabilidades. También debe garantizarse la privacidad de los datos del usuario.

Vulnerar el equipo del usuario quizás no tenga el impacto de vulnerar el servidor, sin embargo es un problema más difícil de erradicar. El número de clientes es muy superior al número de servidores. Por tanto se recomiendan las siguientes medidas para asegurar el equipo del usuario:

- Aplicar actualizaciones (parches) al sistema operativo.
- Uso de antivirus, *firewall*.
- Educación de los usuarios.

Seguridad del servidor Web y los datos

Se debe garantizar la operación continua del servidor, que los datos no sean modificados sin autorización (integridad) y que la información sólo sea distribuida a las personas autorizadas (control de acceso).

Se deben considerar los siguientes puntos:

- Asegurar el servidor en una forma fundamental: el sistema operativo, ya sea por medio de actualizaciones (parches) y habilitando los mecanismos propios de la plataforma.
- Garantizar la seguridad del servidor Web propiamente (IIS, Apache, etc.)
- Auditarse las aplicaciones que interactúan en las dos capas anteriores (módulos, bibliotecas).

Seguridad de la información en tránsito.

Garantizar que la información que viaja entre cliente y servidor no sea leída (confidencialidad), modificada o destruida por terceros. También es importante asegurar que el enlace entre cliente y servidor no pueda interrumpirse fácilmente (disponibilidad).

Esto se puede lograr por diversos medios:

- Asegurando la red físicamente (*switches* en lugar de *hubs*).
- Esconder la información (esteganografía).

Cifrar la información (criptografía) por medio de algoritmos diversos (SSL, VPNs).

2.1.4 Ventajas \ desventajas de aplicaciones Web

A continuación se presentan posibles ventajas y desventajas de las aplicaciones Web.

Ventajas de las aplicaciones Web.

- **Compatibilidad multiplataforma:** se pueden usar desde cualquier sistema operativo porque sólo es necesario tener un navegador actualizado.

- **Actualizaciones inmediatas:** siempre se accede a la última versión de la aplicación. No es necesario realizar actualizaciones desde el cliente.
- **Acceso inmediato:** no necesitan ser descargadas, instaladas y configuradas lo que supone un significativo ahorro de tiempo.
- **Menor impacto frente a errores:** la localización de las aplicaciones Web en el servidor permite actualizarlas rápidamente frente a problemas que surjan de manera que la actualización esté disponible para todos los usuarios. Además evita múltiples problemas de interacción con otro software o conflictos de hardware en la máquina cliente.
- **Consumo bajo de recursos:** la carga de trabajo de estas aplicaciones reside en el servidor por tanto se reducen el uso de recursos de memoria en el cliente. Además al estar alojadas en el servidor no ocupan espacio de disco duro en cliente.
- **Reducción costes:** el mantenimiento del sistema se concentra en el servidor, por tanto, se reducen costes de instalación y actualización en clientes. Además estas aplicaciones permiten ahorrar invertir en costosas infraestructuras de comunicaciones.
- **Alta disponibilidad de servicio y datos:** gracias a la conectividad que proporciona Internet se permite el acceso a la aplicación desde cualquier punto. Esto facilita la movilidad y el teletrabajo.
- **Colaboración:** pueden ser utilizada por múltiples usuarios al mismo tiempo.
- **Seguridad de datos:** la localización de los datos en un servidor los protege frente a posibles virus en el equipo cliente.
- **Portabilidad:** compatibilidad de la aplicación con múltiples sistemas operativos y plataformas existentes en el mercado.

Desventajas de las aplicaciones Web.

- **Conexión a Internet:** la disponibilidad del servicio está supeditada al proveedor.
- **Comunicación no segura:** la comunicación se realiza a través de un medio que no es seguro, lento y con diferentes protocolos de comunicación.
- **Seguridad en el servidor:** las aplicaciones Web deben estar en constante resguardo. Si no son seguras podrían perderse los datos o no ser accesibles.
- **Compatibilidad de idioma:** la mayor parte de las aplicaciones Web se crea en un idioma concreto sin soportar otros lenguajes.

- **Dependencia a plugins:** muchas aplicaciones dependen de determinados complementos por ejemplo Flash, Java... que genera la necesidad de instalación y actualización en cliente.
- **Espacio reducido de almacenamiento:** muchos proveedores de servicios Web ofrecen un espacio limitado y reducido para los archivos del cliente.
- **Menor funcionalidad:** aunque en este sentido los navegadores van mejorando, la funcionalidad que otorga un navegador es menor que la que otorga el sistema operativo, por tanto, pueden aportar menor funcionalidad que las aplicaciones de escritorio.

2.1.5 Patrón de diseño MVC

El patrón MVC [12] (Modelo Vista Controlador) es un patrón de arquitectura de software encargado de separar la lógica de negocio de la interfaz del usuario y es el más utilizado en aplicaciones Web, ya que facilita la funcionalidad, mantenibilidad y escalabilidad del sistema, de forma simple y sencilla, a la vez que permite “no mezclar lenguajes de programación en el mismo código”.

MVC especifica cómo debe ser estructurada una aplicación, las capas que van a componer la misma y la funcionalidad de cada una. Según este patrón, la capa intermedia de una aplicación Web se divide en tres niveles de abstracción:

- Controlador
- Vista
- Modelo

La siguiente figura muestra esta arquitectura. En ella se puede ver cómo se relacionan estos tres bloques funcionales entre sí, su interacción con el resto de las capas de la aplicación.

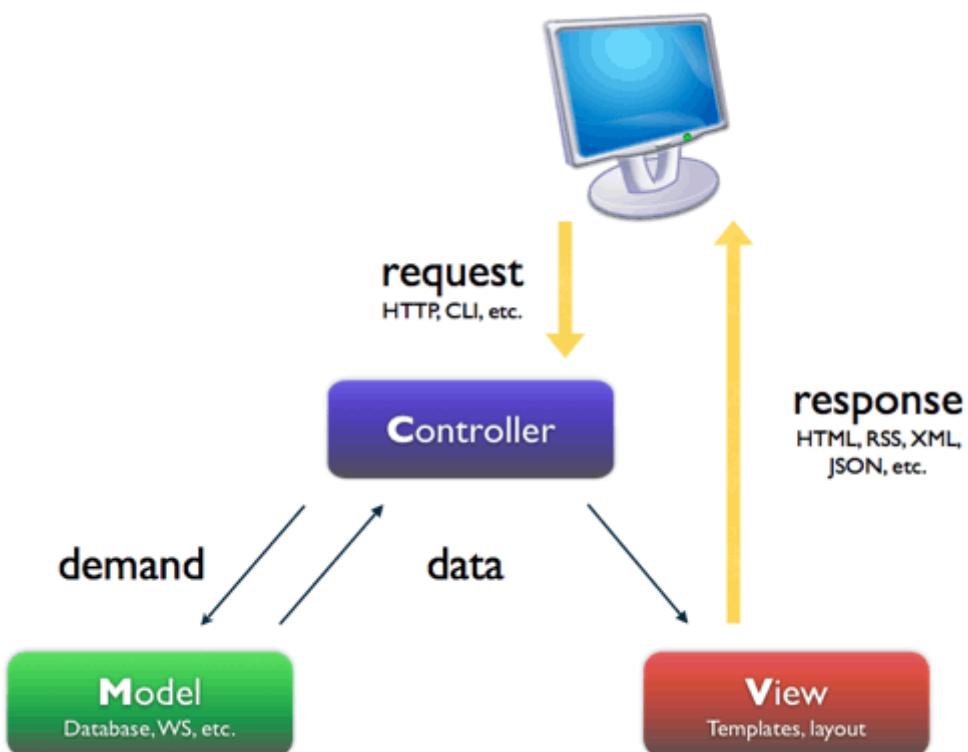


Figura 5. Arquitectura MVC

Funcionamiento básico del patrón MVC:

- El usuario realiza una petición a través de la interfaz de usuario
- El controlador captura el evento
- El controlador hace la llamada al modelo/modelos correspondientes efectuando las modificaciones pertinentes sobre el modelo
- El controlador recibe la información y la envía a la vista
- La vista, recibe datos del modelo y los muestra al usuario

Controlador

Es el intermediario entre la vista y el modelo. Se puede decir que el controlador es el cerebro de la aplicación. Todas las peticiones a la capa intermedia que se realicen desde el cliente son dirigidas al controlador, cuya misión es determinar las acciones a realizar para cada una de estas peticiones e invocar al resto de los componentes de la aplicación (Modelo y Vista) para que realicen las acciones requeridas en cada caso, encargándose también de la coordinación de todo el proceso. En definitiva, controla las interacciones del usuario solicitando los datos al modelo y entregándolos a la vista para que ésta lo presente al usuario.

La centralización del flujo de peticiones en el Controlador proporciona varias ventajas al programador, entre ellas:

- Hace que el desarrollo sea más sencillo y limpio.
- Facilita el posterior mantenimiento de la aplicación haciéndola más escalable.
- Facilita la detección de errores en el código.

Vista

La Vista es la encargada de mostrar la información al usuario permitiendo a los usuarios interactuar con la aplicación. Genera las respuestas que deben ser enviadas al cliente. Cuando esta respuesta tiene que incluir datos proporcionados por el Controlador, el código de la página no será fijo sino que deberá ser generado de forma dinámica. Hay cuatro tipos de Vista:

- Presentación de datos (Gráficos, tablas, listas)
- Formularios para ingreso de datos
- Pestañas de navegación (Menús, hipervínculos, mapas del sitio)
- Ventanas emergentes o pop-ups (Instrucciones, mensajes de ayuda, mensajes de error, diálogos de confirmación)

En una aplicación Web, las vistas serían las páginas HTML que el usuario visualiza en el navegador. A través de estas páginas el usuario interactúa con la aplicación, enviando eventos al servidor a través de peticiones HTTP. En el servidor se encuentra el código de control para estos eventos, que en función del evento concreto actúa sobre el modelo convenientemente. Los resultados de la acción se devuelven al usuario en forma de página HTML mediante la respuesta HTTP.

Modelo

En arquitectura MVC la lógica del negocio de la aplicación, incluyendo el acceso a los datos y su manipulación, está encapsulada dentro del Modelo.

El Modelo lo forman una serie de componentes de negocio independientes del Controlador y la Vista, permitiendo así su reutilización y el desacoplamiento entre las capas.

El modelo vista controlador se aplica en el diseño de interfaces ya que la lógica de la interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica de negocio. El modelo suele ser más estable a lo largo del tiempo y menos sujeto a variaciones mientras que la vista puede cambiar con frecuencia, ya sea por cambio del medio de presentación o por necesidades de usabilidad de la interfaz o simple renovación de la estética de la aplicación

Se trata de realizar un diseño que desacople la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio o de datos.

Los beneficios que ofrece esta arquitectura son:

- Facilita la realización de pruebas.
- Facilita el mantenimiento de las aplicaciones. Se puede sustituir un componente sin afectar a otros.
- Facilita el desarrollo de software. No interfiere en el desarrollo de otros componentes.
- Ofrece la posibilidad de reutilizar componentes en otras aplicaciones.
- Los desarrollos suelen ser más escalables.

También es posible considerar las siguientes desventajas:

- Ceñirse a una estructura predefinida puede incrementar la complejidad del sistema.
- Algunos problemas son más difíciles de resolver siguiendo este patrón.
- Modelo más complejo que otros como por ejemplo *Web Forms*.

La distribución de componentes obliga a crear y mantener un mayor número de ficheros.

2.1.6 Formularios Web

Las páginas de formularios *Web Forms* pueden usarse para crear páginas Web programables que sirvan como interfaz de usuario de las aplicaciones Web. Este tipo de páginas presenta la información al usuario en cualquier explorador o dispositivo cliente e implementa lógica de aplicación mediante el código de la parte servidor. La salida de las páginas de formularios *Web Forms* puede contener casi cualquier lenguaje compatible con HTTP, incluidos HTML, XML, WML y ECMAScript (JScript, JavaScript).

Las páginas de formularios *Web Forms* reúnen las siguientes características:

- Se basan en la tecnología Microsoft ASP.NET, en la que el código que se ejecuta en el servidor genera de forma dinámica salida de páginas Web en un explorador o dispositivo cliente.
- Compatible con cualquier explorador o dispositivo móvil. Las páginas de formularios *Web Forms* presentan automáticamente el código HTML adecuado al explorador para funciones tales como estilos, diseño, etc. Como alternativa, se pueden diseñar las páginas de formularios *Web Forms* para ejecutarse en un explorador determinado, como Microsoft Internet Explorer 5 y aprovechar así todas las funciones de un cliente de explorador de nivel superior.
- Admiten cualquier lenguaje compatible con *Common Language Runtime* de .NET, incluidos Microsoft Visual Basic, Microsoft Visual C# y Microsoft JScript.NET.
- Se crean en el entorno Microsoft .NET Framework. Esto proporciona todos los beneficios del marco de trabajo, incluidos un entorno administrado, seguridad de tipos y herencia.

- Respaldadas en Visual Studio por eficaces herramientas de desarrollo rápido de aplicaciones (RAD, *Rapid Application Development*) destinadas al diseño y la programación de los formularios.
- Extensibles mediante controles que proporcionan posibilidades RAD al desarrollo Web, lo que permite crear con rapidez interfaces de usuario enriquecidas.
- Flexibles gracias a la posibilidad de incorporar a ellas controles creados por los usuarios y de otros fabricantes.

Componentes de los formularios *Web Forms*

En las páginas de formularios *Web Forms*, la programación de la interfaz de usuario se divide en dos partes independientes: el componente visual y el lógico. Se separa la parte visible de un formulario y el código que se oculta detrás y que interactúa con él.

El elemento visual se conoce como la *página* de formularios *Web Forms*, y se compone de un archivo que contiene código HTML estático, o controles de servidor ASP.NET o ambos de forma simultánea.

La lógica de las páginas de formularios *Web Forms* se compone del código creado para interactuar con el formulario. La lógica de programación reside en un archivo independiente del archivo de la interfaz de usuario. Este archivo se conoce como el archivo de "código subyacente".

Estructura de los archivos de formularios *Web Forms*

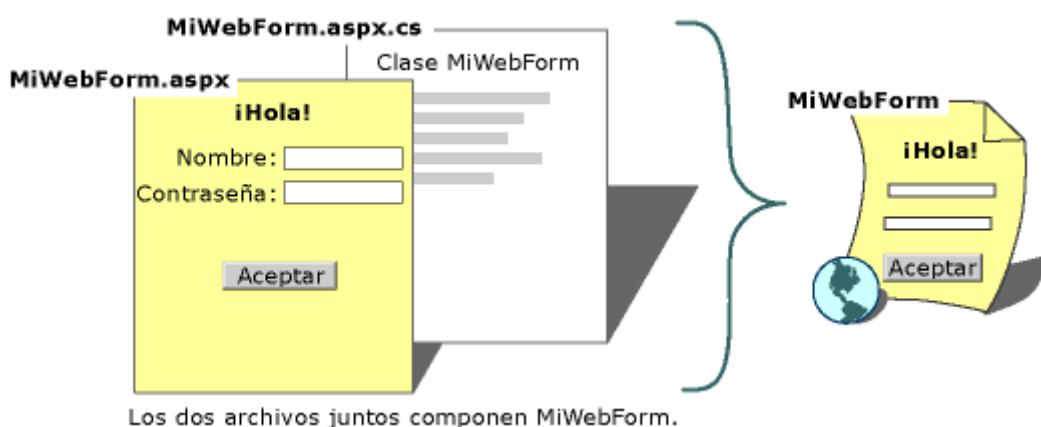


Figura 6. Estructura de archivos de *Web Forms*

Los archivos de código subyacente de todas las páginas de formularios *Web Forms* de un proyecto se compilan en el archivo de biblioteca de vínculos dinámicos (.dll) del

proyecto. El archivo de página .aspx también se compila, pero de un modo distinto. La primera vez que un usuario examina la página .aspx con el explorador, ASP.NET genera automáticamente un archivo de clase .NET que representa a la página y que la compila en un segundo archivo .dll. La clase generada para la página .aspx hereda de la clase del código subyacente que se compiló en el archivo .dll del proyecto. Cuando un usuario solicita la dirección URL de la página Web, los archivos .dll se ejecutan en el servidor y producen de forma dinámica la salida HTML de la página.

Ventajas que aportan las páginas de formularios *Web Forms*

- **Modelo de objetos intuitivo:** la generación de formularios es muy sencilla debido a los diferentes controles de servidor y elementos de formulario que proporciona, reduciendo así el tiempo de desarrollo.
- **Modelo de programación controlado por eventos:** el mecanismo subyacente de captura de eventos en el cliente, su transmisión al servidor y la llamada al método apropiado se realiza de modo automático e invisible para el implementador.
- **Administración intuitiva de los estados:** los formularios *Web Forms* reducen la complejidad de la administración del estado al utilizar controles basados en servidor y estado de vista.
- **Arquitectura basada en páginas:** la separación del componente visual y el lógico puede facilitar la creación de páginas que implementan tareas comunes.
- **Aplicaciones independientes del explorador:** la lógica de la aplicación reside en el servidor, sin embargo, permite generar código cliente.
- **Compatibilidad con *Common Language Runtime* de .NET Framework:** Las aplicaciones pueden crearse en cualquier lenguaje compatible con CLR, entre ellos Microsoft Visual Basic, Visual C# y JScript .NET.
- **Rendimiento de servidor escalable de .NET Framework.** El marco de trabajo de páginas ASP.NET permite escalar las aplicaciones Web de un equipo con un único procesador a una "batería de servidores Web" con varios equipos limpiamente y sin cambios complicados en la lógica de la aplicación.

Compatibilidad del marco de ASP.NET con MVC

La tecnología *Web Forms* [13] [14] y MVC están basadas en el marco de ASP.NET. Por consiguiente, la mayoría de las características del marco de ASP.NET que ha utilizado para crear aplicaciones basadas en formularios *Web Forms* también están a su disposición para el desarrollo de aplicaciones MVC.

Existen determinadas incompatibilidades. Dado que ASP.NET MVC no mantiene información de estado utilizando el estado de vista, debe buscar otras maneras de administrar la información de estado, si la necesita. Además, los controles de servidor

que confían en el estado de vista y devolución (*postback*) no funcionarán conforme a su diseño en una aplicación ASP.NET MVC. Por consiguiente, no debe utilizar controles como **GridView**, **DataList** y **Repeater**.

El ciclo de vida de una página de formularios *Web Forms* es complejo, durante el cual se producen muchos eventos. Estos eventos no se admiten en una aplicación ASP.NET MVC, porque las interacciones entre el modelo, la vista y el controlador usan un modelo de ciclo de vida diferente.

2.2 CRIPTOGRAFÍA: FUNCIÓN RESUMEN

Un *hash* es una función criptográfica de tipo resumen cuyo objetivo es identificar únicamente un conjunto de datos. No es realmente un algoritmo de cifrado sino más bien un algoritmo matemático que genera un resumen (*Digest*, *Hash* o *Checksum*) al aplicarlo a un valor inicial y que permite verificar la integridad del fichero o mensaje pero no permite obtener el original. Actúa como cualquier función matemática y como tal requiere cumplir ciertos criterios para que el resultado obtenido tenga determinadas propiedades y sea un resultado válido.

Propiedades de una función resumen.

- **Facilidad de uso:** Calcular el resumen de una entrada debe ser un trabajo sencillo.
- **Resumen de longitud fija:** La longitud del resultado de la función *hash* siempre es la misma independientemente de la longitud de la entrada.
- **Transformación no reversible (*one way function*):** Función de única dirección. Es imposible obtener el valor de entrada a partir del resumen del mismo.
- **Resultado único:** Para cada entrada la función genera una salida única.
- **Resistencia a colisiones:** Fuerte resistencia frente a colisiones, es decir, es difícil o imposible computacionalmente encontrar dos entradas distintas que produzcan la misma salida.

Debido a sus propiedades, el valor de una función *hash* puede ser usado para verificar la integridad de un texto o realizar comparaciones de textos a partir de su resumen.

Una de las ventajas del uso de funciones *hash* es una mayor rapidez frente a la criptografía de clave pública.

Algoritmo MD5.

MD5 es una función resumen ampliamente difundida. Su algoritmo fue desarrollado por Ronald Rivest en 1995. El algoritmo MD5 (*Message Digest Algorithm 5*, Algoritmo de Resumen del Mensaje 5) es un algoritmo de reducción criptográfico de 128 bits ampliamente usado, es decir, produce un número de 128 bits a partir de un texto de cualquier longitud.

La ventaja de este tipo de algoritmos es la imposibilidad (computacional) de reconstruir la cadena original a partir del resultado, y también la imposibilidad o dificultad de encontrar dos cadenas de texto que generen el mismo resultado.

Esto nos permite usar el algoritmo para transmitir contraseñas a través de un medio inseguro. Simplemente se cifra la contraseña, y se envía de forma cifrada. En el punto de destino, para comprobar si la contraseña es correcta, se cifra de la misma manera y se comparan las formas cifradas.

MD5 comienza rellenando la entrada a una longitud congruente en módulo 448 mod 512. Es decir la longitud del mensaje es 64 bits menos que un entero múltiplo de 512. El relleno consiste en un bit en 1 seguido por tantos bits 0 como sean necesarios. La longitud original del mensaje es almacenada en los últimos 64 bits del relleno.

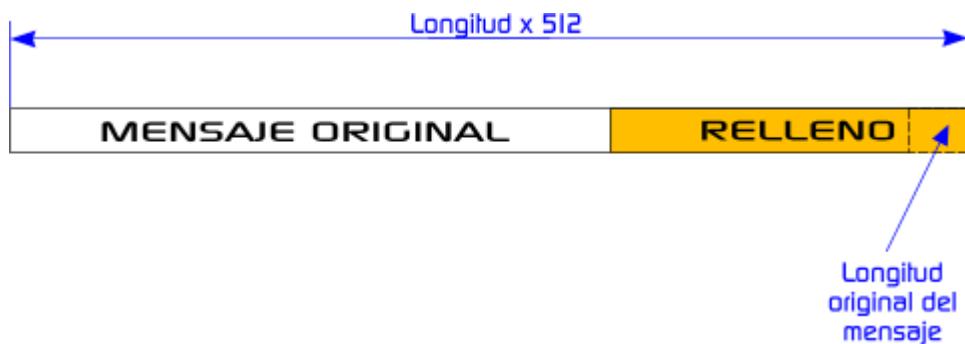


Figura 7. Mensaje cifrado MD5

Adicionalmente se inicializa, con un valor fijo, un buffer de 128 bits. Este buffer puede verse como 4 registros de 32 bits (A, B C, D) y son inicializados con los siguientes valores hexadecimales:

A=67452301;
B=EFCDAB89;
C=98BADCFE;
D=10325476

Durante varias rondas de procesamiento el algoritmo toma bloques de 512 bits de la entrada y los mezcla con los 128 bits del buffer. Este proceso es repetido hasta que todos los bloques de entrada han sido consumidos. El valor resultante en el buffer es el *hash* del mensaje.

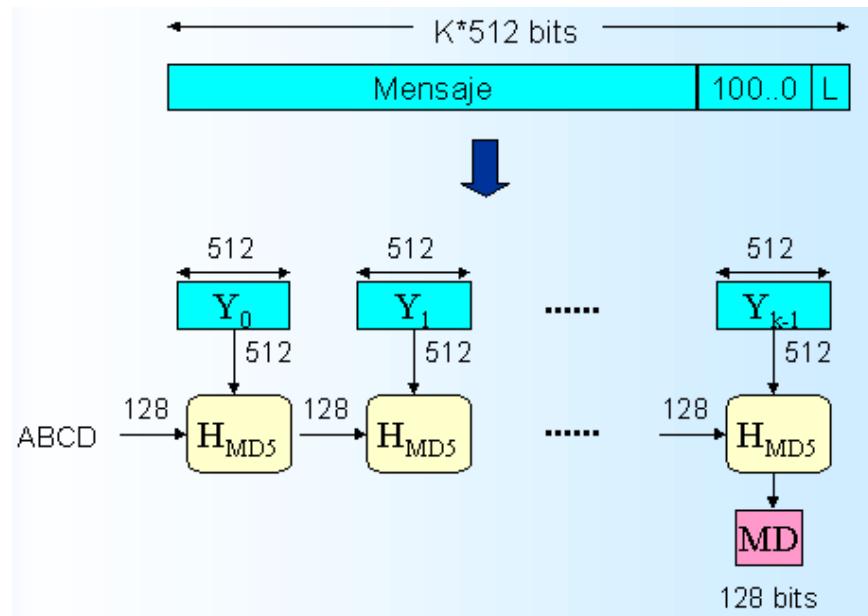


Figura 8. Funcionamiento MD5

2.3 TECNOLOGÍAS UTILIZADAS

.La tecnología empleada para la implementación del proyecto (módulo SUM Web y módulo SUM Server) es Microsoft .NET, siendo el entorno de desarrollo Microsoft Visual Studio y Microsoft SQL Server el sistema de gestión de datos relacional, tal cual se indica en los requisitos de usuario.

Entre los lenguajes de programación empleados destaca Visual Basic .NET como lenguaje principal de desarrollo así como HTML o JavaScript que complementan la implementación.

A continuación se detalla información acerca de las tecnologías utilizadas en el desarrollo del presente proyecto.

2.3.1 Microsoft .NET

El *Framework* de .Net [15] es una infraestructura, actualmente gratuita, que reúne un conjunto de lenguajes y servicios que simplifican en gran medida el desarrollo de aplicaciones. Esta herramienta ofrece un entorno de ejecución altamente distribuido, que permite crear aplicaciones robustas y escalables.

Los principales componentes de este entorno son:

- Lenguajes de compilación
- Biblioteca de clases de .Net
- CLR (*Common Language Runtime*)

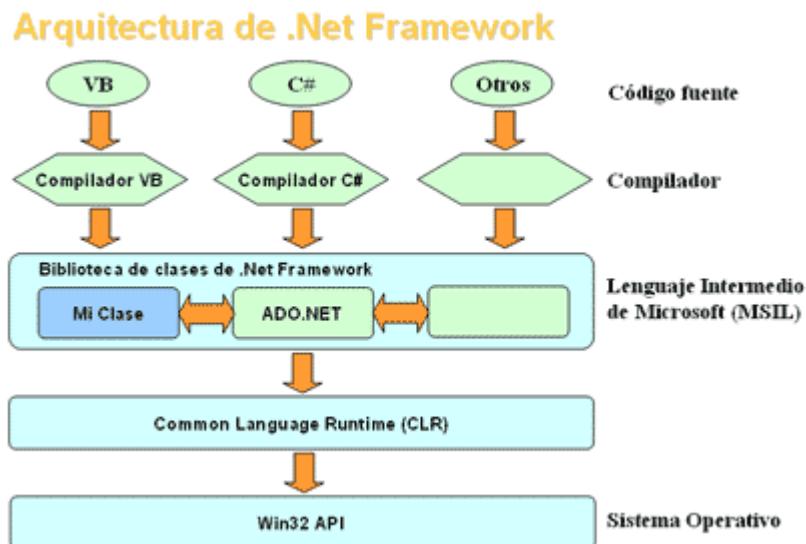


Figura 9. Arquitectura de .Net Framework

Common Language Runtime (CLR)

El CLR es el verdadero núcleo del *Framework* de .Net, ya que es el entorno de ejecución en el que se cargan las aplicaciones desarrolladas en los distintos lenguajes, ampliando el conjunto de servicios que ofrece el sistema operativo.



Figura 10. Estructura interna del entorno de ejecución en lenguaje común

.NET Framework soporta múltiples lenguajes de programación como C#, Visual Basic, C++, etc. Esto es posible debido a que el compilador de la herramienta de desarrollo compila el código fuente de cualquier lenguaje soportado por .Net y genera automáticamente un mismo código, transparente al desarrollo de la aplicación, denominado código intermedio (MSIL, *Microsoft Intermediate Language*). Para generar un código MSIL compatible con el CLR se basa en las reglas que establece el (CLS, *Common Language Specification*).

De esta forma, indistintamente de la herramienta de desarrollo utilizada y del lenguaje fuente elegido, el código generado es siempre el mismo, ya que el MSIL es el único lenguaje que entiende directamente el CLR.

Sin embargo, el código generado en MSIL no es código máquina y por tanto no puede ejecutarse directamente. Se necesita un segundo paso en el que una herramienta denominada compilador JIT (*Just-In-Time*) genera el código máquina real que se ejecuta en la plataforma que tenga la computadora.

De esta forma se consigue con .Net cierta independencia de la plataforma, ya que cada plataforma puede tener su compilador JIT y crear su propio código máquina a partir del código MSIL.

La compilación JIT la realiza el CLR a medida que se invocan los métodos en el programa y, el código ejecutable obtenido, se almacena en la memoria caché de la computadora, siendo recompilado sólo cuando se produce algún cambio en el código fuente.

Biblioteca de clases de .Net

Cuando se está programando una aplicación muchas veces se necesitan realizar acciones como manipulación de archivos, acceso a datos, conocer el estado del sistema,

implementar seguridad, etc. El *Framework* organiza toda la funcionalidad del sistema operativo en un espacio de nombres jerárquico de forma que a la hora de programar resulta bastante sencillo encontrar lo que se necesita.

Para ello, el *Framework* posee un sistema de tipos universal, denominado *Common Type System* (CTS). Este sistema permite que el programador pueda interactuar los tipos que se incluyen en el propio *Framework* (biblioteca de clases de .Net) con los creados por él mismo (clases). De esta forma se aprovechan las ventajas propias de la programación orientada a objetos, como la herencia de clases predefinidas para crear nuevas clases, o el polimorfismo de clases para modificar o ampliar funcionalidades de clases ya existentes.

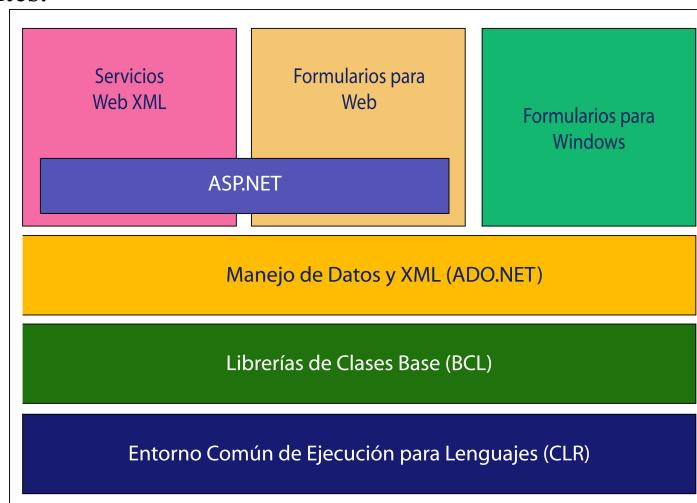


Figura 11. Diagrama básico de la biblioteca de clases de .Net *Framework*

La biblioteca de clases de .Net *Framework* incluye, entre otros, tres componentes clave:

- ASP.NET para construir aplicaciones y servicios Web.
- *Windows Forms* para desarrollar interfaces de usuario.
- ADO.NET para conectar las aplicaciones a bases de datos.

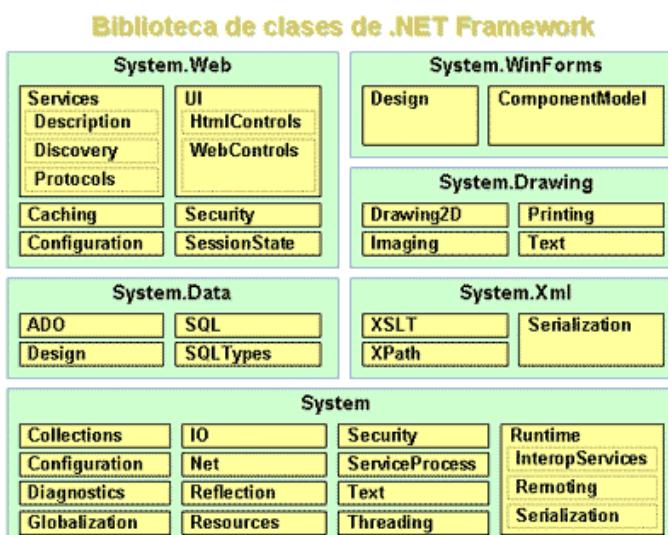


Figura 12. Diagrama detallado de la biblioteca de clases de .Net *Framework*

La forma de organizar la biblioteca de clases de .Net dentro del código es a través de los espacios de nombres (*namespaces*), donde cada clase está organizada en espacios de nombres según su funcionalidad. Por ejemplo, para manejar ficheros se utiliza el espacio de nombres **System.IO** y si lo que se quiere es obtener información de una fuente de datos se utilizará el espacio de nombres **System.Data**.

La principal ventaja de los espacios de nombres de .Net es que de esta forma se tiene toda la biblioteca de clases de .Net centralizada bajo el mismo espacio de nombres (**System**). Además, desde cualquier lenguaje se usa la misma sintaxis de invocación, ya que a todos los lenguajes se aplica la misma biblioteca de clases.

Ensamblados

Uno de los mayores problemas de las aplicaciones actuales es que en muchos casos tienen que tratar con diferentes archivos binarios, elementos de registro, conectividad abierta a bases de datos (ODBC), etc.

Para solucionarlo el *Framework* de .Net maneja un nuevo concepto denominado ensamblado. Los ensamblados son ficheros con forma de EXE o DLL que contienen toda la funcionalidad de la aplicación de forma encapsulada. Por tanto la solución al problema puede ser tan fácil como copiar todos los ensamblados en el directorio de la aplicación.

Con los ensamblados ya no es necesario registrar los componentes de la aplicación. Esto se debe a que los ensamblados almacenan dentro de sí mismos toda la información necesaria en lo que se denomina el manifiesto del ensamblado. El manifiesto recoge todos los métodos y propiedades en forma de meta-datos junto con otra información descriptiva, como permisos, dependencias, etc.

Para gestionar el uso que hacen las aplicaciones de los ensamblados .Net utiliza la llamada caché global de ensamblados (GAC, *Global Assembly Cache*). Así, .Net *Framework* puede albergar en el GAC los ensamblados que puedan ser usados por varias aplicaciones e incluso distintas versiones de un mismo ensamblado.



Figura 13. Diagrama interno de un ensamblado .Net

2.3.2 Microsoft SQL Server

Microsoft SQL Server es un sistema de gestión de bases de datos relacionales producido por Microsoft que funciona sobre una plataforma Windows.

El sistema está basado en el lenguaje Transact-SQL. Incluye compatibilidad integrada con XML. También cuenta con las características de escalabilidad, disponibilidad y seguridad necesarias para operar el componente de almacenamiento de datos de los sitios Web de mayor tamaño. Adaptable a todo tipo de organización y estructura, permite poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea.

SQL Server usa la arquitectura Cliente / Servidor para separar la carga de trabajo en tareas que corran en computadoras tipo Servidor y tareas que corran en computadoras tipo Cliente:

- El Cliente es responsable de la parte lógica y de presentar la información al usuario. Generalmente, el cliente corre en una o más computadoras Cliente, aunque también puede correr en una computadora Servidor con SQL Server.
- SQL Server administra Bases de Datos y distribuye los recursos disponibles del servidor (tales como memoria, operaciones de disco, etc.) entre las múltiples peticiones.

Microsoft SQL Server es una plataforma completa e integrada y permite obtener más valor de las actuales capacidades de tecnología de la información (TI), aumentando la productividad y la agilidad de los departamentos de TI, y creando rápidamente aplicaciones flexibles e innovadoras.

2.3.3 Microsoft Visual Studio

Microsoft Visual Studio es un entorno de desarrollo integrado (IDE) para sistemas operativos Windows. Orientado al entorno de desarrollo de software de .NET Framework. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, y Visual Basic .NET, al igual que entornos de desarrollo Web como ASP.NET.

Visual Studio proporciona las herramientas necesarias para diseñar, desarrollar, depurar e implementar aplicaciones Web, Servicios Web XML y aplicaciones cliente tradicionales.

Permite la creación de soluciones en varios lenguajes y asegura código de calidad durante todo el ciclo de vida de la aplicación, desde el diseño hasta la implementación, facilitando en gran medida la creación de programas y reduciendo el tiempo empleado.

2.3.4 Visual Basic.NET

Visual Basic .NET es un lenguaje de programación de alto nivel de .NET Framework. Se considera un pilar de .NET Framework.

Características del lenguaje de programación Visual Basic .NET

- Es un lenguaje orientado a objetos y eventos que soporta encapsulación, herencia y polimorfismo.
- Es una mejora a Visual Basic formando parte de Visual Studio en la plataforma de .NET Framework y compartiendo entorno de desarrollo con otros lenguajes de programación como Microsoft Visual C++ .NET, Microsoft Visual C# .NET, etc.
- Provee una IDE sencilla de manejar
- Soporte para LINQ

2.3.5 JavaScript

JavaScript (JScript) es un lenguaje de programación que no requiere compilación. Se utiliza principalmente en la parte cliente en el desarrollo de páginas Web. Permite añadir funcionalidad o realizar mejoras en la interfaz de usuario y páginas Web dinámicas.

El lenguaje JavaScript utiliza una sintaxis semejante a la del lenguaje Java, pero al contrario que este, JavaScript no es un lenguaje orientado a objetos propiamente dicho, ya que no dispone de herencia, sino que es un lenguaje basado en prototipos.

Se ha empleado este lenguaje para complementar el desarrollo principal del proyecto, y añadir funciones para validación de datos, modificación de la interfaz de usuario, añadir y/o modificar comportamiento en determinados eventos, etc.

2.3.6 HTML

HTML es un lenguaje que se utiliza fundamentalmente en el desarrollo de páginas Web, comúnmente para establecer la estructura y contenido de un sitio Web, tanto de texto, objetos e imágenes. Los archivos desarrollados en HTML tienen la extensión .htm o .html

El lenguaje HTML funciona por medio de etiquetas que describen la apariencia o función del texto enmarcado. También puede llegar a incluir un script o código que tenga incidencia en el comportamiento del navegador Web.

La funcionalidad de HTML es sencilla y puede ser creado y editado en cualquier editor Web o de textos. Por lo general los diseñadores utilizan herramientas para el desarrollo de aplicaciones Web que permiten generar de manera automática el código fuente HTML.

2.4 ESTÁNDAR ESA

El estándar de la ESA [1] es un conjunto de estándares definidos por la Agencia Espacial Europea para la especificación, desarrollo y mantenimiento del software.

Para realizar el proyecto del sistema SUM se ha utilizado el estándar ESA como guía de desarrollo.

Se considera un proyecto de software pequeño si cumple al menos uno de los siguientes criterios:

- Si se necesitan menos de dos años hombre de esfuerzo para el desarrollo.
- Si se requiere un equipo único de desarrollo de cinco o menos personas.
- Si la cantidad de código fuente es inferior a 10000 líneas, excluyendo los comentarios.

El estándar propone el siguiente ciclo de vida para proyectos de software pequeños.

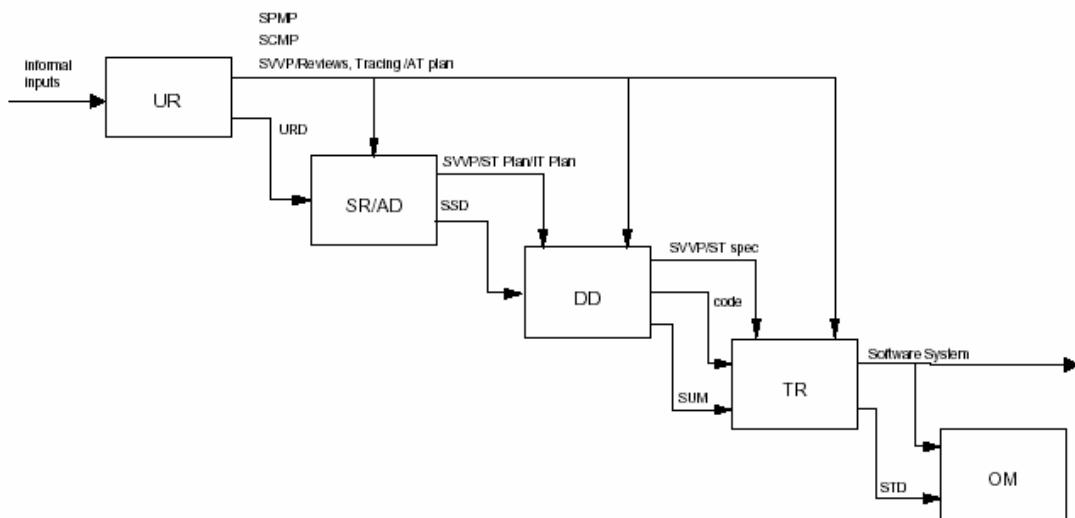


Figura 14. Ciclo de vida de un proyecto de software pequeño según la ESA

Considerando las siguientes fases básicas:

- Fase RU - Definición de los requisitos de usuario.
- Fase RS - Definición de los requisitos de software.
- Fase DA - Definición del diseño arquitectónico.
- Fase DD - Diseño detallado y producción del código.
- Fase TR - Transferencia de software a las operaciones.
- Fase OM - Operaciones y mantención.

En proyectos pequeños, las fases RS y DA están combinadas en una fase RS/DA

CAPÍTULO 3.

ORGANIZACIÓN DEL

PROYECTO

3.1 ROLES ORGANIZATIVOS

A continuación se detalla el conjunto de roles adoptados para la ejecución del presente proyecto así como su estructura organizativa.

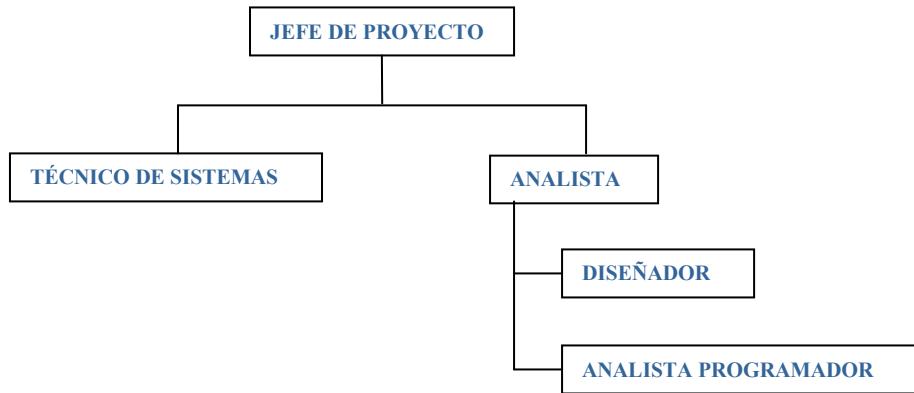


Figura 15. Estructura organizativa

- **Jefe de proyecto.** Gerente y responsable del proyecto. Se encarga de realizar funciones tales como:
 - Comunicaciones con el cliente
 - Provee al proyecto de los recursos humanos y materiales necesarios
 - Planifica el proyecto
 - Supervisa el proyecto así como los recursos humanos que lo componen
 - Detecta posibles riesgos
 - Resuelve conflictos
 - Toma decisiones que involucran un beneficio directo hacia el proyecto
- **Técnico de sistemas:** operario de control y mantenimiento del sistema. Se encarga de las siguientes tareas:
 - Mantenimiento y control de los diferentes servidores para el desarrollo, pruebas y puesta en producción del sistema.
 - Control de red y base de datos.
 - Efectúa las acciones de sistemas solicitadas por los usuarios responsables del desarrollo del proyecto ejecutando paquetes de instalación.
- **Analista:** realiza las tareas correspondientes al análisis de proyecto. Analiza los requisitos del usuario, redacta la documentación específica de análisis y supervisa que los requisitos establecidos sean cumplidos a lo largo del desarrollo del proyecto.

- **Diseñador:** realiza las tareas correspondientes al diseño de proyecto siguiendo el análisis efectuado por el analista y genera la documentación específica de diseño.
- **Analista programador:** desarrolla e implementa el código de las aplicaciones que constituyen el proyecto siguiendo las indicaciones estipuladas por el analista y el diseñador. Toma las decisiones oportunas para un óptimo desarrollo del código de las aplicaciones y efectúa las pruebas necesarias para la consecución de la funcionalidad requerida.

3.2 PLANIFICACIÓN

Las siguientes figuras muestran la descomposición de tareas de las distintas fases de realización del proyecto que conforman la planificación.

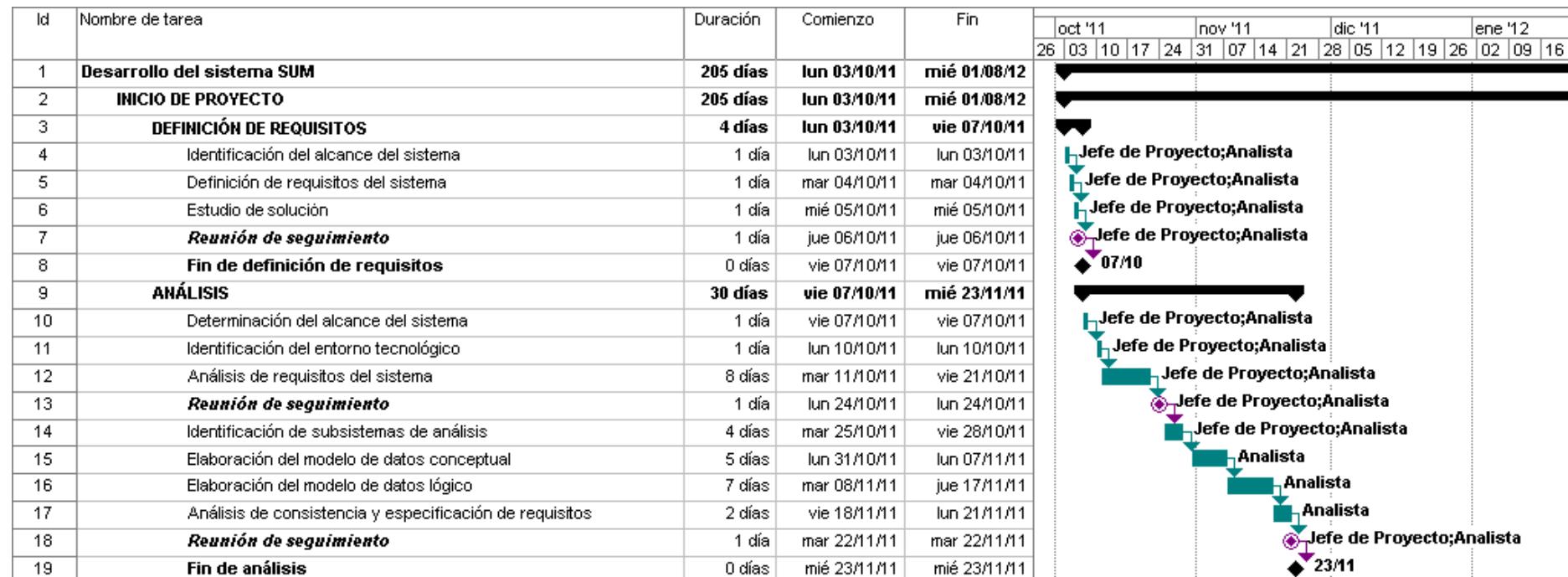


Figura 16. Planificación I

CAPÍTULO 3. ORGANIZACIÓN DEL PROYECTO



Figura 17. Planificación II



Figura 18. Planificación III

CAPÍTULO 3. ORGANIZACIÓN DEL PROYECTO



Figura 19. Planificación IV

CAPÍTULO 4. ANÁLISIS Y DISEÑO

4.1 REQUISITOS DE USUARIO

Esta sección describe los requisitos de usuario del sistema SUM. La especificación de requisitos de usuario se divide en requisitos de capacidad del sistema y requisitos de restricción impuestos al sistema por el usuario.

La especificación de requisitos se realiza en una tabla con las siguientes propiedades:

- Identificador: identificador único de requisito de usuario.
- Descripción: descripción de requisito de usuario.
- Necesidad: refleja la importancia del requisito para el usuario.
- Prioridad: indica la prioridad establecida para un requisito siendo el valor **alta** el de mayor prioridad y **baja** el de menor prioridad.
- Estabilidad: indica si el requisito está sujeto a modificaciones.
- Fuente: indica la procedencia de la especificación del requisito.

4.1.1 Requisitos de capacidad

Los siguientes requisitos hacen referencia a la capacidad del sistema.

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
UR001	Mostrar nombre del departamento.	Deseable	Media	Modificable	Usuario
UR002	Mostrar lista de usuarios del departamento.	Esencial	Alta	No Modificable	Usuario
UR003	Mostrar lista de perfiles del departamento.	Esencial	Alta	No modifiable	Usuario
UR004	Permitir bloquear usuario, desbloquear usuario, cambiar	Esencial	Alta	No Modificable	Usuario

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
	contraseña de usuario, asignar permisos de portafirmas a usuario, consultar permisos de usuario, consultar datos de usuario, consultar datos de perfil, crear usuario, eliminar usuario, asignar perfil a usuario.				
UR005	Las acciones crear usuario, eliminar usuario y asignar perfil a usuario se procesarán en diferido.	Esencial	Alta	No Modificable	Ing. Requisitos
UR006	Mostrar descripción de perfil y grupos o permisos que otorga.	Deseable	Alta	Modificable	Usuario
UR007	Sólo podrán crearse usuarios que ya existan en el directorio de usuarios específico de la empresa.	Esencial	Alta	No modifiable	Usuario
UR008	Al crear un nuevo usuario será obligatorio introducir una contraseña de usuario.	Esencial	Alta	No modifiable	Usuario
UR009	Al bloquear usuario del departamento mostrar indicativo de usuario bloqueado.	Deseable	Media	Modificable	Usuario
UR010	Un usuario bloqueado no tendrá acceso a GEXTEL, mientras que un usuario desbloqueado sí.	Esencial	Alta	No modifiable	Usuario
UR011	Al desbloquear un usuario del departamento se eliminará el indicativo de usuario bloqueado y se permitirá su acceso a GEXTEL.	Deseable	Media	Modificable	Usuario
UR012	Se permitirá cambiar la contraseña de acceso a GEXTEL de un usuario del departamento.	Esencial	Alta	No modifiable	Usuario
UR013	La acción asignar permisos portafirmas mostrará los permisos de portafirmas del departamento y marcará con un check aquellos que tenga asignados el usuario seleccionado.	Esencial	Alta	Modificable	Usuario
UR014	La acción consultar permisos de usuario mostrará todos los permisos que el usuario seleccionado tiene asignados.	Esencial	Alta	No modifiable	Usuario
UR015	Acceso a la aplicación a usuarios con permiso específico.	Esencial	Alta	No modifiable	Ing. Requisitos

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
UR016	La creación de un usuario proporciona acceso a GEXTEL.	Esencial	Alta	No modificable	Usuario

Tabla 1. Requisitos de capacidad

4.1.2 Requisitos de restricción

Los siguientes requisitos hacen referencia a las restricciones impuestas al sistema.

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
UR017	La aplicación será de fácil manejo para un usuario familiarizado con el entorno Windows.	Deseable	Alta	No Modificable	Ing. Requisitos
UR018	Funcionará sobre un ordenador personal con sistema operativo Windows XP o Windows 7.	Esencial	Alta	Modificable	Ing. Requisitos
UR019	Funcionará sobre un ordenador personal con procesador de 3 Ghz, 3 GB de RAM y 40 GB libres de disco duro.	Esencial	Alta	Modificable	Ing. Requisitos
UR020	El navegador en el que se presentará la aplicación será IExplorer 7.	Esencial	Alta	Modificable	Ing. Requisitos
UR021	Para poder validar nuevos usuarios vía LDAP contra el directorio de usuarios de la empresa será necesario disponer de conexión a Internet.	Esencial	Alta	Modificable	Ing. Requisitos
UR022	La generación de código se implementará en lenguaje VB.NET y tecnología de Microsoft .NET. Se utilizará SQL Server como gestor de base de datos relacional.	Esencial	Alta	No Modificable	Ing. Requisitos

Tabla 2. Requisitos de restricción

4.2 REQUISITOS SOFTWARE

Esta sección describe los requisitos software del sistema SUM. La especificación de requisitos software se divide en requisitos funcionales que especifican qué debe hacer el sistema y requisitos no funcionales que especifican cómo debe de llevarse a cabo.

La especificación de requisitos se realiza en una tabla con las siguientes propiedades:

- Identificador: identificador único de requisito software.
- Descripción: descripción de requisito software.
- Necesidad: refleja la importancia del requisito.
- Prioridad: indica la prioridad establecida para un requisito siendo el valor **alta** el de mayor prioridad y **baja** el de menor prioridad.
- Estabilidad: indica si el requisito está sujeto a modificaciones.
- Fuente: indica la procedencia de la especificación del requisito.

4.2.1 Requisitos funcionales

Los requisitos funcionales que a continuación se detallan definen qué hace el sistema.

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
SR001	Mostrar nombre del departamento del usuario conectado.	Deseable	Media	Modificable	UR001
SR002	Mostrar lista de usuarios del departamento.	Esencial	Alta	No Modificable	UR002
SR003	Mostrar lista de usuarios pendientes de alta.	Deseable	Alta	Modificable	UR002
SR004	Mostrar lista de perfiles básicos y especiales del departamento.	Esencial	Alta	No modifiable	UR003
SR005	Permitir bloquear usuario, desbloquear usuario, cambiar	Esencial	Alta	No Modificable	UR004

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
	contraseña de usuario, asignar permisos de portafirmas a usuario, consultar permisos de usuario, consultar datos de usuario, consultar datos de perfil, crear usuario, eliminar usuario, asignar perfil a usuario.				
SR006	Las acciones crear usuario, eliminar usuario y asignar perfil a usuario se procesarán en diferido.	Esencial	Alta	No Modificable	UR005
SR007	Se generará una petición por acción diferida solicitada.	Esencial	Alta	No modifiable	UR005
SR008	Se creará una tarea programada por cada petición de usuario generada.	Esencial	Alta	No modifiable	UR005
SR009	Las tareas programadas generadas serán ejecutadas por un proceso <i>batch</i> .	Esencial	Alta	Modificable	UR005
SR010	Se podrán crear tantas peticiones como sea necesario sin repetir el contenido de las mismas.	Esencial	Alta	No modifiable	UR005
SR011	Mostrar lista de peticiones generadas. Sólo se mostrarán aquellas peticiones con marca de visualización.	Deseable	Alta	Modificable	UR005
SR012	Tendrán marca de visualización aquellas peticiones cuyo estado sea distinto de OK.	Opcional	Media	Modificable	UR005
SR013	Mostrar los siguientes datos de una petición: identificador, usuario que la creó, acción, argumentos, estado y fecha de actualización.	Deseable	Media	Modificable	UR005
SR014	Se podrán eliminar peticiones.	Esencial	Alta	No modifiable	UR005
SR015	Se podrá consultar la descripción de error de las peticiones en las que hayan surgido.	Esencial	Alta	No modifiable	UR005
SR016	Mostrar nombre completo de usuario y perfil asignado al seleccionar un usuario del departamento.	Opcional	Media	Modificable	UR004
SR017	Mostrar descripción de perfil y grupos o permisos que otorga.	Deseable	Alta	Modificable	UR006

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
SR019	Al crear un nuevo usuario será obligatorio introducir una contraseña de usuario.	Esencial	Alta	No modificable	UR008
SR021	Al asignar un perfil básico se eliminarán primero todos los permisos que tuviese el usuario y a continuación se otorgarán los permisos del nuevo perfil.	Esencial	Media	No modificable	UR004
SR022	Al asignar un perfil especial se solicitará confirmación del usuario para eliminar o mantener los permisos que el usuario tuviese previamente asignados.	Deseable	Baja	Modificable	UR004
SR023	La acción asignar perfil se habilitará si está seleccionado un usuario y un perfil al mismo tiempo.	Esencial	Alta	No modificable	UR004
SR024	Las acciones bloquear/desbloquear usuario son exclusivas en función del estado del usuario del departamento seleccionado.	Esencial	Alta	No modificable	UR004
SR025	Al bloquear usuario del departamento mostrar indicativo de usuario bloqueado.	Deseable	Media	Modificable	UR009
SR026	Un usuario bloqueado no tendrá acceso a GEXTEL, mientras que un usuario desbloqueado sí.	Esencial	Alta	No modificable	UR010
SR027	Al desbloquear un usuario del departamento se eliminará el indicativo de usuario bloqueado y se permitirá su acceso a GEXTEL.	Deseable	Media	Modificable	UR011
SR028	Se permitirá cambiar la contraseña de acceso a GEXTEL de un usuario del departamento.	Esencial	Alta	No modificable	UR012
SR029	La acción asignar permisos portafirmas mostrará los permisos de portafirmas del departamento y marcará con un check aquellos que tenga asignados el usuario del departamento seleccionado.	Esencial	Alta	Modificable	UR013
SR030	La acción consultar permisos de usuario mostrará todos los	Esencial	Alta	No modificable	UR014

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
	permisos que el usuario seleccionado tiene asignados.				
SR031	Al eliminar un usuario pendiente de alta se eliminarán las peticiones pendientes de ejecución de ese usuario y el usuario se eliminará de la lista de usuarios pendientes de alta.	Esencial	Alta	No modificable	UR005
SR032	Al eliminar un usuario del departamento se mostrará junto al nombre del usuario una marca indicativa de usuario pendiente de eliminar y se creará la petición correspondiente.	Esencial	Alta	No modificable	UR005
SR033	Al eliminar una petición de eliminación de usuario se eliminará la petición y el indicativo correspondiente del usuario del departamento.	Esencial	Alta	No modificable	UR005
SR036	No puede haber más de un usuario seleccionado al mismo tiempo.	Deseable	Baja	Modificable	UR004
SR037	No puede haber más de un perfil seleccionado al mismo tiempo.	Deseable	Baja	Modificable	UR004
SR038	Al eliminar una petición pendiente de alta de nuevo usuario se eliminarán todas las peticiones pendientes de dicho usuario y el usuario desaparecerá de la lista de usuarios pendientes de alta.	Esencial	Alta	No modificable	UR005
SR040	No se incluirán en la lista de usuarios del departamento los usuarios especiales.	Esencial	Alta	No modificable	UR002
UR041	La creación de un usuario proporciona acceso a GEXTEL.	Esencial	Alta	No modificable	UR016

Tabla 3. Requisitos funcionales

4.2.3 Requisitos no funcionales de interfaz

Los requisitos de interfaz definen el modo en que interactúan y se comunican las interfaces y diferentes módulos del sistema.

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
SR042	La aplicación será de fácil manejo para un usuario familiarizado con el entorno Windows.	Deseable	Alta	No Modificable	UR017
SR043	Funcionará sobre un ordenador personal con sistema operativo Windows XP o Windows 7.	Esencial	Alta	Modificable	UR018
SR045	El navegador en el que se presentará la aplicación será IEExplorer 7.	Esencial	Alta	Modificable	UR020

Tabla 4. Requisitos no funcionales de interfaz

4.2.4 Requisitos no funcionales de operación

Los requisitos de operación definen cómo debe de operar del sistema.

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
SR020	Se solicitará confirmación en las siguientes acciones: eliminar usuario, asignar perfil, eliminar petición, bloquear usuario y desbloquear usuario.	Deseable	Baja	Modificable	UR004

Tabla 5. Requisitos no funcionales de operación

4.2.5 Requisitos no funcionales de recursos

Los requisitos de recursos definen los recursos mínimos para que la aplicación pueda ponerse en marcha.

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
SR044	Funcionará sobre un ordenador personal con procesador de 3 Ghz, 3 GB de RAM y 40 GB libres de disco duro.	Esencial	Alta	Modificable	UR019
SR046	Para validar nuevos usuarios vía LDAP contra el directorio de usuarios de la empresa será necesario disponer de conexión a Internet.	Esencial	Alta	Modificable	UR021
SR047	La generación de código se implementará en lenguaje VB y tecnología de Microsoft .NET. La base de datos debe implementarse en SQL Server.	Esencial	Alta	No Modificable	UR022

Tabla 6. Requisitos no funcionales de recursos

4.2.8 Requisitos no funcionales de seguridad

Los requisitos de seguridad definen las características de seguridad del sistema.

Identificador	Descripción	Necesidad	Prioridad	Estabilidad	Fuente
SR018	Al crear un nuevo usuario se deberá validar vía LDAP contra el directorio de usuarios específico de la empresa.	Esencial	Alta	No modificable	UR007
SR034	Acceso a la aplicación a usuarios con permiso específico.	Esencial	Alta	No modificable	UR015
SR035	El permiso específico de acceso a la aplicación no puede estar contenido en ningún perfil ni será posible otorgarlo desde la	Esencial	Alta	Modificable	UR015

	aplicación.				
SR039	Las contraseñas que manipule el sistema deben ser cifradas mediante la función MD5.	Esencial	Alta	No modificable	UR008

Tabla 7. Requisitos no funcionales de seguridad

4.3 MATRIZ DE TRAZABILIDAD UR/SR

La siguiente tabla establece la relación entre los requisitos de usuario y los requisitos software del sistema. Como puede observarse todos los requisitos de usuario quedan cubiertos por requisitos software.

CAPÍTULO 4. ANÁLISIS Y DISEÑO

/UR	UR0 01	UR0 02	UR0 03	UR0 04	UR0 05	UR0 06	UR0 07	UR0 08	UR0 09	UR0 10	UR0 11	UR0 12	UR0 13	UR0 14	UR0 15	UR0 16	UR0 17	UR0 18	UR0 19	UR0 20	UR0 21	UR0 22
SR001	✓																					
SR002		✓																				
SR003		✓																				
SR004			✓																			
SR005				✓																		
SR006					✓																	
SR007					✓																	
SR008					✓																	
SR009					✓																	
SR010					✓																	
SR011					✓																	
SR012						✓																
SR013						✓																
SR014						✓																
SR015						✓																
SR016						✓																

CAPÍTULO 4. ANÁLISIS Y DISEÑO

/UR	UR0 01	UR0 02	UR0 03	UR0 04	UR0 05	UR0 06	UR0 07	UR0 08	UR0 09	UR0 10	UR0 11	UR0 12	UR0 13	UR0 14	UR0 15	UR0 16	UR0 17	UR0 18	UR0 19	UR0 20	UR0 21	UR0 22
SR017						✓																
SR018							✓															
SR019								✓														
SR020				✓																		
SR021				✓																		
SR022				✓																		
SR023				✓																		
SR024				✓																		
SR025								✓														
SR026									✓													
SR027										✓												
SR028											✓											
SR029												✓										
SR030													✓									
SR031					✓																	
SR032					✓																	

/UR	UR0 01	UR0 02	UR0 03	UR0 04	UR0 05	UR0 06	UR0 07	UR0 08	UR0 09	UR0 10	UR0 11	UR0 12	UR0 13	UR0 14	UR0 15	UR0 16	UR0 17	UR0 18	UR0 19	UR0 20	UR0 21	UR0 22
SR033					✓																	
SR034															✓							
SR035															✓							
SR036					✓																	
SR037					✓																	
SR038					✓																	
SR039									✓													
SR040	✓																					
SR041																✓						
SR042																	✓					
SR043																		✓				
SR044																			✓			
SR045																			✓			
SR046																				✓		
SR047																					✓	

Tabla 8. Matriz de trazabilidad UR/SR

4.4 MATRIZ DE TRAZABILIDAD SR/UC

La siguiente tabla determina qué casos de uso del sistema cubren la funcionalidad especificada en los requisitos software. Como puede observarse toda la funcionalidad establecida en los requisitos software el sistema queda cubierta por al menos un caso de uso.

SR/UC	UC SUM Web													UC SUM Server		
	UC001	UC002	UC003	UC004	UC005	UC006	UC007	UC008	UC009	UC010	UC011	UC012	UC013	UC001	UC002	UC003
SR001												✓				
SR002												✓				
SR003												✓				
SR004												✓				
SR005	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓
SR006						✓	✓	✓					✓	✓	✓	
SR007						✓	✓	✓					✓	✓	✓	
SR008						✓	✓	✓					✓	✓	✓	
SR009													✓	✓	✓	
SR010						✓	✓	✓								
SR011						✓	✓	✓				✓				

SR/UC	UC SUM Web													UC SUM Server		
	UC001	UC002	UC003	UC004	UC005	UC006	UC007	UC008	UC009	UC010	UC011	UC012	UC013	UC001	UC002	UC003
SR012							✓	✓	✓					✓	✓	✓
SR013												✓				
SR014											✓	✓				
SR015										✓		✓				
SR016													✓			
SR017						✓										
SR018							✓									
SR019							✓								✓	
SR020	✓	✓						✓	✓		✓					
SR021									✓							✓
SR022									✓							✓
SR023						✓			✓			✓	✓			
SR024	✓	✓									✓	✓				
SR025	✓										✓					
SR026	✓	✓									✓					
SR027		✓										✓				

SR/UC	UC SUM Web													UC SUM Server		
	UC001	UC002	UC003	UC004	UC005	UC006	UC007	UC008	UC009	UC010	UC011	UC012	UC013	UC001	UC002	UC003
SR028			✓									✓				
SR029				✓												
SR030					✓											
SR031								✓								
SR032								✓								
SR033											✓					
SR034												✓		✓	✓	✓
SR035												✓				
SR036													✓			
SR037						✓										
SR038												✓				
SR039			✓					✓	✓	✓				✓	✓	✓
SR040												✓				
SR041														✓		

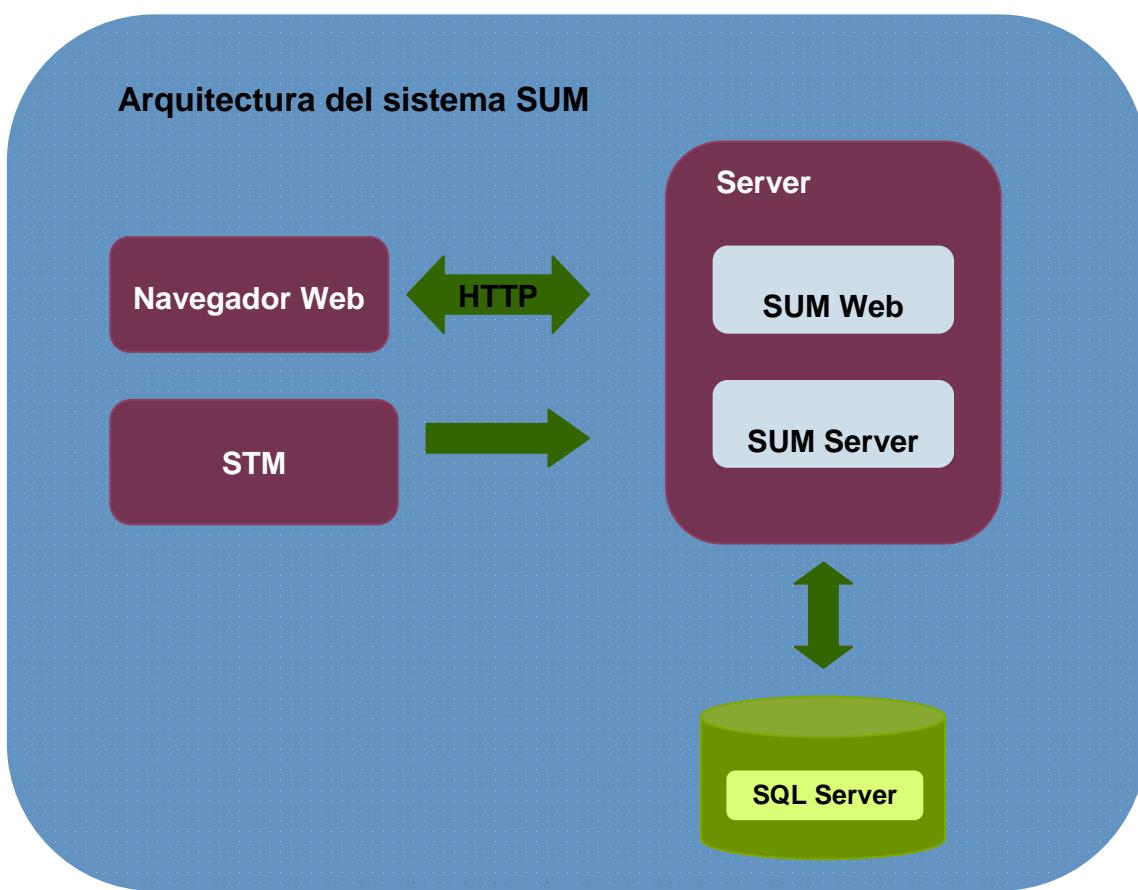
Tabla 9. Matriz de trazabilidad SR/UC

4.5 ARQUITECTURA DEL SISTEMA

La arquitectura del sistema SUM (System Users Management) para la administración de usuarios se compone de los siguientes módulos:

- **Módulo Navegador Web:** representa la parte cliente del sistema. Utiliza una conexión vía http para acceder al sistema SUM.
- **Módulo STM (*System Tasks Management*):** Es un sistema externo de planificación de tareas. Este sistema inicia la ejecución del módulo de servidor SUM Server. En el presente proyecto no se incluye el desarrollo ni la documentación específica de este módulo.
- **Módulo de acceso a datos:** SGBD SQL Server para gestionar la información del sistema de gestión de usuarios.
- **SUM (*System Users Management*):** Sistema de gestión de usuarios formado por dos módulos:
 - **SUM Web:** módulo Web de gestión de usuarios. Ejecuta acciones de gestión de usuarios *on line* y crea peticiones de acciones diferidas de gestión de usuarios.
 - **SUM Server:** módulo de servidor encargado de tramitar las acciones diferidas de gestión de usuarios generadas en el módulo SUM Web.

La siguiente figura representa la arquitectura del sistema SUM.



4.6 DESCRIPCIÓN DEL SISTEMA

El sistema de gestión de usuarios SUM (*System Users Management*) se encuentra desplegado en el servidor de producción. Se integra dentro de un marco de trabajo establecido en un sistema de gestión de expedientes de telecomunicaciones. Este sistema de gestión de usuarios permite a los usuarios con un rol específico, de administración de usuarios, gestionar los usuarios así como los permisos de acceso de los usuarios del sistema adscritos a su mismo departamento. Permite generar determinadas acciones *on line* y/o diferidas.

Las acciones de gestión se subdivididen en dos grupos:

Acciones *on line*

- Bloqueo / desbloqueo de usuarios
- Cambio de contraseñas
- Asignación de permisos de firma

- Consulta de permisos e información de usuario
- Consulta de información de perfiles
- Consulta de peticiones generadas
- Eliminación de peticiones

Acciones diferidas

- Alta de nuevos usuarios
- Asignación de perfiles predefinidos a usuarios
- Eliminación de usuarios

El sistema de gestión de usuarios SUM consta de dos módulos diferenciados: SUM Web y SUM Server que se detallan a continuación.

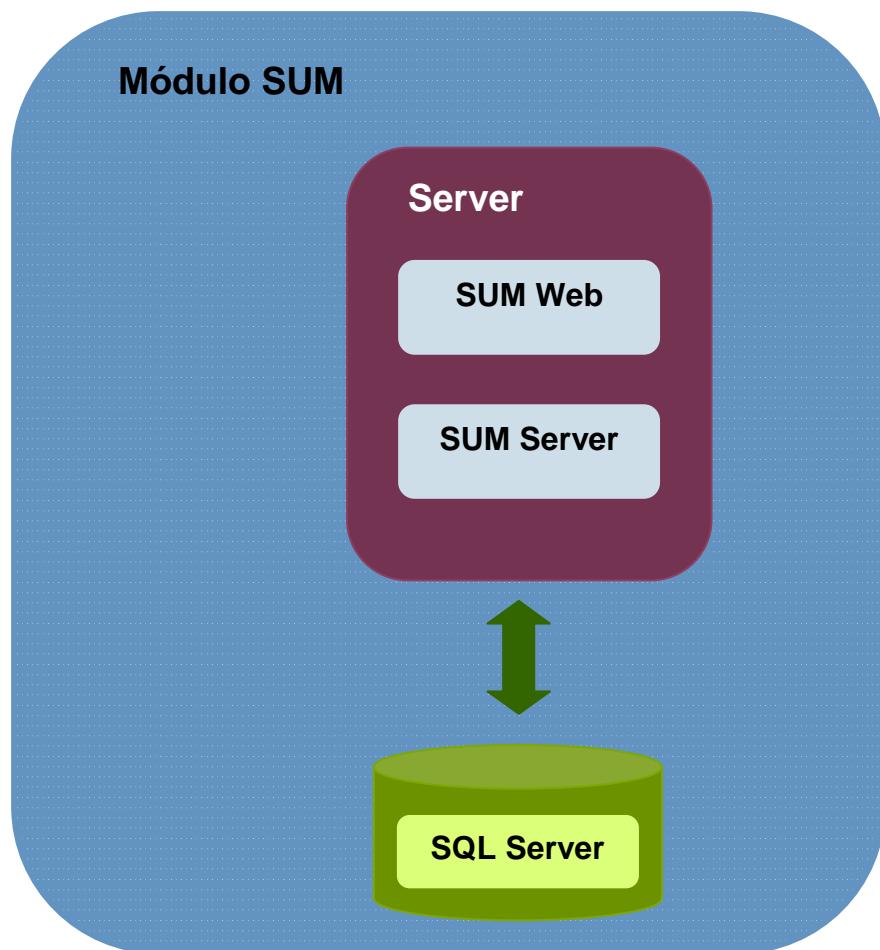


Figura 21. Sistema SUM

4.6.1 Módulo SUM Web

Módulo on-line en arquitectura Web desarrollado en tecnología Microsoft ASP.Net con lenguaje Visual Basic, haciendo uso del sistema de gestión de base de datos SQL Server para el acceso a la información. Para el cifrado de datos se utiliza la función resumen MD5.

El módulo SUM Web se integra en un marco de trabajo establecido, es decir, se accede a una Intranet corporativa y se establece una sesión en un entorno de trabajo concreto que mediante un enlace proporciona acceso a la aplicación Web. Sólo aquellas personas que dispongan del privilegio de administrador de usuarios podrán acceder a la aplicación y gestionar los usuarios de su mismo departamento. Lo normal es que dispongan de este rol los jefes de cada uno de los departamentos de la empresa.

Este módulo será el que implemente la interfaz de usuario del sistema de gestión de usuarios, realice las acciones on-line, genere peticiones de usuario para las acciones diferidas y permita la gestión de peticiones generadas y la consulta de información del departamento.

- **Gestión de usuarios on-line:** grupo compuesto por las operaciones de bloqueo y desbloqueo de usuarios, modificación de contraseñas y asignación de permisos de firma.
- **Gestión de usuarios diferida:** la generación de peticiones de acciones diferidas de creación y eliminación de usuarios así como la asignación de nuevos perfiles a un usuario conforman el núcleo de este grupo.
- **Gestión de peticiones:** este grupo permite consultar información detallada sobre aquellas peticiones generadas por el usuario que tras ser ejecutadas finalizaron con error y ofrece la posibilidad de eliminar cualquier tipo de petición generada haya sido o no ejecutada.
- **Gestión de información:** en este grupo se encuentran las operaciones de consulta de información de perfiles del departamento y consulta de permisos e información de usuario.

4.6.1.1 Módulo SUM Web: diagrama de casos de uso

Los casos de uso describen la funcionalidad del sistema basándose en los requisitos previamente establecidos. En el diagrama de casos de uso se representan los actores y los casos de uso principales del sistema, detallando las relaciones que existen entre ellos.

A continuación se muestra el diagrama de casos de uso del sistema SUM Web.

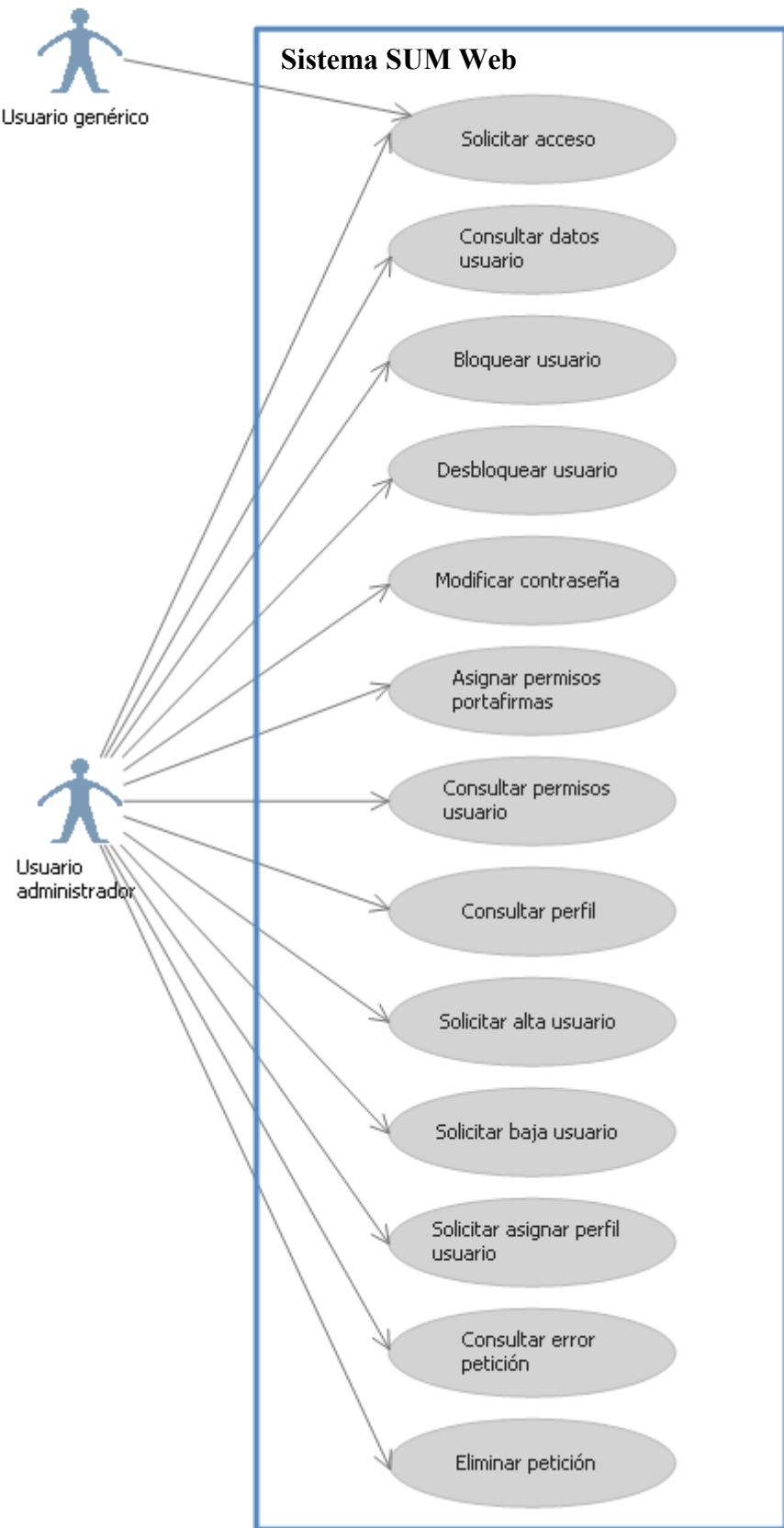


Figura 22. Módulo SUM Web: diagrama de casos de uso

El principal usuario del sistema es el usuario administrador con privilegios para la administración y gestión de los usuarios de su departamento. Una vez validado, este usuario será quien tenga permiso para efectuar las diferentes funcionalidades que ofrece el módulo SUM Web.

Otro posible usuario participante es un usuario genérico sin permisos de administración para la gestión de usuarios. Este tipo de usuario al no disponer de permisos de acceso no será validado por el sistema.

4.6.1.2 Módulo SUM Web: descripción de casos de uso

Este apartado facilita una descripción detallada de los diferentes casos de uso del sistema SUM Web.

UC001 BLOQUEAR USUARIO	
Caso de uso	Bloquear usuario.
Actores	Usuario administrador.
Objetivo	Bloquear un usuario del sistema.
Descripción	Bloquea el usuario seleccionado para impedir su acceso al sistema.
Precondiciones	Usuario seleccionado no bloqueado.
Postcondiciones	Usuario bloqueado.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario selecciona un usuario no bloqueado de la lista de usuarios del departamento. 2. El usuario selecciona la opción bloquear usuario 3. El usuario confirma acción. 4. El sistema modifica a bloqueado el estado del usuario. 5. El sistema incluye una marca indicativa de usuario bloqueado en la lista de usuarios del departamento. 6. El sistema habilita la opción desbloquear usuario.
Flujo alternativo	<p>(3.1). En caso de (cancelar acción) el sistema finaliza la acción sin efectuar cambios.</p> <p>(4.1). En caso de (error de actualización de BD) el sistema muestra un mensaje indicativo.</p>
Requisitos origen	SR005, SR020, SR024, SR025, SR026

Tabla 10. SUM Web. Caso de uso: bloquear usuario

UC002 DESBLOQUEAR USUARIO	
Caso de uso	Desbloquear usuario.
Actores	Usuario administrador.
Objetivo	Desbloquear un usuario del sistema.
Descripción	Desbloquea el usuario seleccionado para permitir su acceso al sistema.

UC002 DESBLOQUEAR USUARIO	
Precondiciones	Usuario seleccionado bloqueado.
Postcondiciones	Usuario desbloqueado.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario selecciona un usuario bloqueado de la lista de usuarios del departamento. 2. El usuario selecciona la opción desbloquear usuario. 3. El usuario confirma acción. 4. El sistema modifica a desbloqueado el estado del usuario. 5. El sistema elimina la marca de usuario bloqueado de la lista de usuarios del departamento. 6. El sistema habilita la opción bloquear usuario.
Flujo alternativo	<p>(3.1). En caso de (cancelar acción) el sistema finaliza la acción sin efectuar cambios.</p> <p>(4.1). En caso de (error de actualización de BD) el sistema muestra un mensaje indicativo.</p>
Requisitos origen	SR005, SR020, SR204, SR026, SR027

Tabla 11. SUM Web. Caso de uso: desbloquear usuario

UC003 MODIFICAR CONTRASEÑA	
Caso de uso	Modificar contraseña.
Actores	Usuario administrador.
Objetivo	Modificar contraseña de usuario.
Descripción	Modifica la contraseña del usuario seleccionado.
Precondiciones	Usuario seleccionado.
Postcondiciones	Contraseña modificada.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario selecciona un usuario de la lista de usuarios del departamento. 2. El usuario selecciona la opción modificar contraseña. 3. El sistema muestra el formulario específico para modificar contraseña. 4. El usuario introduce la nueva contraseña. 5. El usuario confirma la nueva contraseña. 6. El usuario confirma acción. 7. El sistema cifra la nueva contraseña. 8. El sistema modifica la contraseña del usuario.
Flujo alternativo	<p>(5.1) En caso de (contraseña no válida) el sistema muestra un mensaje indicativo.</p> <p>(6.1). En caso de (cancelar acción) el sistema finaliza la acción sin efectuar cambios.</p> <p>(8.1). En caso de (coincidir el usuario seleccionado con el usuario de acceso) tras modificar contraseña, el sistema actualizará la información de sesión del usuario en curso.</p> <p>(8.2). En caso de (error de actualización) el sistema muestra un mensaje indicativo.</p>

UC003 MODIFICAR CONTRASEÑA

Requisitos origen	SR005, SR028, SR039
--------------------------	---------------------

Tabla 12. SUM Web. Caso de uso: modificar contraseña**UC004 ASIGNAR PERMISOS PORTAFIRMAS**

Caso de uso	Asignar permisos portafirmas.
Actores	Usuario administrador.
Objetivo	Asignar permisos de portafirmas a un usuario.
Descripción	Asignar, eliminar o modificar los permisos de portafirmas de usuario.
Precondiciones	Usuario seleccionado.
Postcondiciones	Permisos de portafirmas asignados a usuario.
Flujo normal	<ol style="list-style-type: none"> El usuario selecciona un usuario de la lista de usuarios del departamento. El usuario selecciona la opción asignar permisos portafirmas. El sistema muestra el formulario específico para asignar permisos portafirmas con los permisos de portafirmas asignados al departamento, marcando aquellos que el usuario seleccionado tiene asignados. El usuario efectúa una nueva selección de permisos. El usuario confirma acción. El sistema registra los nuevos permisos de portafirmas del usuario.
Flujo alternativo	<p>(3.1). En caso de (no existir permisos de portafirmas asignados al departamento) el sistema muestra un mensaje indicativo.</p> <p>(5.1). En caso de (cancelar acción) el sistema finaliza la acción sin efectuar cambios.</p> <p>(6.1). En caso de (error de actualización) el sistema muestra un mensaje indicativo.</p>
Requisitos origen	SR005, SR029

Tabla 13. SUM Web. Caso de uso: asignar permisos portafirmas**CU005 CONSULTAR PERMISOS USUARIO**

Caso de uso	Consultar permisos usuario.
Actores	Usuario administrador.
Objetivo	Consultar permisos de usuario.
Descripción	Mostrar los permisos que tiene asignados un usuario.
Precondiciones	Usuario seleccionado.
Postcondiciones	Permisos de usuario mostrados.

CU005 CONSULTAR PERMISOS USUARIO	
Flujo normal	<ol style="list-style-type: none"> 1. El usuario selecciona un usuario de la lista de usuarios del departamento. 2. El usuario selecciona la opción consultar permisos usuario. 3. El sistema muestra un listado con nombre y descripción de los permisos asignados al usuario.
Flujo alternativo	(3.1). En caso de (no existir permisos asignados al usuario) el sistema muestra un mensaje indicativo.
Requisitos origen	SR005, SR030

Tabla 14. SUM Web. Caso de uso: consultar permisos usuario

UC006 CONSULTAR PERFIL	
Caso de uso	Consultar perfil.
Actores	Usuario administrador.
Objetivo	Consultar datos de un perfil.
Descripción	Mostrar los permisos que otorga un perfil específico.
Precondiciones	Perfil seleccionado.
Postcondiciones	Datos de perfil mostrados.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario selecciona un perfil de la lista de perfiles del departamento. 2. El sistema muestra la descripción del perfil así como un listado con nombre y descripción de los permisos que otorga. 3. El sistema activa la acción de asignar perfil (sólo si hay también un usuario seleccionado).
Flujo alternativo	(1.1). En caso de (existir un perfil previamente seleccionado) el sistema deseleccionará el perfil que estuviese previamente seleccionado.
Requisitos origen	SR005, SR017, SR023, SR037

Tabla 15. SUM Web. Caso de uso: consultar perfil

UC007 SOLICITAR ALTA USUARIO	
Caso de uso	Solicitar alta usuario.
Actores	Usuario administrador.
Objetivo	Crear una solicitud diferida de alta de usuario en el sistema.
Descripción	Crea una solicitud diferida para el alta de un nuevo usuario del departamento.
Precondiciones	
Postcondiciones	Usuario creado.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción nuevo usuario.

UC007 SOLICITAR ALTA USUARIO

2. El sistema muestra el formulario específico de nuevo usuario.
3. El usuario introduce el nombre del usuario a crear.
4. El sistema valida que el nuevo usuario exista en el dominio de la empresa y muestra su nombre completo. Valida que no haya sido creado previamente en el sistema y valida que no exista ninguna petición de alta pendiente para el nuevo usuario.
5. El usuario introduce contraseña.
6. El usuario confirma la nueva contraseña.
7. El usuario confirma acción.
8. El sistema cifra la contraseña del nuevo usuario.
9. El sistema obtiene la contraseña cifrada del usuario solicitante.
10. El sistema crea una tarea programada de alta de usuario estableciendo los siguientes valores:
 - Identificador de usuario solicitante
 - Estado de tarea programada = PDT
 - Fecha ejecución = fecha de inicio de ejecución
 - Fecha estado = fecha actual
 - Fecha de creación = fecha actual
 - Fecha de actualización = fecha actual
 - Argumentos en el siguiente orden:
 - Acción a ejecutar = N
 - Nombre de usuario solicitante
 - Contraseña cifrada de usuario solicitante
 - Nombre de nuevo usuario
 - Nombre completo de nuevo usuario
 - Contraseña cifrada de nuevo usuario
11. El sistema crea una petición y la asocia a la tarea programada previamente generada. Será necesario establecer los siguientes valores al crear la petición:
 - Identificador de usuario solicitante
 - Identificador de departamento del usuario solicitante
 - Nombre de nuevo usuario
 - Acción a ejecutar = N
 - Argumentos en el siguiente orden:
 - Nombre completo de nuevo usuario
 - Contraseña cifrada de nuevo usuario
 - Estado de petición = PDT
 - Identificador de tarea programada asociada
 - Visibilidad = S
 - Fecha de actualización = fecha actual
 - Fecha de creación = fecha actual
12. El sistema actualiza la lista de peticiones.
13. El sistema actualiza la lista de usuarios pendientes de alta.

UC007 SOLICITAR ALTA USUARIO	
Flujo alternativo	<p>(4.1). En caso de (no existir el usuario en el dominio de la empresa) el sistema muestra un mensaje indicativo.</p> <p>(4.2). En caso de (existir el usuario en el sistema) el sistema muestra un mensaje indicativo.</p> <p>(4.3). En caso de (existir una petición de alta pendiente para el nuevo usuario) el sistema muestra un mensaje indicativo.</p> <p>(6.1). En caso de (contraseña no válida) el sistema muestra un mensaje indicativo.</p> <p>(7.1). En caso de (cancelar acción) el sistema finaliza la acción sin efectuar cambios.</p> <p>(10.1). En caso de (error al generar tarea programada) el sistema muestra un mensaje indicativo.</p> <p>(11.1). En caso de (error al generar petición) el sistema muestra un mensaje indicativo y elimina la tarea asociada.</p>
Requisitos origen	SR005, SR006, SR007, SR008, SR010, SR011, SR012, SR018, SR019, SR039

Tabla 16. SUM Web. Caso de uso: solicitar alta usuario

UC008 SOLICITAR BAJA USUARIO	
Caso de uso	Solicitar baja usuario.
Actores	Usuario administrador.
Objetivo	Crear una solicitud diferida de eliminación de usuario del sistema.
Descripción	Crea una solicitud diferida para eliminar un usuario del departamento.
Precondiciones	Usuario seleccionado.
Postcondiciones	Usuario eliminado.
Flujo normal	<ol style="list-style-type: none"> El usuario selecciona un usuario de la lista de usuarios del departamento o la lista de usuarios pendientes de alta. El usuario selecciona la opción eliminar usuario. El usuario confirma acción. El sistema valida que no existe una solicitud pendiente. El sistema incluye una marca de usuario pendiente de eliminación en la lista de usuarios del departamento. El sistema obtiene la contraseña cifrada del usuario solicitante. El sistema crea una tarea programada de eliminación de usuario estableciendo los siguientes valores: <ul style="list-style-type: none"> ○ Identificador de usuario solicitante ○ Estado de tarea programada = PDT ○ Fecha ejecución = fecha de inicio de ejecución ○ Fecha estado = fecha actual ○ Fecha de creación = fecha actual ○ Fecha de actualización = fecha actual ○ Argumentos en el siguiente orden:

UC008 SOLICITAR BAJA USUARIO	
	<ul style="list-style-type: none"> ▪ Acción a ejecutar = D ▪ Nombre de usuario solicitante ▪ Contraseña cifrada de usuario solicitante ▪ Nombre de usuario a eliminar <p>8. El sistema crea una petición y la asocia a la tarea programada previamente generada. Será necesario establecer los siguientes valores al crear la petición:</p> <ul style="list-style-type: none"> ○ Identificador de usuario solicitante ○ Identificador de departamento del usuario solicitante ○ Nombre de usuario a eliminar ○ Acción a ejecutar = D ○ Estado de petición = PDT ○ Identificador de tarea programada asociada ○ Visibilidad = S ○ Fecha de actualización = fecha actual ○ Fecha de creación = fecha actual <p>9. El sistema actualiza la lista de peticiones.</p>
Flujo alternativo	<p>(3.1). En caso de (cancelar acción) el sistema finaliza la acción sin efectuar cambios.</p> <p>(4.1). En caso de (que el usuario ya tenga una marca de usuario eliminado) el sistema no continúa el proceso y debe de mostrar un mensaje indicativo puesto que ya existe una petición de eliminación de ese usuario.</p> <p>(4.2). En caso de (eliminar un usuario pendiente de alta) el sistema actualiza la lista de usuarios pendientes de alta y elimina la petición pendiente de alta del usuario así como las peticiones pendientes del usuario a eliminar y no continúa el proceso.</p> <p>(7.1). En caso de (error al generar tarea programada) el sistema muestra un mensaje indicativo.</p> <p>(8.1). En caso de (error al generar petición) el sistema muestra un mensaje indicativo y elimina la tarea asociada.</p>
Requisitos origen	SR005, SR006, SR007, SR008, SR010, SR011, SR012, SR020, SR031, SR032, SR039

Tabla 17. SUM Web. Caso de uso: solicitar baja usuario

UC009 SOLICITAR ASIGNAR PERFIL USUARIO	
Caso de uso	Solicitar asignar perfil usuario.
Actores	Usuario administrador.
Objetivo	Crear una solicitud diferida de asignación de perfil a un usuario del sistema.
Descripción	Crea una solicitud diferida para asignar un perfil específico a un usuario del departamento.

UC009 SOLICITAR ASIGNAR PERFIL USUARIO	
Precondiciones	Usuario y perfil seleccionados.
Postcondiciones	Perfil asignado a un usuario.
Flujo normal	<p>1. El usuario selecciona un usuario de la lista de usuarios del departamento o la lista de usuarios pendiente de alta.</p> <p>2. El usuario selecciona un perfil de la lista de perfiles del departamento.</p> <p>3. El usuario selecciona la opción asignar perfil.</p> <p>4. El usuario confirma acción.</p> <p>5. El sistema comprueba que no exista una petición pendiente de las mismas características.</p> <p>6. El sistema obtiene la contraseña cifrada del usuario solicitante.</p> <p>7. El sistema crea una tarea programada de asignación de perfil a usuario estableciendo los siguientes valores:</p> <ul style="list-style-type: none"> o Identificador de usuario solicitante o Estado de tarea programada = PDT o Fecha ejecución = fecha de inicio de ejecución o Fecha estado = fecha actual o Fecha de creación = fecha actual o Fecha de actualización = fecha actual o Argumentos en el siguiente orden: <ul style="list-style-type: none"> ▪ Acción a ejecutar = P ▪ Nombre de usuario solicitante ▪ Contraseña cifrada de usuario solicitante ▪ Nombre de usuario de asignación ▪ Identificador de perfil a asignar ▪ <i>True</i> <p>8. El sistema crea una petición y la asocia a la tarea programada previamente generada. Será necesario establecer los siguientes valores al crear la petición:</p> <ul style="list-style-type: none"> o Identificador de usuario solicitante o Identificador de departamento del usuario solicitante o Nombre de usuario de asignación de perfil o Acción a ejecutar = P o Argumentos en el siguiente orden: <ul style="list-style-type: none"> ▪ Identificador de perfil a asignar ▪ <i>True</i> o Estado de petición = PDT o Identificador de tarea programada asociada o Visibilidad = S o Fecha de actualización = fecha actual o Fecha de creación = fecha actual <p>9. El sistema actualiza la lista de peticiones.</p>
Flujo alternativo	(4.1). En caso de (cancelar acción) el sistema finaliza la acción sin efectuar cambios.

UC009 SOLICITAR ASIGNAR PERFIL USUARIO	
	<p>(4.2). En caso de (asignar un perfil de tipo especial) el usuario debe confirmar si desea mantener los permisos del usuario y el sistema asocia la respuesta de la confirmación como argumento en la correspondiente petición y tarea programada sustituyendo el valor <i>True</i> indicado en el flujo normal.</p> <p>(5.1). En caso de (existir una petición pendiente de las mismas características) el sistema no continúa el proceso y debe de mostrar un mensaje indicativo.</p> <p>(7.1). En caso de (error al generar tarea programada) el sistema muestra un mensaje indicativo.</p> <p>(8.1). En caso de (error al generar petición) el sistema muestra un mensaje indicativo y elimina la tarea asociada.</p>
Requisitos origen	SR005, SR006, SR007, SR008, SR010, SR011, SR012, SR020, SR021, SR022, SR023, SR039

Tabla 18. SUM Web. Caso de uso: solicitar asignar perfil usuario

UC010 CONSULTAR ERROR PETICIÓN	
Caso de uso	Consultar error petición.
Actores	Usuario administrador.
Objetivo	Consultar información de petición.
Descripción	Consulta de información de una petición del departamento.
Precondiciones	Petición en estado de error seleccionada.
Postcondiciones	Información de petición mostrada.
Flujo normal	<ol style="list-style-type: none"> El usuario selecciona una petición de la lista de peticiones del departamento. El usuario selecciona la opción consultar error. El sistema muestra la descripción de error de ejecución de la tarea asociada a la petición seleccionada.
Flujo alternativo	
Requisitos origen	SR015

Tabla 19. SUM Web. Caso de uso: consultar error petición

UC011 ELIMINAR PETICIÓN	
Caso de uso	Eliminar petición.
Actores	Usuario administrador.
Objetivo	Eliminar una petición del sistema.
Descripción	Eliminación de una petición del departamento.
Precondiciones	Petición seleccionada.
Postcondiciones	Petición eliminada.
Flujo normal	<ol style="list-style-type: none"> El usuario selecciona una petición de la lista de peticiones

UC011 ELIMINAR PETICIÓN	
	<p>del departamento.</p> <ol style="list-style-type: none"> 2. El usuario selecciona la opción eliminar petición. 3. El usuario confirmar acción. 4. El sistema elimina la petición seleccionada, elimina la tarea programada asociada a la petición eliminada y actualiza la lista de peticiones.
Flujo alternativo	<p>(3.1). En caso de (cancelar acción) el sistema finaliza la acción sin efectuar cambios.</p> <p>(4.1). En caso de (eliminar una petición de eliminación de usuario del departamento) además el sistema elimina la marca de usuario pendiente de eliminación de la lista de usuarios del departamento.</p> <p>(4.2). En caso de (eliminar una petición de alta de nuevo usuario) el sistema eliminará todas las peticiones pendientes relativas al nuevo usuario, sus tareas programadas asociadas y actualizará la lista de nuevos usuarios eliminando el usuario de ella.</p>
Requisitos origen	SR014, SR020, SR033, SR038

Tabla 20. SUM Web. Caso de uso: eliminar petición

UC012 SOLICITAR ACCESO	
Caso de uso	Solicitar acceso.
Actores	Usuario genérico y usuario administrador.
Objetivo	Controlar el acceso al sistema, permitiendo que accedan exclusivamente aquellos usuarios con privilegios de administración para la gestión de usuarios.
Descripción	Control de acceso al sistema.
Precondiciones	El usuario solicita acceso desde el sistema global de generación de expedientes.
Postcondiciones	Permitir o denegar el acceso al sistema al usuario solicitante.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario solicita acceso a la aplicación. 2. El sistema valida el permiso de acceso del usuario. 3. El sistema permite el acceso a la aplicación al usuario administrador e inicializa la ventana principal con los siguientes valores: <ul style="list-style-type: none"> o Lista de usuarios del departamento ordenada alfabéticamente. Se marcarán con el símbolo correspondiente los usuarios bloqueados y pendientes de eliminar. No se incluirán en la lista los usuarios especiales. o Lista de usuarios pendientes de alta en el departamento. o Lista de perfiles del departamento. o Lista de peticiones visibles del departamento y

UC012 SOLICITAR ACCESO	
	<ul style="list-style-type: none"> botones de eliminar petición y consultar petición desactivados. ○ Botones desactivados (bloquear, desbloquear, modificar contraseña, asignar permisos portafirmas, consultar permisos, eliminar usuario y asignar perfil). ○ Botón nuevo usuario activado. ○ Título con nombre del departamento del usuario. ○ No hay elementos seleccionados.
Flujo alternativo	<p>(2.1). En caso de (no tener el usuario permiso de acceso a la aplicación) el sistema no permite el acceso y muestra un mensaje indicativo.</p> <p>(3.1). Si no existen peticiones visibles el sistema muestra un mensaje indicativo y no muestra los botones relativos a eliminar y consultar petición.</p>
Requisitos origen	SR001, SR002, SR003, SR004, SR005, SR011, SR013, SR014, SR015, SR023, SR024, SR025, SR026, SR027, SR028, SR034, SR035, SR040

Tabla 21. SUM Web. Caso de uso: solicitar acceso

UC013 CONSULTAR DATOS USUARIO	
Caso de uso	Consultar datos usuario.
Actores	Usuario administrador.
Objetivo	Consultar información de usuario.
Descripción	Mostrar información de usuario.
Precondiciones	Usuario seleccionado.
Postcondiciones	Información de usuario mostrada.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario selecciona un usuario de la lista de usuarios del departamento. 2. El sistema muestra el nombre completo del usuario seleccionado e información del perfil que tiene asignado el usuario. 3. El sistema habilita las opciones: <ul style="list-style-type: none"> ○ Bloquear (si el usuario está bloqueado) ○ Desbloquear (si el usuario está desbloqueado) ○ Modificar contraseña ○ Asignar permisos portafirmas ○ Consultar permisos ○ Nuevo usuario ○ Eliminar usuario ○ Asignar perfil (sólo si hay también un perfil seleccionado)
Flujo alternativo	(1.1). En caso de (existir un usuario del departamento o

UC013 CONSULTAR DATOS USUARIO	
	pendiente de alta seleccionado) el sistema deseleccionará el usuario que estuviese previamente seleccionado.
Requisitos origen	SR005, SR016, SR023, SR024, SR036

Tabla 22. SUM Web. Caso de uso: consultar datos usuario

4.6.1.3 Módulo SUM Web: diagramas de secuencia

Este apartado representa los diagramas de secuencia del módulo SUM Web. Cada diagrama de secuencia muestra la secuencia de comunicación mediante mensajes entre objetos necesarios para la realización de un caso de uso.

Cada diagrama se identifica por un código secuencial único y un texto descriptivo de la acción implementada.

SDSWXXX → Diagrama de secuencia XXX del módulo SUM Web

SDSW001: Bloquear usuario

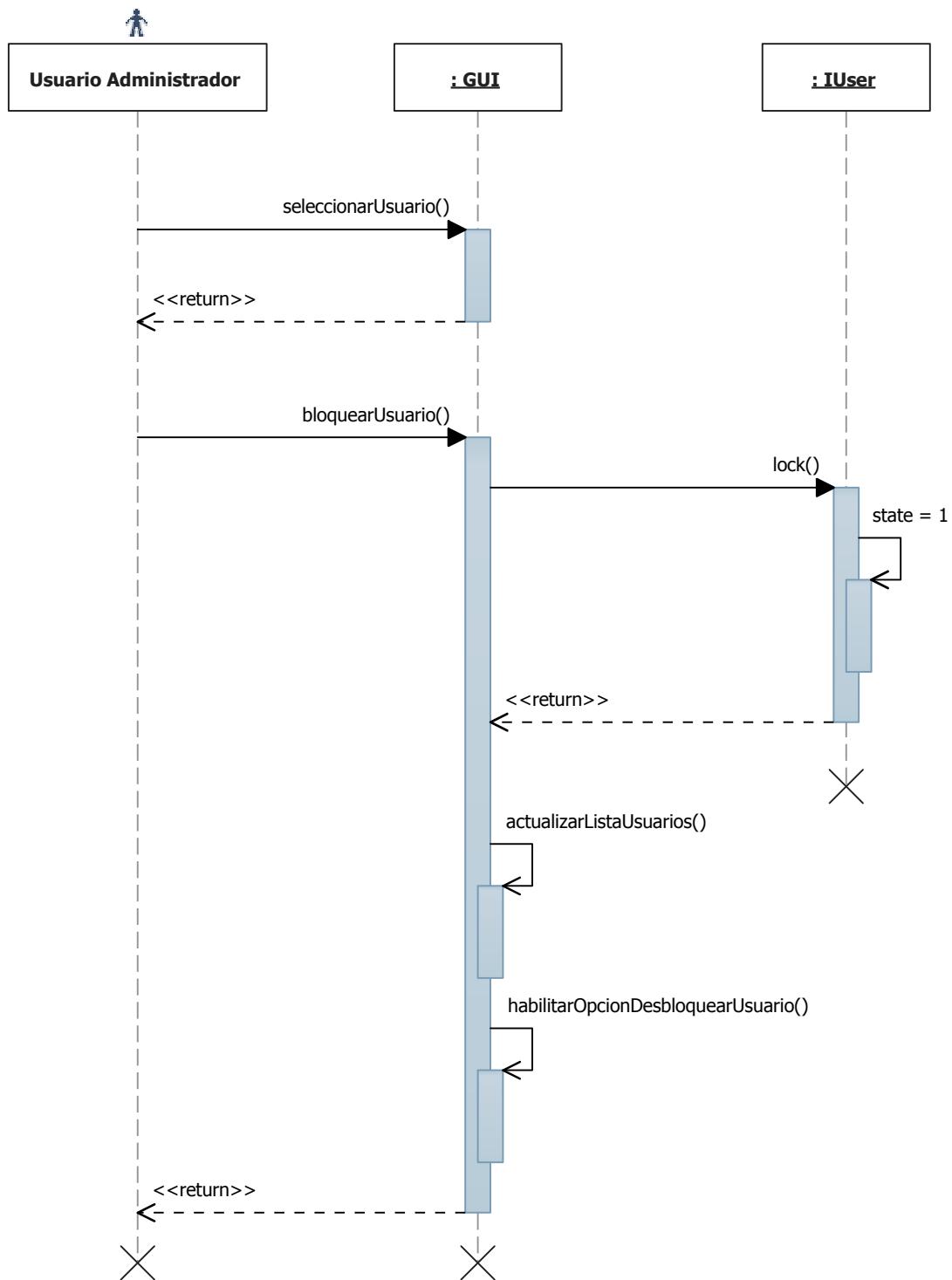
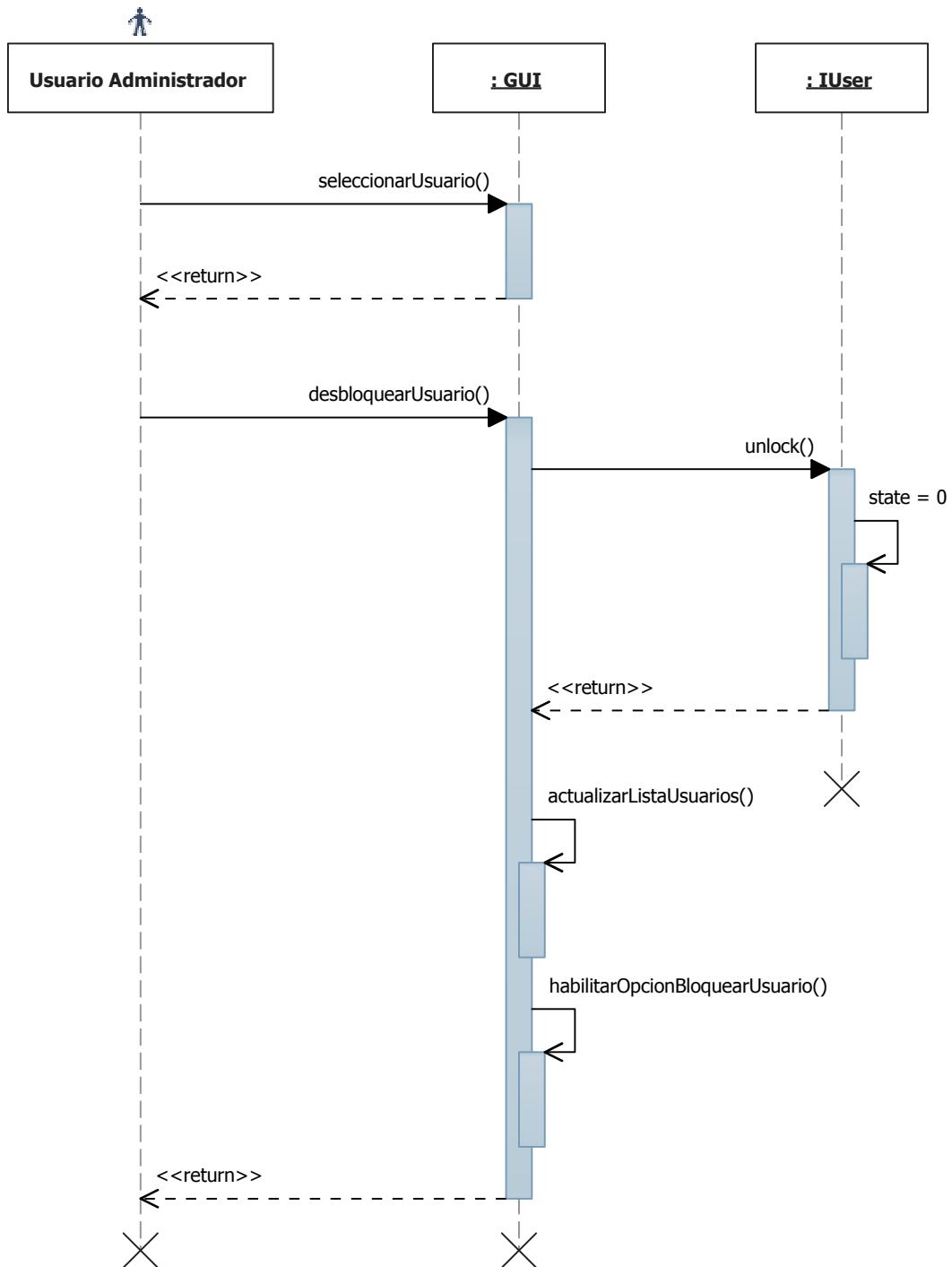


Figura 23. Módulo SUM Web: diagrama de secuencia bloquear usuario

SDSW002: Desbloquear usuario**Figura 24. Módulo SUM Web: diagrama de secuencia desbloquear usuario**

SDSW003: Modificar contraseña

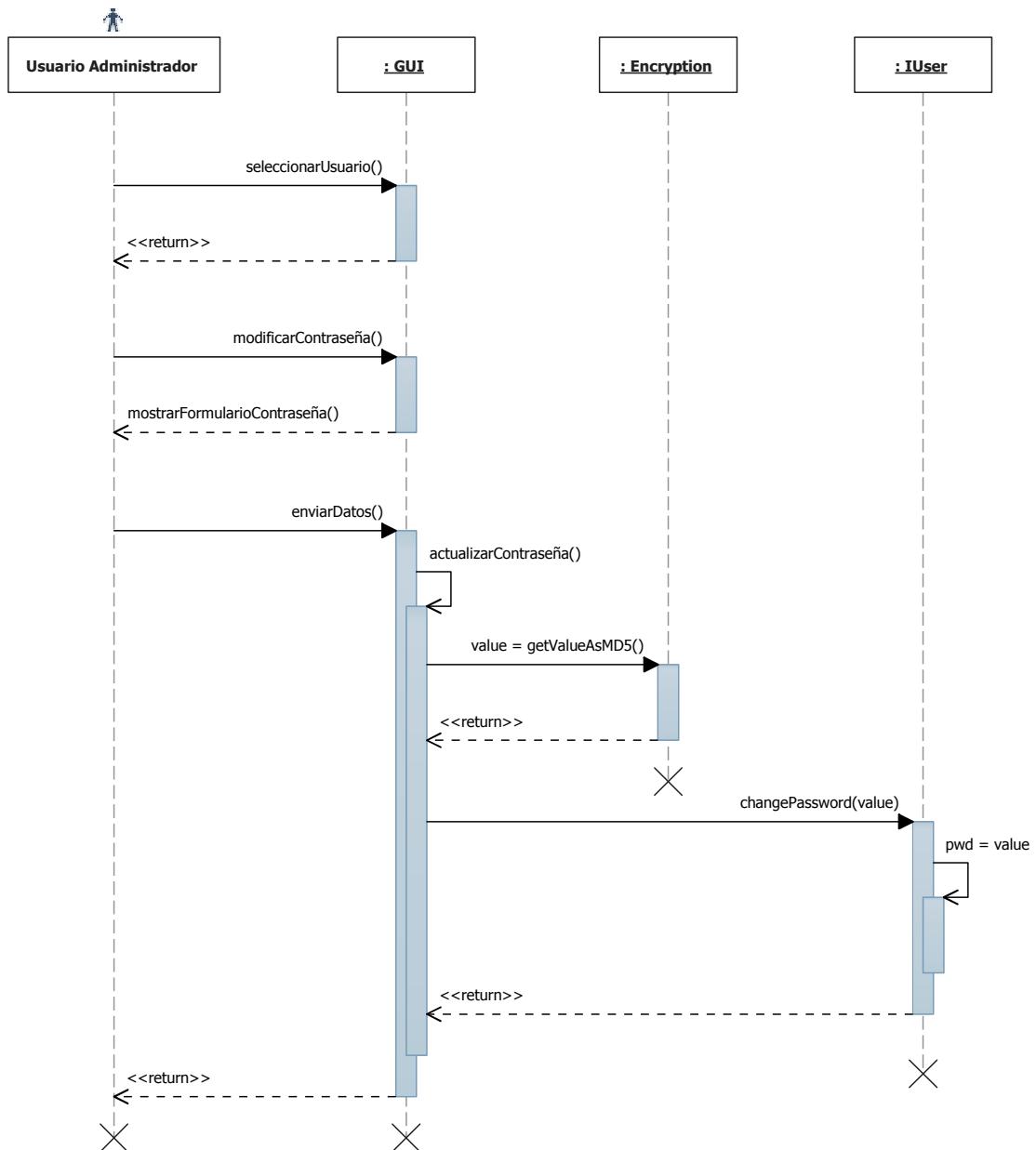


Figura 25. Módulo SUM Web: diagrama de secuencia modificar contraseña

SDSW004: Asignar permisos portafirmas

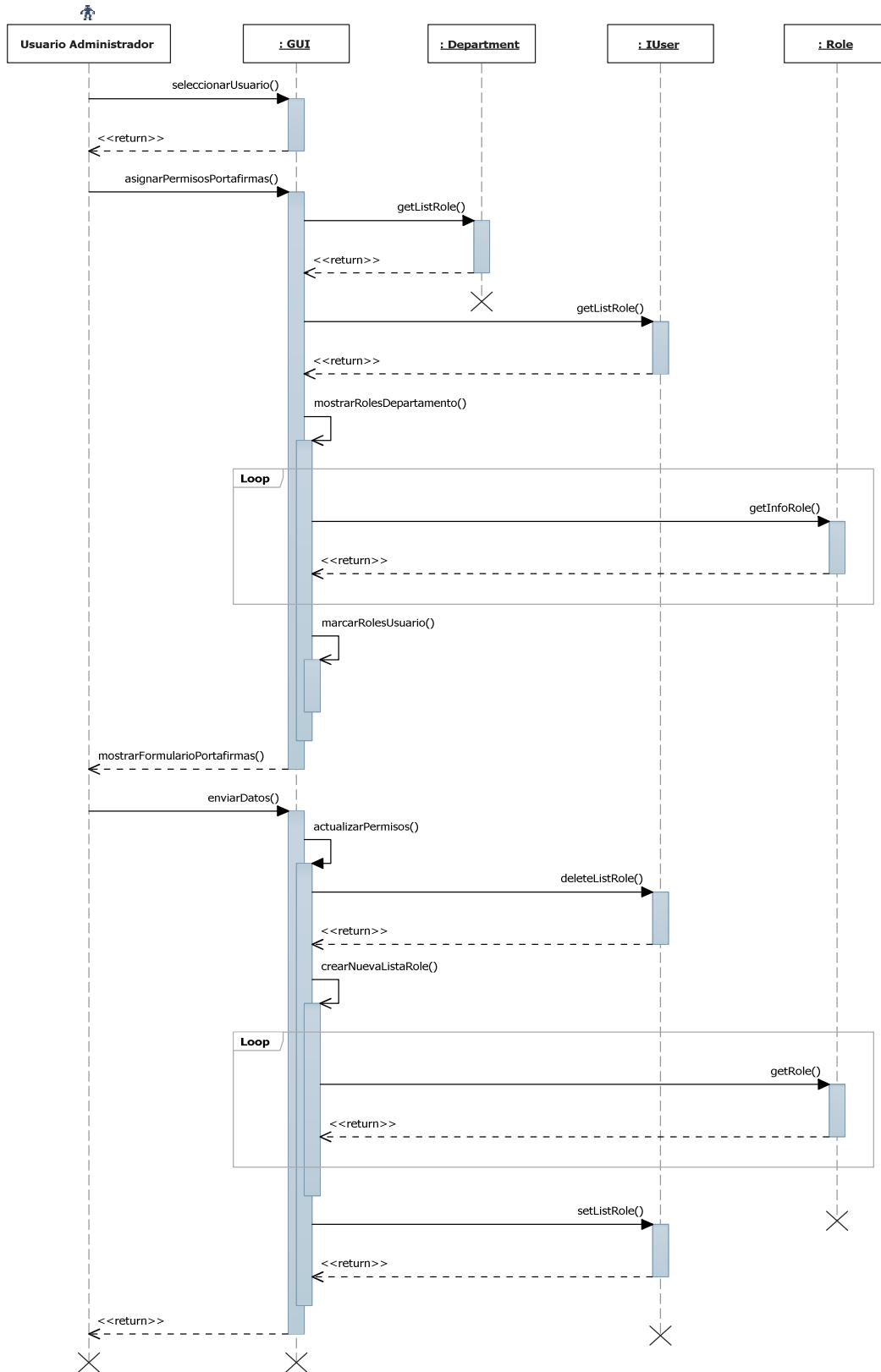


Figura 26. Módulo SUM Web: diagrama de secuencia asignar permisos portafirmas

SDSW005: Consultar permisos usuario

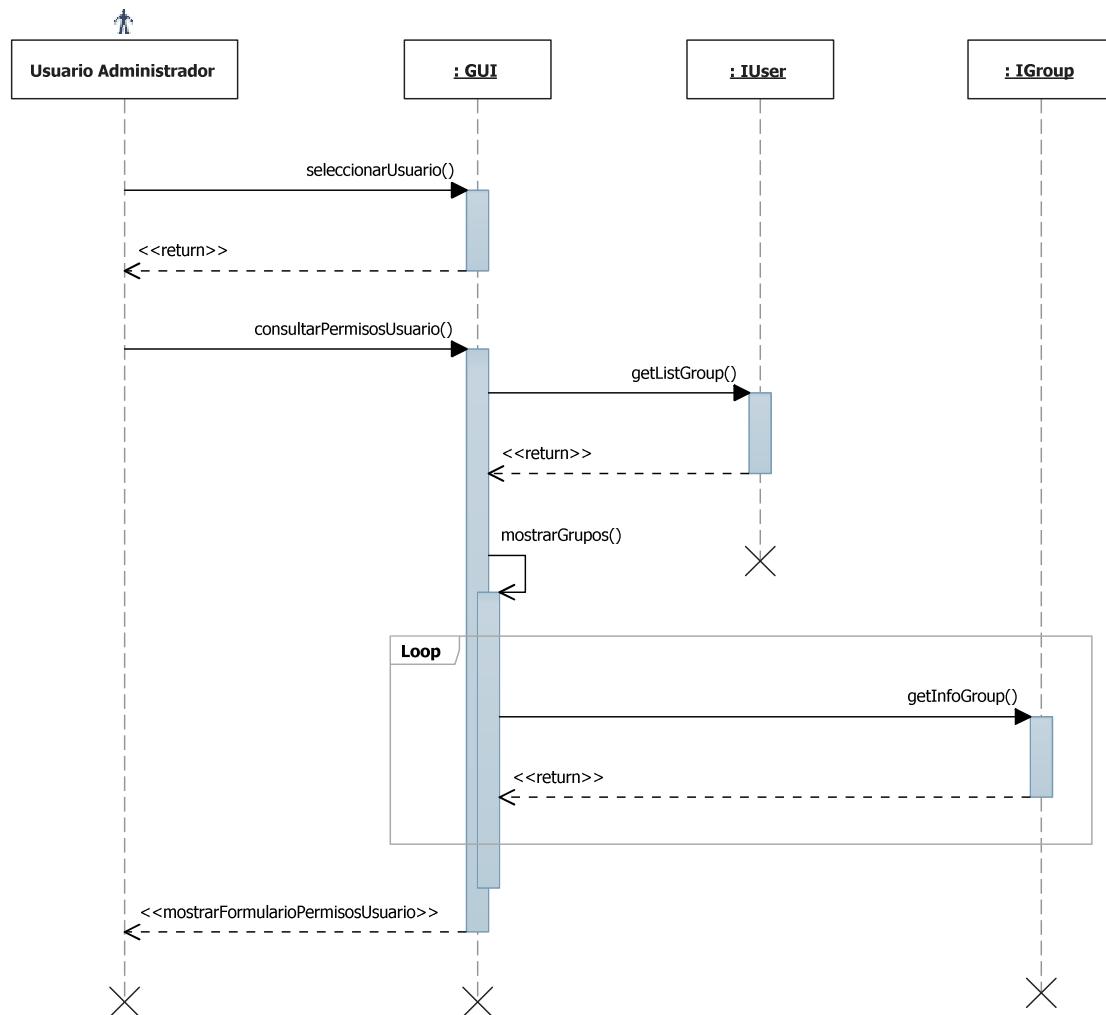


Figura 27. Módulo SUM Web: diagrama de secuencia consultar permisos usuario

SDSW006: Consultar perfil

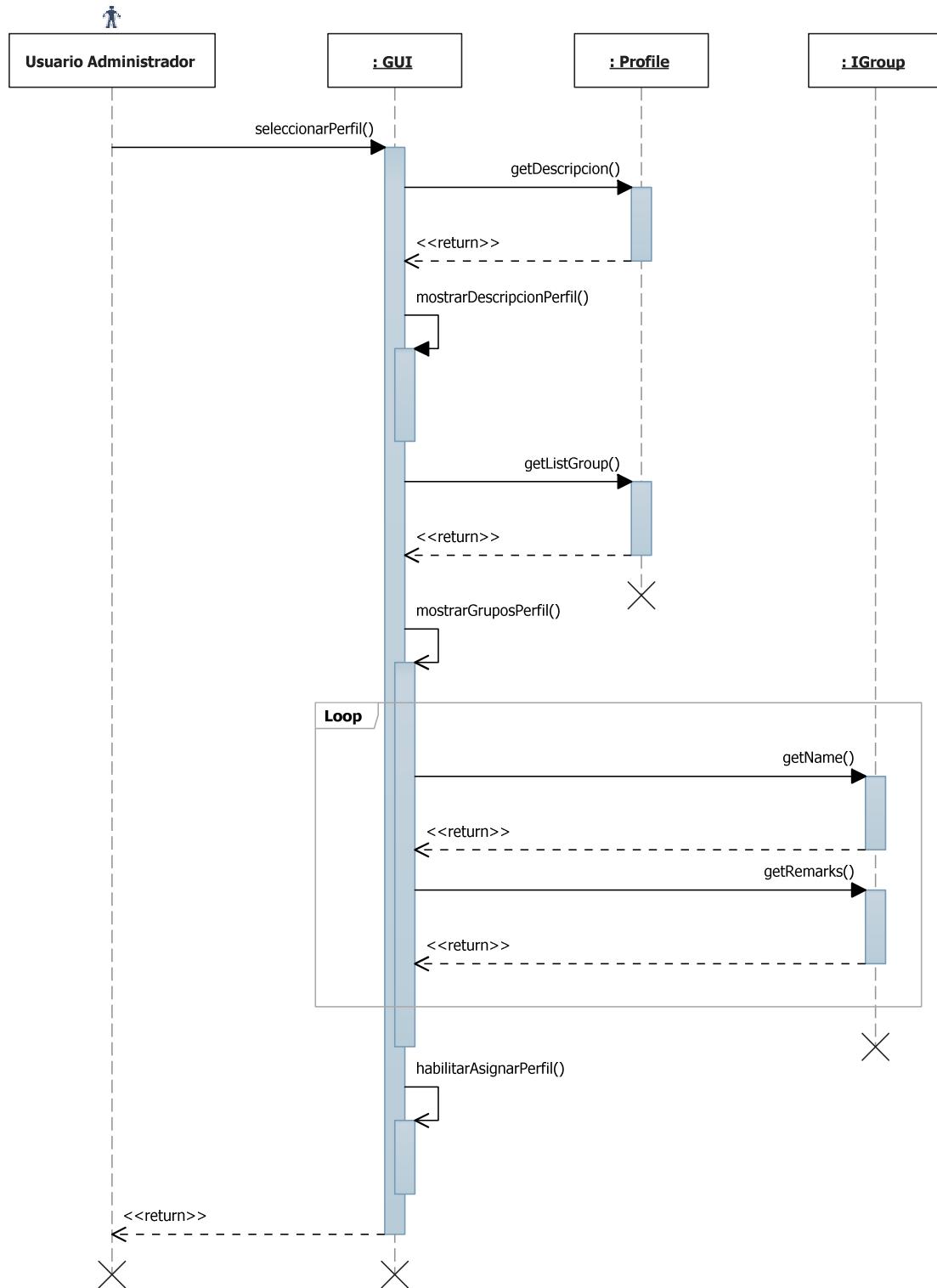


Figura 28. Módulo SUM Web: diagrama de secuencia consultar perfil

SDSW007: Solicitar alta usuario

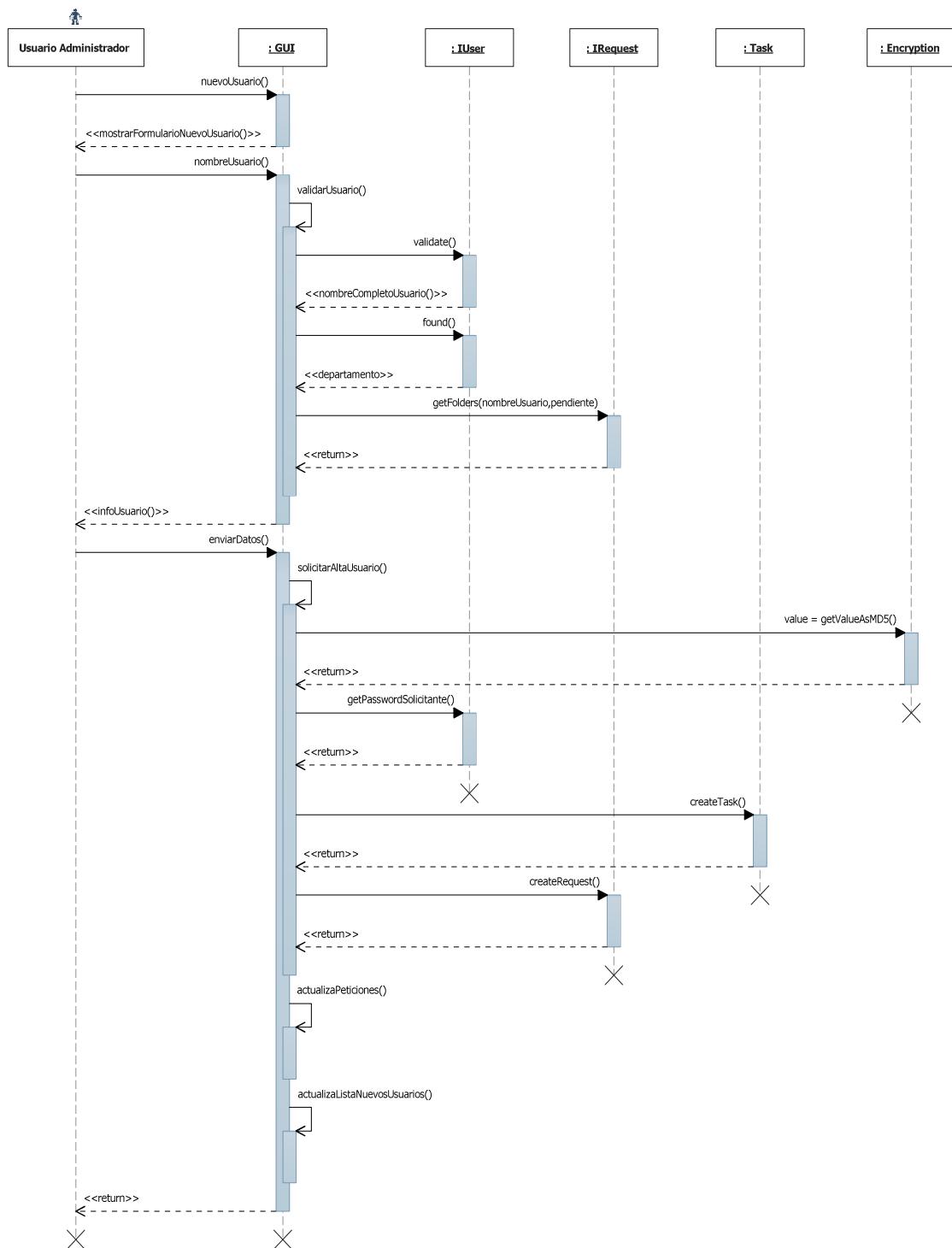


Figura 29. Módulo SUM Web: diagrama de secuencia solicitar alta usuario

SDSW008: Solicitar baja usuario

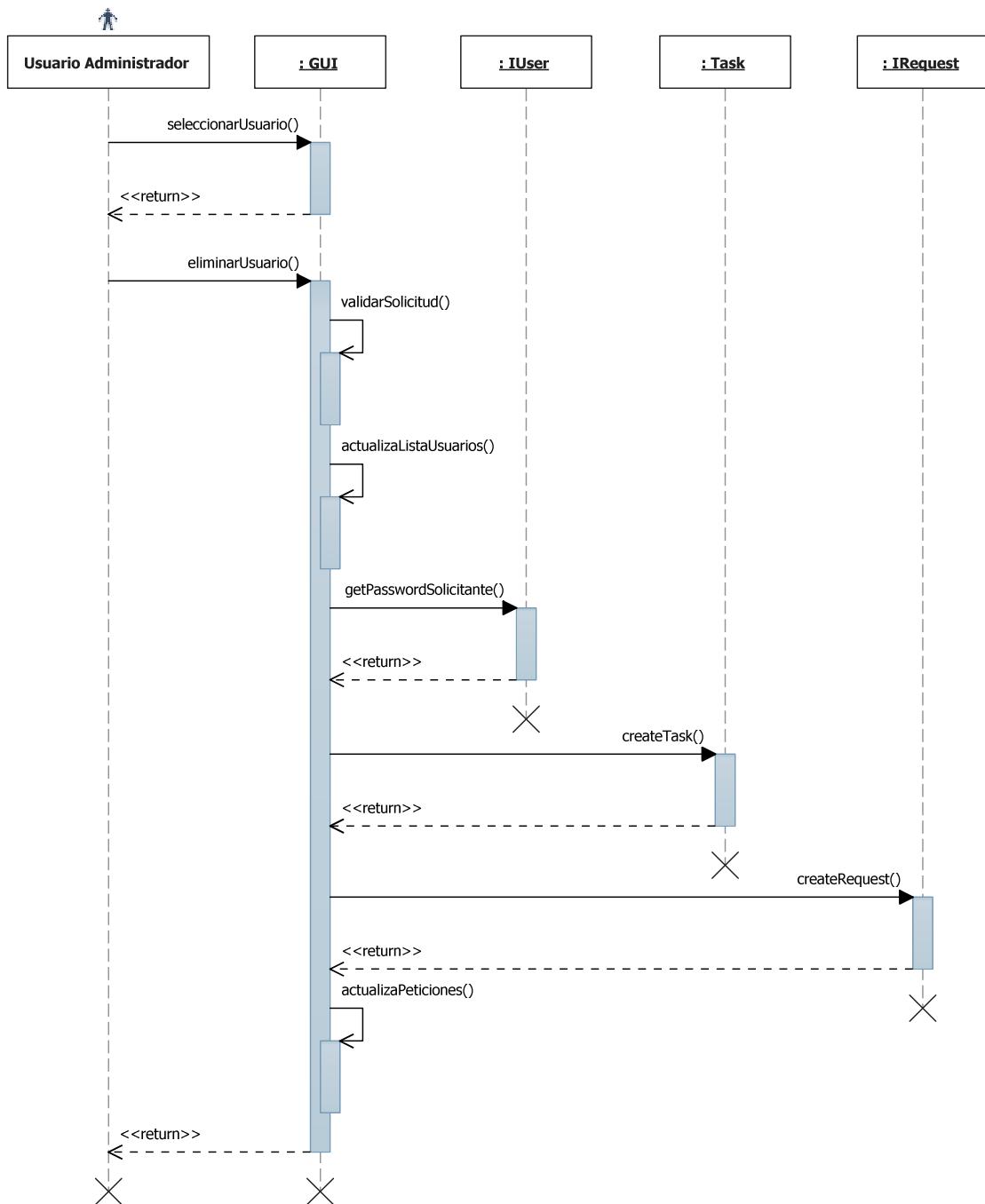
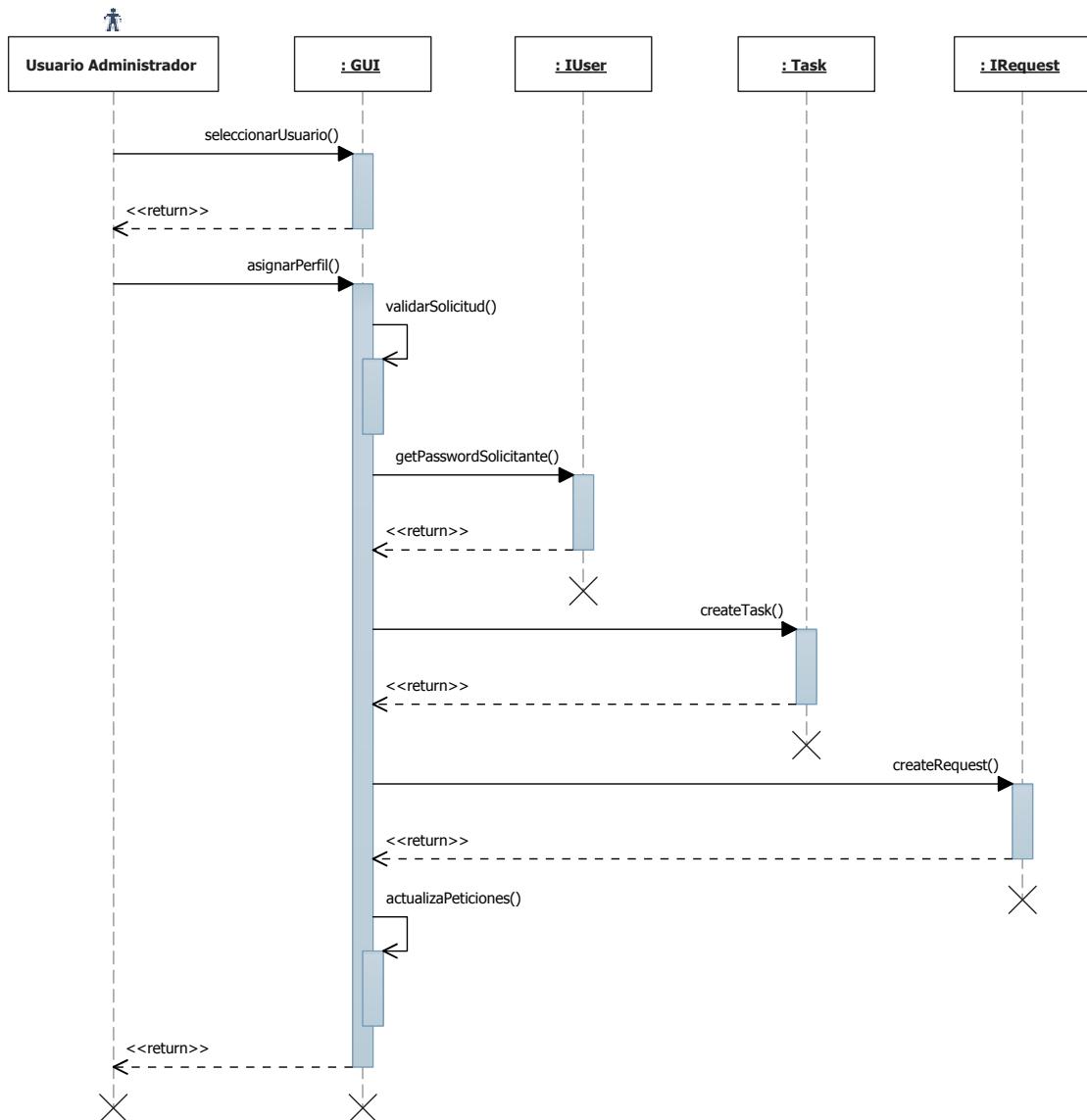
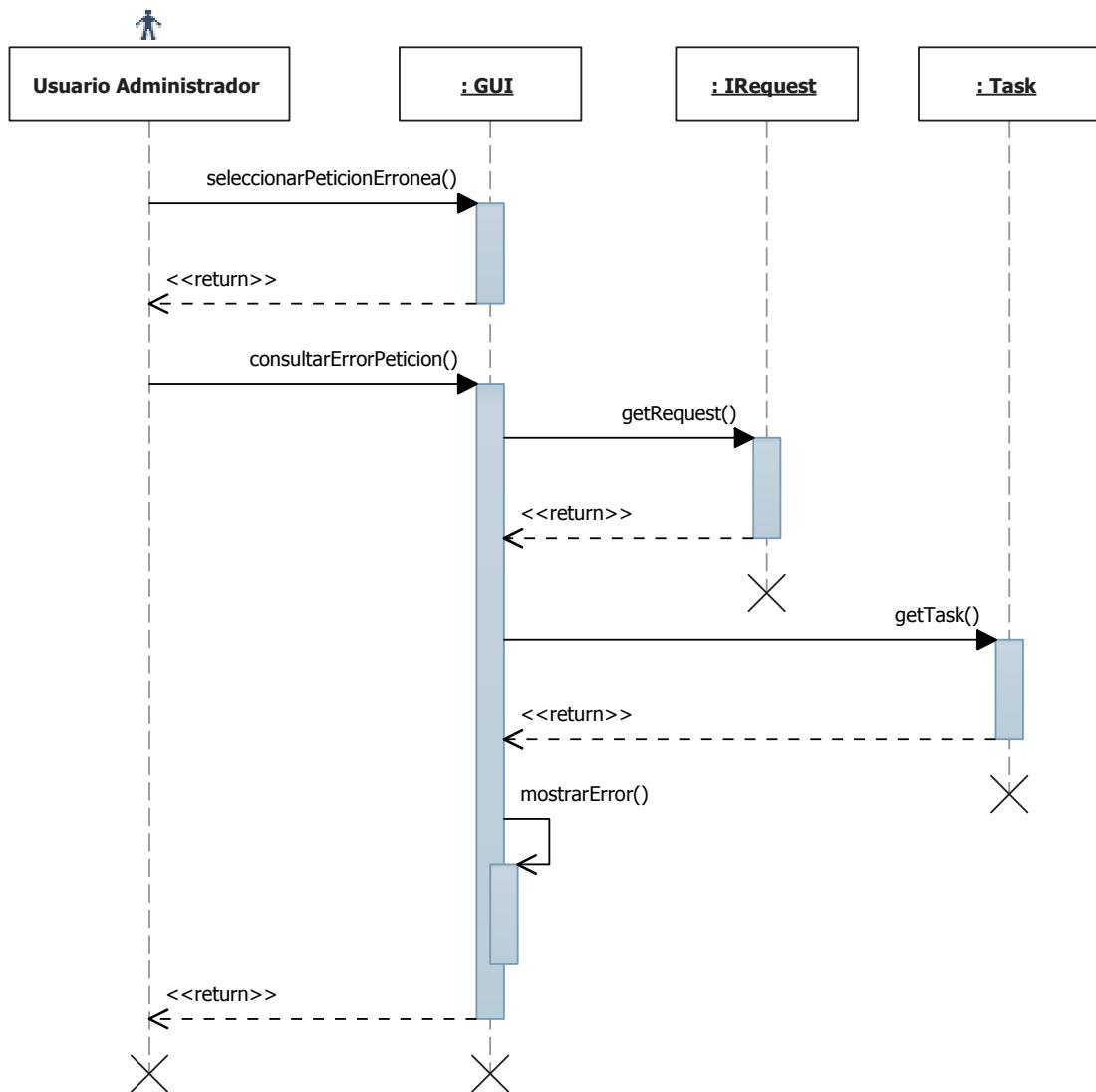
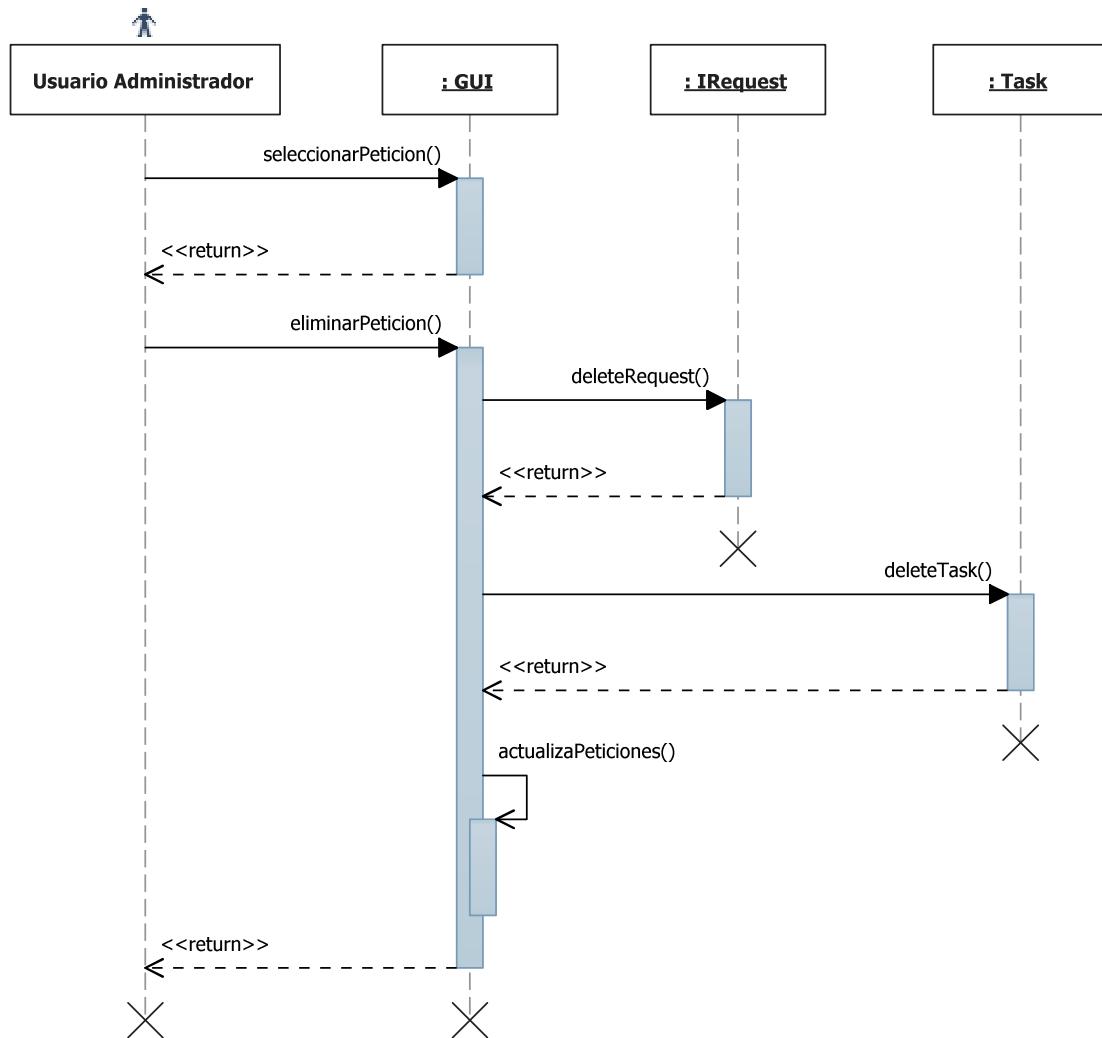


Figura 30. Módulo SUM Web: diagrama de secuencia solicitar baja usuario

SDSW009: Solicitar asignar perfil usuario**Figura 31. Módulo SUM Web: diagrama de secuencia solicitar asignar perfil usuario**

SDSW010: Consultar error petición**Figura 32. Módulo SUM Web: diagrama de secuencia consultar error petición**

SDSW011: Eliminar petición**Figura 33. Módulo SUM Web: diagrama de secuencia eliminar petición**

SDSW012: Solicitar acceso

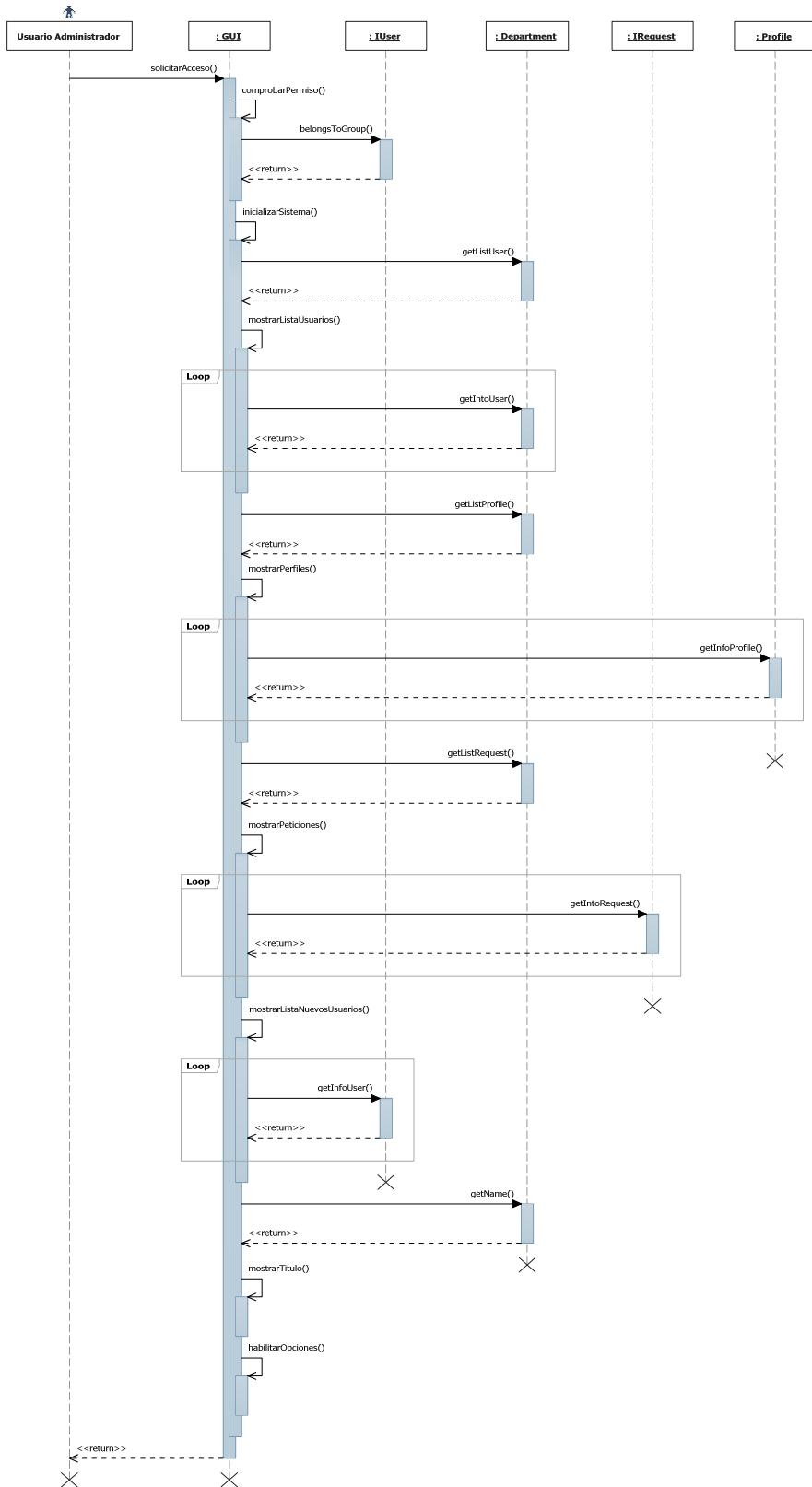
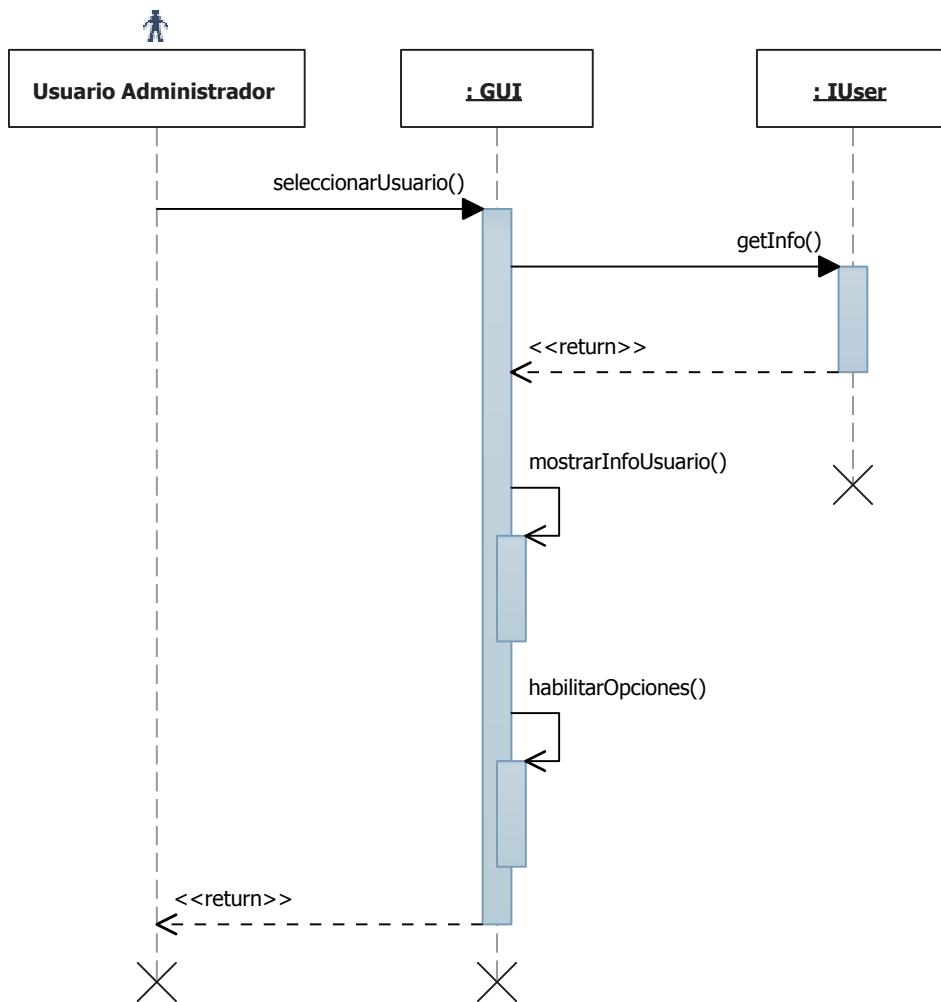


Figura 34. Módulo SUM Web: diagrama de secuencia solicitar acceso

SDSW013: Consultar datos usuario**Figura 35. Módulo SUM Web: diagrama de secuencia consultar datos usuario**

4.6.1.4 Módulo SUM Web: diagrama de clases

Dado que el paradigma de programación empleado en el desarrollo del proyecto es el de orientación a objetos, el diseño del sistema se ha realizado con una estructura de clases.

A continuación se detallan las principales clases de negocio del módulo SUM Web.

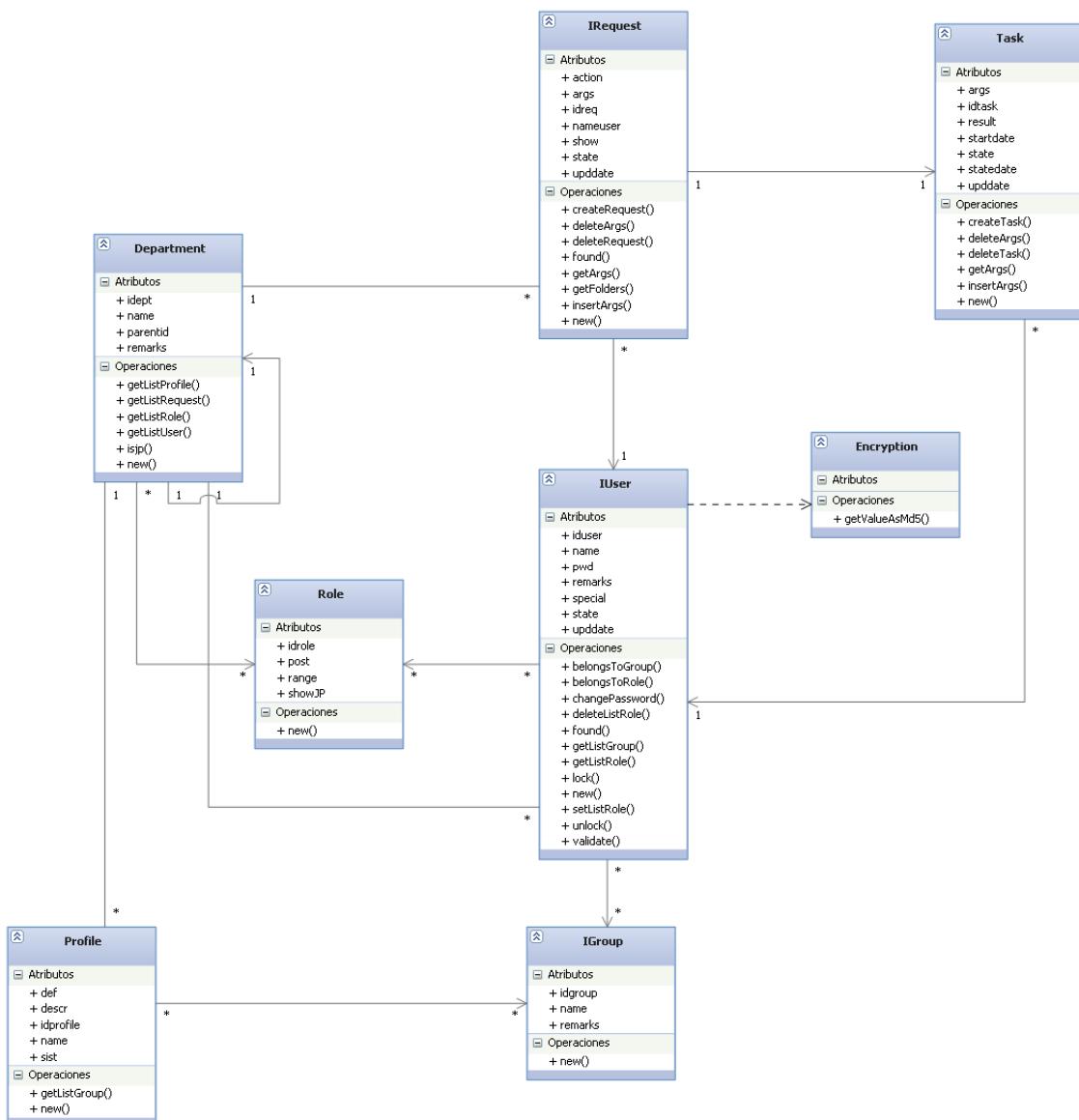


Figura 36. Módulo SUM Web: diagrama de clases

4.6.1.5 Módulo SUM Web: descripción de clases

En las siguientes tablas se representa información detallada de las clases definidas en el apartado anterior. Para cada clase conoceremos su nombre, descripción, atributos que la componen y operaciones o métodos que contiene.

CL001 IUSER	
Clase	<i>IUser</i>
Descripción	Clase que representa la información de un usuario del sistema.
Atributos	<ul style="list-style-type: none"> ▪ iduser: identificador único de usuario ▪ name: nombre de usuario ▪ pwd: clave interna de usuario ▪ state: estado (bloqueado o desbloqueado) del usuario ▪ remarks: comentarios ▪ upddate: fecha de última actualización ▪ special: indica si el usuario es de tipo especial
Operaciones	<ul style="list-style-type: none"> ▪ belongsToGroup(): método que indica si el usuario pertenece a un grupo específico que recibe como parámetro ▪ belongsToRole(): método que indica si el usuario tiene asignado un rol específico que recibe como parámetro ▪ getListGroup(): método que devuelve una lista de grupos a los que pertenece un usuario ▪ getListRole(): método que devuelve una lista de los roles firmantes de un usuario ▪ deleteListRole(): método que elimina la lista de roles firmantes de un usuario ▪ setListRole(): método que establece una lista de roles firmantes a un usuario ▪ lock(): método que modifica el estado a bloqueado de un usuario ▪ unlock(): método modifica el estado a desbloqueado de un usuario ▪ changePassword(): método que modifica la clave de un usuario ▪ validate(): método que valida si el usuario existe en el directorio de empresa e indica si es un usuario válido de creación. Devuelve nombre y apellidos del usuario. ▪ found(): método que comprueba si existe un usuario en el sistema, en cuyo caso, devuelve el departamento al que pertenece ▪ new(): constructor base de la clase

Tabla 23. SUM Web. Clase IUSER

CL002 DEPARTMENT	
Clase	Department
Descripción	Clase que representa la información de un departamento del sistema.
Atributos	<ul style="list-style-type: none"> ▪ iddept: identificador único de departamento ▪ name: nombre interno de departamento ▪ parentid: identificador del departamento padre ▪ remarks: comentarios
Operaciones	<ul style="list-style-type: none"> ▪ getListUser(): método que devuelve una lista de usuarios del departamento ▪ getLitsRequest(): método que devuelve una lista de las peticiones del departamento ▪ getListProfile(): método que devuelve una lista de perfiles del departamento ▪ getListRole(): método que devuelve una lista de roles firmantes del departamento ▪ isjp(): método que indica si el departamento es o no provincial ▪ new(): constructor base de la clase

Tabla 24. SUM Web. Clase DEPARTMENT

CL003 PROFILE	
Clase	Profile
Descripción	Clase que representa la información de un perfil del sistema.
Atributos	<ul style="list-style-type: none"> ▪ idprofile: identificador único de perfil ▪ name: nombre interno de perfil ▪ sist: indica si se trata o no un perfil especial ▪ descr: descripción de perfil ▪ def: contenido del perfil
Operaciones	<ul style="list-style-type: none"> ▪ getListGroup(): método que devuelve una lista de grupos del perfil ▪ new(): constructor base de la clase
Notas	La definición o contenido de un perfil sigue un formato XML establecido. Ver anexo “Formato del XML de descripción de perfiles”

Tabla 25. SUM Web. Clase DEPARTMENT

CL004 IGROUP	
Clase	IGroup
Descripción	Clase que representa la información de un grupo de permisos del sistema.
Atributos	<ul style="list-style-type: none"> ▪ idgroup: identificador único de grupo ▪ name: nombre interno de grupo ▪ remarks: remarks
Operaciones	<ul style="list-style-type: none"> ▪ new(): constructor base de la clase

Tabla 26. SUM Web. Clase IGROUP

CL005 IREQUEST	
Clase	IRequest
Descripción	Clase que representa la información de una petición de usuario del sistema.
Atributos	<ul style="list-style-type: none"> ▪ idreq: identificador único de petición ▪ args: lista de argumentos de la petición ▪ nameuser: nombre de usuario sobre el que recae la acción de la petición ▪ action: tipo de acción solicitada (N, P, D) ▪ state: estado de la petición ▪ show: indica si será visible o no la petición ▪ upddate: fecha de última actualización
Operaciones	<ul style="list-style-type: none"> ▪ createRequest(): método para la creación de una petición ▪ deleteRequest(): método para la eliminación de una petición ▪ deleteArgs(): elimina la lista de argumentos de una petición ▪ getArgs(): devuelve la lista de argumentos de una petición ▪ insertArgs(): genera la lista de argumentos de una petición ▪ getFolders(): devuelve una lista de peticiones que cumplen una determinada condición que se recibe como parámetro ▪ found(): método que indica si ya existe una petición de las mismas características ▪ new(): constructor base de la clase

Tabla 27. SUM Web. Clase IREQUEST

CL006 TASK	
Clase	Task
Descripción	Clase que representa la información de una tarea programada del sistema.
Atributos	<ul style="list-style-type: none"> ▪ idtask: identificador único de tarea programada ▪ args: lista de argumentos de la tarea ▪ startdate: fecha de comienzo de la tarea ▪ state: estado de la tarea ▪ statedate: fecha de última actualización del estado ▪ result: resultado de la ejecución de la tarea ▪ upddate: fecha de última actualización
Operaciones	<ul style="list-style-type: none"> ▪ createTask(): método para la creación de una tarea programada ▪ deleteTask(): método para la eliminación de una tarea programada ▪ deleteArgs(): elimina la lista de argumentos de una tarea ▪ getArgs(): devuelve la lista de argumentos de una tarea ▪ insertArgs(): genera la lista de argumentos de una tarea ▪ new(): constructor base de la clase

Tabla 28. SUM Web. Clase TASK

CL007 ENCRYPTION	
Clase	Encryption
Descripción	Clase para el cifrado de datos.
Atributos	
Operaciones	<ul style="list-style-type: none"> ▪ encrypt(): método compartido para el cifrado de datos.
Notas	Para la implementación de esta clase será necesario hacer uso de la librería de Microsoft .NET System.Security.Cryptography que proporciona servicios criptográficos, entre los que se incluyen la codificación y descodificación segura de los datos.

Tabla 29. SUM Web. Clase ENCRYPTION

CL008 ROLE	
Clase	Role
Descripción	Clase que representa la información de un rol firmante del sistema.
Atributos	<ul style="list-style-type: none"> ▪ idrole: identificador único de rol firmante ▪ post: cargo del firmante del documento ▪ range: ámbito de tramitación donde se aplica el rol ▪ showJP: indica si los administradores de las Jefaturas Provinciales pueden dar permisos de paso a firma de documentos dirigidos a este rol
Operaciones	<ul style="list-style-type: none"> ▪ new(): constructor base de la clase

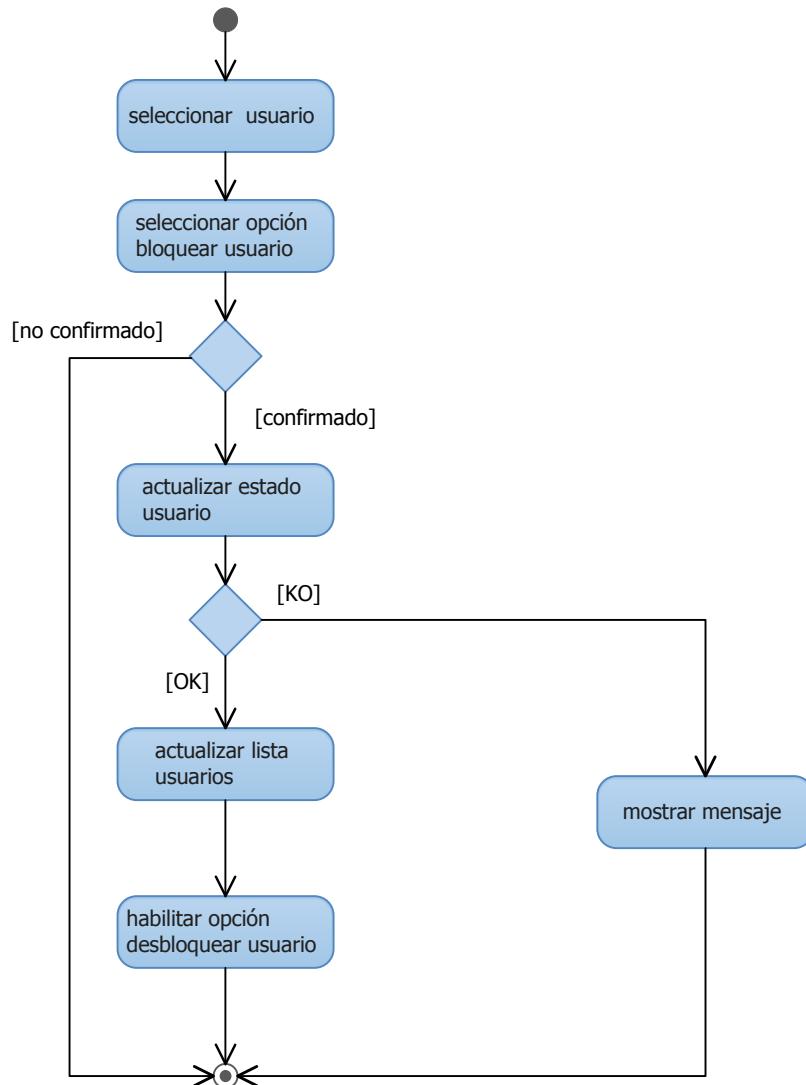
Tabla 30. SUM Web. Clase ROLE

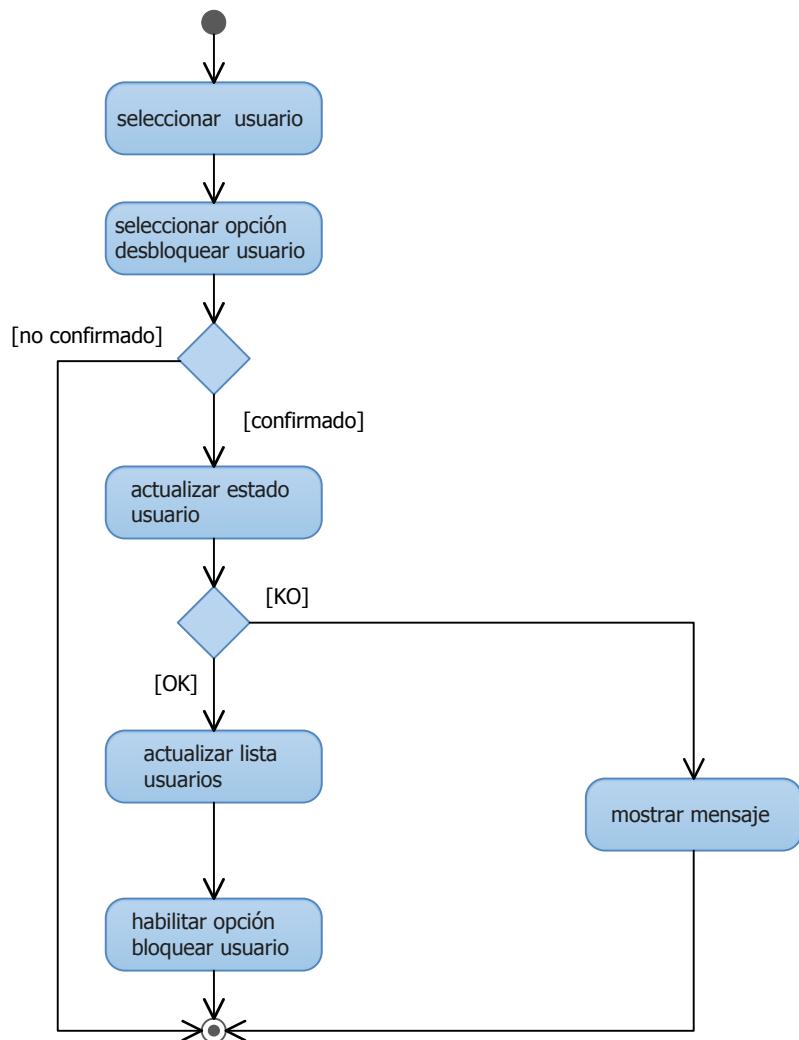
4.6.1.6 Módulo SUM Web: diagramas de actividad

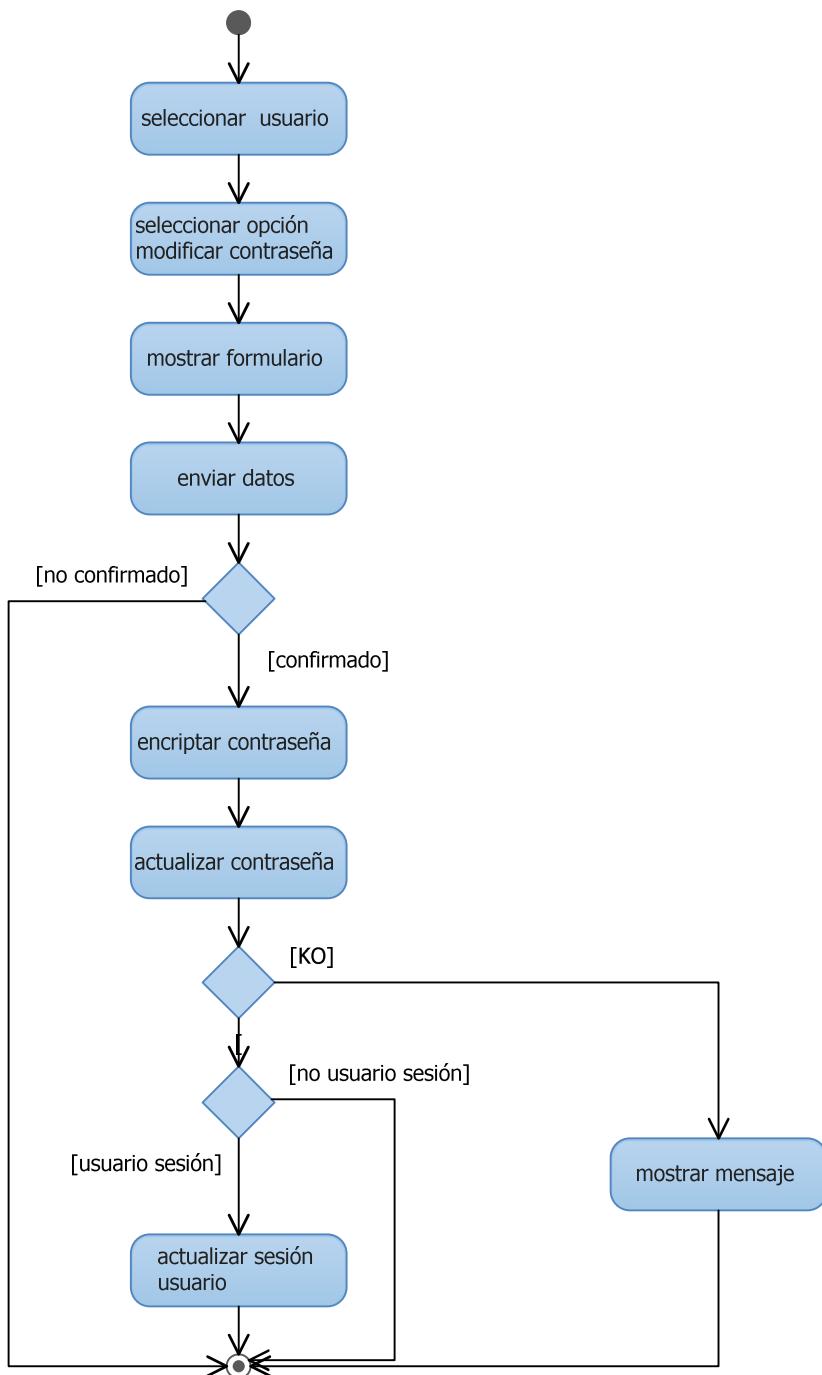
Los siguientes diagramas de actividad detallan la secuencia de acciones necesarias para la realización de los casos de uso que representan el comportamiento del módulo SUM Web.

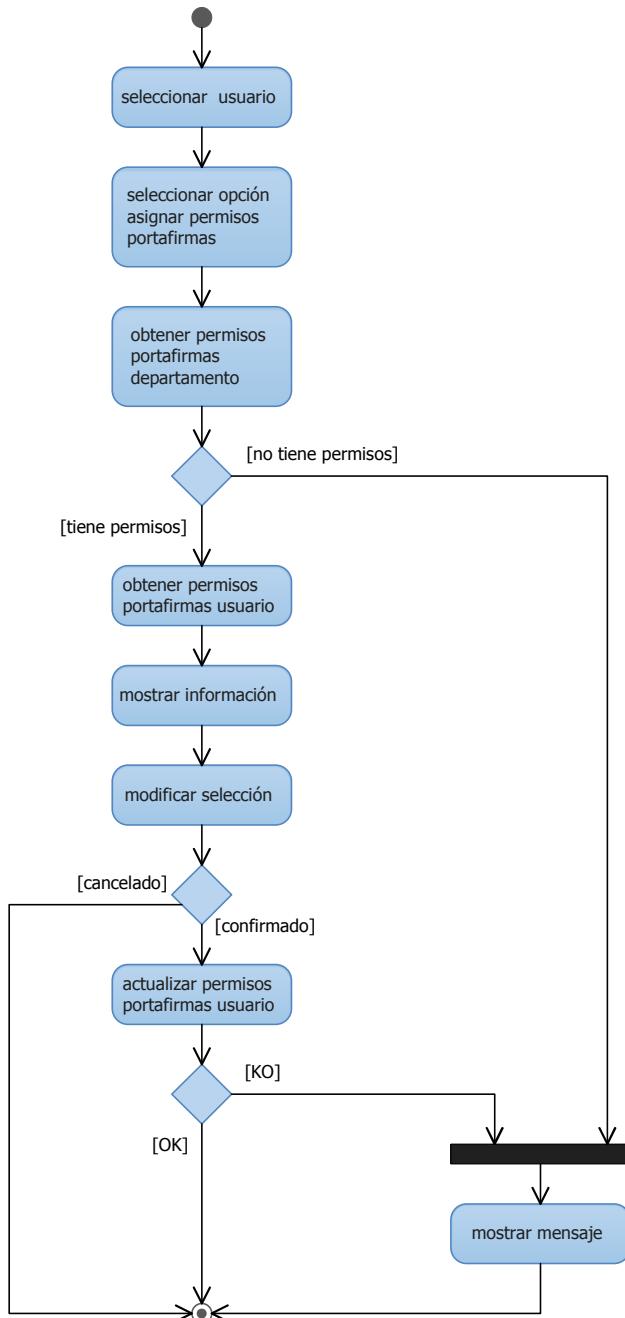
Cada diagrama se identifica por un código secuencial único y un texto descriptivo de la acción implementada.

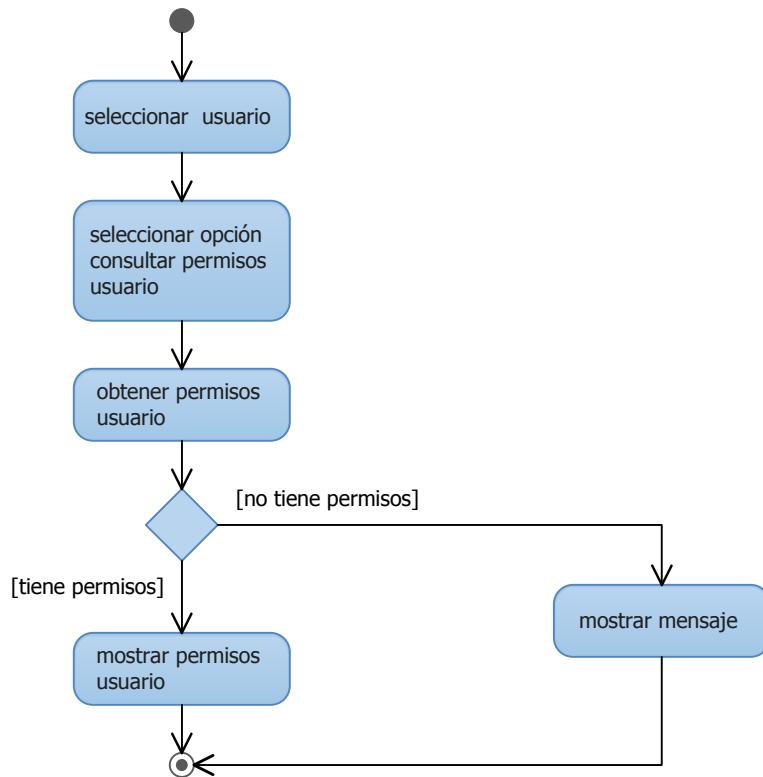
ADSWXXX → Diagrama de actividad XXX del módulo SUM Web

ADSW001: Bloquear usuario**Figura 37. Módulo SUM Web: diagrama de actividad bloquear usuario**

ADSW002: Desbloquear usuario**Figura 38. Módulo SUM Web: diagrama de actividad desbloquear usuario**

ADSW003: Modificar contraseña**Figura 39. Módulo SUM Web: diagrama de actividad modificar contraseña**

ADSW004: Asignar permisos portafirmas**Figura 40. Módulo SUM Web: diagrama de actividad asignar permisos portafirmas**

ADSW005: Consultar permisos usuario**Figura 41. Módulo SUM Web: diagrama de actividad consultar permisos usuario**

ADSW006: Consultar perfil

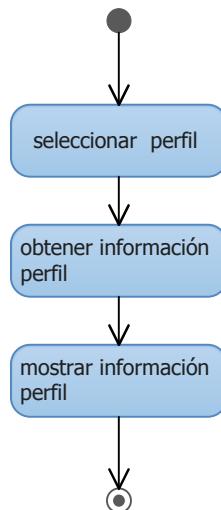


Figura 42. Módulo SUM Web: diagrama de actividad consultar perfil

ADSW007: Solicitar alta usuario

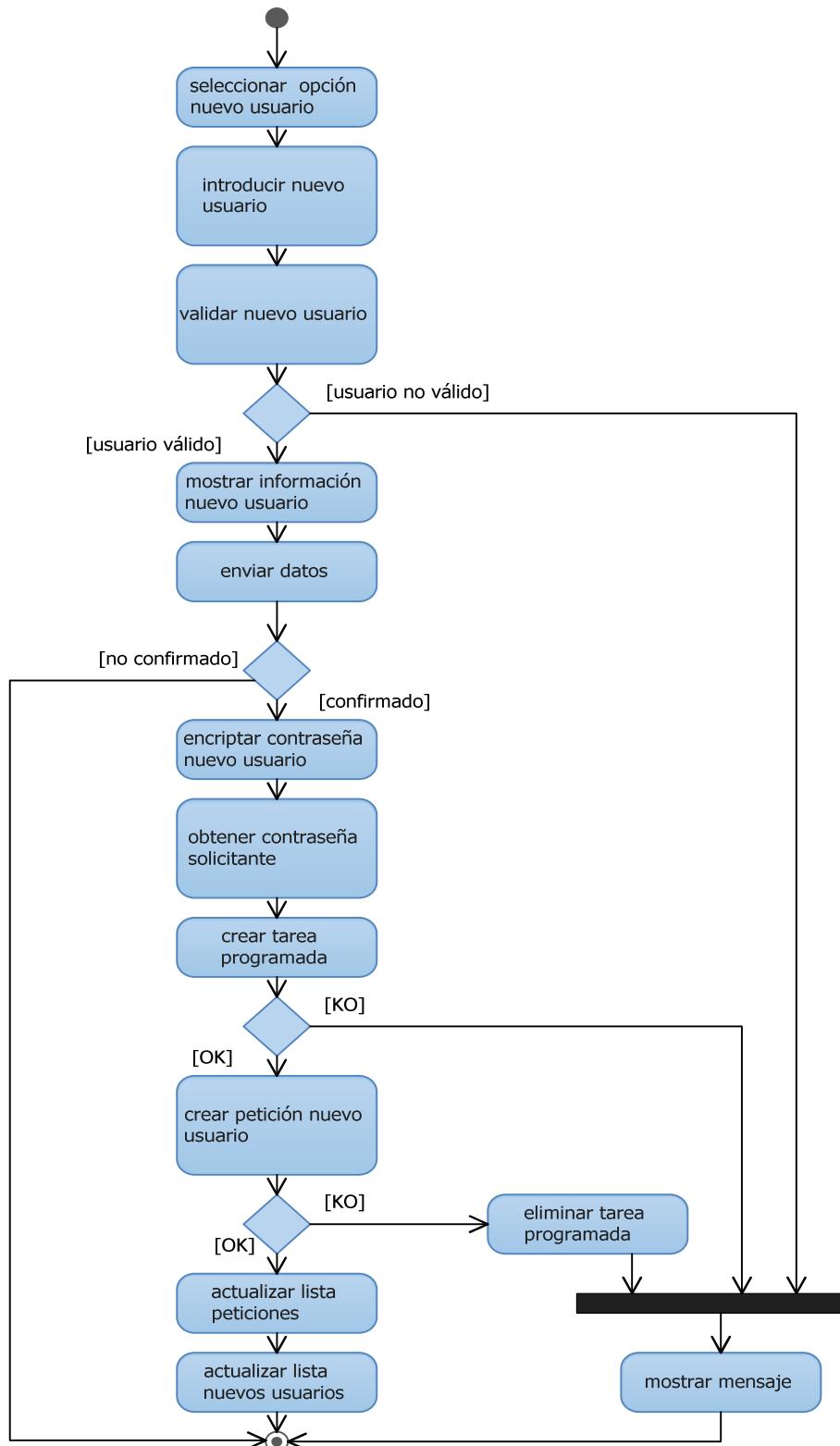


Figura 43. Módulo SUM Web: diagrama de actividad solicitar alta usuario

ADSW008: Solicitar baja usuario

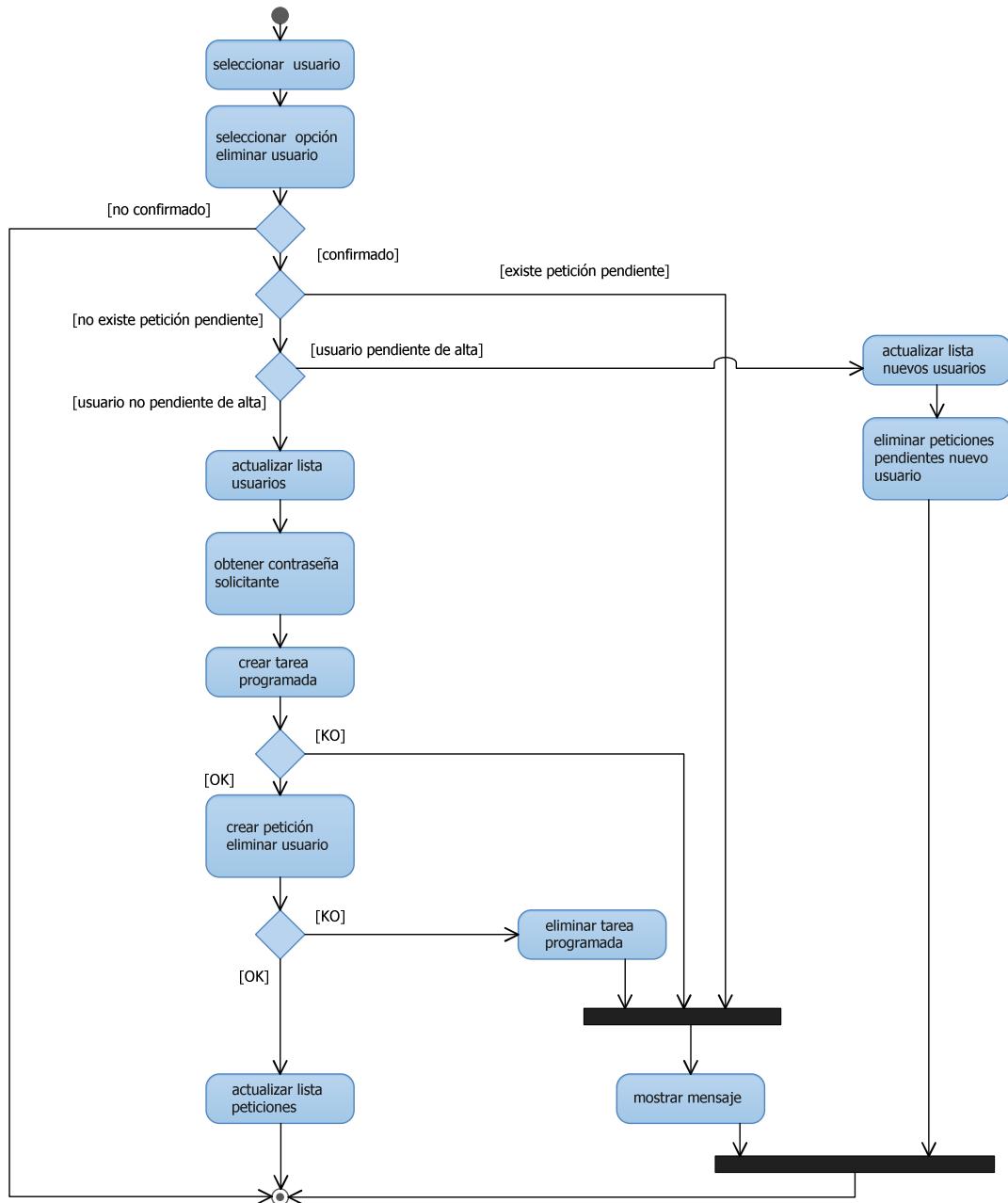


Figura 44. Módulo SUM Web: diagrama de actividad solicitar baja usuario

ADSW009: Solicitar asignar perfil usuario

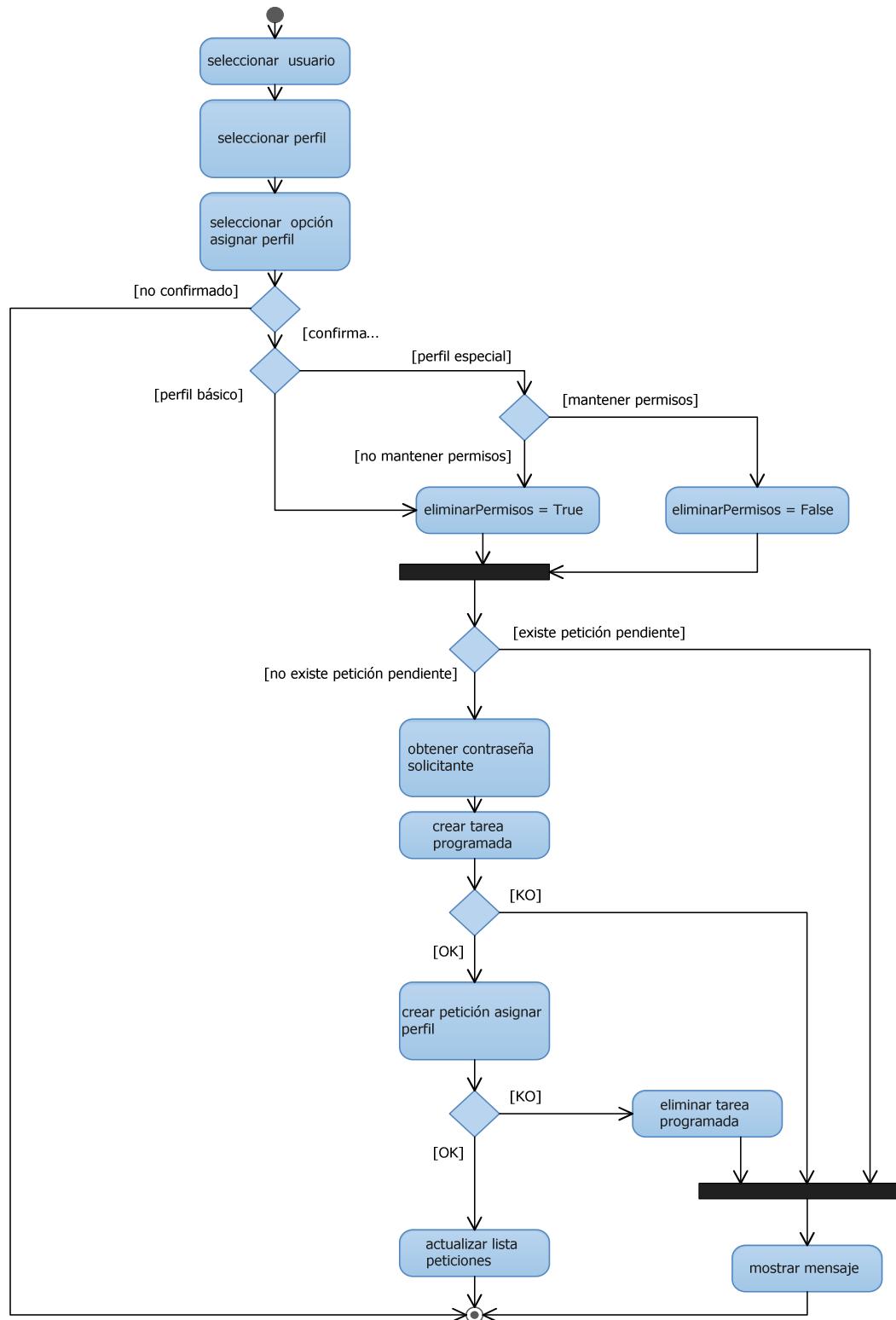


Figura 45. Módulo SUM Web: diagrama de actividad solicitar asignar perfil usuario

ADSW010: Consultar error petición

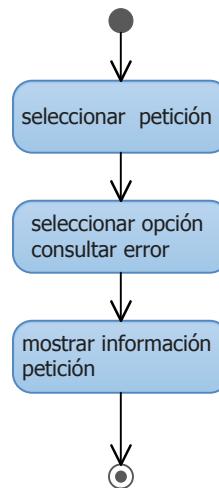


Figura 46. Módulo SUM Web: diagrama de actividad consultar error petición

ADSW011: Eliminar petición

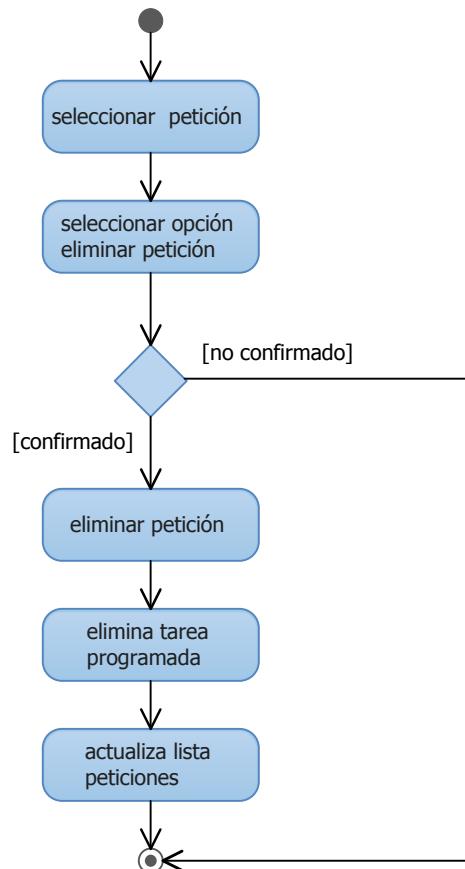


Figura 47. Módulo SUM Web: diagrama de actividad eliminar petición

ADSW012: Solicitar acceso

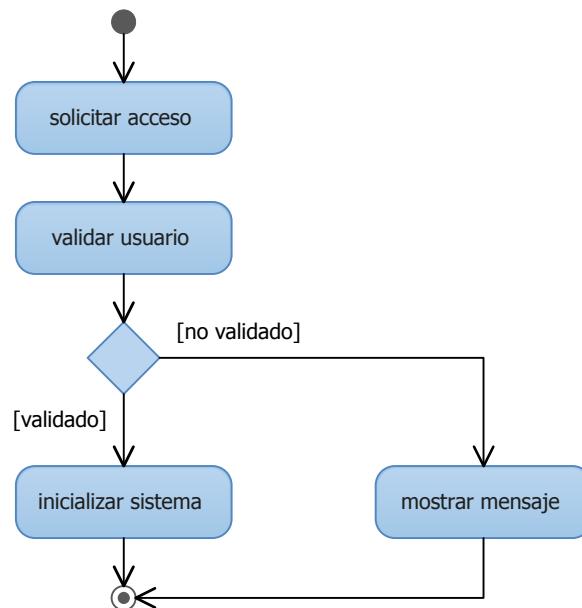


Figura 48. Módulo SUM Web: diagrama de actividad solicitar acceso

ADSW013: Consultar datos usuario



Figura 49. Módulo SUM Web: diagrama de actividad consultar datos usuario

4.6.1.7 Módulo SUM Web: mapa Web

Al pulsar el enlace de acceso a la aplicación se abre la ventana de "Inicio" que comprueba los permisos de usuario. Si el usuario cumple los requisitos de acceso se abre la ventana "Menú" que es la ventana principal de la que parten todas las navegaciones que se detallan a continuación:

- Al pulsar el botón "Cambiar contraseña" desde el menú principal se abre la ventana de "CambiarContraseña" como ventana modal.
- Al pulsar el botón "Nuevo usuario" desde el menú principal se abre la ventana de "NuevoUsuario" como ventana modal.
- Al pulsar el botón "Permisos Portafirmas" desde el menú principal se abre la ventana de "PermisosPortafirmas" como ventana modal.
- Al pulsar el botón "Permisos Usuario" desde el menú principal se abre la ventana de "PermisosUsuario" como ventana modal.

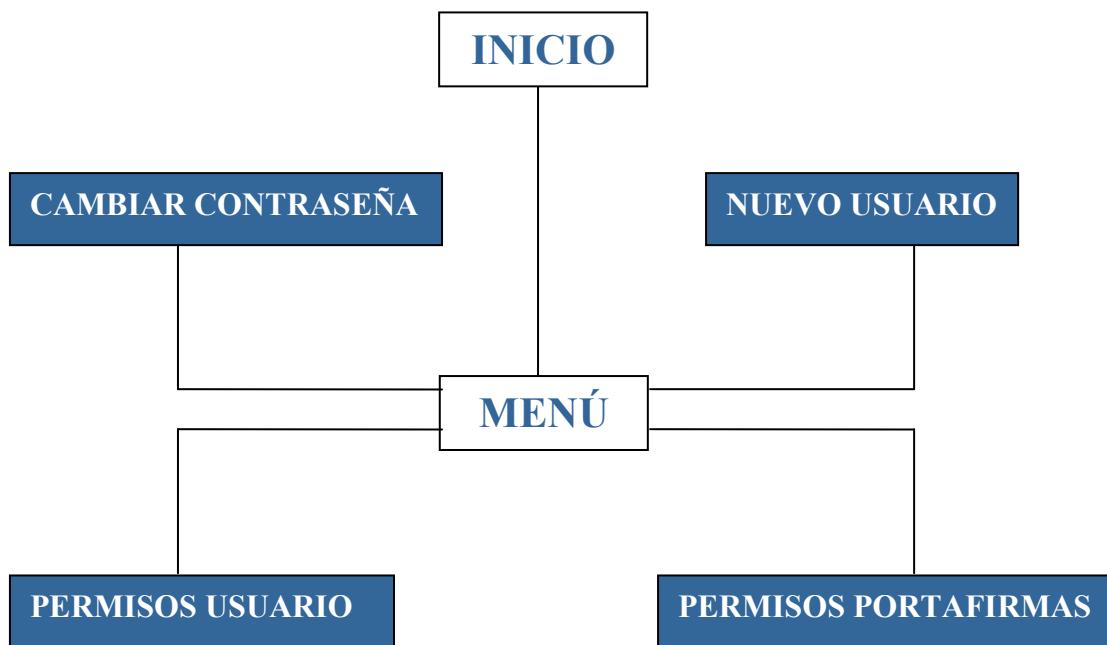


Figura 50. Módulo SUM Web: mapa Web

4.6.1.8 Módulo SUM Web: diseño de interfaces

A continuación se especifica el diseño que deben de presentar las distintas interfaces de la aplicación.

Menú

Ventana principal que presenta las opciones de menú disponibles al usuario. Muestra un listado de usuarios existentes, un listado de usuarios pendientes de alta, un listado de perfiles básicos, un listado de perfiles especiales, un listado de peticiones, espacio reservado para información de usuario, espacio reservado para información de perfil, nombre del departamento y opciones disponibles.



Figura 51. Menú principal gestión de usuarios

Si no existen peticiones del departamento a mostrar en la ventana principal se mostrará un mensaje indicativo.

CAPÍTULO 4. ANÁLISIS Y DISEÑO

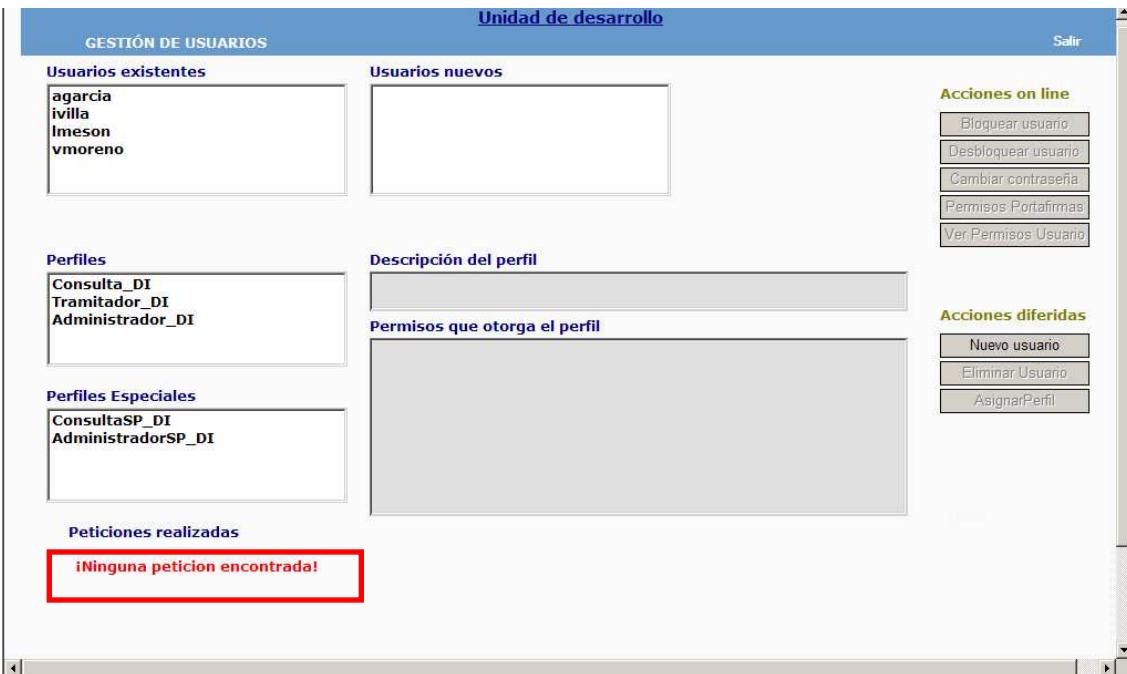


Figura 52. Menú principal gestión de usuarios

Información de usuario

Al seleccionar un usuario se presentará en el espacio reservado para ello bajo la lista de usuarios existentes de la ventana principal la información correspondiente.

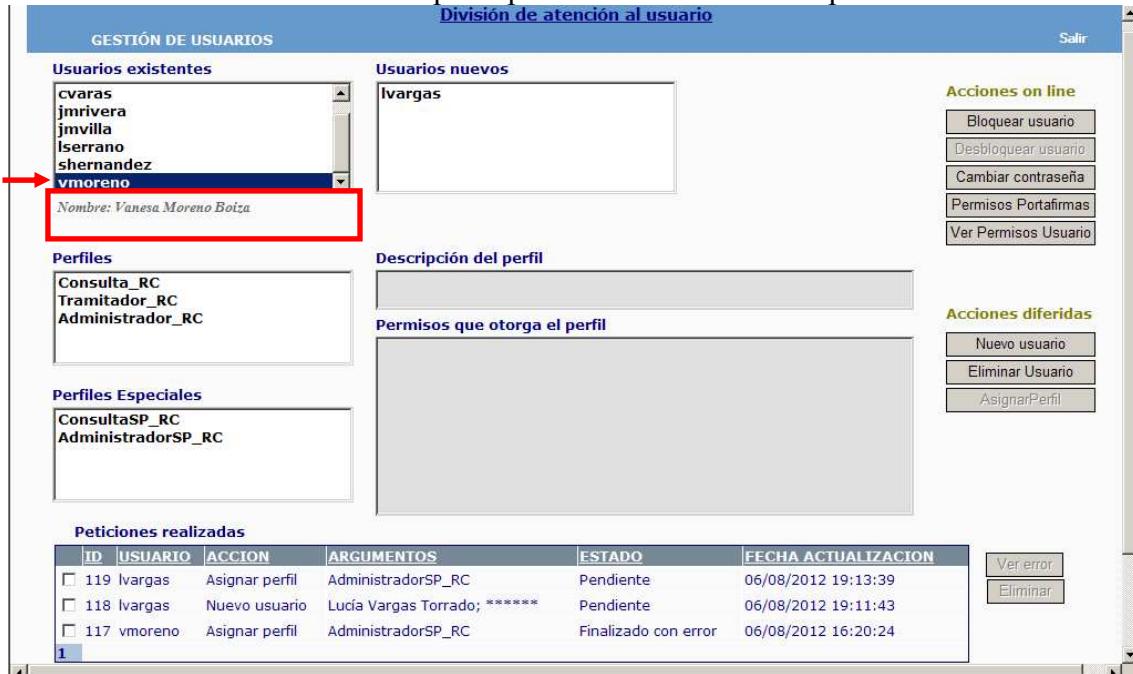


Figura 53. Menú principal gestión de usuarios

Información de perfil

Al seleccionar un perfil se presentará en el espacio destinado para ello de la ventana principal la información correspondiente.

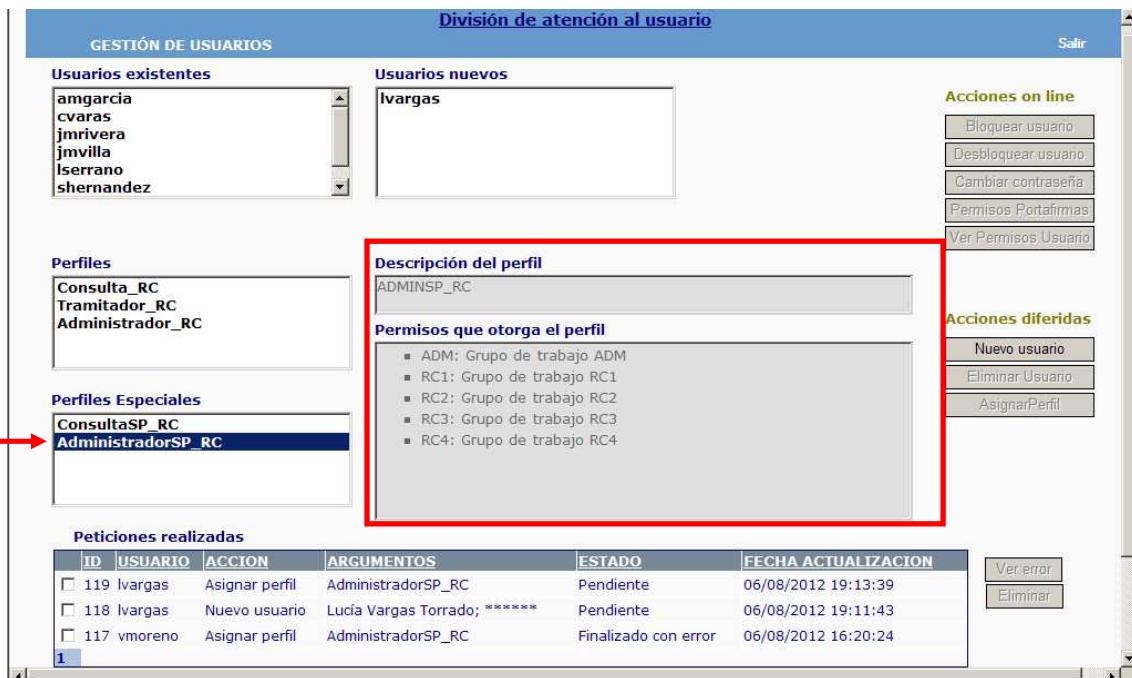


Figura 54. Menú principal gestión de usuarios

Nuevo usuario

Ventana que muestra el formulario para solicitar el alta de un nuevo usuario. Una vez introducido el nombre de usuario mostrará el nombre completo del usuario.

The screenshot shows a 'NuevoUsuario - Diálogo de página web' (NewUser - Web page dialog) window. It has four input fields: 'Usuario:' with the value 'psoto', 'Contraseña' with three masked dots, 'Repetir contraseña:' with three masked dots, and 'Nombre completo:' with the value 'Purificación Soto Pola'. At the bottom are two buttons: 'Aceptar' (Accept) and 'Cancelar' (Cancel).

Figura 55. Menú principal gestión de usuarios

Permisos usuario

Ventana que muestra la descripción de los permisos asignados al usuario seleccionado.

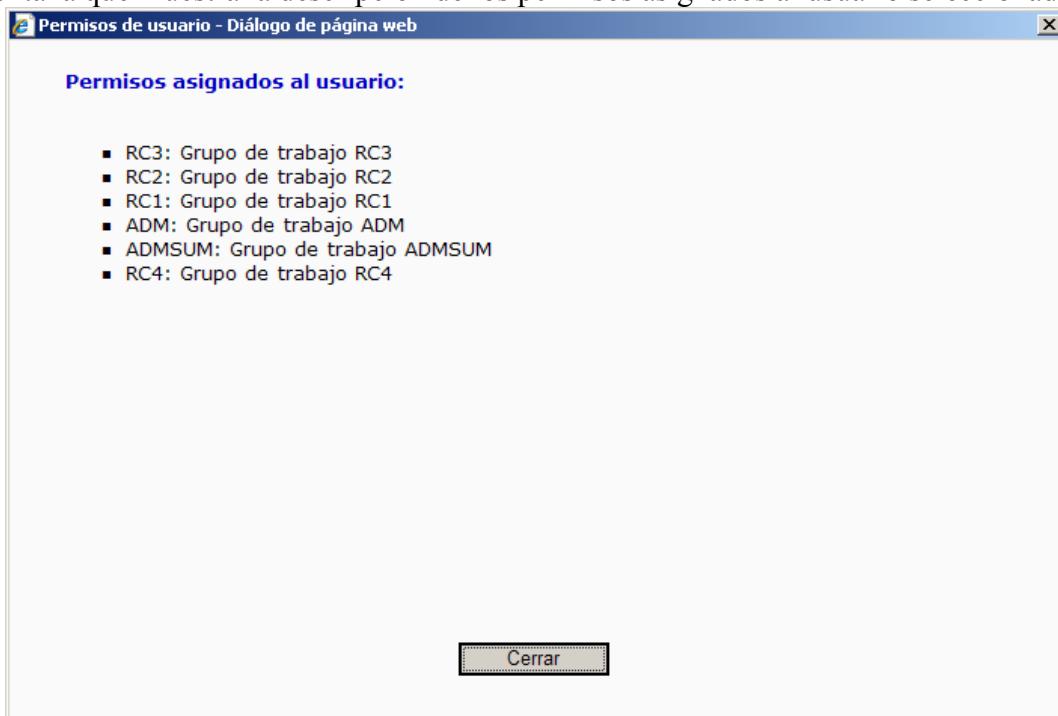


Figura 56. Menú principal gestión de usuarios

En caso de que el usuario no disponga de permisos deberá indicarse mediante un mensaje.

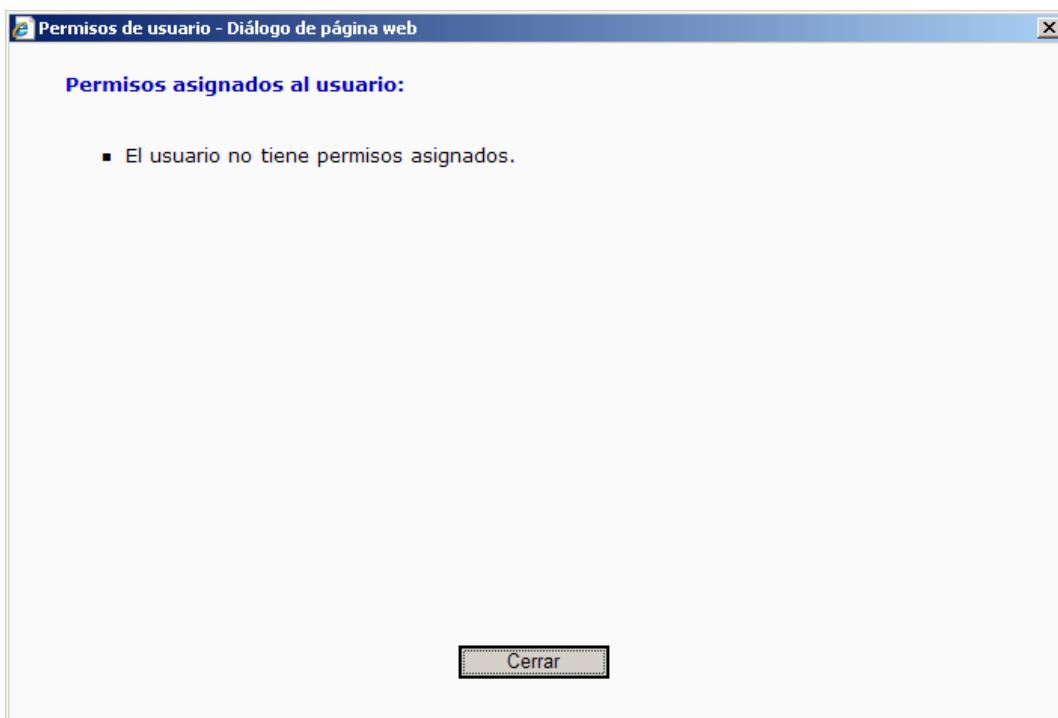


Figura 57. Menú principal gestión de usuarios

Permisos portafirmas

Ventana que muestra los permisos de portafirmas asignados al departamento. Aparecerán marcados todos aquellos que el usuario seleccionado tenga asignados.

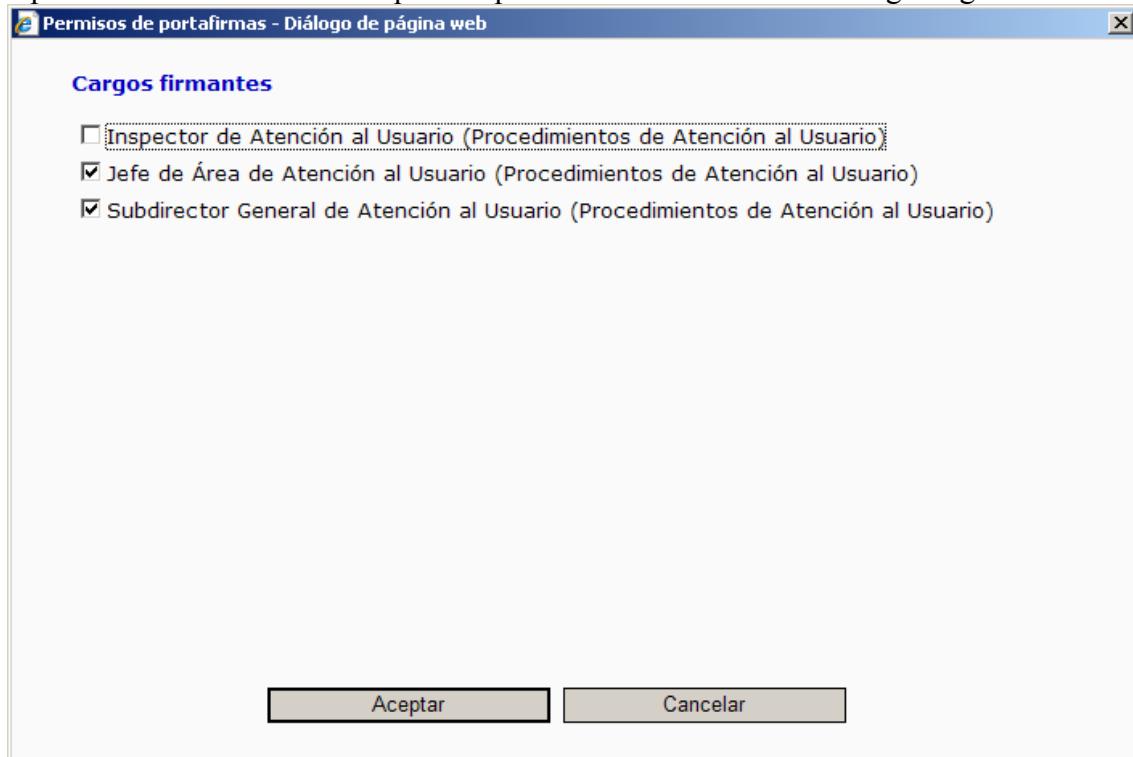


Figura 58. Menú principal gestión de usuarios

Cambiar contraseña

Ventana que muestra el formulario para modificar la contraseña de usuario.



Figura 59. Menú principal gestión de usuarios

4.6.2 Módulo SUM Server

Módulo de servidor, ejecutable de tipo "consola" sin interfaz de usuario que se ejecutará en modo *batch* en el servidor de aplicaciones. Este módulo se desarrollará en lenguaje Visual Basic y tecnología Microsoft .Net, haciendo uso del sistema gestor de base de datos SQL Server para el acceso a la información.

El módulo SUM Server será el que ejecute las tareas deferidas correspondientes a peticiones de usuario generadas desde el módulo *on line* SUM Web.

- **Gestión de alta de usuarios:** incluye nuevos usuarios en el sistema.
- **Gestión de asignación de perfiles:** asigna perfiles específicos a usuarios del sistema.
- **Gestión de eliminación de usuarios:** se encarga de eliminar usuarios del sistema de manera no definitiva, es decir, generando un histórico de usuarios eliminados. Además al eliminar un usuario se deben de eliminar todos los permisos que tenga asociados.
- **Gestión de peticiones y tareas programadas:** actualiza el estado, fecha y resultado de las peticiones de usuario ejecutadas así como de las tareas programadas asociadas a dichas peticiones.

Al tratarse de un módulo de servidor no se encuentra integrado en ningún sistema. La aplicación será iniciada a través de un sistema externo de planificación de tareas (STM) que gestiona tareas planificadas y programadas. Para ello, debe de cumplir con los siguientes requisitos.

Requisitos de SUM Server para la integración con el subsistema STM.

Este módulo será invocado siempre que haya alguna tarea programada del tipo correspondiente en el subsistema de planificación de tareas del entorno de trabajo. Por tanto este módulo debe cumplir los requisitos necesarios para la integración con dicho subsistema, a saber:

- Debe ser un ejecutable de tipo "consola", sin interfaz de usuario
- Recibirá todos los argumentos de entrada que necesite por la línea de comandos del sistema
- Su primer argumento será el identificador de la tarea programada que debe procesar
- Al finalizar su ejecución, valiéndose de dicho identificador, debe de actualizar la información de estado de la tarea programada. Esto supone la actualización de los campos estado, fecha y hora de actualización y resultado (breve descripción del resultado de la ejecución)

Al iniciarse, el módulo SUM Server, recibirá todos los argumentos de entrada necesarios por línea de comandos del sistema. La sintaxis de invocación a la aplicación seguirá el siguiente formato:

- Ruta y nombre del fichero ejecutable
- Identificador de tarea programada
- Identificador de acción a ejecutar:
 - N: alta de nuevo usuario
 - P: asignación de perfil a usuario
 - D: eliminación de usuario
- Nombre de usuario solicitante
- Contraseña cifrada de usuario solicitante
- Argumentos en función de la acción a ejecutar.

Descripción de argumentos de entrada para alta de usuario.

- Nombre de usuario a crear
- Contraseña cifrada de usuario a crear
- Nombre completo de usuario a crear

Descripción de argumentos de entrada para asignación de perfil a usuario.

- Nombre de usuario de asignación
- Identificador del perfil a asignar
- *True/False* para eliminar o mantener los permisos previos del usuario de asignación

Descripción de argumentos de entrada para eliminación de usuario.

- Nombre de usuario a eliminar

Ejemplos de invocación por línea de comandos al modulo SUM Server:

El siguiente fragmento de código muestra un ejemplo de creación de nuevo usuario.

```
“““C:\SUM\SUMServer.exe” 1 “N” “admin” “xvfv” “vmoreno” “urwd” “Vanesa Moreno Boiza”””
```

El siguiente fragmento de código muestra un ejemplo de asignación de perfil a usuario.

```
“““C:\SUM\SUMServer.exe” 2 “P” “admin” “xvfv” “vmoreno” 10 “True”””
```

El siguiente fragmento de código muestra un ejemplo de eliminación de usuario.

```
"""C:\SUM\SUMServer.exe" 3 "D" "admin" "xvfs" "vmoreno"""
```

4.6.2.1 Módulo SUM Server: diagrama de casos de uso

Los casos de uso describen la funcionalidad del sistema basándose en los requisitos previamente establecidos. En el diagrama de casos de uso se representan los actores y los casos de uso principales del sistema, detallando las relaciones que existen entre ellos.

A continuación se muestra el diagrama de casos de uso del sistema SUM Server.

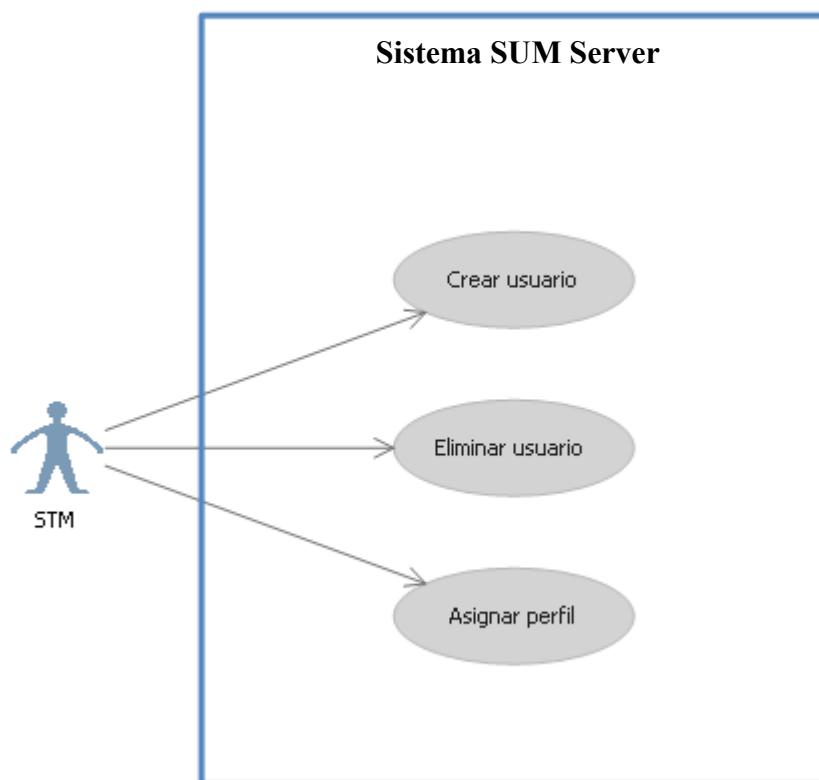


Figura 60. Módulo SUM Server: diagrama de casos de uso

El principal usuario del sistema es el subsistema de planificación de tareas STM (System Tasks Management). Este sistema externo será quien active el módulo SUM Server.

4.6.2.2 Módulo SUM Server: descripción de casos de uso

Este apartado facilita una descripción detallada de los diferentes casos de uso del sistema SUM Server.

UC001 CREAR USUARIO	
Caso de uso	Crear usuario.
Actores	Usuario STM.
Objetivo	Crear un usuario en el sistema.
Descripción	Crear un nuevo usuario en el departamento del usuario que generó la acción diferida.
Precondiciones	Recepción de parámetros para la creación de un nuevo usuario.
Postcondiciones	Usuario creado.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario STM invoca al módulo SUM Server a través de la línea de comandos indicando los siguientes parámetros en el orden especificado: <ul style="list-style-type: none"> o Ruta fichero ejecutable o Identificador de tarea programada a ejecutar o Identificador de acción a ejecutar = N o Nombre usuario solicitante o Contraseña cifrada de usuario solicitante o Nombre de usuario a crear o Contraseña cifrada de usuario a crear o Nombre completo de usuario a crear 2. El sistema comprueba el número de argumentos de entrada recibidos así como la acción a ejecutar. 3. El sistema valida el permiso de acceso y la contraseña del usuario solicitante. 4. El sistema obtiene el identificador del departamento del usuario solicitante que será el departamento destino de creación del nuevo usuario. 5. El sistema crea un nuevo usuario de tipo no especial en el departamento destino. 6. El sistema actualiza los campos de petición: estado OK, fecha y hora de actualización de estado y marcar como no visible. 7. El sistema actualiza estado OK, fecha y hora de actualización y descripción de tarea programada.
Flujo alternativo	<p>(2.1). En caso de (no recibir el número de argumentos correspondiente a la acción requerida) el sistema muestra un mensaje indicativo.</p> <p>(3.1). En caso de (no validar usuario) el sistema no continua el proceso y actualiza estado ERR de petición y tarea programada correspondientes.</p> <p>(5.1). En caso de (error) el sistema actualiza estado ERR de petición y no continua el proceso.</p>

UC001 CREAR USUARIO	
	(5.2). En caso de (error) el sistema actualizar estado ERR de tarea programada y no continua el proceso.
Requisitos origen	SR005, SR006, SR007, SR008, SR009, SR012, SR019, SR034, SR039, UR041

Tabla 31. SUM Server. Caso de uso: crear usuario

UC002 ELIMINAR USUARIO	
Caso de uso	Eliminar usuario.
Actores	Usuario STM.
Objetivo	Eliminar un usuario del sistema.
Descripción	Se elimina un determinado usuario del sistema.
Precondiciones	Recepción de parámetros para la eliminación de un usuario.
Postcondiciones	Usuario eliminado.
Flujo normal	<p>1. El usuario STM invoca al módulo SUM Server a través de la línea de comandos indicando los siguientes parámetros en el orden especificado:</p> <ul style="list-style-type: none"> ○ Ruta fichero ejecutable ○ Identificador de tarea programada a ejecutar ○ Identificador de acción a ejecutar = D ○ Nombre usuario solicitante ○ Contraseña cifrada de usuario solicitante ○ Nombre de usuario a eliminar <p>2. El sistema comprueba el número de argumentos de entrada recibidos así como la acción a ejecutar.</p> <p>3. El sistema valida el permiso de acceso y la contraseña del usuario solicitante.</p> <p>4. El sistema obtiene el identificador del departamento del usuario solicitante.</p> <p>5. El sistema valida que el departamento del usuario solicitante sea el mismo que el departamento del usuario sobre el que recae la acción.</p> <p>6. El sistema crea un usuario eliminado (tabla <i>Remuser</i>).</p> <p>7. El sistema elimina los roles firmantes del usuario.</p> <p>8. El sistema elimina los grupos asignados al usuario.</p> <p>9. El sistema elimina el usuario.</p> <p>10. El sistema actualiza campos de petición: estado OK, fecha y hora de actualización de estado y marcar como no visible.</p> <p>11. El sistema actualiza estado OK, fecha y hora de actualización y descripción de tarea programada.</p>
Flujo alternativo	<p>(2.1). En caso de (no recibir el número de argumentos correspondiente a la acción requerida) el sistema muestra un mensaje indicativo.</p> <p>(3.1). En caso de (no validar usuario) el sistema no continua el</p>

UC002 ELIMINAR USUARIO

	<p>proceso y actualiza estado ERR la petición y tarea programada correspondientes.</p> <p>(5.1). En caso de (no validar departamento de usuario) el sistema no continua el proceso y actualiza estado ERR de petición y tarea programada correspondientes.</p> <p>(8.1). En caso de (error) el sistema actualiza estado ERR de petición.</p> <p>(8.2). En caso de (error) el sistema actualiza estado ERR de tarea programada.</p>
Requisitos origen	SR005, SR006, SR007, SR008, SR009, SR012, SR034, SR039

Tabla 32. SUM Server. Caso de uso: eliminar usuario

UC003 ASIGNAR PERFIL

Caso de uso	Asignar perfil.
Actores	Usuario STM.
Objetivo	Asignar un perfil a un usuario del sistema.
Descripción	Asignación de los permisos que otorga un perfil específico a un usuario específico.
Precondiciones	Recepción de parámetros para la asignación de perfil a usuario.
Postcondiciones	Perfil asignado a un usuario.
Flujo normal	<ol style="list-style-type: none"> El usuario STM invoca al módulo SUM Server a través de la línea de comandos indicando los siguientes parámetros en el orden especificado: <ul style="list-style-type: none"> ○ Ruta fichero ejecutable ○ Identificador de tarea programada a ejecutar ○ Identificador de acción a ejecutar = P ○ Nombre usuario solicitante ○ Contraseña cifrada de usuario solicitante ○ Nombre de usuario a asignar perfil ○ Identificador del perfil a asignar ○ <i>True or False</i> El sistema comprueba el número de argumentos de entrada recibidos así como la acción a ejecutar. El sistema valida el permiso de acceso y contraseña del usuario solicitante. El sistema obtiene el identificador del departamento del usuario solicitante. El sistema valida que el departamento del usuario solicitante sea el mismo que el departamento del usuario sobre el que recae la acción. El sistema obtiene los grupos del perfil a asignar. El sistema asigna los grupos al usuario. El sistema actualiza campos de petición: estado OK, fecha y hora de actualización de estado y marca la petición

UC003 ASIGNAR PERFIL	
	<p>como no visible.</p> <p>9. El sistema actualiza estado OK, fecha y hora de actualización y descripción de tarea programada.</p>
Flujo alternativo	<p>(2.1). En caso de (no recibir el número de argumentos correspondiente a la acción requerida) el sistema muestra un mensaje indicativo.</p> <p>(3.1). En caso de (no validar usuario) el sistema no continua el proceso y actualiza estado ERR la petición y tarea programada correspondientes.</p> <p>(5.1). En caso de (no validar departamento de usuario) el sistema no continua el proceso y actualiza estado ERR la petición y tarea programada correspondientes.</p> <p>(7.1). En caso de (recibir un parámetro para eliminar los permisos previos del usuario) el sistema eliminará los grupos asociados al usuario de manera previa a la nueva asignación.</p> <p>(7.2). En caso de (error) el sistema actualiza estado ERR de petición.</p> <p>(7.3). En caso de (error) el sistema actualiza estado ERR de tarea programada.</p>
Requisitos origen	SR005, SR006, SR007, SR008, SR009, SR012, SR021, SR022, SR034, SR039

Tabla 33. SUM Server. Caso de uso: asignar perfil

4.6.2.3 Módulo SUM Server: diagramas de secuencia

Este apartado representa los diagramas de secuencia del módulo SUM Server. Cada diagrama de secuencia muestra la secuencia de comunicación mediante mensajes entre objetos necesarios para la realización de un caso de uso.

Cada diagrama se identifica por un código secuencial único y un texto descriptivo de la acción implementada.

SDSSXXX → Diagrama de secuencia XXX del módulo SUM Server

SDSS001: Crear usuario

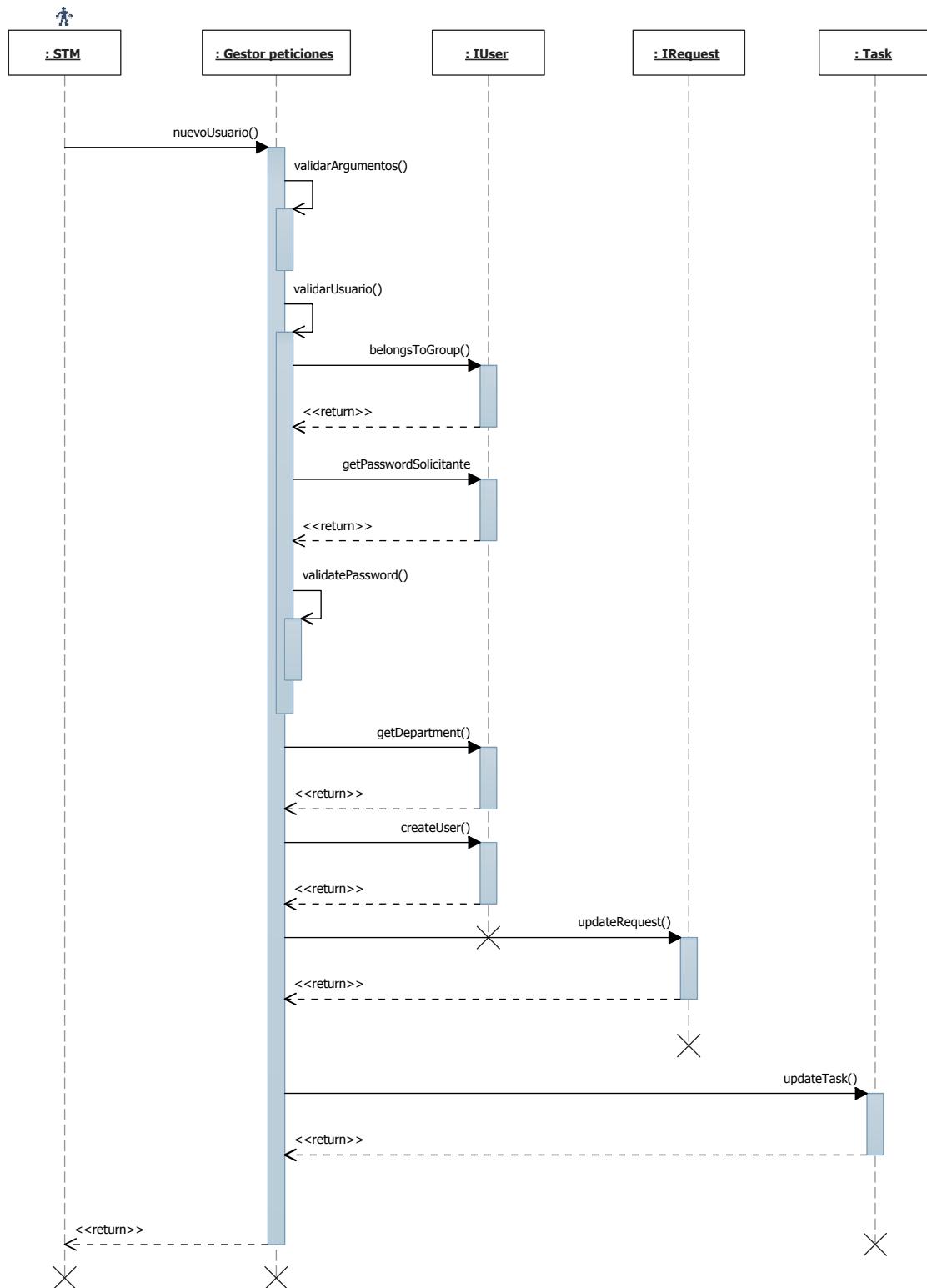


Figura 61. Módulo SUM Server: diagrama de secuencia crear usuario

SDSS002: Eliminar usuario

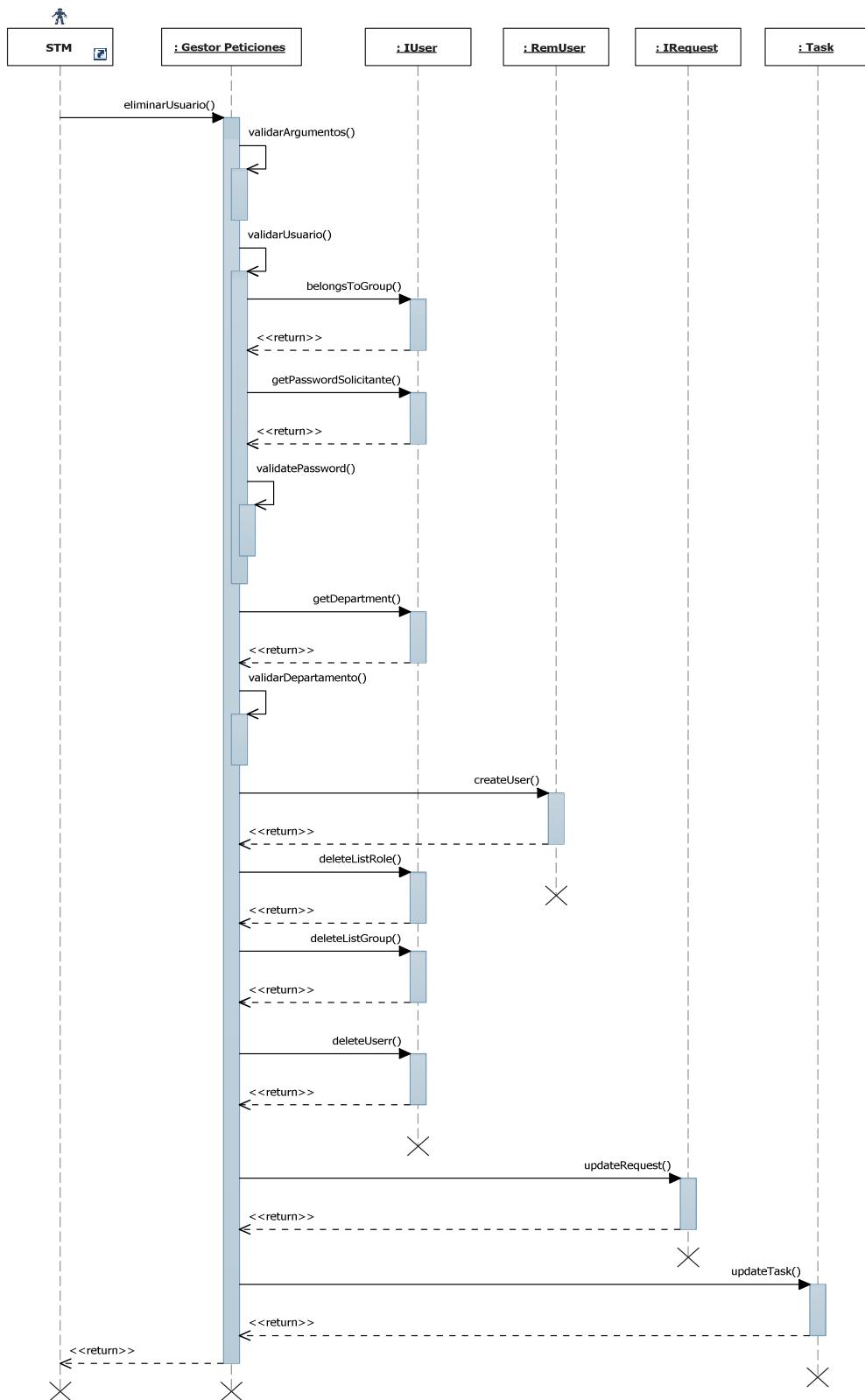


Figura 62. Módulo SUM Server: diagrama de secuencia eliminar usuario

SDSS003: Asignar perfil

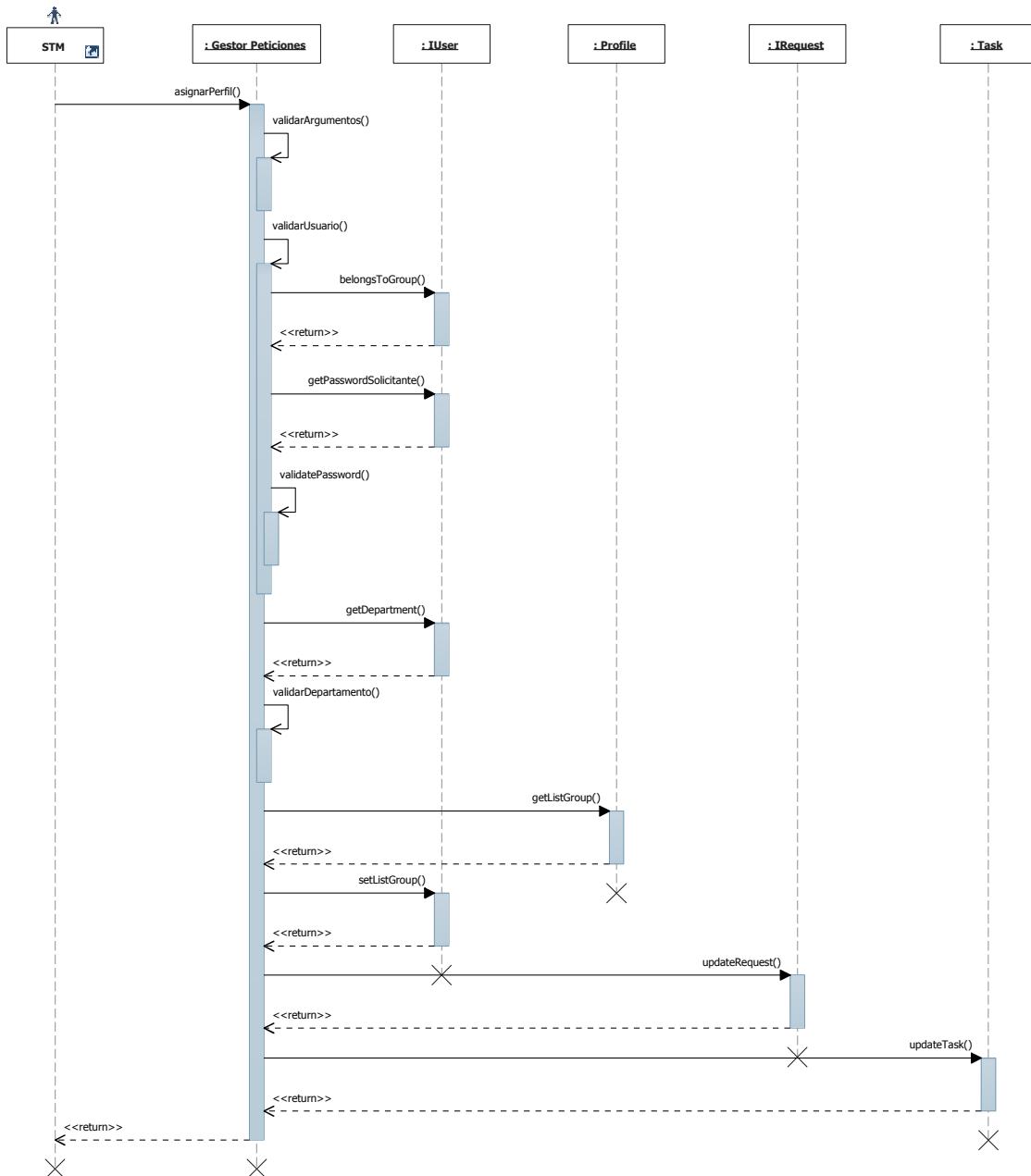


Figura 63. Módulo SUM Server: diagrama de secuencia asignar perfil

4.6.2.4 Módulo SUM Server: diagrama de clases

Dado que el paradigma de programación empleado en el desarrollo del proyecto es el de orientación a objetos, el diseño del sistema se ha realizado con una estructura de clases.

A continuación se detallan las principales clases de negocio del módulo SUM Server.

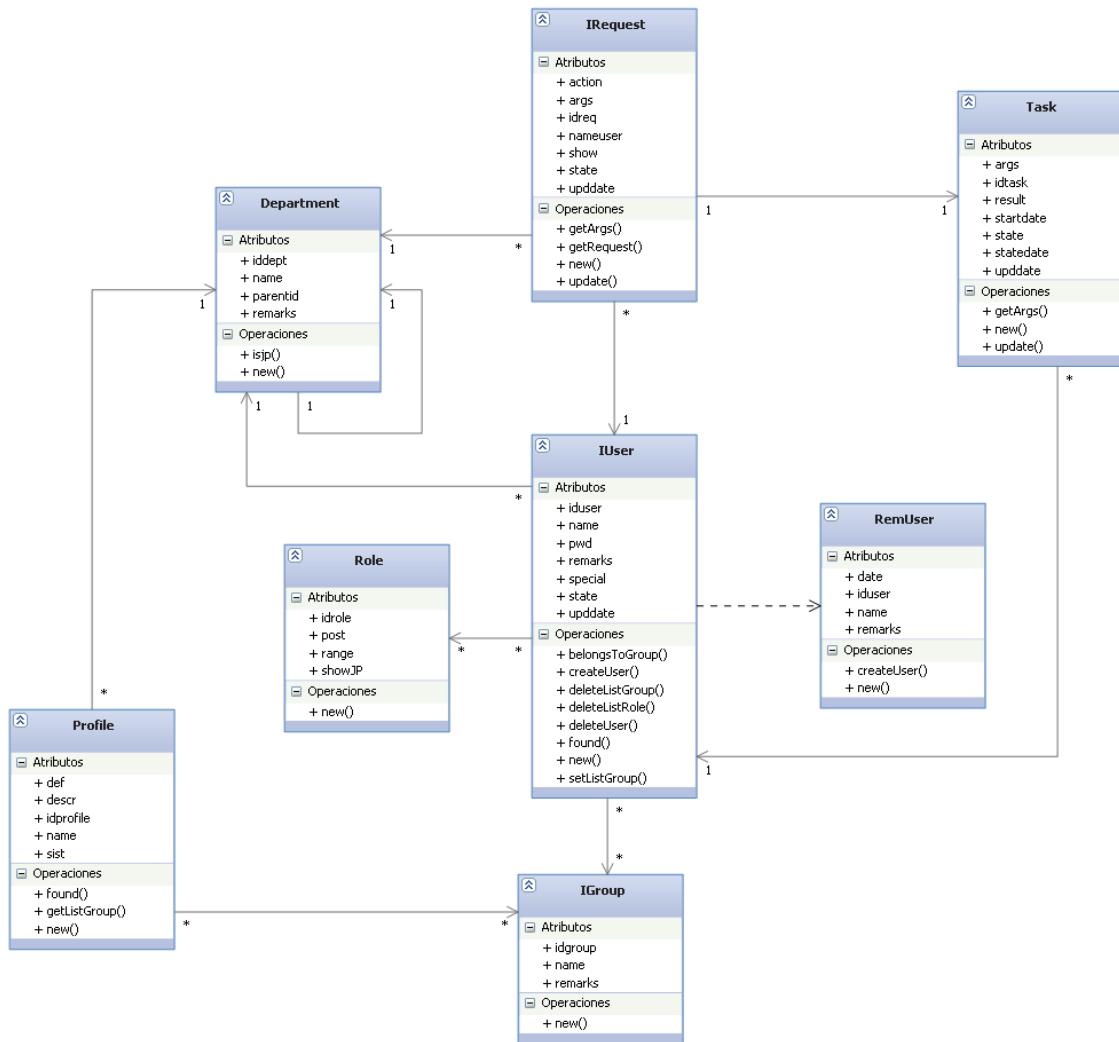


Figura 64. Módulo SUM Server: diagrama de clases

4.6.2.5 Módulo SUM Server: descripción de clases

En las siguientes tablas se representa información detallada de las clases definidas en el apartado anterior. Para cada clase conoceremos su nombre, descripción, atributos que la componen y operaciones o métodos que contiene.

CL001 IUSER	
Clase	<i>IUser</i>
Descripción	Clase que representa la información de un usuario del sistema.
Atributos	<ul style="list-style-type: none"> ▪ iduser: identificador único de usuario ▪ name: nombre de usuario ▪ pwd: clave interna de usuario ▪ state: estado (bloqueado o desbloqueado) del usuario ▪ remarks: comentarios ▪ special: indica si el usuario es de tipo especial ▪ state: estado (bloqueado o desbloqueado) del usuario ▪ upupdate: fecha de actualización
Operaciones	<ul style="list-style-type: none"> ▪ createUser(): método que crea un usuario. Recibe como parámetro el identificador del departamento de creación del usuario ▪ deleteUser(): método que elimina un usuario. Recibe un parámetro que indica si la eliminación será definitiva. Por defecto el borrado será no definitivo y se creará un objeto <i>remuser</i> ▪ setListGroup(): método que establece una lista de grupos al usuario ▪ deleteListGroup(): método que elimina la lista de grupos del usuario ▪ deleteListRole(): método que elimina una lista de roles firmantes del usuario ▪ belongsToGroup: método que indica si el usuario pertenece a un grupo específico que recibe como parámetro ▪ found: método que comprueba si existe un usuario en el sistema, en cuyo caso, devuelve el departamento al que pertenece ▪ new(): constructor base de la clase

Tabla 34. SUM Server. Clase IUSER

CL002 REMUSER	
Clase	RemUser
Descripción	Clase que representa la información de un usuario eliminado del sistema.
Atributos	<ul style="list-style-type: none"> ▪ iduser: identificador único de usuario ▪ name: nombre de usuario

CL002 REMUSER	
	<ul style="list-style-type: none"> ▪ remarks: comentarios ▪ date: fecha de eliminación
Operaciones	<ul style="list-style-type: none"> ▪ createUser(): método que crea un usuario eliminado del sistema ▪ new(): constructor base de la clase

Tabla 35. SUM Server. Clase REMUSER

CL003 PROFILE	
Clase	Profile
Descripción	Clase que representa la información de un perfil del sistema.
Atributos	<ul style="list-style-type: none"> ▪ idprofile: identificador único de perfil ▪ name: nombre interno de perfil ▪ sist: indica si se trata o no un perfil especial ▪ def: contenido del perfil ▪ descr: descripción de perfil
Operaciones	<ul style="list-style-type: none"> ▪ getListGroup(): método que devuelve una lista de grupos de un perfil ▪ found(): método que indica si existe un perfil específico ▪ new(): constructor base de la clase
Notas	La definición o contenido de un perfil sigue un formato XML establecido. Ver anexo “Formato del XML de descripción de perfiles”

Tabla 36. SUM Server. Clase PROFILE

CL004 IREQUEST	
Clase	IRequest
Descripción	Clase que representa la información de una petición de usuario del sistema.
Atributos	<ul style="list-style-type: none"> ▪ idreq: identificador único de petición ▪ args: lista de argumentos de la petición ▪ nameuser: nombre de usuario sobre el que recae la acción de la petición ▪ action: tipo de acción solicitada (N, P, D) ▪ state: estado de la petición ▪ show: indica si será visible o no la petición ▪ upddate: fecha de última actualización
Operaciones	<ul style="list-style-type: none"> ▪ update(): método que actualiza el estado de una petición de usuario ▪ getArgs(): método que devuelve la lista de argumentos de la petición ▪ getRequest(): método que devuelve la petición correspondiente al identificador que recibe como parámetro ▪ new(): constructor base de la clase

Tabla 37. SUM Server. Clase IREQUEST

CL005 TASK	
Clase	Task
Descripción	Clase que representa la información de una tarea programada del sistema.
Atributos	<ul style="list-style-type: none"> ▪ idtask: identificador único de tarea programada ▪ args: lista de argumentos de la tarea ▪ startdate: fecha de comienzo de la tarea ▪ state: estado de la tarea ▪ statedate: fecha de última actualización del estado ▪ result: resultado de la ejecución de la tarea ▪ upddate: fecha de última actualización
Operaciones	<ul style="list-style-type: none"> ▪ update(): método para la actualización de una tarea programada ▪ getArgs(): método que devuelve la lista de argumentos de la tarea ▪ new(): constructor base de la clase

Tabla 38. SUM Server. Clase TASK

CL006 DEPARTMENT	
Clase	Department
Descripción	Clase que representa la información de un departamento del sistema.
Atributos	<ul style="list-style-type: none"> ▪ iddept: identificador único de departamento ▪ name: nombre interno de departamento ▪ parentid: identificador del departamento padre ▪ remarks: comentarios
Operaciones	<ul style="list-style-type: none"> ▪ isjp(): método que indica si el departamento es o no provincial ▪ new(): constructor base de la clase

Tabla 39. SUM Server. Clase DEPARTMENT

CL007 IGROUP	
Clase	IGroup
Descripción	Clase que representa la información de un grupo de permisos del sistema.
Atributos	<ul style="list-style-type: none"> ▪ idgroup: identificador único de grupo ▪ name: nombre interno de grupo ▪ remarks: remarks
Operaciones	<ul style="list-style-type: none"> ▪ new(): constructor base de la clase

Tabla 40. SUM Server. Clase IGROUP

CL008 ROLE	
Clase	Role
Descripción	Clase que representa la información de un rol firmante del sistema.

CL008 ROLE

Atributos	<ul style="list-style-type: none"> ▪ idrole: identificador único de rol firmante ▪ post: cargo del firmante del documento ▪ range: ámbito de tramitación donde se aplica el rol ▪ showJP: indica si los administradores de las Jefaturas Provinciales pueden dar permisos de paso a firma de documentos dirigidos a este rol
Operaciones	<ul style="list-style-type: none"> ▪ new(): constructor base de la clase

Tabla 41. SUM Server. Clase ROLE

4.6.2.6 Módulo SUM Server: diagramas de actividad

Los siguientes diagramas de actividad detallan la secuencia de acciones necesarias para la realización de los casos de uso que representan el comportamiento del módulo SUM Server.

Cada diagrama se identifica por un código secuencial único y un texto descriptivo de la acción implementada.

ADSSXXX→ Diagrama de actividad XXX del módulo SUM Server

ADSS001: Crear usuario

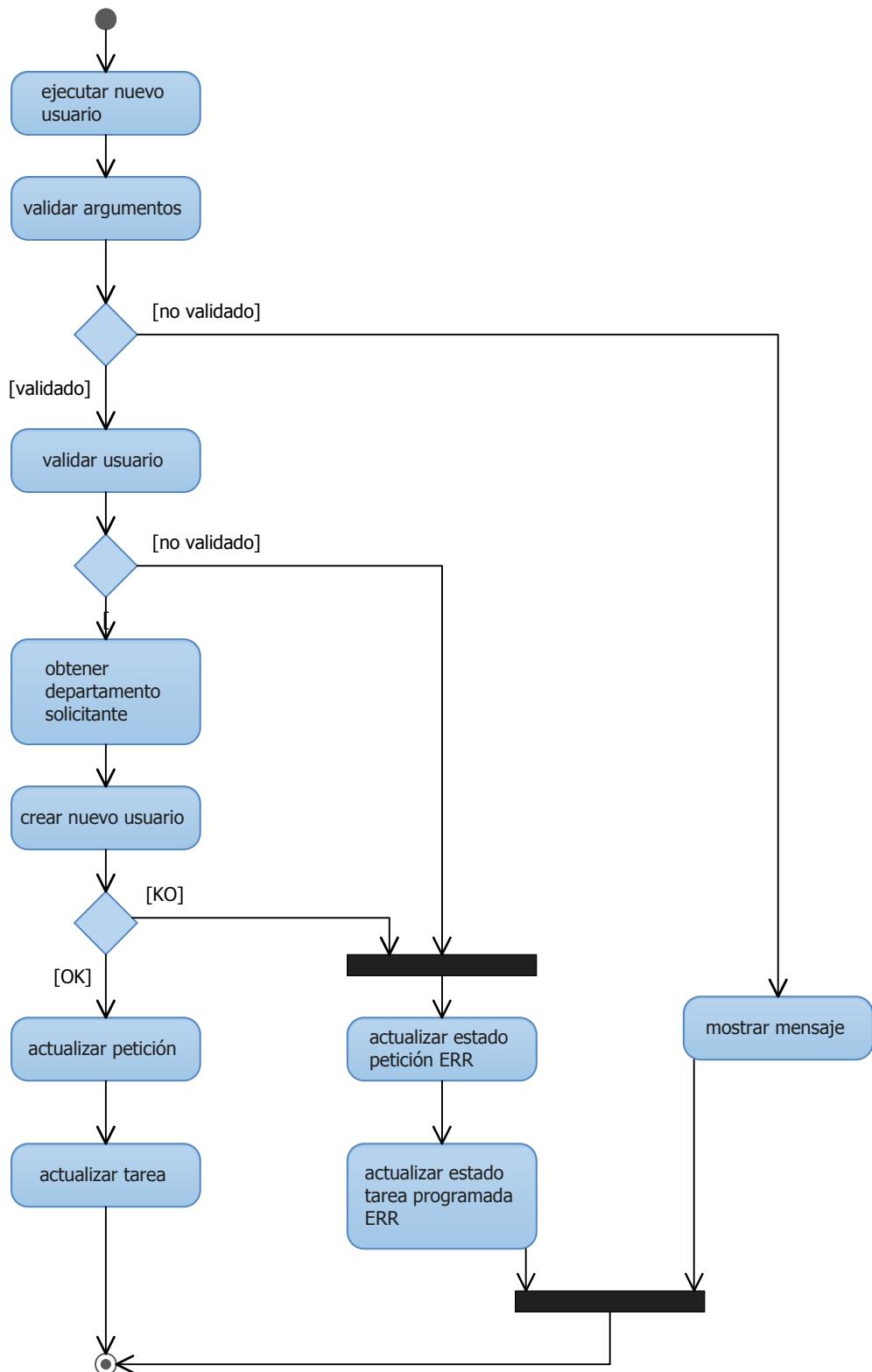


Figura 65. Módulo SUM Server: diagrama de actividad crear usuario

ADSW002: Eliminar usuario

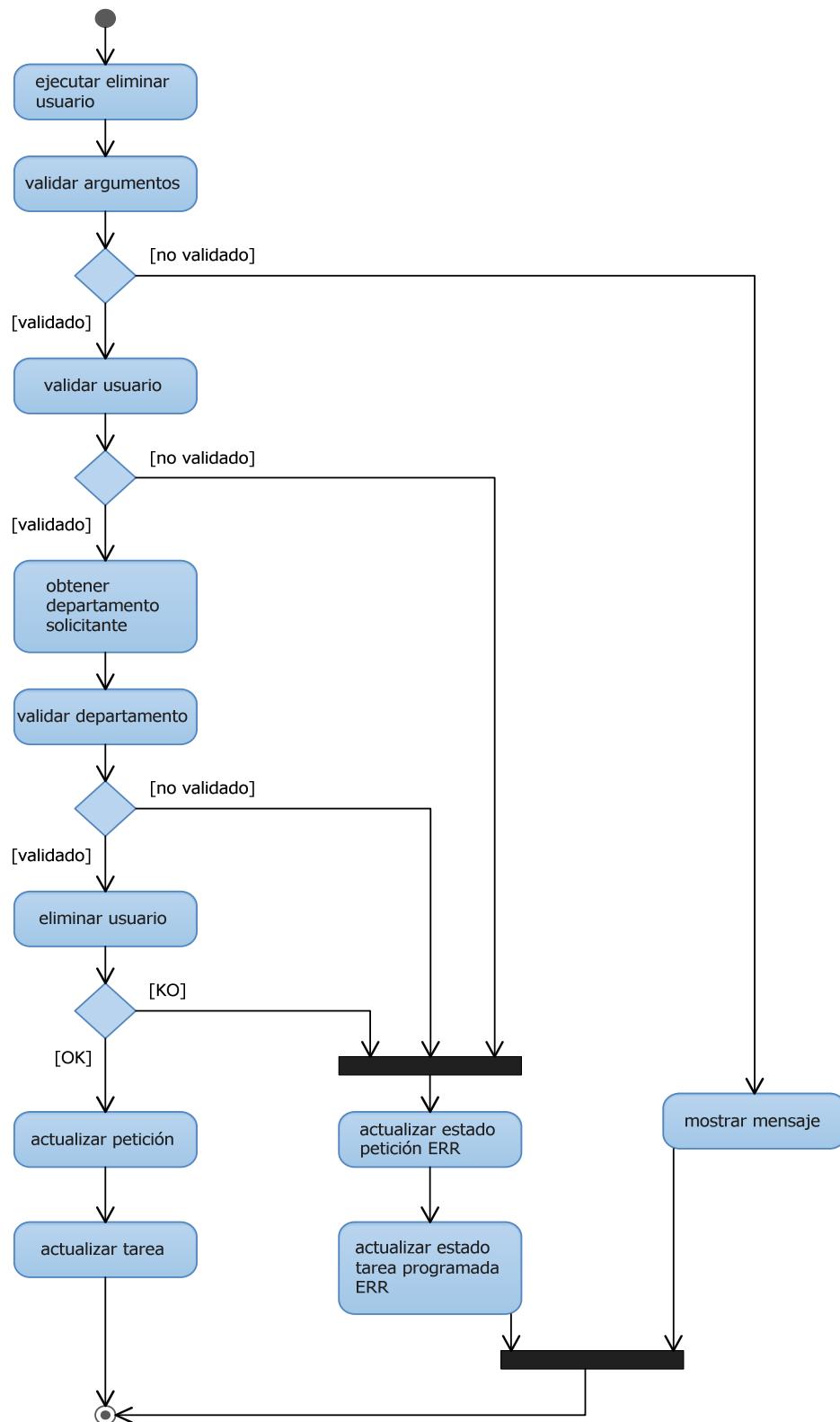


Figura 66. Módulo SUM Server: diagrama de secuencia eliminar usuario

ADSW003: Asignar perfil

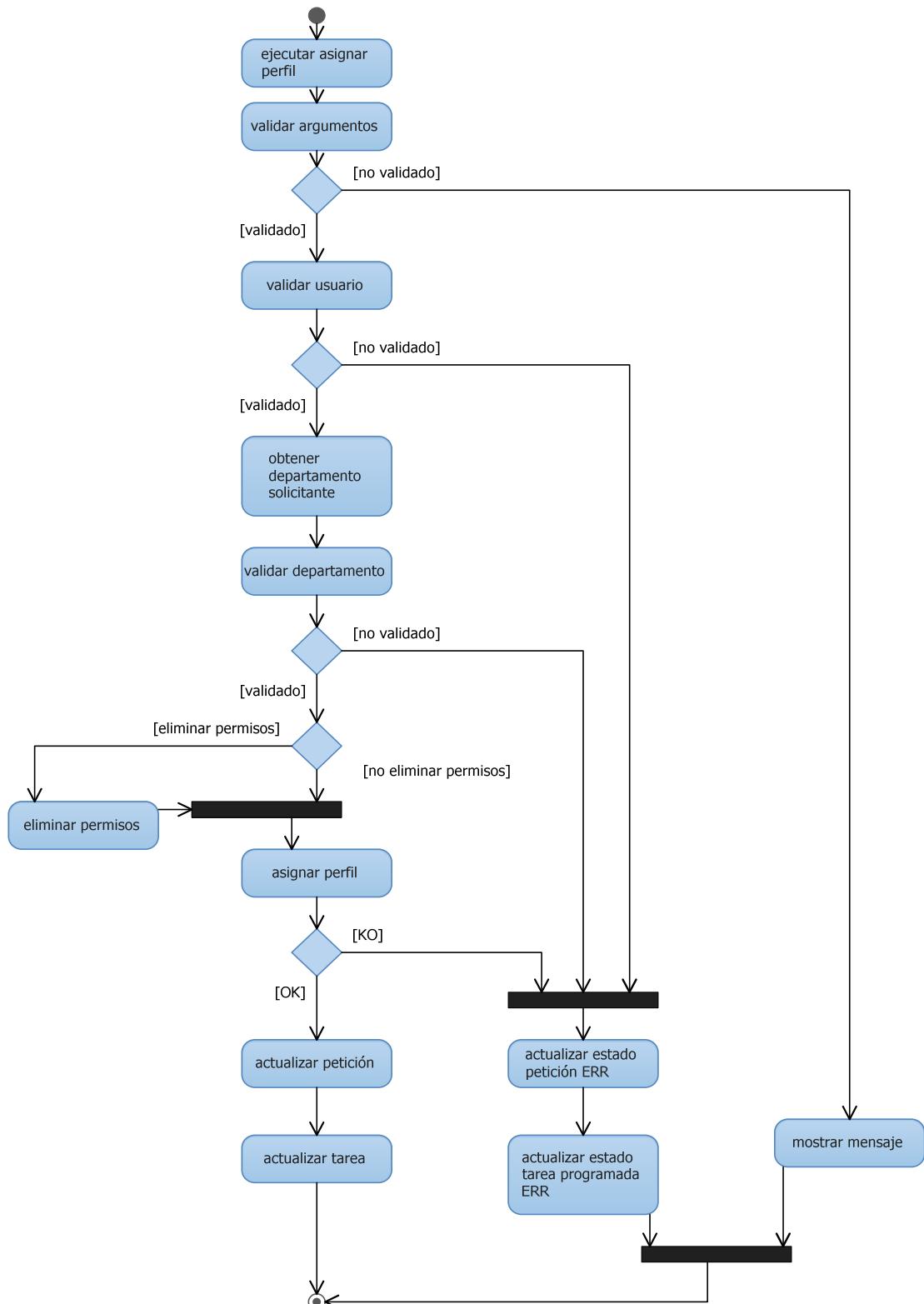


Figura 67. Módulo SUM Server: diagrama de secuencia asignar perfil

4.7 MODELO DE DATOS

A continuación se presentan los modelos entidad/relación y relacional que conforman el modelo de datos del sistema SUM. La finalidad del modelo de datos es proporcionar y facilitar de una manera adecuada el acceso a la información del sistema.

4.7.1 Modelo E/R

El modelo entidad/relación es un modelo de datos conceptual que muestra las entidades relevantes de un sistema de información así como la forma en la que se distribuyen y relacionan las diferentes tablas del sistema de información. La siguiente imagen muestra el modelo entidad/relación del sistema de gestión de usuarios SUM.

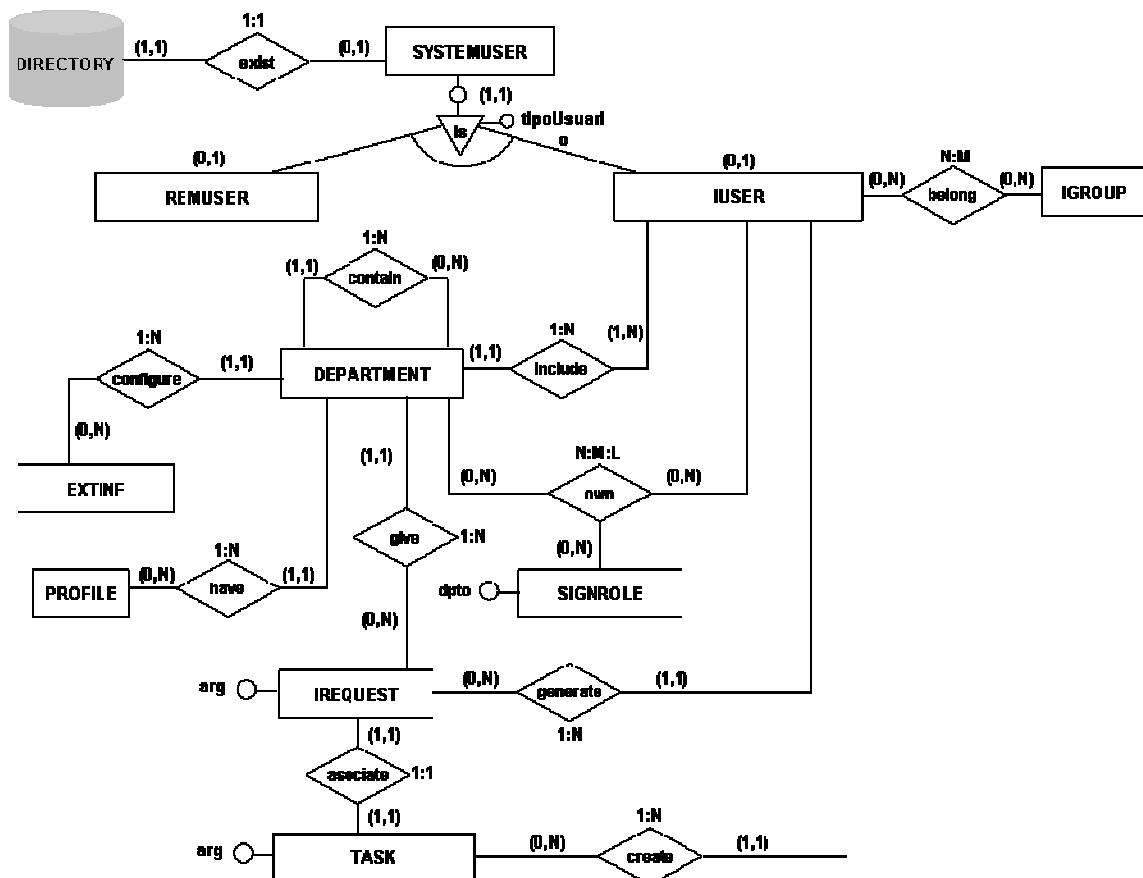


Figura 68. Modelo E/R

4.7.2 Modelo Relacional

El modelo relacional es un modelo de datos lógico que representa la transformación del diseño conceptual y su normalización para realizar un diseño físico de la base de datos.

La notación utilizada en el modelo relacional es la siguiente:

TIPO	SIGNIFICADO
<u>negrita</u>	Clave principal
*	Clave alternativa
-----	Atributo opcional
<i>cursiva</i>	Clave ajena
B:R	Borrado restringido
M:C	Modificación en cascada
B:M:C	Borrado y modificación en cascada

Tabla 42. Modelo relacional: notación

REMUSER (iduser, name*, remarks, date)

B:R, M:C

→ **IUSER (iduser, name* password, iddept, state, remarks, upddate, crdate, special)**

B:M:C

→ **DEPARTMENT (iddept, name*, parentid, remarks, upddate, crdate)**

B:M:C

EXTINF (idextinf, iddept, jpit)

→ **IGROUP (idgroup, name*, remarks, upddate, crdate)**

B:M:C

USERGROUP (iduser, idgroup)

B:R, M:C

PROFILE (idprofile, name*, iddept, sist, descr, def, upddate, crdate)

→ **SIGNROLE (idrole, post, range, showJPIT, upddate, crdate)**

B:M:C

B:M:C

SIGNROLEVALUE (idrole, iddept)

B:M:C

B:M:C

B:M:C

SIGNPERM (iduser, iddept, idrole)

B:R, M:C

→ **TASK (idtask, startDate, iduser, state, stateDate, result, upddate, crdate)**

B:M:C

TASKVALUE (idtask, numorder, arg)

B:M:C

B:R, M:C

B:R, M:C

→ **IREQUEST(idreq, iduser, iddept, nameuser, action, state, idtask, show, upddate, crdate)**

B:M:C

REQUESTVALUE (idreq, numorder, arg)

4.7.3 Descripción de jerarquías

Este apartado describe las jerarquías representadas en el modelo E/R.

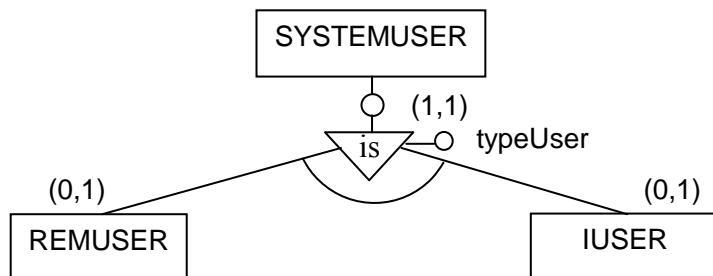


Figura 69. Jerarquía de usuarios

Se trata de una jerarquía total exclusiva que representa la clasificación de las distintas personas que existen en el sistema. Todas las personas existen bien en la entidad *remuser* o en la entidad *iuser* pero no pueden existir en ambas entidades a la vez. El atributo discriminante de la jerarquía es *typeUser* y tomará uno de los siguientes valores: *remuser* o *iuser*.

Se opta por representar en el modelo relacional las entidades *remuser* y *iuser* por separado para mantener una clasificación más detallada de los distintos tipos de usuario del sistema.

4.7.4 Descripción de entidades del sistema

A continuación se describen las entidades representadas en el Modelo E/R.

La notación utilizada en la columna Tipo para cada identificador de entidad es la siguiente:

ABREVIATURA	SIGNIFICADO
SP/SC	Simple/Compuesto
OB/OP	Obligatorio/Opcional
UNI	Univaluado
PK	Clave primaria
AK	Clave alternativa
FK	Clave ajena

Tabla 43. Modelo E/R: notación columna tipo

» **SYSTEMUSER:** Almacena información de todos los usuarios del sistema.

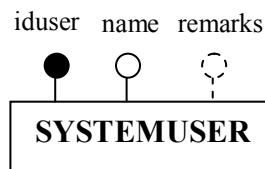


Figura 70. Entidad SYSTEMUSER

Está compuesta por los siguientes identificadores:

Entidad SYSTEMUSER			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
iduser	PK, SP, OB, UNI	int	Identificador de usuario
name	AK, SP, OB, UNI	varchar(32)	Nombre interno de usuario
remarks	SP, OP, UNI	varchar(254)	Comentarios
RESTRICCIONES			
Esta entidad no se representa debido a la falta de relaciones propias y a los pocos atributos que posee. Sus entidades hijas <i>IUser</i> y <i>Remuser</i> heredan sus atributos.			

Tabla 44. Entidad SYSTEMUSER

» **REMUSER:** Almacena información de los usuarios que han sido eliminados del sistema.

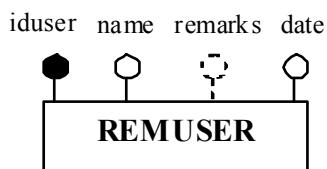


Figura 71. Entidad REMUSER

Está compuesta por los siguientes identificadores:

Entidad REMUSER			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
iduser	PK, SP, OB, UNI	int	Identificador de usuario
name	AK, SP, OB, UNI	varchar(32)	Nombre interno de usuario eliminado
remarks	SP, OP, UNI	varchar(254)	Comentarios
date	SP, OB, UNI	datetime	Fecha de eliminación del usuario
RESTRICCIONES			
Los atributos <i>iduser</i> , <i>name</i> y <i>remarks</i> son heredados de la entidad <i>Systemuser</i> .			

Tabla 45. Entidad REMUSER

» **IUSER:** Almacena información de los usuarios activos del sistema.

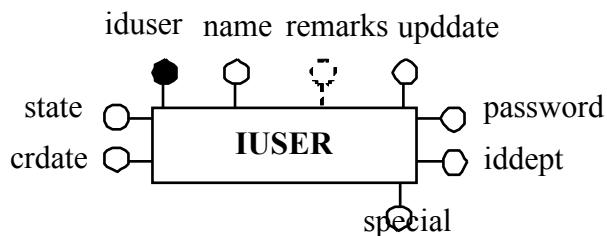


Figura 72. Entidad IUSER

Está compuesta por los siguientes identificadores:

Entidad IUSER			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
iduser	PK, SP, OB, UNI	int	Identificador de usuario
name	AK, SP, OB, UNI	varchar(32)	Nombre interno de usuario
remarks	SP, OP, UNI	varchar(254)	Comentarios
crdate	SP, OB, UNI	datetime	Fecha de creación del usuario
upddate	SP, OB, UNI	datetime	Fecha de última actualización del usuario
password	SP, OB, UNI	varchar(68)	Clave interna de usuario
iddept	FK, SP, OB, UNI	int	Identificador del departamento al que pertenece el usuario
state	SP, OB, UNI	int	Indica si el usuario se encuentra o no bloqueado en el sistema
special	SP, OP, UNI	varchar(1)	Flag (S/N) que indica si se trata de un usuario de tipo especial ('S') o no ('N')

RESTRICCIONES

Los atributos *iduser*, *name* y *remarks* son heredados de la entidad *Systemuser*.

El atributo *iduser* es un identificador generado de forma secuencial.

El atributo *state* tomará valor 1 para usuarios bloqueados y valor 0 en caso contrario.

Tabla 46. Entidad IUSER

❖ **DEPARTMENT:** Almacena información de los departamentos existentes en el sistema.

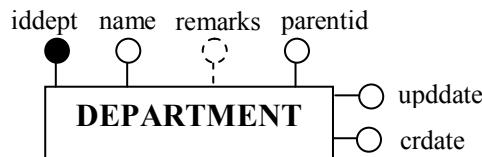


Figura 73. Entidad DEPARTMENT

Está compuesta por los siguientes identificadores:

Entidad DEPARTMENT						
ATRIBUTO	TIPO	FORMATO	DESCRIPCION			
iddept	PK, SP, OB, UNI	int	Identificador de departamento			
name	AK, SP, OB, UNI	varchar(128)	Nombre interno de departamento			
remarks	SP, OP, UNI	varchar(254)	Comentarios			
parentid	FK, SP, OP, UNI	int	Identificador del departamento padre			
upddate	SP, OB, UNI	datetime	Fecha de última actualización del departamento			
crdate	SP, OB, UNI	datetime	Fecha de creación del departamento			
RESTRICCIONES						
El atributo <i>iddept</i> es un identificador generado de forma secuencial.						

Tabla 47. Entidad DEPARTMENT

» **EXTINF:** Almacena información extra de los diferentes departamentos.

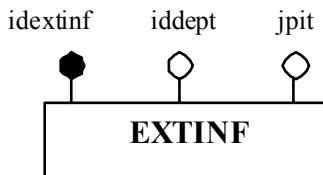


Figura 74. Entidad EXTINF

Está compuesta por los siguientes identificadores:

Entidad EXTINF			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
idextinf	PK, SP, OB, UNI	int	Identificador de información
iddept	FK, SP, OB, UNI	int	Identificador de departamento
jpit	SP, OB, UNI	varchar(1)	Flag (S/N), indica si el departamento es jefatura provincial ('S') o servicio central ('N')
RESTRICCIONES			
El atributo <i>idextinf</i> es un identificador generado de forma secuencial.			
El atributo <i>jpit</i> tomará valor S si el departamento es una jefatura provincial o N en si es un servicio central.			

Tabla 48. Entidad EXTINF

» **IGROUP:** Almacena información de los grupos existentes en el sistema. Un grupo contiene permisos para realizar una determinada acción en el sistema.

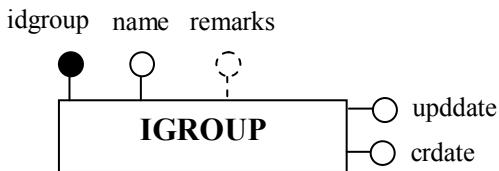


Figura 75. Entidad IGROUOP

Está compuesta por los siguientes identificadores:

Entidad GROUP			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
idgroup	PK, SP, OB, UNI	int	Identificador de grupo
name	AK, SP, OB, UNI	varchar(32)	Nombre interno de grupo
remarks	SP, OP, UNI	varchar(254)	Comentarios
upddate	SP, OB, UNI	datetime	Fecha de última actualización del grupo
crdate	SP, OB, UNI	datetime	Fecha de creación del grupo
RESTRICCIONES			
El atributo <i>idgroup</i> es un identificador generado de forma secuencial.			

Tabla 49. Entidad GROUP

» **PROFILE:** Almacena información de los perfiles existentes en el sistema. Un perfil contiene un conjunto de permisos (grupos) para realizar determinadas acciones en el sistema. Al asignar un perfil se asignan en bloque los grupos que conforman el perfil.

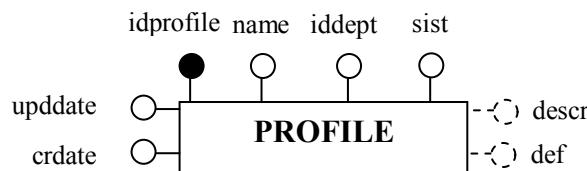


Figura 76. Entidad PROFILE

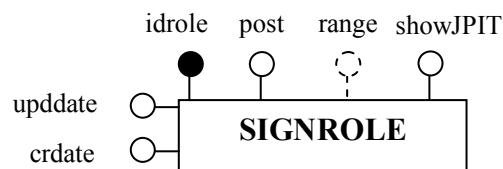
Está compuesta por los siguientes identificadores:

Entidad PROFILE			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
idprofile	PK, SP, OB, UNI	int	Identificador de perfil
name	AK, SP, OB, UNI	varchar(32)	Nombre interno de perfil
iddept	FK, SP, OB, UNI	int	Identificador de departamento al que pertenece
sist	SP, OB, UNI	varchar(1)	Flag (S/N) que indica si se trata de un perfil especial ('S') o básico ('N')
descr	SP, OP, UNI	varchar(250)	Descripción del perfil
def	SP, OP, UNI	text	Definición del perfil en formato XML
upddate	SP, OB, UNI	datetime	Fecha de última actualización del perfil

crdate	SP, OB, UNI	datetime	Fecha de creación del perfil
RESTRICCIONES			
El atributo <i>idprofile</i> es un identificador generado de forma secuencial.			
El atributo <i>sist</i> tomará valor S si se trata de un perfil especial y N en caso contrario.			
El formato del perfil definido en el campo <i>def</i> debe seguir el formato especificado en el anexo “Formato del XML de descripción de perfiles”.			

Tabla 50. Entidad PROFILE

» **SIGNROLE:** Almacena información de los roles que se pueden adoptar como firmantes de documentos.

**Figura 77. Entidad SIGNROLE**

Está compuesta por los siguientes identificadores:

Entidad SIGNROLE			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
idrole	PK, SP, OB, UNI	int	Identificador de rol
post	SP, OB, UNI	varchar(100)	Cargo del firmante del documento
range	SP, OP, UNI	varchar(100)	Texto descriptivo que establece en qué ámbito de tramitación se aplica este rol
showJPIT	SP, OB, UNI	varchar(1)	Flag (S/N) que indica si los administradores de las Jefaturas Provinciales pueden dar permisos de paso a firma de documentos dirigidos a este rol
upddate	SP, OB, UNI	datetime	Fecha de última actualización del rol
crdate	SP, OB, UNI	datetime	Fecha de creación del rol
RESTRICCIONES			
El atributo <i>idrole</i> es un identificador generado de forma secuencial.			
El atributo <i>showJPIT</i> tomará valor S si los administradores de las Jefaturas provinciales pueden dar permisos de paso a firma de documentos dirigidos a este rol o			

N en caso contrario.

Tabla 51. Entidad SIGNROLE

TASK: Almacena información de las tareas programadas creadas para ejecutar acciones en el sistema. Las tareas programadas ejecutan las acciones solicitadas en las peticiones de usuario.

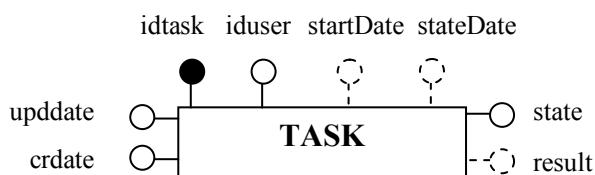


Figura 78. Entidad TASK

Está compuesta por los siguientes identificadores:

Entidad TASK				
ATRIBUTO	TIPO	FORMATO	DESCRIPCION	
idtask	PK, SP, OB, UNI	int	Identificador de tarea programada	
iduser	FK, SP, OB, UNI	int	Identificador de usuario que solicita la ejecución de la tarea	
state	SP, OB, UNI	varchar(3)	Estado de la tarea programada	
startDate	SP, OP, UNI	datetime	Fecha de comienzo de ejecución de la tarea.	
stateDate	SP, OP, UNI	datetime	Fecha de última actualización del estado de la tarea.	
result	SP, OP, UNI	varchar(1000)	Texto descriptivo del resultado de la tarea	
crdate	SP, OB, UNI	datetime	Fecha de creación de la tarea	
update	SP, OB, UNI	datetime	Fecha de actualización de la tarea	

Tabla 52. Entidad TASK

REQUEST: Almacena información de las peticiones de usuario creadas como solicitudes para ejecutar acciones en el sistema. Las peticiones van asociadas a tareas programadas para que puedan ejecutarse las acciones pertinentes.

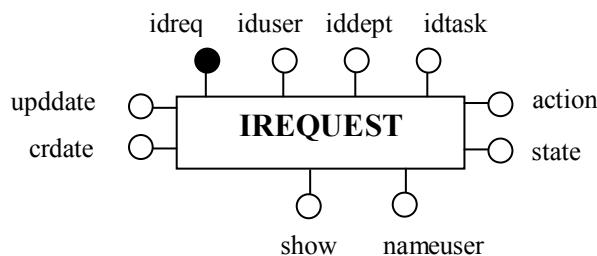


Figura 79. Entidad REQUEST

Está compuesta por los siguientes identificadores:

Entidad REQUEST			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
idreq	PK, SP, OB, UNI	int	Identificador de petición de usuario
iduser	FK, SP, OB, UNI	int	Identificador de usuario que genera la petición
iddept	FK, SP, OB, UNI	int	Identificador de departamento
nameuser	SP, OB, UNI	varchar(32)	Nombre interno de usuario sobre el que recae la acción de la petición.
action	SP, OB, UNI	varchar(1)	Acción solicitada.
state	SP, OB, UNI	varchar(3)	Estado de la petición
idtask	FK, SP, OB, UNI	int	Identificador de tarea programada asociada.
show	SP, OB, UNI	varchar(1)	Flag (S/N) que indica si se debe mostrar o no la petición.
upddate	SP, OB, UNI	datetime	Fecha de la última actualización de la petición.
crdate	SP, OB, UNI	datetime	Fecha de creación de la petición.

RESTRICCIONES
El atributo <i>idreq</i> es un identificador generado de forma secuencial.
El atributo <i>action</i> tomará valor:

- D para eliminar usuario,

- N para la creación de un nuevo usuario y
- P para la asignación de un perfil.

El atributo *state* tomará valor:

- ERR si finalizó su ejecución con error,
- OK si finalizó correctamente,
- PDT si está pendiente de ejecución o
- PRC si está en proceso.

El atributo *show* tomará valor S si se desea mostrar la petición o N en caso contrario. Por defecto las peticiones en estado OK tendrán este atributo con valor N.

Tabla 53. Entidad REQUEST

4.7.5. Descripción de entidades provenientes de campos multivaluados

A continuación se describen las entidades que provienen de la resolución de campos multivaluados.

» **SIGNROLEVALUE:** Almacena información de los departamentos a los que se aplican los roles de firmantes de documentos. Es la resolución del campo multivaluado “*iddept*” de la entidad “**SIGNROLE**”.

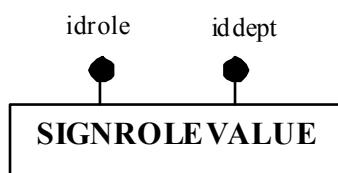


Figura 80. Entidad SIGNROLEVALUE

Está compuesta por los siguientes identificadores:

Entidad SIGNROLEVALUE			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
idrole	PK, FK, SP, OB, UNI	int	Identificador de rol firmante
iddept	PK, FK, SP, OB, UNI	int	Identificador de departamento
RESTRICCIONES			

Tabla 54. Entidad SIGNROLEVALUE

» **TASKVALUE:** Almacena información de los argumentos pasados como parámetros a las tareas programadas. Es la resolución del campo multivaluado “arg” de la entidad “TASK”.

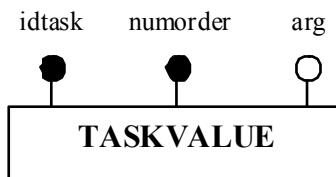


Figura 81. Entidad TASKVALUE

Está compuesta por los siguientes identificadores:

Entidad TASKVALUE			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
idtask	PK, FK, SP, OB, UNI	int	Identificador de tarea programada
numorder	PK, SP, OB, UNI	int	Número de orden del argumento
arg	SP, OB, UNI	varchar(254)	Argumento de tarea programada

RESTRICCIONES			
Se introduce como parte de la clave primaria un número de orden secuencial debido a que el campo “arg” es un texto que podría repetirse en diferentes tareas y por sí mismo no puede ser clave.			

Tabla 55. Entidad TASKVALUE

» **REQUESTVALUE:** Almacena información de los argumentos pasados como parámetros a las peticiones de usuario. Es la resolución del campo multivaluado “arg” de la entidad “REQUEST”.

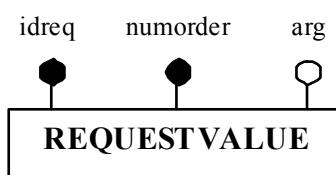


Figura 82. Entidad REQUESTVALUE

Está compuesta por los siguientes identificadores:

Entidad REQUESTVALUE			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
idreq	PK, FK, SP, OB, UNI	int	Identificador de petición de usuario
numorder	PK, SP, OB, UNI	int	Número de orden del argumento
arg	SP, OB, UNI	varchar(254)	Argumento de la petición de usuario
RESTRICCIONES			
Se introduce como parte de la clave primaria un número de orden secuencial debido a que el campo “arg” es un texto que podría repetirse en diferentes tareas y por sí mismo no puede ser clave.			

Tabla 56. Entidad REQUESTVALUE

4.7.6 Descripción de entidades provenientes de interrelaciones

A continuación se describen las entidades que provienen de la resolución de interrelaciones.

» **USERGROUP**: Almacena información de la asociación de los grupos a los que pertenecen los usuarios. Proviene de la interrelación “pertenece”.

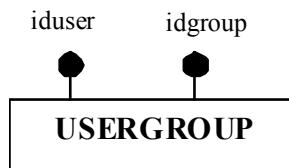


Figura 83. Entidad USERGROUP

Está compuesta por los siguientes identificadores:

Entidad USERGROUP			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
iduser	PK, FK, SP, OB, UNI	int	Identificador de usuario
idgroup	PK, FK, SP, OB, UNI	int	Identificador de grupo
RESTRICCIONES			

Tabla 57. Entidad USERGROUP

» **SIGNPERM:** Almacena información de la asociación entre los roles y los usuarios de los departamentos que pueden enviar documentos a firma a dicho rol. Proviene de la interrelación “posee”.

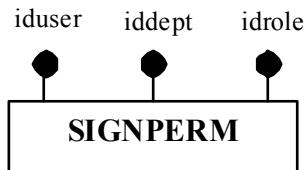


Figura 84. Entidad SIGNPERM

Está compuesta por los siguientes identificadores:

Entidad SIGNPERM			
ATRIBUTO	TIPO	FORMATO	DESCRIPCION
iduser	PK, FK, SP, OB, UNI	int	Identificador de usuario
iddept	PK, FK, SP, OB, UNI	int	Identificador de departamento
idrole	PK, FK, SP, OB, UNI	Int	Identificador de rol firmante
RESTRICCIONES			

Tabla 58. Entidad SIGNPERM

4.7.7 Modelo físico de datos

A continuación se presenta el diseño físico de la base de datos del sistema SUM.

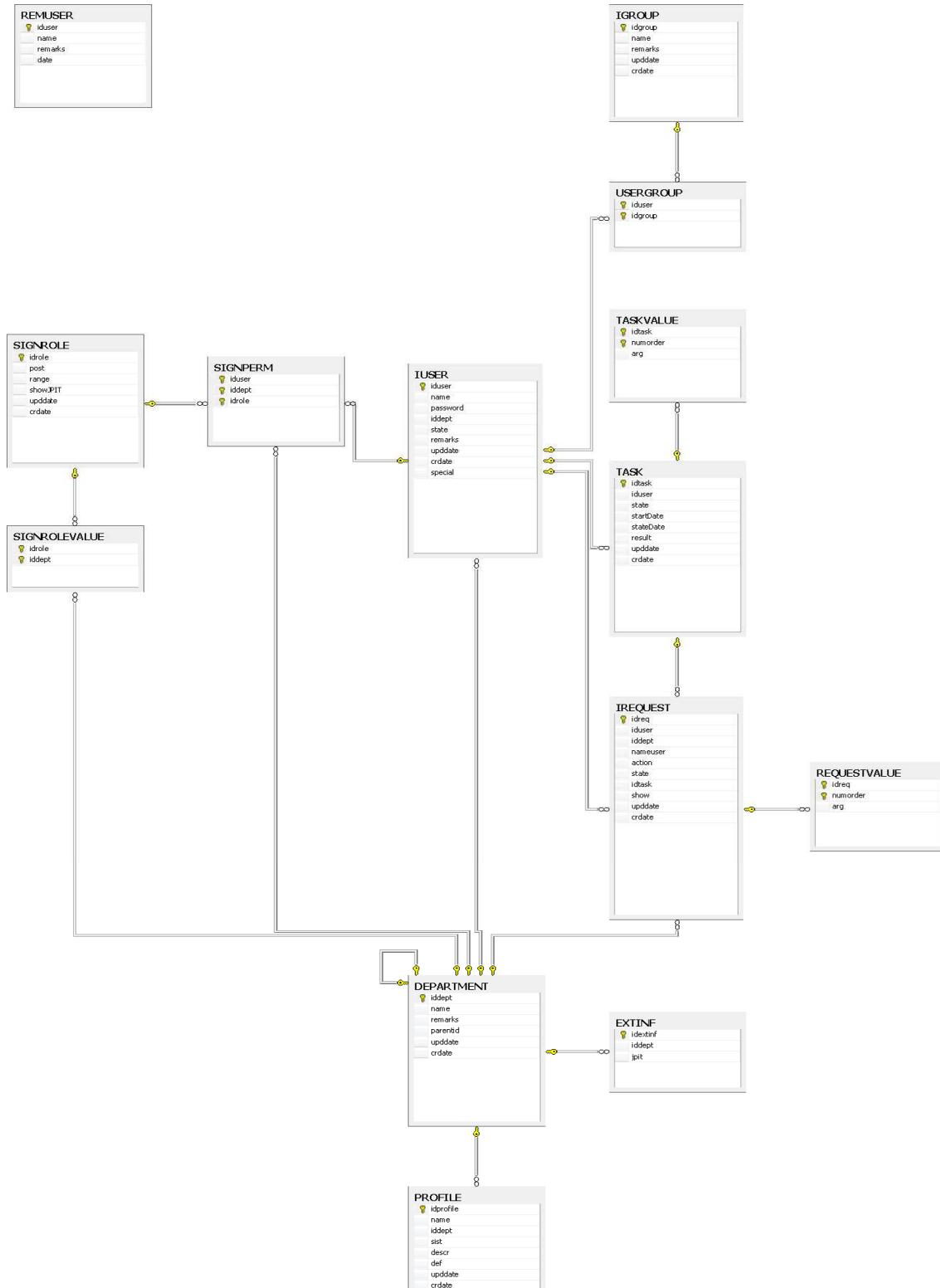


Figura 85. Modelo físico de datos

CAPÍTULO 5.

IMPLEMENTACIÓN

Este capítulo aborda la implementación del sistema SUM basándose en el análisis y diseño realizados en el capítulo anterior.

Como ya ha sido explicado, el sistema SUM gestiona a nivel departamental los diferentes usuarios y sus permisos de acceso al sistema GEXTEL. Se compone de dos módulos independientes SUM Web y SUM Server.

Ficheros generados en el módulo SUM Web

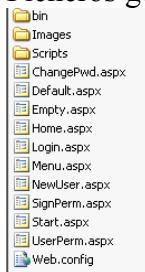


Figura 86. Ficheros módulo SUM Web

Ficheros generados en el módulo SUM Server

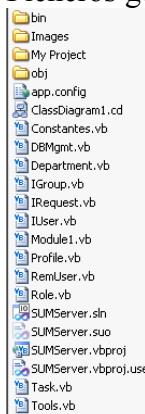


Figura 87. Ficheros módulo SUM Server

La implementación de los módulos del sistema se basa en el paradigma de programación orientado a objetos. La tecnología empleada para la construcción del sistema es Microsoft .NET, lenguaje Visual Basic (vb.NET). El sistema gestor de base de datos es SQL Server. El apartado [2.3](#) ofrece información detallada sobre las tecnologías utilizadas.

La extensión del sistema SUM medida en líneas de código fuente es la siguiente.

MÓDULO	TOTAL LÍNEAS
SUM Web	3.429
SUM Server	1215
	4.644

Tabla 59. Líneas de código

La estructura en términos de implementación del sistema se basa en el siguiente diagrama:

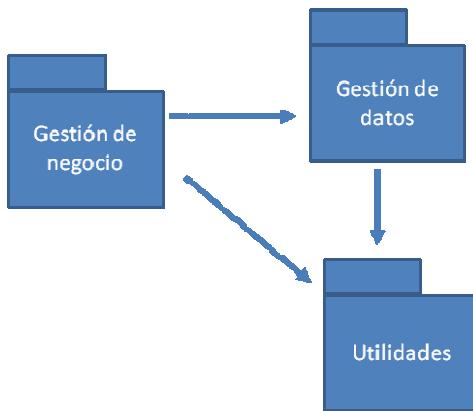


Figura 88. Diagrama paquetes

El paquete de **Utilidades** contiene una clase *Tools* que proporciona utilidades genéricas para el desarrollo. Facilita la creación de ficheros de log y permite mostrar mensajes al usuario.

El siguiente fragmento de código representa un método de la clase *Tools* que permite escribir información en un fichero de Log.

```

Public Sub WriteLog(ByVal pMsg As String)
    Dim path As String = "C:\LogFiles"
    Dim nameFile As String = "SUMWeb" & "_" &
    Format(System.DateTime.Now.Year, "0000") &
    Format(System.DateTime.Now.Month, "00") & Format(System.DateTime.Now.Day,
    "00") & ".txt"

    If Not System.IO.Directory.Exists(path) Then
        System.IO.Directory.CreateDirectory(path)
    End If

    OpenLog(path & "\" & nameFile)
    FichLog.WriteLine(Date.Now & " || " & pMsg)
    FichLog.Close()
    FichLog = Nothing
    fsoLog = Nothing

End Sub
  
```

El siguiente fragmento de código representa un ejemplo del contenido de un fichero de Log generado desde el módulo SUM Web.

```
07/08/2012 9:16:22 || Se ha mostrado el resultado de permisos y roles del usuario: shernandez  
07/08/2012 14:15:38 || Se ha mostrado el resultado de permisos y roles del usuario: lserrano  
07/08/2012 14:17:22 || Se ha mostrado el resultado de permisos y roles del usuario: vmoreno  
07/08/2012 15:31:26 || Creada tarea: 158, al crear el nuevo usuario: lvargas  
07/08/2012 15:31:26 || Creada petición: 120, al crear el nuevo usuario: lvargas  
07/08/2012 15:33:09 || Creada tarea: 159, para asignar el perfil: "AdministradorSP_DI", al usuario: lvargas  
07/08/2012 15:33:09 || Creada petición: 121, para asignar el perfil: "AdministradorSP_DI", al usuario: lvargas  
07/08/2012 15:34:16 || Se han eliminado las peticiones y tareas asignadas al nuevo usuario: lvargas  
07/08/2012 15:36:02 || Creada tarea: 160, al crear el nuevo usuario: lvargas  
07/08/2012 15:36:03 || Creada petición: 122, al crear el nuevo usuario: lvargas  
07/08/2012 15:36:17 || Creada tarea: 161, para eliminar al usuario: vmoreno  
07/08/2012 15:36:17 || Creada petición: 123, para eliminar al usuario: vmoreno  
07/08/2012 15:40:02 || Bloqueado el usuario: vmoreno  
07/08/2012 15:40:58 || Desbloqueado el usuario: vmoreno
```

El paquete **Gestión de datos** contiene una clase *DBMgmt* que hace uso de ADO.NET para el acceso a datos.

El siguiente fragmento de código de la clase *DBMgmt* representa cómo se establece la conexión a base de datos.

```
connStr = "Data Source=" & My.Settings.DBserver & ";Initial Catalog=" &  
My.Settings.DBdatabase & ";User ID=" & My.Settings.DBuser & ";Password=" &  
My.Settings.DBpwd  
conn = New SqlConnection(connStr)  
conn.Open()
```

Los archivos *Web.config* del módulo SUM Web y *app.config* del módulo SUM Server contienen el valor de los parámetros necesarios para el establecimiento de la conexión a la base de datos.

El siguiente fragmento de código pertenece al fichero *Web.config* del módulo SUM Web.

```
<applicationSettings>
  <SUMWeb.My.MySettings>
    <setting name="DBserver" serializeAs="String">
      <value>PCXXXX</value>
    </setting>
    <setting name="DBdatabase" serializeAs="String">
      <value>BDSUM</value>
    </setting>
    <setting name="DBuser" serializeAs="String">
      <value>pepe</value>
    </setting>
    <setting name="DBpwd" serializeAs="String">
      <value>pepe</value>
    </setting>
  </SUMWeb.My.MySettings>
</applicationSettings>
```

La información se actualiza en el sistema mediante el uso de transacciones para asegurar una actualización completa. El siguiente fragmento de código muestra un ejemplo de actualización de una petición de usuario.

```
Try
conn = DBMgmt.OpenDbConnection()
    If conn Is Nothing Then Throw New Exception("Ocurrió un error al establecer la
conexión de acceso a datos")

dateActual = Now()

cmdText = String.Format("UPDATE IREQUEST SET STATE='{0}', UPDDATE='{1}',
SHOW='N'      WHERE      IDREQ={2}",      getValueState(state),      dateActual,
Convert.ToString(_idreq))
    sqlt = conn.BeginTransaction()
    command = New SqlCommand(cmdText, conn)
    command.Transaction = sqlt
    res = command.ExecuteNonQuery()
    If res <= 0 Then
        Throw New Exception("Error al actualizar el estado de la petición con
identificador " & _idreq)
    End If
    _state = state
    _upddate = dateActual
    _show = False
    sqlt.Commit()
Catch ex As Exception
    sqlt.Rollback()
    Throw ex
Finally
    If Not command Is Nothing Then command.Dispose()
    If Not sqlt Is Nothing Then sqlt.Dispose()
    If Not conn Is Nothing AndAlso conn.State <> ConnectionState.Closed Then
        conn.Close()
End Try
```

Los diferentes permisos de acceso a GEXTEL se agrupan en perfiles. El contenido de un perfil se representa mediante una estructura de archivo XML ([ver anexo I](#)). Para el manejo de archivos XML se utiliza la clase *System.XmlDocument* de la librería *System.XML*.

El siguiente fragmento de código representa el acceso al contenido de un perfil.

```
xmlDoc.LoadXml(_def)
node = xmlDoc.DocumentElement.FirstChild

While Not node Is Nothing
    Select Case node.Name.ToUpper()
        Case valueGroup
            For i = 0 To node.Attributes.Count - 1
                If node.Attributes.Item(i).Name.ToUpper() = fieldGroup.ToUpper() Then
                    value = Convert.ToInt32(node.Attributes.Item(i).Value)
                    If value > 0 Then
                        IGroups.Add(value)
                    End If
                End If
            Next
        Case Else
            Exit Select
        End Select
    node = node.NextSibling
End While
```

El paquete **Gestión de negocio** contiene todas las clases de negocio del sistema. El contenido de este paquete es diferente para cada módulo del sistema (SUM Web y SUM Server). Representa el contenido del diagrama de clases correspondiente.

De manera genérica se utilizan módulos para la definición de constantes y formularios web que hacen uso de las clases de negocio.

Los métodos y funciones siguen una estructura común que permite el manejo de excepciones apoyándose en la clase System.Exception.

```
Try
Catch ex As Exception
    Throw New Exception(ex.Message.ToString)
End Try
```

5.1 IMPLEMENTACIÓN DEL MÓDULO SUM Web

Este módulo desarrollado en arquitectura Web implementa la interfaz de usuario del sistema SUM. Permite ejecutar acciones *on line* sobre los usuarios de un departamento y establecer solicitudes de acciones diferidas.

Acciones *on line*

- Bloqueo / desbloqueo de usuarios
- Cambio de contraseñas
- Asignación de permisos de firma
- Consulta de permisos e información de usuario
- Consulta de información de perfiles
- Consulta de peticiones generadas
- Eliminación de peticiones

Acciones diferidas

- Alta de nuevos usuarios
- Asignación de perfiles predefinidos a usuarios
- Eliminación de usuarios

En la implementación del módulo SUM Web cabe destacar que el código ha sido diseñado utilizando una estrategia de comportamiento basada en “conurrencia optimista” [2]. Esta estrategia se utiliza en un entorno multiusuario y permite actualizar información en una base de datos, de manera que cuando un usuario desea actualizar información, la aplicación determine si otro usuario la ha modificado o no desde que se leyó. En un modelo de concurrencia optimista, se considera que ha habido una infracción si, después de que un usuario recibe un valor de la base de datos, otro usuario modifica el valor antes de que el primer usuario haya intentado modificarlo.

La concurrencia optimista suele utilizarse en entornos con poca contención de datos. Esto mejora el rendimiento porque no es necesario bloquear registros (caso de concurrencia pesimista), permitiendo que el servidor pueda atender a un mayor número de clientes en menor tiempo.

Para implementar este comportamiento se compara la información recuperada de base de datos al cargar un objeto con la información recuperada de base de datos en el momento de la actualización. Si no hay diferencias se procede a realizar la actualización en la base de datos.

El siguiente fragmento de código muestra un ejemplo.

```

If _modified Then
    'a la hora de modificar un usuario primero hacer una select con las mismas
    características que tiene el objeto instanciado, es decir, todos los campos que no hayan sido
    modificados en mi instancia debo compararlos... Los objetos de mi instancia son los últimos que
    deseo guardar en BD por tanto no se comparan.
    'Si me devuelve algún registro la select es que no se ha modificado, en caso contrario
    alguien ha modificado ese registro en(BD) y por tanto no debo permitir realizar la operación,
    hay que lanzar excepción.
    conn = DBMgmt.OpenDbConnection()
    If conn Is Nothing Then Throw New Exception("Ocurrió un error al establecer la
    conexión de acceso a datos")
    cmdText = "SELECT COUNT(*) FROM IUSER WHERE IDUSER=" &
    Convert.ToString(_id)

    If _name = _origUser.name Then
        cmdText = cmdText & " AND NAME=''' & _name & '''"
    End If
    If _pwd = _origUser.pwd Then
        cmdText = cmdText & " AND PASSWORD=''' & _pwd & '''"
    End If

    ...
    command = New SqlCommand(cmdText, conn)
    dr = command.ExecuteReader(CommandBehavior.SingleResult)
    While dr.Read()
        value = IIf(dr.GetInt32(0) <> 1, -1, 0)
    End While
    dr.Close()

    If value = 0 Then
        'SE CARGAN LOS VALORES DEL OBJETO EN BD
        cmdText = String.Format("UPDATE IUSER SET NAME=''' & _name & ''',
        PASSWORD=''' & _pwd & ''', IDDEPT=''' & _iddept & ''', STATE=''' & Convert.ToInt32(_state)
        & ''', REMARKS=''' & _remarks & ''', UPDDATE=''' & Now() & ''', SPECIAL=''' &
        IIf(_special, "S", "N") & '' WHERE IDUSER=''' & _id)
        sqlt = conn.BeginTransaction()
        Try
            command = New SqlCommand(cmdText, conn)
            command.Transaction = sqlt
            res = command.ExecuteNonQuery()
            If res <= 0 Then
                Throw New Exception("Error al grabar datos de usuario")
            End If
            _modified = False
            reload()
            sqlt.Commit()
        Catch ex As Exception
            sqlt.Rollback()
            Throw ex
        End Try
    End If
End If

```

Los apartados [4.6.1.4](#) y [4.6.1.5](#) exponen información detallada de las clases que conforman el paquete de negocio del módulo SUM Web. A continuación se especifican detalles de la implementación de estas clases.

- *IUser*: clase que representa la información de un usuario del sistema. Contiene métodos para permitir el cambio de contraseñas, bloqueo y desbloqueo de usuario, validación de usuarios en el sistema, comprobación de pertenencia a grupos y roles, obtención de roles y grupos de un usuario.

La librería System.DirectoryServices facilita el uso de LDAP para la validación de usuarios en el directorio de la empresa. El siguiente fragmento de código muestra cómo se efectúa la validación mediante la clase System.DirectoryServices de esta librería.

```
entry = New System.DirectoryServices.DirectoryEntry("LDAP://" &
Environment.UserDomainName)
mySearcher = New System.DirectoryServices.DirectorySearcher(entry)
mySearcher.Filter = "(anr=" & nameUser & ")"
For Each result In mySearcher.FindAll()
    Try
        txt =
        result.GetDirectoryEntry().Properties.Item("userPrincipalName").Item(0)

        txt = txt.ToLower
        pos = txt.IndexOf("@")
        If pos >= 0 Then txt = txt.Substring(0, pos)
        If nameUser.ToLower = txt Then
            found = True
            fullName = result.GetDirectoryEntry().Properties.Item("name").Item(0)
        End If
    Catch x As Exception
    End Try
End If
```

- *Department*: clase que representa la información de un departamento del sistema. Contiene métodos para obtener información extendida, así como la lista de usuarios, roles y perfiles del departamento.

El siguiente fragmento de código se corresponde con la función que obtiene la lista de usuarios del departamento.

```

Public Function getListUser() As Hashtable
    Dim conn As SqlClient.SqlConnection = Nothing
    Dim command As SqlClient.SqlCommand = Nothing
    Dim dr As SqlClient.SqlDataReader = Nothing
    Dim cmdText As String = String.Empty
    Dim userId As Int32 = 0
    Dim users As Hashtable = Nothing

    Try
        users = New Hashtable
        conn = DBMgmt.OpenDbConnection()
        If conn Is Nothing Then Throw New Exception("Ocurrió un error al establecer la conexión de acceso a datos")
        cmdText = "SELECT IDUSER FROM IUSER WHERE IDDEPT=" &
        Convert.ToString(_iddept)
        command = New SqlClient.SqlCommand(cmdText, conn)
        dr = command.ExecuteReader(CommandBehavior.SingleResult)
        While dr.Read()
            userId = dr.GetInt32(0)
            'users.Add(userId, Nothing)
            users.Add(userId, New IUser(userId))
        End While

        Return users
    Catch ex As Exception
        Return Nothing
        Throw ex
    Finally
        If Not command Is Nothing Then command.Dispose()
        If Not conn Is Nothing AndAlso conn.State <> ConnectionState.Closed Then
            conn.Close()
            If Not dr Is Nothing AndAlso Not dr.IsClosed Then dr.Close()
        End Try
    End Function

```

- *Profile*: clase que representa la información de un perfil del sistema. Contiene métodos para obtener información del perfil y su contenido.
- *IGroup*: clase que representa la información de un grupo de permisos del sistema.
- *Role*: clase que representa la información de un rol firmante del sistema.
- *IRequest*: clase que representa la información de una petición de usuario del sistema. Contiene métodos para crear, eliminar, validar y obtener información de peticiones. Una petición puede contener múltiples argumentos.

El atributo args de la clase *IRequest* contiene los argumentos de la petición. Se define de la siguiente manera:

```
Public args As List(Of String)
```

La clase *IRequest* hereda de la clase *CollectionBase* permitiendo así redefinir y adaptar métodos de manipulación de listas. El siguiente fragmento de código muestra la adaptación de la función para añadir valores a una lista.

```
Public Function Add(ByVal value As String) As Integer
    _modified = True
    Return List.Add(value)
End Function 'Add
Public args As List(Of String)
```

- *Task*: clase que representa la información de una tarea programada del sistema. Contiene métodos para crear, eliminar y obtener información de tareas. Una tarea puede contener múltiples argumentos. Los argumentos de una tarea son una lista de valores. Hereda de *CollectionBase* para permitir la manipulación adaptada de los métodos para listas.
- *Encryption*: clase para el cifrado de datos.

El esquema de cifrado de datos que sigue es MD5. El apartado [2.2](#) proporciona información detallada sobre este método. La librería **System.Security.Cryptography** proporciona servicios criptográficos, entre los que se incluyen la codificación y descodificación segura de los datos.

A continuación se ofrece un fragmento de código que representa la función empleada para el cifrado de contraseñas.

```
Public Shared Function GetValueAsMd5(ByVal pwd As String) As String
    Dim provider As System.Security.Cryptography.MD5 = New
    System.Security.Cryptography.MD5CryptoServiceProvider()
    Dim passwordBytes() As Byte
    Dim md5Bytes() As Byte
    Dim passwordAsMd5 As StringBuilder = New StringBuilder()
    Dim user As IUser = Nothing

    Try
        If String.IsNullOrEmpty(pwd) Then
            Throw New ArgumentException("La password de usuario no puede tener valor
nulo ni estar vacío.", "userName")
        End If

        passwordBytes = Encoding.UTF8.GetBytes(pwd)
        md5Bytes = provider.ComputeHash(passwordBytes)

        For Each md5Byte As Byte In md5Bytes
            passwordAsMd5.AppendFormat("{0:x2}", md5Byte)
        Next

        Return passwordAsMd5.ToString()
    Catch ex As Exception
        'oTools.showMessage(Me, "Error: " & ex.Message.ToString, True)
        Return String.Empty
    End Try
End Function
```

5.2 IMPLEMENTACIÓN DEL MÓDULO SUM Server

Este módulo de servidor se encarga de ejecutar las acciones diferidas solicitadas mediante peticiones de usuario desde el módulo SUM Web.

Acciones diferidas

- Alta de nuevos usuarios
- Asignación de perfiles predefinidos a usuarios
- Eliminación de usuarios

Los apartados [4.6.2.4](#) y [4.6.2.5](#) exponen información detallada de las clases que conforman el paquete de negocio del módulo SUM Web. A continuación se especifican detalles de la implementación de estas clases.

- *IUser*: clase que representa la información de un usuario del sistema. Contiene métodos para crear, eliminar, validar, asignar grupos y roles así como obtener información de usuario.

```

Public Sub deleteUser()
    Dim conn As SqlClient.SqlConnection = Nothing
    Dim sqlt As System.Data.SqlClient.SqlTransaction = Nothing
    Dim command As SqlClient.SqlCommand = Nothing
    Dim res As Integer = 0
    Dim cmdText As String = String.Empty

    Try
        conn = DBMgmt.OpenDbConnection()
        If conn Is Nothing Then Throw New Exception("Ocurrió un error al establecer la conexión de acceso a datos")
        cmdText = String.Format("DELETE IUSER WHERE IDUSER=" &
Convert.ToString(_id))
        sqlt = conn.BeginTransaction()
        command = New SqlClient.SqlCommand(cmdText, conn)
        command.Transaction = sqlt
        res = command.ExecuteNonQuery()
        If res <= 0 Then
            Throw New Exception("Error al eliminar el usuario " & _name)
        End If
        sqlt.Commit()
    Catch ex As Exception
        sqlt.Rollback()
        Throw ex
    Finally
        If Not command Is Nothing Then command.Dispose()
        If Not sqlt Is Nothing Then sqlt.Dispose()
        If Not conn Is Nothing AndAlso conn.State <> ConnectionState.Closed Then
conn.Close()
    End Try
End Sub

```

El siguiente fragmento de código corresponde al método de eliminar usuario.

- *RemUser*: clase que representa la información de un usuario eliminado del sistema. Permite mantener un histórico de usuarios eliminados. Contiene métodos para la creación de este tipo de usuarios.

El siguiente fragmento de código corresponde a la función crear histórico de usuario eliminado.

```

Public Shared Function createUser(ByVal name As String, ByVal remarks As String) As
RemUser
    Dim usr As RemUser
    Dim idu As Integer

    Try
        usr = New RemUser
        usr._name = name
        usr._remarks = remarks

        idu = usr.store()

        usr = Nothing
        usr = New RemUser(idu)

        Return usr
    Catch ex As Exception
        Return Nothing
        Throw ex
    End Try

End Function

```

- *Profile*: clase que representa la información de un perfil del sistema. Contiene métodos para obtener información del perfil y su contenido.
- *IRequest*: clase que representa la información de una petición de usuario del sistema. Contiene métodos para actualizar y obtener información de peticiones de usuario.

El siguiente fragmento de código muestra la sentencia de actualización del estado de una petición de usuario.

```

dateActual = Now()

cmdText = String.Format("UPDATE IREQUEST SET STATE='{0}', UPDDATE='{1}', 
SHOW='N' WHERE IDREQ={2}", getValueState(state), dateActual,
Convert.ToString(_idreq))

```

El tipo de una petición se define en el siguiente fragmento de código.

```

Case ReqAction.P
    value = "P"      ' representa la solicitud de asignar perfil a usuario
Case ReqAction.N
    value = "N"      ' representa la solicitud de crear nuevo usuario
Case ReqAction.D
    value = "D"      ' representa la solicitud de eliminar usuario

```

- *Task*: clase que representa la información de una tarea programada del sistema.
Contiene métodos para actualizar y obtener información de tareas programadas.

La recuperación del estado de una tarea se representa mediante el siguiente fragmento de código.

```

Case TaskState.PDT
    value = "PDT"    ' representa estado pendiente de ejecución
Case TaskState.ERR
    value = "ERR"    ' representa estado finalizado con error
Case TaskState.OK
    value = "OK"     ' representa estado finalizado correctamente
Case TaskState.PRC
    value = "PRC"    ' representa estado en proceso

```

El tipo de una tarea se define en el siguiente fragmento de código.

```

Case TaskAction.P
    value = "P"      ' representa asignar perfil a usuario
Case TaskAction.N
    value = "N"      ' representa crear nuevo usuario
Case TaskAction.D
    value = "D"      ' representa eliminar usuario

```

- *Department*: clase que representa la información de un departamento del sistema.
- *IGroup*: clase que representa la información de un grupo de permisos del sistema.
- *Role*: clase que representa la información de un rol firmante del sistema.

El apartado [4.6.2](#) y el [anexo III](#) exponen información detallada para la iniciación del sistema SUM Server.

CAPÍTULO 6.

VERIFICACIÓN

6.1 VERIFICACIÓN DEL SISTEMA SUM Web

Este capítulo aborda la verificación del sistema implementado. El objetivo es comprobar que las aplicaciones funcionan de manera correcta en base a los requisitos previamente establecidos.

La verificación se ha efectuado a diferentes niveles entre las fases de implementación y la implantación del sistema.

Durante la implementación del sistema se han realizado pruebas unitarias y funcionales para cada una de las funciones verificando el correcto funcionamiento del sistema y comprobando que cumple la funcionalidad establecida en los requisitos.

Finalizada la implementación se han realizado pruebas para comprobar el funcionamiento global de las aplicaciones utilizando diferentes datos de prueba. Se ha verificado y validado la funcionalidad global y se ha comprobado la correcta accesibilidad e iniciación de los diferentes módulos que conforman el sistema SUM.

A medida que las aplicaciones han sido probadas y evaluadas a nivel local se ha escalado la solución a diferentes entornos de pruebas hasta su implantación final en el servidor de producción. Los diferentes entornos utilizados para la realización de las pruebas tienen características *hardware* y *software* similares al entorno final, siendo el entorno de preproducción el más parecido al entorno de producción.



Figura 89. Entornos de prueba

Una vez realizada la implantación del sistema en el entorno de producción se ha efectuado la supervisión de la correcta accesibilidad, iniciación e inicialización del sistema.

6.1 Verificación del sistema SUM Web

A continuación se especifican los principales escenarios de pruebas para el sistema SUM Web.

Cada tabla tiene las siguientes propiedades:

- Identificador: identificador único de prueba.

PRFSWXXX → pruebas funcionales del módulo SUM Web donde XXX es un nº secuencial.

- Condición: descripción de condiciones necesarias para la realización de la prueba.
- Descripción: descripción de la prueba a realizar.
- Resultado: descripción del resultado esperado.
- Requisitos: indica los requisitos de software relacionados.
- Validación: indica si el resultado de la prueba ha sido o no satisfactorio.

PRFSW001	
Condición	Acceder al sistema con un usuario administrador.
Descripción	Comprobar que un usuario administrador accede al sistema.
Resultado	El usuario accede al sistema.
Requisitos	SR034, SR035
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 60. PRFSW001

PRFSW002	
Condición	Acceder al sistema con un usuario que no sea administrador.
Descripción	Comprobar que un usuario no administrador no accede al sistema.
Resultado	La aplicación debe mostrar un mensaje indicativo al usuario y no permitir el acceso a la aplicación.
Requisitos	SR034, SR035
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 61. PRFSW002

PRFSW003	
Condición	Acceder al sistema con un usuario administrador.
Descripción	Comprobar inicialización del sistema.
Resultado	La aplicación debe de mostrar una etiqueta de título de departamento, el nombre de la aplicación, un botón para salir, una lista para usuarios existentes, una lista para nuevos usuarios, una lista para perfiles básicos, una lista para perfiles especiales, espacio para la descripción del perfil, espacio para el contenido de un perfil, lista de peticiones si existe, cinco botones de acciones <i>on line</i> , tres botones de acciones diferidas y dos botones para la lista de peticiones, siendo correctos el aspecto y la colocación de los elementos de la ventana.

PRFSW003		
Requisitos	SR001, SR002, SR003, SR004, SR005, SR006, SR011, SR013, SR014, SR015	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 62. PRFSW003

PRFSW004		
Condición	Acceder al sistema con un usuario administrador.	
Descripción	Comprobar inicialización del sistema.	
Resultado	La aplicación debe inicializar los valores título, lista de nuevos usuarios, lista de usuarios existentes, perfiles básicos, perfiles especiales y peticiones según la información del departamento del usuario de acceso. No deben ser incluidos los usuarios especiales del departamento en la lista de usuarios existentes. Ningún elemento debe estar seleccionado y el único botón habilitado debe de ser el botón <i>Nuevo usuario</i> .	
Requisitos	SR001, SR002, SR003, SR004, SR005, SR006, SR011, SR012, SR013, SR014, SR015, SR040	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 63. PRFSW004

PRFSW005		
Condición	Acceder al sistema con un usuario administrador de un departamento que no tenga peticiones visibles.	
Descripción	Comprobar la lista de peticiones al iniciar el sistema.	
Resultado	La aplicación no debe de mostrar la lista de peticiones y en su caso debe de mostrar una etiqueta indicando que no existen peticiones.	
Requisitos	SR011, SR012	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 64. PRFSW005

PRFSW006		
Condición	Situarse en la ventana principal de la aplicación.	
Descripción	Salir de la aplicación.	
Resultado	Al salir de la aplicación se debe mostrar el menú inicial que da acceso a la aplicación.	
Requisitos		
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 65. PRFSW006

PRFSW007	
Condición	Situarse en la ventana principal de la aplicación.
Descripción	Seleccionar un usuario existente.
Resultado	Se resaltará la selección efectuada. Se habilitarán los botones de las acciones <i>on line</i> y el botón <i>Eliminar usuario</i> . Se mostrará información del usuario bajo la lista de usuarios existentes.
Requisitos	SR005, SR006, SR016
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 66. PRFSW007

PRFSW008	
Condición	Situarse en la ventana principal de la aplicación.
Descripción	Seleccionar un usuario bloqueado.
Resultado	Se habilitará el botón <i>Desbloquear usuario</i> .
Requisitos	SR005, SR024
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 67. PRFSW008

PRFSW009	
Condición	Situarse en la ventana principal de la aplicación.
Descripción	Seleccionar un usuario no bloqueado.
Resultado	Se habilitará el botón <i>Bloquear usuario</i> .
Requisitos	SR005, SR024
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 68. PRFSW009

PRFSW010	
Condición	Situarse en la ventana principal de la aplicación.
Descripción	Seleccionar un usuario de la lista de nuevos usuarios.
Resultado	Se resaltará la selección efectuada y se habilitará el botón <i>Eliminar usuario</i> .
Requisitos	SR005, SR006
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 69. PRFSW010

PRFSW011	
Condición	Situarse en la ventana principal de la aplicación.
Descripción	Comprobar que no es posible bloquear o desbloquear usuarios nuevos.
Resultado	Los botones <i>Bloquear usuario</i> y <i>Desbloquear usuario</i> no estarán habilitados al seleccionar un nuevo usuario.
Requisitos	SR005, SR025, SR027
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 70. PRFSW011

PRFSW012	
Condición	Situarse en la ventana principal de la aplicación.
Descripción	Comprobar que no es posible seleccionar dos usuarios o dos perfiles simultáneamente.
Resultado	Sólo se habilitará la última selección realizada.
Requisitos	SR036, SR037
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 71. PRFSW012

PRFSW013	
Condición	Situarse en la ventana principal de la aplicación.
Descripción	Comprobar que es posible seleccionar de manera simultánea un usuario y un perfil.
Resultado	Se resaltará el usuario y el perfil seleccionados de manera simultánea.
Requisitos	SR023
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 72. PRFSW013

PRFSW014	
Condición	Situarse en la ventana principal de la aplicación.
Descripción	Seleccionar un perfil.
Resultado	Se mostrará la descripción y el contenido del perfil en los espacios habilitados para ello. Si existe un usuario seleccionado se habilitará el botón <i>Asignar Perfil</i> .
Requisitos	SR005, SR017, SR023
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 73. PRFSW014

PRFSW015	
Condición	Situarse en la ventana principal de la aplicación.
Descripción	Seleccionar una petición.
Resultado	Se marcará el check correspondiente a la petición seleccionada en la lista de peticiones y se habilitará el botón <i>Eliminar</i> . En caso de que la petición seleccionada se encuentre en un estado finalizado con error, se habilitará el botón <i>Ver error</i> .
Requisitos	SR014, SR015
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 74. PRFSW015

PRFSW016	
Condición	Situarse en la ventana principal de la aplicación.
Descripción	Comprobar que al modificar las condiciones de habilitación de los botones activos deben desactivarse.
Resultado	Sólo podrá estar activo el botón Asignar perfil si se seleccionan de manera simultánea un usuario y un perfil. El botón Eliminar usuario sólo puede estar activo si existe un usuario seleccionado. El botón Eliminar sólo puede estar activo si existe una petición seleccionada. El botón Ver error sólo puede estar activo si existe una petición finalizada con error seleccionada. Los botones de acciones <i>on line</i> sólo pueden estar activos si existe un usuario existente seleccionado.
Requisitos	SR015, SR023
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 75. PRFSW016

PRFSW017	
Condición	Seleccionar un usuario existente.
Descripción	Bloquear usuario.
Resultado	Se solicitará confirmación al usuario. Si cancela no habrá cambios en el sistema, si confirma la acción el usuario quedará bloqueado y no podrá acceder al sistema GEXTEL. Aparecerá una marca de bloqueo junto al nombre del usuario en la lista de usuarios existentes.
Requisitos	SR005, SR020, SR024, SR025, SR026
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 76. PRFSW017

PRFSW018	
Condición	Seleccionar un usuario bloqueado.
Descripción	Desbloquear usuario.
Resultado	Se solicitará confirmación al usuario. Si cancela no habrá cambios en el sistema, si confirma la acción el usuario quedará desbloqueado y podrá acceder al sistema GEXTEL. Desaparecerá la marca de bloqueo junto al nombre del usuario en la lista de usuarios existentes.
Requisitos	SR005, SR020, SR024, SR027
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 77. PRFSW018

PRFSW019	
Condición	Seleccionar un usuario existente.
Descripción	Seleccionar botón <i>Cambiar contraseña</i> .
Resultado	Debe mostrarse la ventana de cambio de contraseñas con el nombre del usuario seleccionado.
Requisitos	SR005
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 78. PRFSW019

PRFSW020	
Condición	Seleccionar botón <i>Cambiar contraseña</i> .
Descripción	Modificar contraseña.
Resultado	Al confirmar la contraseña y pulsar Aceptar se solicitará confirmación al usuario. Si cancela, no se realizarán cambios en el sistema, si acepta la contraseña de acceso al sistema GEXTEL del usuario será actualizada. La contraseña actualizada en el sistema debe estar cifrada.
Requisitos	SR005, SR028, SR039
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 79. PRFSW020

PRFSW021	
Condición	Seleccionar botón <i>Cambiar contraseña</i> .
Descripción	Validar contraseña.
Resultado	Introducir una contraseña errónea en longitud, no confirmar contraseña o no llenar algún campo obligatorio supondrá que la aplicación muestre un mensaje de error al usuario y no efectúe cambios en el sistema.

PRFSW021

Requisitos	SR005	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 80. PRFSW021

PRFSW022

Condición	Seleccionar usuario existente.	
Descripción	Seleccionar botón <i>Permisos portafirmas</i> .	
Resultado	Debe mostrarse una ventana con los cargos firmantes asociados al departamento. Si no existen cargos asociados al departamento se mostrará un mensaje indicativo. En caso de que el usuario tenga asociado algún cargo firmante aparecerá seleccionado.	
Requisitos	SR005, SR029	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 81. PRFSW022

PRFSW023

Condición	Seleccionar usuario existente y el botón <i>Permisos portafirmas</i> .	
Descripción	Modificar los permisos portafirmas del usuario.	
Resultado	Si el usuario cancela no habrá cambios en el sistema, si pulsa el botón <i>Aceptar</i> se actualizará en el sistema la información relativa a los cargos firmantes asociados al usuario.	
Requisitos	SR005, SR029	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 82. PRFSW023

PRFSW024

Condición	Seleccionar usuario existente.	
Descripción	Seleccionar botón <i>Ver permisos usuario</i> .	
Resultado	Debe mostrarse una ventana con los permisos asociados al usuario. Si no tiene permisos se mostrará un mensaje indicativo. En caso contrario, se mostrará el nombre y la descripción de los grupos asociados al usuario.	
Requisitos	SR005, SR030	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 83. PRFSW024

PRFSW025		
Condición		
Descripción	Seleccionar botón <i>Nuevo usuario</i> .	
Resultado	Debe mostrarse la ventana de nuevo usuario.	
Requisitos	SR005, SR006	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 84. PRFSW025

PRFSW026		
Condición	Seleccionar botón <i>Nuevo usuario</i> .	
Descripción	Validar datos nuevo usuario.	
Resultado	Si los datos introducidos no son válidos, falta algún campo obligatorio o no se confirma la contraseña se mostrará un mensaje indicativo al usuario tras pulsar el botón <i>Aceptar</i> .	
Requisitos	SR005, SR006, SR019	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 85. PRFSW026

PRFSW027		
Condición	Seleccionar botón <i>Nuevo usuario</i> .	
Descripción	Crear usuario existente en el dominio de la empresa.	
Resultado	Al introducir el nombre del usuario el campo nombre completo será rellenado con el nombre completo del usuario. Si la información introducida es correcta y se pulsa el botón <i>Aceptar</i> se creará una petición de alta de usuario y la lista de peticiones y de nuevos usuarios serán actualizadas.	
Requisitos	SR005, SR006, SR007, SR008, SR010, SR011, SR019, SR041	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 86. PRFSW027

PRFSW028		
Condición	Seleccionar botón <i>Nuevo usuario</i> .	
Descripción	Crear usuario no existente en el dominio de la empresa.	
Resultado	Al introducir el nombre del usuario, el sistema mostrará un mensaje indicando que el usuario no existe en el dominio de la empresa y es posible solicitar su alta.	
Requisitos	SR005, SR006, SR018	

PRFSW028**Validación**

<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo
-----------------------------------	----------------------------------

Tabla 87. PRFSW028**PRFSW029****Condición**Seleccionar botón *Nuevo usuario*.**Descripción**

Crear usuario de dominio de empresa ya existente en el sistema GEXTEL.

Resultado

Al introducir el nombre del usuario, el sistema mostrará un mensaje indicando que el usuario ya existe y no es posible solicitar su alta.

Requisitos

SR005, SR006

Validación

<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo
-----------------------------------	----------------------------------

Tabla 88. PRFSW029**PRFSW030****Condición**

Seleccionar usuario existente.

Descripción

Eliminar usuario.

Resultado

Si se cancela la acción no se realizarán cambios en el sistema, si se confirma se creará una petición para eliminar el usuario y aparecerá una marca indicativa junto al nombre del usuario. La lista de peticiones se verá actualizada.

Requisitos

SR005, SR006, SR007, SR008, SR010, SR011, SR012, SR020, SR032

Validación

<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo
-----------------------------------	----------------------------------

Tabla 89. PRFSW030**PRFSW031****Condición**

Seleccionar usuario nuevo.

Descripción

Eliminar usuario.

Resultado

Si se cancela la acción no se realizarán cambios en el sistema, si se confirma se elimina el usuario de la lista así como las peticiones asociadas. La lista de peticiones se verá actualizada.

Requisitos

SR005, SR006, SR020, SR031

Validación

<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo
-----------------------------------	----------------------------------

Tabla 90. PRFSW031

PRFSW032

Condición	Seleccionar usuario y seleccionar botón <i>Asignar perfil</i> .	
Descripción	Asignar perfil básico a un usuario.	
Resultado	Si se cancela la acción no habrá cambios en el sistema, si se confirma se creará una petición de asignación de perfil y la lista de peticiones será actualizada. En la petición se incluirá el valor <i>True</i> para eliminar los permisos previos del usuario.	
Requisitos	SR005, SR006, SR007, SR008, SR010, SR011, SR012, SR020, SR021, SR023	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 91. PRFSW032

PRFSW033

Condición	Seleccionar usuario y seleccionar botón <i>Asignar perfil</i> .	
Descripción	Asignar perfil especial a un usuario.	
Resultado	Si se cancela la acción no habrá cambios en el sistema, si se confirma se creará una petición de asignación de perfil y la lista de peticiones será actualizada. En la petición se incluirá el valor <i>True</i> o <i>False</i> según la selección del usuario para eliminar los permisos previos.	
Requisitos	SR005, SR006, SR007, SR008, SR010, SR011, SR012, SR020, SR022, SR023	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 92. PRFSW033

PRFSW034

Condición	Seleccionar una petición finalizada con error.	
Descripción	Seleccionar botón <i>Ver error</i> .	
Resultado	Se mostrará información sobre el error producido.	
Requisitos	SR015	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 93. PRFSW034

PRFSW035

Condición	Seleccionar una petición.	
Descripción	Seleccionar botón <i>Eliminar</i> .	
Resultado	Si cancela no habrá cambios en el sistema, si confirma la petición será eliminada y la lista de peticiones será actualizada.	
Requisitos	SR011, SR012, SR014, SR020	

PRFSW035

Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo
-------------------	-----------------------------------	----------------------------------

Tabla 94. PRFSW035

PRFSW036

Condición	Seleccionar una petición de nuevo usuario.
Descripción	Seleccionar botón <i>Eliminar</i> .
Resultado	Si cancela no habrá cambios en el sistema, si confirma la petición será eliminada. El usuario será eliminado de la lista de nuevos usuarios. Si existen peticiones pendientes asociadas al mismo usuario serán eliminadas. La lista de peticiones será actualizada.
Requisitos	SR011, SR012, SR014, SR020, SR038
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 95. PRFSW036

PRFSW037

Condición	Seleccionar una petición de eliminar usuario.
Descripción	Seleccionar botón <i>Eliminar</i> .
Resultado	Si cancela no habrá cambios en el sistema, si confirma desaparecerá la marca del nombre de usuario en la lista de usuarios existentes. La petición será eliminada. La lista de peticiones será actualizada.
Requisitos	SR011, SR012, SR014, SR020, SR033
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 96. PRFSW037

PRFSW038

Condición	Crear una petición
Descripción	Validar la creación de peticiones.
Resultado	Se creará la petición solicitada si no existe otra petición con el mismo contenido.
Requisitos	SR008, SR010
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 97. PRFSW038

PRFSW039	
Condición	
Descripción	Comprobar el acceso a la aplicación.
Resultado	Se debe acceder a la aplicación desde el menú de GEXTEL.
Requisitos	
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 98. PRFSW039

6.2 VERIFICACIÓN DEL SISTEMA SUM Server

A continuación se especifican los principales escenarios de pruebas para el sistema SUM Server.

Cada tabla tiene las siguientes propiedades:

- Identificador: identificador único de prueba.

PRFSSXXX → pruebas funcionales del módulo SUM Server

- Condición: descripción de condiciones necesarias para la realización de la prueba.
- Descripción: descripción de la prueba a realizar.
- Resultado: descripción del resultado esperado.
- Requisitos: indica los requisitos de software relacionados.
- Validación: indica si el resultado de la prueba ha sido o no satisfactorio.

PRFSS001	
Condición	Solicitar acción.
Descripción	Validar parámetros.
Resultado	El sistema validará el nº de argumentos según la acción solicitada. En caso de no ser válido el sistema finalizará su ejecución con error, actualizando el estado y la fecha de modificación de la tarea y la petición correspondientes.
Requisitos	SR006, SR009, SR012
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 99. PRFSS001

PRFSS002	
Condición	Solicitar acción.
Descripción	Validar contraseña.
Resultado	El sistema validará que la contraseña recibida por parámetro se corresponda con la contraseña real del usuario solicitante. Si no coincide el sistema finalizará su ejecución con error, actualizando el estado y la fecha de modificación de la tarea y la petición correspondientes.
Requisitos	SR006, SR009, SR012
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 100. PRFSS002

PRFSS003	
Condición	Solicitar acción.
Descripción	Validar permisos de usuario solicitante.
Resultado	El sistema validará que los permisos del usuario solicitante sean suficientes para realizar la acción solicitada. Si no es así el sistema finalizará su ejecución con error, actualizando el estado y la fecha de modificación de la tarea y la petición correspondientes.
Requisitos	SR006, SR009, SR012
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 101. PRFSS003

PRFSS004	
Condición	Solicitar eliminación de usuario o asignación de perfil a usuario.
Descripción	Validar departamento de usuario solicitante.
Resultado	Si el departamento del usuario solicitante no se corresponde con el departamento del usuario a eliminar, el sistema finalizará su ejecución con error, actualizando el estado y la fecha de modificación de la tarea y la petición correspondientes.
Requisitos	SR006, SR009, SR012
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 102. PRFSS004

PRFSS005	
Condición	Solicitar alta de usuario.
Descripción	Crear usuario.
Resultado	Si las validaciones son correctas el sistema creará un nuevo usuario de

PRFSS005	
	tipo no especial en el departamento del usuario solicitante con los datos recibidos del nuevo usuario. La contraseña debe estar cifrada. Actualiza el estado y la fecha de actualización de la petición y tarea correspondientes. La petición se marca no visible.
Requisitos	SR005, SR006, SR009, SR012, SR019, SR041
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 103. PRFSS005

PRFSS006	
Condición	Solicitar eliminación de usuario.
Descripción	Eliminar usuario.
Resultado	Si las validaciones son correctas el sistema creará un usuario en la tabla <i>RemUser</i> , eliminará el usuario así como los roles y permisos asociados. Actualiza el estado y la fecha de actualización de la petición y tarea correspondientes. La petición se marca no visible.
Requisitos	SR005, SR006, SR009, SR012
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 104. PRFSS006

PRFSS007	
Condición	Solicitar asignar perfil a usuario con parámetro <i>True</i> para eliminar los permisos previos.
Descripción	Asignar perfil a usuario.
Resultado	Si las validaciones son correctas el sistema eliminará los permisos del usuario y asignará los del nuevo perfil. Actualiza el estado y la fecha de actualización de la petición y tarea correspondientes. La petición se marca no visible.
Requisitos	SR005, SR006, SR009, SR012, SR021, SR022
Validación	<input type="checkbox"/> Correcto <input type="checkbox"/> Erróneo

Tabla 105. PRFSS007

PRFSS008	
Condición	Solicitar asignar perfil a usuario con parámetro <i>False</i> para no eliminar los permisos previos.
Descripción	Asignar perfil a usuario.
Resultado	Si las validaciones son correctas el sistema asignará los permisos del nuevo perfil sin duplicar. Actualiza el estado y la fecha de actualización de la petición y tarea correspondientes. La petición se marca no visible.

PRFSS008

Requisitos	SR005, SR006, SR009, SR012, SR022	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 106. PRFSS008**PRFSS009**

Condición	Solicitar acción.	
Descripción	Comprobar error de las operaciones.	
Resultado	Si al realizar las operaciones sucede algún error, el sistema finaliza la ejecución y actualiza el estado y fecha de la tarea y petición correspondiente a finalizado con error.	
Requisitos	SR006, SR009, SR012	
Validación	<input type="checkbox"/> Correcto	<input type="checkbox"/> Erróneo

Tabla 107. PRFSS009

Además de las pruebas descritas se han realizado pruebas unitarias y de implantación del sistema para validar el correcto funcionamiento de los elementos de diseño de la interfaz, el formato correcto de los datos de entrada, los ficheros implantados en el servidor de producción, presentación de navegador, acceso correcto a las aplicaciones, etc.

Como conclusión tras la realización de las pruebas, la evaluación del sistema cumple con la funcionalidad y los requisitos establecidos. Por tanto, se han alcanzado los objetivos previstos de manera satisfactoria.

6.3 MATRIZ DE TRAZABILIDAD SR/PR

La siguiente tabla establece la relación entre los requisitos de usuario y los requisitos software del sistema. Como puede observarse todos los requisitos de usuario quedan cubiertos por requisitos software.

CAPÍTULO 6. EVALUACIÓN

PR/SR	S R 00 1	S R 00 2	S R 00 3	S R 00 4	S R 00 5	S R 00 6	S R 00 7	S R 00 8	S R 00 9	S R 01 0	S R 01 1	S R 01 2	S R 01 3	S R 01 4	S R 01 5	S R 01 6	S R 01 7	S R 01 8	S R 01 9	S R 02 0	S R 02 1	S R 02 2	S R 02 3	S R 02 4	S R 02 5	S R 02 6	S R 02 7	S R 02 8	S R 02 9	S R 03 0	S R 03 1	S R 03 2	S R 03 3	S R 03 4	S R 03 5	S R 03 6	S R 03 7	S R 03 8	S R 03 9	S R 04 0	S R 04 1
PRFSW001																																									
PRFSW002																																									
PRFSW003	✓	✓	✓	✓	✓	✓				✓		✓	✓	✓																											
PRFSW004	✓	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓																										
PRFSW005										✓	✓																														
PRFSW006																																									
PRFSW007					✓	✓												✓																							
PRFSW008						✓																																			
PRFSW009							✓																																		
PRFSW010								✓	✓																																
PRFSW011									✓																																
PRFSW012																																						✓	✓		
PRFSW013																																									
PRFSW014										✓																															
PRFSW015												✓	✓																												
PRFSW016													✓																												
PRFSW017														✓																											
PRFSW018															✓																										
PRFSW019																✓																									
PRFSW020																	✓																					✓			
PRFSW021																		✓																							

CAPÍTULO 6. EVALUACIÓN

PR/SR	S R 00 1	S R 00 2	S R 00 3	S R 00 4	S R 00 5	S R 00 6	S R 00 7	S R 00 8	S R 00 9	S R 01 0	S R 01 1	S R 01 2	S R 01 3	S R 01 4	S R 01 5	S R 01 6	S R 01 7	S R 01 8	S R 01 9	S R 02 0	S R 02 1	S R 02 2	S R 02 3	S R 02 4	S R 02 5	S R 02 6	S R 02 7	S R 02 8	S R 02 9	S R 03 0	S R 03 1	S R 03 2	S R 03 3	S R 03 4	S R 03 5	S R 03 6	S R 03 7	S R 03 8	S R 03 9	S R 04 0	S R 04 1					
PRFSW022																																														
PRFSW023																																														
PRFSW024																																														
PRFSW025																																														
PRFSW026																																														
PRFSW027																																														
PRFSW028																																														
PRFSW029																																														
PRFSW030																																														
PRFSW031																																														
PRFSW032																																														
PRFSW033																																														
PRFSW034																																														
PRFSW035																																														
PRFSW036																																														
PRFSW037																																														
PRFSW038																																														
PRFSW039																																														
PRFSS001																																														
PRFSS002																																														
PRFSS003																																														

Tabla 108. Matriz trazabilidad SR/PR

CAPÍTULO 7.

CONCLUSIONES Y

LÍNEAS FUTURAS

7.1 CONCLUSIONES

En el apartado [1.2 Objetivos generales](#) se realizaron una serie de especificaciones que debía de cumplir el sistema. Una vez finalizadas todas las etapas del proyecto se puede determinar que los objetivos generales del proyecto para la creación de un sistema de gestión de usuarios SUM se han cumplido de manera satisfactoria.

Se han conseguido los siguientes objetivos generales:

- Gestión de usuarios a nivel departamental permitiendo el mantenimiento de usuarios.
- Gestión de permisos de usuarios a nivel departamental, permitiendo el mantenimiento de los permisos de acceso de los usuarios al sistema GEXTEL.
- Generación de dos módulos **SUM Web** que permite ejecutar acciones *on line* y **SUM Server** que permiten ejecutar acciones diferidas evitando así la interferencia con procesos en curso de la tramitación de expedientes relativas a los usuarios objeto de modificación.
- Permiso de acceso o manipulación del sistema SUM a usuarios con privilegios de administración para la gestión de usuarios a nivel departamental.
- Independización de la labor de mantenimiento de usuarios y configuración de permisos de usuario del área de desarrollo informático.
- Disminución en gran medida del tiempo empleado en la labor de mantenimiento de usuarios en el área de desarrollo informático.
- Reducción de costes empleados en la labor de mantenimiento de usuarios en el área de desarrollo informático.

Se han ampliado los objetivos generales generando un sistema robusto, fácilmente mantenable y escalable.

Se determina que se han seguido las directrices establecidas en el estándar de la ESA a lo largo del desarrollo del proyecto y generado la documentación específica correspondiente.

Como inconvenientes encontrados durante el desarrollo del proyecto destaca la imposibilidad de crear relación directa entre las siguientes entidades del modelo de datos.

Profile - IGroup (grupos de un perfil)
IUser - Profile (perfil de un usuario)

Esto se debe a que el modelo de datos del sistema SUM ha tenido que adaptarse a un modelo de datos genérico del sistema GEXTEL con entidades no modificables.

Debido a esta limitación fue necesaria buscar una alternativa de solución para poder obtener la información relacional necesaria. En el caso *Profile - IGroup*, el contenido de los grupos de un perfil se extrae de un campo donde se guarda esa

información en formato de archivo XML. Para el caso *IUser – Profile*, se comparan los grupos asignados a un usuario con los de los perfiles del departamento.

La opción más sencilla habría sido implementar la relación directa entre las entidades expuestas. No ha sido así para mostrar la realidad del sistema implantado e incorporar el uso de archivos en formato XML al presente proyecto.

Por último indicar que el sistema SUM es un sistema real, actualmente en uso, implantado en la intranet de una empresa de carácter público.

A nivel personal, tener la experiencia de realizar un proyecto completo, fase a fase, con la perspectiva de los diferentes roles implicados, no sólo ha sido la culminación real de mi carrera universitaria sino que ha llegado más lejos. Me ha permitido aplicar, ampliar, afianzar conocimientos y aumentar la visión global sobre un proyecto proporcionándome un mayor entendimiento y seguridad en el desempeño de mis funciones.

Esta experiencia, aunque costosa, la considero positiva y muy recomendable.

7.2 LÍNEAS FUTURAS

El sistema SUM ofrece la posibilidad de abrir diferentes líneas de trabajo futuras en base a la explotación de los datos registrados, el entorno tecnológico, la inclusión de nueva funcionalidad y la adaptación a otros sistemas de información.

Explotación de los datos

La información registrada mediante el sistema permite la generación de estadísticas e informes de seguimiento relativos a:

- Usuarios eliminados.
- Nuevos usuarios.
- Peticiones generadas según su tipo.
- Total peticiones generadas en un plazo determinado.
- Peticiones finalizadas erróneamente en un plazo determinado.
- Peticiones finalizadas correctamente en un plazo determinado.

Nueva funcionalidad

El sistema permite añadir fácilmente nueva funcionalidad.

- Permitir que cada uno de los departamentos pueda gestionar sus propios perfiles, de tal manera que pudieran generarse nuevos perfiles o realizar modificaciones en los ya existentes. Para esto sería necesario limitar los grupos a incorporar en un perfil para que no puedan disponer de permisos ilimitados.
- Permitir mover a un usuario de departamento previa confirmación del administrador del departamento destino, eliminando la necesidad de causar baja en el departamento origen y posterior alta en el departamento destino, de esta

manera, se evitaría perder el identificador de usuario y mantener el histórico de acciones realizadas por un usuario.

Entorno tecnológico

- Adaptación a dispositivos móviles.
- Adaptación a otros sistemas operativos.

Adaptación a otros sistemas de información

El diseño de este sistema permite su parametrización, en base al negocio, y servir como solución en diferentes tipos de actividades empresariales como gestión de personal, control de fichaje, control de acceso y actividades, control de funciones y responsabilidades, etc. Facilita la integración e implantación en otros sistemas.

Actualmente un sistema externo de gestión de tareas programadas (STM) inicia la ejecución del módulo SUM Server, encargado de ejecutar las acciones diferidas solicitadas. La implantación del sistema SUM en otra empresa requeriría de un nuevo desarrollo para simular el comportamiento del sistema STM o en su defecto sería posible realizar una sencilla adaptación al sistema SUM para permitir la activación automática del módulo de servidor.

ANEXOS.

ANEXO I. FORMATO XML DE DESCRIPCIÓN DE PERFILES

Un perfil permite especificar un conjunto de derechos o permisos (grupos). Añadir un perfil específico a un usuario es asignar permisos en bloque a un usuario. Estos derechos incluyen la pertenencia a grupos concretos.

La descripción XML de un perfil almacenada en cada carpeta de la tabla de perfiles sigue la siguiente definición.

Se trata de un texto XML con un nodo raíz llamado **PERFIL**, del cual cuelgan nodos de tipo **GRUPO**. Este tipo de nodo tiene dos atributos: **GROUPID**, que contiene el identificador de un grupo de usuarios, y **NOMBRE**, que contiene el nombre del grupo. El atributo **NOMBRE** se especifica sólo a efectos de legibilidad del XML, ya que lo que verdaderamente se procesa es el id. Este nodo se utiliza para indicar que el usuario pertenece a ese grupo.

El nodo raíz **PERFIL** tiene un atributo **NOMBRE** en el que se puede especificar un nombre para ese perfil.

Ejemplo de definición de un perfil de tramitador RC.

```
<?xml version="1.0" encoding="windows-1252"?>
<PERFIL NOMBRE="TRAMITADOR_RC">
  <GRUPO GROUPID="3" NOMBRE="RC" />
  <GRUPO GROUPID="1" NOMBRE="TRAMITADOR" />
</PERFIL>
```

ANEXO II. INSTALACIÓN Y CONFIGURACIÓN DE LAS HERRAMIENTAS DE DESARROLLO

A continuación se indica como debe de realizarse la instalación y/o configuración de las herramientas de desarrollo del sistema.

Instalación de *Internet Information Server (IIS)*

Para realizar la instalación de *Internet Information Server* es necesario realizar los siguientes pasos:

1. En el menú Inicio, haga clic Panel de Control.
2. Seleccione la opción Agregar o quitar programas.
3. Seleccione la opción Agregar componentes de Windows
4. En el asistente para componentes de Windows, en la lista de componentes seleccione la opción Servicios de *Internet Information Server (IIS)*

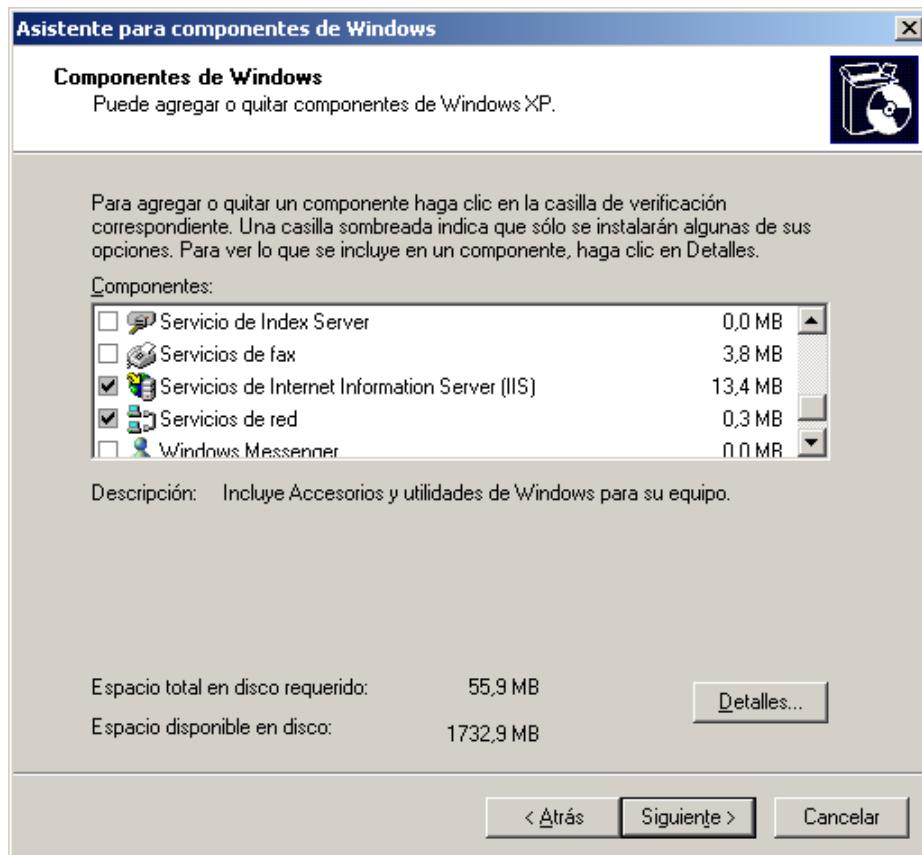


Figura 90. Instalación IIS: componentes Windows

5. Haga clic en siguiente.
6. Cuando el asistente complete la instalación, haga clic en Finalizar.

Instalación de Microsoft Visual Studio 2010

El procedimiento a seguir para la instalación de Microsoft Visual Studio 2010 es el siguiente:

1. Insertar CD-ROM de aplicación y esperar la ejecución del inicio automático.
2. Se iniciará el asistente para la instalación de Microsoft Visual Studio 2010. Pulsaremos en "Instalar Microsoft Visual Studio 2010".



Figura 91. Instalación MVS: inicio

3. En el primer paso de la instalación de Microsoft Visual Studio 2010 Ultimate, desmarcaremos "Sí, enviar a Microsoft Corporation información sobre la instalación" (si no queremos enviar esta información) y pulsaremos "Siguiente".

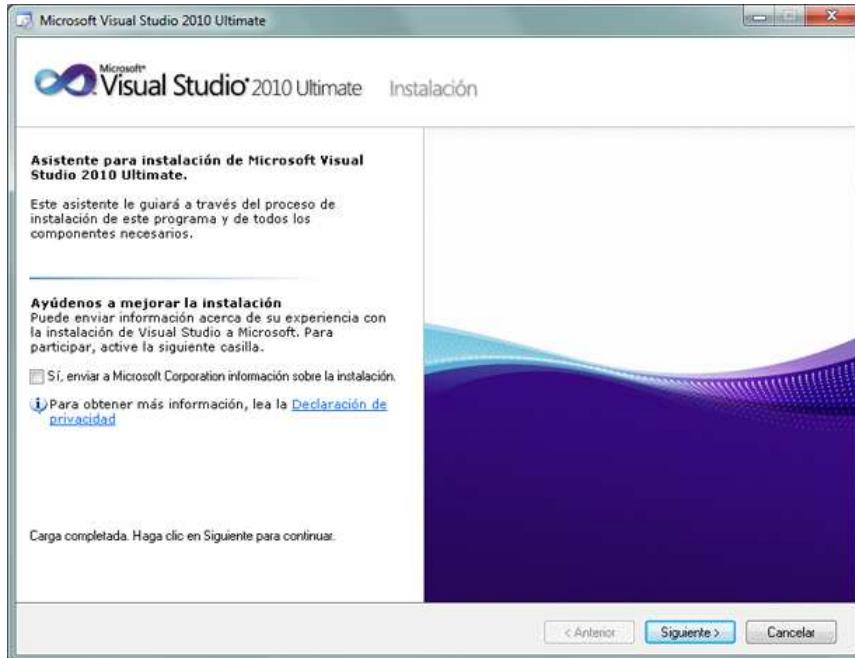


Figura 92. Instalación MVS: asistente

4. Leeremos los términos de licencia del software de Microsoft Visual Studio 2010 Ultimate (edición de prueba). Si estamos de acuerdo marcaremos "He leído los términos de la licencia y los acepto". Pulsaremos "Siguiente" para continuar.

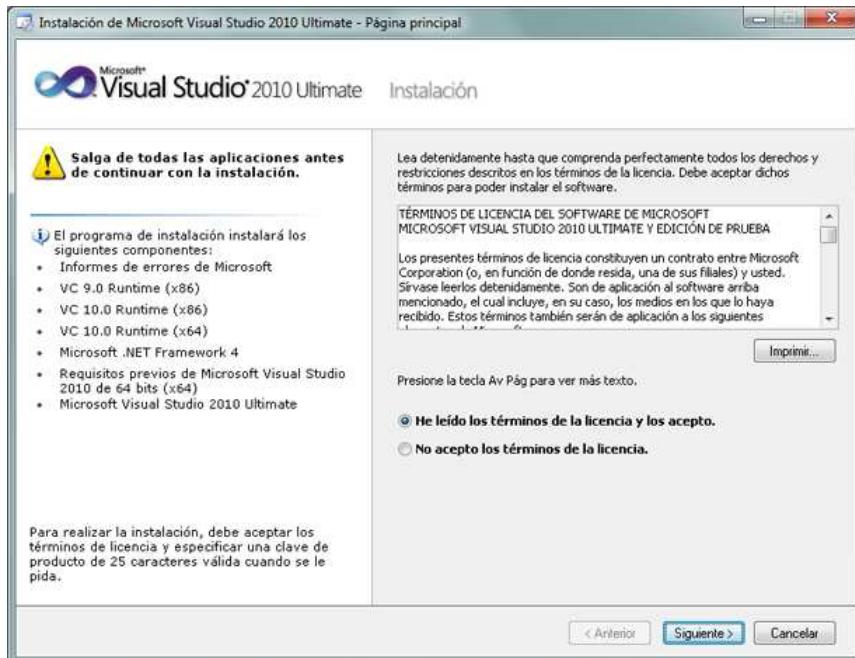


Figura 93. Instalación MVS: página principal

5. A continuación marcaremos "Personalizar" para seleccionar los lenguajes de programación y las herramientas que se quieran instalar. En "Ruta de instalación del producto" indicaremos la unidad y carpeta de destino de la instalación.

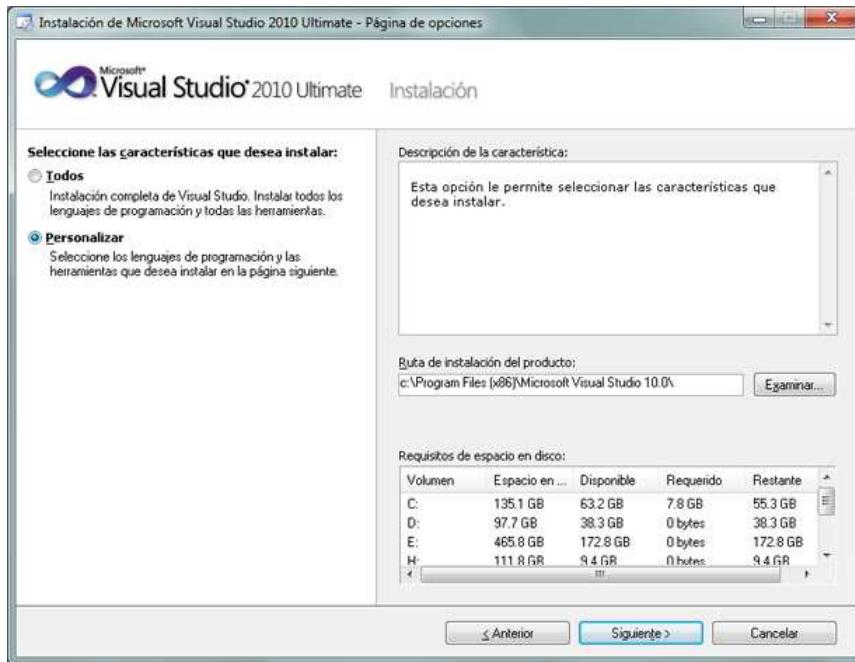


Figura 94. Instalación MVS: características

6. Seleccionaremos los lenguajes a instalar. Deberá seleccionar el lenguaje Visual Basic, ya que es el lenguaje que se utilizará en la implementación del proyecto.
7. Seleccionaremos también las características a instalar:
- Microsoft Office Developer Tools.
 - Dotfuscator Software Services.
 - Microsoft SQL Server 2008 Express.
 - Microsoft SharePoint Developer Tools.
8. Una vez seleccionadas las características a instalar pulsaremos en el botón "Instalar".

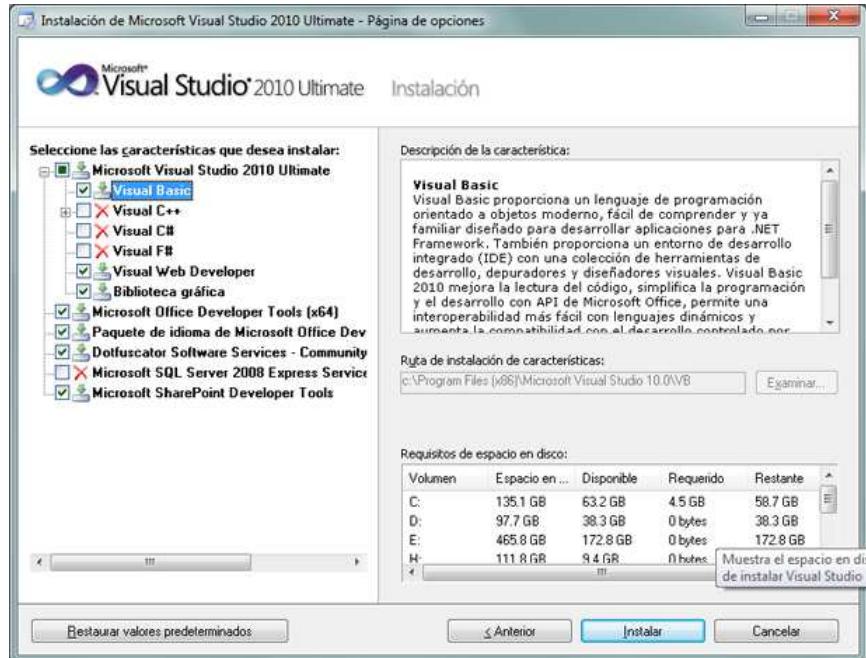


Figura 95. Instalación MVS: opciones

9. Se iniciará la instalación de Microsoft Visual Studio 2010 Ultimate.

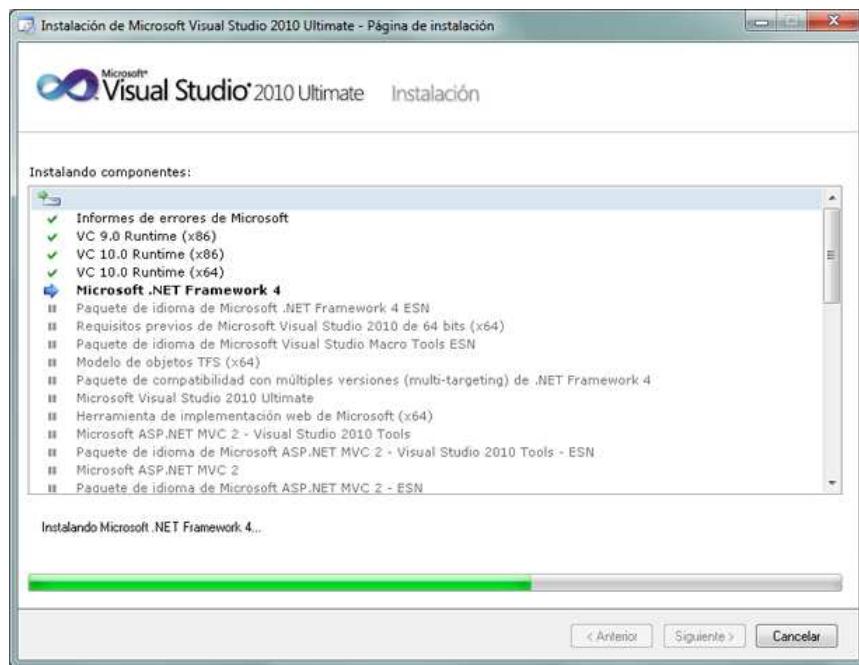


Figura 96. Instalación MVS: componentes

10. Tras la instalación, el asistente para instalar Visual Studio mostrará la siguiente ventana, con el texto "Correcto. Se instaló Visual Studio 2010 y finalizó el programa de instalación". Pulsaremos "Finalizar".

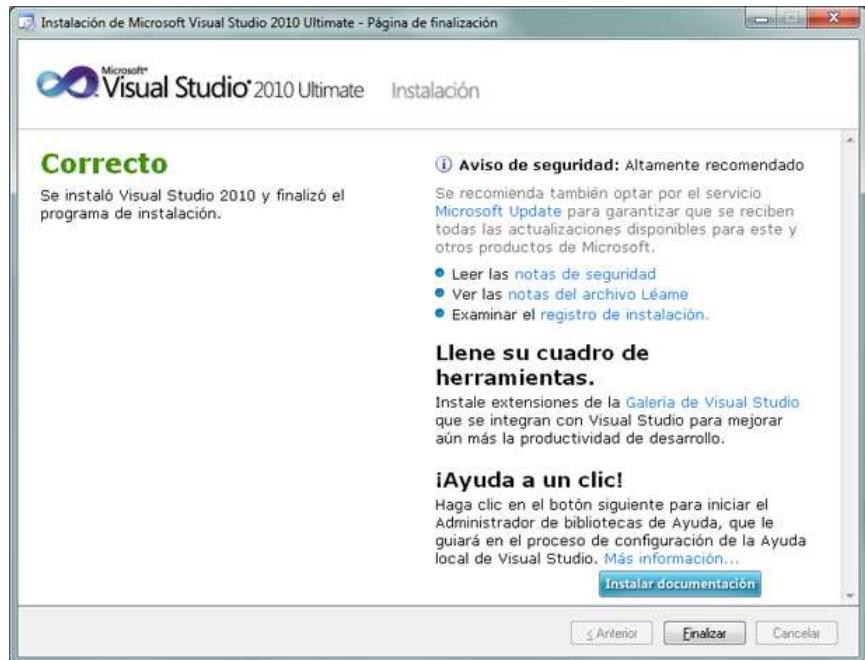


Figura 97. Instalación MVS: fin

11. Tras el proceso de instalación puede que pida reiniciar el equipo.



Figura 98. Instalación MVS: reiniciar

12. Con el texto: "Debe reiniciar el equipo para completar la instalación. El programa de instalación continuará automáticamente después de haber reiniciado el equipo".

13. Desde el botón "Iniciar" - "Todos los programas" podremos ver el grupo de programas "Microsoft Visual Studio 2010".

14. Tras iniciar por primera vez Microsoft Visual Studio 2010 Ultimate, seleccionaremos la configuración de entorno predeterminada, por ejemplo "Configuración de desarrollo de Visual Basic".

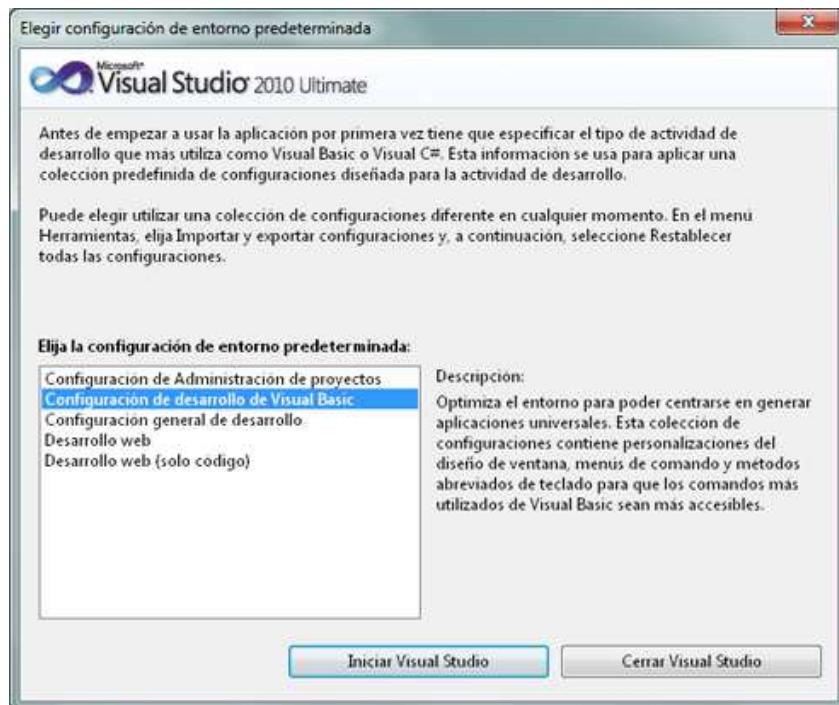


Figura 99. Instalación MVS: configuración

15. Y ya podremos disponer de este entorno de programación.

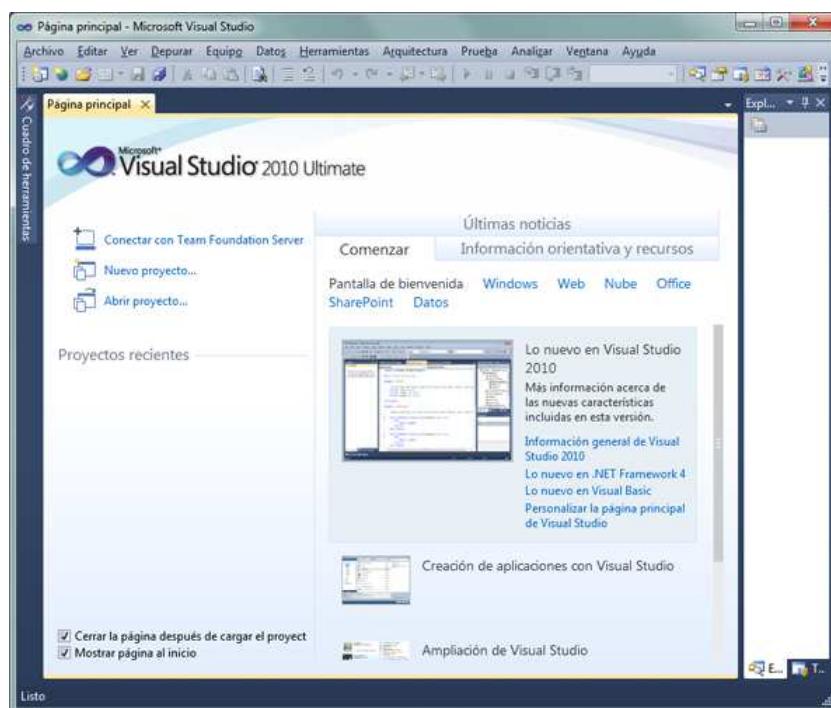


Figura 100. Instalación MVS: aplicación

Instalación de Microsoft SQL Server 2005

El procedimiento a seguir para la instalación de Microsoft SQL Server 2005 es el siguiente:

1. Insertar CD-ROM de aplicación y esperar la ejecución del inicio automático.
2. Se iniciará el asistente para la instalación de Microsoft SQL Server 2005. Aceptaremos la licencia y pulsaremos "Siguiente"

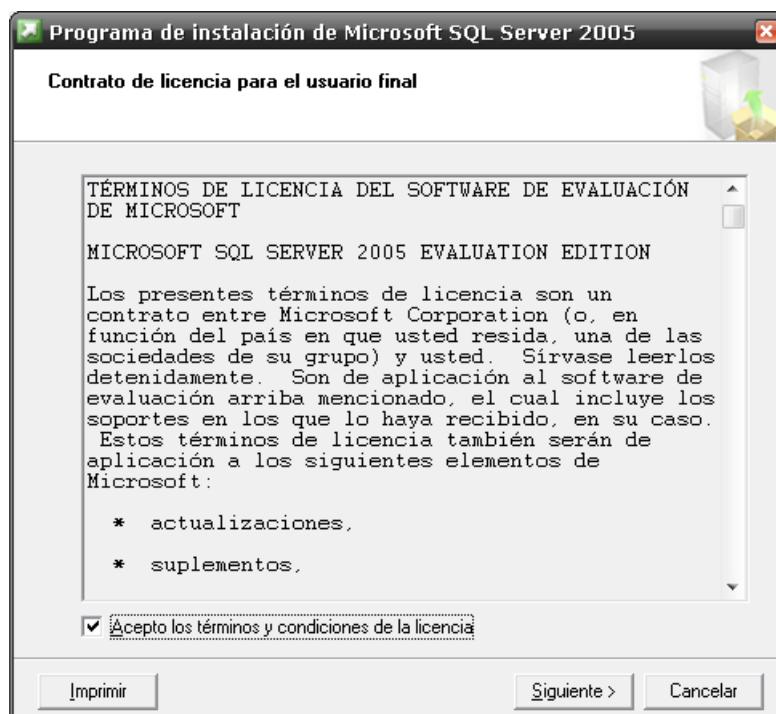


Figura 101. Instalación MSQLS: licencia

3. Pulsaremos “Instalar”.



Figura 102. Instalación MSQLS: requisitos previos

4. Se prepara la instalación.



Figura 103. Instalación MSQLS: requisitos instalando

5. Pulsaremos “Siguiente”.

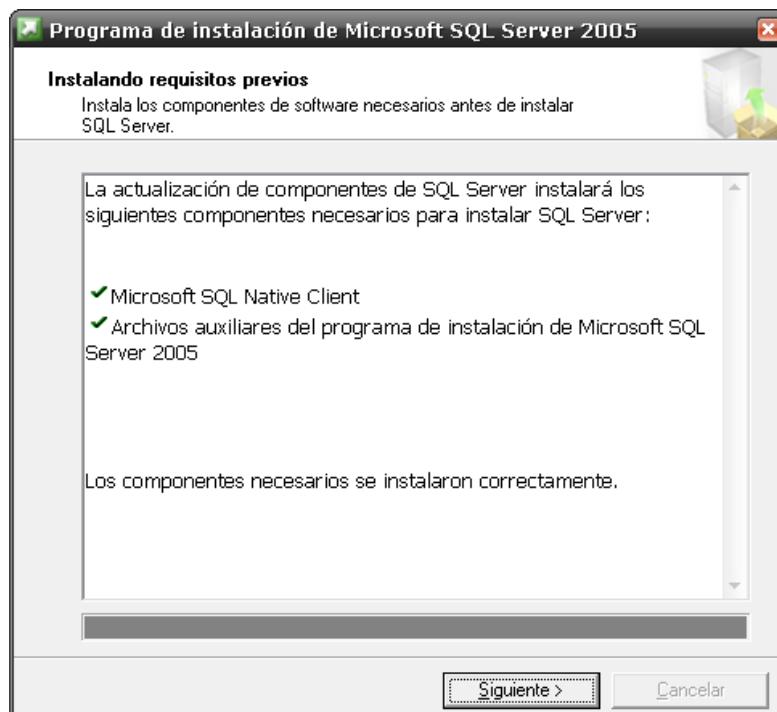


Figura 104. Instalación MSQLS: requisitos fin

6. Para continuar deben de cumplirse los requisitos que nos indica. Pulsaremos “Siguiente”.

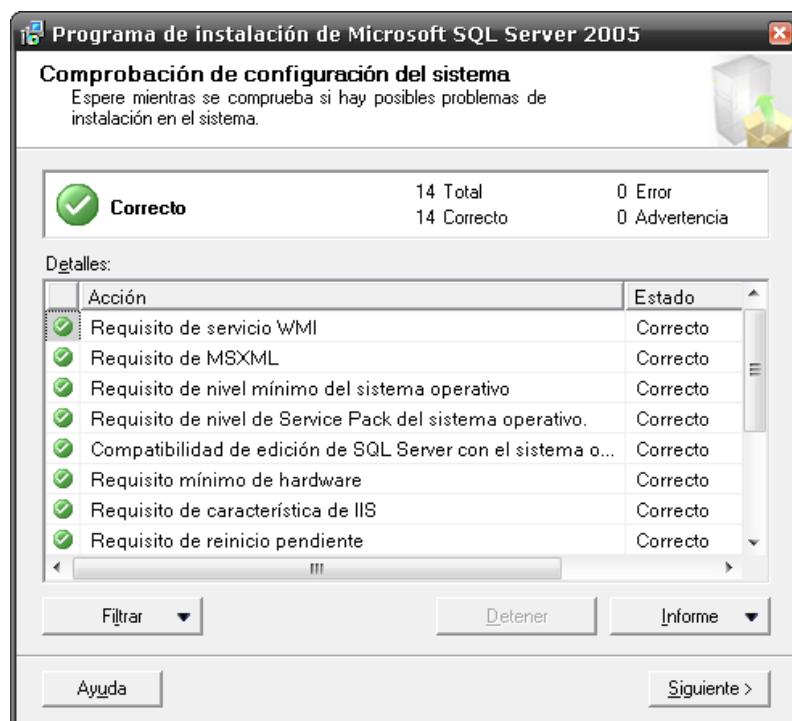


Figura 105. Instalación MSQLS: configuración sistema I



Figura 106. Instalación MSQLS: configuración sistema II

7. Introduciremos la información de registro. Pulsaremos “Siguiente”.



Figura 107. Instalación MSQLS: información de registro

8. Seleccionaremos “Avanzadas”.



Figura 108. Instalación MSQSL: componentes

9. Si tenemos poco espacio en el disco C: esta es la instalación mínima que se requiere. Cambiamos la ruta de instalación a D: (aunque se precisa de cierto espacio en C:). Mínimo: **867 en C:** y **411 en D:**



Figura 109. Instalación MSQSL: características I

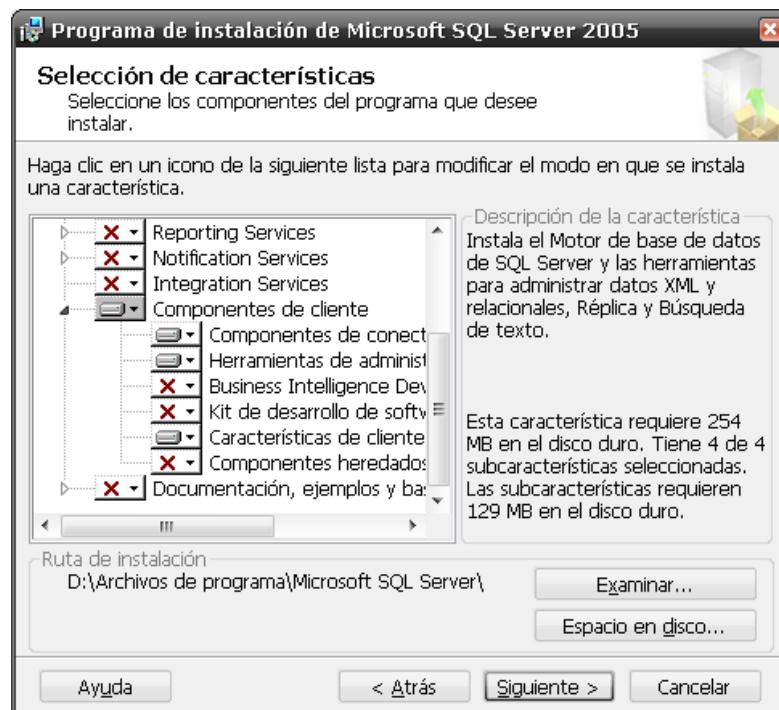


Figura 110. Instalación MSQLS: características II

10. Si no tenemos problemas de espacio en disco, esta sería la adecuada.

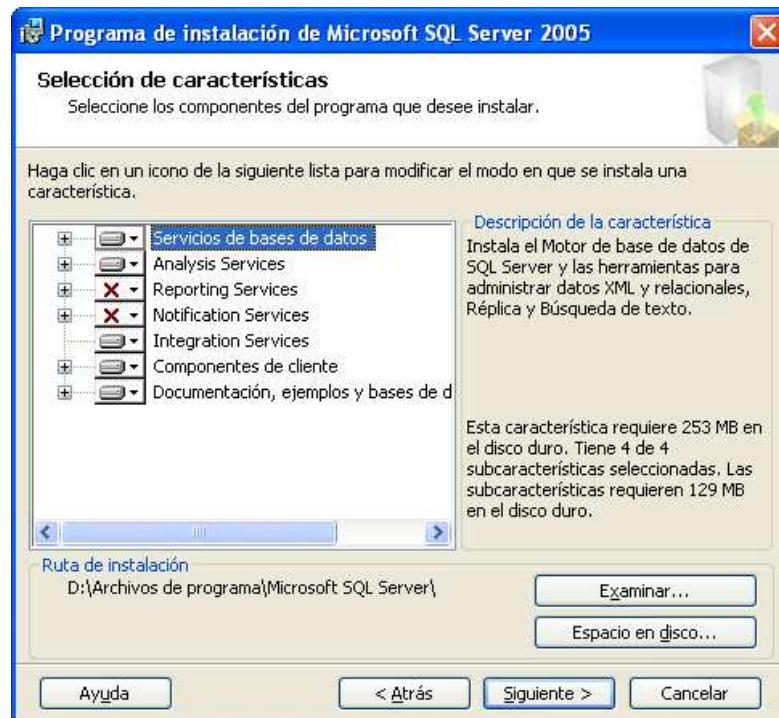


Figura 111. Instalación MSQLS: características III

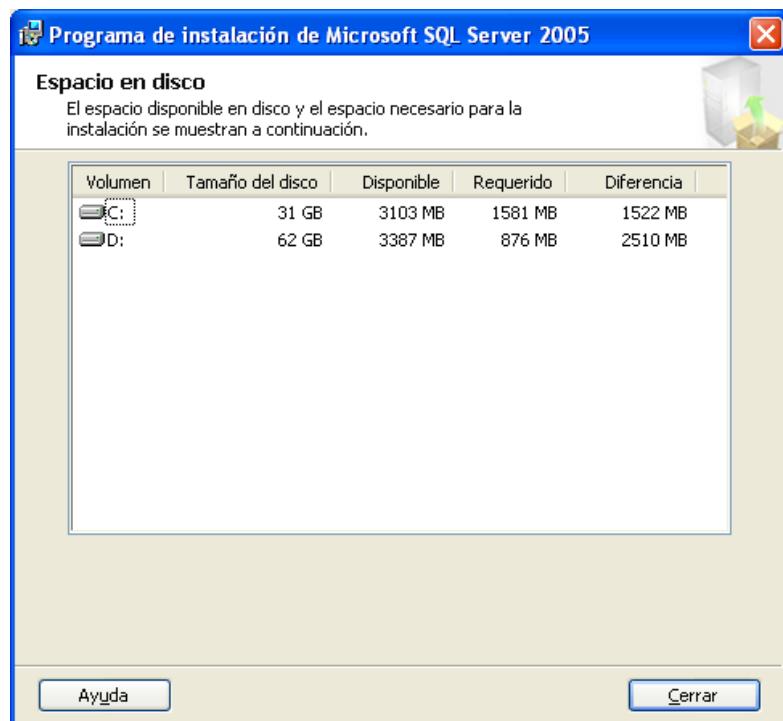


Figura 112. Instalación MSQLS: espacio en disco

11. Elegimos “Instancia predeterminada”.



Figura 113. Instalación MSQLS: instancia

12. Pulsaremos “Siguiente”.



Figura 114. Instalación MSQLS: cuenta de servicio I

13. Si hemos instalado más servicios podemos iniciarlos.



Figura 115. Instalación MSQLS: cuenta de servicio II

14. Elegiremos modo de autenticación de Windows.

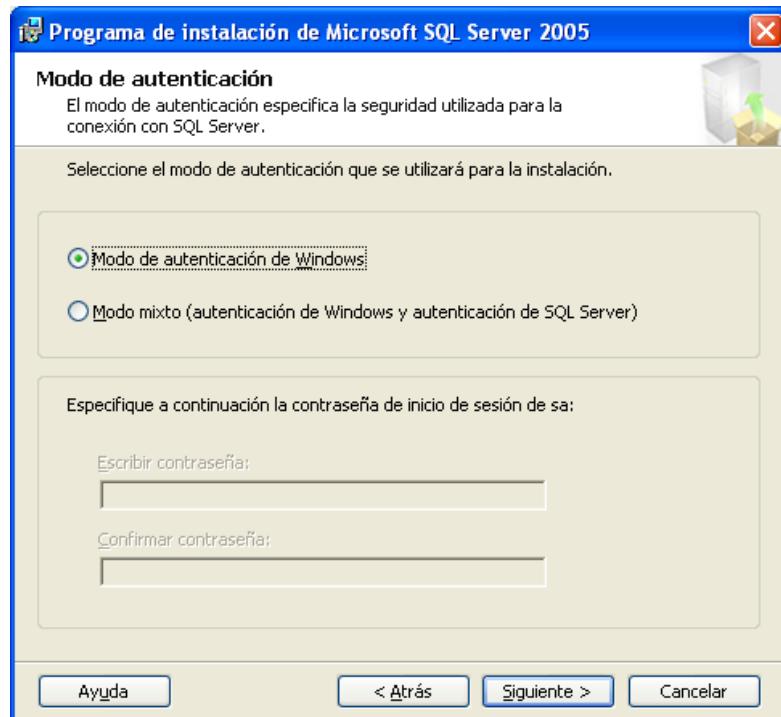


Figura 116. Instalación MSQLS: autenticación

15. Seleccionaremos la configuración y pulsaremos “Siguiente”.

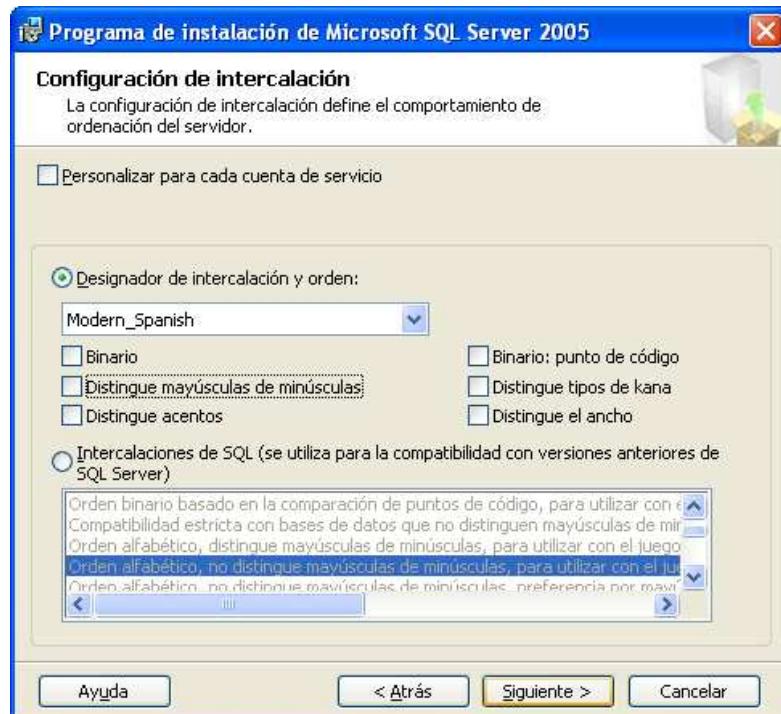


Figura 117. Instalación MSQLS: intercalación

16. Volveremos a pulsar “Siguiente”.

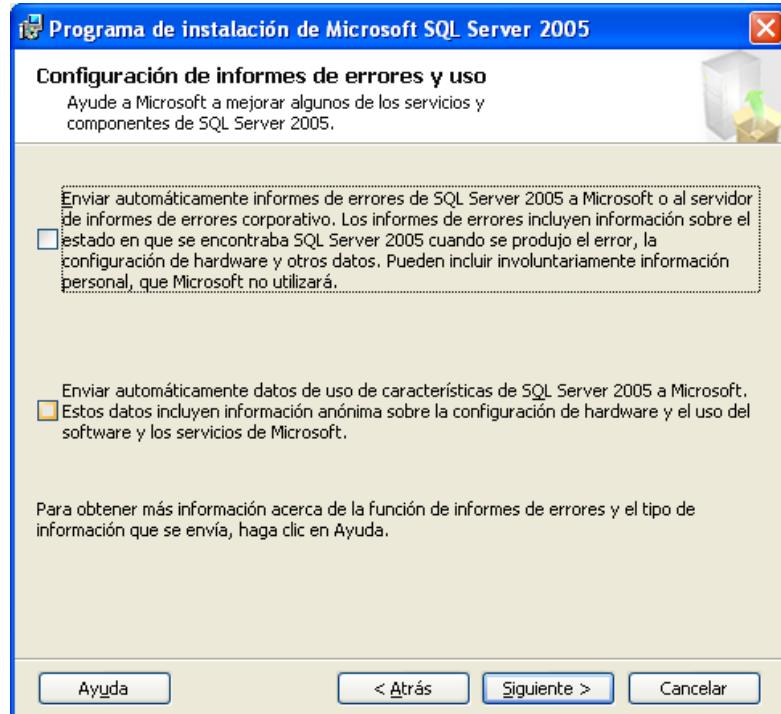


Figura 118. Instalación MSQLS: informe errores

17. La instalación ya está preparada. El nº de componentes variará según la selección realizada.

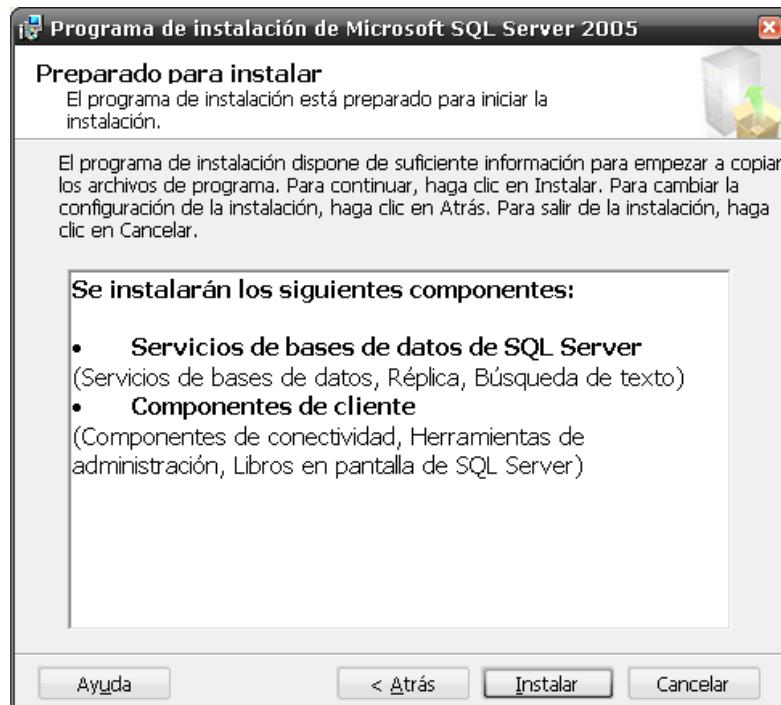


Figura 119. Instalación MSQLS: instalación I

18. Pulsaremos “Instalar” y comenzará la instalación.

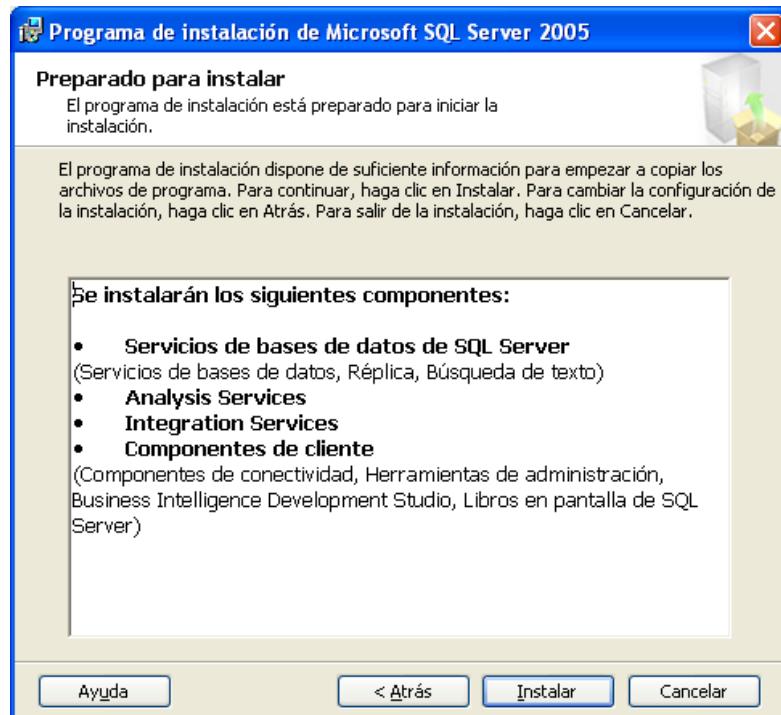


Figura 120. Instalación MSQLS: instalación II



Figura 121. Instalación MSQLS: progreso instalación I

19. Al finalizar deberemos pulsar “Siguiente”.



Figura 122. Instalación MSQLS: progreso instalación II

20. De nuevo pulsaremos “Siguiente”.



Figura 123. Instalación MSQLS: progreso instalación III

21. En este punto la instalación ha finalizado. Pulsaremos finalizar.

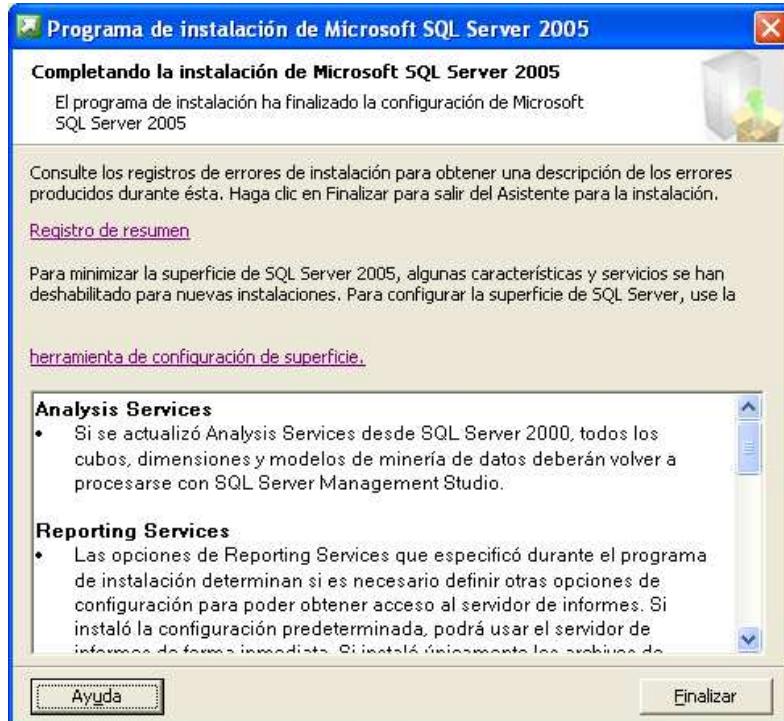


Figura 124. Instalación MSQLS: fin instalación

22. Para iniciar SQL Server Management Studio iremos al menú de Inicio > Todos los programas y tendremos un nuevo grupo llamado **Microsoft SQL Server 2005** y dentro un acceso directo llamado **SQL Server Management Studio** que nos sirve para iniciar esta herramienta de administración.

ANEXO III. CONFIGURACIÓN DEL SISTEMA SUM

A continuación se especifican el procedimiento a seguir para la configuración del sistema de gestión de usuarios SUM.

Configuración y publicación del módulo SUM Web

Para la publicación del sitio SUM Web es necesario disponer de un servidor Web de aplicaciones, es decir, tener instalados los servicios de *Internet Information Server* (IIS). Partiendo de esta premisa, será necesario realizar la siguiente configuración.

1. El módulo SUM Web deberá estar disponible en una ubicación física del servidor Web de aplicaciones. Normalmente una dirección destinada a la publicación de aplicaciones Web.
2. En el menú Inicio de Windows, haga clic en “Panel de Control”.
3. A continuación, haga clic en ”Herramientas administrativas”, “Servicios de *Internet Information Server (IIS)*”.
4. Se abrirá una nueva ventana para administrar los servicios de *Internet Information Server*. Pulse botón derecho sobre la opción Sitio Web predeterminado y seleccione Nuevo > Directorio virtual.



Figura 125. Configuración SUM Web: directorio virtual

5. A continuación se abrirá el asistente para configurar un directorio virtual. Pulse “Siguiente”.



Figura 126. Configuración SUM Web: asistente

6. Introduzca el alias correspondiente y pulse “Siguiente”-

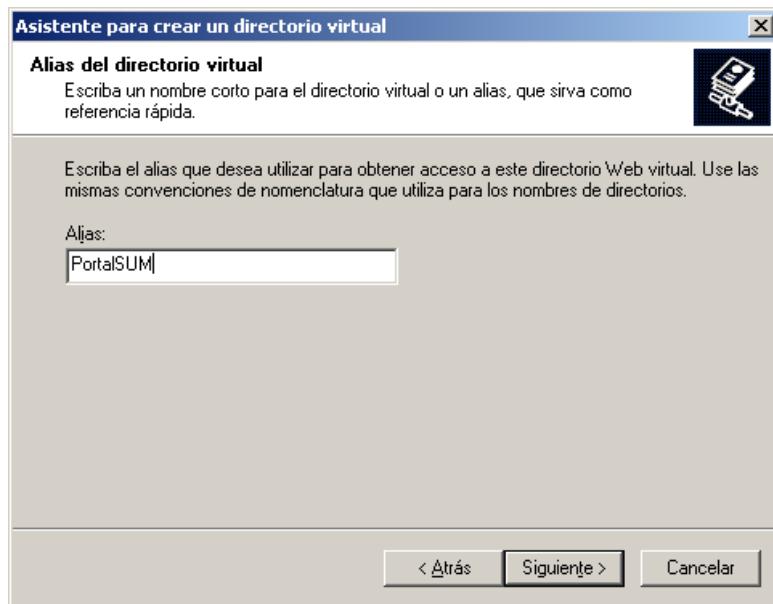


Figura 127. Configuración SUM Web: alias directorio

7. Introduzca la ubicación física correspondiente a la aplicación Web y pulse “Siguiente”.

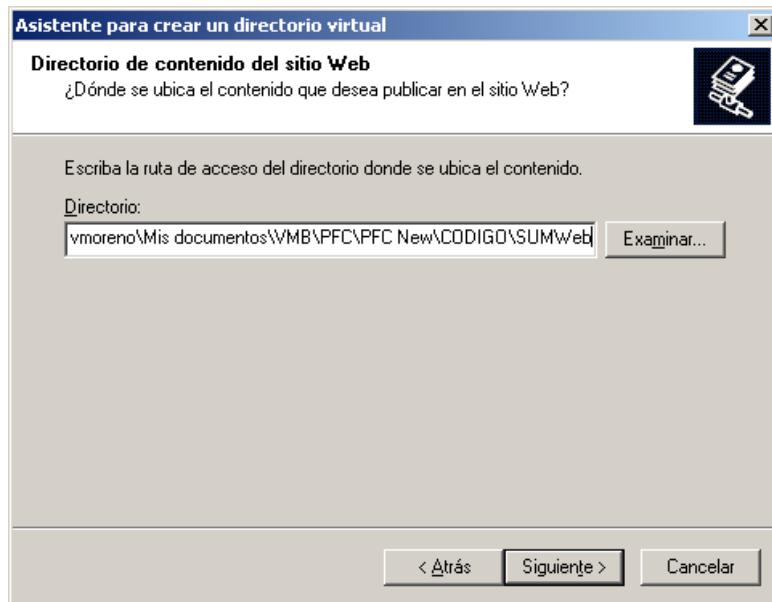


Figura 128. Configuración SUM Web: path directorio

8. Configure las opciones correspondientes y pulse “Siguiente”.

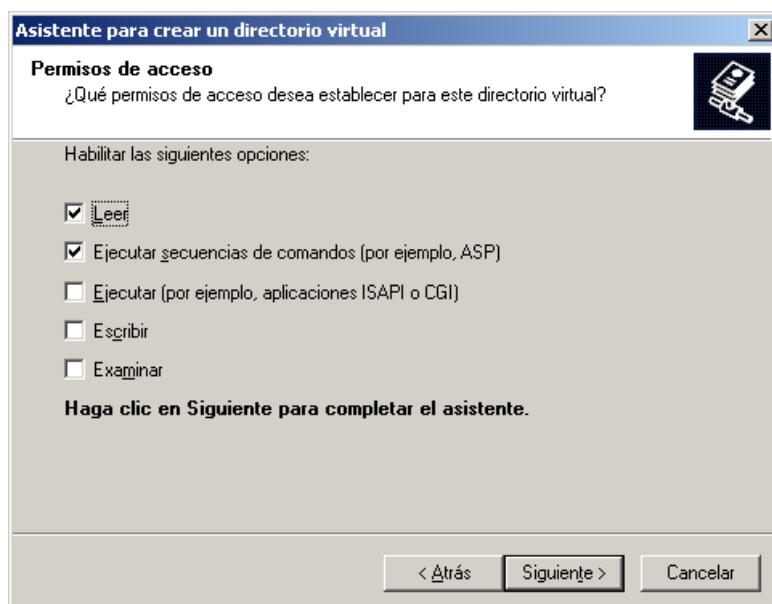


Figura 129. Configuración SUM Web: permisos acceso directorio

9. Para terminar la creación pulse “Finalizar”.

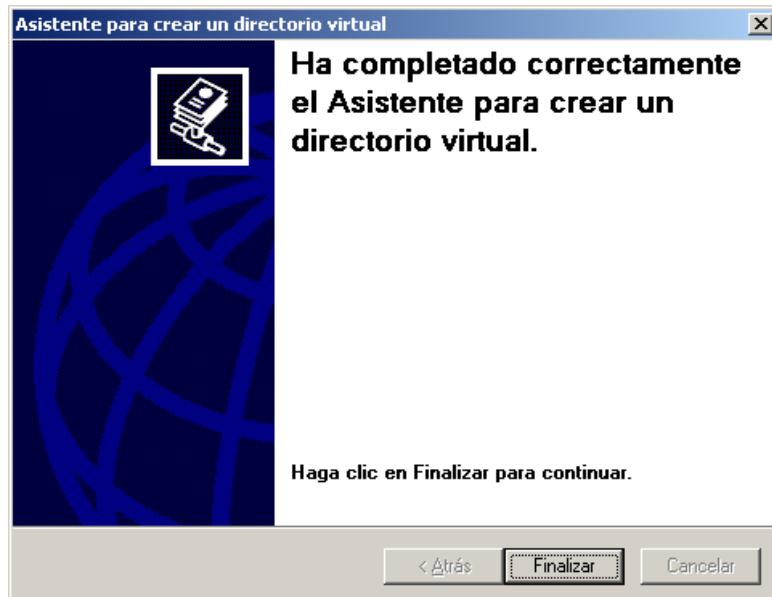


Figura 130. Configuración SUM Web: fin directorio virtual

10. El nuevo directorio aparecerá colgando del Sitio Web predeterminado.

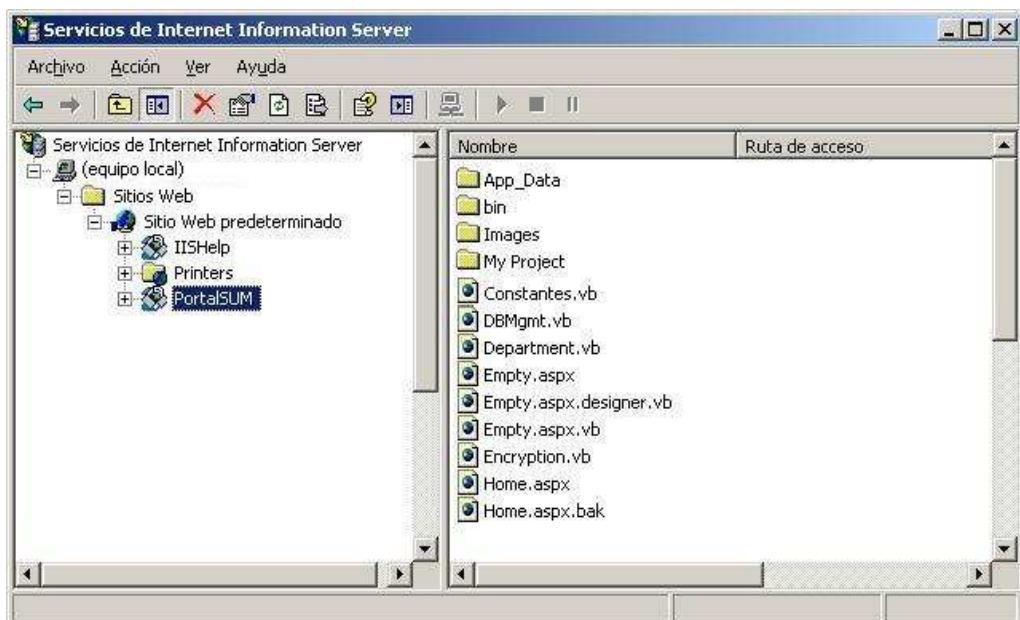


Figura 131. Configuración SUM Web: sitio Web predeterminado

11. Con el botón derecho sobre el nuevo directorio virtual si selecciona “Propiedades” puede configurar las opciones deseadas.

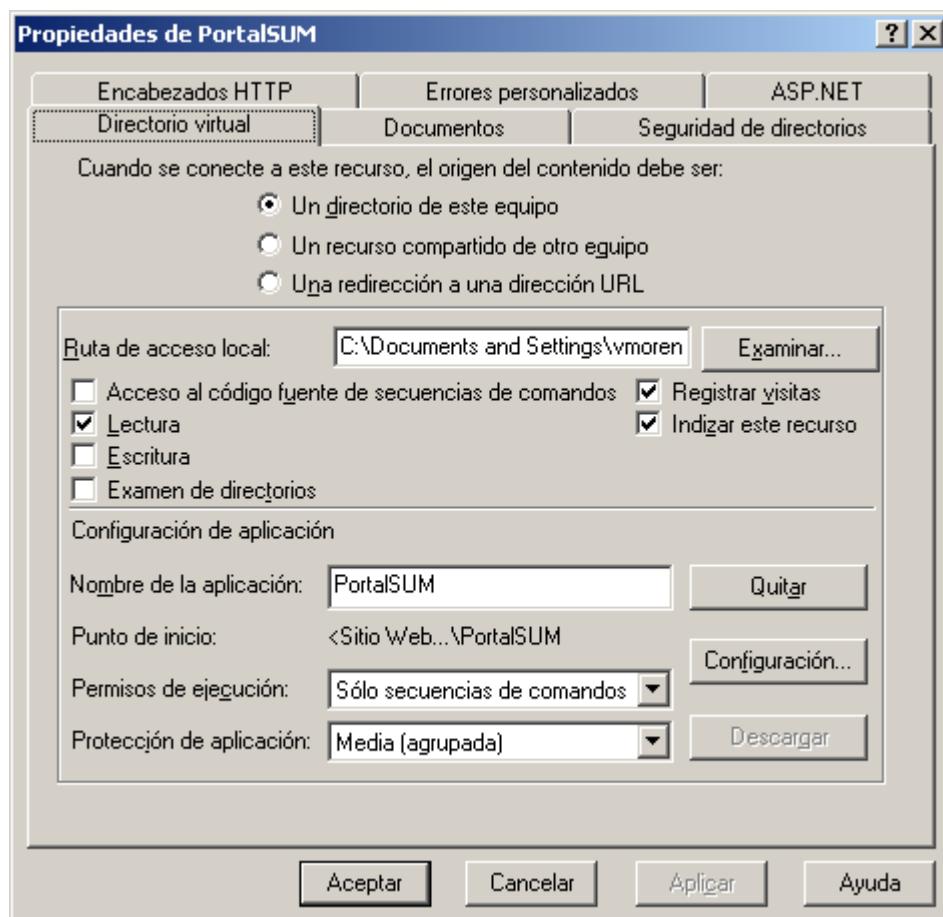


Figura 132. Configuración SUM Web: propiedades directorio

Acceso al módulo SUM Web

Para acceder al módulo Web del sistema de gestión de usuarios SUM deberá abrir un navegador de Internet e introducir en la barra de direcciones la siguiente URL (Uniform Resource Locator):

<http://localhost\PortalSUM\Login.aspx>

Configuración del módulo SUM Server

El módulo SUM Server no requiere de ninguna configuración específica, tan sólo deberá estar disponible en una ubicación física del servidor de aplicaciones para que pueda ser iniciada desde el sistema de gestión de tareas STM.

Normalmente la ubicación física en el servidor será una dirección destinada a las aplicaciones de tipo módulos de servidor.

Acceso al módulo SUM Server

El módulo SUM Server no será accesible por el usuario final. Como ya se explicó en el apartado [4.6.2](#) se trata de un módulo de servidor que se iniciará mediante el sistema externo de planificación de tareas STM. La configuración del sistema STM es ajeno a este proyecto por tanto no se va a especificar, tan sólo decir que el sistema STM deberá conocer la ubicación física del módulo SUM Server.

Si necesitamos invocar al módulo SUMServer sin necesidad de hacer uso del sistema externo de planificación de tareas STM bien porque urge efectuar la operación y no deseamos esperar el tiempo correspondiente a la acción diferida o bien por necesidad de realizar acciones sobre usuarios no existentes en el dominio (acción no permitida desde el módulo SUM Web), será posible inicializar la aplicación por invocación desde línea de comandos. Para ello se ejecutará el fichero **ExecuteSUMServer.vbs** que estará situado en una ubicación del servidor de aplicaciones. El fichero contiene las instrucciones para iniciar la aplicación pasando los parámetros correspondientes a la acción requerida, tan sólo deberá conocer la ruta de la ubicación física de la aplicación en el servidor de aplicaciones.

De manera excepcional y para cubrir las posibles necesidades de realizar acciones extra en el módulo SUM Server, se han habilitado las siguientes opciones que permiten no hacer uso del sistema externo de planificación de tareas STM. Si no existe tarea asociada a la acción requerida el identificador de tarea en la línea de invocación será 0.

- NX: creación de un usuario sin hacer uso del sistema externo de planificación de tareas.
- PX: asignación de perfil a usuario sin hacer uso del sistema externo de planificación de tareas.
- DX: eliminación de un usuario sin hacer uso del sistema externo de planificación de tareas.

A continuación se incluye el contenido del fichero para la invocación de la aplicación por línea de comandos con un ejemplo de cada una de las posibles acciones. Por cada ejecución del fichero sólo podrá realizarse una de las diferentes acciones. La acción a realizar deberá ser configurada con los datos correspondientes en el momento de la ejecución.

```
option explicit
Call Main()
MsgBox "Operación finalizada"
Sub Main
On Error Resume Next

Dim scriptobject
Set scriptobject = CreateObject("WScript.Shell")
```

```

'El fichero sólo podrá ejecutar una de las opciones disponibles a la vez: N, NX,
P, PX, D, DX
'Caso N: nuevo usuario
'Argumentos: args(0)=IdTarea      args(1)=Acción=N      args(2)=nameUserAdmin
args(3)=pwdUserAdmin args(4)=userName  args(5)=userPwd  args(6)=remarks
'scriptobject.run      """C:\Documents      and      Settings\vmoreno\Mis
documentos\VMB\PFC\PFC New\CODIGO\SUMServer\bin\Debug\SUMServer.exe"""
142 """"N"""" """"vmoreno"""" """"9003d1df22eb4d3820015070385194c8"""" """/lmeson"""
""""9003d1df22eb4d3820015070385194c8"""" """/Luisa Mesón Hernández"""

'Caso NX: nuevo usuario extra
'Argumentos: args(0)=IdTarea=0      args(1)=Acción=NX      args(2)=userName
args(3)=userPwd  args(4)=deptId  args(5)=remarks
'scriptobject.run      """C:\Documents      and      Settings\vmoreno\Mis
documentos\VMB\PFC\PFC New\CODIGO\SUMServer\bin\Debug\SUMServer.exe"""
0 """/NX"""" """/lmeson"""" """/9003d1df22eb4d3820015070385194c8"""" 1 """/Luisa Mesón
Hernández"""

'Caso P: asignar perfil a usuario
'Argumentos: args(0)=IdTarea      args(1)=Acción=P      args(2)=nameUserAdmin
args(3)=pwdUserAdmin      args(4)=userName      args(5)=idProfile
args(6)=deletePerms=True/False
'scriptobject.run      """C:\Documents      and      Settings\vmoreno\Mis
documentos\VMB\PFC\PFC New\CODIGO\SUMServer\bin\Debug\SUMServer.exe"""
143 """/P"""" """/vmoreno"""" """/9003d1df22eb4d3820015070385194c8"""" """/lmeson"""" 4
"""/True"""

'Caso PX: asignar perfil a usuario extra
'Argumentos: args(0)=IdTarea      args(1)=Acción=PX      args(2)=userName
args(3)=idProfile  args(4)=deletePerms=True/False
'scriptobject.run      """C:\Documents      and      Settings\vmoreno\Mis
documentos\VMB\PFC\PFC New\CODIGO\SUMServer\bin\Debug\SUMServer.exe"""
0 """/PX"""" """/lmeson"""" 2 """/True"""

'scriptobject.run      """C:\Documents      and      Settings\vmoreno\Mis
documentos\VMB\PFC\PFC New\CODIGO\SUMServer\bin\Debug\SUMServer.exe"""
0 """/PX"""" """/lmeson"""" 5 """/False"""

'Caso D: eliminar usuario
'Argumentos: args(0)=IdTarea      args(1)=Acción=D      args(2)=nameUserAdmin
args(3)=pwdUserAdmin args(4)=userName
'scriptobject.run      """C:\Documents      and      Settings\vmoreno\Mis
documentos\VMB\PFC\PFC New\CODIGO\SUMServer\bin\Debug\SUMServer.exe"""
144 """/D"""" """/vmoreno"""" """/9003d1df22eb4d3820015070385194c8"""" """/lmeson"""

'Caso DX: eliminar usuario extra
'Argumentos: args(0)=IdTarea  args(1)=Acción=DX  args(2)=nameUser
'scriptobject.run      """C:\Documents      and      Settings\vmoreno\Mis
documentos\VMB\PFC\PFC New\CODIGO\SUMServer\bin\Debug\SUMServer.exe"""
0 """/DX"""" """/lmeson"""

end sub

```

ANEXO IV. MANUAL DE USUARIO

A continuación se presenta el manual de usuario del sistema que realiza una descripción de las posibles acciones de la aplicación para la gestión de usuarios en el marco del proyecto GEXTEL.

Acceso al sistema de gestión de usuarios

Para acceder al sistema de gestión de usuarios es necesario acceder al menú principal del sistema corporativo GEXTEL habiéndose identificado previamente en el sistema.



Figura 133. Acceso GEXTEL

En caso de no introducir los datos solicitados se mostrará la siguiente ventana:

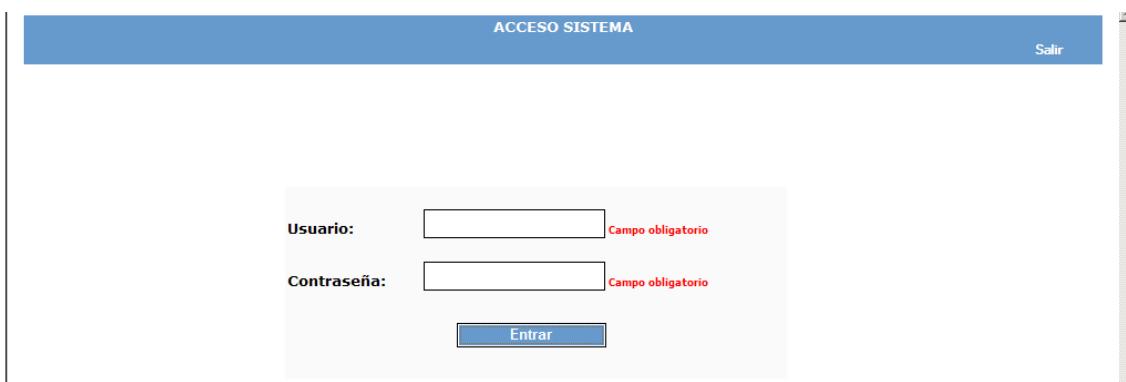


Figura 134. Acceso erróneo GEXTEL

Tras acceder al sistema se encuentra el menú principal de GEXTEL. El enlace que aparece en la siguiente imagen nos permite el acceso directo a la aplicación de gestión de usuarios.

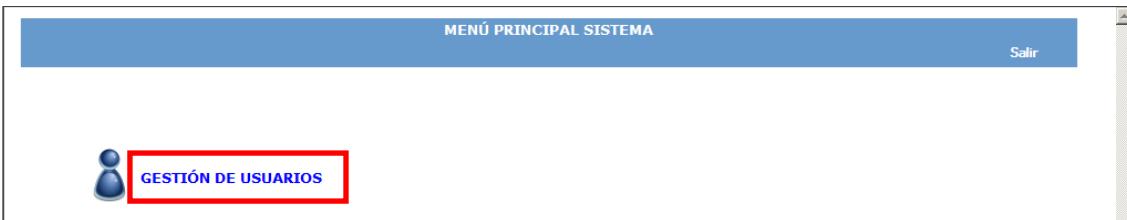


Figura 135. Acceso aplicación gestión de usuarios

En caso de no disponer del permiso necesario para el acceso a la aplicación se presenta la siguiente ventana:



Figura 136. Acceso erróneo aplicación gestión de usuarios

Gestión de usuarios

La aplicación permite a los usuarios con permiso de administración de usuarios gestionar los usuarios del sistema GEXTEL adscritos a su mismo departamento.

La gestión que puede realizar incluye:

Acciones diferidas

- Alta de usuarios nuevos
- Baja de usuarios
- Asignación de perfiles predefinidos a usuarios.

Acciones on line

- Bloqueo / desbloqueo de usuarios
- Cambiar contraseña de usuario
- Consultar permisos de portafirmas
- Ver permisos de usuario

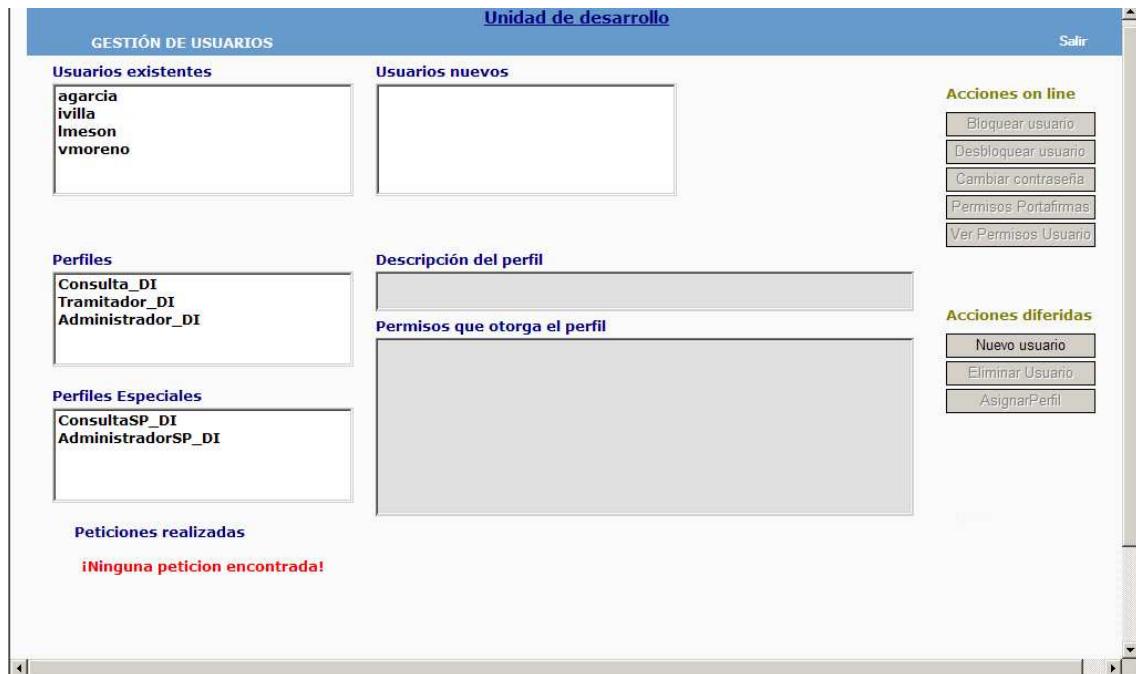


Figura 137. Ventana inicio gestión de usuarios

Al seleccionar un usuario de la lista se muestra información adicional bajo la lista de usuarios del departamento. Se muestra el nombre completo del usuario y en caso de que el usuario tenga un perfil no especial del departamento en curso asignado se mostrará el nombre de dicho perfil. Sólo se mostrará el nombre del perfil asignado al usuario en caso de que los permisos que tenga asignados el usuario coincidan en su totalidad con los de un perfil no especial del departamento, es decir, si el usuario posee más o menos permisos que los contenidos en un perfil no especial del departamento no se mostrará ningún perfil específico.

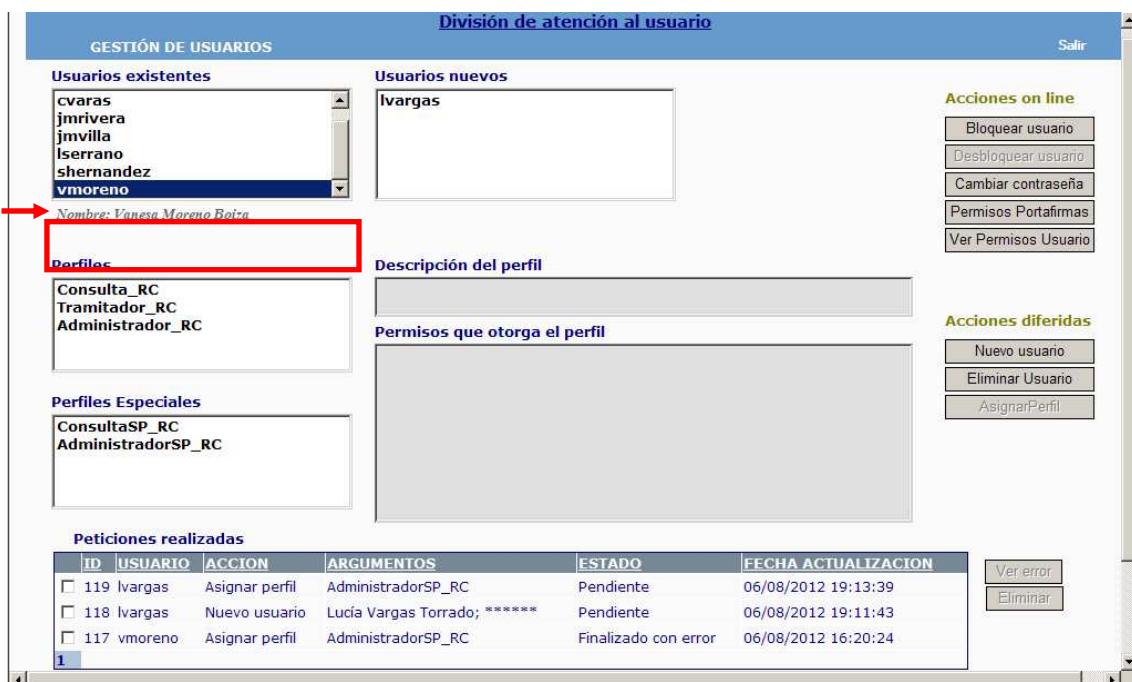


Figura 138. Información de usuario

1.1 Acciones diferidas

Las acciones diferidas son peticiones cuya ejecución se realiza a posteriori. Se registran en la lista de acciones con estado pendiente y una vez realizadas dicho estado cambia a estado *finalizado correctamente* o *finalizado con error*. Estas acciones implican una validación posterior, con lo que la acción no será efectiva hasta el día posterior a la petición.

Peticiones realizadas					
ID	USUARIO	ACCION	ARGUMENTOS	ESTADO	FECHA ACTUALIZACION
<input type="checkbox"/> 109	Ivargas	Nuevo usuario	Ivargas;*****	Pendiente	06/08/2012 12:44:17
1					

Figura 139. Lista de peticiones

La lista de peticiones incluye únicamente las peticiones de acciones diferidas.

Una vez han finalizado con estado *finalizado correctamente*, desaparecerán del listado, quedando únicamente las que tenga estado pendiente y/o finalizado con error.

Las peticiones tramitadas durante una sesión aparecerán en color rojo, para facilitar su seguimiento. Mientras no se cierre la sesión continuarán en color rojo.

De cada petición se muestran los siguientes datos: identificador de petición, usuario implicado, acción a realizar, datos informativos de la acción a realizar, estado y fecha de actualización.

Las peticiones aparecen ordenadas de manera descendente por fecha, pero se pueden ordenar por cualquier campo, únicamente es necesario pulsar sobre el título del campo, y se podrá ordenar por ese campo, de forma ascendente o descendente.

Se listarán todas las peticiones correspondientes al departamento del usuario conectado, de tal modo que no podrán ver las peticiones realizadas por otros departamentos.

Acciones sobre peticiones:

Con las peticiones se pueden realizar dos opciones: eliminar la petición, o en caso que la petición genere un error al tramitarse, nos permite visualizar un mensaje descriptivo del error.

Peticiones realizadas					
ID	USUARIO	ACCION	ARGUMENTOS	ESTADO	FECHA ACTUALIZACION
<input checked="" type="checkbox"/> 117	vmoreno	Asignar perfil	AdministradorSP_RC	Finalizado con error	06/08/2012 16:20:24
<input type="checkbox"/> 116	Ivargas	Asignar perfil	Administrador_RC	Pendiente	06/08/2012 16:19:19
<input type="checkbox"/> 115	Ivargas	Nuevo usuario	Lucía Vargas Torrado; *****	Pendiente	06/08/2012 16:18:48
1					

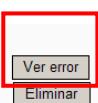


Figura 140. Acciones con lista de peticiones

Para realizar la acción de eliminar, debe seleccionar la petición o las peticiones que desea eliminar. La selección se realiza pinchando sobre . La acción de eliminar permite eliminar tantas peticiones como sean necesarias. Una vez seleccionadas pulse el botón **Eliminar** y aparecerá un mensaje de confirmación.



Figura 141. Mensaje confirmación eliminar petición

Una vez aceptado, desaparecerán las peticiones seleccionadas de la lista de peticiones.

En el caso que las peticiones estén en estado pendiente, si se eliminan no se tramitarían las peticiones.

Para *ver error*, debe seleccionar una única petición, pulsar el botón **Ver error** y aparecerá un mensaje informativo del error.



Figura 142. Mensaje informativo error tramitación

1.1.1 Alta usuarios nuevos

La acción alta de usuarios nuevos, permite crear un usuario que se encuentre incluido dentro del dominio de red.

Pulse el botón **Nuevo usuario** que se encuentra en la página principal de la aplicación.

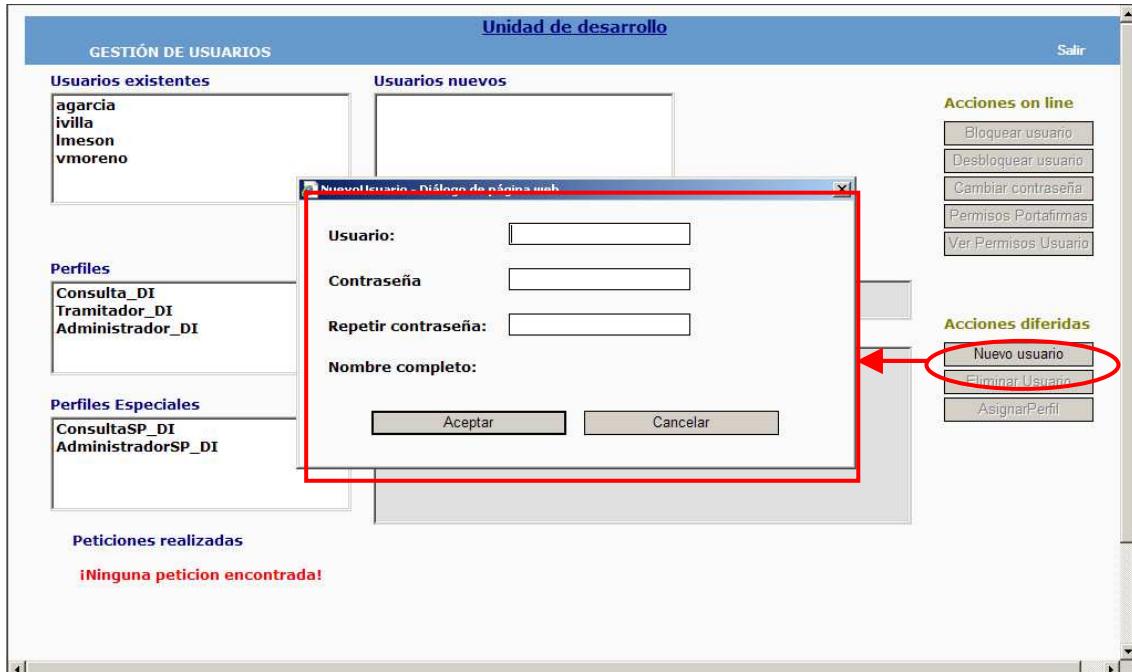


Figura 143. Inicio nuevo usuario

Una vez pulsado aparecerá una pantalla solicita usuario, contraseña y validación de contraseña.

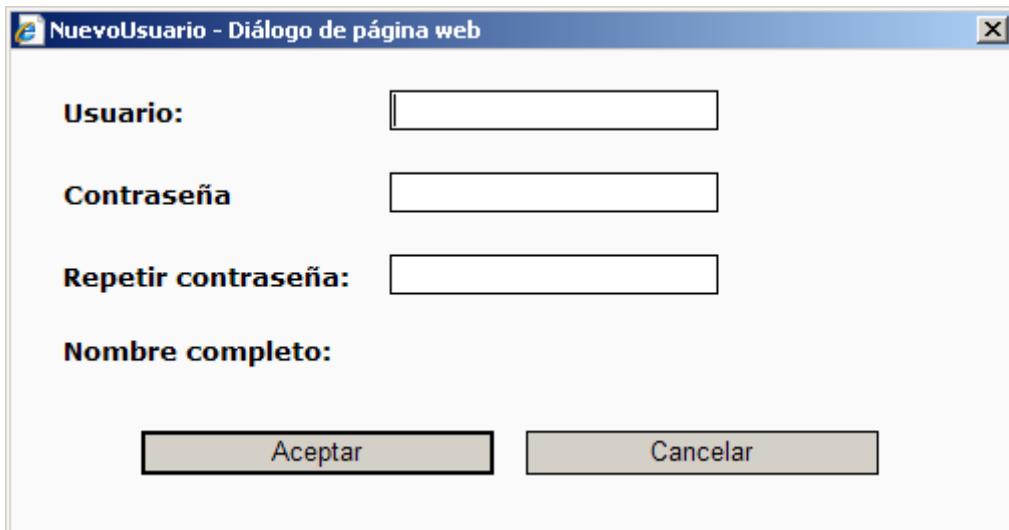


Figura 144. Nuevo usuario

En el campo usuario se debe incluir el nombre de usuario con la identificación de dominio de red de Windows. Al llenar este campo nos pueden surgir varias situaciones.

Usuario no incluido en dominio:

Si el usuario no esta incluido en el dominio o el nombre incluido no corresponde con ningún usuario del dominio, no nos permitirá dar de alta al usuario en la aplicación y aparecerá un mensaje informativo.

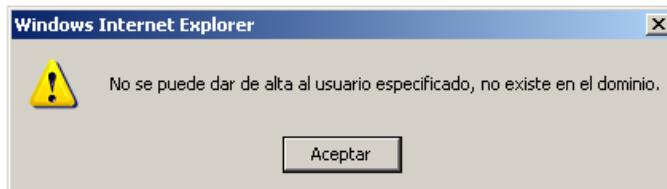


Figura 145. Usuario no existente en dominio

Usuario incluido en dominio:

Para que el usuario pueda ser dado de alta en la aplicación, es necesario que lo reconozca el dominio. Una vez introducido el usuario, al cambiar de campo, nos aparecerá automáticamente en el campo *nombre completo* el nombre del usuario.

A screenshot of a 'NuevoUsuario - Diálogo de página web' dialog box. It contains four input fields: 'Usuario' with the value 'lvargas', 'Contraseña' with three redacted dots, 'Repetir contraseña:' with three redacted dots, and 'Nombre completo:' with the value 'Lucía Vargas Torrado'. At the bottom are two buttons: 'Aceptar' and 'Cancelar'.

Figura 146. Usuario reconocido en dominio

Usuario existente en GEXTEL:

Si el usuario ya existe en el sistema GEXTEL, aparecerá un mensaje informativo indicando el departamento al que pertenece y nos impedirá dar de alta el usuario.



Figura 147. Usuario existe

Una vez reconocido el usuario, como usuario de dominio, se procede a incluir la contraseña de aplicación. Esta contraseña no es la contraseña del usuario de dominio, sino la contraseña que va a utilizar este usuario en las aplicaciones.

La contraseña debe tener como mínimo tres caracteres y puede contener cualquier carácter alfanumérico. Nos solicita la confirmación de la contraseña introducida, para corroborar que la contraseña introducida es la correcta.

Tanto el campo contraseña como repetir contraseña son obligatorios, por lo que para que el alta de cualquier usuario sea correcta debe de tener correctamente rellenados dichos campos.

Al pulsar aceptar, si los valores introducidos en los campos contraseña y repetir contraseña, no son correctos aparecerán mensajes informativos, con las peculiaridades que se deben de modificar.

Si la contraseña tiene una longitud inferior a 3 caracteres, aparecerá:



Figura 148. Longitud de contraseña inválida

Si la confirmación de la contraseña no es correcta o no hemos introducido los datos necesarios, aparecerá:



Figura 149. Aviso de ausencia de datos

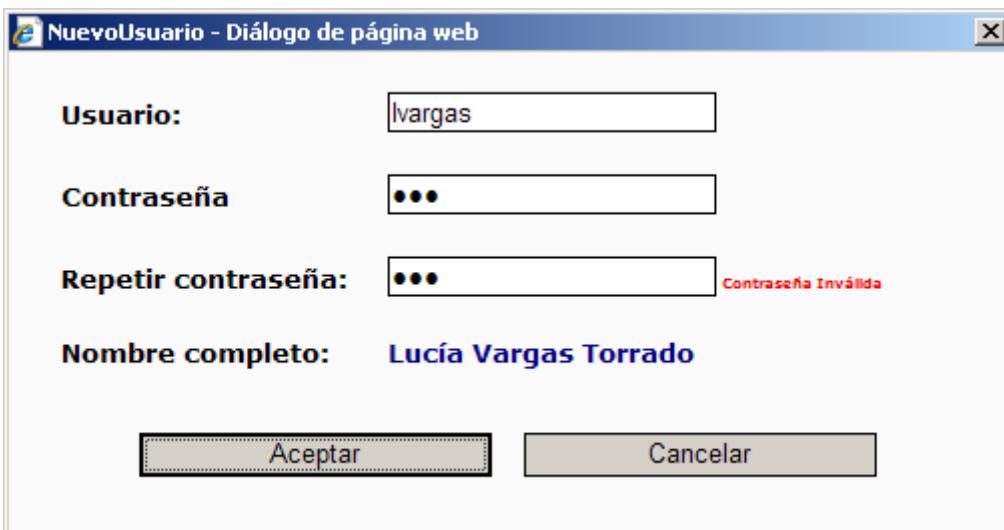


Figura 150. Confirmación contraseña errónea

Si al aceptar el alta del usuario es correcta aparecerá en la lista de peticiones, una tarea con estado pendiente con el usuario dado de alta.

También aparecerá en el cuadro de *usuarios nuevos*, el usuario para el que se ha tramitado el alta. Mientras que se encuentre en el cuadro de usuarios nuevos, el usuario todavía no estará dado de alta en GEXTEL. Por lo tanto el usuario no podrá acceder a las aplicaciones del portal hasta que no aparezca en el cuadro de *usuarios existentes*.



Figura 151. Alta de usuario correcta

Un usuario de nueva creación, por defecto no tiene asignado ningún perfil, por lo que será necesario que realice una asignación de perfil, para ello puede ver el apartado [1.1.3. Asignación de perfiles a usuarios](#).

Puede retroceder el alta del usuario siempre que se encuentre la petición en estado *pendiente* de dos modos:

Una opción es seleccionar el usuario en el cuadro *usuarios nuevos* y pulsar el botón *eliminar usuario* . Desaparecerá el usuario del cuadro *usuarios nuevos* y la petición de alta del usuario de las lista de peticiones.

Otra opción de retroceder el alta del usuario, es seleccionar la petición en la lista de peticiones y pulsar el botón *eliminar* . Desaparecerá la petición de la lista de peticiones y el usuario del cuadro de *nuevos usuarios*.

1.1.2 Baja de usuarios

Para realizar la baja de un usuario, es necesario seleccionar en la lista de usuarios existentes el usuario para el cual queremos tramitar la baja y pulsar el botón *eliminar usuario*.

ANEXOS

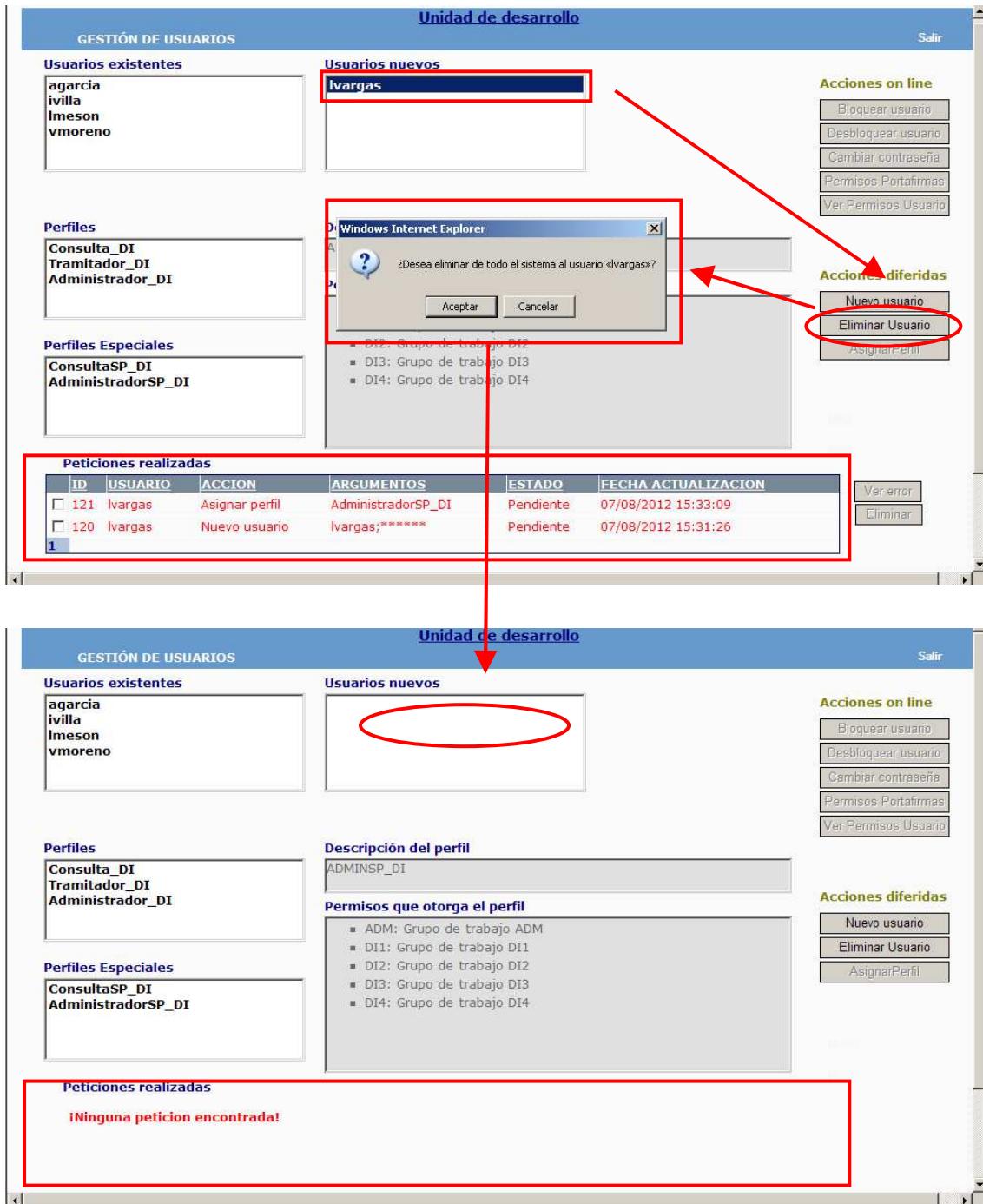


Figura 152. Baja usuario

Antes de realizar la petición de baja de usuario nos pide una confirmación de borrado.



Figura 153. Confirmación baja usuario

Una vez realizada la confirmación de baja del usuario en cuestión, aparecen varios indicadores:

Nos encontramos una cruz en el cuadro de usuarios existentes, justo al lado del usuario en cuestión, del modo (x). Nos indica que ese usuario se encuentra implicado en una petición de baja de usuario en estado pendiente.

Si el usuario a eliminar está incluido en la lista de usuarios nuevos se eliminará el usuario de la lista y las peticiones pendientes del mismo en caso de existir.

Si no se trata de un usuario nuevo aparece en la lista de peticiones una nueva petición con los datos de la baja y el estado de la petición.

The screenshot shows the 'GESTIÓN DE USUARIOS' (User Management) application interface. At the top, it says 'Unidad de desarrollo' and 'Salir'. On the left, there's a sidebar with 'GESTIÓN DE USUARIOS' and several sections: 'Usuarios existentes' (with entries 'adgarcia', 'ivilla' (highlighted with a red box), 'limeson', and 'vmoreno (x)'), 'Perfiles' (with 'Consulta_DI', 'Tramitador_DI', and 'Administrador_DI'), 'Perfiles Especiales' (with 'ConsultaSP_DI' and 'AdministradorSP_DI'), and 'Peticiones realizadas' (with a table showing a single row for '122 Ivargas Nuevo usuario'). On the right, there are two columns of buttons: 'Acciones on line' (Bloquear usuario, Desbloquear usuario, Cambiar contraseña, Permisos Portafirmas, Ver Permisos Usuario) and 'Acciones diferidas' (Nuevo usuario, Eliminar Usuario, AsignarPerfil). A red box highlights the 'Peticiones realizadas' table.

IM	USUARIO	ACTION	ARGUMENTOS	ESTADO	FECHA ACTUALIZACIÓN
<input type="checkbox"/> 123	vmoreno	Eliminar usuario		Pendiente	07/08/2012 15:36:17
<input type="checkbox"/> 122	Ivargas	Nuevo usuario	Lucía Vargas Torrado; *****	Pendiente	07/08/2012 15:36:03
1					

Figura 154. Indicadores tramitación baja usuario

Puede retroceder la baja del usuario siempre que se encuentre la petición en estado *pendiente*. Seleccione la petición de *eliminar usuario* y pulse *eliminar*

eliminara la petición de la lista de peticiones y desaparecerá el indicativo de usuario eliminado (x).

1.1.3 Asignación de perfiles a usuarios

La acción de asignar perfiles a usuarios, permite asignar un perfil a un usuario nuevo, reasignar perfiles o añadir permisos especiales a usuarios existentes.

Existen dos tipos de perfiles: perfiles que contienen permisos básicos del departamento y perfiles especiales que contienen permisos independientes del resto de perfiles.

Se pueden asignar cualquiera de los perfiles que aparecen en pantalla independientemente de su tipo. Cada departamento tiene acceso a unos perfiles predefinidos.

Para asignar perfiles a usuarios, será necesario seleccionar el usuario, lo puede seleccionar tanto si se encuentra en la lista de *usuarios nuevos* como si se encuentra en la lista de *usuarios existentes*. Una vez seleccionado el usuario en cuestión, debe de seleccionar el perfil que deseé asignar. Cuando se selecciona el perfil, en el cuadro Descripción del perfil aparecerá las características del perfil seleccionado.

- !** - Un usuario puede tener asignados varios perfiles especiales pero sólo un perfil básico.
- Al asignar un perfil básico a un usuario, se eliminan todos los permisos previos que tenga el usuario.
 - Al asignar un perfil especial a un usuario, se solicita confirmación para sobrescribir o no los permisos del usuario. Si se sobrescriben los permisos se eliminan todos los permisos que el usuario tenga, en caso contrario, se añaden los nuevos permisos manteniendo los que ya tuviera asignados el usuario.

Una vez seleccionado el usuario y el perfil, se habilitara el botón **Asignar Perfil**.

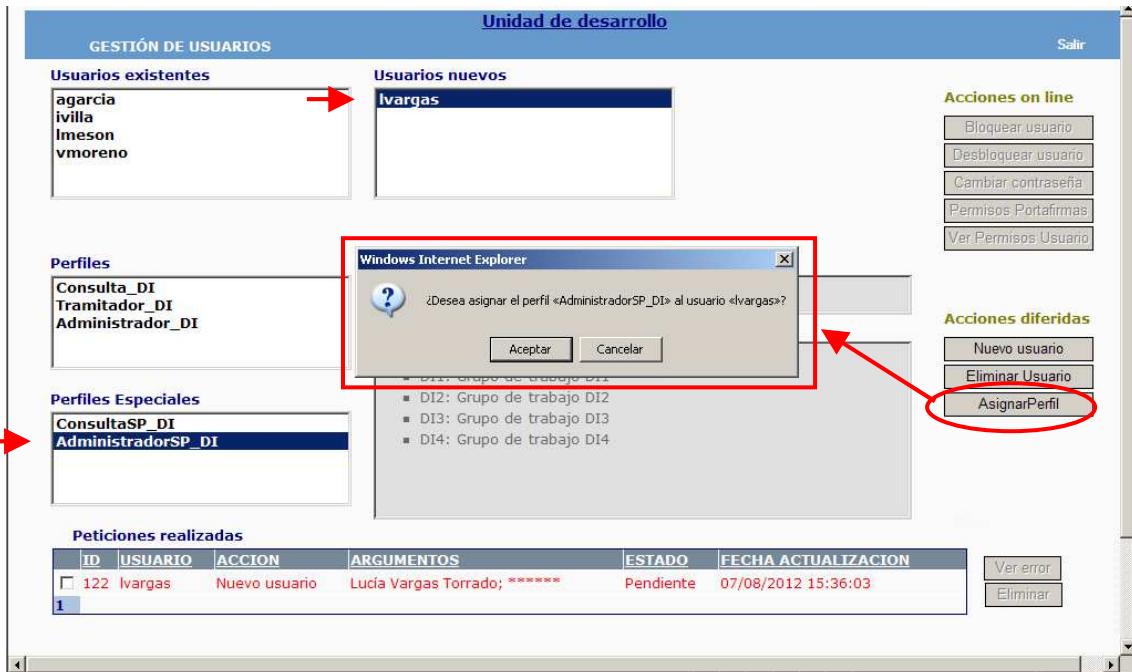


Figura 155. Asignar perfil

Una vez pulsado el botón **AsignarPerfil** aparecerá un mensaje de confirmación.



Figura 156. Confirmación asignar perfil especial

Si en lugar de un perfil especial se asigna un perfil básico la ventana de confirmación es la siguiente:

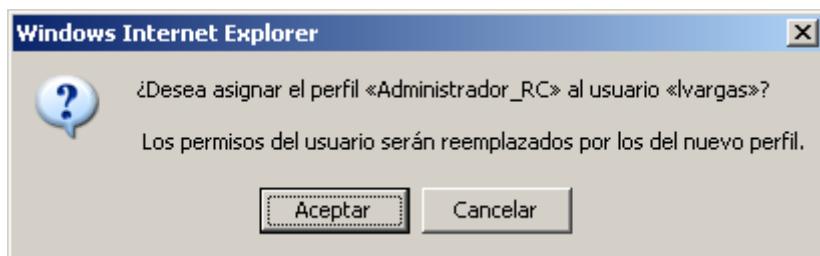


Figura 157. Confirmación asignar perfil básico

Si se trata de la asignación de un perfil especial se solicitará confirmación para eliminar todos los permisos que el usuario tenga asignados previamente.

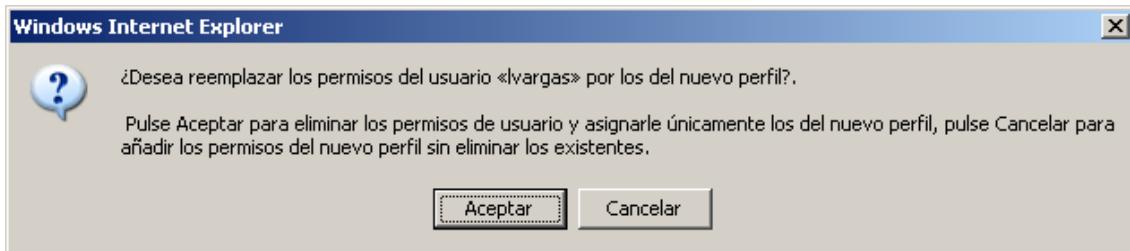


Figura 158. Confirmación eliminar permisos asignar perfil especial

Si la selección es correcta pulse *aceptar* y aparecerá en la lista de peticiones, una línea nueva con el estado pendiente.

Peticiones realizadas					
ID	USUARIO	ACCION	ARGUMENTOS	ESTADO	FECHA ACTUALIZACION
112	lvargas	Asignar perfil	AdministradorSP_DI	Pendiente	06/08/2012 12:58:04

Figura 159. Tabla de peticiones asignar perfil

Si se ha seleccionado un perfil especial para asignar al usuario, se solicitará confirmación para eliminar todos los permisos que el usuario tenga asignados previamente.

El perfil será asignado al usuario cuando la petición sea procesada.

1.2 Acciones *on line*

Acciones *on line* son acciones que se ejecutan en el momento que se tramitan, sin que sea necesario ninguna espera. Estas acciones incluyen bloqueo de usuario, desbloqueo de usuario y cambio de contraseña.

1.2.1 Bloqueo / desbloqueo de usuarios

Para bloquear un usuario necesita seleccionar el usuario en cuestión en el cuadro de *usuarios existentes* y pulsar el botón *bloquear usuario*.

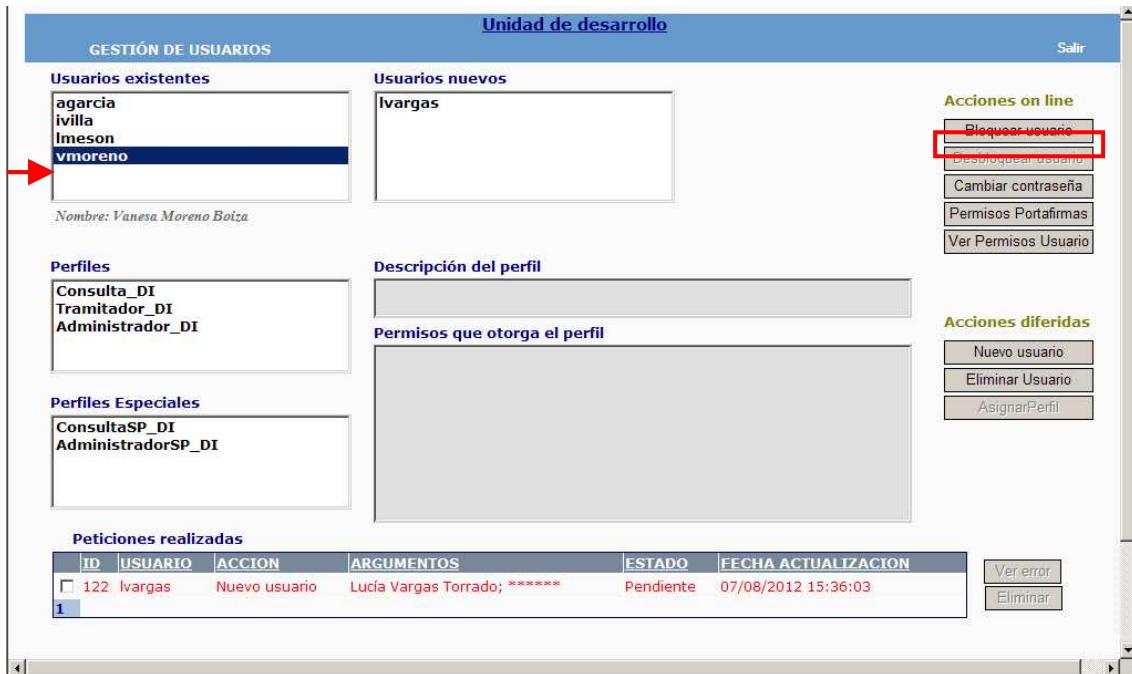


Figura 160. Bloqueo usuario

Una vez pulsado el botón *bloquear usuario* aparece una mensaje de confirmación.



Figura 161. Confirmación bloqueo usuario

Una vez pulsado *aceptar* el usuario quedara bloqueado no permitiendo el acceso a las aplicaciones.

Como indicador que el usuario en cuestión ha sido bloqueado, nos aparecerá un signo de exclamación entre paréntesis, justo al lado del usuario bloqueado en el cuadro de usuarios existentes (!)

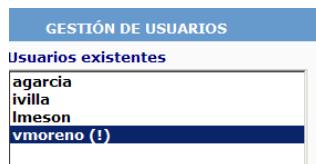


Figura 162. Indicador usuario bloqueado

Para desbloquear un usuario debemos seleccionar en el cuadro de usuarios existente un usuario que se encuentre bloqueado. Una vez seleccionado pulse el botón *desbloquear usuario*.

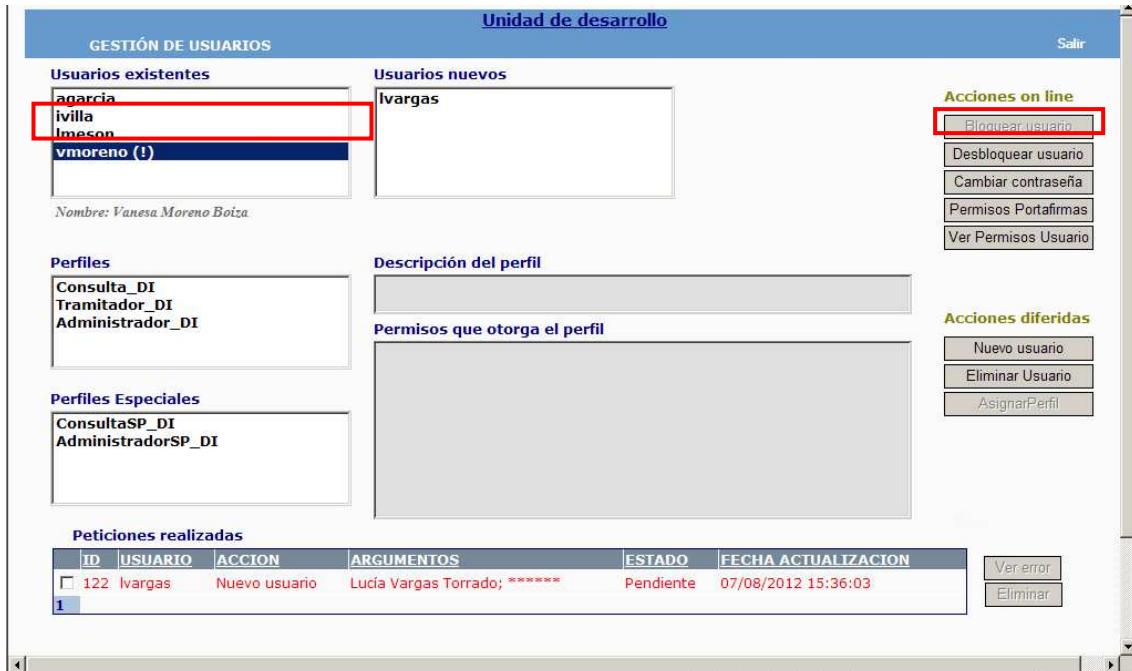


Figura 163. Desbloquear usuario

Antes de desbloquear un usuario, aparecerá un mensaje de confirmación.



Figura 164. Confirmación desbloqueo usuario

Una vez pulsado el botón aceptar, el usuario será desbloqueado, lo podrá comprobar viendo que en el cuadro *usuarios existentes* ya no aparece el indicador de usuario bloqueado (!).

1.2.2 Cambio de contraseñas

El cambio de contraseñas se realiza seleccionando el usuario que quiere cambiar la contraseña en el cuadro de usuario existentes y pulsar el botón *cambiar contraseña*.

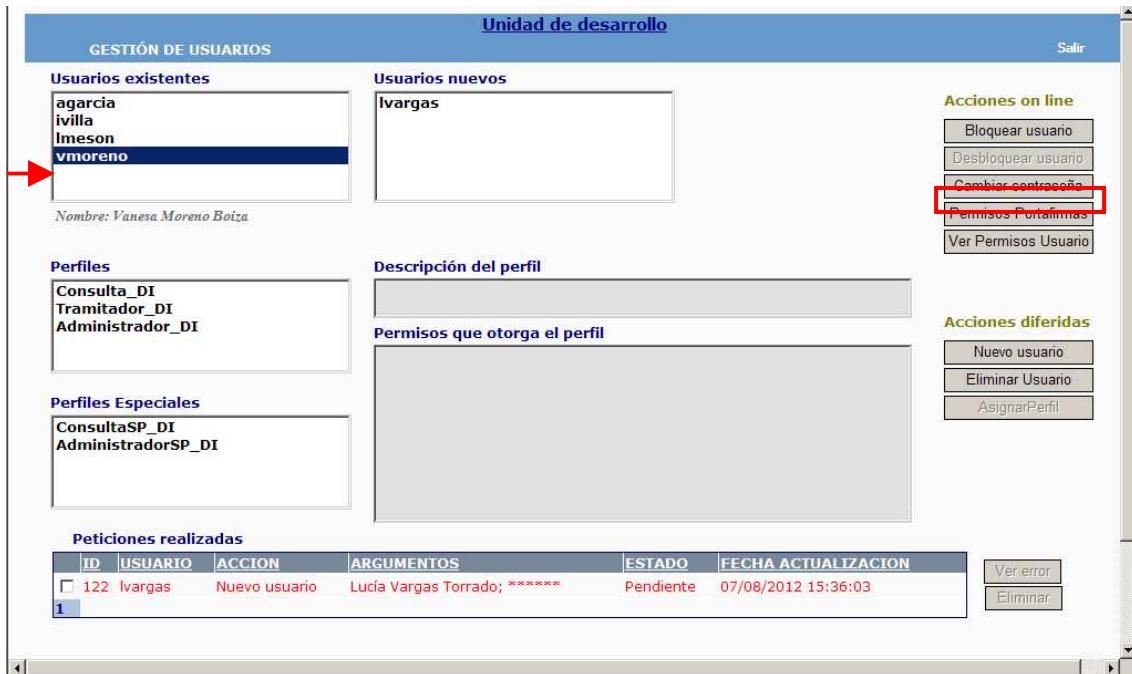


Figura 165. Cambio contraseña usuario

Una vez pulsado el botón cambiar contraseña aparece la pantalla con el usuario indicado y los campos nueva contraseña y un campo de confirmación de contraseña.

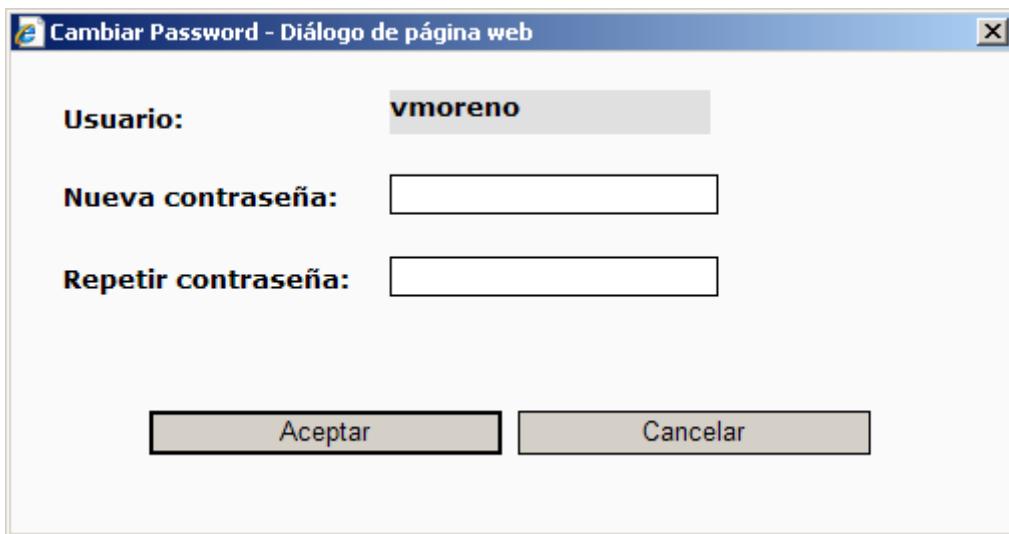


Figura 166. Formulario cambio contraseña

Para poder cambiar la contraseña debe de tener un mínimo de tres caracteres y puede contener cualquier valor alfanumérico. Debe de llenar el campo nueva contraseña y los valores introducidos en los campos nueva contraseña y repetir contraseña deben de ser iguales, en caso contrario serán informados de la incidencia sufrida.

The screenshot shows a password change dialog box. It has two input fields: 'Nueva contraseña:' and 'Repetir contraseña:'. The first field has a red border and the error message 'Campo obligatorio' to its right. The second field also has a red border and the error message 'Contraseña inválida' to its right. Below the fields are two buttons: 'Aceptar' and 'Cancelar'.

Si los datos introducidos son correctos aparecerá un mensaje informativo con el texto *contraseña modificada*.



Figura 168. Confirmación cambio contraseña

1.2.3 Permisos portafirmas

La asignación de permisos en el portafirmas se realiza seleccionando el usuario que quiere asignar permisos en el cuadro de usuario existentes y pulsar el botón *permisos portafirmas*.

Para ello será necesario que el departamento al que se pertenezca tenga permisos para el portafirmas.

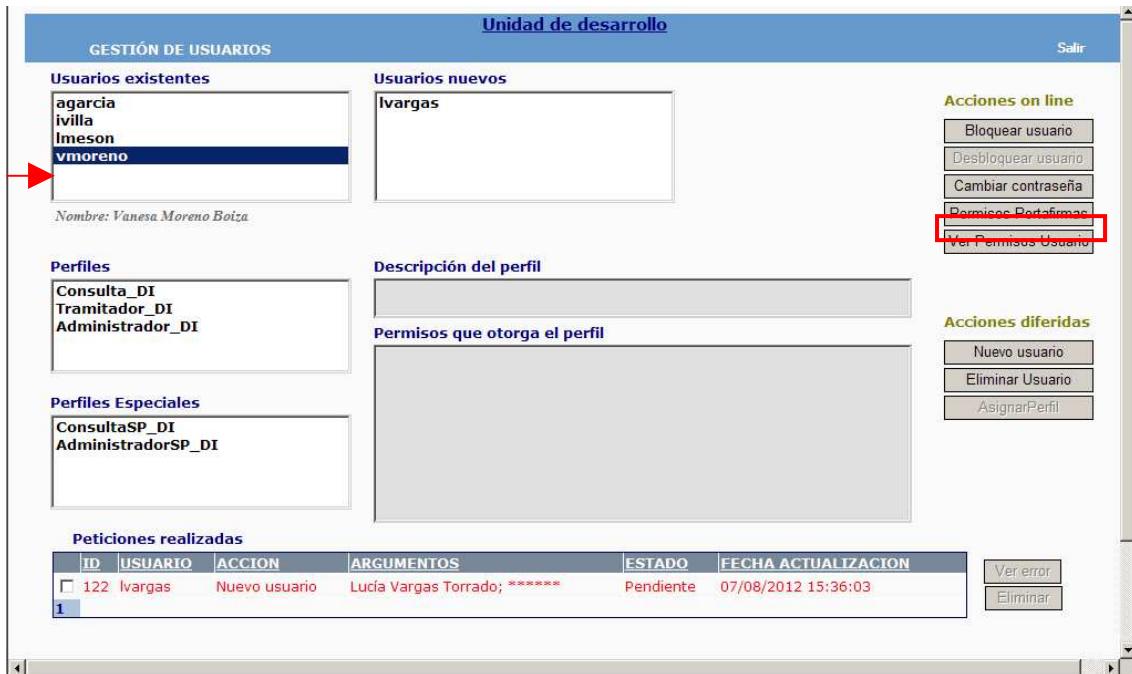


Figura 169. Permisos portafirmas

Se abre una pantalla en la que aparecerán los cargos firmantes. Si aparece alguno de los cargados marcado , indica que el usuario tiene asignado ese permiso. Se pueden modificar los datos, siempre quedaran asignados los valores que hayamos seleccionado antes de pulsar el botón *aceptar*.

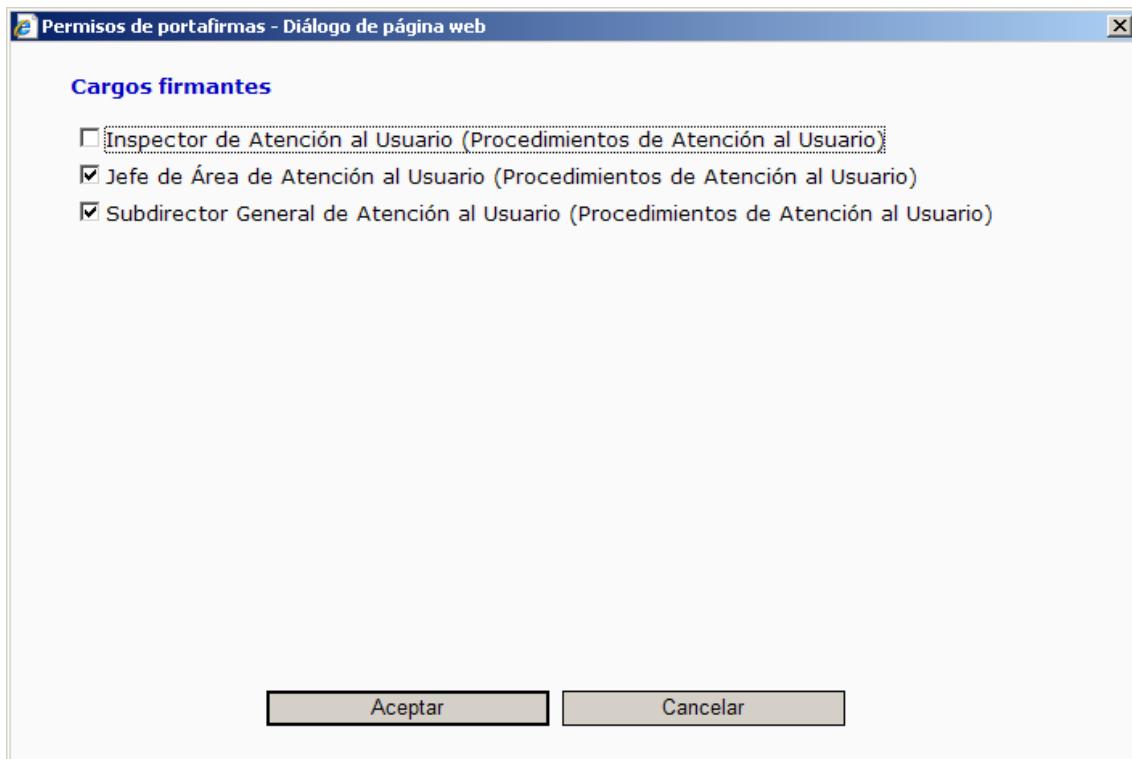


Figura 170. Cargos firmantes

Una vez seleccionados los permisos y pulsar el botón *aceptar* aparecerá un mensaje informativo de la correcta asignación de los permisos.



Figura 171. Confirmación permisos de firma modificados.

Si el departamento no dispone de permisos de firma, la aplicación le informará mediante un mensaje.



1.2.4 Permisos grupos

Esta funcionalidad permite conocer al momento los permisos o grupos asignados a un usuario. Para ello primero será seleccionar de la caja de *usuarios existentes* y a continuación pulsar sobre el botón **Permisos Grupos**. Aparecerá una ventana con una lista de los permisos que tiene asignados el usuario, según los grupos a los que pertenezca, así como una pequeña descripción de los mismos.

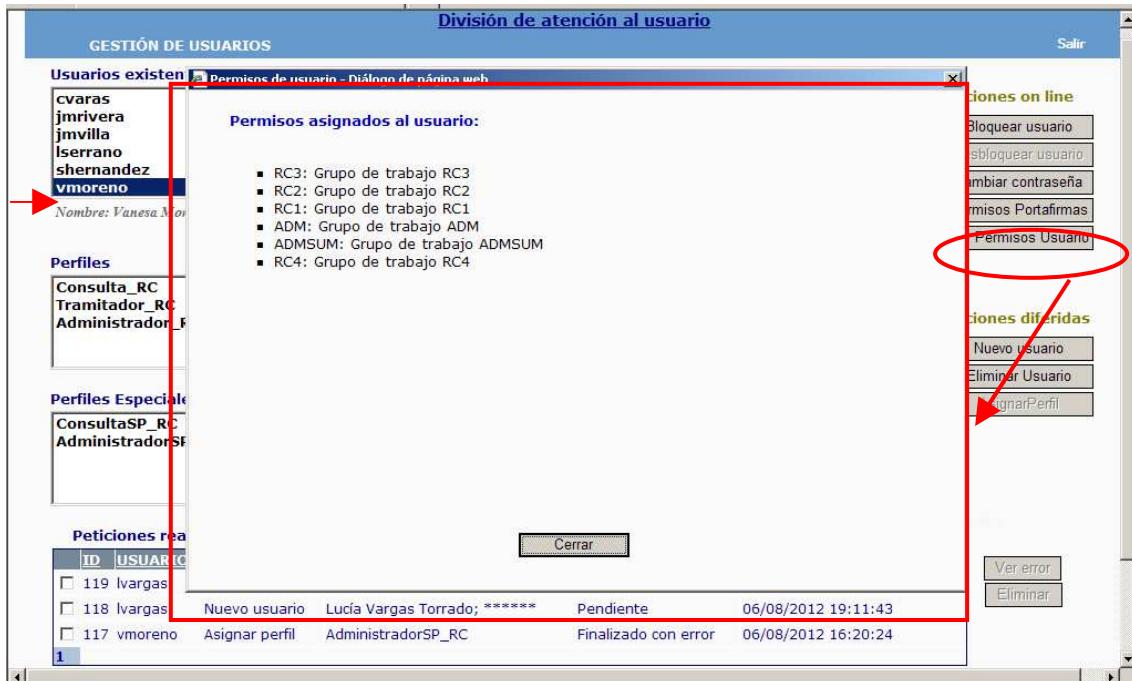


Figura 172. Permisos de usuario

Cabe la posibilidad de que el usuario elegido no esté adscrito a ningún grupo. En ese caso la aplicación le avisará mediante una ventana informativa como se muestra a continuación:

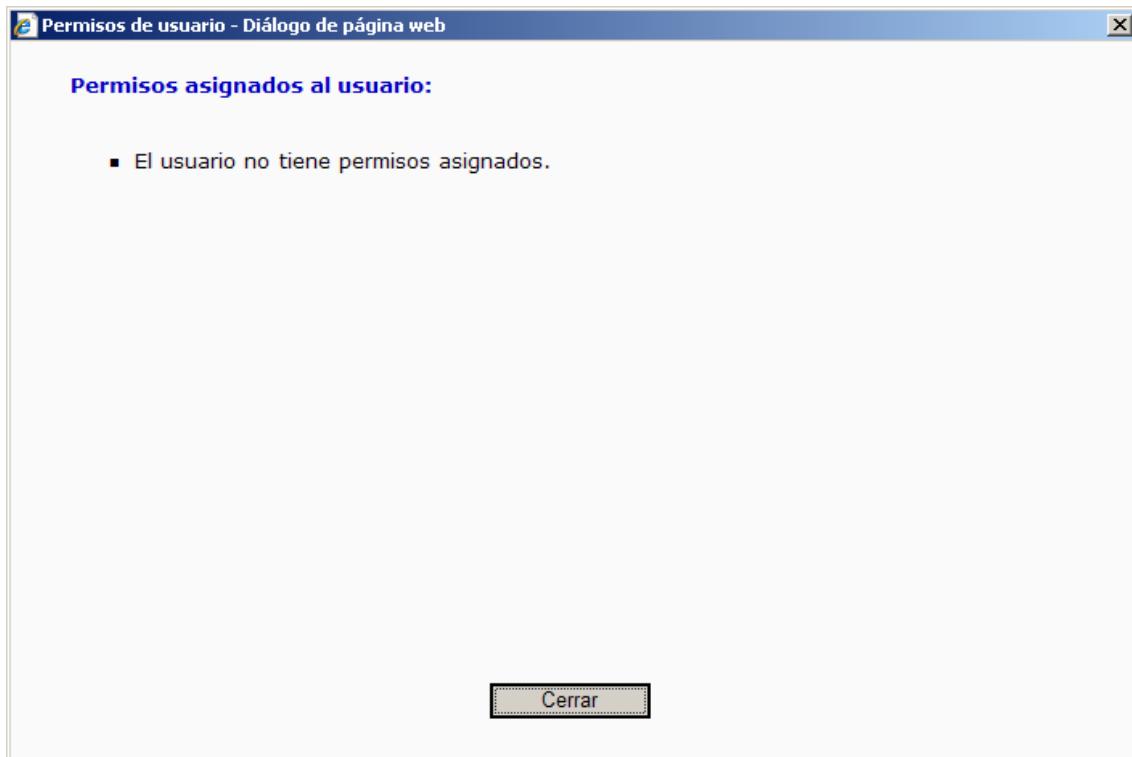


Figura 173. Mensaje de usuario sin permisos

1.2.5 Consulta descripción de perfil

Podemos conocer las características de un perfil, tan solo es necesario seleccionar el perfil deseado y aparecerá su descripción en el recuadro *descripción de perfil* y su contenido en el recuadro *permisos que otorga el perfil* en el que mostrará la lista de grupos asociados a ese perfil.

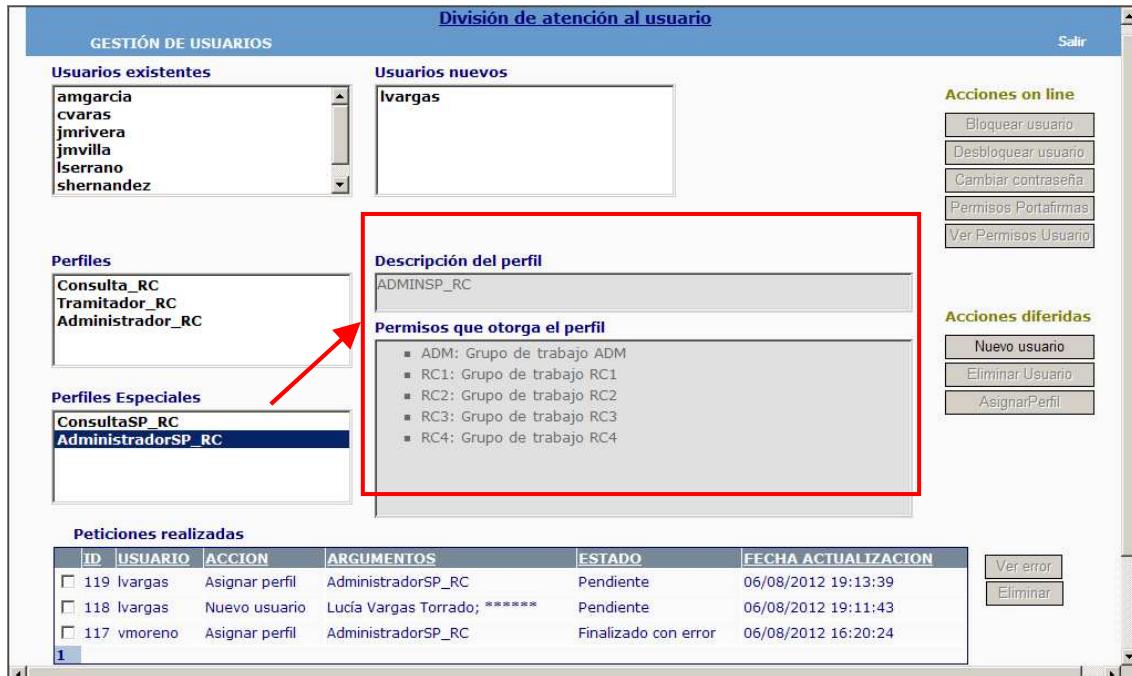


Figura 174. Información de perfil

El proyecto *Análisis y Diseño de una plataforma Web para un sistema de gestión de usuarios* consiste en el desarrollo de un sistema SUM que permita la gestión de usuarios a nivel departamental. Se compone de dos módulos SUM Web y SUM Server que cubren la funcionalidad requerida.

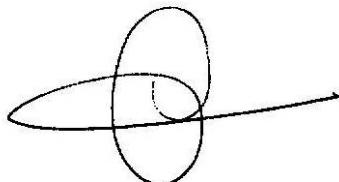
Las fases que abordan la realización del proyecto, siguiendo el estándar de la ESA [1] son las siguientes fases:

- Definición de requisitos
- Análisis
- Diseño
- Implementación y Evaluación
- Implantación

El presupuesto total de este proyecto asciende a la cantidad de SETENTA Y NUEVE MIL NOVECIENTOS CINCUENTA Y DOS EUROS”.

Leganés a 29 de octubre de 2012

El ingeniero proyectista



Fdo. Vanesa Moreno Boiza

ANEXO VI. GLOSARIO

Este apartado describe el glosario de términos específicos utilizado en el documento actual:

Término	Descripción
.NET	<i>Framework</i> de Microsoft que hace un énfasis en la transparencia de redes, con independencia de plataforma de hardware, permitiendo un rápido desarrollo de aplicaciones.
ADO.NET	Microsoft <i>ActiveX Data Objects</i> . Conjunto de clases que exponen servicios de acceso a datos para el programador de .NET.
ASP.NET	<i>Framework</i> para aplicaciones Web desarrollado por Microsoft. Es usado por programadores para construir sitios Web dinámicos, aplicaciones Web y servicios Web XML.
ASPX	Extensión de los páginas (formularios Web) de ASP.NET
Batch	Se denomina proceso <i>batch</i> , a la ejecución de un programa sin el control o supervisión directa del. Este tipo de programas se caracterizan porque su ejecución no precisa ningún tipo de interacción con el usuario. En el documento se hace referencia a un módulo de servidor.
BD	Base de datos. Colección de información. Sistema de archivos electrónico.
Bit	Dígito del sistema de numeración binario que representa un 0 o un 1.
Byte	Unidad básica de almacenamiento binario de datos compuesta por una secuencia de 8 bits.
Cliente-servidor	La arquitectura cliente-servidor es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, que le da respuesta.
CLR	<i>Common Language Runtime</i> . Motor en tiempo de ejecución de .NET <i>Framework</i> . Se encarga de ejecutar el código de los lenguajes del .NET <i>Framework</i> .
CLS	Determina las reglas necesarias para crear código MSIL compatible con el CLR de .NET
DLL	<i>Dinamyc Link Library</i> (Biblioteca de enlace dinámico). Archivos de código ejecutable que se cargan bajo demanda de un programa por parte del sistema operativo.
ECMAScript	Especificación de lenguaje de programación, basado en lenguaje JavaScript. Propuesto como estándar por Netscape Communications Corporation.
ESA	Conjunto de estándares definidos por la Agencia Espacial Europea para la especificación, desarrollo y mantenimiento del software.
EXE	Extensión de fichero que denota un fichero ejecutable (un programa) en los sistemas operativos MSDOS y Microsoft Windows.
Flash	Programa informático para edición multimedia.

Framework	Estructura de soporte (plataforma, entorno, marco de trabajo) definida, en la cual un proyecto de software puede ser organizado y desarrollado.
GAC	<i>Global Assembly Cache.</i> Mecanismo implícito de gestión de memoria implementado en algunos lenguajes de programación de tipo interpretado o semiinterpretado.
Gantt	Diagrama de Gantt es una herramienta gráfica cuyo objetivo es mostrar el tiempo de dedicación previsto para diferentes tareas o actividades a lo largo de un tiempo total determinado.
GB	Gigabyte. Unidad de almacenamiento de información que equivale a 10^9 bytes.
GEXTEL	Sistema de Gestión de Expedientes de Telecomunicaciones.
GHz	Gigahercio. Unidad de almacenamiento de información que equivale a 10^9 Hz.
HTTP	<i>HyperText Transfer Protocol</i> (Protocolo de transferencia de hipertexto) es el método más común de intercambio de información en la <i>World Wide Web</i> , el método mediante el cual se transfieren las páginas Web a un ordenador.
HTML	Lenguaje utilizado fundamentalmente en el desarrollo de páginas Web, comúnmente para establecer la estructura y contenido de un sitio Web, tanto de texto, objetos e imágenes. Los archivos desarrollados en HTML tienen la extensión .htm o .html
Hz	Unidad de frecuencia del Sistema Internacional de Unidades.
IDE	Entorno de Desarrollo Integrado. Aplicación compuesta por un conjunto de herramientas útiles para la programación.
IE o IExplorer	Navegador Web desarrollado por Microsoft para el sistema operativo Microsoft Windows.
IIS	<i>Internet Information Server</i> es un servidor de Microsoft destinado a la publicación, mantenimiento y gestión de páginas Web.
Internet	Conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas heterogéneas que la componen funcionen como una red lógica única, de alcance mundial.
Java	Lenguaje de programación de alto nivel orientado a objetos.
JavaScript (JScript)	Lenguaje de programación utilizado principalmente en la parte cliente en el desarrollo de páginas Web.
JIT	<i>Just in time.</i> Un compilador JIT (Just-In-Time) genera el código máquina que se ejecuta en la plataforma que tenga la computadora.
JP	Jefatura Provincial.
JPIT	Jefatura Provincial de Inspección de Telecomunicaciones.
LDAP	<i>Lightweight Directory Access Protocol</i> ó Protocolo Ligero de Acceso a Directorios. Es un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar información en un entorno de red.
LINQ	<i>Language Integrate Query.</i> Conjunto de extensiones de .NET Framework que habilitan la definición de consultas (de tipo SQL)

	integradas en el lenguaje de programación.
MD5	<i>Message-Digest Algorithm 5.</i> Algoritmo de reducción criptográfico de 128 bits ampliamente usado. Diseñado por Ronald Rivest en 1991.
MSIL	Código intermedio generado por el compilador de la herramienta de desarrollo de .NET
MSQLS	Microsoft SQL Server. Es un sistema de gestión de bases de datos relacionales producido por Microsoft que funciona sobre una plataforma Windows.
MVC	Modelo Vista Controlador. Patrón de arquitectura de software, utilizado en aplicaciones Web, encargado de separar la lógica de negocio de la interfaz del usuario.
MVS	Microsoft Visual Studio. Es un entorno de desarrollo integrado (IDE) para sistemas operativos Windows, orientado al entorno de desarrollo de software de .NET Framework.
Navegador Web	Un navegador Web o explorador Web es una aplicación software libre que permite al usuario recuperar y visualizar documentos de hipertexto, comúnmente descritos en HTML, desde servidores Web de todo el mundo a través de Internet.
ODBC	<i>Open Database Connectivity.</i> Estándar de acceso a Bases de Datos desarrollado por Microsoft Corporation, cuyo objetivo es permitir el acceso a cualquier base de datos desde cualquier aplicación, con independencia del SGBD que almacene los datos.
PDA	Asistente personal digital.
Perfil	Conjunto de permisos o grupos de acceso.
SD	Diagrama de secuencia. Diagrama UML utilizado para modelar la interacción entre objetos de un sistema.
SDSS	Diagrama de secuencia SUM Server
SDSW	Diagrama de secuencia SUM Web
SGBD	Sistema Gestor de Base de Datos. Programa que permite definir, crear y utilizar una base de datos.
SQL	<i>Structured Query Language.</i> Lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre los datos. Una de sus características es el manejo del álgebra y del cálculo relacional permitiendo efectuar consultas con el fin de recuperar información de interés de una base de datos, así como también hacer cambios sobre la misma.
SQL Server	Sistema de gestión de bases de datos relacionales producido por Microsoft que funciona sobre una plataforma Windows.
SR	Requisito del software.
STM	<i>System Tasks Management.</i> Sistema de planificación de tareas.
SUM	<i>System Users Management.</i> Sistema de gestión de usuarios.
SUM Server	<i>System Users Management Server.</i> Sistema servidor de gestión de usuarios.
SUM Web	<i>System Users Management Web.</i> Sistema Web de gestión de usuarios.

SSL	<i>Secure Sockets Layer</i> (capa de conexión segura). Protocolo criptográfico que proporciona comunicaciones seguras por una red, comúnmente Internet.
TI	Tecnologías de la información.
Transact-SQL	Extensión del lenguaje SQL, propiedad de Microsoft.
UC	Caso de uso. Secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre un sistema.
UML	<i>Unified Modeling Language</i> . Lenguaje unificado de modelado de sistemas software.
UR	Requisito de usuario.
URL	<i>Uniform Resource Locator</i> . Localizador de recursos uniforme. Secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos en Internet para su localización o identificación.
VB o Visual Basic	Aplicación y lenguaje de programación de alto nivel desarrollados por Alan Cooper para Microsoft.
VB.NET	Lenguaje de programación de alto nivel de .NET Framework
VPN	Virtual Private Network (red privada virtual). Tecnología de red que permite una extensión segura de la red local sobre una red pública o no controlada.
WAP	Wireless Application Protocol . Protocolo de aplicaciones inalámbricas.
Web o World Wide Web	Sistema de distribución de información basado en hipertexto o hipermedios enlazados y accesibles a través de Internet.
Windows	Familia de sistemas operativos desarrollados por Microsoft.
Windows XP	Versión de Microsoft Windows.
WML	Wireless Markup Language. Lenguaje de origen XML para construir páginas para telefonía móvil y PDA dotados de tecnología WAP.
XML	<i>Extensible Markup Language</i> . Lenguaje de marcas que permite definir la gramática de lenguajes específicos para estructurar documentos grandes.

Tabla 109. Glosario de términos

ANEXO VII. REFERENCIAS

- ESA *lite*.** (Octubre 2011)
[1] <http://www.ie.inf.uc3m.es/grupo/docencia/reglada/Is1y2/ESA/BSSC962-ES.PDF>
- Concurrencia optimista.** (Enero 2012)
[2] [http://msdn.microsoft.com/es-es/library/aa0416cz\(v=vs.80\).aspx](http://msdn.microsoft.com/es-es/library/aa0416cz(v=vs.80).aspx)
- BD.**
[3] Elmasri R. y Navathe, S. B. (2007). Fundamentos de Sistemas de Bases de Datos. Pearson Addison Wesley.
- [4] A. de Miguel, P. Martínez, E. Castro, J. M: Cavero, D. Cuadra, A. Iglesias, C. Nieto. Diseño de Bases de Datos: Problemas Resueltos. RA-MA (2001).
- Ingeniería de software.**
[5] Ingeniería del Software. Un Enfoque Práctico. Roger Pressman (Editorial McGraw-Hill)
- [6] Ingeniería de Software. Ian Sommerville. (Prentice Hall)
- UML.** (Noviembre 2011)
[7] UML y patrones. Introducción al análisis y diseño orientado a objetos. Craig Larman. Pearson Addison Wesley
[8] <http://msdn.microsoft.com/es-es/library/dd409465.aspx>
[9] <http://msdn.microsoft.com/es-es/library/dd465153.aspx>
[10] <http://msdn.microsoft.com/es-es/library/dd409389.aspx>
[11] <http://msdn.microsoft.com/es-es/library/bb972214.aspx>
- MVC.** (Mayo 2012)
[12] <http://msdn.microsoft.com/en-us/library/ff649643.aspx>
- MVC-Web Forms. .** (Mayo 2012)
[13] [http://msdn.microsoft.com/es-es/library/dd381619\(v=vs.98\).aspx](http://msdn.microsoft.com/es-es/library/dd381619(v=vs.98).aspx)
[14] [http://msdn.microsoft.com/es-es/library/cc437550\(v=vs.71\).aspx](http://msdn.microsoft.com/es-es/library/cc437550(v=vs.71).aspx)
- Framework .NET .** (Mayo 2012)
[15] <http://msdn.microsoft.com/es-es/library/zw4w595w.aspx>
- Seguridad de la información. .** (Mayo 2012)
[16] Cryptography and Network Security. Principles and Practice. William Stallings. Prentice Hall, 1999.
- MD5.** (Enero 2012)
[17] <http://msdn.microsoft.com/es-es/library/system.security.cryptography.md5.aspx>

