

UNIVERSIDAD HISPANOAMERICANA
ESCUELA DE INGENIERÍA INFORMÁTICA
TESINA PARA OPTAR POR EL GRADO DE BACHILLERATO EN INGENIERÍA
INFORMÁTICA

Sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca
Sequeira y asociados.

Sustentante:

Angie López Villegas

Tutor:

Luis Navarro Sánchez

II Cuatrimestre, 2021

Tabla de contenido

Índice de figuras.....	viii
Índice de tablas	xii
Declaración jurada	xvi
Carta tutor	xvii
Carta lector.....	xviii
Carta de autorización del autor para la consulta	xix
Dedicatoria.....	xx
Agradecimientos	xxi
CAPÍTULO I: PLANTEAMIENTO DEL TEMA.....	1
Introducción capítulo I	2
1.1 Antecedentes y justificación del proyecto.....	3
1.1.1 Marco de referencia empresarial y contextual.....	3
1.1.1.1 Información general	3
1.1.1.2. Misión	3
1.1.1.3. Visión.....	3
1.1.1.4. Objetivos	3
1.1.1.5 Organización	4
1.1.1.6 Historia de la organización	4
1.1.1.7 Tendencia en el mercado	5
1.1.2 Justificación del proyecto	6
1.2 Definición del problema.....	6
1.2.1 Problemática.....	6
1.2.1.1 Diagrama de Ishikawa.....	7

1.2.2 Problema general	7
1.2.3 Problemas específicos.....	8
1.3 Objetivos	8
1.3.1 Objetivo general	8
1.3.2 Objetivos específicos.....	8
1.4 Alcance y limitaciones	9
1.4.1 Alcance del proyecto	9
1.4.2 Limitaciones del proyecto	10
1.5 Cronograma del proyecto	11
CAPÍTULO II: MARCO TEÓRICO	12
Introducción capítulo II.....	13
2.1 Ámbito legal.....	14
2.1.1 Caso	15
2.1.2 Expediente	15
2.1.3 Despacho jurídico.....	15
2.1.4 Partes	15
2.1.5 Representante legal.....	15
2.1.6 Notificación	16
2.1.7 Apelación.....	16
2.2 Ámbito informático.....	16
2.2.1 Software.....	17
2.2.1.1 Sistema de gestión.....	19
2.2.1.1.1 Gestión	19
2.2.1.1.2 Sistema de información.....	20
2.2.1.1.2.1 Ciclo de vida del software	20

2.2.1.1.3 Metodologías de programación.....	22
2.2.1.1.3.1 Modelo de cascada.....	22
2.2.1.2 Internet	23
2.2.1.2.1 Web	23
2.2.1.2.2 Internet Information Services (IIS)	23
2.2.1.2.3 Arquitecturas de software.....	24
2.2.1.2.3.1 Cliente – servidor.....	24
2.2.1.2.4 Aplicaciones web	28
2.2.1.2.5 API	31
2.2.1.2.6 Host	32
2.2.1.2.7 Dominio.....	32
2.2.1.3 Análisis del proyecto.....	33
2.2.1.3.1 Historia de Usuario.....	34
2.2.1.3.2 Módulo	35
2.2.1.3.3 Requerimientos.....	35
2.2.1.3.3.1 Funcionales.....	36
2.2.1.3.3.2 No funcionales.....	36
2.2.1.3.4 Características de requerimientos.....	37
2.2.1.4 Diseño del proyecto	37
2.2.1.4.1 Diagrama.....	37
2.2.1.4.2 UML	38
2.2.1.4.2.1 Diagrama de casos de uso.....	39
2.2.1.4.2.2 Diagrama de clases	42
2.2.1.4.2.3 Diagrama de secuencia	45
2.2.1.4.2.4 Diagrama de actividad	48

2.2.1.4.3 Diagrama entidad-relación	50
2.2.1.5 Desarrollo del proyecto.....	51
2.2.1.5.1 Base de datos.....	51
2.2.1.5.1.1 Tabla.....	52
2.2.1.5.1.2 Relación.....	53
2.2.1.5.2 Microsoft SQL Server	53
2.2.1.5.3 Lenguaje de programación.....	54
2.2.1.5.3.1 HTML.....	55
2.2.1.5.3.2 CSS.....	55
2.2.1.5.3.3 Javascript	56
2.2.1.5.3.4 JQuery.....	57
2.2.1.5.3.5 Bootstrap.....	57
2.2.1.5.3.6 Owl Carousel	58
2.2.1.5.3.7 C#.....	58
2.2.1.5.3.8 Microsoft Visual Studio.....	59
2.2.1.5.3.9 Modelo Vista Controlador	59
2.2.1.5.3.10 Razor.....	62
2.2.1.5.3.11 Entity Framework.....	62
2.2.1.5.3.12 Linq.....	64
2.2.1.5.3.13 ASP.NET MVC	64
2.2.1.5.4 Pruebas	64
2.2.1.6 Implementación del proyecto.....	65
2.3 Tecnología legal	65
2.3.1 Iuscibernética.....	65
2.3.1.1 Informática jurídica.....	66

2.3.2 Legaltech	71
CAPÍTULO III: MARCO METODOLÓGICO	72
Introducción capítulo III.....	73
3.1 Tipo y enfoque de la investigación	74
3.1.1 Tipo de investigación.....	74
3.1.2 Enfoque de la investigación.....	74
3.2 Fuentes y sujetos de información	74
3.2.1 Fuentes de información	74
3.2.1.1 Fuentes primarias	75
3.2.1.2 Fuentes secundarias	75
3.2.2 Sujetos de información.....	75
3.3 Técnicas y herramientas de recolección de datos.....	76
3.3.1 Entrevista.....	76
3.4 Variables de la investigación.....	77
3.5 Diseño de la investigación.....	78
3.6 Matriz de coherencia	79
CAPÍTULO IV: DIAGNÓSTICO	86
Introducción capítulo IV	87
4.1 Descripción de la situación actual.....	88
4.1.1 Diagnóstico administrativo u operativo.....	88
4.1.2 Diagnóstico técnico	92
4.1.3 Diagnóstico de percepción.....	95
4.2 Determinación de brechas y conclusiones	95
CAPÍTULO V: PROPUESTA DEL PROYECTO	98
Introducción capítulo V.....	99

5.1 Análisis del proyecto.....	100
5.1.1 Requerimientos funcionales	100
5.1.2 Requerimientos no funcionales	123
5.1.3 Historias de usuario	125
5.2 Diseño del proyecto.....	148
5.2.1 Casos de uso	148
5.2.1.1 Manejos de sesión	148
5.2.1.2 Módulo de clientes.....	152
5.2.1.3 Módulo de expedientes	156
5.2.1.4 Módulo de reportes	161
5.2.1.5 Módulo de agenda.....	163
5.2.1.6 Módulo de mantenimiento	172
5.2.1.7 Módulo de Honorarios	179
5.2.2 Diagrama de clases	182
5.2.3 Diagrama de actividades.....	185
5.2.4 Diagrama de secuencia	212
5.2.5 Diagrama de entidad-relación.....	231
5.3 Desarrollo del proyecto	233
5.3.1 Base de datos	233
5.3.2 Pantallas de la aplicación web	251
5.4 Implementación del proyecto	273
5.4.1 Implementación	273
CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES	274
Introducción capítulo VI	275
6.1 Conclusiones	276

6.2 Recomendaciones.....	277
CAPÍTULO VII: APÉNDICES Y ANEXOS.....	278
7.1 Referencias Bibliográficas	279
7.2 Anexos.....	284
7.2.1 Entrevistas con el abogado	284
7.2.2 Carta del abogado	289

Índice de figuras

Figura 1. Organigrama del bufete	4
Figura 2. Diagrama de Ishikawa.	7
Figura 3. Cronograma del proyecto.	11
Figura 4. Esquema marco teórico	14
Figura 5. Fases del ciclo de vida de un proyecto.	21
Figura 6. Modelo Cliente/Servidor.	25
Figura 7. Modelo Cliente/Servidor de 2 capas.....	26
Figura 8. Modelo Cliente/Servidor multicapa	28
Figura 9. Relación entre una aplicación, una API y una librería	31
Figura 10. Dominios de primer nivel.....	33
Figura 11. Tipos de relación de diagramas de caso de uso.	41
Figura 12. Cuatro tipos de relaciones además de flechas y líneas de comportamiento de UML usadas para representar las relaciones.....	41
Figura 13. Ejemplo de relaciones todo/parte y de agregación.	45
Figura 14. Símbolos especializados usados para dibujar un diagrama de secuencias.	47
Figura 15. Símbolos especializados para dibujar diagramas de actividad.....	49
Figura 16. Elementos de un diagrama Entidad-Relación.....	51
Figura 17. Arquitectura de Entity Framework.....	63
Figura 18. Diseño de la investigación.....	78
Figura 19. Agenda del abogado, 2018.	89
Figura 20. Agenda del abogado, 2018.	90
Figura 21. Agenda del abogado, 2019.	90
Figura 22. Agenda del abogado, 2019.	91
Figura 23. Pizarra en oficina del abogado.	92
Figura 24. Computador de trabajo	93
Figura 25. Procesador del computador.	94
Figura 26. Computador del abogado.....	95
Figura 27. Diagrama de caso de uso, Manejos de sesión.....	149
Figura 28. Diagrama de caso de uso, Módulo de clientes	152
Figura 29. Diagrama de caso de uso 03, Módulo de expedientes.....	157

Figura 30. Diagrama de caso de uso 04, Módulo de reportes.....	161
Figura 31. Diagrama de caso de uso 05, Módulo de agenda, citas.....	163
Figura 32. Diagrama de caso de uso 06, Módulo de agenda, recordatorios.....	164
Figura 33. Diagrama de caso de uso 07, Módulo de mantenimiento.....	172
Figura 34. Diagrama de caso de uso 08, Módulo de honorarios.....	179
Figura 35. Diagrama de clases, ViewModel.....	183
Figura 36. Diagrama de clases, EntityFramework.....	184
Figura 37. Diagrama de actividad, Iniciar sesión.....	185
Figura 38. Diagrama de actividad, Recuperar contraseña.....	186
Figura 39. Diagrama de actividad, Cerrar sesión.....	187
Figura 40. Diagrama de actividad, Agregar clientes.....	188
Figura 41. Diagrama de actividad, Consultar cliente.....	189
Figura 42. Diagrama de actividad, Modificar cliente.....	190
Figura 43. Diagrama de actividad, Eliminar cliente.....	191
Figura 44. Diagrama de actividad, Generar reporte.....	191
Figura 45. Diagrama de actividad, Agregar honorario.....	192
Figura 46. Diagrama de actividad, Agregar abonos.....	193
Figura 47. Diagrama de actividad, Consultar honorario.....	193
Figura 48. Diagrama de actividad, Agregar expediente.....	194
Figura 49. Diagrama de actividad, Modificar expediente.....	195
Figura 50. Diagrama de actividad, Agregar caso.....	196
Figura 51. Diagrama de actividad, Consultar expediente.....	197
Figura 52. Diagrama de actividad, Crear usuario.....	198
Figura 53. Diagrama de actividad, Consultar usuario.....	199
Figura 54. Diagrama de actividad, Modificar usuario.....	200
Figura 55. Diagrama de actividad, Eliminar usuario.....	201
Figura 56. Diagrama de actividad, Agregar tipo de proceso.....	202
Figura 57. Diagrama de actividad, Consultar usuario.....	203
Figura 58. Diagrama de actividad, Consultar usuario.....	204
Figura 59. Diagrama de actividad, Agregar cita.....	205
Figura 60. Diagrama de actividad, Consultar cita.....	206

Figura 61. Diagrama de actividad, Modificar cita.	207
Figura 62. Diagrama de actividad, Cancelar cita.	208
Figura 63. Diagrama de actividad, Agregar recordatorio.	209
Figura 64. Diagrama de actividad, Consultar recordatorio.	210
Figura 65. Diagrama de actividad, Modificar recordatorio.	211
Figura 66. Diagrama de actividad, Eliminar recordatorio.	212
Figura 67. Diagrama de secuencia, Iniciar sesión.	213
Figura 68. Diagrama de secuencia, Recuperar contraseña	214
Figura 69. Diagrama de secuencia, Agregar cliente.	215
Figura 70. Diagrama de secuencia, Buscar cliente.	216
Figura 71. Diagrama de secuencia, Eliminar cliente.	216
Figura 72. Diagrama de secuencia, Modificar cliente.	216
Figura 73. Diagrama de secuencia, Crear usuario.	217
Figura 74. Diagrama de secuencia, Consultar usuario.	218
Figura 75. Diagrama de secuencia, Eliminar cliente.	218
Figura 76. Diagrama de secuencia, Modificar usuario.	219
Figura 77. Diagrama de secuencia, Agregar tipo de proceso.	220
Figura 78. Diagrama de secuencia, Eliminar tipo de proceso.	220
Figura 79. Diagrama de secuencia, Cambiar contraseña.	220
Figura 80. Diagrama de secuencia, Agregar honorario.	221
Figura 81. Diagrama de secuencia, Agregar abonos.	222
Figura 82. Diagrama de secuencia, Consultar honorario.	223
Figura 83. Diagrama de secuencia, Generar reporte.	223
Figura 84. Diagrama de secuencia, Agregar expediente.	224
Figura 85. Diagrama de secuencia, Modificar expediente.	224
Figura 86. Diagrama de secuencia, Buscar expediente.	225
Figura 87. Diagrama de secuencia, Agregar caso.	226
Figura 88. Diagrama de secuencia, agregar cita.	226
Figura 89. Diagrama de secuencia, agregar recordatorio.	227
Figura 90. Diagrama de secuencia, consultar cita.	227
Figura 91. Diagrama de secuencia, consultar recordatorio.	228

Figura 92. Diagrama de secuencia, modificar cita.....	229
Figura 93. Diagrama de secuencia, modificar recordatorio.....	229
Figura 94. Diagrama de secuencia, eliminar cita.....	230
Figura 95. Diagrama de secuencia, eliminar recordatorio.....	231
Figura 96. Diagrama entidad-relación.....	232
Figura 97. Diagrama de base de datos.....	233
Figura 98. Pantalla de inicio de sesión.....	252
Figura 99. Pantalla de recuperación de contraseña.....	253
Figura 100. Menú vertical de abogado.....	254
Figura 101. Pantalla principal de administrador.....	254
Figura 102. Pantalla de reportes.....	255
Figura 103. Vista principal Honorarios.....	256
Figura 104. Pantalla agregar honorario para abogado.....	257
Figura 105. Agregar honorario para administrador.....	258
Figura 106. Agregar abono.....	259
Figura 107. Pantalla Ver abonos.....	260
Figura 108. Pantalla expediente no registrado.....	261
Figura 109. Pantalla expediente registrado.....	263
Figura 110. Pantalla de mantenimiento, vista para abogado.....	264
Figura 111. Pantalla mantenimiento para administrador.....	265
Figura 112. Pantalla de módulo de clientes, vista principal.....	266
Figura 113. Pantalla de clientes, vista con cliente cargado.....	268
Figura 114. Pantalla crear cliente.....	269
Figura 115. Pantalla principal.....	270
Figura 116. Pantalla agregar evento.....	271
Figura 117. Pantalla de modificar evento, parte 1.....	272
Figura 118. Pantalla de modificar evento, parte 2.....	272

Índice de tablas

Tabla 1. Sujetos de información.	76
Tabla 2. Variables de la investigación.	77
Tabla 3. Diseño de la investigación.	79
Tabla 4. Matriz de coherencia.....	85
Tabla 5. Determinación de brechas del diagnóstico.	97
Tabla 6. Requerimiento funcional 01 Iniciar sesión.	100
Tabla 7. Requerimiento funcional 02, Consultar usuario.	101
Tabla 8. Requerimiento funcional 03, Crear usuario.	102
Tabla 9. Requerimiento funcional 04, Modificar usuario.	102
Tabla 10. Requerimiento funcional 05, Eliminar usuario.	103
Tabla 11. Requerimiento funcional 05, Cambiar contraseña.	104
Tabla 12. Requerimiento funcional 07, Agregar tipo de proceso.	104
Tabla 13. Requerimiento funcional 08, Eliminar tipo de proceso.	105
Tabla 14. Requerimiento funcional 09, Consultar cliente.	106
Tabla 15. Requerimiento funcional 10, Agregar cliente.	106
Tabla 16. Requerimiento funcional 11, Modificar cliente.	107
Tabla 17. Requerimiento funcional 12, Eliminar cliente.	108
Tabla 18. Requerimiento funcional 13, Re direccionar a expediente.	108
Tabla 19. Requerimiento funcional 14, Consultar expediente.	109
Tabla 20. Requerimiento funcional 15, Crear expediente.	110
Tabla 21. Requerimiento funcional 16, Modificar expediente.	110
Tabla 22. Requerimiento funcional 17, Agregar notificación.	111
Tabla 23. Requerimiento funcional 18, Consultar cita.	112
Tabla 24. Requerimiento funcional 19, Agregar cita.	112
Tabla 25. Requerimiento funcional 20, Modificar cita.	113
Tabla 26. Requerimiento funcional 21, Cancelar cita.	113
Tabla 27. Requerimiento funcional 22, Consultar recordatorio.	114
Tabla 28. Requerimiento funcional 23, Agregar recordatorio.	115
Tabla 29. Requerimiento funcional 24, Modificar recordatorio.	115
Tabla 30. Requerimiento funcional 25, Eliminar recordatorio.	116

Tabla 31. Requerimiento funcional 26, Crear reporte.	117
Tabla 32. Requerimiento funcional 27, Enviar email de actualización para notificaciones del expediente.	117
Tabla 33. Requerimiento funcional 28, Enviar email de cita.	118
Tabla 34. Requerimiento funcional 29, Roles de usuario.	118
Tabla 35. Requerimiento funcional 30, Consultar tipo de proceso.	119
Tabla 36. Requerimiento funcional 31, Cambiar contraseña.	120
Tabla 37. Requerimiento funcional 32, Enviar email con nueva contraseña.	120
Tabla 38. Requerimiento funcional 33, Enviar email al nuevo usuario.	121
Tabla 39. Requerimiento funcional 34, Consultar honorarios.	121
Tabla 40. Requerimiento funcional 35, Agregar honorarios.	122
Tabla 41. Requerimiento funcional 36, Agregar abonos.	123
Tabla 42. Requerimiento funcional 37, Agregar caso.	123
Tabla 43. Requerimiento no funcional 01, Desarrollo.	124
Tabla 44. Requerimiento no funcional 02, Base de datos.	124
Tabla 45. Requerimiento no funcional 03, Acceso y uso de la base de datos.	125
Tabla 46. Requerimiento no funcional 04, Diseño web.	125
Tabla 47. Historia de usuario 01, Iniciar sesión.	127
Tabla 48. Historia de usuario 02, Crear un nuevo usuario.	128
Tabla 49. Historia de usuario 03, Modificar usuario.	129
Tabla 50. Historia de usuario 04, Eliminar usuario.	130
Tabla 51. Historia de usuario 05, Recuperar contraseña.	131
Tabla 52. Historia de usuario 06, Agregar tipo de proceso.	132
Tabla 53. Historia de usuario 07, Eliminar tipo de proceso.	133
Tabla 54. Historia de usuario 08, Agregar cliente.	134
Tabla 55. Historia de usuario 09, Modificar cliente.	135
Tabla 56. Historia de usuario 10, Eliminar cliente.	136
Tabla 57. Historia de usuario 11, Crear expediente.	138
Tabla 58. Historia de usuario 12, Modificar expediente.	139
Tabla 59. Historia de usuario 13, Agregar notificación.	140
Tabla 60. Historia de usuario 14, Agregar cita.	141

Tabla 61. Historia de usuario 15, Modificar cita.	142
Tabla 62. Historia de usuario 16, Eliminar cita.	143
Tabla 63. Historia de usuario 17, Agregar recordatorio.	144
Tabla 64. Historia de usuario 18, Modificar recordatorio.	146
Tabla 65. Historia de usuario 19, Eliminar recordatorio.	147
Tabla 66. Historia de usuario 20, Crear reporte.....	148
Tabla 67. Caso de uso 01, Iniciar sesión.....	150
Tabla 68. Caso de uso 02, Recuperar contraseña.....	151
Tabla 69. Caso de uso 03, Cerrar sesión.....	152
Tabla 70. Caso de uso 04, Agregar cliente.	154
Tabla 71. Caso de uso 05, Consultar cliente.	154
Tabla 72. Caso de uso 06, Modificar cliente.	155
Tabla 73. Caso de uso 07, Eliminar cliente.	156
Tabla 74. Caso de uso 08, Agregar expediente.....	158
Tabla 75. Caso de uso 09, Consultar expediente.	159
Tabla 76. Caso de uso 10, Modificar expediente.....	160
Tabla 77. Caso de uso 11, Agregar caso.	161
Tabla 78. Caso de uso 12, Generar reporte.....	162
Tabla 79. Caso de uso 13, Agregar cita.	165
Tabla 80. Caso de uso 14, Consultar cita.....	166
Tabla 81. Caso de uso 15, Modificar cita.	167
Tabla 82. Caso de uso 16, Cancelar cita.	168
Tabla 83. Caso de uso 17, Agregar recordatorio.	169
Tabla 84. Caso de uso 18, Consultar recordatorio.....	170
Tabla 85. Caso de uso 19, Modificar recordatorio.....	171
Tabla 86. Caso de uso 20, Eliminar recordatorio.....	171
Tabla 87. Caso de uso 21, Crear usuario.	173
Tabla 88. Caso de uso 22, Consultar usuario.....	174
Tabla 89. Caso de uso 23, Modificar usuario.	175
Tabla 90. Caso de uso 24, Eliminar usuario.	176
Tabla 91. Caso de uso 25, Agregar tipo de proceso.....	177

Tabla 92. Caso de uso 26, Eliminar tipo de proceso.....	178
Tabla 93. Caso de uso 27, Cambiar contraseña.	179
Tabla 94. Caso de uso 28, Agregar honorarios.	180
Tabla 95. Caso de uso 29, Agregar abonos.....	181
Tabla 96. Caso de uso 30, Consultar honorarios.	182
Tabla 97. Tabla de clientes.	235
Tabla 98. Tabla de apelaciones.	235
Tabla 99. Tabla de citas.	236
Tabla 100. Tabla de cambios.	237
Tabla 101. Tabla de tipo de proceso.	238
Tabla 102. Tabla de expedientes.....	238
Tabla 103. Tabla de representante legal.	239
Tabla 104. Tabla de depósitos.	240
Tabla 105. Tabla de documentos.	242
Tabla 106. Tabla de partes involucradas.....	243
Tabla 107. Tabla de despachos judiciales.....	243
Tabla 108. Tabla de estado del proceso.....	244
Tabla 109. Tabla de notificaciones.	245
Tabla 110. Tabla de recordatorios.	246
Tabla 111. Tabla de usuarios.	247
Tabla 112. Tabla de teléfonos de los usuarios.	247
Tabla 113. Tabla de inicio de sesión.....	248
Tabla 114. Tabla de correos enviados.....	249
Tabla 115. Tabla de casos.	250
Tabla 116. Tabla de honorarios.	251
Tabla 117. Tabla de abonos a honorarios.	251

Declaración jurada

DECLARACIÓN JURADA

Yo Angie López Villegas, mayor de edad, portador de la cédula de identidad número 604460167 egresado de la carrera de Ingeniería Informática de la Universidad Hispanoamericana, hago constar por medio de éste acto y debidamente apercibido y entendido de las penas y consecuencias con las que se castiga en el Código Penal el delito de perjurio, ante quienes se constituyen en el Tribunal Examinador de mi trabajo de tesis para optar por el título de Bachillerato en Ingeniería Informática, juro solemnemente que mi trabajo de investigación titulado: Sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados., es una obra original que ha respetado todo lo preceptuado por las Leyes Penales, así como la Ley de Derecho de Autor y Derecho Conexos número 6683 del 14 de octubre de 1982 y sus reformas, publicada en la Gaceta número 226 del 25 de noviembre de 1982; incluyendo el numeral 70 de dicha ley que advierte; artículo 70. Es permitido citar a un autor, transcribiendo los pasajes pertinentes siempre que éstos no sean tantos y seguidos, que puedan considerarse como una producción simulada y sustancial, que redunde en perjuicio del autor de la obra original. Asimismo, quedo advertido que la Universidad se reserva el derecho de protocolizar este documento ante Notario Público.

En fe de lo anterior, firmo en la ciudad de San José, a los 07 días del mes de diciembre del año dos mil veintiuno.

Angie
Firma del estudiante
Cédula: 604460167

CARTA DEL TUTOR

San José, 07 de Diciembre, 2021

**Dirección de Carrera
Ingeniería Informática
Universidad Hispanoamericana**

Estimado(a)

La estudiante Angie López Villegas, cédula de identidad número 6-0446-0167, me ha presentado, para efectos de revisión y aprobación, el trabajo de investigación denominado "**Sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados**", el cual ha elaborado para optar por el grado académico de Bachillerato.

En mi calidad de tutor, he verificado que se han hecho las correcciones indicadas durante el proceso de tutoría y he evaluado los aspectos relativos a la elaboración del problema, objetivos, justificación; antecedentes, marco teórico, marco metodológico, tabulación, análisis de datos; conclusiones y recomendaciones.

De los resultados obtenidos por el postulante, se obtiene la siguiente calificación:

a)	ORIGINAL DEL TEMA	10%	7
b)	CUMPLIMIENTO DE ENTREGA DE AVANCES	20%	15
C)	COHERENCIA ENTRE LOS OBJETIVOS, LOS INSTRUMENTOS APLICADOS Y LOS RESULTADOS DE LA INVESTIGACION	30%	28
d)	RELEVANCIA DE LAS CONCLUSIONES Y RECOMENDACIONES	20%	18
e)	CALIDAD, DETALLE DEL MARCO TEORICO	20%	18
	TOTAL		86

En virtud de la calificación obtenida, se avala el traslado al proceso de lectura.

Atentamente,

Luis Navarro S
Cédula identidad 2-0484-0537

CARTA DE LECTOR

**Universidad Hispanoamericana
Sede Llorente
Escuela de Ingeniería Informática**

Estimados señores

El estudiante **Angie López Villegas**, cédula de identidad **604460167**, me ha presentado para efectos de revisión y aprobación, el trabajo de investigación denominado "**Sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados**", el cual ha elaborado para obtener su grado de Bachillerato en Ingeniería Informática.

He revisado y he hecho las observaciones relativas al contenido analizado, particularmente lo relativo a la coherencia entre el marco teórico y análisis de datos, la consistencia de los datos recopilados y la coherencia entre éstos y las conclusiones; asimismo, la aplicabilidad y originalidad de las recomendaciones, en términos de aporte de la investigación.

Por consiguiente, este trabajo cuenta con mi aval para ser presentado en la defensa pública.

Atte. **ALEJANDRO BOGANTES SALAZAR (FIRMA)**  Digitally signed by
ALEJANDRO
BOGANTES
SALAZAR (FIRMA)
Date: 2022.01.19
15:49:48 -06'00'

Firma: _____

**MSc. Alejandro Bogantes Salazar
Cédula: 303940389**

Carta de autorización del autor para la consulta

**UNIVERSIDAD HISPANOAMERICANA
CENTRO DE INFORMACION TECNOLOGICO (CENIT)
CARTA DE AUTORIZACIÓN DE LOS AUTORES PARA LA CONSULTA, LA
REPRODUCCION PARCIAL O TOTAL Y PUBLICACIÓN ELECTRÓNICA
DE LOS TRABAJOS FINALES DE GRADUACION**

Puntarenas, 27 de enero del 2022.

Señores:


Universidad Hispanoamericana
Centro de Información Tecnológico (CENIT)

Estimados Señores:

El suscrito (a) **Angie López Villegas** con número de identificación **604460167** autor (a) del trabajo de graduación titulado "**Sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados**" presentado y aprobado en el año **2021** como requisito para optar por el título de **Bachillerato en Ingeniería Informática**; **SI** autorizo al Centro de Información Tecnológico (CENIT) para que con fines académicos, muestre a la comunidad universitaria la producción intelectual contenida en este documento.

De conformidad con lo establecido en la Ley sobre Derechos de Autor y Derechos Conexos N° 6683, Asamblea Legislativa de la República de Costa Rica.

Cordialmente,


Firma y Documento de Identidad **604460167**

Dedicatoria

Este proyecto está dedicado primeramente a Dios, quien por Su misericordia se ha placido en permitirme llegar hasta aquí y tener la oportunidad de cumplir esta meta.

Y en segundo lugar, a mi familia, quienes han estado siempre para mí brindándome su apoyo y comprensión durante todos estos años, y más aún durante estos últimos meses mientras realizaba este proyecto.

Agradecimientos

Agradezco a Dios por permitirme llegar hasta este punto.

A mi mamá, por darme todo su apoyo y comprensión, por siempre brindarme una palabra o un abrazo cuando lo necesité durante este proceso y por estar para mí y animarme a continuar y esforzarme cuando yo ya no quería continuar.

A mi familia, por darme su apoyo, por animarme y soportarme durante este tiempo.

Agradezco a mis profesores de la Universidad, a mis compañeros de cursos, y a mis amigos, quienes de una u otra manera contribuyeron a mi formación académica.

Por último, agradezco el apoyo, la guía y motivación brindada por mi tutor, el profesor Luis Navarro Sánchez, para poder realizar este proyecto de graduación de la mejor manera.

CAPÍTULO I: PLANTEAMIENTO DEL TEMA

Introducción capítulo I

Este proyecto consta en una aplicación web para el bufete Fonseca Sequeira y asociados que tendrá como función la gestión de expedientes jurídicos. El sistema constará del módulo de clientes, para agregar, revisar y modificar información del cliente; el módulo de expedientes, que se encargará de la creación y actualización de los expedientes de los casos que el abogado esté llevando; el módulo de agenda, que permitirá agendar citas de atención al cliente; el módulo de reportes que permite generar reportes bajo ciertos parámetros; y el de configuraciones que permite el manejo de usuarios, entre otras cosas.

Este capítulo brinda información acerca de la empresa en la cual se realizará el proyecto, así mismo también se expone el contexto y la justificación del mismo. También se desarrollan los objetivos, alcances y delimitaciones para el proyecto de gestión de expedientes jurídicos para el bufete Fonseca Sequeira y asociados.

1.1 Antecedentes y justificación del proyecto

1.1.1 Marco de referencia empresarial y contextual

1.1.1.1 Información general

Nombre de la empresa: Bufete Fonseca Sequeira y asociados

Año de fundación: 2000

Negocio al que se dedica: Abogacía.

1.1.1.2. Misión

Defender los intereses personales y económicos de nuestros clientes para que puedan alcanzar sus objetivos con un asesoramiento transparente y comprensible basado en las leyes de Costa Rica, con la mayor confidencialidad y respeto.

1.1.1.3. Visión

Ser un despacho de referencia, capaz de dar soluciones legales de calidad, con eficiencia y la mayor rapidez, utilizando las tecnologías para una mayor efectividad, con esfuerzo, dedicación y especializaciones.

1.1.1.4. Objetivos

- Dar un servicio de calidad a cada cliente, asumir su defensa de manera responsable y ser consecuente con los principios éticos.
- Procurar la especialización constante en la defensa y asesoría de empresas y demás personas.
- Defender los derechos de las familias de este país ante los actos de injusticia realizados por personas o instituciones privadas o del Estado.
- Tener un alto aporte de ética profesional.

- Defender y promover los derechos humanos desde una perspectiva integral partiendo de su indivisibilidad y la interdependencia de todos los derechos y libertades.

1.1.1.5 Organización

Actualmente el bufete Fonseca Sequeira y asociados no cuenta con otros abogados aparte del dueño y tampoco cuenta con alguien encargado de agendar citas y de la atención al cliente, por lo que él es quien se encarga de todos los aspectos del bufete. La organización del mismo sería de la siguiente manera.

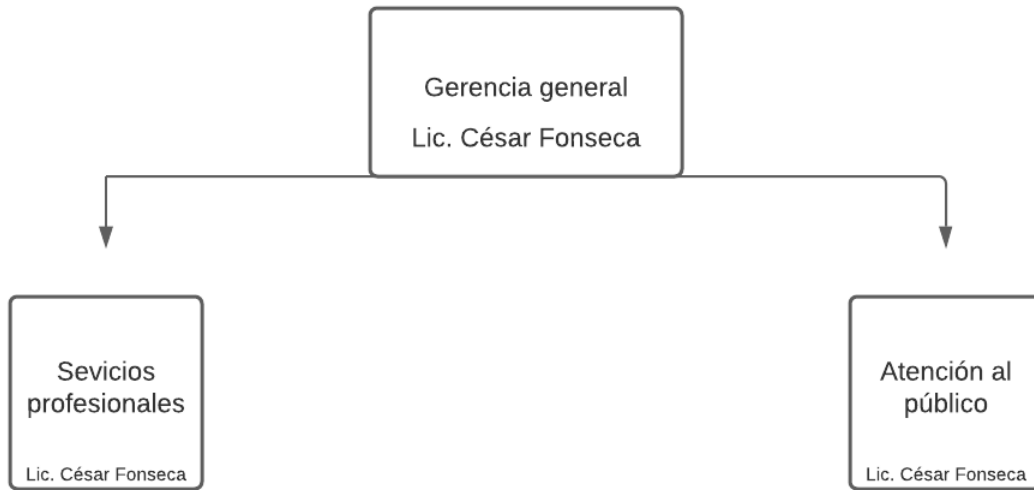


Figura 1. Organigrama del bufete

Fuente: elaboración propia

1.1.1.6 Historia de la organización

El bufete Fonseca Sequeira y asociados nace en el año 2000, y es fundado por el abogado César Fonseca, quien a raíz de trabajar en el Poder Judicial como auxiliar en la parte penal, le nace la idea de estudiar leyes, y es precisamente en el mes de agosto del año 2000 cuando culmina todos los estudios de leyes y se incorpora en ese mismo mes al Colegio de Abogados de Costa Rica. En ese momento decide establecer formalmente un bufete en Puntarenas Centro. Así

es como comienza toda una historia de 20 largos años de trayectoria profesional, con especialización en las ramas de derecho penal, familiar y laboral. En la actualidad, el bufete Fonseca Sequeira, además de la oficina en Puntarenas, cuenta con otra oficina en Jicaral.

1.1.1.7 Tendencia en el mercado

A pesar de que el sector jurídico es conocido por ser un sector muy tradicional y ser usualmente hermético a los cambios que ha estado experimentando la sociedad, la digitalización, que ha transformado sectores enteros como el sector financiero o el de comercio, ha transformado también el sector jurídico, e incluso esta transformación ha dado origen a un concepto llamado “legaltech”. Corrales et al. (2019) menciona que LegalTech, también conocido como “Tecnología legal,” “Informática legal,” o “Leyes y TI”, está disruptiendo la forma en que los abogados practican la abogacía. Se refiere a la adopción de tecnología innovadora para mejorar los servicios legales. En otras palabras, el hardware y el software usados en la profesión legal. LegalTech es, por lo tanto, la respuesta al crecimiento exponencial de nuevas tecnologías lo cual está influenciando directamente el sector legal, conocido como uno de los mercados más conservadores.

Algunos ejemplos de tecnologías disponibles para el sector jurídico en Costa Rica son:

- Master Lex:

Se presenta como una empresa líder en el mercado centroamericano con más de dos décadas de experiencia en el desarrollo de eficientes herramientas para la sistematización, administración y publicación digital de información de relevancia jurídica y notarial. Master Lex posee el módulo administrador que se encarga del control de expedientes y facturación y estandariza, automatiza e integra procesos. Este módulo está disponible de forma web.

- GlobaLex:

GlobaLex es un software jurídico que se encarga de realizar cálculos, mantener actualizada los órganos de jurisprudencia, tener las leyes disponibles al alcance del cliente mediante un buscador, tiene un apartado para notariado y un asistente virtual. También posee un módulo llamado “GlobaCorp”, que tiene las funciones de seguimiento de clientes, asuntos

y expedientes; recordatorios, alertas y seguimientos; reportes y gráficos gerenciales, control de honorarios y gastos y facturación digital.

1.1.2 Justificación del proyecto

La decisión de realizar este proyecto se da con la finalidad de poder tener un registro y una gestión digital de cada expediente y de cada documento relacionado a los expedientes.

Casillas et al. (2017) mencionan que “Como en todos los oficios de la sociedad, está ocurriendo una intensa transformación de la labor del abogado, impulsada por el uso de las TIC” (p.46). Debido a la importancia que tiene la información que maneja el bufete Fonseca Sequeira y, la necesidad de informatización para mantenerse en las tendencias actuales del mercado de la abogacía y que están ocurriendo en todos los oficios, como recalca Casillas et al. (2017), que permiten acelerar procesos y mejorar la calidad del servicio a los clientes se ha visto la necesidad de implementar un sistema de gestión de expedientes jurídicos.

Este permitirá que el trabajo diario sea optimizado, y se simplifique la gestión de expedientes, así como las tareas que se realizan entre las cuales están: el mantenimiento de los expedientes jurídicos, el manejo de una agenda de citas en las que se establezcan las citas con los clientes, el manejo de una agenda que permite la creación de alarmas pertinentes al vencimiento del plazo para realizar una tarea específica relacionada con algún caso, el manejo de la información de cada cliente, y la separación de cada caso y expediente con su correspondiente abogado encargado. También permitirá al bufete acercarse cada vez más al cumplimiento de los objetivos, misión y visión que poseen como organización.

1.2 Definición del problema

1.2.1 Problemática

En este bufete actualmente el manejo de la documentación, de los expedientes y de las citas se realiza de forma física, por lo que en las consultas la búsqueda de la información y de los expedientes pertinentes al caso toma más tiempo, lo que entorpece y retrasa el trabajo del abogado.

El manejo de citas se realiza también en físico, estas citas se agendan en un libro con la información del cliente, el día y la hora de la cita, y es gestionada por él mismo, lo cual no le permite al abogado tener un control sencillo de estas citas, ni le permite enfocarse en su totalidad en tareas propias de su profesión.

El bufete Fonseca Sequeira tiene otra oficina aparte de la principal ubicada en Puntarenas Centro, por lo que debe cargar con los expedientes y los documentos necesarios de una oficina a otra, esto hace que el peligro de pérdida de los documentos sea mayor. De igual forma, al tener todos los registros en físico, el control de fechas de citas, de entrega de documentos y de seguimientos de casos se torna tedioso para el abogado, lo que puede llevar a la ineficiencia en su trabajo.

1.2.1.1 Diagrama de Ishikawa

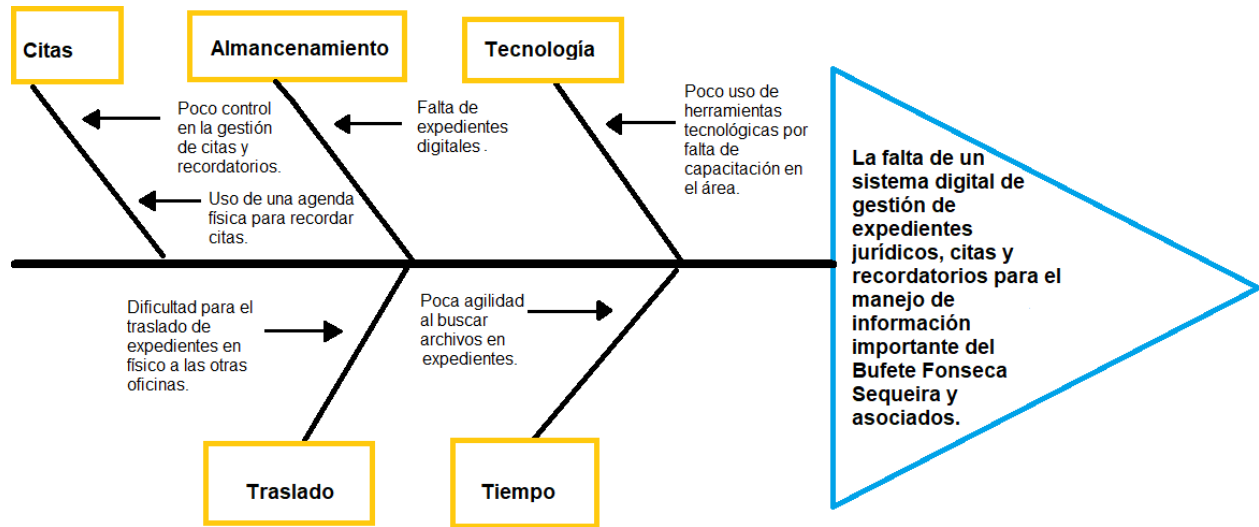


Figura 2. Diagrama de Ishikawa.

Fuente: elaboración propia

1.2.2 Problema general

¿Cómo automatizar el proceso de gestión de expedientes jurídicos, citas y recordatorios, y así optimizar el manejo de la información dentro del Bufete Fonseca Sequeira y asociados?

1.2.3 Problemas específicos

¿Qué requerimientos se deben identificar para automatizar el proceso de gestión de expedientes jurídicos, citas y recordatorios del bufete?

¿Qué información se debe considerar y analizar para mejorar el proceso de gestión de expedientes jurídicos, citas y recordatorios del bufete?

¿Cómo se debe desarrollar el sistema para que ayude en tiempo y dinero a la gestión de expedientes jurídicos, citas y recordatorios del bufete?

1.3 Objetivos

1.3.1 Objetivo general

Desarrollar e implementar un sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados, para la optimización del manejo de la información dentro del mismo.

1.3.2 Objetivos específicos

- Recolectar y analizar la información que permita determinar los requerimientos funcionales y no funcionales que se deben aplicar en el sistema.
- Diseñar los módulos que requiere el sistema web así como su base de datos, para guardar la información de los expedientes jurídicos, citas y recordatorios del bufete.
- Desarrollar los módulos del sistema junto con sus respectivos mantenimientos y reportes de los datos ingresados en el sistema.
- Implementar el sistema web desarrollado para la gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados.

1.4 Alcance y limitaciones

1.4.1 Alcance del proyecto

Este proyecto consta del desarrollo de un sistema web que permita realizar la gestión de expedientes jurídicos, clientes, citas y recordatorios para el bufete Fonseca Sequeira y asociados.

Para el primer entregable, se levantará una lista de requerimientos del sistema, donde permita conocer que debe realizar el mismo, dicha lista es proporcionada por la persona interesada en el desarrollo del sistema, en este caso el abogado del bufete.

En el segundo entregable se desarrollarán los casos de usos donde se explique el funcionamiento del sistema y los actores que lo utilizarán, esto con el propósito de apreciar una forma más fácil para entender que realizará el mismo, además se tendrá que desarrollar una base de datos con sus respectivas tablas, para el registro de todos los datos en el sistema, ya que el bufete no cuenta con una ya hecha.

El tercer entregable comprenderá el desarrollo del sistema. Este entregable consta de los siguientes módulos:

- El control de clientes, en el que se podrá visualizar la información relacionada al cliente como la información de contacto, los casos y expedientes, resumen de actividad, en el que se detallan las citas y consultas que ha tenido, y los correos que se han enviado a este cliente. De igual forma tendrá un buscador para buscar al cliente deseado. Permite crear un nuevo perfil de un cliente y abrir un expediente o caso para trabajar en él de inmediato.
- La gestión de expedientes, en ella se crearán los expedientes y los casos en los que el abogado trabajará, permitirá subir anexos como fotografías, videos o documentos, de igual forma podrán ser modificados. Tendrá una barra de búsqueda para que pueda buscar el expediente o caso que necesite consultar o editar.
- La gestión de la agenda, que permitirá agendar citas con los clientes, una vez agendada la cita el sistema le enviará un correo electrónico al cliente con la confirmación y los datos de la cita. El sistema tendrá un tablero o una agenda donde muestre las citas que el abogado tiene en esa semana; también, esta agenda

permitirá al abogado crear alarmas pertinentes al vencimiento del plazo para realizar una tarea específica relacionada con algún caso o alguna cita.

- El módulo de reportes, que deberá brindar la opción de generar un reporte relacionado a alguno de los parámetros que se brindarán como opción. Estas opciones serán: generar reportes de un cliente, de un expediente, de correos enviados a un cliente y de notificaciones relacionadas a un expediente.
- El módulo de configuraciones, el cual consiste en un apartado para agregar, modificar y eliminar tipos de procesos que se mostrarán en la lista de despachos judiciales que aparecerá en el módulo de expedientes, también tendrá un apartado de gestión de usuarios, para crear, modificar y eliminar usuarios con su respectivo rol dentro del sistema.

El cuarto entregable es la implementación de la aplicación web para la gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados. Este entregable tiene como resultado una aplicación completamente funcional al que el abogado pueda ingresar a través de una computadora con acceso a internet.

1.4.2 Limitaciones del proyecto

- El Lic. Cesar Fonseca Sequeira, dueño del bufete, no cuenta con ningún tipo de servidor que pueda proveer para el hospedaje de la base de datos y el hospedaje de la aplicación web del proyecto.
- Se usará un servicio de hospedaje web para el hospedaje de la base de datos y de la aplicación web.
- La aplicación web se desarrollará utilizando Microsoft Visual Studio 2019, empleando la edición y licencia de Community Edition.
- Una vez implementada la aplicación web se proveerá el servicio de web hosting por 3 meses.
- Tras la finalización de los 3 meses quedará a criterio del dueño del bufete continuar el pago del servicio para la continuidad del mismo.

1.5 Cronograma del proyecto

Actividad/ Semana	Fecha de inicio	Fecha de finalización
Capítulo I: Planteamiento del tema	03/06/2021	25/06/2021
Antecedentes y justificación del proyecto	03/06/2021	07/06/2021
Definición del problema	08/06/2021	14/06/2021
Objetivos	15/06/2021	19/06/2021
Alcances y limitaciones	20/06/2021	25/06/2021
Capítulo II: Marco teórico	30/06/2021	04/08/2021
Consulta de fuentes	30/06/2021	14/07/2021
Desarrollo de conceptos	15/07/2021	04/08/2021
Capítulo III: Marco metodológico	05/08/2021	26/08/2021
Tipo y enfoque de la investigación	05/08/2021	08/08/2021
Fuentes y sujetos de investigación	09/08/2021	12/08/2021
Técnicas y herramientas de recolección de datos	13/08/2021	16/08/2021
Variables de la investigación	17/08/2021	21/08/2021
Diseño de la investigación	22/08/2021	24/08/2021
Matriz de coherencia	25/08/2021	26/08/2021
Capítulo IV: Diagnóstico	27/08/2021	17/09/2021
Descripción de la situación actual	27/08/2021	07/09/2021
Determinación de brechas y conclusiones	08/09/2021	17/09/2021
Capítulo V: Propuesta del proyecto	20/09/2021	23/11/2021
Análisis del proyecto	20/09/2021	27/09/2021
Diseño del proyecto	28/09/2021	12/10/2021
Desarrollo del proyecto	13/10/2021	15/11/2021
Implementación del proyecto	16/11/2021	23/11/2021
Capítulo VI: Conclusiones y recomendaciones	24/11/2021	30/11/2021
Conclusiones	24/11/2021	26/11/2021
Recomendaciones	28/11/2021	30/11/2021
Cierre	01/12/2021	10/12/2021
Revisión del tutor	01/12/2021	10/12/2021
Cambios respectivos	01/12/2021	10/12/2021

Figura 3. Cronograma del proyecto.

Fuente: elaboración propia

CAPÍTULO II: MARCO TEÓRICO

Introducción capítulo II

En este capítulo se abarcan los distintos conceptos, términos e información académica relacionados con el proyecto a desarrollar. Para mayor comprensión y orden la información se dividirá en tres grupos: conceptos e información relacionada al área legal y jurídica, aquí se encontrará todos aquellos términos legales necesarios para comprender el proyecto; información relacionada al ámbito informático, que contendrá todos los conceptos relacionados a la aplicación web a realizar, este mismo se subdivide en: sistema de gestión, internet, análisis del proyecto, diseño del proyecto, desarrollo del proyecto e implementación del proyecto; e información relacionada con la tecnología legal, que es la aplicación de la informática al área legal.

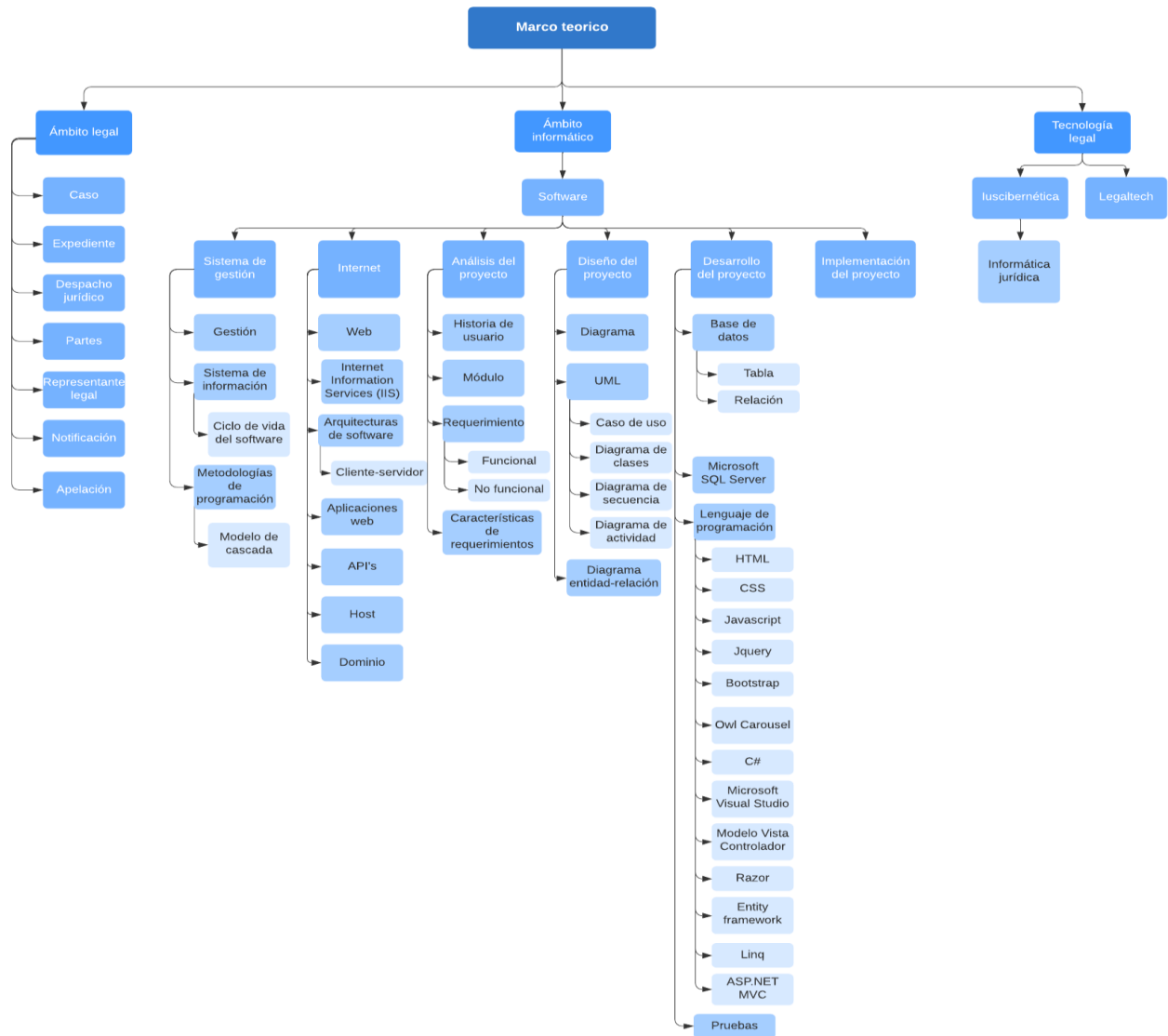


Figura 4. Esquema marco teórico

. Fuente: elaboración propia

2.1 Ámbito legal

En este apartado se abarcarán los términos legales que serán necesarios para comprender de una mejor manera el proyecto a desarrollarse.

2.1.1 Caso

De acuerdo con el diccionario usual del Poder Judicial de Costa Rica, un caso es un asunto tratado, o un asunto que se dirime en un juzgado o tribunal judicial. ("Diccionario Usual del Poder Judicial - caso", 2021)

2.1.2 Expediente

De acuerdo con Torres & Cuevas (2008) un expediente se puede definir como un negocio o asunto que se ventila ante los tribunales, a instancia de parte interesada, o de oficio, pero sin existir un juicio contradictorio. En tal sentido, pueden calificarse de expedientes todos los actos de jurisdicción voluntaria. También se le define como conjunto de papeles, documentos y otras pruebas o antecedentes, que pertenecen a un asunto o negocio, relacionado con oficinas públicas o privadas.

2.1.3 Despacho jurídico

Despacho es definido como la habitación o aposento para atender los negocios, se aplica especialmente al bufete de los abogados, y judicial se refiere a perteneciente al juicio, o atinente a la administración de la justicia (Torres & Cuevas, 2008). Por lo que para este proyecto se entenderá por despacho judicial el departamento de administración de la justicia encargado del caso.

2.1.4 Partes

Se entenderá como parte a cada una de las personas que por voluntad, intereses o determinación legal interviene en un acto jurídico plural. Contratante, Litigante; sea demandante, o actor, sea demandado o reo; y también en el proceso criminal, el querellante o el acusado. (p.282).

2.1.5 Representante legal

Un representante se define como aquel que representa, y el término legal se refiere aquello mandado por la ley o contenido en ella, así como conforme a la ley, por lo que se entiende que

un representante legal es alguien que obra jurídicamente en nombre de otro y que es reconocido por la ley. (p.220, 331).

2.1.6 Notificación

El diccionario usual del Poder Judicial de Costa Rica tiene varias descripciones para este término.

1. Comunicación oficial y formal de una resolución administrativa o judicial.
2. Acto jurídico procesal mediante el que se comunica a alguien el contenido de una resolución determinada.
3. Documento, material o informático, que contiene la comunicación de una resolución, oficial y formal, de una autoridad administrativa o judicial. (“Diccionario Usual del Poder Judicial – notificación”, 2021)

Por lo tanto, con base en esto, una notificación será considerada como un documento, físico o informático, oficial y formal que comunica el contenido de una resolución judicial.

2.1.7 Apelación

Una apelación consiste en una petición incidental previa en el que el apelante o el apelado solicitan del tribunal superior que extienda o no al efecto suspensivo la apelación admitida en primera instancia; este recurso es el que permite que un tribunal superior al que dictó la providencia recurrida la revise, pudiendo confirmarla, modificarla o rechazarla en todo o en parte. (Casado, 2016)

2.2 Ámbito informático

Esta sección consistirá de todos los términos, conceptos e información relacionados al área de la informática y que serán fundamentales para la realización del sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados.

2.2.1 Software

El software es: 1) instrucciones (programas de cómputo) que cuando se ejecutan proporcionan las características, función y desempeño buscados; 2) estructuras de datos que permiten que los programas manipulen en forma adecuada la información, y 3) información descriptiva tanto en papel como en formas virtuales que describen la operación y uso de los programas. (Pressman, 2021).

Ahora, el software también tiene características que lo identifican:

1. El software se desarrolla o modifica con intelecto; no se manufactura en el sentido clásico: esto quiere decir, que, aunque tanto el software como el hardware (las partes físicas y tangibles de la computadora) son construidos por personas y su alta calidad se obtiene por medio de un buen diseño, no pueden compararse, pues el software al ser un elemento de un sistema lógico no puede tratarse como producto de un proyecto de manufactura, como sí sucede con el hardware. (p.4).
2. El software no se “desgasta”: esto significa que a diferencia del hardware, el software no se ve afectado por factores ambientales tales como acumulación de suciedad, vibración, altas temperaturas, etc. Sin embargo, Pressman resalta que aunque el software no se desgasta, sí se deteriora, esto debido a los cambios que se realizan en él con el paso del tiempo y que pueden introducir fallos que causan que el software se deteriore. (p.4).
3. Aunque la industria se mueve hacia la construcción basada en componentes, la mayor parte del software se construye para un uso individualizado: A medida que evoluciona una disciplina de ingeniería, se crea un conjunto de componentes estandarizados para el diseño. Los tornillos estándar y los circuitos integrados preconstruidos son sólo dos de los miles de componentes estándar que utilizan los ingenieros mecánicos y eléctricos conforme diseñan nuevos sistemas. Los componentes reutilizables han sido creados para que el ingeniero pueda concentrarse en los elementos verdaderamente innovadores de un diseño; es decir, en las partes de éste que representan algo nuevo. En el mundo del hardware, volver a usar componentes es una parte natural del proceso de ingeniería. En el del software, es algo que apenas ha empezado a hacerse a gran escala. Un componente de software debe diseñarse e implementarse de modo que pueda volverse a usar en muchos programas diferentes. Los modernos componentes reutilizables incorporan tanto los datos

como el procesamiento que se les aplica, lo que permite que el ingeniero de software cree nuevas aplicaciones a partir de partes susceptibles de volverse a usar. Por ejemplo, las actuales interfaces interactivas de usuario se construyen con componentes reutilizables que permiten la creación de ventanas gráficas, menús desplegable y una amplia variedad de mecanismos de interacción. Las estructuras de datos y el detalle de procesamiento que se requieren para construir la interfaz están contenidos en una librería de componentes reusables para tal fin (p.5-6).

Existen siete tipos de software de computadoras:

1. Software de sistemas: son programas escritos para dar servicio a otros programas, como los editores, compiladores y componentes de sistemas operativos.
2. Software de aplicaciones: programas aislados que resuelven una necesidad específica de negocios. Estos procesan datos comerciales o técnicos que facilitan la toma de decisiones administrativas, además de controlar funciones de negocios en tiempo real como el procesamiento de transacciones en punto de venta o el control de procesos de manufactura en tiempo real.
3. Software de ingeniería y ciencias: se han caracterizado por algoritmos “devoradores de números”; sin embargo, las aplicaciones modernas han comenzado a hacer en tiempo real diseños asistidos por computadora y simulaciones de sistemas, y hasta han adoptado características del software de sistemas.
4. Software incrustado: reside dentro de un producto o sistema y se usa para implementar y controlar características y funciones para el usuario final y para el sistema en sí, como el control del tablero de un horno de microondas o funciones digitales en un auto como el control del combustible.
5. Software de línea de productos: es diseñado para proporcionar una capacidad específica para uso de muchos consumidores diferentes. Puede enfocarse en un mercado específico, como por ejemplo un control de inventario de productos o puede enfocarse en un mercado masivo de consumidores, como un procesador de texto, hojas de cálculo, etc.
6. Aplicaciones web: llamadas “webapps”, esta categoría de software centrado en redes agrupa una amplia gama de aplicaciones. En su forma más sencilla, las webapps son poco más que un conjunto de archivos de hipertexto vinculados que presentan

información con uso de texto y gráficas limitadas, sin embargo, con el paso del tiempo han evolucionado y pueden integrarse con bases de datos corporativas y aplicaciones de negocios. De este tipo de aplicaciones se hablará más adelante a más detalle.

7. Software de inteligencia artificial: hace uso de algoritmos no numéricos para resolver problemas complejos que no son fáciles de tratar computacionalmente o con el análisis directo. Las aplicaciones en esta área incluyen robótica, sistemas expertos, reconocimiento de patrones (imagen y voz), redes neurales artificiales, demostración de teoremas y juegos.

2.2.1.1 Sistema de gestión

El proyecto a desarrollarse consiste en un sistema de gestión de expedientes, por lo que es necesario repasar los conceptos de gestión y sistemas de información, así como el ciclo de desarrollo de un proyecto de software y el modelo de desarrollo que se usará para poder comprender a mayor amplitud la naturaleza de este proyecto.

2.2.1.1.1 Gestión

En el estudio de Mora-Pisco, Duran-Vasco y Zambrano-Loor (2016) se habla ampliamente acerca de este término, la palabra gestión proviene del Latín *gestio*. Este término hace la referencia a la administración de recursos, sea dentro de una institución estatal o privada, para alcanzar los objetivos propuestos por la misma. Para ello uno o más individuos dirigen los proyectos laborales de otras personas para poder mejorar los resultados, que de otra manera no podrían ser obtenidos. (Concepto de Gestión. 2015)

Según se ha citado, gestión es la acción y el efecto de gestionar y administrar. De una forma más específica, una gestión es una diligencia, entendida como un trámite necesario para conseguir algo o resolver un asunto, habitualmente de carácter administrativo o que conlleva documentación, otra referencia; es también un conjunto de acciones u operaciones relacionadas con la administración y dirección de una organización. (Concepto de Gestión. 2015).

2.2.1.1.2 Sistema de información

De acuerdo con Laudon & Laudon (2016), un conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar los procesos de toma de decisiones y de control en una organización. Además de apoyar la toma de decisiones, la coordinación y el control, los sistemas de información también pueden ayudar a los gerentes y trabajadores del conocimiento a analizar problemas, visualizar temas complejos y crear nuevos productos.

Los sistemas de información contienen información sobre personas, lugares y cosas importantes dentro de la organización, o en el entorno que la rodea. Por información nos referimos a los datos que se han modelado en una forma significativa y útil para los seres humanos. Por el contrario, los datos son flujos de elementos en bruto que representan los eventos que ocurren en las organizaciones o en el entorno físico antes de ordenarlos e interpretarlos de forma que las personas los puedan comprender y usar.

Hay tres actividades en un sistema de información que producen los datos necesarios para que las organizaciones tomen decisiones, controlen las operaciones, analicen problemas y creen nuevos productos o servicios. Estas actividades son: entrada, procesamiento y salida. La entrada captura o recolecta los datos en bruto desde dentro de la organización o a través de su entorno externo. El procesamiento convierte esta entrada en bruto en un formato significativo. La salida transfiere la información procesada a las personas que harán uso de ella, o a las actividades para las que se utilizará. (p.16-18).

2.2.1.1.2.1 Ciclo de vida del software

El ciclo de vida del software abarca el proceso de desarrollo y el mantenimiento durante su explotación. Las fases del ciclo de vida del software son: análisis, diseño, codificación, pruebas, integración y mantenimiento. (Palomo & Gil, 2020).

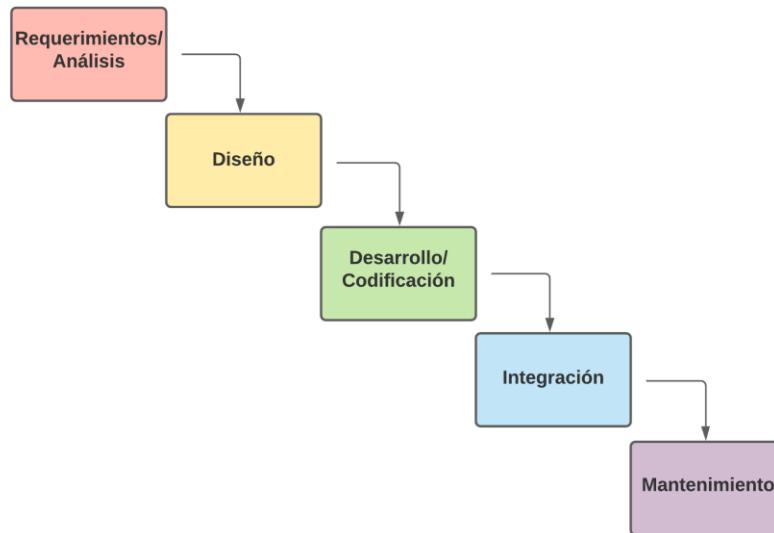


Figura 5. Fases del ciclo de vida de un proyecto.

Fuente: elaboración propia.

1. **Análisis:** en esta fase se analizan las necesidades que tienen los usuarios del futuro sistema de software y que deben ser satisfechos mediante el funcionamiento del mismo. El cliente que realiza el encargo expone sus necesidades, requisitos que debe cumplir el software y la empresa que lo va a realizar los recoge y analiza. De acuerdo con estas especificaciones, la empresa genera una especificación precisa del sistema a desarrollar.
2. **Diseño:** esta fase consiste en elaborar un esquema o diseño donde se contemplen los elementos necesarios para que el sistema funcione según lo especificado en el análisis. En esta fase no solo se diseña el sistema para su funcionamiento también se establece la organización del sistema para su construcción. El resultado de la fase de diseño suele ser un documento de carácter gráfico, donde se presentan todos los elementos componentes del sistema y la organización pormenorizada de cada uno de ellos.
3. **Codificación:** en la fase de codificación se produce materialmente lo que va a hacer funcionar el sistema de software. Se construirán por separado cada uno de los elementos que se han definido en la fase de diseño, utilizando para ello las

herramientas pertinentes: lenguajes de programación, sistemas de bases de datos, sistemas de información, etc.

4. Integración: Después de construir todos los componentes se procede a unificarlos con el fin de armar el sistema completo. En esta fase se deben realizar múltiples pruebas para asegurarse de que funciona correctamente.
5. Explotación: Esta fase no forma parte del ciclo de desarrollo de un producto de software, aunque sí influye en el resto de fases. Esta fase comprende el periodo de funcionamiento de la aplicación.
6. Mantenimiento: Esta sucede durante la fase de explotación. En la fase de explotación pueden encontrarse errores que no se vieron en la fase de desarrollo, o se pueden querer implementar mejoras. Ante estas situaciones el sistema debe evolucionar para responder a nuevas demandas. Esta evolución se desarrolla en la fase de mantenimiento. (p.34-35).

2.2.1.1.3 Metodologías de programación

La metodología de programación o de desarrollo de software es un enfoque estructurado para el desarrollo de software que incluye modelos de sistemas, notaciones, reglas, sugerencias de diseño y guías de procesos. (Zumba Gamboa, 2018).

2.2.1.1.3.1 Modelo de cascada

Los modelos de desarrollo de software fueron hechos para ordenar el proceso de desarrollo, existen varios tipos de modelos de desarrollo de software como el modelo en espiral, el modelo en fuente y el más tradicional, el modelo en cascada. Este modelo es el más sencillo, dado que las fases se desarrollan en forma secuencial. (Noriega, Lainez, & Durango, 2017).

Este modelo comienza por la especificación de los requerimientos por parte del cliente, y avanza a través de planeación, modelado, construcción y despliegue, para concluir con el apoyo del software terminado. Es el paradigma más antiguo de la ingeniería de software, sin embargo tiene algunos problemas o desventajas, como las siguientes:

1. Un proyecto real rara vez seguirá el flujo secuencial que propone el modelo de cascada.

2. Existe la posibilidad de que el cliente tenga problemas para mencionar de forma explícita los requerimientos, el modelo de cascada necesita que esto se haga y tiene dificultades para aceptar esta incertidumbre inicial.
3. Se podrá ver una versión funcional del programa hasta ya avanzado el proyecto, por lo que el cliente deberá ser paciente. (Pressman, 2021).

2.2.1.2 Internet

Según Tsatsou (2014), el término técnico de internet es una red de redes de computadoras que trabajan basadas en el protocolo TCP/IP. También se define como una red global de comunicaciones, descentralizada mediada por la conjunción de computadoras y telecomunicaciones.

2.2.1.2.1 Web

La Web (originalmente llamada World Wide Web, www en direcciones de sitios), es una de las formas en las que la información puede ser compartida por el internet. Es único porque permite vincular documentos entre sí mediante enlaces de hipertexto, y así, formando una enorme “red” de información conectada. La Web usa un protocolo llamado HTTP (HyperText Transfer Protocol). En otras palabras, la Web es un subconjunto del internet, es solo una de las muchas formas en que la información puede ser transferida entre computadoras conectadas. (Robbins, 2012).

2.2.1.2.2 Internet Information Services (IIS)

IIS es el acrónimo de Internet Information Services o Internet Information Server, es el servidor web de Microsoft, consta de una serie de servicios para que los ordenadores que funcionan con Windows se puedan convertir en un servidor de Internet o Intranet. Su antecedente más remoto fue el servidor web desarrollado en el Centro Académico Europeo de Microsoft Windows NT (EMWAC). IIS es considerado como parte integral de los sistemas operativos de Microsoft, por lo que el lanzamiento de las versiones está emparejado con una versión del sistema operativo, su primera versión comercial fue la 1.0 lanzada como un

complemento gratuito con Windows NT 3.51 en 1995. Su versión actual es la 8.0 lanzada en septiembre de 2012 junto con Windows Server 2012 (Microsoft, 2012). Si bien el hecho de que IIS sea poco portable al estar amarrado a una versión específica de sistema operativo puede resultar una limitante, la verdad es que también le permite aprovechar al máximo las funcionalidades del sistema operativo y lograr un buen rendimiento. Así, la escalabilidad está limitada al cambio de hardware más potente pero siempre con Windows Server 2012. A partir de la versión 7.0, se rediseña completamente la arquitectura haciéndolo modular. Los módulos, llamados extensiones, pueden ser añadidos o removidos individualmente de acuerdo a las necesidades específicas de los usuarios, lo que mejora su escalabilidad y rendimiento. Existe un número creciente de extensiones, provenientes tanto de Microsoft como de terceros. (Sánchez Martínez, 2012).

2.2.1.2.3 Arquitecturas de software

De acuerdo con Peña y Rojas (2016), Actualmente no existe una definición única para el concepto de arquitectura de software, el término ha sido abordado por un gran número de autores,¹ no obstante, se reconoce como la definición más completa la dada por la IEEE Std 1471-2000: "La arquitectura de software es la organización fundamental de un sistema enmarcada en sus componentes, las relaciones entre ellos, y el ambiente, y los principios que orientan su diseño y evolución".

Al diseñar una arquitectura de software se crean y representan componentes que interactúan entre sí, con responsabilidades específicas y se organizan de forma tal que se logren los requerimientos establecidos. Se puede partir con patrones de soluciones probados que se conocen con el nombre de estilos arquitectónicos, patrones arquitectónicos y patrones de diseño.

Existen distintos tipos de estilos de arquitectura de software, sin embargo para este proyecto se enfocará solamente en el modelo Cliente/Servidor

2.2.1.2.3.1 Cliente – servidor

El modelo 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. Las aplicaciones Clientes realizan peticiones a una o varias aplicaciones Servidores, que deben encontrarse en ejecución para atender dichas demandas. El modelo Cliente/Servidor permite diversificar el trabajo que realiza cada aplicación, de forma que los Clientes no se sobrecarguen, cosa que ocurriría si ellos mismos desempeñan las funciones que le son proporcionadas de forma directa y transparente. En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema. Tanto el Cliente como el Servidor son entidades abstractas que pueden residir en la misma máquina o en máquinas diferentes.

Una disposición muy común son los sistemas multicapa en los que el servidor se descompone en diferentes programas que pueden ser ejecutados por diferentes computadoras aumentando así el grado de distribución del sistema. (Marini, 2012).

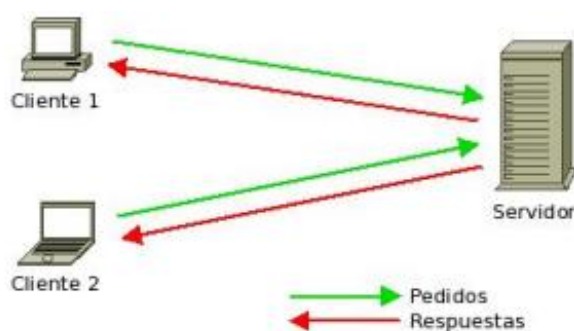


Figura 6. Modelo Cliente/Servidor.

Fuente: Marini (2012, p.1)

El modelo cliente/servidor puede variar en cantidad de capas, puede ser de 2 capas o puede ser multicapa.

1. Modelo cliente/servidor de 2 capas:

Uno de los objetivos de las aplicaciones de 2 capas es separar la lógica de acceso a los datos de lo que es la interfaz de usuario y trasladarla al servidor. Habitualmente, se implementan servicios como procedimientos almacenados en el sistema gestor de datos; con esto se pretende reducir la carga de los clientes y centralizar las

operaciones comunes de acceso a los datos. El Sistema Gestor de Datos también suele incorporar la funcionalidad necesaria para trabajar en entornos multiusuarios.

En este modelo intervienen únicamente dos entidades: El Cliente y El Servidor.

El papel de Cliente lo desempeña la aplicación final del usuario, que implementará todas las funciones correspondientes a la lógica de presentación, más algunas de las funciones relacionadas con la lógica del negocio, como pueden ser determinadas validaciones de datos y condiciones de recuperación.

El papel de Servidor lo desempeña el propio SGBD, el cual se ocupará de todas las funciones correspondientes a la lógica de datos, más las restantes funciones correspondientes a la lógica del negocio, mediante la codificación de Procedimientos Almacenados.

Este es el modelo C/S más sencillo y más utilizado habitualmente. En la mayor parte de los casos, el desarrollador de una aplicación de este tipo, desarrolla únicamente la aplicación Cliente y utiliza al propio motor de BD como aplicación servidora, de modo que no se codifica la aplicación Servidora propiamente dicha. (p.3, 4).

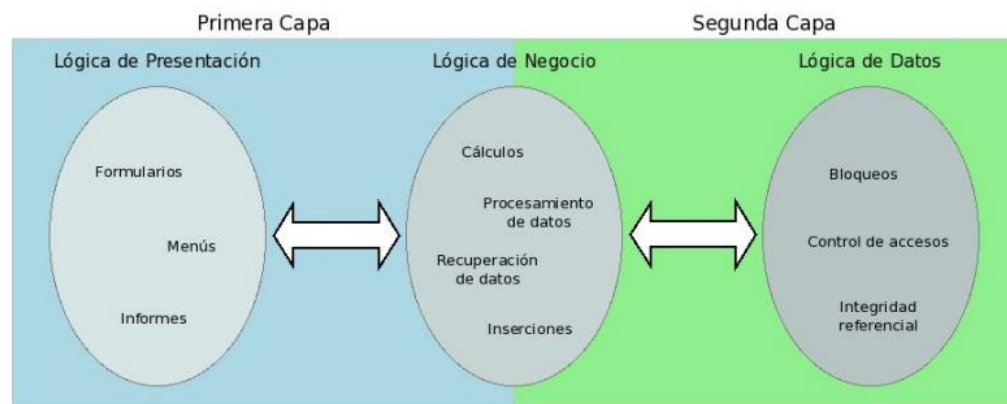


Figura 7. Modelo Cliente/Servidor de 2 capas.

Fuente: Marini (2012, p.5)

2. Modelo Cliente/Servidor multicapa:

La arquitectura cliente/servidor genérica tiene dos tipos de nodos en la red: clientes y servidores. Consecuentemente, estas arquitecturas genéricas se refieren a veces como arquitecturas de dos niveles o dos capas.

Algunas redes disponen de tres tipos de nodos:

- Clientes que interactúan con los usuarios finales.
- Servidores de aplicación que procesan los datos para los clientes.
- Servidores de la base de datos que almacenan los datos para los servidores de aplicación.

Esta configuración se llama una arquitectura de tres capas.

Este modelo aporta una flexibilidad adicional en la construcción de aplicaciones cuando éstas aumentan su complejidad. Influye tanto en el modelo de aplicación (lógicas de presentación, del negocio y de datos) como en la distribución de los servicios. El modelo conceptual de una aplicación establece sus definiciones, reglas y relaciones así como su estructura. Hay partes de la lógica que residen en el cliente, normalmente las que se refieren a la interfaz de usuario, mientras que las del negocio y de datos suelen residir en los servidores, que proporcionan los mecanismos necesarios para el trabajo en entornos multiusuarios. En este tipo de modelo se aplica íntegramente el modelo de servicios ya que, cada una de las capas se corresponde con cada una de las lógicas descritas.

Para llevar a cabo la implementación de un modelo como éste, se hace uso de los mismos recursos que en el modelo de dos capas. En la actualidad las técnicas y lenguajes de programación de servidores han avanzado de tal forma de permitir desarrollos modulares (por ejemplo mediante la arquitectura MVC, Model-Control-View), orientación a objetos y existen innumerables frameworks y librerías para simplificar la tarea de los desarrolladores de software.

Una de las características principales de este modelo reside en la desconexión total entre la lógica de presentación y la lógica de los datos. Las conexiones que se producen, se dan entre las lógicas de presentación y del negocio, y las lógicas del negocio y la de datos. Este modelo hace que la aplicación final sea completamente independiente del origen de los datos que procesa, tarea que pasa a ser competencia directa del componente especializado.

A pesar de esto, no es necesario que las distintas lógicas residan en máquinas diferentes; en la mayoría de los casos, es perfectamente compatible su

implementación en la misma máquina, si bien este diseño no es el más habitual. (p.5-6).

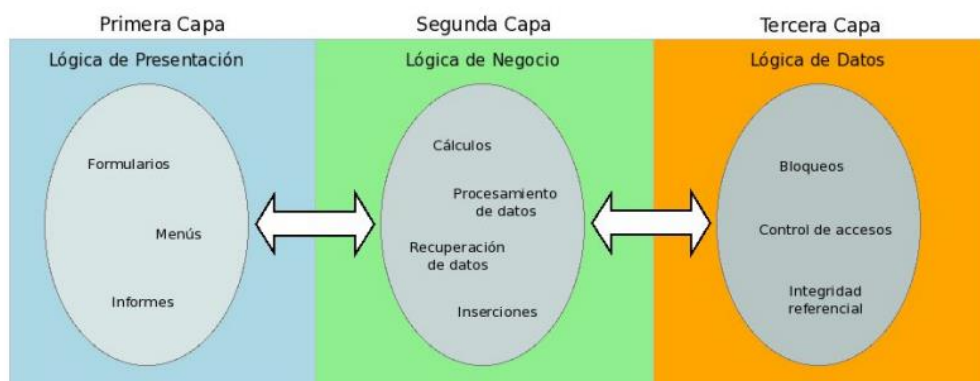


Figura 8. Modelo Cliente/Servidor multicapa

Fuente: Marini (2012, p.6).

2.2.1.2.4 Aplicaciones web

De las aplicaciones web o “webapps” ya se ha hablado anteriormente. Son aplicaciones que se conforman de un conjunto de archivos de hipertexto vinculados que presentan información y que, según han evolucionado pueden conectarse con bases de datos, aplicaciones de negocios, etc. (Pressman, 2021).

En los primeros días de la Red Mundial (entre 1990 y 1995), los sitios web consistían en poco más que un conjunto de archivos de hipertexto vinculados que presentaban la información con el empleo de texto y gráficas limitadas. Al pasar el tiempo, el aumento de HTML por medio de herramientas de desarrollo (XML, Java) permitió a los ingenieros de la web brindar capacidad de cómputo junto con contenido de información. Habían nacido los sistemas y aplicaciones basados en la web (denominó a éstas en forma colectiva como webapps). En la actualidad, las webapps se han convertido en herramientas sofisticadas de cómputo que no sólo proporcionan funciones aisladas al usuario final, sino que también se han integrado con bases de datos corporativas y aplicaciones de negocios.

Powell [Pow98] sugiere que los sistemas y aplicaciones basados en web “involucran una mezcla entre las publicaciones impresas y el desarrollo de software, entre la mercadotecnia y la

computación, entre las comunicaciones internas y las relaciones exteriores, y entre el arte y la tecnología”. La gran mayoría de webapps presenta los siguientes atributos:

1. Uso intensivo de redes

Una webapp reside en una red y debe atender las necesidades de una comunidad diversa de clientes. La red permite acceso y comunicación mundiales (por ejemplo, internet) o tiene acceso y comunicación limitados (por ejemplo, una intranet corporativa).

2. Concurrencia

A la webapp puede acceder un gran número de usuarios a la vez. En muchos casos, los patrones de uso entre los usuarios finales varían mucho.

3. Carga impredecible

El número de usuarios de la webapp cambia en varios órdenes de magnitud de un día a otro. El lunes tal vez la utilicen cien personas, el jueves quizá 10 000 usen el sistema.

4. Rendimiento

Si un usuario de la webapp debe esperar demasiado (para entrar, para el procesamiento por parte del servidor, para el formado y despliegue del lado del cliente), él o ella quizá decidan irse a otra parte.

5. Disponibilidad

Aunque no es razonable esperar una disponibilidad de 100%, es frecuente que los usuarios de webapps populares demanden acceso las 24 horas de los 365 días del año. Los usuarios en Australia o Asia quizá demanden acceso en horas en las que las aplicaciones internas de software tradicionales en Norteamérica no estén en línea por razones de mantenimiento.

6. Orientadas a los datos

La función principal de muchas webapp es el uso de hipermedios para presentar al usuario final contenido en forma de texto, gráficas, audio y video. Además, las webapps se utilizan en forma común para acceder a información que existe en bases de datos que no son

parte integral del ambiente basado en web (por ejemplo, comercio electrónico o aplicaciones financieras).

7. Contenido sensible

La calidad y naturaleza estética del contenido constituye un rasgo importante de la calidad de una webapp.

8. Evolución continua

A diferencia del software de aplicación convencional que evoluciona a lo largo de una serie de etapas planeadas y separadas cronológicamente, las aplicaciones web evolucionan en forma continua. No es raro que ciertas webapp (específicamente su contenido) se actualicen minuto a minuto o que su contenido se calcule en cada solicitud.

9. Inmediatez

Aunque la inmediatez —necesidad apremiante de que el software llegue con rapidez al mercado— es una característica en muchos dominios de aplicación, es frecuente que las webapps tengan plazos de algunos días o semanas para llegar al mercado.

10. Seguridad

Debido a que las webapps se encuentran disponibles con el acceso a una red, es difícil o imposible limitar la población de usuarios finales que pueden acceder a la aplicación. Con el fin de proteger el contenido sensible y brindar modos seguros de transmisión de los datos, deben implementarse medidas estrictas de seguridad a través de la infraestructura de apoyo de una webapp y dentro de la aplicación misma.

11. Estética

Parte innegable del atractivo de una webapp es su apariencia y percepción. Cuando se ha diseñado una aplicación para comercializar o vender productos o ideas, la estética tiene tanto que ver con el éxito como el diseño técnico. (p.9-10).

2.2.1.2.5 API

Definimos las APIs como interfaces legibles por máquinas que conectan múltiples aplicaciones, tienen el control de la interacción con la aplicación, y elimina la necesidad de conocer el funcionamiento interno de cómo la funcionalidad de una API es proporcionada. (Wulf & Blohm, 2020).

Sagdeo (2018) menciona que Una interfaz de programación de aplicaciones ("API") es la interfaz entre una aplicación y una biblioteca. Consiste esencialmente en dos partes: (1) su código de declaración y (2) su estructura, secuencia y organización. "Código de declaración" es el código que un desarrollador de aplicaciones (es decir, un desarrollador usando la API) necesita usar para invocar una funcionalidad específica en la biblioteca. La estructura, secuencia y organización ("SSO", Structure, Sequence and Organization) de la API es esencialmente la taxonomía bajo la cual se estructura el código declarante. La siguiente figura muestra la relación entre una aplicación, una API y una biblioteca. En un lado de la interfaz, una aplicación puede usar la API sin saber nada sobre cómo funciona la biblioteca subyacente (es decir, sobre su código de implementación), siempre que se ajusten al código de declaración de la API y al SSO. Por otro lado, un implementador de API puede implementar una biblioteca de la manera que crea conveniente, siempre que se ajuste al SSO de la API y al código de declaración. Y, aunque una API y una biblioteca correspondiente a menudo son creadas por la misma organización, en principio son obras completamente independientes.

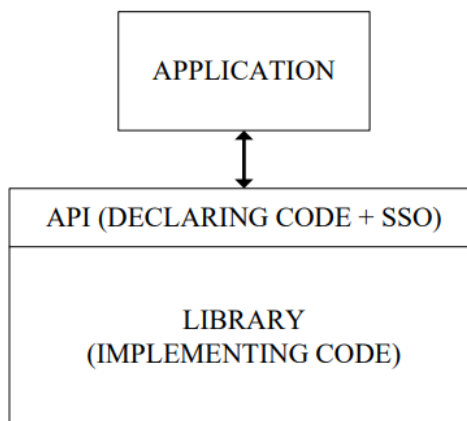


Figura 9. Relación entre una aplicación, una API y una librería

Fuente: Sagdeo (2018, p.240).

2.2.1.2.6 Host

Un servicio de hospedaje Web o web hosting mantiene un servidor Web grande o una serie de servidores, además de proporcionar espacio a los suscriptores que pagan una cuota por mantener sus sitios Web. (Laudon & Laudon, 2016).

2.2.1.2.7 Dominio

Talledo San Miguel (2015), menciona que de forma genérica, un dominio es un nombre legible y entendible por el ser humano que se utiliza para asociar a un grupo de dispositivos o equipos conectados a la red. Puede ser una identificación local y/o Internet.

Tiene como propósito realizar una traslación o traducción de la identificación de los equipos y/o redes, que se hace a través de valores numéricos, a un lenguaje entendible.

Los dominios se organizan como un árbol. Hay una raíz y esta se propaga a través del tronco creando ramas. Incluso habrá ramas que nacen de otras ramas.

Dependiendo de su localización en la jerarquía, un dominio puede ser de primer nivel (top-level), segundo nivel o tercer nivel. Se pueden añadir todos los niveles que queramos, pero no son habituales. Los que siguen son los dominios de primer nivel que veremos con frecuencia:

Dominio	Descripción
edu	Instituciones universitarias.
com	Organizaciones comerciales.
org	Organizaciones no comerciales. Las redes privadas UUCP suelen estar en este dominio.
net	Pasarelas y otras redes administrativas.
mil	El ejército norteamericano.
gov	El gobierno norteamericano.
uuc	Dominio para redes UUCP.
es	Dominio territorial de España. Cada país suele tener su propio dominio territorial. P.e. uk=United Kingdom.
info	Van dirigidos a páginas web informativas.
biz	Es un dominio de Internet genérico (TLD) previsto para ser usado en negocios.
name	Su uso está previsto para la representación de nombres personales, sobrenombres, nombres de usuario, pseudónimos u otros tipos de marcas de identificación de personas.
tel	Su uso está pensado para servicios de comunicación por Internet.

Figura 10. Dominios de primer nivel.

Fuente: Talledo San Miguel (2015, p.18)

Todos estos dominios raíz los podemos clasificar dentro de los dominios cualificados. Son dominios accesibles a través de Internet y para ser accesibles se deben registrar en empresas registradoras de dominios debidamente acreditadas. La organización que se encarga de velar por el buen funcionamiento del registro y dispone de la Base de Datos de todos los registros genéricos es InterNIC-Public Information Regarding Internet Domain Name Registration Services (www.internic.net).

Los dominios, para ser accesibles a través de Internet, deben ser dados de alta en los servicios DNS. De esta forma podrá "apuntar" nuestro nombre de dominio a una identificación del servidor donde esté "alojado".

2.2.1.3 Análisis del proyecto

En este apartado se estarán desarrollando los términos relacionados con la fase de análisis del ciclo de vida de desarrollo de un software. De acuerdo con Kendall & Kendall (2005), en su libro Análisis y Diseño de sistemas, la fase de análisis es la primera fase del ciclo, en esta fase se

identifican los problemas objetivos y oportunidades, se determinan los requerimientos con base en la información de los usuarios y se analizan las necesidades del sistema.

2.2.1.3.1 Historia de Usuario

Describen, en una o dos frases, una funcionalidad de software desde el punto de vista del usuario, con el lenguaje que éste emplearía. El foco está puesto en qué necesidades o problemas soluciona lo que se va a construir.

Las historias de usuario son una herramienta que agiliza la administración de requisitos, reduciendo la cantidad de documentos formales y tiempo necesarios. Forman parte de la fórmula de captura de funcionalidades definida en 2001 por Ron Jeffries de las tres Cs:

- Card: cada historia de usuario se reduce hasta hacerla fácil de memorizar y de sintetizar en una tarjeta o post-it. La tarjeta sirve como recordatorio y promesa de una conversación posterior.
- Conversation: el equipo de desarrollo y el propietario del producto añaden criterios de aceptación a cada historia poco antes de su implementación. Los cambios son bienvenidos en agilidad, por lo que no tiene sentido profundizar en estos detalles antes. La situación puede variar mucho desde el momento en el que se sintetiza la funcionalidad en la tarjeta hasta que se implementa.
- Confirmation: el propietario del producto o usuario de negocio confirma que el equipo de desarrollo ha entendido y recogido correctamente sus requisitos revisando los criterios de aceptación. A veces se pueden presentar transformados en escenarios de pruebas.

Algunas de las ventajas de las historias son:

- Proporcionan la documentación necesaria fomentando a la vez el debate.
- Fomentan la colaboración entre todos interesados y el equipo ágil.
- Se escriben en el lenguaje del usuario, manteniendo así una relación cercana con el cliente.
- Involucran y captan al cliente para el proceso y para el producto.
- Por su naturaleza son independientes.

- Facilitan la planificación e implementación.
- Son ideales para proyectos con requisitos volátiles o no muy claros.
- Fomentan aplazar los detalles no imprescindibles.
- Son pequeñas y por tanto fáciles trabajar.
- Permiten dividir los proyectos en pequeñas entregas.
- Permiten estimar fácilmente su esfuerzo de desarrollo.
- Funcionan para el desarrollo iterativo, ya que al ser pequeñas representan requisitos del modelo de negocio que pueden implementarse en poco tiempo (días o semanas).
- Necesitan poco mantenimiento. (Menzinsky, López, Palacio, Sobrino, Álvarez & Rivas, 2018).

2.2.1.3.2 Módulo

En programación, un módulo es un fragmento de un programa que se desarrolla de forma independiente del resto del programa. Esta independencia hace posible un mecanismo de compilación por separado que limita la complejidad del programa que se está desarrollando. Al compilarse el módulo por separado, la persona que lo desarrolla sólo debe preocuparse de él, prescindiendo en parte de cómo se utiliza este módulo dentro del programa. Quien escriba el resto del programa no debe preocuparse de los detalles del módulo sino sólo de cómo utilizarlo. (Gallardo Ruiz & García López, s.f.).

2.2.1.3.3 Requerimientos

Wieggers y Beatty (2013), en su libro *Software Requirements*, mencionan que el concepto más acertado para requerimiento es el siguiente: requerimientos son especificaciones de qué debería ser implementado. Son descripciones de cómo debería comportarse el sistema, la propiedad de un sistema o atributo. Pueden tener una restricción en el proceso de desarrollo del sistema. De acuerdo a este libro, esta definición reconoce los diversos tipos de información que colectivamente se denominan "los requisitos". Los requisitos abarcan tanto la visión del usuario del comportamiento del sistema externo como la visión del desarrollador de algunas características internas. Incluyen tanto el comportamiento del sistema en condiciones específicas

como aquellas propiedades que hacen que el sistema sea adecuado y quizás incluso agradable para su uso por parte de los operadores previstos.

2.2.1.3.3.1 Funcionales

Los requisitos funcionales especifican los comportamientos que exhibirá el producto en condiciones específicas. Describen lo que los desarrolladores deben implementar para permitir que los usuarios realicen sus tareas (requisitos del usuario), satisfaciendo así los requisitos comerciales. Esta alineación entre los tres niveles de requisitos es esencial para el éxito del proyecto. Los requisitos funcionales a menudo se escriben en forma de declaraciones en futuro: "El Pasajero podrá imprimir tarjetas de embarque para todos los segmentos de vuelo para los que se haya registrado" o "Si el perfil del Pasajero no indica una preferencia de asiento, el sistema de reservas asignará un asiento". (Wieggers y Beatty, 2013).

2.2.1.3.3.2 No funcionales

Los requisitos distintos a los funcionales pueden especificar no qué hace el sistema, sino qué tan bien hace esas cosas. Podrían describir características o propiedades importantes del sistema. Estos incluyen la disponibilidad, facilidad de uso, seguridad, rendimiento y muchas otras características del sistema. Algunas personas consideran que los requisitos no funcionales son sinónimo de atributos de calidad, pero eso es demasiado restrictivo. Por ejemplo, las restricciones de diseño e implementación también son requisitos no funcionales, al igual que los requisitos de interfaz externa.

Aún otros requisitos no funcionales abordan el entorno en el que opera el sistema, como la plataforma, la portabilidad, la compatibilidad y las limitaciones. Muchos productos también se ven afectados por requisitos de cumplimiento, normativos y de certificación. Puede haber requisitos de localización para los productos que deben tener en cuenta las culturas, idiomas, leyes, monedas, terminología, ortografía y otras características de los usuarios. (Wieggers y Beatty, 2013).

2.2.1.3.4 Características de requerimientos

De acuerdo con Arias Chaves (2011), es importante no perder de vista que un requerimiento debe ser:

- Especificado por escrito: Como todo contrato o acuerdo entre dos partes.
- Posible de probar o verificar. Si un requerimiento no se puede comprobar, entonces ¿cómo se sabe si se cumplió con él o no?
- Conciso: Un requerimiento es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.
- Completo: Un requerimiento está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.
- Consistente: Un requerimiento es consistente si no es contradictorio con otro requerimiento.
- No ambiguo: Un requerimiento no es ambiguo cuando tiene una sola interpretación. El lenguaje usado en su definición, no debe causar confusiones al lector.

2.2.1.4 Diseño del proyecto

De acuerdo con Kendall & Kendall (2005), en la fase de diseño del proyecto se toma la información recolectada durante la fase de análisis para realizar el diseño lógico del sistema de información, el diseño de bases de datos también está incluido en esta fase, así como las interfaces gráficas de usuario (GUIs, Grafical User Interfaces). En este apartado se abarcan conceptos relacionados a este proceso.

2.2.1.4.1 Diagrama

Representación gráfica de un conjunto de elementos del modelo y sus relaciones. En UML generalmente corresponde a un grafo conexo de nodos (elementos) y arcos (relaciones). (López & Ramiro, 2011).

2.2.1.4.2 UML

El Lenguaje de Modelado Unificado (UML) es “un lenguaje estándar para escribir diseños de software. El UML puede usarse para visualizar, especificar, construir y documentar los artefactos de un sistema de software intensivo” [Boo05]. En otras palabras, tal como los arquitectos de edificios crean planos para que los use una compañía constructora, los arquitectos de software crean diagramas de UML para ayudar a los desarrolladores de software a construir el software. Si usted entiende el vocabulario del UML (los elementos pictóricos de los diagramas y su significado) puede comprender y especificar con mucha más facilidad un sistema, y explicar su diseño a otros.

Grady Booch, Jim Rumbaugh e Ivar Jacobson desarrollaron el UML a mediados de los años noventa del siglo pasado con mucha realimentación de la comunidad de desarrollo de software. El UML fusionó algunas notaciones de modelado que competían entre sí y que se usaban en la industria del software en la época. En 1997, UML 1.0 se envió al Object Management Group, un consorcio sin fines de lucro involucrado en especificaciones de mantenimiento para su empleo en la industria de la computación. El UML 1.0 se revisó y dio como resultado la adopción del UML 1.1 ese mismo año. El estándar actual es UML 2.0 y ahora es un estándar ISO. Puesto que este estándar es tan nuevo, muchas antiguas referencias, como [Gam95], no usan notación de UML.

UML 2.0 proporciona 13 diferentes diagramas para su uso en modelado de software.

Se debe observar que existen muchas características opcionales en diagramas de UML. El UML ofrece dichas opciones (en ocasiones complejas) de modo que pueda expresar todos los aspectos importantes de un sistema. Al mismo tiempo, tiene la flexibilidad para suprimir aquellas partes del diagrama que no son relevantes para el aspecto que se va a modelar, con la finalidad de evitar confundir el diagrama con detalles irrelevantes. Por tanto, la omisión de una característica particular no significa que ésta se encuentre ausente; puede significar que la característica se suprimió. (Pressman, 2021).

De acuerdo con Kendall & Kendall (2005, p.664). Los seis diagramas de UML más utilizados son:

1. Diagrama de casos de uso, que describe cómo se usa el sistema.

2. Escenario de casos de uso (aunque técnicamente no es un diagrama), es una descripción verbal de las excepciones para el comportamiento principal descrito por el caso de uso principal.
3. Diagrama de actividades, ilustra el flujo general de actividades. Cada caso de uso podría crear un diagrama de actividades.
4. Diagrama de secuencias, muestran la secuencia de actividades y las relaciones de las clases. Cada caso de uso podría crear uno o más diagramas de secuencias. Una alternativa para un diagrama de secuencias es un diagrama de colaboración, el cual contiene la misma información en formato diferente.
5. Diagrama de clases, muestran las clases y las relaciones.
6. Diagramas de gráfico de estado, muestra las transiciones de estado. Cada clase podría crear un diagrama de gráfico de estado, el cual es útil para determinar los métodos de la clase.

Para este proyecto se abarcarán únicamente los diagramas UML de casos de uso, de clases, de actividad y de secuencia.

2.2.1.4.2.1 Diagrama de casos de uso

El UML está basado fundamentalmente en una técnica de análisis orientado a objetos conocida como modelado de casos de uso, en la cual la palabra “uso” se pronuncia como sustantivo en lugar de verbo. Un modelo de caso de uso describe lo que hace un sistema sin describir cómo lo hace; es decir, es un modelo lógico del sistema. El modelo de caso de uso refleja la vista del sistema desde la vista de un usuario fuera del sistema (es decir, los requerimientos del sistema). El UML se puede usar para analizar el modelo de caso de uso y para derivar objetos del sistema y sus interacciones entre sí y con los usuarios del sistema. Usando las técnicas de UML, analiza más a fondo los objetos y sus interacciones para derivar comportamiento del objeto, atributos y relaciones.

Un modelo de caso de uso divide la funcionalidad del sistema en comportamientos, servicios y respuestas (los casos de uso) que son significativos para los casos de uso.

Un diagrama de casos de uso contiene el actor y símbolos de caso de uso, junto con líneas de conexión. Los actores son parecidos a las entidades externas; existen fuera del sistema. El término “actor” se refiere a un papel particular de un usuario del sistema. Por ejemplo, un actor podría ser un empleado, pero también podría ser un cliente en el almacén de la compañía. Aunque quizás es la misma persona en el mundo real, se representa como dos símbolos diferentes en un diagrama de caso de uso, debido a que la persona interactúa con el sistema en diferentes papeles. El actor existe fuera del sistema e interactúa con éste de una forma específica. Un actor puede ser un humano, otro sistema o un dispositivo tal como un teclado, módem o conexión Web. Los actores pueden iniciar una instancia de un caso de uso. Un actor podría interactuar con uno o más casos de uso y viceversa.

Un diagrama de caso de uso contiene líneas de conexión llamadas relaciones. Las relaciones activas se denominan como relaciones de comportamiento y se emplean principalmente en estos diagramas. Hay cuatro tipos básicos de relaciones de comportamiento: comunica, incluye, extiende y generaliza.

1. **Comunica:** La relación de comportamiento comunica se usa para conectar a un actor con un caso de uso. La tarea del caso de uso es dar alguna clase de resultado que es benéfico para el actor del sistema.
2. **Incluye:** Esta relación describe la situación en que un caso de uso contiene un comportamiento que es común para más de un caso de uso. Es decir, el caso de uso común se incluye en otros casos de uso.
3. **Extiende:** La relación extiende describe la situación en la que un caso de uso posee el comportamiento que permite al nuevo caso de uso manejar una variación o excepción del caso de uso básico.
4. **Generaliza:** La relación generaliza implica que una cosa es más típica que otra. Esta declaración podría existir entre dos actores o dos casos de uso. (p.665- 667).

Símbolo	Relación
	Comunica
	Incluye
	Extiende
	Generaliza

Figura 11. Tipos de relación de diagramas de caso de uso.

Fuente: elaboración propia

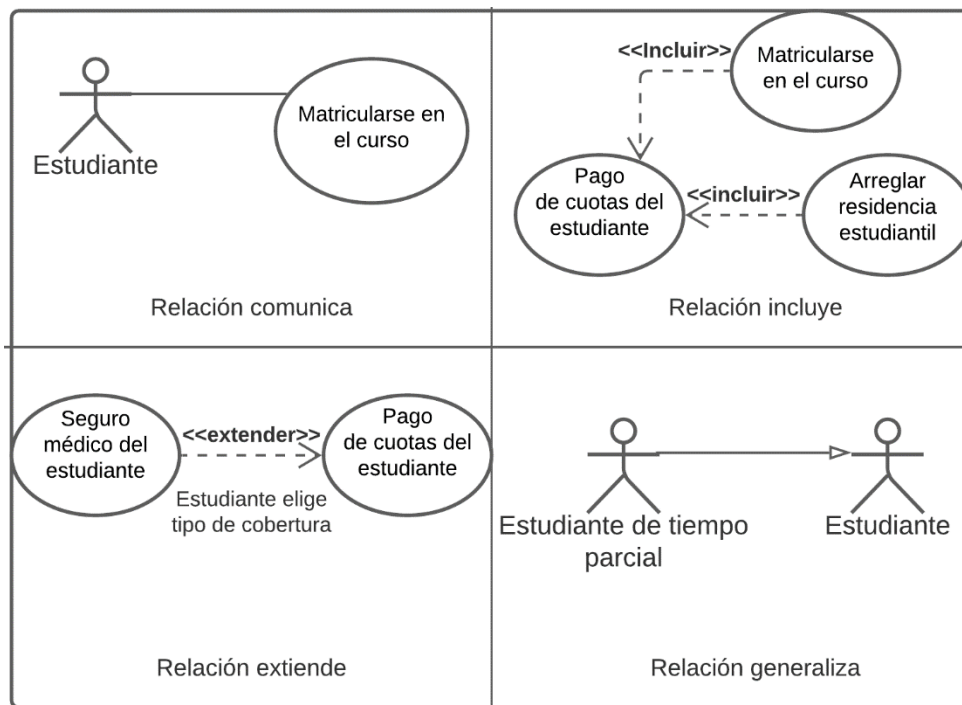


Figura 12. Cuatro tipos de relaciones además de flechas y líneas de comportamiento de UML usadas para representar las relaciones.

Fuente: adaptado de Kendall & Kendall (2005, p.667).

2.2.1.4.2.2 Diagrama de clases

Para modelar clases, incluidos sus atributos, operaciones, relaciones y asociaciones con otras clases, el UML proporciona un diagrama de clase, que aporta una visión estática o de estructura de un sistema, sin mostrar la naturaleza dinámica de las comunicaciones entre los objetos de las clases.

Los elementos principales de un diagrama de clase son cajas, que son los íconos utilizados para representar clases e interfaces. Cada caja se divide en partes horizontales. La parte superior contiene el nombre de la clase. La sección media menciona sus atributos. Un atributo es algo que un objeto de dicha clase conoce o puede proporcionar todo el tiempo. Por lo general, los atributos se implementan como campos de la clase, pero no necesitan serlo. Podrían ser valores que la clase puede calcular a partir de sus variables o valores instancia y que puede obtener de otros objetos de los cuales está compuesto. Por ejemplo, un objeto puede conocer siempre la hora actual y regresarla siempre que se le solicite. Por tanto, sería adecuado mencionar la hora actual como un atributo de dicha clase de objetos. Sin embargo, el objeto muy probablemente no tendría dicha hora almacenada en una de sus variables instancia, porque necesitaría actualizar de manera continua ese campo. En vez de ello, el objeto probablemente calcularía la hora actual (por ejemplo, a través de consulta con objetos de otras clases) en el momento en el que se le solicite la hora. La tercera sección del diagrama de clase contiene las operaciones o comportamientos de la clase. Una operación es lo que pueden hacer los objetos de la clase. Por lo general, se implementa como un método de la clase.

Cada atributo puede tener un nombre, un tipo y un nivel de visibilidad. El tipo y la visibilidad son opcionales. El tipo sigue al nombre y se separa de él mediante dos puntos. La visibilidad se indica mediante un -, #, ~ o + precedente, que indica, respectivamente, visibilidad privada, protegida, paquete o pública. También es posible especificar que un atributo es estático o de clase, subrayándolo. Cada operación puede desplegarse con un nivel de visibilidad, parámetros con nombres y tipos, y un tipo de retorno.

Una clase abstracta o un método abstracto se indica con el uso de cursivas en el nombre del diagrama de clase.

Los diagramas de clase también pueden mostrar relaciones entre clases. Una clase que sea una subclase de otra clase se conecta con ella mediante una flecha con una línea sólida y con una punta triangular hueca. La flecha apunta de la subclase a la superclase. En UML, tal relación se llama generalización. (Pressman, 2021).

Hay otros tipos de relaciones, entre los cuales están:

- Asociaciones: el tipo más simple de relación es una asociación, o una conexión estructural entre clases u objetos. Las asociaciones se muestran como una línea simple en un diagrama de clases. Los puntos finales de la línea se etiquetan con un símbolo que indica la multiplicidad, que es lo mismo que la cardinalidad en un diagrama de entidad-relación. Un cero representa ninguno, un uno representa uno y sólo uno y un asterisco representa muchos. La notación 0..1.
- Relaciones todo/parte: estas relaciones surgen cuando una clase representa al objeto total y otras clases representan partes del mismo. El todo actúa como contenedor de las partes. Estas relaciones se muestran en un diagrama de clases mediante una línea con un diamante en un extremo. El diamante se conecta al objeto total. Una relación todo/parte podría ser un objeto entidad que tiene partes distintas, como un sistema de cómputo que incluye computadora, copiadora, monitor, etc., o un automóvil que tiene motor, sistema de frenos, transmisión, etc. Las relaciones todo/parte también se pueden usar para describir una interfaz de usuario, en la cual una pantalla de GUI contiene una serie de objetos como listas, cuadros o botones de opción, o tal vez un área de encabezado, cuerpo y pie. Las relaciones todo/parte tienen varias categorías: agregación, colección y composición.
- Agregación. A menudo, una agregación se describe como una relación "tiene un". La agregación proporciona un medio para mostrar que el objeto total se compone de la suma de sus partes (otros objetos). En el ejemplo de matriculación del estudiante, el departamento tiene un curso y el curso es para un departamento. Esta es una relación más débil, porque un departamento podría cambiarse o eliminarse y el curso todavía existiría. Un paquete de computadora podría no estar

disponible, pero las impresoras y otros componentes todavía existen. El diamante al final de la línea de la relación no aparece sólido

- Colección: una colección consta de un todo y sus miembros. Este podría ser un distrito electoral con votantes o una biblioteca con libros. Los votantes o libros podrían cambiar pero el todo conserva su identidad. Esta es una asociación débil.
- Composición: la composición, una relación todo/parte en la cual el todo tiene una responsabilidad por la parte, es una relación aún más fuerte, y normalmente se muestra con un diamante sólido. Las palabras clave para la composición son que una clase "siempre contiene" a otra clase. Si el todo se elimina, todas las partes se eliminan. Un ejemplo sería una póliza de seguro con cláusulas adicionales. Si la póliza se cancela, las cláusulas adicionales también se cancelan. En una base de datos, se podría establecer integridad referencial para eliminar los registros hijos en cascada. En una universidad hay una relación de composición entre un curso y una tarea así como entre un curso y un examen. Si el curso se elimina, las tareas y exámenes también se eliminan. (Kendall & Kendall, 2005).

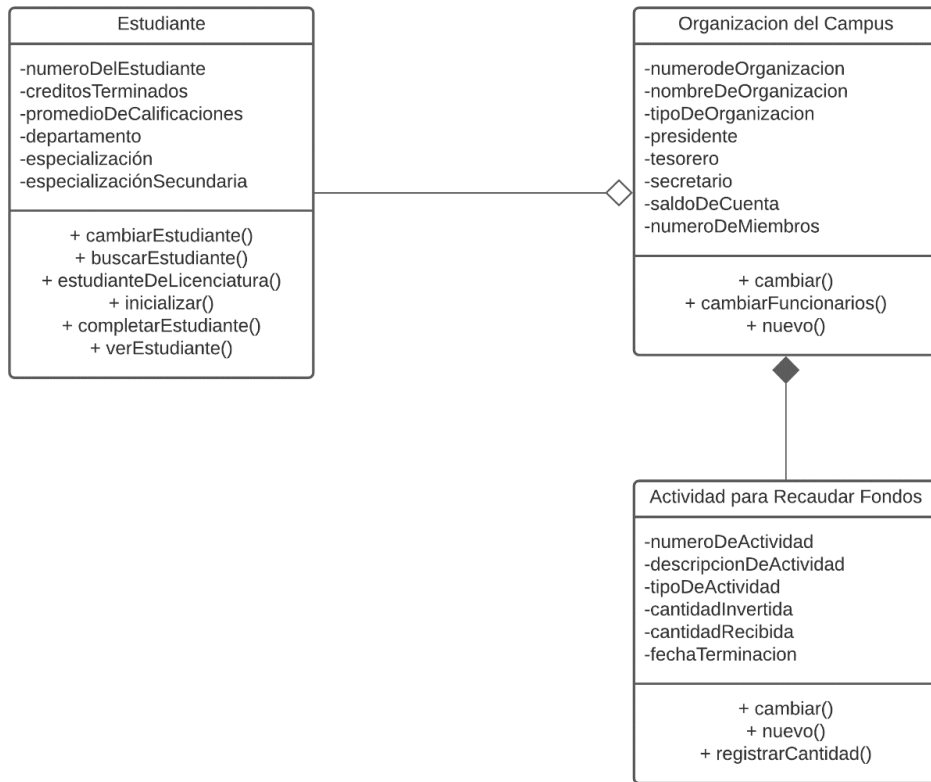


Figura 13. Ejemplo de relaciones todo/parte y de agregación.

Fuente: adaptado de Kendall & Kendall (2005, p.687).

2.2.1.4.2.3 Diagrama de secuencia

Los diagramas de secuencias pueden ilustrar una sucesión de interacciones entre casos o instancias de objetos en un periodo determinado. Los diagramas de secuencias se utilizan con frecuencia para representar el proceso descrito en los escenarios de caso de uso. En la práctica, los diagramas de secuencias se derivan del análisis de casos de uso y se emplean en el diseño de sistemas para generar las interacciones, relaciones y métodos de los objetos del sistema. Los diagramas de secuencias se utilizan para mostrar el patrón general de las actividades o interacciones en un caso de uso. Cada escenario de caso de uso podría crear un diagrama de secuencias, aunque no siempre se crean diagramas de este tipo para los escenarios menores

Con respecto a los símbolos que se utilizan en diagramas de secuencias, los actores y las clases o instancias de los objetos se muestran en recuadros en la parte superior del diagrama. El objeto del extremo izquierdo es el objeto inicial y podría ser una persona (para la cual se emplea símbolo de actor caso de uso), una ventana, un cuadro de diálogo u otra interfaz de usuario. Algunas de las interacciones sólo son físicas, como firmar un contrato. Los rectángulos de la parte superior usan indicadores en el nombre para denotar si el rectángulo representa un objeto una clase, o una clase y un objeto.

nombreDelObjeto: Un nombre seguido de dos puntos representa un objeto.

clase: Dos puntos seguidos de un nombre representan una clase.

nombreDelObjeto:clase Un nombre, seguido de dos puntos y otro nombre, representa un objeto de una clase.

Una línea vertical representa la trayectoria de la vida de la clase o del objeto, que comienza cuando se crea y finaliza cuando se destruye: Una X en el fondo de la trayectoria de la vida indica cuando se destruye el objeto Una barra lateral o rectángulo vertical en la trayectoria de la vida muestran el enfoque de control cuando el objeto se encuentra realizando algo.

Las flechas horizontales muestran mensajes o signos que se envían entre las clases. Los mensajes pertenecen a la clase receptora. Hay algunas variaciones en las flechas de mensaje. Las puntas de flecha solidas representan llamadas síncronas, que son las más comunes. Es tas se usan cuando la clase emisora espera una respuesta de la clase receptora, y el control se devuelve a la clase emisora cuando la clase que recibe el mensaje termina su ejecución.

Las flechas con media punta (o abiertas) representan llamadas asíncronas, es decir, llamadas que se envían sin esperar a que sean devueltas a la clase que las emite. Un ejemplo podría ser el de usar un menú para ejecutar un programa. Un retorno se muestra como una flecha, a veces con una línea punteada. Los mensajes se etiquetan mediante alguno de los formatos siguientes:

- El nombre del mensaje seguido por paréntesis vacíos: nombreDelMensaje().
- El nombre del mensaje seguido por parámetros entre paréntesis: nombreDelMensaje(parámetro1, parámetro2...).

- El nombre del mensaje seguido por el tipo del parámetro, nombre del parámetro y cualquier valor predeterminado para el parámetro entre paréntesis:
nombreDelMensaje(tipoDelParámetro:nombreDelParámetro(valorPredeterminado)).
Los tipos de parámetro indican el tipo de los datos, como numérico, alfanumérico o de tipo de fecha.
- El mensaje podría ser un estereotipo, como <<Create>>, lo cual indica que se crea un nuevo objeto como resultado del mensaje.

En el diagrama de secuencias el tiempo se despliega de arriba abajo; la primera interacción se representa en la parte superior del diagrama, y la última, en la parte inferior. Las flechas de interacción comienzan en la barra del actor o del objeto que inicia la interacción, y terminan apuntando hacia la barra del actor o el objeto que recibe la solicitud de interacción. El actor, la clase o el objeto iniciales se muestran a la izquierda. Este podría ser el actor que inicia la actividad o podría ser una clase que represente la interfaz de usuario. (p. 675, 676).

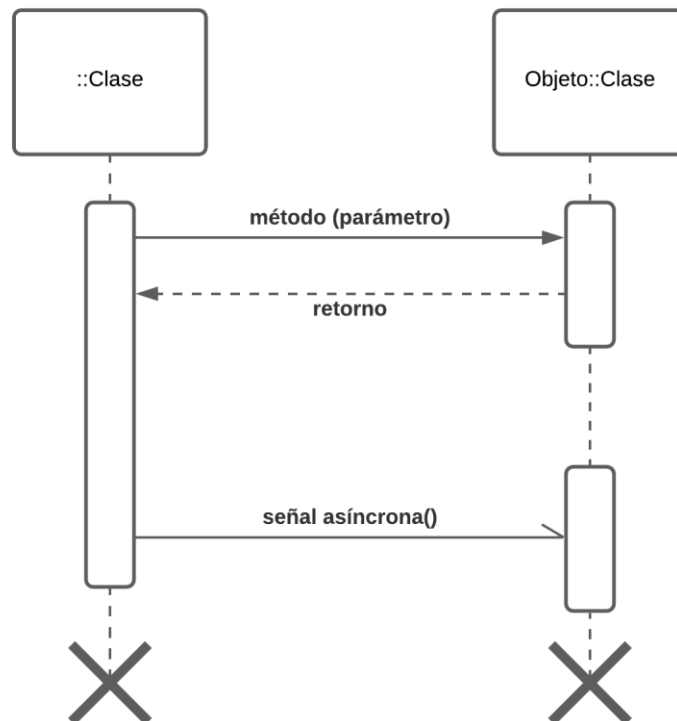


Figura 14. Símbolos especializados usados para dibujar un diagrama de secuencias.

Fuente: adaptado de Kendall & Kendall (2005, p.676).

2.2.1.4.2.4 Diagrama de actividad

Los diagramas de actividades muestran las secuencias de actividades de un proceso, incluyendo las actividades secuenciales, las actividades paralelas y las decisiones que se toman. Por lo general, un diagrama de actividades se elabora para un caso de uso y podría reflejar los diferentes escenarios posibles.

Un rectángulo con esquinas redondeadas representa una actividad, ya sea manual, como firmar un documento legal; o automatizada, como un método o un programa. Una flecha representa un evento. Los eventos representan cosas que ocurren en un tiempo y lugar determinados.

Un diamante representa una decisión (también conocida como rama) o una fusión. Las decisiones tienen una flecha que entra en el diamante y varias que salen de él. Se podría incluir una condición que muestre los valores que puede tomar dicha condición. Las fusiones muestran varios eventos que se combinan para formar otro evento.

Un rectángulo largo y plano representa una barra de sincronización. Esta barra se utiliza para representar actividades paralelas, y podría representar un evento entrando a ella y varios eventos saliendo de la misma, lo que se conoce como bifurcación. Una sincronización en la cual varios eventos se fusionan en uno solo se conoce como unión.

Hay dos símbolos que muestran el inicio y el final del diagrama. El estado inicial se muestra como un círculo sólido. El estado final se muestra como un círculo negro rodeado por un círculo blanco.

Los rectángulos que rodean otros símbolos llamados carriles (swimlanes) indican un particionamiento y se utilizan para mostrar cuáles actividades se realizan en qué plataforma, como un navegador, un servidor o un mainframe; o para mostrar actividades realizadas por diferentes grupos de usuarios. Los carriles son zonas que pueden describir la lógica y la responsabilidad de una clase.

Los diagramas de actividades se crean preguntando qué pasa en primer lugar, que pasa en segundo lugar, y así sucesivamente. Se debe determinar si las actividades se realizan en secuencia o en paralelo. Si se han creado diagramas de flujo de datos físicos, se podrían examinar para determinar la secuencia de actividades. Se deben buscar lugares donde se tomen decisiones, y preguntar qué ocurre con los resultados de cada una de las decisiones. Los diagramas de actividades se podrían crear examinando todos los escenarios para un caso de uso.

El diagrama de actividades proporciona un mapa de un caso de uso, y permite al analista experimentar con la transferencia de partes del diseño a plataformas diferentes y plantearse la pregunta ¿qué pasaría si? para una variedad de decisiones. El uso de símbolos únicos y carriles favorece que las personas prefieran este diagrama para comunicarse con otros. (p.671-675).

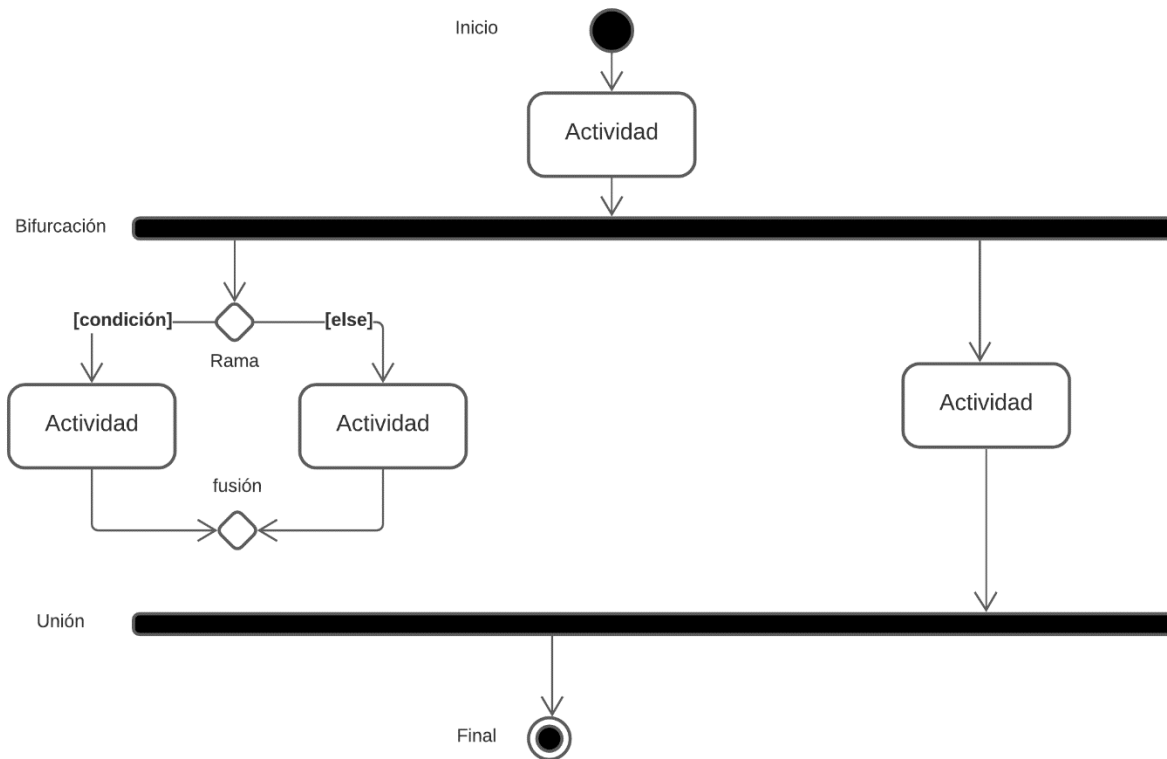


Figura 15. Símbolos especializados para dibujar diagramas de actividad.

Fuente: adaptado de Kendall & Kendall (2005, p.672).

2.2.1.4.3 Diagrama entidad-relación

Denominado por sus siglas como: E-R; Este modelo representa a la realidad a través de un Esquema gráfico empleando los terminología de Entidades, que son objetos que existen y son los elementos principales que se identifican en el problema a resolver con el diagramado y se distinguen de otros por sus características particulares denominadas Atributos, el enlace que rige la unión de las entidades está representada por la relación del modelo.

En un DER, cada entidad se representa mediante un rectángulo, cada relación mediante un rombo y cada dominio (conjunto donde toma valores el atributo) mediante un círculo. Mediante líneas se conectan las entidades con las relaciones, igual que las entidades con los dominios, representando a los atributos. Los Atributos Llaves se representan subrayando el correspondiente conjunto de valores.

En ocasiones, una entidad no puede ser identificada únicamente por el valor de sus propios atributos. En estos casos, se utilizan conjuntamente las relaciones con los atributos para lograr la requerida identificación unívoca. Estas entidades reciben el nombre de entidades débiles y se representan en el DER con un doble rectángulo. El MER restringe las relaciones a usar para identificar las entidades débiles a relaciones binarias del tipo 1: N. Así, por ejemplo, una ocurrencia de "trabajador" puede tener N ocurrencias "persona-dependiente" asociadas, donde además, la existencia de las ocurrencias en la segunda entidad depende de la existencia de una ocurrencia que le corresponda en la primera entidad. Por ejemplo, en el modelo habrá personas dependientes de un trabajador sólo si ese trabajador existe. Para indicar esa dependencia en la existencia se usa una saeta en el DER. La llave de una entidad débil se forma combinando la llave de la entidad regular que la determina con algún otro atributo que defina unívocamente cada entidad débil asociada a una entidad regular dada. (Una entidad se denomina regular si no es débil).

En una relación, la llave es la combinación de las llaves de todas las entidades asociadas. Para cada relación se determina su tipo (simple o complejo) y en el DER se escribe el tipo de correspondencia. Por ejemplo, una empresa puede tener varios (n) trabajadores asociados y un trabajador pertenece a una sola empresa. En la relación Trabajador-Máquina-Pieza, un trabajador puede trabajar en n máquinas, produciendo p piezas, o una pieza puede ser producida por m

trabajadores en n máquinas. Aquí, m, n y p no identifican un número específico, sino solamente el tipo de correspondencia que se establece en la relación. (Aguilera, 2015).

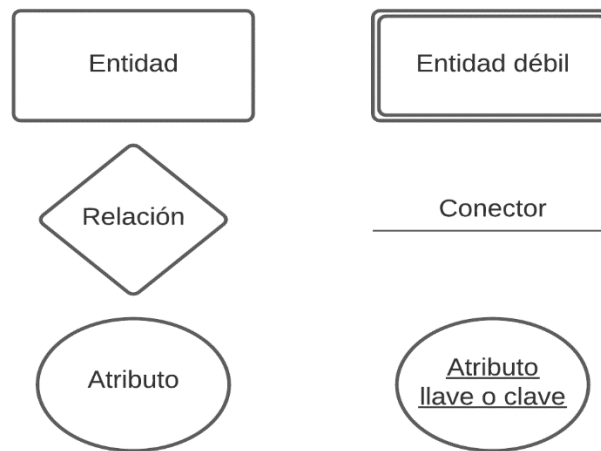


Figura 16. Elementos de un diagrama Entidad-Relación.

Fuente: elaboración propia.

2.2.1.5 Desarrollo del proyecto

En esta sección se ampliarán conceptos relacionados a esta fase del proyecto, por lo que se hablará de bases de datos, lenguajes de programación y todo aquello que vaya a ser utilizado para el desarrollo del sistema web.

2.2.1.5.1 Base de datos

Kendall & Kendall (2005), en su libro Análisis y diseño de sistemas, exponen que una base de datos es una fuente central de datos destinados a compartirse entre muchos usuarios para una diversidad de aplicaciones. El corazón de una base de datos lo constituye el sistema de administración de datos (DBMS, database management system), cual permite la creación, modificación y actualización de la base de datos, la recuperación de datos y la generación de informes y pantallas.

Entre los objetivos de efectividad de la base de datos están los siguientes:

1. Asegurar que los datos se puedan compartir entre los usuarios para una diversidad de aplicaciones.
2. Mantener datos que sean exactos y consistentes.
3. Asegurar que todos los datos requeridos por las aplicaciones actuales y futuras se podrán acceder con facilidad.
4. Permitir a la base de datos evolucionar conforme aumenten las necesidades de los usuarios.
5. Permitir a los usuarios construir su vista personal de los datos sin preocuparse por la forma en que los datos se encuentren almacenados físicamente.

La anterior lista de objetivos de objetivos nos proporciona un recordatorio de las ventajas y desventajas del enfoque de base de datos. Primero, la compartición de los datos significa que estos deben almacenarse una sola vez. Como consecuencia, esto ayuda a lograr la integridad de los datos, debido a que los cambios en los datos se realizan con mayor facilidad y confiabilidad si estos aparecen solo una vez en lugar de en muchos archivos diferentes.

Cuando un usuario necesita datos específicos, una base de datos bien diseñada anticiparía dicha necesidad (o quizás ya se habrían usado en otra aplicación). Por tanto, es más probable que los datos estén disponibles en una base de datos que en un sistema de archivos convencional. Una base de datos bien diseñada también puede ser más flexible que los archivos separados; es decir, una base de datos puede evolucionar conforme cambien las necesidades de los usuarios y las aplicaciones.

Finalmente, el enfoque de base de datos tiene la ventaja de permitir a los usuarios obtener su propia vista de los datos. Los usuarios no tienen que preocuparse por la estructura real de la base de datos o su almacenamiento físico.

2.2.1.5.1.1 Tabla

Las tablas son objetos de base de datos que contienen todos sus datos. En las tablas, los datos se organizan con arreglo a un formato de filas y columnas, similar al de una hoja de cálculo. Cada fila representa un registro único y cada columna un campo dentro del registro. Por ejemplo, en una tabla que contiene los datos de los empleados de una compañía puede haber una

fila para cada empleado y distintas columnas en las que figuren detalles de los mismos, como el número de empleado, el nombre, la dirección, el puesto que ocupa y su número de teléfono particular. (Microsoft, 2019).

2.2.1.5.1.2 Relación

Estas son las asociaciones entre las entidades, objetos o eventos sobre el cual alguien escoge recopilar datos, (a veces se conocen como asociaciones de datos). Existen distintos tipos de relaciones, entre los cuales están:

1. Relación uno a uno (1:1), este tipo de relación especifica que cada entidad solo se puede relacionar con una entidad.
2. Relación uno a muchos (1: M) o muchos a uno, en esta relación una entidad puede relacionarse con más de una entidad.
3. Relación muchos a muchos (M: N), esta relación describe la posibilidad de que las entidades podrían muchas asociaciones en cualquier dirección. (Kendall & Kendall, 2005, p.447).

2.2.1.5.2 Microsoft SQL Server

Microsoft SQL Server es un sistema de gestión de base de datos relacional (RDBMS) producido por Microsoft. Su principal lenguaje de consulta es Transact-SQL, una aplicación de las normas ANSI / ISO estándar Structured Query Language (SQL) utilizado por ambas Microsoft y Sybase.

Algunas características de Microsoft SQL Server son:

- Soporte de transacciones.
- Escalabilidad, estabilidad y seguridad.
- Soporta procedimientos almacenados.
- Incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.

- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información.
- Además permite administrar información de otros servidores de datos.

Este sistema incluye una versión reducida, llamada MSDE con el mismo motor de base de datos pero orientado a proyectos más pequeños, que en su versión 2005 pasa a ser el SQL Express Edition, que se distribuye en forma gratuita.

Microsoft SQL Server constituye la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos como son Oracle, Sybase ASE, PostgreSQL o MySQL. (Santamaría & Hernández, 2016).

2.2.1.5.3 Lenguaje de programación

De acuerdo con San Martín Cuenca & Tusa Jumbo (2015), un lenguaje de programación es un vocabulario o conjunto de reglas gramaticales que instruyen a una computadora a desarrollar una tarea específica. Es decir, es un sistema notacional para describir instrucciones de una forma legible tanto para la máquina como para el ser humano.

Los lenguajes de programación se caracterizan por su eficiencia, expresividad, capacidad de mantenimiento, legibilidad, confiabilidad, seguridad, simplicidad, y productividad.

Las ventajas que existen en el estudio de los lenguajes de programación, son las siguientes:

- Aumenta la capacidad para producir software.
- Mejora la habilidad para desarrollar mejores algoritmos.
- Potencia el uso de los lenguajes de programación.
- Incrementa el vocabulario.
- Permite elegir mejor el lenguaje a utilizar para cada tarea.
- Hace más fácil aprender nuevos lenguajes de programación.

2.2.1.5.3.1 HTML

Laudon & Laudon (2016) mencionan que HTML (Lenguaje de marcado de hipertexto) es un lenguaje de descripción de páginas para especificar la forma en que se colocan el texto, los gráficos, el video y el sonido en una página Web, y para crear vínculos dinámicos a otras páginas Web y objetos. Mediante el uso de estos vínculos, un usuario sólo necesita apuntar a una palabra clave o gráfico resaltado, hacer clic en él y transportarse de inmediato a otro documento.

En un principio, HTML se diseñó para crear y vincular documentos estáticos compuestos en su mayor parte de texto. Sin embargo, en la actualidad, la Web es mucho más social e interactiva; muchas páginas Web tienen elementos multimedia (imágenes, audio y video). Las aplicaciones de complementos de terceros como Flash, Silverlight y Java se requieren para integrar estos medios enriquecidos con las páginas Web. No obstante, estos complementos requieren programación adicional y ejercen presión en el procesamiento de computadora. La siguiente evolución de HTML, conocida como HTML5, resuelve este problema al hacer posible la incrustación de imágenes, audio, video y otros elementos directamente en un documento sin complementos que hagan uso intensivo del procesador. HTML5 facilita que las páginas Web funcionen en distintos dispositivos de visualización, tanto en dispositivos móviles como en equipos de escritorio; además, respalda el almacenamiento de datos sin conexión para las app que se ejecutan a través de Web.

2.2.1.5.3.2 CSS

Cascade Style Sheet (CSS), hoja de estilo en cascada, en español, es el estándar de W3C para definir la presentación de documentos escritos en HTML y cualquier lenguaje XML. La presentación se refiere a cómo es presentado o entregado el documento al usuario, ya sea en una pantalla de computadora, de móvil, impreso o leído en voz alta por un lector de pantalla. Con las hojas de estilo encargándose de la presentación, el HTML puede encargarse de definir la estructura del documento y significado, como es su intención.

Ahora, utilizar CSS tiene grandes beneficios, como los siguientes:

1. Control preciso de tipo y diseño: se puede lograr una precisión similar a la de la impresión usando CSS. Incluso hay un conjunto de propiedades destinadas específicamente para páginas impresas.
2. Menos trabajo: se puede cambiar la apariencia de un sitio completo editando una hoja de estilo.
3. Sitios más accesibles: Cuando todos los asuntos de presentación son manejados por CSS, se puede marcar el contenido de manera significativa, haciéndolo más accesible para dispositivos móviles o no visuales.
4. Soporte de navegador confiable: Todos los navegadores actualmente en uso son compatibles con CSS2 y con algunas partes de CSS3. (Robbins, 2012).

2.2.1.5.3.3 Javascript

JavaScript es el lenguaje de programación que agrega interactividad y comportamientos personalizados a nuestros sitios. Es un lenguaje de programación del lado del cliente, lo que significa que se ejecuta en la máquina del usuario y no en el servidor, como lo hacen otros lenguajes de programación web. Eso significa que JavaScript (y la forma en que se usa) depende de las capacidades y la configuración del navegador. Es posible que ni siquiera esté disponible, ya sea porque el usuario ha optado por apagarlo o porque el dispositivo no lo admite, algo que los buenos desarrolladores tienen en cuenta y planifican. JavaScript es también lo que se conoce como un lenguaje de programación dinámico y de tipo flexible.

Como lenguaje de programación dinámico, JavaScript no necesita ejecutarse a través de ningún tipo de compilador que interprete el código legible por humanos en algo que el navegador pueda entender. El navegador lee el código de la misma manera que nosotros y lo interpreta sobre la marcha.

JavaScript también está escrito de manera flexible. Todo esto significa que no necesariamente hay que decirle a JavaScript qué es una variable. Si se establece una variable con un valor de 5, no es necesario que especifique de forma programática esa variable como un número, Javascript lo reconoce como tal. (p.459-460).

2.2.1.5.3.4 JQuery

De acuerdo con Duckett, Ruppert & Moore (2014), en su libro Javascript y JQuery: interactive front-end web development, jQuery es una librería de Javascript, es decir, un archivo de javascript que se incluye en las páginas web. Este permite encontrar elementos utilizando selectores de estilos CSS y hacer algo con ellos utilizando métodos de jQuery. Algunas similitudes con el DOM de Javascript son:

1. Los selectores de jQuery realizan tareas similares a las del DOM, pero la sintaxis es más simple.
2. Se pueden almacenar objetos jQuery en una variable, al igual que como se hacía con los nodos del DOM.
3. Se pueden utilizar métodos y propiedades de jQuery (como los métodos y propiedades del DOM) para manipular los nodos del DOM que se seleccionen.

Así como hay similitudes entre jQuery y el DOM de Javascript, también hay diferencias, entre las cuales están:

1. Es multiplataforma, no hay necesidad de escribir código de respaldo.
2. Seleccionar elementos es más simple (dado que son selectores CSS) y es más adecuado.
3. El manejo de eventos es más simple dado que usa un método que funciona en la mayoría de navegadores.
4. Los métodos afectan todos los elementos seleccionados sin necesidad de realizar un bucle que recorra cada elemento.
5. Se proveen métodos adicionales para tareas populares requeridas, como animación.
6. Una vez que se ha hecho una selección se le pueden aplicar múltiples métodos.

2.2.1.5.3.5 Bootstrap

Bootstrap es un producto open source, es decir que es accesible al público, puede ser visto, modificado y distribuido a conveniencia, de Mark Otto y Jacob Thornton, quienes, cuando fue lanzado por primera vez, era empleados de Twitter. Había una necesidad de estandarizar las herramientas de front-end de los ingenieros en la empresa. El proyecto fue presentado como una

solución a los problemas de mantenimiento y escalabilidad debido al uso de múltiples librerías para cumplir con los requerimientos de front-end.

Desde que Bootstrap se lanzó en 2011, ha tomado mucha popularidad y ha evolucionado de ser un proyecto impulsado totalmente en CSS a contener plugins de Javascript e íconos que van de la mano con botones y formularios. (Spurlock, 2013).

2.2.1.5.3.6 Owl Carousel

Owl Carousel es un plugin de JQuery que permite crear carruseles responsivos de manera sencilla. Es totalmente personalizable tanto para usuarios novatos como para usuarios avanzados; permite mejorar la experiencia de navegación en móvil, la mayoría de opciones que ofrecen son completamente responsivas, es decir que se ajustan a los distintos tipos de pantalla que existen; utiliza la aceleración de hardware con transiciones CSS3 Translate3d, también posee respaldo CSS2 compatible con navegadores más antiguos; por último, Owl Carousel admite la estructura modular de complementos. Por lo tanto, puede separar complementos que no usará en su proyecto o crear nuevos que se adapten a sus necesidades. (Deutsch, s. f.).

2.2.1.5.3.7 C#

C# es un lenguaje de programación simple, moderno, orientado a objetos y con seguridad de tipos. C # tiene sus raíces en la familia de lenguajes C y será familiar de inmediato para los programadores de C, C ++ y Java. C# está estandarizado por ECMA International como el estándar ECMA-334 y por ISO / IEC como el estándar ISO / IEC 23270. El compilador C# de Microsoft para .NET Framework es una implementación conforme de ambos estándares.

C# es un lenguaje orientado a objetos, pero C# incluye además soporte para programación orientada a componentes. El diseño de software contemporáneo se basa cada vez más en componentes de software en forma de paquetes de funcionalidad auto contenidos y auto descriptivos. La clave de tales componentes es que presentan un modelo de programación con propiedades, métodos y eventos; tienen atributos que proporcionan información declarativa sobre el componente; e incorporan su propia documentación. C# proporciona construcciones de lenguaje para respaldar directamente estos conceptos, lo que convierte a C# en un lenguaje muy

natural para crear y utilizar componentes de software. (Hejlsberg, Torgersen, Wiltamuth, & Golde, 2008)

2.2.1.5.3.8 Microsoft Visual Studio

Visual Studio es la interfaz de desarrollo de Microsoft. Se compone de un conjunto de herramientas que permiten a los desarrolladores crear aplicaciones para las plataformas .NET. Visual Studio se divide en varias ediciones:

- **Community:** Microsoft proporciona gratuitamente esta edición de Visual Studio 2017. Su objetivo es servir en la formación de los estudiantes, desarrolladores open source y desarrolladores particulares. Reúne todas las funcionalidades básicas para la creación de proyectos. Esta edición contiene todas las herramientas de desarrollo multiplataforma para las aplicaciones móviles Windows, iOS y Android. La galería Visual Studio permite acceder a numerosas herramientas, modelos y controles para acelerar el desarrollo.
- **Professional:** Edición dirigida a desarrolladores profesionales individuales o a grupos pequeños de trabajo. Las herramientas de depuración y pruebas unitarias forman parte de las funcionalidades más destacables de esta edición.
- **Enterprise:** Para los equipos profesionales que trabajan en proyectos que necesitan más interacción entre los miembros. Las funcionalidades destacables son las herramientas de test, las herramientas colaborativas (Team Foundation Server) y las herramientas de arquitectura y modelización de aplicaciones. (Hugon, 2018).

Durante la elaboración y el desarrollo de este proyecto se usará el Visual Studio Community 2019.

2.2.1.5.3.9 Modelo Vista Controlador

En el artículo de Fernández Romero y Díaz González (2012), se indica que buscando un poco de información histórica, es posible afirmar que el patrón Modelo/Vista/Controlador o MVC (Model/View/Controller) fue descrito por primera vez en 1979 por Trygve Reenskaug[1] e introducido como parte de la versión Smalltalk-80 del lenguaje de programación Smalltalk.

Fue diseñado para reducir el esfuerzo de programación necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos. Sus características principales están dadas por el hecho de que, el Modelo, las Vistas y los Controladores se tratan como entidades separadas; esto hace que cualquier cambio producido en el Modelo se refleje automáticamente en cada una de las Vistas. Este modelo de arquitectura se puede emplear en sistemas de representación gráfica de datos, donde se presentan partes del diseño con diferente escala de aumento, en ventanas separadas.

Este modelo de arquitectura presenta varias ventajas:

- Separación clara entre los componentes de un programa; lo cual permite su implementación por separado.
- Interfaz de Programación de Aplicaciones API (Application Programming Interface) muy bien definida; cualquiera que use el API, podrá reemplazar el Modelo, la Vista o el Controlador, sin aparente dificultad.
- Conexión entre el Modelo y sus Vistas dinámica; se produce en tiempo de ejecución, no en tiempo de compilación.

Al incorporar el modelo de arquitectura MVC a un diseño, las piezas de un programa se pueden construir por separado y luego unir las en tiempo de ejecución. Si uno de los componentes, posteriormente, se observa que funciona mal, puede reemplazarse sin que las otras piezas se vean afectadas. Este escenario contrasta con la aproximación monolítica típica de muchos programas de pequeña y mediana complejidad. Todos tienen un Frame que contiene todos los elementos, un controlador de eventos, un montón de cálculos y la presentación del resultado. Ante esta perspectiva, hacer un cambio aquí no es nada trivial.

El Modelo es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico de los Controladores o de las Vistas, ni siquiera contiene referencias a ellos. Es el propio sistema el que tiene encomendada la responsabilidad de mantener enlaces entre el Modelo y sus Vistas, y notificar a las Vistas cuando cambia el Modelo.

Este se encarga de:

- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
- Definir reglas de negocio. (la funcionalidad del sistema). Un ejemplo de regla puede ser: "Si la mercancía pedida no está en el almacén, consultar el tiempo de entrega estándar del proveedor". Es opcional, pues las reglas de negocio, pueden estar también en los controladores, directamente en las acciones.
- Notificará a las vistas los cambios que en los datos pueda producir un agente externo si se está ante un modelo activo (por ejemplo, un fichero bath que actualiza los datos, un temporizador que desencadena una inserción, etc.).

El Controlador es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el Modelo, centra toda la interacción entre la Vista y el Modelo. Cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del Modelo o por alteraciones de la Vista. Interactúa con el Modelo a través de una referencia al propio Modelo.

Este es responsable de:

- Recibir los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
- Contener reglas de gestión de eventos, del tipo "Si Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas. Una de estas peticiones a las vistas puede ser una llamada al método "Actualizar ()". Una petición al modelo puede ser "Obtener_tiempo_de_entrega(nueva_orden_de_venta)".

La Vista es el objeto que maneja la presentación visual de los datos representados por el Modelo. Genera una representación visual del Modelo y muestra los datos al usuario. Interactúa preferentemente con el Controlador, pero es posible que trate directamente con el Modelo a través de una referencia al propio Modelo.

La vista se encarga de:

- Recibir datos procesados por el controlador o del modelo y mostrarlos al usuario.
- Tienen un registro de su controlador asociado.

- Pueden dar el servicio de "Actualización ()", para que sea invocado por el controlador o por el modelo cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes. (p. 48-50)

2.2.1.5.3.10 Razor

Razor es una sintaxis de plantilla que le permite combinar código y contenido de una manera fluida y expresiva. Aunque presenta algunos símbolos y palabras claves nuevas, Razor no es un lenguaje nuevo. En cambio, Razor le permite escribir código usando lenguajes que probablemente ya conozca, como C # o Visual Basic .NET.

La curva de aprendizaje de Razor es muy corta, ya que le permite trabajar con sus habilidades existentes en lugar de requerir que aprenda un idioma completamente nuevo. (Chadwick, 2011).

2.2.1.5.3.11 Entity Framework

En el libro *Mastering Entity Framework*, Singh (2015), menciona que Entity Framework es un mapeador de objetos relacionales (ORM) de Microsoft que permite a los desarrolladores de la aplicación trabajar con datos relacionales como modelos comerciales. Eso elimina la necesidad de la mayor parte del código de plomería que escriben los desarrolladores (mientras utilizando ADO.NET) para acceder a los datos. Entity Framework proporciona un sistema basado en modelos que hace que la creación de una capa de acceso a datos sea muy fácil para los desarrolladores liberándolos de escribir un código de acceso a datos similar para todos los modelos de dominio. La versión inicial de Entity Framework fue Entity Framework 3.5. Fue lanzado con .NET Framework 3.5 SP1 y Visual Studio 2008 SP1. Entity Framework ha evolucionado mucho desde entonces y actualmente va por la versión 6.

Entity Framework facilita la tarea de crear una capa de acceso a datos al permitir el acceso de datos, representando los datos como un modelo conceptual, es decir, un conjunto de entidades y relaciones. La aplicación puede realizar el CRUD básico (crear, leer, actualizar y eliminar) operaciones y administrar fácilmente uno a uno, uno a muchos y muchos a muchos relaciones entre las entidades.

A continuación, se muestran algunos beneficios de usar Entity Framework:

- El tiempo de desarrollo se reduce ya que los desarrolladores no tienen que escribir todo el código de plomería ADO.NET necesario para el acceso a los datos
- Podemos tener toda la lógica de acceso a los datos escrita en un lenguaje de nivel superior, como C # en lugar de escribir consultas SQL y más almacenados
- Dado que las tablas de la base de datos no pueden tener relaciones avanzadas (herencia) como las entidades de dominio pueden, el modelo de negocio, es decir, el modelo conceptual puede ser utilizado para adaptarse al dominio de la aplicación utilizando relaciones entre las entidades.
- El almacén de datos subyacente se puede reemplazar con relativa facilidad si usamos un ORM ya que toda la lógica de acceso a datos está presente en nuestra aplicación en lugar de la capa de datos. Si no se utiliza un ORM, sería comparativamente difícil para hacerlo.

La arquitectura de Entity Framework es la siguiente:

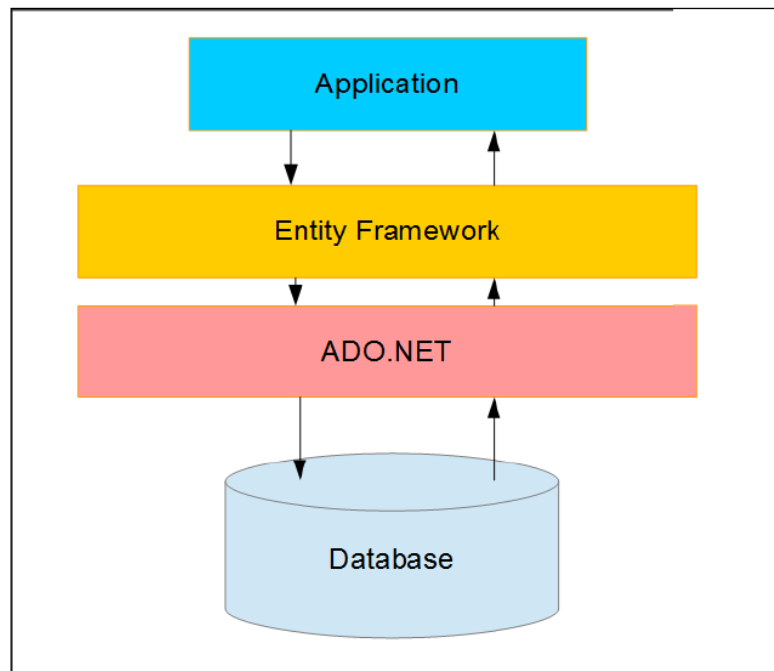


Figura 17. Arquitectura de Entity Framework.

Fuente: Singh (2015, p.8).

En la figura anterior, se puede ver que Entity Framework está escrito encima del marco ADO.NET, y debajo, todavía está usando los métodos ADO.NET y clases para realizar las operaciones de datos. (p.8).

2.2.1.5.3.12 Linq

Language-Integrated Query (LINQ), es una técnica para consultar datos desde lenguajes .NET. LINQ to Entities es el mecanismo que facilita el uso de LINQ para escribir consultas contra nuestro modelo conceptual, como el modelo de datos de entidad.

Cuando usamos LINQ hacia Entidades para ejecutar las consultas LINQ hacia la entidad del modelo de datos, estas consultas LINQ se compilan primero para determinar qué datos queremos traer. Luego, se ejecuta y desde la perspectiva de la aplicación, los resultados se retornarán como objetos CLR, que es algo que .NET puede entender. (p.168)

2.2.1.5.3.13 ASP.NET MVC

El ASP.NET MVC es un patrón de arquitectura que provee una alternativa al ASP.NET Web Forms, para la creación de aplicaciones Web basadas en el MVC (Model View Controller). El Framework MVC es definido por el namespace System.Web.Mvc. El ASP.NET MVC es un patrón al que muchos desarrolladores están acostumbrados, donde separan en capas el modelo, la visualización y los controles. Algunas aplicaciones Web pueden beneficiarse del framework MVC, mientras que otras continuarán usando el patrón tradicional de ASP.NET, que está basado en Web Forms y postbacks. Ningún tema, sea el MVC o el Web Forms, excluirá a la otra, pudiendo inclusive ser utilizadas a la vez. (Arias, 2015)

2.2.1.5.4 Pruebas

De acuerdo con Kendall & Kendall (2005), antes de poner el sistema en funcionamiento es necesario probarlo. Es mucho menos costoso encontrar los problemas antes que el sistema se entregue a los usuarios. Primero se realiza una serie de pruebas con datos de muestra para

determinar con precisión cuáles son los problemas y posteriormente se realiza otra con datos reales del sistema.

Las pruebas son un aspecto crucial en el control de calidad de software y, dentro de estas, las pruebas funcionales, en las cuales se hace una verificación dinámica del comportamiento de un sistema, basada en la observación de un conjunto seleccionado de ejecuciones controladas o casos de prueba [1]. Las pruebas funcionales son aquellas que se aplican al producto final, y permiten detectar en qué puntos el producto no cumple sus especificaciones, es decir, comprobar su funcionalidad [2]. Para realizarlas se debe hacer una planificación que consiste en definir los aspectos a examinar y la forma de verificar su correcto funcionamiento, punto en el cual adquieren sentido los casos de prueba. (Palacio, 2009).

2.2.1.6 Implementación del proyecto

Este apartado es el último relacionado al ciclo de vida de desarrollo del proyecto, dado que es la última fase. Kendall & Kendall (2005, p.13) mencionan que durante esta fase se capacita a los usuarios para utilizar el nuevo sistema de software, la conversión gradual del sistema anterior al actual si fuera el caso, instalación de equipo y la puesta en producción del nuevo sistema.

2.3 Tecnología legal

En este último apartado se desarrollarán los términos que fusionan la informática con la abogacía. Tecnología que bien dirigida y aplicada puede ser de provecho para potenciar el ejercicio profesional de los abogados.

2.3.1 Iuscibernética

Como derivación de la cibernética aparece la iuscibernética, según la denominación propuesta por Mario G. Losano, en un trabajo publicado años atrás, recogido en una obra de divulgación titulado "Giuscibernética": máquinas y modelos cibernéticos en el derecho. Para Losano la iuscibernética recoge en principio, todas las aplicaciones de la cibernética al Derecho.

Nos evoca, dice, la idea de una cuantificación del fenómeno jurídico; es un método de investigación, cualquiera que sea la materia a que se aplique. El derecho puede ser analizado en diversos planos o niveles. En todos ellos es aplicable el método cibernético.

Losano señala cuatro Filosofías para la investigación cibernética del derecho. En ese sentido, a juicio propio, lo más cercano a una conceptualización que se tiene y de los que varios estudiosos deberían aproximar sus ideas, esta sobre el objeto de la iuscibernética que se desglosa en los siguientes campos de investigación:

1. El estudio de las relaciones entre el derecho y el sistema social según un modelo cibernético
2. La concepción del derecho como un sistema cerrado, autónomo y autorregulado.
3. Formalización del lenguaje jurídico, condición previa a su incorporación al proceso cibernético
4. El tratamiento de las normas jurídicas como informaciones y su ordenación electrónica.

Otra forma en la que se puede definir iuscibernética es: La ciencia general de estudio y aplicación concernientes a las ciencias jurídicas en su desarrollo e implementación de las tecnologías de la comunicación e información, entre el mundo litigante y los administradores de justicia. (Menchaca, 2014).

La iuscibernética se divide en 4 disciplinas que derivan de ella: jurimetría, informática jurídica, modelística jurídica y derecho informático. Para este proyecto se desarrollará más acerca de la informática jurídica.

2.3.1.1 Informática jurídica

La informática jurídica es la disciplina que sostiene ya no la sustitución del juez sino la ayuda, entre otras, a la función jurisdiccional por las computadoras, pudiéndose obtener de manera más eficaz y eficiente la información jurídica.

Exponiendo que es una disciplina bifronte en la que se entrecruzan una metodología tecnológica con sus posibilidades y modalidades de tal aplicación. La informática jurídica estudia el tratamiento automatizado de las fuentes del conocimiento jurídico a través de los

sistemas de documentación legislativa, jurisprudencial y doctrinal (informática jurídica documental); las fuentes de producción jurídica, a través de la elaboración informática de los factores lógicoformales que concurren en proceso legislativo y en la decisión judicial (informática jurídica decisional); y los procesos de organización de la infraestructura o medios instrumentales con los que se gestiona el Derecho (informática jurídica de gestión).

La Informática Jurídica como todo sistema de tratamiento de información, requiere de elementos de evaluación que le permitan obligarse a la propia ciencia superar los retos exigidos por el ser humano, estos son los que determinan sus características.

Para alcanzar una adecuada evaluación del tratamiento automatizado de datos jurídicos hay que apuntar las características propias que son: eficiente, aprovechable, accesible y ágil.

- **Eficiente:** La informática Jurídica debe permitir determinar las respuestas de informaciones necesitadas en relación con las informaciones memorizadas por el computador.
Se entenderá por eficiente a la Informática Jurídica cuando se nos sea permitido alcanzar la mayor cantidad de información o datos requeridos del banco de datos que ofrezcan información concerniente a la solicitada.
- **Aprovechable:** Es la relación entre el uso efectivo de informaciones recuperadas y el máximo uso posible del sistema. Una vez alcanzada la posibilidad de que el ordenador se encuentre ofreciendo la información solicitada, apartando de la que no tenga nada que ver con la pertinencia de la materia, se procurará que ofrezca la mayor cantidad de información.
- **Accesible:** El nivel de dificultad que tenga el usuario para alcanzar la información requerida es otro elemento que no debe descuidar la Informática Jurídica. Los computadores y sobre todo el software que se desarrolle para el manejo del jurista deberán tener una orientación clara y fácil de manejo.
- **Ágil:** Es el tiempo que transcurre entre el momento que el usuario realiza los cuestionamientos o solicitudes al ordenador y el momento en que la computadora contesta con la debida propiedad. Obvio es pensar que los adelantos tecnológicos tienden a agilizar los sistemas y procesos instituidos.

El desarrollo de medios que permitan transportar la información de manera más rápida, no solo le significa al hombre la posibilidad de alcanzar el cumplimiento de mayores objetivos, sino que permite el incremento productivo y por tanto el desarrollo de las sociedades tecnológicamente conectadas.

No sería necesaria la Informática Jurídica sino cumpliera su cuarto elemento de evaluación, esto es que tenga y se mejore cada día su velocidad operativa.

La informática jurídica se divide en dos grandes ramas:

- Informática jurídica documental

La Informática Documental se encamina principalmente a la formación de bancos de datos que incluyan información idónea y suficiente para ser consultada y obtenida cuando sea menester contar con ella, es decir, que son los bancos de datos jurídicos que archivan y recuperan información legislativa, doctrinaria, y jurisprudencial.

Si bien existe diversidad de criterios y metodologías para el procesamiento de la información jurídica, está claro que en su tratamiento deben considerarse tres aspectos básicos:

1. La selección de datos que es labor de juristas y se refiere a normativa, jurisprudencia, doctrina, bibliografía.
2. El programa mediante el cual se procesará la información, que deberá permitir su modificación y consulta, que es una labor de analistas.
3. La uniformidad del lenguaje, necesaria para la correcta búsqueda de documentos, que es una labor interdisciplinaria, básicamente a cargo de especialistas en clasificación jurídica y manejo de herramientas de control terminológico.

El objeto de esta área es la automatización en el tratamiento y recuperación de información jurídica. Para el profesional del Derecho es imprescindible tener un acceso rápido y directo a la legislación y la jurisprudencia. No obstante, dado el flujo incesante de nuevas leyes y de decisiones jurisprudenciales, gestionar esta información de forma manual es algo imposible. En las circunstancias actuales se hace necesaria la utilización de sistemas informatizados de documentación jurídica (bases de datos de legislación,

jurisprudencia y doctrina), que le permitan al jurista el acceso a grandes volúmenes de información de una manera eficaz.

- Informática Jurídica de Gestión

La Informática de Gestión, es el instrumento que puede y de hecho hace de toda actividad jurídica algo organizado, es decir que la Informática de Gestión es el medio para organizar y hacer más funcionales las instituciones públicas o privadas que puedan de alguna forma tocar o tratar temas jurídicos. Es la que genera sistemas informáticos aplicables a actividades específicas relativas al derecho; por ejemplo, registro y seguimiento de proyectos de leyes, labor de los juzgados, oficinas administrativas y cámaras legislativas. Bajo este epígrafe entraría la gestión de todas las labores de tipo rutinario que se llevan a cabo en un despacho jurídico o en un juzgado. De esta manera, se trataría de automatizar tareas tales como la contabilidad y facturación, la creación y almacenamiento de textos, el archivo de datos (clientes, personal), la gestión y seguimiento de expedientes, etc.

Otro aspecto importante lo constituye la posibilidad de establecer una comunicación entre ordenadores. Esto permite el intercambio de información entre los propios ordenadores del despacho mediante la utilización de una red local. Por otro lado, la creciente expansión de las redes de área amplia, cuyo ejemplo más patente lo constituye la red INTERNET y que permiten conectar ordenadores sin límite de distancias, hace posible la comunicación del despacho con otros bufetes, con los tribunales, los colegios de abogados, notarías, registros, etc. Con ello, además de obtener la información sin necesidad de perder tiempo en desplazamientos, se puede conseguir el acceso a la misma en cualquier momento sin la limitación que imponen los horarios de oficina.

Este tipo de Informática está denominada por el procesamiento de datos, de modo tal que puede hacer referencia a sistemas tales como:

- Sistemas de Gestión de Oficinas Jurídicas (públicas y privadas);
- Sistematización del Reparto Judicial.
- Sistematización del Registro Penitenciario.
- Sistematización de Notariado y Registro
- Sistematización del registro de Contravenciones

- Sistematización de Consultorios Jurídicos.

La Informática de Gestión puede dividirse en:

1. **Informática Registral:** La informática registral, es la que se ocupa de todos los tipos de datos, empleando un tratamiento electrónico de los mismos en registros sean públicos o privados, con la posibilidad de modificarlos a futuro. En está lo que se trata es de facilitar a los diferentes usuarios son datos fehacientes en todos los registros oficiales con mayor rapidez y facilidad de acceso que por los métodos tradicionales. La Informática Registral permite igualmente llevar a cabo labores complementarias tan importantes como estadística, eliminar documentos escritos, facilitar los trámites, etc.

En conclusión podemos concebirlo, a su vez, como informática documental, porque el Registro procesa, archiva y recupera la información contenida en documentos. Y es de gestión porque los archivos del Registro están "vivos" ya que se van modificando con el transcurso del tiempo y van incorporando nuevas informaciones bajo el mismo rótulo. Por este motivo es que prefiere incluir a la informática registral dentro del campo de la informática de gestión.

2. **Informática Operacional:** Se utiliza en el campo jurídico procesal para controlar la actuación o etapas de un proceso específico, tanto en la vía administrativa como en la judicial. Es decir que permite que por ejemplo en un juzgado el ordenador controle toda la actuación o pasos que en una clase de proceso específico, se vayan cumpliendo los pasos o mejor las etapas del proceso en concreto.
3. **Informática Decisional:** Es el empleo de los ordenadores, en la ayuda de la toma de decisiones. La Informática Jurídica Decisional es llamada o conocida también como Informática Metadocumental y su propósito es entregar soluciones a los problemas jurídicos planteados, a través de la inteligencia artificial, resuelve automáticamente casos, mediante computadores y sistemas expertos.

La Informática Decisional, es la aplicación más avanzada, ya que se refiere a la inteligencia artificial o a sistemas expertos. Permite que el ordenador aborde respuestas similares a los que abordaría un experto humano en una materia determinada. Esta rama se ocupa de la

automatización de los procesos de decisión a los que se enfrentan los juristas. Es decir, se trataría de proporcionar soluciones a los problemas y no simplemente documentación sobre los mismos.

La informática jurídica decisional está relacionada con un campo denominado Inteligencia Artificial, que se podría definir de manera muy simple como el área de estudio dentro de la informática cuyo objetivo es hacer razonar a un ordenador de manera similar a como lo hacen los humanos. De esta manera, y trasladada al campo jurídico, la IA se aplicaría a la creación de “sistemas inteligentes”, que incorporarían conocimientos jurídicos y estrategias de razonamiento apropiadas, con el fin de servir de ayuda al usuario en la resolución de problemas legales. Otras posibles aplicaciones podrían ser: el desarrollo de sistemas “inteligentes” para la enseñanza del Derecho, la ayuda en los procesos legislativos, o como una alternativa a los sistemas tradicionales de recuperación de información jurídica. (p.119- 126).

Para este proyecto el tipo de informática jurídica que se aplicará es la informática jurídica de gestión, y de tipo registral, pues el proyecto consta en un sistema de registro y gestión de expedientes jurídicos para el bufete Fonseca Sequeira y asociados.

2.3.2 Legaltech

Como se menciona en el capítulo I, legalTech es la adopción de tecnologías innovadoras para mejorar los servicios legales, o en otras palabras, el hardware y el software usado en la profesión legal. El término LegalTech es el resultado del crecimiento exponencial de nuevas tecnologías, el cual está influenciando directamente al sector legal. (Corrales et al, 2019).

CAPÍTULO III: MARCO METODOLÓGICO

Introducción capítulo III

En este capítulo se detallará el método de investigación del proyecto, el tipo de investigación y el enfoque, las fuentes de información y los sujetos de investigación, así como el diseño de la investigación de datos, las técnicas de recolección de datos, las variables de la investigación y la coherencia entre los objetivos, los elementos del marco teórico, los entregables y los instrumentos.

3.1 Tipo y enfoque de la investigación

3.1.1 Tipo de investigación

De acuerdo con Daen (2011), la investigación de campo se apoya en información del objeto de estudio o de los involucrados en él, a partir de indagación de campo como en la investigación.

Para este proyecto el tipo de investigación es una investigación de campo, dado que es un proyecto que trata de dar solución a un problema en un tiempo determinado, y la investigación se realiza en un ambiente natural donde está el objeto de estudio y los involucrados en él, que cumplen la función de ser fuente de información; en este caso, el ambiente natural donde se realizaría la investigación es el bufete de abogados Fonseca Sequeira y asociados.

3.1.2 Enfoque de la investigación

El enfoque cualitativo se enfoca en comprender las cualidades de un fenómeno. Su preocupación no es prioritariamente medir, si no cualificar y describir el fenómeno social a partir de los rasgos determinantes, según sean percibidos por los elementos mismos que están dentro de la situación estudiada. (Guerrero Bejarano, 2016).

Por lo que, con base a la definición antes descrita, se puede decir que el enfoque de esta investigación es cualitativo. Dado que para este proyecto no se aplicarán cuestionarios, sino que la información se obtendrá por medio de entrevistas y observación, y la prioridad no es medir algo sino cualificar el sistema a desarrollar.

3.2 Fuentes y sujetos de información

3.2.1 Fuentes de información

De acuerdo con Torres, Paz & Salazar (s.f.), las fuentes de información son todos aquellos medios de los cuales procede la información, que satisfacen las necesidades de conocimiento de una situación o problema presentado, que posteriormente será utilizado para lograr los objetivos esperados. De acuerdo a su origen se clasifican en:

- Fuentes primarias

- Fuentes secundarias

3.2.1.1 Fuentes primarias

Las fuentes primarias son aquellas en las que los datos provienen directamente de la población o muestra de la población. (Torres et al., s.f.).

Las fuentes primarias de información para este proyecto es la siguiente:

- Entrevistas con el dueño del bufete.

3.2.1.2 Fuentes secundarias

Las fuentes secundarias son aquellas que parten de datos pre-elaborados, como pueden ser datos obtenidos de anuarios estadísticos, de Internet, de medios de comunicación. (Torres et al., s.f.).

Las fuentes de información secundarias para este proyecto son las siguientes:

- Libros de temas relacionados a la investigación científica, análisis y diseño de sistemas de información, desarrollo web y términos jurídicos.
- Artículos de revistas publicados en internet.
- Documentación de plugins y tecnologías a utilizar en el proyecto.

3.2.2 Sujetos de información

Los sujetos de información son todas aquellas personas a las que se recurre para obtener la información específica y detallada importante para el proyecto.

En este caso, para este proyecto, el sujeto de información será el licenciado César Fonseca, abogado y dueño del bufete Fonseca Sequeira y asociados.

Puesto laboral o descripción general	Profesión u oficio	Experiencia	Relación con el tema
--------------------------------------	--------------------	-------------	----------------------

Abogado	Abogado	20 años	Dueño y fundador del bufete.
---------	---------	---------	------------------------------

Tabla 1. Sujetos de información.

Fuente: elaboración propia

3.3 Técnicas y herramientas de recolección de datos

Las técnicas de recolección de datos son mecanismos e instrumentos que se utilizan para reunir y medir información de forma organizada y con un objetivo específico. (Caro, 2021).

En esta sección se describen las técnicas y herramientas que se utilizarán en el proyecto para la recolección de datos.

3.3.1 Entrevista

Es una conversación generalmente entre 2 personas, (uno el entrevistador y otro el entrevistado). Las preguntas pueden ir registradas en una boleta que se llama cuestionario o bien se puede auxiliar de una grabadora para registrar los datos obtenidos. Cuando la entrevista y el cuestionario son utilizados en forma personal se le denomina: Face to Face (cara a cara). Cuando se posee una estructura fija de cuestionamientos o una secuencia de preguntas fijadas con anterioridad se conoce como entrevista dirigida. Cuando el entrevistador hace participar en un tema fijado anticipadamente, dejándole la iniciativa de la conversación y que toda su narración sea espontánea se conoce como entrevista no dirigida. La ventaja que tiene es que la gente generalmente responde cuando es confrontada en persona. El entrevistador puede notar reacciones específicas y eliminar malos entendidos sobre alguna pregunta hecha. Sus limitaciones a parte del costo, es el adiestramiento que tenga el entrevistador. Este es el método más completo para adquirir información por estar en contacto directo con la fuente. (Torres et al., s.f.).

Se realizarán entrevistas para obtener la información necesaria y tomar los requerimientos para el desarrollo del sistema web.

3.4 Variables de la investigación

Las variables de la investigación se detallan en la siguiente tabla:

Objetivos específicos	Variables asociadas	Descripción
1. Recolectar y analizar la información que permita determinar los requerimientos funcionales y no funcionales que se deben aplicar en el sistema.	<ul style="list-style-type: none"> • Entrevista al dueño del bufete Fonseca Sequeira. • Análisis de la información obtenida en la entrevista. 	Recolección de los requerimientos funcionales y no funcionales para el sistema web a desarrollar.
2. Diseñar los módulos que requiere el sistema web así como su base de datos, para guardar la información de los expedientes jurídicos, citas y recordatorios del bufete.	<ul style="list-style-type: none"> • Requerimientos funcionales y no funcionales. 	Creación de un diseño del sistema web que cumpla con los requerimientos obtenidos en la fase de análisis.
3. Desarrollar los módulos del sistema junto con sus respectivos mantenimientos y reportes de los datos ingresados en el sistema.	<ul style="list-style-type: none"> • Diseño del sistema web. 	Desarrollo del sistema en cuestión con base en el diseño creado y cumpliendo con los requerimientos obtenidos de la fase de análisis.
4. Implementar el sistema web desarrollado para la gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados.	<ul style="list-style-type: none"> • Hospedaje web. • Desarrollo del proyecto. 	Implementación del sistema web en el hospedaje correspondiente.

Tabla 2. Variables de la investigación.

Fuente: elaboración propia

3.5 Diseño de la investigación

A continuación se presentan las diferentes fases que componen el proyecto, así como la explicación de los mismos.



Figura 18. Diseño de la investigación

Fuente: elaboración propia

Etapa	Descripción
Etapa 1: Análisis	Con la ayuda de técnicas de recolección de datos como entrevistas se obtiene la información necesaria para la creación y análisis de los requerimientos necesarios para el sistema.
Etapa 2: Diseño	Se diseñan los módulos a desarrollar del sistema web, así como su base de datos, con base en la información recolectada y los requerimientos generados en la etapa de análisis.
Etapa 3: Desarrollo	Utilizando como base el diseño del sistema web generado en la etapa de diseño, se procede a desarrollar el sistema.

Etapa 4: Implementación	Se lleva a cabo la implementación del sistema web.
-------------------------	--

Tabla 3. Diseño de la investigación.

Fuente elaboración propia

3.6 Matriz de coherencia

Objetivo	Entregable	Fase, parte o etapa de la metodología del proyecto que posibilita la realización del entregable	Técnicas/ métodos de recolección de la información	Instrumentos	Temas relacionados para marco teórico
1. Recolectar y analizar la información que permita determinar los requerimientos funcionales y no funcionales que se deben	Entregable 1: lista de requerimientos del sistema, donde se permita conocer qué debe realizar el mismo.	Etapa 1	Entrevista. Análisis de documentación	Reuniones	2.1 Ámbito legal 2.1.1 Caso 2.1.2 Expediente 2.1.3 Despacho jurídico 2.1.4 Partes 2.1.5 Representante legal 2.1.6 Notificación

<p>aplicar en el sistema.</p>					<p>2.1.7 Apelación 2.2 Ámbito informático 2.2.1 Software 2.2.1.1 Sistema de gestión 2.2.1.1.1 Gestión 2.2.1.3 Análisis del proyecto 2.2.1.3.1 Historia de Usuario 2.2.1.3.2 Módulo 2.2.1.3.3 Requerimientos 2.2.1.3.3.1 Funcionales 2.2.1.3.3.2 No funcionales 2.2.1.3.4 Características de</p>
-------------------------------	--	--	--	--	---

					<p>requerimientos</p> <p>2.3</p> <p>Tecnología legal</p> <p>2.3.1</p> <p>Intercibernética</p> <p>2.3.1.1</p> <p>Informática jurídica</p> <p>2.3.2</p> <p>Legaltech</p>
<p>2. Diseñar los módulos que requiere el sistema web así como su base de datos, para guardar la información de los expedientes jurídicos, citas y recordatorios del bufete.</p>	<p>Entregable 2: casos de usos donde se explique el funcionamiento del sistema y los actores que lo utilizarán, así como distintos diagramas UML, esto con el propósito de apreciar una</p>	<p>Etapa 2</p>	<p>Técnicas de diseño de sistemas, para la realización de un diseño que concuerde con los requisitos recolectados.</p>	<p>Requerimientos funcionales y no funcionales.</p>	<p>2.2.1.1.2</p> <p>Sistema de información</p> <p>2.2.1.1.2.1</p> <p>Ciclo de vida del software</p> <p>2.2.1.1.3</p> <p>Metodologías de programación</p> <p>2.2.1.1.3.1</p> <p>Modelo de cascada</p> <p>2.2.1.2</p> <p>Internet</p> <p>2.2.1.2.1</p> <p>Web</p>

	<p>forma más fácil para entender que realizará el mismo, de igual manera se diseña la base de datos.</p>			<p>2.2.1.2.3 Arquitecturas de software 2.2.1.2.3.1 Cliente – servidor 2.2.1.2.4 Aplicaciones web 2.2.1.4 Diseño del proyecto 2.2.1.4.1 Diagrama 2.2.1.4.2 UML 2.2.1.4.2.1 Diagrama de casos de uso 2.2.1.4.2.2 Diagrama de clases 2.2.1.4.2.3 Diagrama de secuencia 2.2.1.4.2.4 Diagrama de actividad 2.2.1.4.3 Diagrama</p>
--	--	--	--	--

					entidad- relación
3. Desarrollar los módulos del sistema junto con sus respectivos mantenimientos y reportes de los datos ingresados en el sistema.	Entregable 3: código desarrollado del sistema. Este entregable tendrá el sistema desarrollado, con los siguientes módulos: Clientes, Expedientes, Agenda, Reportes y Configuraciones.	Etapas 3	Técnicas de desarrollo de software, para la creación de un sistema que concuerde con los requisitos recolectados y el diseño realizado.	Requerimientos funcionales y no funcionales. Diagramas UML, casos de uso, historias de usuario y diseño de base de datos.	2.2.1.5 Desarrollo del proyecto 2.2.1.5.1 Base de datos 2.2.1.5.1.1 Tabla 2.2.1.5.1.2 Relación 2.2.1.5.2 Microsoft SQL Server 2.2.1.5.3 Lenguaje de programación 2.2.1.5.3.1 HTML 2.2.1.5.3.2 CSS 2.2.1.5.3.3 Javascript 2.2.1.5.3.4 jQuery 2.2.1.5.3.5 Bootstrap 2.2.1.5.3.6 Owl Carousel 2.2.1.5.3.7 C#

					2.2.1.5.3.8 Microsoft Visual Studio 2.2.1.5.3.9 Modelo Vista Controlador 2.2.1.5.3.10 Razor 2.2.1.5.3.11 Entity Framework 2.2.1.5.3.12 LINQ 2.2.1.5.3.13 ASP.NET MVC 2.2.1.5.4 Pruebas
4. Implementar el sistema web desarrollado para la gestión de expedientes jurídicos, citas y recordatorios para el bufete	Entregable 4: implementación de la aplicación web para la gestión de expedientes jurídicos, citas y recordatorios para el bufete	Etapa 4	Hospedaje web del sistema desarrollado y de su base de datos.	Sistema desarrollado.	2.2.1.2.2 Internet Information Services (IIS) 2.2.1.2.6 Host 2.2.1.2.7 Dominio 2.2.1.6 Implementación del proyecto

Fonseca Sequeira y asociados.	Fonseca Sequeira y asociados. Este entregable tiene como resultado una aplicación completamente funcional al que el abogado pueda ingresar a través de una computadora a con acceso a internet.				
-------------------------------	--	--	--	--	--

Tabla 4. Matriz de coherencia.

Fuente: elaboración propia

CAPÍTULO IV: DIAGNÓSTICO

Introducción capítulo IV

Este capítulo es el diagnóstico de la situación. En este capítulo se describe la situación con respecto al área administrativa de la empresa, con respecto al área técnica y sobre la percepción, para luego poder definir las brechas es decir lo que hay actualmente con lo que se desea alcanzar, estas son las conclusiones del diagnóstico.

4.1 Descripción de la situación actual

4.1.1 Diagnóstico administrativo u operativo

Como se ha establecido anteriormente, el licenciado César Fonseca es el único abogado y el encargado de todos los aspectos en el bufete. Él se encarga de todos los aspectos administrativos, atiende a los clientes y se encarga de encargarse de la comunicación con ellos continuamente.

Con respecto a la gestión de los expedientes, el bufete maneja los expedientes únicamente de forma física, por lo que, dada la situación de que el bufete tiene oficinas en distintos lugares, el traslado de forma segura e íntegra de estos expedientes puede ser difícil debido a muchos factores como el clima, la lluvia, accidentes fuera del control del abogado, etc.

De acuerdo con el licenciado, al ver que estos expedientes podrían sufrir daños por el constante traslado, se decide empezar a utilizar un documento de Microsoft Excel para la gestión de los expedientes de los clientes, pero esta solución no fue la más adecuada para el bufete, pues no se adecuaba a las necesidades del mismo. Debido a que esta solución no fue la adecuada, decidió dejar de utilizarlo y eliminar el archivo.

De acuerdo al abogado, el proceso operativo que se realiza en cada cita consta de seis pasos: el primero, que sería el inicio y saludo; el segundo paso es cuando el cliente comienza a relatar y explicar la situación y la razón por la que está allí; el tercer paso es realizado por el abogado, en este paso el abogado le brinda la asesoría al cliente sobre las cosas que puede hacer y las que no basándose en la ley y utilizando códigos como marco de referencia para la asesoría; si el cliente está de acuerdo con lo que sugiere el abogado se procede al paso cuatro que es dar el precio e iniciar la negociación, y si todo va bien se establece las formas de pago; el paso cinco es la solicitud de evidencias o pruebas para el caso; y por último, el paso seis que consta de la redacción de escritos ya sea demanda, defensa o lo que solicite el caso.

La gestión de la agenda de citas y recordatorios la solía manejar con una agenda física donde escribía los días de los juicios y la hora, el día y la hora en que un cliente llegaba a la oficina, o algún recordatorio de algo que debía hacer como ir a preguntar por algún proceso, ir a hablar con algún testigo, etc. Actualmente decidió empezar a llevar estas cosas con el calendario

de google que está integrado en su teléfono, sin embargo aún mantiene una pizarra en su oficina con fechas importantes que debe recordar.

Las siguientes imágenes son fotografías tomadas de las agendas del 2018 y 2019 del abogado, y de la pizarra que actualmente tiene.

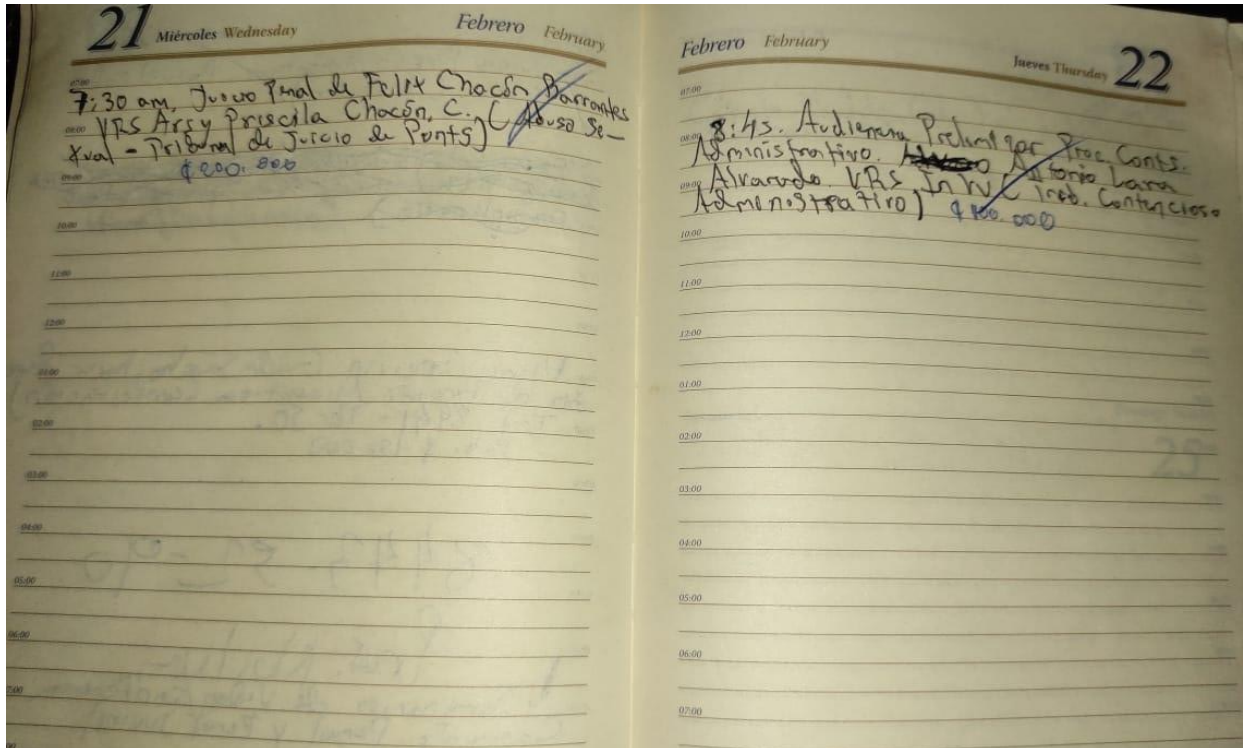


Figura 19. Agenda del abogado, 2018.

Fuente: Bufete Fonseca Sequeira y asociados.

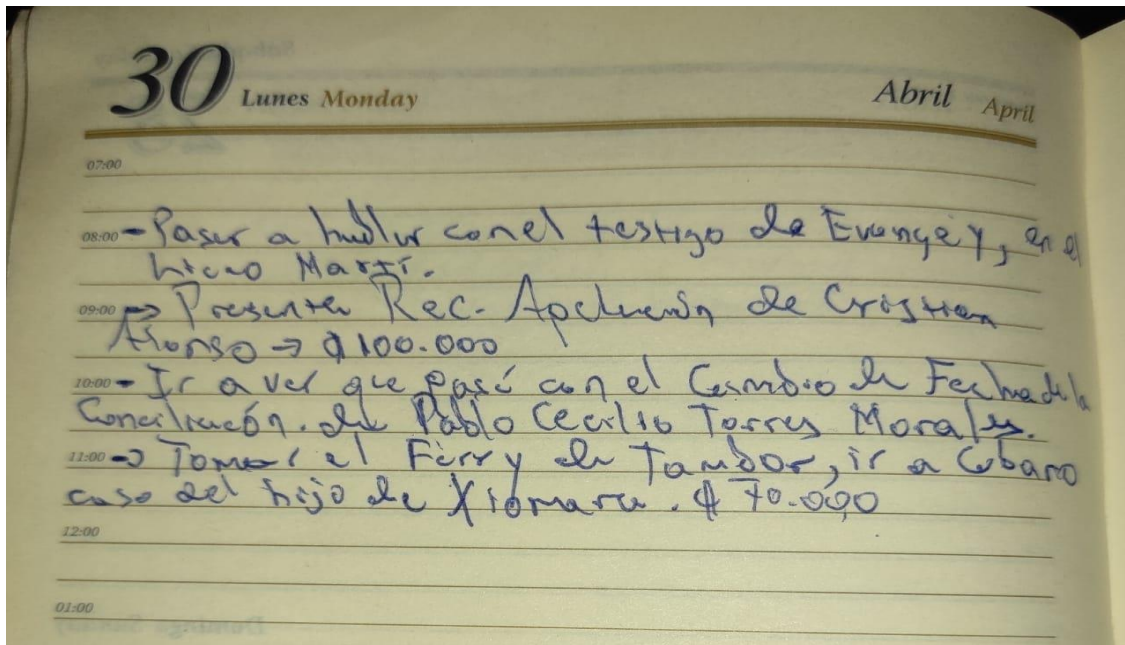


Figura 20. Agenda del abogado, 2018.

Fuente: Bufete Fonseca Sequeira y asociados.

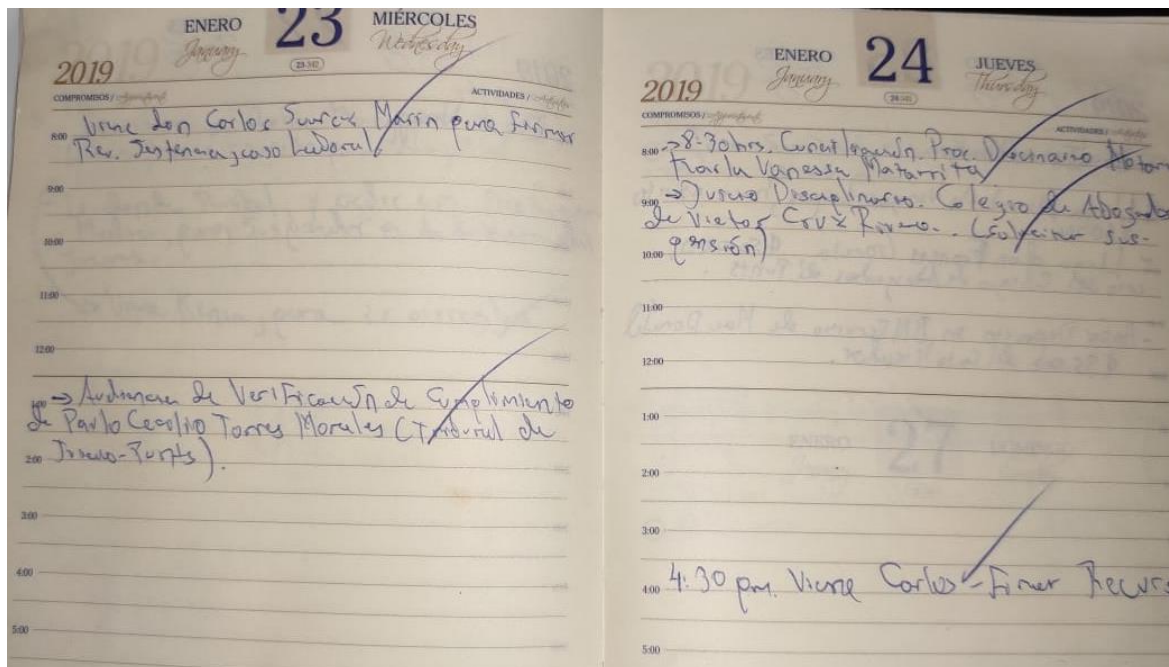


Figura 21. Agenda del abogado, 2019.

Fuente: Bufete Fonseca Sequeira y asociados.

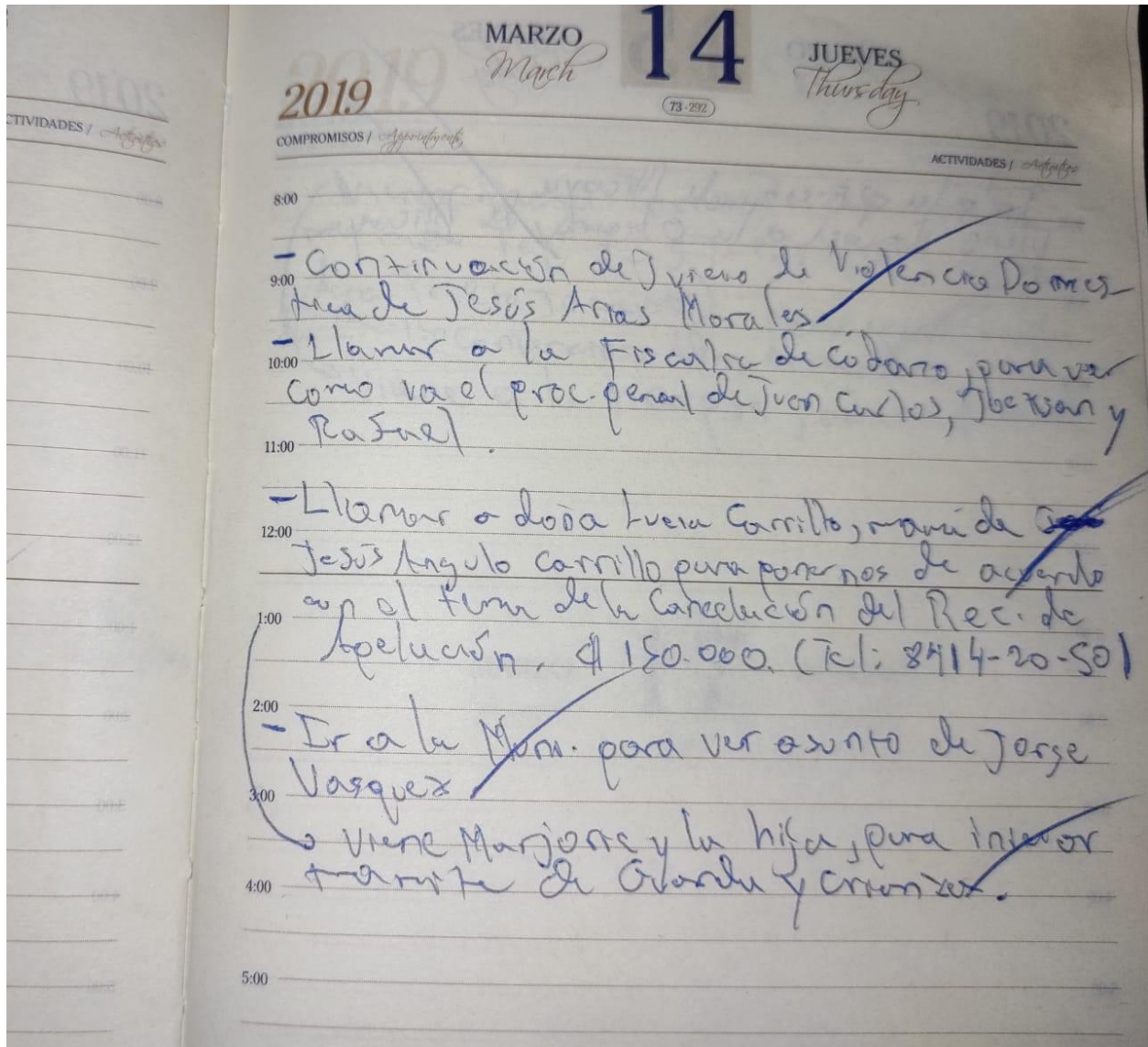


Figura 22. Agenda del abogado, 2019.

Fuente: Bufete Fonseca Sequeira y asociados.

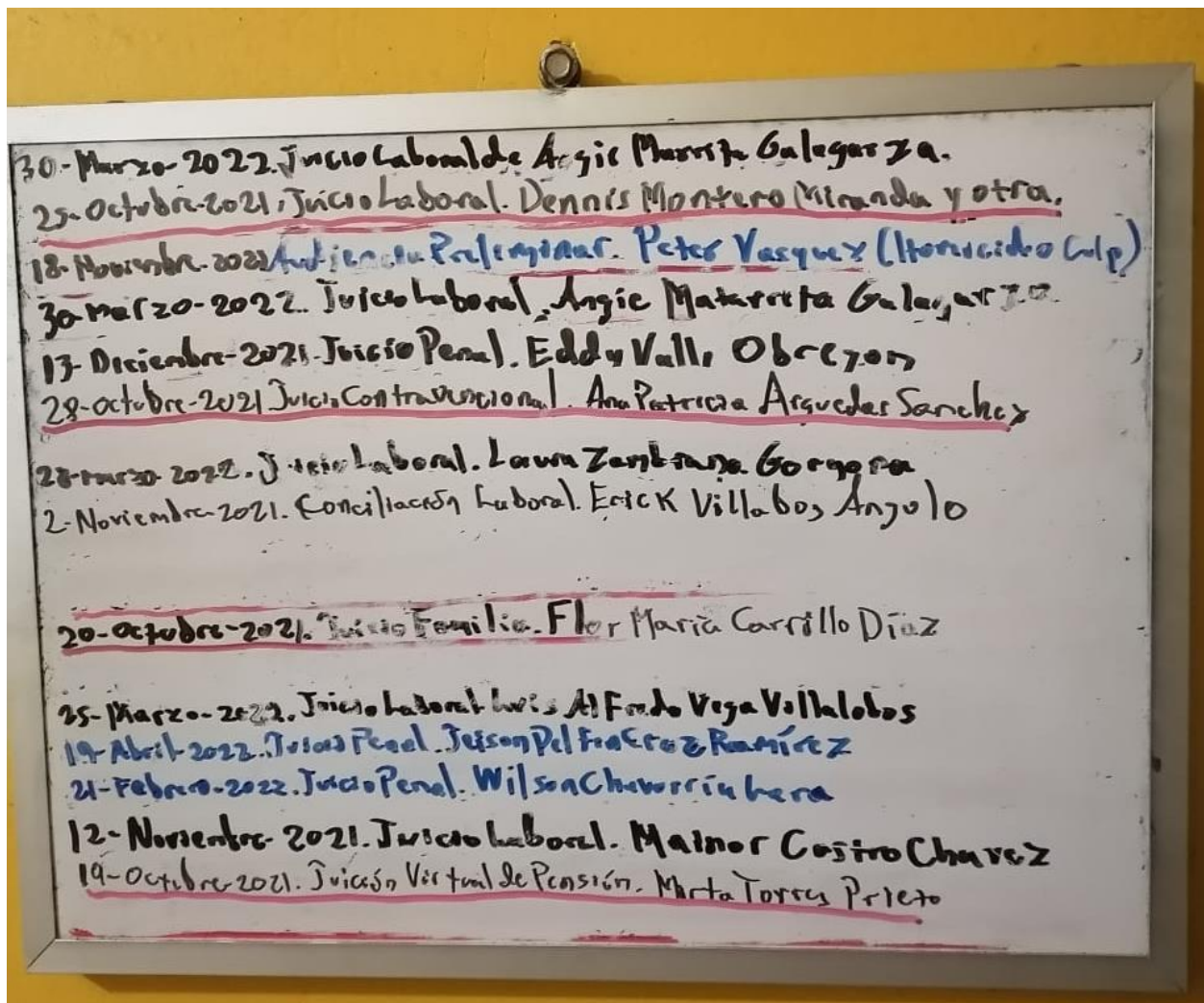


Figura 23. Pizarra en oficina del abogado.

Fuente: Bufete Fonseca Sequeira y asociados.

4.1.2 Diagnóstico técnico

El bufete Fonseca Sequeira y asociados no cuenta con una infraestructura de TI robusta, dado que no ha sido necesaria pues sus registros se manejan en físico.

Cuenta con una computadora portátil marca Toshiba, modelo Satellite C55-C, posee un Intel® Pentium® CPU N3700 de 1.60 GHz, 1601 Mhz, de memoria RAM cuenta con 4,00GB y de memoria virtual cuenta con 5,73GB. Tiene un tipo de sistema basado en x64, y como sistema operativo utiliza Microsoft Windows 8.1. Su disco duro es de 453GB.

Esta computadora portátil es la que traslada a las distintas oficinas que tiene, esta computadora tiene la capacidad de ser utilizada para el uso regular de una computadora pero no para almacenar un sistema de escritorio robusto para la gestión de expedientes.

Con respecto al internet, la oficina principal, ubicada en Puntarenas Centro, tiene acceso a internet por medio de conexión inalámbrica (WiFi), el servicio es brindado por la compañía Cabletica, y la velocidad es de 30Mb. Para el otro establecimiento utiliza el internet que brinda el plan que posee su celular, dado que solo va allí un día de la semana.

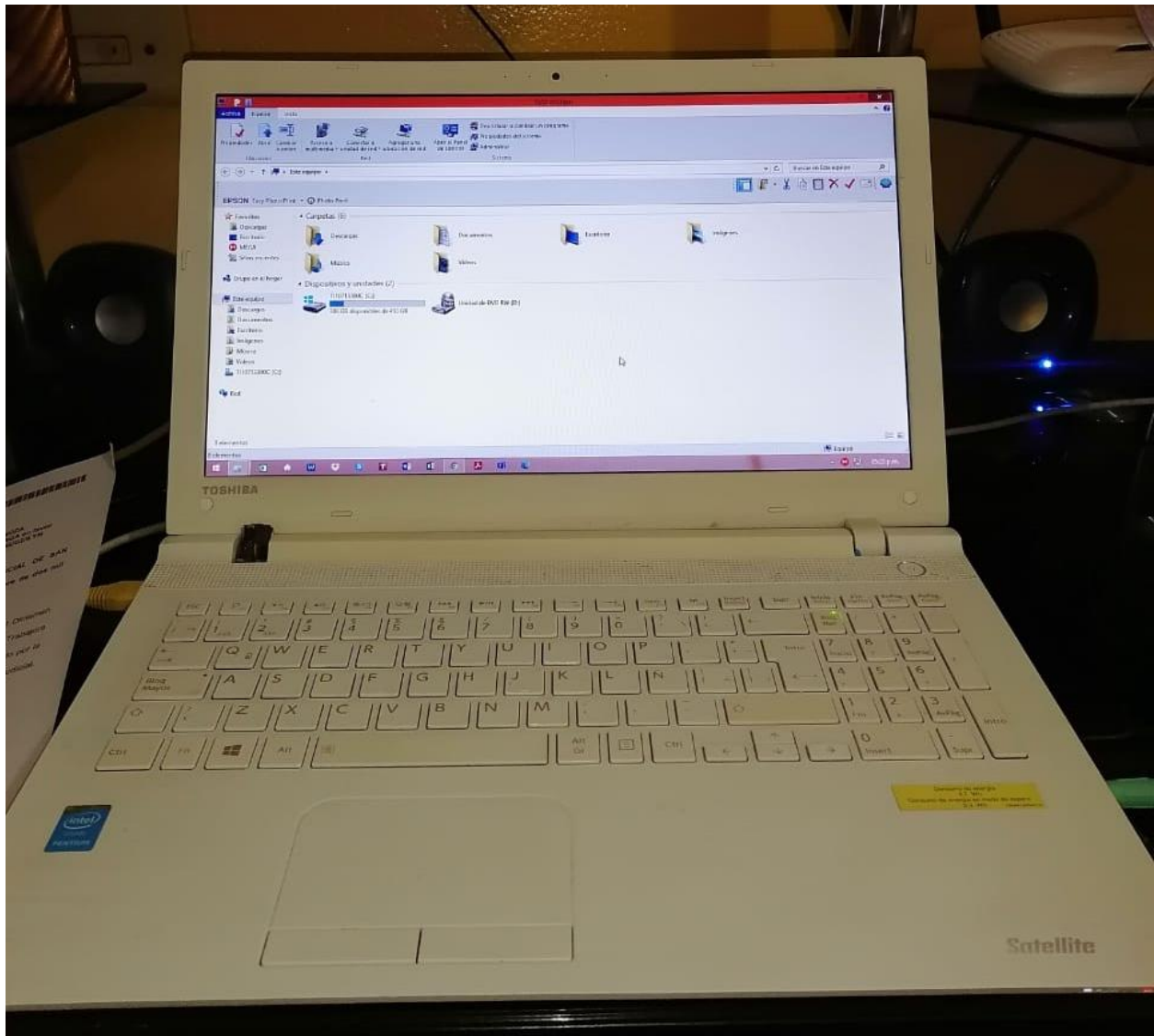


Figura 24. Computador de trabajo

Fuente: Bufete Fonseca Sequeira y asociados.



Figura 25. Procesador del computador.

Fuente: Bufete Fonseca Sequeira y asociados.

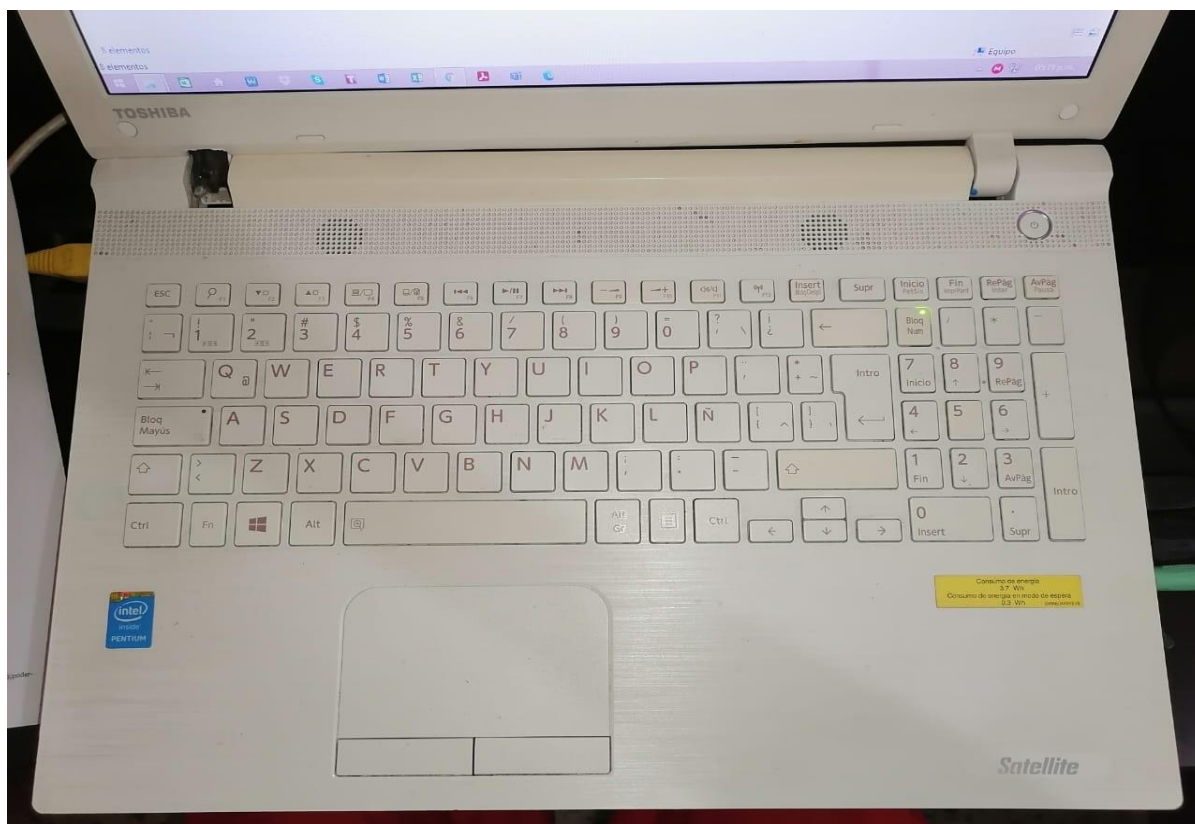


Figura 26. Computador del abogado

Fuente: Bufete Fonseca Sequeira y asociados.

4.1.3 Diagnóstico de percepción

Las actividades realizadas fueron entrevistas con el licenciado César Fonseca, dueño y abogado del bufete Fonseca Sequeira y asociados. En estas entrevistas el abogado expresó sus limitaciones y la necesidad del sistema en cuestión, así como todos los detalles que se necesitan para que el sistema funcione y sea de utilidad para su empresa.

4.2 Determinación de brechas y conclusiones

Situación actual	Brecha	Conclusiones
- Los expedientes se mantienen y	- No se cuenta con un sistema de gestión de expedientes virtual.	- Se requiere la creación de un sistema web de gestión de

<p>administran de forma física.</p>		<p>expedientes jurídicos para mantener la integridad de los expedientes en físico.</p>
<ul style="list-style-type: none"> - Se intentó implementar la gestión de expedientes por medio de un documento de Microsoft Excel. 	<ul style="list-style-type: none"> - No se cuenta con un sistema de gestión de expedientes que se adapte a las necesidades del bufete. 	<ul style="list-style-type: none"> - Se requiere la creación de un sistema web de gestión de expedientes jurídicos que permita mantener la integridad de los expedientes y que se adecúe a las necesidades del bufete.
<ul style="list-style-type: none"> - Todo lo relacionado con agendar citas, re agendar y/o atender al cliente es gestionado por el abogado y se registra de forma física. 	<ul style="list-style-type: none"> - No se cuenta con un sistema de gestión de citas para llevar un control más ordenado de las mismas. 	<ul style="list-style-type: none"> - Se requiere la creación de un sistema web de gestión de expedientes jurídicos, citas y recordatorios que permita mantener un mayor orden en el registro de las citas y recordatorios y sus modificaciones.
<ul style="list-style-type: none"> - El equipo técnico que posee el bufete es una computadora portátil con poca capacidad para almacenar un sistema de escritorio robusto para la gestión 	<ul style="list-style-type: none"> - No se cuenta con un sistema web de gestión de expedientes para la gestión de expedientes, citas y recordatorios, para poder utilizar. 	<ul style="list-style-type: none"> - Se requiere la creación de un sistema web de gestión de expedientes jurídicos, citas y recordatorios, dado a la facilidad de acceso a internet y a la

de sus expedientes, y conexión a internet en cada oficina.		poca capacidad de la computadora para almacenar un sistema de escritorio.
--	--	--

Tabla 5. Determinación de brechas del diagnóstico.

Fuente: elaboración propia.

CAPÍTULO V: PROPUESTA DEL PROYECTO

Introducción capítulo V

Este capítulo presenta de forma detallada tanto la propuesta como el desarrollo del proyecto, el capítulo estará dividido por secciones las cuales serán: análisis, diseño, desarrollo e implementación, que son las fases que posee el proyecto, tal como se define en el capítulo III.

La sección del análisis del proyecto presenta las especificaciones que expresa el cliente con respecto al proyecto a desarrollar en forma de requerimientos que se dividen en requerimientos funcionales y requerimientos no funcionales; además, también adjunta historias de usuario que explican de forma sencilla y en palabras que el cliente entiende, una función del software.

La sección del diseño del proyecto expone la secuencia de funcionamiento del software, desde con qué partes del sistema pueden interactuar los distintos usuarios del sistema y la secuencia que seguirán estas interacciones, hasta la estructura de las clases y base de datos.

La sección de desarrollo muestra el desarrollo de la base de datos y su composición, así como el proyecto en desarrollo y sus pantallas.

Y, por último, la sección de la implementación, que detalla la información relacionada a la implementación y el hosting web.

5.1 Análisis del proyecto

En esta sección del capítulo se presentarán los requerimientos funcionales y no funcionales que se lograron recolectar como resultado de las entrevistas realizadas con el abogado dueño del bufete.

5.1.1 Requerimientos funcionales

Los requerimientos funcionales se encargan de especificar el comportamiento que debe tener el sistema bajo ciertas condiciones. A continuación se presentan los requerimientos funcionales que el sistema web de gestión de expedientes jurídicos, citas y recordatorios debe poseer.

Requerimiento funcional 01: Iniciar Sesión

RQF01	<i>Iniciar sesión</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Inicio del sistema
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá tener un formulario de inicio de sesión que valide si el usuario está registrado en la base de datos para poder darle acceso al sistema.
Dependencias	<ul style="list-style-type: none">• <i>RQF03 crear usuario</i>• <i>RQF02 consultar usuario</i>
Actores	Usuarios del bufete Fonseca Sequeira y asociados.
Importancia/Prioridad	Media
Comentarios	

Tabla 6. Requerimiento funcional 01 Iniciar sesión.

Fuente: elaboración propia.

Requerimiento funcional 02: Consultar usuario

RQF02	<i>Consultar usuario</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Inicio del sistema
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá consultar en la base de datos si el usuario ingresado está registrado.
Dependencias	
Actores	Usuario del sistema de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados
Importancia/Prioridad	Alta
Comentarios	

Tabla 7.Requerimiento funcional 02, Consultar usuario.

Fuente: elaboración propia.

Requerimiento funcional 03: Crear usuario

RQF03	<i>Crear usuario</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Mantenimiento
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá crear un usuario con un rol específico para poder usar el sistema.
Dependencias	<ul style="list-style-type: none"> • <i>RQF02 Consultar usuario</i> • <i>RQF01 Iniciar sesión</i> • <i>RQF33 Enviar email al nuevo usuario</i>

Actores	Usuario con rol de administrador en el sistema de gestión de expedientes jurídicos, citas y recordatorios del bufete Fonseca Sequeira y asociados.
Importancia/Prioridad	Media
Comentarios	

Tabla 8. Requerimiento funcional 03, Crear usuario.

Fuente: elaboración propia.

Requerimiento funcional 04: Modificar usuario

RQF04	<i>Modificar usuario</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Mantenimiento
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá permitir al usuario cambiar sus datos registrados en la base de datos, tal como sus números de teléfono o correo electrónicos.
Dependencias	<ul style="list-style-type: none"> • <i>RQF02 Consultar usuario</i> • <i>RQF01 Iniciar sesión</i>
Actores	Usuario del sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Baja
Comentarios	

Tabla 9. Requerimiento funcional 04, Modificar usuario.

Fuente: elaboración propia.

Requerimiento funcional 05: Eliminar usuario

RQF05	<i>Eliminar usuario</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Mantenimiento
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá permitir al usuario administrador eliminar un usuario del sistema.
Dependencias	<ul style="list-style-type: none"> • <i>RQF02 Consultar usuario</i> • <i>RQF01 Iniciar sesión</i>
Actores	Usuario con rol de administrador en el sistema de gestión de expedientes jurídicos, citas y recordatorios
Importancia/Prioridad	Baja
Comentarios	

Tabla 10. Requerimiento funcional 05, Eliminar usuario.

Fuente: elaboración propia.

Requerimiento funcional 06: Recuperar contraseña

RQF06	<i>Recuperar contraseña</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Inicio del sistema.
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema solicitará el email registrado del usuario que desea cambiar la contraseña, validará si existe, y, si sí existe entonces envía un email con una nueva contraseña.
Dependencias	<ul style="list-style-type: none"> • <i>RQF02 Consultar usuario.</i> • <i>RQF32 Enviar email con nueva contraseña.</i>

Actores	Usuario del sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Baja
Comentarios	

Tabla 11. Requerimiento funcional 05, Cambiar contraseña.

Fuente: elaboración propia.

Requerimiento funcional 07: Agregar tipo de proceso

RQF07	<i>Agregar tipo de proceso</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Mantenimiento
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá agregar los tipos de procesos que el abogado desee.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF30 Consultar tipo de proceso</i>
Actores	Usuario con rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios
Importancia/Prioridad	Media
Comentarios	

Tabla 12. Requerimiento funcional 07, Agregar tipo de proceso.

Fuente: elaboración propia.

Requerimiento funcional 08: Eliminar tipo de proceso

RQF08	<i>Eliminar tipo de proceso</i>
Creado por	Angie López Villegas

Versión	1 (19/08/2021)
Módulo	Mantenimiento
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá permitir eliminar los tipos de procesos que el abogado desee.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF30 Consultar tipo de proceso</i>
Actores	Usuario del sistema con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Baja
Comentarios	

Tabla 13. Requerimiento funcional 08, Eliminar tipo de proceso.

Fuente: elaboración propia.

Requerimiento funcional 09: Consultar cliente

RQF09	<i>Consultar cliente</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Clientes
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá buscar en la base de datos el usuario que el abogado desea buscar, si el cliente está registrado el sistema deberá mostrar la información relacionada a él, como las citas que ha tenido, expedientes relacionados, etc.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i>
Actores	Usuario con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios

Importancia/Prioridad	Media
Comentarios	

Tabla 14. Requerimiento funcional 09, Consultar cliente.

Fuente: elaboración propia.

Requerimiento funcional 10: Agregar cliente

RQF10	<i>Agregar cliente</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Clientes
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá agregar un cliente nuevo que se asigne al abogado que lo agrega, y envía un email al usuario agregado con información para iniciar sesión.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF09 Consultar cliente</i>
Actores	Usuario con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios
Importancia/Prioridad	Alta
Comentarios	

Tabla 15. Requerimiento funcional 10, Agregar cliente.

Fuente: elaboración propia.

Requerimiento funcional 11: Modificar cliente

RQF11	<i>Modificar cliente</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)

Módulo	Cliente
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema luego de haber realizado la consulta sobre el cliente y con la información desplegada deberá permitir al abogado cambiar información del cliente, como números de teléfono o correos electrónicos.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF09 Consultar cliente</i>
Actores	Usuario con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios
Importancia/Prioridad	Alta
Comentarios	

Tabla 16. Requerimiento funcional 11, Modificar cliente.

Fuente: elaboración propia.

Requerimiento funcional 12: Eliminar cliente

RQF12	<i>Eliminar cliente</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Cientes
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá dar la opción de inactivar el cliente. El cliente no será eliminado de la base de datos, pero sí cambiará el estado del cliente a inactivo.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF09 Consultar cliente</i>

Actores	Usuario con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios
Importancia/Prioridad	Baja
Comentarios	

Tabla 17. Requerimiento funcional 12, Eliminar cliente.

Fuente: elaboración propia.

Requerimiento funcional 13: Re direccionar a expediente

RQF13	<i>Re direccionar a expediente</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Clientes
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá, luego de realizar la consulta del cliente y de desplegar la información relacionada al mismo, si existe algún expediente relacionado entonces este deberá dar la opción de que al darle click, lo re direcciona al módulo de expedientes para poder realizar las modificaciones necesarias.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF09 Consultar cliente</i>
Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios
Importancia/Prioridad	Baja
Comentarios	

Tabla 18. Requerimiento funcional 13, Re direccionar a expediente.

Fuente: elaboración propia.

Requerimiento funcional 14: Consultar expediente

RQF14	<i>Consultar expediente</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Expedientes
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá verificar en la base de datos que el expediente está registrado, para poder realizar modificaciones o crear uno nuevo
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i>
Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios
Importancia/Prioridad	Media
Comentarios	

Tabla 19. Requerimiento funcional 14, Consultar expediente.

Fuente: elaboración propia.

Requerimiento funcional 15: Crear expediente

RQF15	<i>Crear expediente</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Expedientes
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá permitir crear un expediente al usuario
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF14 Consultar expediente</i>

Actores	Usuarios con el rol de abogados en el sistema de gestión de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Alta
Comentarios	

Tabla 20. Requerimiento funcional 15, Crear expediente.

Fuente: elaboración propia.

Requerimiento funcional 16: Modificar expediente

RQF16	<i>Modificar expediente</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Expedientes
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá, luego de realizar la consulta para verificar que el expediente existe y extraer la información, permitir realizar cambios y agregar a los datos del mismo.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF014 Consultar expediente</i>
Actores	Usuarios con el rol de abogados en el sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Alta
Comentarios	

Tabla 21. Requerimiento funcional 16, Modificar expediente.

Fuente: elaboración propia.

Requerimiento funcional 17: Agregar notificación

RQF17	<i>Agregar notificación</i>
--------------	------------------------------------

Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Expedientes
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá permitir agregar una nueva notificación relacionada a un expediente que fue previamente consultado, sin la necesidad de realizar una modificación completa del expediente.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF14 Consultar expediente</i>
Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Media
Comentarios	

Tabla 22. Requerimiento funcional 17, Agregar notificación.

Fuente: elaboración propia.

Requerimiento funcional 18: Consultar cita

RQF18	<i>Consultar cita</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Agenda
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá generar una consulta de las citas agendadas y desplegarlas en el módulo.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i>
Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.

Importancia/Prioridad	Baja
Comentarios	

Tabla 23. Requerimiento funcional 18, Consultar cita.

Fuente: elaboración propia.

Requerimiento funcional 19: Agregar cita

RQF19	<i>Agregar cita</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Agenda
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá tener una opción para crear una cita con un cliente, que se almacene en la agenda del abogado.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i>
Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Alta
Comentarios	

Tabla 24. Requerimiento funcional 19, Agregar cita.

Fuente: elaboración propia.

Requerimiento funcional 20: Modificar cita

RQF20	<i>Modificar cita</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Agenda

Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá tener la opción de modificar una cita que ya haya sido agregada.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF18 Consultar cita</i>
Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Media
Comentarios	

Tabla 25. Requerimiento funcional 20, Modificar cita.

Fuente: elaboración propia.

Requerimiento funcional 21: Cancelar cita

RQF21	<i>Cancelar cita</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Agenda
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá dar la posibilidad de cancelar la cita registrada.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF18 Consultar cita</i>
Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Media
Comentarios	

Tabla 26. Requerimiento funcional 21, Cancelar cita.

Fuente: elaboración propia.

Requerimiento funcional 22: Consultar recordatorio

RQF22	<i>Consultar recordatorio</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Agenda
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá generar una consulta de los recordatorios agregados y desplegarlos en el módulo.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i>
Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Baja
Comentarios	

Tabla 27. Requerimiento funcional 22, Consultar recordatorio.

Fuente: elaboración propia.

Requerimiento funcional 23: Agregar recordatorio

RQF23	<i>Agregar recordatorio</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Agenda
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá tener la opción de agregar un nuevo recordatorio.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i>

Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Alta
Comentarios	

Tabla 28. Requerimiento funcional 23, Agregar recordatorio.

Fuente: elaboración propia.

Requerimiento funcional 24: Modificar recordatorio

RQF24	<i>Modificar recordatorio</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Agenda
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá dar la posibilidad de modificar un recordatorio ya registrado y que está desplegado en el módulo.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF22 Consultar recordatorio</i>
Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Media
Comentarios	

Tabla 29. Requerimiento funcional 24, Modificar recordatorio.

Fuente: elaboración propia.

Requerimiento funcional 25: Eliminar recordatorio

RQF25	<i>Eliminar recordatorio</i>
Creado por	Angie López Villegas

Versión	1 (19/08/2021)
Módulo	Agenda
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá dar la posibilidad de eliminar un recordatorio ya registrado y que está desplegado en el módulo.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i> • <i>RQF22 Consultar recordatorio</i>
Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Media
Comentarios	

Tabla 30. Requerimiento funcional 25, Eliminar recordatorio.

Fuente: elaboración propia.

Requerimiento funcional 26: Crear reporte

RQF26	<i>Crear reporte</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Reportes
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá tener la posibilidad de permitir al abogado generar un reporte con base en las siguientes opciones: generar reportes de un cliente, de un expediente, de correos enviados a un cliente o de notificaciones relacionadas a un expediente.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i>
Actores	Usuarios con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.

Importancia/Prioridad	Media
Comentarios	

Tabla 31. Requerimiento funcional 26, Crear reporte.

Fuente: elaboración propia.

Requerimiento funcional 27: Enviar email de actualización para notificaciones del expediente

RQF27	<i>Enviar email de actualización para notificaciones del expediente</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Expedientes
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	Al agregar una nueva notificación al expediente, el sistema deberá enviar automáticamente un email al cliente con la información de la notificación.
Dependencias	<ul style="list-style-type: none"> • <i>RQF17 Agregar notificación</i>
Actores	
Importancia/Prioridad	Media
Comentarios	

Tabla 32. Requerimiento funcional 27, Enviar email de actualización para notificaciones del expediente.

Fuente: elaboración propia.

Requerimiento funcional 28: Enviar email de cita

RQF28	<i>Enviar email de cita</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Agenda

Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema deberá enviar automáticamente un email al cliente con la información de la cita que se ha generado o sobre los cambios que se realicen sobre ella.
Dependencias	<ul style="list-style-type: none"> • <i>RQF19 Agregar cita</i> • <i>RQF20 Modificar cita</i> • <i>RQF21 Cancelar cita</i>
Actores	
Importancia/Prioridad	Media
Comentarios	

Tabla 33. Requerimiento funcional 28, Enviar email de cita.

Fuente: elaboración propia.

Requerimiento funcional 29: Roles de usuario

RQF29	<i>Roles de usuario</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Módulo	Todo el sistema
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema tendrá dos tipos de usuarios: abogados y administradores.
Dependencias	
Actores	
Importancia/Prioridad	Media
Comentarios	

Tabla 34. Requerimiento funcional 29, Roles de usuario.

Fuente: elaboración propia.

Requerimiento funcional 30: Consultar tipo de proceso

RQF30	<i>Consultar tipo de proceso</i>
Creado por	Angie López Villegas
Versión	1 (11/10/2021)
Módulo	Mantenimiento
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema consultará automáticamente los tipos de procesos existentes en el sistema.
Dependencias	<ul style="list-style-type: none"> • <i>RQF01 Iniciar sesión</i>
Actores	Usuario con el rol de abogado en el sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Media
Comentarios	

Tabla 35. Requerimiento funcional 30, Consultar tipo de proceso.

Fuente: elaboración propia.

Requerimiento funcional 31: Cambiar contraseña

RQF31	<i>Cambiar contraseña</i>
Creado por	Angie López Villegas
Versión	1 (11/10/2021)
Módulo	Mantenimiento.
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El usuario digitará la contraseña actual y la nueva contraseña, el sistema verificará que la contraseña actual concuerde con la que está almacenada en la base de datos.

Dependencias	<ul style="list-style-type: none"> • <i>RQF02 Consultar usuario</i>
Actores	Usuario del sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Baja
Comentarios	

Tabla 36. Requerimiento funcional 31, Cambiar contraseña.

Fuente: elaboración propia.

Requerimiento funcional 32: Enviar email con nueva contraseña

RQF32	<i>Enviar email con nueva contraseña</i>
Creado por	Angie López Villegas
Versión	1 (11/10/2021)
Módulo	Inicio del sistema.
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El usuario enviará un email automáticamente al usuario que solicitó la recuperación de contraseña.
Dependencias	<ul style="list-style-type: none"> • <i>RQF02 Consultar usuario</i>
Actores	
Importancia/Prioridad	Baja
Comentarios	

Tabla 37. Requerimiento funcional 32, Enviar email con nueva contraseña.

Fuente: elaboración propia.

Requerimiento funcional 33: Enviar email al nuevo usuario

RQF33	<i>Enviar email al nuevo usuario</i>
Creado por	Angie López Villegas
Versión	1 (11/10/2021)

Módulo	Inicio del sistema.
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema enviará un email al usuario agregado al sistema con la información necesaria para el inicio de sesión.
Dependencias	<ul style="list-style-type: none"> • <i>RQF02 Consultar usuario</i>
Actores	
Importancia/Prioridad	Baja
Comentarios	

Tabla 38. Requerimiento funcional 33, Enviar email al nuevo usuario.

Fuente: elaboración propia.

Requerimiento funcional 34: Consultar honorarios

RQF34	<i>Consultar honorarios</i>
Creado por	Angie López Villegas
Versión	1 (22/10/2021)
Módulo	Honorarios.
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema buscará en la base de datos los registros sobre el cobro de honorarios que solicite el abogado.
Dependencias	<ul style="list-style-type: none"> • <i>RQF02 Consultar usuario</i>
Actores	Usuario del sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Baja
Comentarios	

Tabla 39. Requerimiento funcional 34, Consultar honorarios.

Fuente: elaboración propia.

Requerimiento funcional 35: Agregar honorarios

RQF35	<i>Agregar honorarios</i>
Creado por	Angie López Villegas
Versión	1 (22/10/2021)
Módulo	Honorarios.
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El actor agrega un nuevo registro de honorarios.
Dependencias	<ul style="list-style-type: none"> • <i>RQF34 Consultar honorarios</i>
Actores	Usuario del sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Baja
Comentarios	

Tabla 40. Requerimiento funcional 35, Agregar honorarios.

Fuente: elaboración propia.

Requerimiento funcional 36: Agregar abonos

RQF36	<i>Agregar abonos</i>
Creado por	Angie López Villegas
Versión	1 (22/10/2021)
Módulo	Honorarios.
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El usuario actualiza el registro de honorarios, agrega abonos.
Dependencias	<ul style="list-style-type: none"> • <i>RQF34 Consultar honorarios</i>
Actores	Usuario del sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Baja

Comentarios	
--------------------	--

Tabla 41. Requerimiento funcional 36, Agregar abonos.

Fuente: elaboración propia.

Requerimiento funcional 37: Agregar caso

RQF37	<i>Agregar caso</i>
Creado por	Angie López Villegas
Versión	1 (22/10/2021)
Módulo	Honorarios.
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El abogado agrega un caso al expediente.
Dependencias	<ul style="list-style-type: none"> • <i>RQF02 Consultar usuario</i> • <i>RQF14 Consultar expediente</i>
Actores	Usuario del sistema de gestión de expedientes jurídicos, citas y recordatorios.
Importancia/Prioridad	Baja
Comentarios	

Tabla 42. Requerimiento funcional 37, Agregar caso.

Fuente: elaboración propia.

5.1.2 Requerimientos no funcionales

Los requerimientos no funcionales describen características o cualidades importantes del sistema. Los requerimientos no funcionales del sistema de gestión de expedientes jurídicos, citas y recordatorios son los siguientes:

Requerimiento no funcional 01: Desarrollo

RQNF01	<i>Desarrollo</i>
---------------	-------------------

Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	El sistema se desarrollará en Visual Studio Community 2019 con MVC y Razor, en el lenguaje C#.
Dependencias	
Importancia/Prioridad	Alta
Comentarios	

Tabla 43. Requerimiento no funcional 01, Desarrollo.

Fuente: elaboración propia.

Requerimiento no funcional 02: Base de datos

RQNF02	<i>Base de datos</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	La base de datos se manejará en SQL Server.
Dependencias	
Importancia/Prioridad	Media
Comentarios	

Tabla 44. Requerimiento no funcional 02, Base de datos.

Fuente: elaboración propia.

Requerimiento no funcional 03: Acceso y uso de la base de datos

RQNF03	<i>Acceso y uso de la base de datos</i>
Creado por	Angie López Villegas

Versión	1 (19/08/2021)
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	Para el acceso y uso de las bases de datos, se utilizarán las funciones de Entity Framework.
Dependencias	
Importancia/Prioridad	Media
Comentarios	

Tabla 45. Requerimiento no funcional 03, Acceso y uso de la base de datos.

Fuente: elaboración propia.

Requerimiento no funcional 04: Diseño web

RQNF04	<i>Diseño web</i>
Creado por	Angie López Villegas
Versión	1 (19/08/2021)
Fuentes	Entrevista con el abogado, dueño del bufete Fonseca Sequeira y asociados.
Descripción	Para el diseño web del sistema se utilizará Bootstrap 4.6, HTML y CSS
Dependencias	
Importancia/Prioridad	Baja
Comentarios	

Tabla 46. Requerimiento no funcional 04, Diseño web.

Fuente: elaboración propia.

5.1.3 Historias de usuario

Las historias de usuario son una representación de lo que debería hacer una función del software desde el punto de vista del usuario final. Por lo general no se utiliza el lenguaje técnico.

Las historias de usuario del proyecto del sistema web de gestión de expedientes jurídicos, citas y recordatorios, son las siguientes:

Historia de usuario 01: Iniciar sesión

Identificador (ID) de la historia		HU-01	
Enunciado de la historia			
Rol	Como usuario del sistema.		
Característica / Funcionalidad	Necesito iniciar sesión.		
Razón / Resultado	Con la finalidad de acceder a las funcionalidades del sistema.		
Criterios de aceptación			
Número (#) de escenario	1	2	3
Criterio de aceptación (Título)	Usuario no registrado.	Contraseña no es correcta.	Campos vacíos.
Contexto	En caso que el usuario ingresado no se encuentre registrado.	En caso que la contraseña ingresada no coincida con la asignada al usuario.	En caso que el usuario no complete todos los campos necesarios para iniciar sesión.
Evento	Cuando ingrese el usuario para ingresar al sistema.	Cuando ingrese una contraseña para ingresar al sistema.	Cuando deje campos necesarios en blanco.
Resultado / Comportamiento esperado	El sistema deberá mostrar un mensaje indicando que el usuario no se encuentra	El sistema mostrara un mensaje indicando que la contraseña o el usuario no son	El sistema mostrara un mensaje indicando que verifique los

	registrado, que por favor verifique la información ingresada.	correctos, que verifique los datos ingresados.	espacios que faltan por rellenar.
--	---	--	-----------------------------------

Tabla 47. Historia de usuario 01, Iniciar sesión.

Fuente: elaboración propia

Historia de usuario 02: Crear un nuevo usuario

Identificador (ID) de la historia		HU-02
Enunciado de la historia		
Rol	Como usuario administrador.	
Característica / Funcionalidad	Necesito crear un usuario nuevo.	
Razón / Resultado	Para darle acceso al sistema a un nuevo usuario.	
Criterios de aceptación		
Número (#) de escenario	1	2
Criterio de aceptación (Título)	Ingreso de usuario.	No ingresar datos del usuario.
Contexto	En caso de que se ingresen correctamente los datos para crear un usuario nuevo.	En caso de que hagan falta datos necesarios para completar el registro.
Evento	Cuando ingrese un usuario.	Cuando ingrese un usuario.

Resultado / Comportamiento esperado	El sistema mostrará un mensaje diciendo que el usuario se agregó correctamente y le enviará un email al usuario agregado.	El sistema mostrará un mensaje diciendo que hace falta información.
--	---	---

Tabla 48. Historia de usuario 02, Crear un nuevo usuario.

Fuente: elaboración propia

Historia de usuario 03: Modificar usuario

Identificador (ID) de la historia		HU-03
Enunciado de la historia		
Rol	Como usuario del sistema.	
Característica / Funcionalidad	Necesito cambiar los datos de mi perfil.	
Razón / Resultado	Para mantener mi información actualizada.	
Criterios de aceptación		
Número (#) de escenario	1	2
Criterio de aceptación (Título)	Ingreso de información.	No ingresar información.
Contexto	En caso de que el usuario realice el cambio de información.	En caso de que el usuario no modifique la información existente.

Evento	Cuando intente cambiar su información.	Cuando intente cambiar su información.
Resultado / Comportamiento esperado	El sistema mostrará un mensaje diciendo que los datos fueron actualizados.	El sistema mostrará un mensaje diciendo que los datos fueron actualizados, aunque los datos sean los mismos.

Tabla 49. Historia de usuario 03, Modificar usuario.

Fuente: elaboración propia

Historia de usuario 04: Eliminar usuario

Identificador (ID) de la historia	HU-04	
Enunciado de la historia		
Rol	Como usuario administrador del sistema.	
Característica / Funcionalidad	Necesito eliminar un usuario registrado.	
Razón / Resultado	Para que éste ya no tenga acceso al sistema.	
Criterios de aceptación		
Número (#) de escenario	1	2
Criterio de aceptación (Título)	Usuario no encontrado.	Usuario encontrado.

Contexto	En caso de que no se encuentre el usuario que se quiere eliminar.	En caso de que se encuentre el usuario.
Evento	Cuando se ingrese el identificador del usuario a eliminar.	Cuando se ingrese el identificador del usuario a eliminar.
Resultado / Comportamiento esperado	El sistema mostrará un mensaje diciendo que el usuario no fue encontrado, que verifique la información ingresada.	El sistema procederá a permitir la eliminación del usuario. Una vez eliminado se muestra un mensaje diciendo que el usuario fue eliminado.

Tabla 50. Historia de usuario 04, Eliminar usuario.

Fuente: elaboración propia

Historia de usuario 05: Recuperar contraseña

Identificador (ID) de la historia	HU-05
Enunciado de la historia	
Rol	Como usuario del sistema.
Característica / Funcionalidad	Necesito cambiar mi contraseña.
Razón / Resultado	Porque la olvidé o porque creo que es momento de cambiarla.
Criterios de aceptación	

Número (#) de escenario	1	2
Criterio de aceptación (Título)	Usuario encontrado.	Usuario no encontrado.
Contexto	En caso de que se encuentre el usuario al que se le quiere recuperar la contraseña.	En caso de que no se encuentre el usuario al que se le quiere recuperar la contraseña.
Evento	Al intentar recuperar la contraseña.	Al intentar recuperar la contraseña.
Resultado / Comportamiento esperado	Se mostrará un mensaje diciendo que la nueva contraseña ha sido enviada a su correo. Y, se envía un email al usuario.	Se muestra un mensaje diciendo que el usuario no está registrado, que verifique la información proporcionada.

Tabla 51. Historia de usuario 05, Recuperar contraseña.

Fuente: elaboración propia

Historia de usuario 06: Agregar tipo de proceso

Identificador (ID) de la historia	HU-06
Enunciado de la historia	
Rol	Como usuario abogado.
Característica / Funcionalidad	Necesito agregar un tipo de proceso con el que suelo trabajar.

Razón / Resultado	Para poder seleccionarlo en el área de expedientes.	
Criterios de aceptación		
Número (#) de escenario	1	2
Criterio de aceptación (Título)	Tipo de proceso ya registrado.	Tipo de proceso no registrado.
Contexto	En caso de que ese tipo ya esté registrado.	En caso de que el tipo de proceso no se haya registrado.
Evento	Al agregar tipo de proceso.	Al agregar tipo de proceso.
Resultado / Comportamiento esperado	Se muestra un mensaje diciendo que el proceso ya está registrado.	Se muestra un mensaje diciendo que el tipo de proceso se registró exitosamente.

Tabla 52. Historia de usuario 06, Agregar tipo de proceso.

Fuente: elaboración propia

Historia de usuario 07: Eliminar tipo de proceso

Identificador (ID) de la historia	HU-07
Enunciado de la historia	
Rol	Como usuario abogado.

Característica / Funcionalidad	Necesito eliminar un tipo de proceso.
Razón / Resultado	Ya no lo pienso utilizar.
Criterios de aceptación	
Número (#) de escenario	1
Criterio de aceptación (Título)	Tipo de proceso encontrado.
Contexto	En caso de que el tipo de proceso se encuentre en la barra de selección.
Evento	Al eliminar el tipo de proceso.
Resultado / Comportamiento esperado	Se permite el proceso de eliminación, y se muestra un mensaje diciendo que el tipo de proceso se eliminó correctamente.

Tabla 53. Historia de usuario 07, Eliminar tipo de proceso.

Fuente: elaboración propia

Historia de usuario 08: Agregar cliente

Identificador (ID) de la historia	HU-08
Enunciado de la historia	

Rol	Como usuario abogado.		
Característica / Funcionalidad	Necesito agregar un cliente nuevo al sistema.		
Razón / Resultado	Para poder asignarle expedientes.		
Criterios de aceptación			
Número (#) de escenario	1	2	3
Criterio de aceptación (Título)	Cliente ya registrado.	Ingreso de cliente.	Espacios en blanco.
Contexto	En caso de que el cliente que se quiere agregar ya esté agregado.	En caso de que el cliente se agregue correctamente.	En caso de que hagan falta campos por rellenar para completar el registro.
Evento	Al crear un cliente.	Al crear un cliente.	Al crear un cliente.
Resultado / Comportamiento esperado	Se muestra un mensaje diciendo que el cliente ya está agregado.	Se muestra un mensaje diciendo que el cliente se agregó de manera exitosa.	Se muestra un mensaje que diga que hacen falta llenar espacios para agregar al cliente.

Tabla 54. Historia de usuario 08, Agregar cliente.

Fuente: elaboración propia

Historia de usuario 09: Modificar cliente

Identificador (ID) de la historia		HU-09
Enunciado de la historia		
Rol	Como usuario abogado.	
Característica / Funcionalidad	Necesito cambiar información del cliente.	
Razón / Resultado	Debido a que los datos que tengo ya son obsoletos.	
Criterios de aceptación		
Número (#) de escenario	1	2
Criterio de aceptación (Título)	Cliente no encontrado.	Modificación de cliente.
Contexto	En caso de que el cliente no se encuentre en la búsqueda previa.	En caso de que el cliente se encuentre y se cambien los datos necesarios.
Evento	Al buscar el cliente a modificar.	Al modificar los datos del cliente.
Resultado / Comportamiento esperado	Se muestra los resultados similares a la búsqueda realizada.	Se realiza la modificación.

Tabla 55. Historia de usuario 09, Modificar cliente.

Fuente: elaboración propia

Historia de usuario 10: Eliminar cliente

Identificador (ID) de la historia		HU-12
Enunciado de la historia		
Rol	Como usuario abogado.	
Característica / Funcionalidad	Necesito deshabilitar un cliente.	
Razón / Resultado	Ya no es mi cliente.	
Criterios de aceptación		
Número (#) de escenario	1	2
Criterio de aceptación (Título)	Cliente eliminado.	Cliente no encontrado.
Contexto	En caso de que el cliente se encuentre y se proceda a eliminar o deshabilitar.	En caso de que el cliente no se encuentre en la búsqueda previa.
Evento	Al eliminar al cliente.	Al buscar el cliente a eliminar.
Resultado / Comportamiento esperado	Se muestra un mensaje diciendo que la información del cliente fue modificado.	Se muestra los resultados similares a la búsqueda realizada.

Tabla 56. Historia de usuario 10, Eliminar cliente.

Fuente: elaboración propia

Historia de usuario 11: Crear expediente

Identificador (ID) de la historia		HU-11
Enunciado de la historia		
Rol	Como usuario abogado.	
Característica / Funcionalidad	Necesito crear un expediente.	
Razón / Resultado	Para trabajar en el caso de un cliente.	
Criterios de aceptación		
Número (#) de escenario	1	2
Criterio de aceptación (Título)	Expediente ya existe.	Crear expediente.
Contexto	En caso de que el expediente ya haya sido creado.	En caso de que el expediente no haya sido creado.
Evento	Al escribir el número de expediente.	Al escribir el número de expediente.
Resultado / Comportamiento esperado	Se muestra la información de este.	Se permite la creación del mismo. Al finalizar se muestra un mensaje

		diciendo que el expediente se creó correctamente.
--	--	---

Tabla 57. Historia de usuario 11, Crear expediente.

Fuente: elaboración propia

Historia de usuario 12: Modificar expediente

Identificador (ID) de la historia		HU-12
Enunciado de la historia		
Rol	Como usuario abogado.	
Característica / Funcionalidad	Necesito modificar un expediente.	
Razón / Resultado	Es necesario agregar actualizaciones del mismo.	
Criterios de aceptación		
Número (#) de escenario	1	2
Criterio de aceptación (Título)	Modificar expediente.	Expediente no encontrado.
Contexto	En caso de que el expediente se encuentre y se modifiquen los datos deseados.	En caso de que el expediente no se encuentre en la búsqueda previa.

Evento	Al modificar el expediente.	Al buscar el expediente a modificar.
Resultado / Comportamiento esperado	Se muestra un mensaje diciendo que el expediente fue actualizado correctamente.	Se permite crear un expediente nuevo. El espacio del número del expediente se mantiene habilitado para permitir la búsqueda nuevamente.

Tabla 58. Historia de usuario 12, Modificar expediente.

Fuente: elaboración propia

Historia de usuario 13: Agregar notificación

Identificador (ID) de la historia	HU-13	
Enunciado de la historia		
Rol	Como usuario abogado.	
Característica / Funcionalidad	Necesito agregar una notificación que me enviaron con respecto al expediente.	
Razón / Resultado	Mantener el expediente actualizado, y notificarlo a los clientes.	
Criterios de aceptación		
Número (#) de escenario	1	2

Criterio de aceptación (Título)	Campos incompletos.	Agregar notificación.
Contexto	En caso de que falten datos que agregar.	En caso de que se agregue la notificación.
Evento	Al agregar una notificación.	Al agregar una notificación.
Resultado / Comportamiento esperado	Se muestra un mensaje diciendo que faltan espacios por llenar.	Se muestra un mensaje diciendo que la notificación se agregó correctamente y se envía un email al cliente con la información de la notificación.

Tabla 59. Historia de usuario 13, Agregar notificación.

Fuente: elaboración propia

Historia de usuario 14: Agregar cita

Identificador (ID) de la historia	HU-14
Enunciado de la historia	
Rol	Como usuario del sistema.
Característica / Funcionalidad	Necesito agregar una cita en mi agenda.
Razón / Resultado	Mantener mi agenda actualizada.
Criterios de aceptación	

Número (#) de escenario	1	2	3
Criterio de aceptación (Título)	Fecha y hora ya existente.	Registrar cita.	Información faltante.
Contexto	En caso de que la cita a agendar tenga la misma fecha y hora que otra ya registrada.	En caso de registrar una cita.	En caso de que falte información para poder agendar la cita.
Evento	Al agendar una cita.	Al registrar una cita.	Al registrar una cita.
Resultado / Comportamiento esperado	Se muestra un mensaje diciendo que ese horario ya está seleccionado.	Se muestra un mensaje diciendo que la cita se agendó de forma correcta. Se envía un email al cliente con la información de la cita.	Se muestra un mensaje diciendo que hace falta información para poder agendar la cita.

Tabla 60. Historia de usuario 14, Agregar cita.

Fuente: elaboración propia

Historia de usuario 15: Modificar cita

Identificador (ID) de la historia	HU-15
Enunciado de la historia	
Rol	Como usuario del sistema.

Característica / Funcionalidad	Necesito cambiar la información de la cita.		
Razón / Resultado	Debido a inconvenientes en la fecha y hora acordada, ya sea de parte del cliente como de parte del abogado.		
Criterios de aceptación			
Número (#) de escenario	1	2	3
Criterio de aceptación (Título)	Falta información.	Modificación de la cita.	Fecha y hora ya existente.
Contexto	En caso de que falte información para completar la acción.	En caso de modificar la cita.	En caso de que la nueva fecha y hora de la cita tenga la misma fecha y hora que otra ya registrada.
Evento	Al modificar una cita.	Al modificar la cita.	Al modificar una cita.
Resultado / Comportamiento esperado	Se muestra un mensaje diciendo que hace falta información por llenar.	Se muestra un mensaje diciendo que la cita ha sido modificada exitosamente. Se envía un email al cliente con la información de la cita.	Se muestra un mensaje diciendo que ese horario ya está seleccionado.

Tabla 61. Historia de usuario 15, Modificar cita.

Fuente: elaboración propia

Historia de usuario 16: Eliminar cita

Identificador (ID) de la historia	HU-16
Enunciado de la historia	
Rol	Como usuario del sistema.
Característica / Funcionalidad	Necesito eliminar una cita de la agenda.
Razón / Resultado	Debido a alguna cancelación.
Criterios de aceptación	
Número (#) de escenario	1
Criterio de aceptación (Título)	Eliminar cita.
Contexto	En caso de eliminar cita.
Evento	Al eliminar cita.
Resultado / Comportamiento esperado	Se muestra un mensaje diciendo que la cita fue eliminada con éxito. Se envía un email al cliente con la notificación de cancelación de cita.

Tabla 62. Historia de usuario 16, Eliminar cita.

Fuente: elaboración propia

Historia de usuario 17: Agregar recordatorio

Identificador (ID) de la historia	HU-17
Enunciado de la historia	
Rol	Como usuario del sistema.
Característica / Funcionalidad	Necesito agregar un recordatorio.
Razón / Resultado	Es algo necesario de recordar.
Criterios de aceptación	
Número (#) de escenario	1
Criterio de aceptación (Título)	Agregar recordatorio.
Contexto	En caso de agregar recordatorio.
Evento	Al agregar un recordatorio.
Resultado / Comportamiento esperado	Se muestra un mensaje diciendo que el recordatorio se agregó exitosamente.

Tabla 63. Historia de usuario 17, Agregar recordatorio.

Fuente: elaboración propia

Historia de usuario 18: Modificar recordatorio

Identificador (ID) de la historia	HU-18
Enunciado de la historia	
Rol	Como usuario del sistema.
Característica / Funcionalidad	Necesito modificar cambiar la información de un recordatorio.
Razón / Resultado	Actualización de información relacionada al recordatorio.
Criterios de aceptación	
Número (#) de escenario	1
Criterio de aceptación (Título)	Modificar recordatorio.
Contexto	En caso de modificar recordatorio.
Evento	Al modificar recordatorio.
Resultado / Comportamiento esperado	Se muestra un mensaje diciendo que el recordatorio fue modificado con éxito.

Tabla 64. Historia de usuario 18, Modificar recordatorio.

Fuente: elaboración propia

Historia de usuario 19: Eliminar recordatorio

Identificador (ID) de la historia		HU-19
Enunciado de la historia		
Rol	Como usuario del sistema.	
Característica / Funcionalidad	Necesito eliminar un recordatorio.	
Razón / Resultado	Ya no es necesario.	
Criterios de aceptación		
Número (#) de escenario	1	
Criterio de aceptación (Título)	Eliminar recordatorio.	
Contexto	En caso de eliminar un recordatorio.	
Evento	Al eliminar un recordatorio.	

Resultado / Comportamiento esperado	Se muestra un mensaje diciendo que el recordatorio se eliminó correctamente.
--	--

Tabla 65. Historia de usuario 19, Eliminar recordatorio.

Fuente: elaboración propia

Historia de usuario 20: Crear reporte

Identificador (ID) de la historia	HU-20
Enunciado de la historia	
Rol	Como usuario abogado.
Característica / Funcionalidad	Necesito generar un reporte.
Razón / Resultado	Quiero verlo y/ o darlo a un cliente.
Criterios de aceptación	
Número (#) de escenario	1
Criterio de aceptación (Título)	Generar reporte.
Contexto	En caso de que seleccione el tipo de reporte que desea.

Evento	Al generar reporte.
Resultado / Comportamiento esperado	Se genera el reporte con los parámetros seleccionados.

Tabla 66. Historia de usuario 20, Crear reporte.

Fuente: elaboración propia

5.2 Diseño del proyecto

En esta sección del capítulo se desarrolla el diseño del sistema, en este se incluyen los diagramas de caso de uso, de clases, de actividades, de secuencia y de entidad-relación.

5.2.1 Casos de uso

Los casos de uso del proyecto de desarrollo del sistema de gestión de expedientes jurídicos, citas y recordatorios son los siguientes:

5.2.1.1 Manejos de sesión

El siguiente diagrama de casos de uso muestra aquellos casos de uso que tienen relación con el inicio y cierre de sesión, así como la recuperación de contraseña en caso de olvido. De igual forma se incluye la descripción de cada uno.

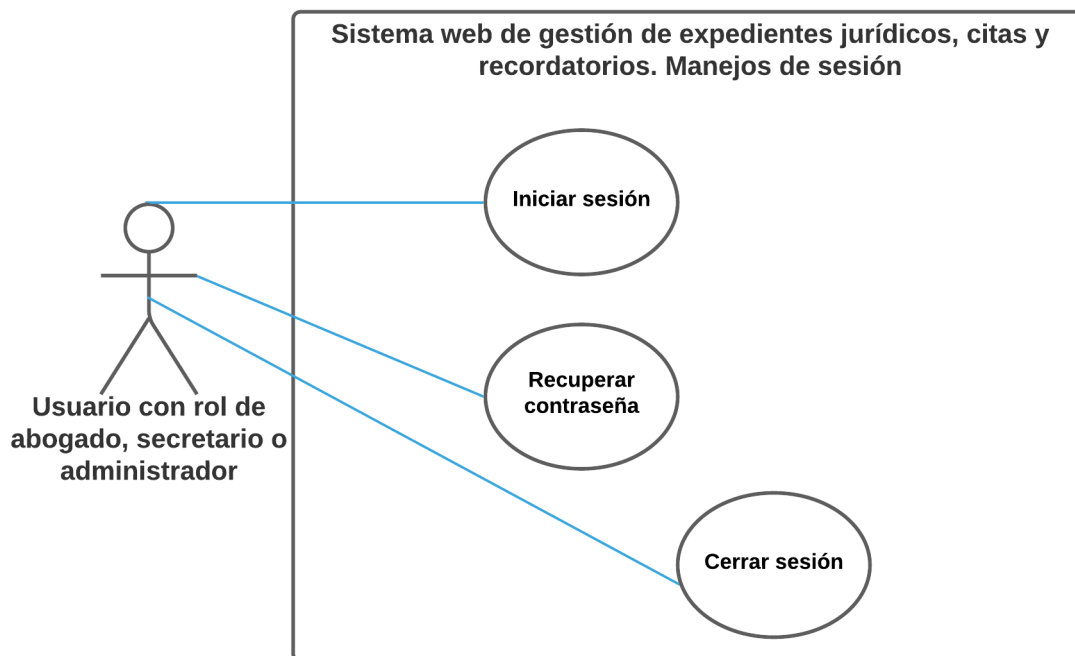


Figura 27. Diagrama de caso de uso, Manejos de sesión

Fuente: elaboración propia.

Código	CU-01	Nombre	Iniciar sesión
Actores	Administrador, Abogado.		
Objetivo	Iniciar sesión para los usuarios del sistema.		
Dependencias	<ul style="list-style-type: none"> • RQF01 Iniciar sesión. • RQF02 Consultar usuario. 		
Descripción	El actor entra al sistema web con un usuario y una contraseña.		
Precondición	Estar registrado anteriormente en el sistema.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • Actor entra a sitio web. • Sistema muestra pantalla para iniciar sesión. • Actor digita usuario y contraseña. 		

	<ul style="list-style-type: none"> • Actor presiona botón “Iniciar sesión”. • Sistema verifica que la información dada coincida con la almacenada en la base de datos. • Sistema permite al actor entrar. • Sistema muestra pantalla principal.
Post condición	Actor ingresa y puede utilizar el sistema.
Flujos alternos	<ol style="list-style-type: none"> 1. Actor deja alguno de los espacios en blanco. <ol style="list-style-type: none"> 1.1 El actor deja alguno de los espacios en blanco. 1.2 El actor presiona el botón “Iniciar sesión”. 1.3 El sistema muestra un mensaje que diga que hay espacios en blanco. 2. Actor digita incorrectamente el usuario o la contraseña. <ol style="list-style-type: none"> 2.1 El actor digita el usuario o la contraseña incorrectamente. 2.2 El usuario presiona el botón “Iniciar sesión”. 2.3 El sistema muestra un mensaje diciendo que el usuario o la contraseña son incorrectos.

Tabla 67. Caso de uso 01, Iniciar sesión.

Fuente: elaboración propia.

Código	CU-02	Nombre	Recuperar contraseña
Actores	Administrador, Abogado.		
Objetivo	Cambiar la contraseña para los usuarios del sistema.		
Dependencias	<ul style="list-style-type: none"> • RQF06 Recuperar la contraseña. • RQF32 Enviar email con contraseña nueva. 		
Descripción	El actor puede cambiar su contraseña por una nueva.		
Precondición	Estar registrado anteriormente en el sistema.		

Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • Actor entra a sitio web. • Sistema muestra pantalla para iniciar sesión. • Actor presiona botón olvidar contraseña. • Sistema muestra pantalla de recuperación de contraseña. • Actor digita su usuario. • Actor presiona botón recuperar contraseña. • Sistema verifica que el usuario esté registrado. • Sistema muestra mensaje diciendo que se ha enviado un email al email registrado. • Sistema envía un email al actor con una nueva contraseña.
Post condición	Actor recupera su contraseña y puede acceder al sistema.
Flujos alternos	<ol style="list-style-type: none"> 1. Usuario digitado no encontrado. <ol style="list-style-type: none"> 1.1 Actor digita su usuario. 1.2 Actor presiona botón recuperar contraseña. 1.3 Sistema verifica que el usuario esté registrado. 1.4 Sistema muestra mensaje diciendo que el usuario no fue encontrado.

Tabla 68. Caso de uso 02, Recuperar contraseña.

Fuente: elaboración propia.

Código	CU-03	Nombre	Cerrar sesión
Actores	Administrador, Abogado.		
Objetivo	Salir del sistema web.		
Dependencias	<ul style="list-style-type: none"> • RQF01 Iniciar sesión. 		
Descripción	El actor termina de utilizar la aplicación y desea cerrar la sesión iniciada.		
Precondición	Debe estar registrado anteriormente y haber iniciado sesión en la aplicación.		

Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor presiona el botón “Cerrar sesión”. • Se cierra la sesión. • El sistema muestra la pantalla de inicio de sesión.
Post condición	Se cierra la sesión.
Flujos alternos	

Tabla 69. Caso de uso 03, Cerrar sesión.

Fuente: elaboración propia.

5.2.1.2 Módulo de clientes

El módulo de clientes cuenta con los siguientes casos de uso:

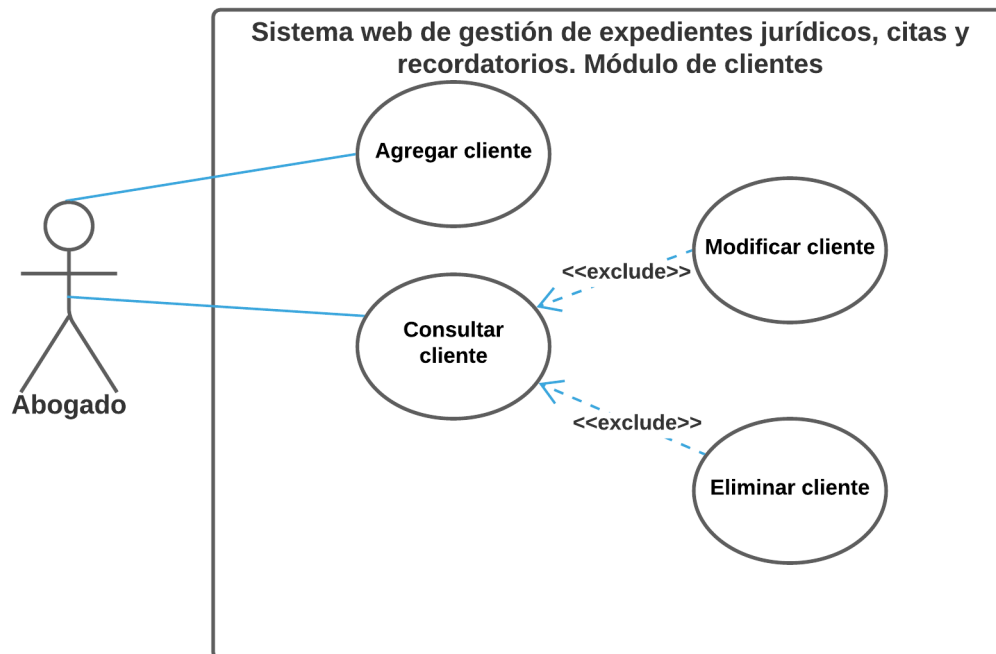


Figura 28. Diagrama de caso de uso, Módulo de clientes

Fuente elaboración propia.

Código	CU-04	Nombre	Agregar cliente.
Actores	Abogado.		
Objetivo	Permitir al abogado ingresar un nuevo cliente al sistema.		
Dependencias	<ul style="list-style-type: none"> • RQF09 Consultar cliente. • RQF10 Agregar cliente. 		
Descripción	El actor ingresa la información de un nuevo cliente y la almacena en la base de datos.		
Precondición	El actor debe haber iniciado sesión en el sistema y tener rol de abogado.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de clientes. • El actor presiona el botón “Crear nuevo perfil de cliente”. • El sistema re direcciona al actor a la pantalla de crear cliente. • El sistema muestra pantalla de crear cliente. • El actor digita los datos requeridos para agregar al cliente. • El actor presiona el botón de agregar cliente. • El sistema agrega al cliente. • El sistema muestra un mensaje diciendo que el cliente fue agregado con éxito. 		
Post condición	El cliente se agrega.		
Flujos alternos	<ol style="list-style-type: none"> 1. El actor deja espacios sin rellenar. <ol style="list-style-type: none"> 1.1 El actor deja espacios en blanco. 1.2 El actor presiona el botón de agregar cliente. 1.3 El sistema muestra un mensaje diciendo que hay espacios en blanco. 2. El cliente ya fue agregado. <ol style="list-style-type: none"> 2.1 El actor ingresa la información del cliente. 2.2 El actor presiona el botón de agregar cliente. 		

	<p>2.3 El sistema verifica la información.</p> <p>2.4 El sistema muestra un mensaje diciendo que el cliente ya existe en la base de datos.</p>
--	--

Tabla 70. Caso de uso 04, Agregar cliente.

Fuente: elaboración propia.

Código	CU-05	Nombre	Consultar cliente
Actores	Abogado.		
Objetivo	Verificar si el cliente ya está registrado.		
Dependencias	<ul style="list-style-type: none"> • RQF09 Consultar cliente. • RQF13 Re direccionar a expediente. 		
Descripción	El actor consulta si un cliente ya está registrado en el sistema.		
Precondición	El actor debe haber iniciado sesión y debe tener rol de abogado.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de clientes. • El actor digita el nombre del cliente en el buscador. • El sistema devuelve los resultados. • El actor selecciona el nombre del cliente. • La información del cliente se despliega en la pantalla. 		
Post condición	La información del cliente se despliega en la pantalla.		
Flujos alternos	<ol style="list-style-type: none"> 1. El cliente no se encuentra en la búsqueda. <ol style="list-style-type: none"> 1.1 El actor digita el nombre del cliente en el buscador. 1.2 El sistema devuelve los resultados similares a la búsqueda. 		

Tabla 71. Caso de uso 05, Consultar cliente.

Fuente: elaboración propia.

Código	CU-06	Nombre	Modificar cliente
Actores	Abogado.		
Objetivo	Cambiar información almacenada del cliente.		
Dependencias	<ul style="list-style-type: none"> • CU-05 Consultar cliente. • RQF09 Consultar cliente. • RQF11 Modificar cliente. 		
Descripción	El actor cambia la información almacenada relacionada a un cliente en específico.		
Precondición	El actor debe haber iniciado sesión y tener rol de abogado.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra en el módulo de clientes. • El actor realiza la consulta del cliente. • El sistema despliega la información del cliente. • El actor realiza los cambios deseados. • El sistema muestra un mensaje diciendo que los cambios se realizaron con éxito. 		
Post condición	Se realizan los cambios.		
Flujos alternos			

Tabla 72. Caso de uso 06, Modificar cliente.

Fuente: elaboración propia.

Código	CU-07	Nombre	Eliminar cliente
Actores	Abogado.		
Objetivo	Eliminar un cliente del sistema.		
Dependencias	<ul style="list-style-type: none"> • CU-05 Consultar cliente. 		

	<ul style="list-style-type: none"> • RQF09 Consultar cliente. • RQF12 Eliminar cliente.
Descripción	El actor elimina un cliente del sistema.
Precondición	El actor debe haber iniciado sesión y tener rol de abogado.
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra en el módulo de clientes. • El actor realiza la búsqueda del cliente. • El actor selecciona el nombre del cliente. • El sistema despliega la información del cliente. • El actor selecciona la opción de desactivar cliente. • El actor presiona el botón de modificar cliente. • El sistema muestra un mensaje diciendo que el cliente se modificó. • El cliente se desactiva.
Post condición	El cliente se desactiva.
Flujos alternos	

Tabla 73. Caso de uso 07, Eliminar cliente.

Fuente: elaboración propia.

5.2.1.3 Módulo de expedientes

En el siguiente diagrama se exponen de forma gráfica los casos de uso del módulo de expedientes. De igual forma se detalla cada caso de uso en las siguientes tablas.

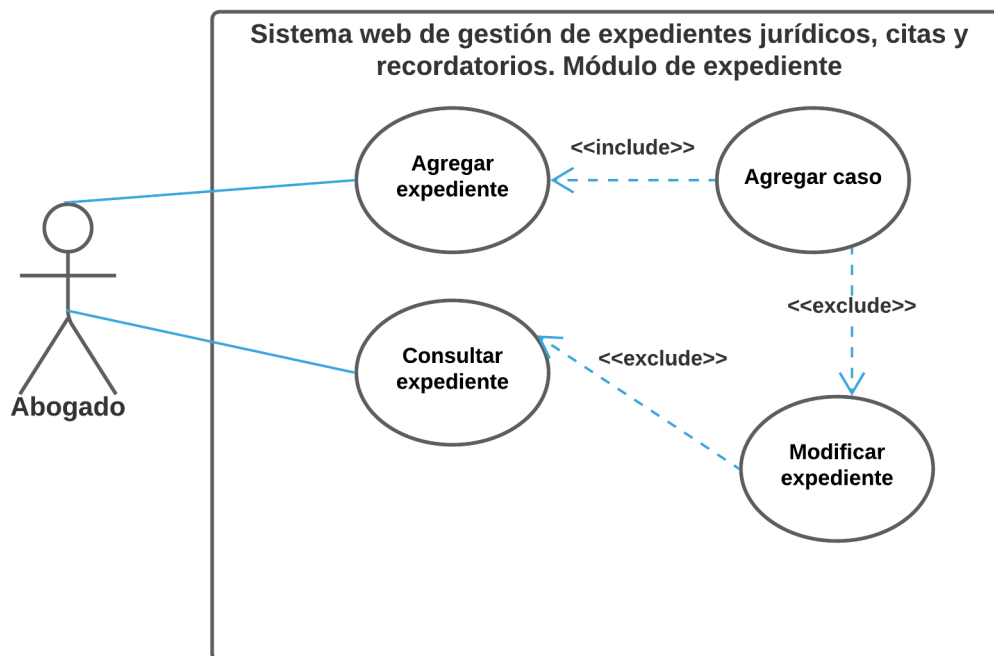


Figura 29. Diagrama de caso de uso 03, Módulo de expedientes.

Fuente: elaboración propia

Código	CU-08	Nombre	Agregar expediente
Actores	Abogado.		
Objetivo	Crear un nuevo expediente en el cual trabajar.		
Dependencias	<ul style="list-style-type: none"> • CU-09 Consultar expediente. • RQF15 Crear expediente. • RQF14 Consultar expediente. 		
Descripción	El actor agrega un expediente en el que va a trabajar.		
Precondición	El actor debe haber iniciado sesión y tener rol de abogado.		

Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de expedientes. • El actor digita el identificador del expediente. • El sistema verifica que es un expediente nuevo. • El actor puede proceder a ingresar información en el expediente. • El actor presiona el botón agregar expediente. • El expediente se registra en el sistema. • El sistema muestra un mensaje diciendo que el expediente se agregó de manera satisfactoria.
Post condición	El expediente se agrega.
Flujos alternos	<ol style="list-style-type: none"> 1. El expediente ya existe. <ol style="list-style-type: none"> 1.1 El actor digita el identificador del expediente. 1.2 El sistema realiza la verificación. 1.3 El sistema despliega la información del expediente digitado. 2. Espacios en blanco. <ol style="list-style-type: none"> 2.1 El actor deja en blanco espacios obligatorios del formulario. 2.2 El actor presiona el botón crear expediente. 2.3 El sistema muestra un mensaje diciendo que hay espacios en blanco que deben llenarse.

Tabla 74. Caso de uso 08, Agregar expediente.

Fuente: elaboración propia.

Código	CU-09	Nombre	Consultar expediente
Actores	Abogado.		
Objetivo	Buscar un expediente.		
Dependencias	<ul style="list-style-type: none"> • RQF14 Consultar expediente. 		
Descripción	El actor consulta por un expediente en el sistema.		

Precondición	El actor debe haber iniciado sesión y tener rol de abogado.
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor digita el identificador del expediente. • El sistema realiza la búsqueda. • El sistema despliega la información del expediente consultado.
Post condición	El sistema despliega la información del expediente consultado.
Flujos alternos	<ol style="list-style-type: none"> 1. El expediente no se encuentra. <ol style="list-style-type: none"> 1.1 El actor digita el identificador del expediente. 1.2 El sistema realiza la búsqueda. 1.3 El sistema permite al actor proceder a crear el expediente.

Tabla 75. Caso de uso 09, Consultar expediente.

Fuente: elaboración propia.

Código	CU-10	Nombre	Modificar expediente
Actores	Abogado.		
Objetivo	Actualizar la información de un expediente.		
Dependencias	<ul style="list-style-type: none"> • CU-09 Consultar expediente. • RQF14 Consultar expediente. • RQF16 Modificar expediente. • RQF27 Enviar email de actualización para notificaciones del expediente. • RQF17 Agregar notificación. 		
Descripción	El actor busca un expediente y lo modifica.		
Precondición	<ul style="list-style-type: none"> • El actor debe haber iniciado sesión y tener rol de abogado. • El expediente tiene que estar registrado. 		

Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor digita el identificador del expediente. • El sistema realiza la búsqueda. • El sistema despliega la información del expediente. • El actor actualiza la información. • El actor presiona el botón modificar expediente. • El sistema realiza la modificación. • El sistema muestra un mensaje diciendo que el expediente se actualizó con éxito.
Post condición	El expediente es modificado.
Flujos alternos	

Tabla 76. Caso de uso 10, Modificar expediente.

Fuente: elaboración propia.

Código	CU-11	Nombre	Agregar caso
Actores	Abogado.		
Objetivo	Agregar un caso a un expediente.		
Dependencias	<ul style="list-style-type: none"> • CU-09 Consultar expediente. • CU-08 Agregar expediente. • RQF14 Consultar expediente. • RQF37 Agregar caso. 		
Descripción	El actor agrega un caso a un expediente ya sea nuevo, o existente.		
Precondición	<ul style="list-style-type: none"> • El actor debe haber iniciado sesión y tener rol de abogado. 		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor digita el identificador del expediente. • El sistema verifica que el expediente no existe. • El actor llena la información del caso y expediente nuevo. 		

	<ul style="list-style-type: none"> • El actor presiona el botón crear expediente. • El sistema crea el caso relacionado al expediente nuevo.
Post condición	El caso es creado.
Flujos alternos	<ol style="list-style-type: none"> 1. El expediente ya tiene un caso creado. <ol style="list-style-type: none"> 1.1 El abogado presiona el botón de agregar nuevo caso. 1.2 Se crea un caso nuevo para ese expediente.

Tabla 77. Caso de uso 11, Agregar caso.

Fuente: elaboración propia.

5.2.1.4 Módulo de reportes

A continuación, se muestran tanto el diagrama de casos de uso, como la descripción del caso de uso del módulo de reportes.

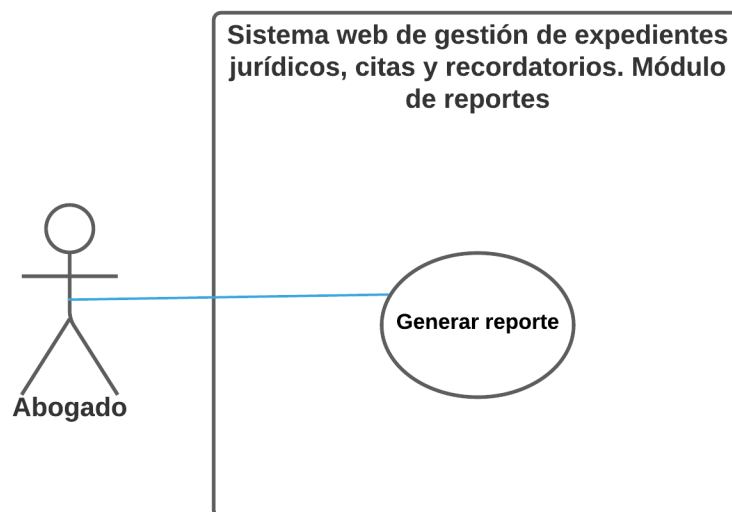


Figura 30. Diagrama de caso de uso 04, Módulo de reportes.

Fuente: elaboración propia.

Código	CU-12	Nombre	Generar reporte
Actores	Abogado.		
Objetivo	Permitir al abogado generar un reporte con base a los parámetros solicitados.		
Dependencias	<ul style="list-style-type: none"> • RQF26 Crear reporte. 		
Descripción	El actor genera un reporte que contenga la información que él desea con ayuda de opciones a seleccionar. Las opciones son: generar reportes de un cliente, de un expediente, de correos enviados a un cliente o de notificaciones relacionadas a un expediente.		
Precondición	El actor debe haber iniciado sesión y tener rol de abogado.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de reportes. • El actor digita la cédula del cliente o el número de expediente. • El actor presiona el botón “Buscar” • El sistema muestra los resultados. • El usuario presiona el botón del tipo de reporte que desea. • El sistema genera el reporte. 		
Post condición	Se genera el reporte.		
Flujos alternos	<ol style="list-style-type: none"> 1. El actor no digita nada para buscar. <ol style="list-style-type: none"> 1.1 El actor no digita el número de identificación del cliente ni el número de expediente en el espacio de búsqueda. 1.2 El actor presiona el botón buscar. 1.3 El sistema muestra un mensaje diciendo que el espacio de la búsqueda es obligatorio. 		

Tabla 78. Caso de uso 12, Generar reporte.

Fuente: elaboración propia.

5.2.1.5 Módulo de agenda

Este diagrama de casos de uso, demuestra de manera gráfica los casos de uso que tendrá este módulo y la descripción de cada uno está detallado en las tablas agregadas.

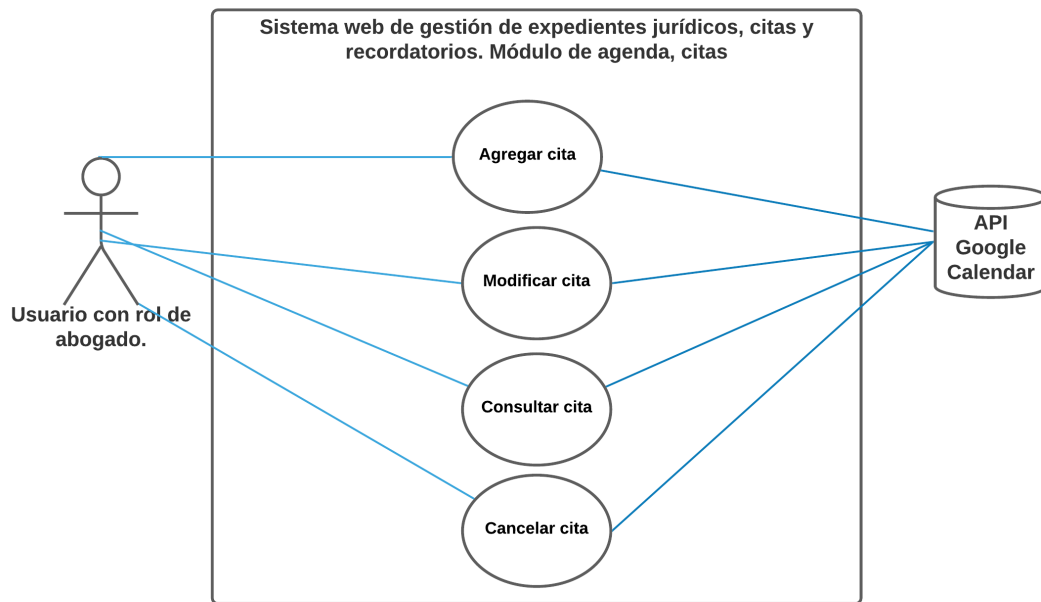


Figura 31. Diagrama de caso de uso 05, Módulo de agenda, citas.

Fuente: elaboración propia.

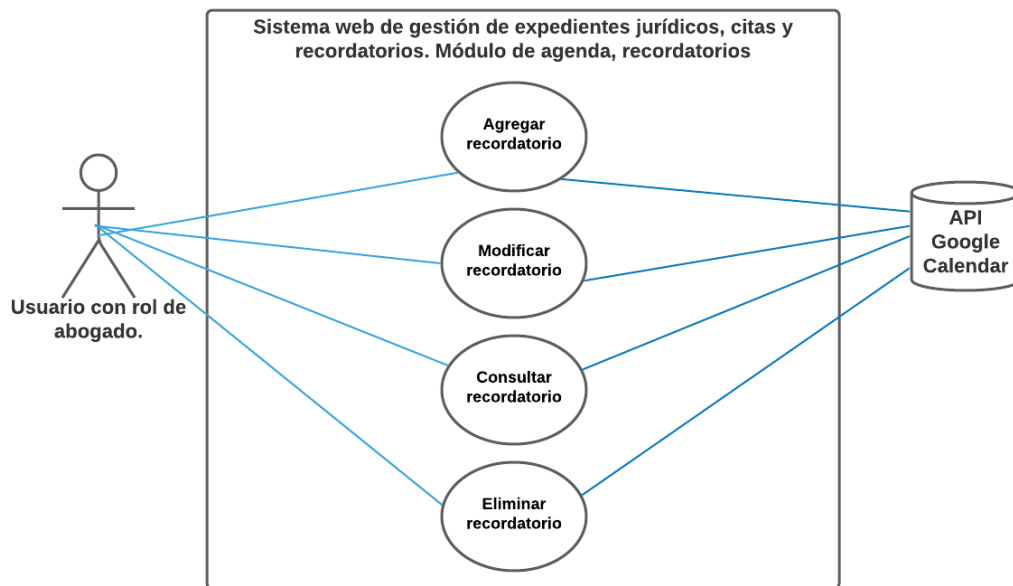


Figura 32. Diagrama de caso de uso 06, Módulo de agenda, recordatorios.

Fuente: elaboración propia.

Código	CU-13	Nombre	Agregar cita
Actores	Abogado.		
Objetivo	Agregar una cita en la agenda del abogado.		
Dependencias	<ul style="list-style-type: none"> • RQF18 Consultar cita. • RQF19 Agregar cita. • RQF28 Enviar email de cita. 		
Descripción	Se agrega una cita a la agenda de un abogado.		
Precondición	El actor debe haber iniciado sesión.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de agenda. • El actor presiona el botón de agregar evento. • El sistema muestra el formulario de agregar evento. 		

	<ul style="list-style-type: none"> • El actor llena la información de la cita y selecciona el tipo de evento a “cita”. • El actor presiona el botón agregar evento. • La información se envía al API de Google Calendar. • El sistema agrega la cita. • El sistema muestra un mensaje diciendo que la cita se agregó correctamente. • Se envía un email al cliente con la información de la cita agendada.
Post condición	La cita se agrega.
Flujos alternos	<ol style="list-style-type: none"> 1. La fecha y hora ya está seleccionada. <ol style="list-style-type: none"> 1.1 El actor presiona el botón de agregar evento. 1.2 El sistema muestra un mensaje diciendo que la fecha y hora ya está tomada. 2. El actor deja espacios obligatorios en blanco. <ol style="list-style-type: none"> 2.1 El actor deja espacios en blanco. 2.2 El actor presiona el botón agregar evento. 2.3 El sistema muestra un mensaje diciendo que faltan espacios por llenar.

Tabla 79. Caso de uso 13, Agregar cita.

Fuente: elaboración propia.

Código	CU-14	Nombre	Consultar cita
Actores	Abogado.		
Objetivo	Consultar citas en la agenda del abogado.		
Dependencias	<ul style="list-style-type: none"> • RQF18 Consultar cita. 		
Descripción	Se consultan citas en la agenda de un abogado.		

Precondición	El actor debe haber iniciado sesión.
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de agenda. • El sistema muestra las citas agendadas.
Post condición	Se muestran los resultados.
Flujos alternos	

Tabla 80. Caso de uso 14, Consultar cita.

Fuente: elaboración propia.

Código	CU-15	Nombre	Modificar cita
Actores	Abogado.		
Objetivo	Modificar una cita en la agenda del abogado.		
Dependencias	<ul style="list-style-type: none"> • RQF18 Consultar cita. • RQF20 Modificar cita • RQF28 Enviar email de cita. 		
Descripción	Se modifica una cita en la agenda de un abogado.		
Precondición	<p>El actor debe haber iniciado sesión.</p> <p>La cita tiene que estar registrada.</p>		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de agenda. • El actor selecciona la cita que desea modificar. • El sistema muestra la pantalla de modificar cita. • El actor cambia la información de la cita. • El actor presiona el botón modificar. • La información se envía al API de Google Calendar. • El sistema modifica la cita. 		

	<ul style="list-style-type: none"> • El sistema muestra un mensaje diciendo que la cita se ha modificado correctamente. • El sistema envía un email al cliente con la información de la cita actualizada.
Post condición	La cita se modifica.
Flujos alternos	<ol style="list-style-type: none"> 1. La fecha y hora ya está seleccionada. <ol style="list-style-type: none"> 1.1 El actor presiona el botón de modificar. 1.2 El sistema muestra un mensaje diciendo que la fecha y hora ya está tomada. 2. El actor deja espacios obligatorios en blanco. <ol style="list-style-type: none"> 1.1 El actor deja espacios en blanco. 1.2 El actor presiona el botón modificar. 1.3 El sistema muestra un mensaje diciendo que faltan espacios por llenar.

Tabla 81. Caso de uso 15, Modificar cita.

Fuente: elaboración propia.

Código	CU-16	Nombre	Cancelar cita
Actores	Abogado.		
Objetivo	Cancelar una cita en la agenda del abogado.		
Dependencias	<ul style="list-style-type: none"> • RQF18 Consultar cita. • RQF21 Cancelar cita • RQF28 Enviar email de cita. 		
Descripción	Se cancela una cita en la agenda de un abogado.		
Precondición	<p>El actor debe haber iniciado sesión.</p> <p>La cita debe estar registrada.</p>		

Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de agenda. • El sistema realiza la consulta de citas agendadas. • El actor selecciona la cita que desea eliminar. • El actor presiona el botón eliminar cita. • La acción se envía al API de Google Calendar. • El sistema elimina la cita. • El sistema muestra un mensaje diciendo que la cita se ha cancelado correctamente. • El sistema envía un email al cliente donde se informa la cancelación de la cita.
Post condición	La cita se elimina.
Flujos alternos	

Tabla 82. Caso de uso 16, Cancelar cita.

Fuente: elaboración propia.

Código	CU-17	Nombre	Agregar recordatorio
Actores	Abogado.		
Objetivo	Agregar un recordatorio en la agenda del abogado.		
Dependencias	<ul style="list-style-type: none"> • RQF22 Consultar recordatorio. • RQF23 Agregar recordatorio. 		
Descripción	Se agrega un recordatorio en la agenda de un abogado.		
Precondición	El actor debe haber iniciado sesión.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de agenda. • El actor presiona el botón de agregar evento. • El sistema muestra formulario de agregar evento. 		

	<ul style="list-style-type: none"> • El actor llena la información del recordatorio y selecciona el tipo de evento a “recordatorio”. • El actor presiona el botón agregar evento. • La información se envía al API de Google Calendar. • El sistema agrega el recordatorio. • El sistema muestra un mensaje diciendo que el recordatorio se agregó correctamente.
Post condición	El recordatorio se agrega.
Flujos alternos	<ol style="list-style-type: none"> 1. El actor deja espacios obligatorios en blanco. <ol style="list-style-type: none"> 1.1 El actor deja espacios en blanco. 1.2 El actor presiona el botón agregar evento. 1.3 El sistema muestra un mensaje diciendo que faltan espacios por llenar.

Tabla 83. Caso de uso 17, Agregar recordatorio.

Fuente: elaboración propia.

Código	CU-18	Nombre	Consultar recordatorio
Actores	Abogado.		
Objetivo	Consultar recordatorios en la agenda del abogado.		
Dependencias	<ul style="list-style-type: none"> • RQF22 Consultar recordatorio. 		
Descripción	Se consultan recordatorios en la agenda de un abogado.		
Precondición	El actor debe haber iniciado sesión.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de agenda. • El sistema muestra los recordatorios almacenados. 		

Post condición	Se muestran los resultados.
Flujos alternos	

Tabla 84. Caso de uso 18, Consultar recordatorio.

Fuente: elaboración propia.

Código	CU-19	Nombre	Modificar recordatorio
Actores	Abogado.		
Objetivo	Modificar un recordatorio en la agenda del abogado.		
Dependencias	<ul style="list-style-type: none"> • RQF22 Consultar recordatorio. • RQF24 Modificar recordatorio 		
Descripción	Se modifica un recordatorio en la agenda de un abogado.		
Precondición	<p>El actor debe haber iniciado sesión.</p> <p>El recordatorio tiene que estar registrado.</p>		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de agenda. • El actor selecciona el recordatorio que desea modificar. • El sistema muestra la pantalla de modificar recordatorios. • El actor cambia la información del recordatorio. • El actor presiona el botón modificar. • La información se envía al API de Google Calendar. • El sistema modifica el recordatorio. • El sistema muestra un mensaje diciendo que el recordatorio se ha modificado correctamente. 		
Post condición	El recordatorio se modifica.		
Flujos alternos	<ol style="list-style-type: none"> 1. El actor deja espacios obligatorios en blanco. <ol style="list-style-type: none"> 1.1 El actor deja espacios en blanco. 		

	<p>1.2 El actor presiona el botón modificar.</p> <p>1.3 El sistema muestra un mensaje diciendo que faltan espacios por llenar.</p>
--	--

Tabla 85. Caso de uso 19, Modificar recordatorio.

Fuente: elaboración propia.

Código	CU-20	Nombre	Eliminar recordatorio
Actores	Abogado.		
Objetivo	Eliminar un recordatorio en la agenda del abogado.		
Dependencias	<ul style="list-style-type: none"> • RQF22 Consultar recordatorio. • RQF25 Eliminar recordatorio. 		
Descripción	Se elimina un recordatorio en la agenda de un abogado.		
Precondición	<p>El actor debe haber iniciado sesión.</p> <p>El recordatorio debe estar registrado.</p>		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de agenda. • El actor selecciona el recordatorio que desea eliminar. • El actor presiona el botón eliminar. • La acción se envía al API de Google Calendar. • El sistema elimina el recordatorio. • El sistema muestra un mensaje diciendo que el recordatorio se ha eliminado correctamente. 		
Post condición	El recordatorio se elimina.		
Flujos alternos			

Tabla 86. Caso de uso 20, Eliminar recordatorio.

Fuente: elaboración propia.

5.2.1.6 Módulo de mantenimiento

Los casos de uso de este módulo se detallan gráficamente por medio del siguiente diagrama, y se explican en las siguientes tablas.

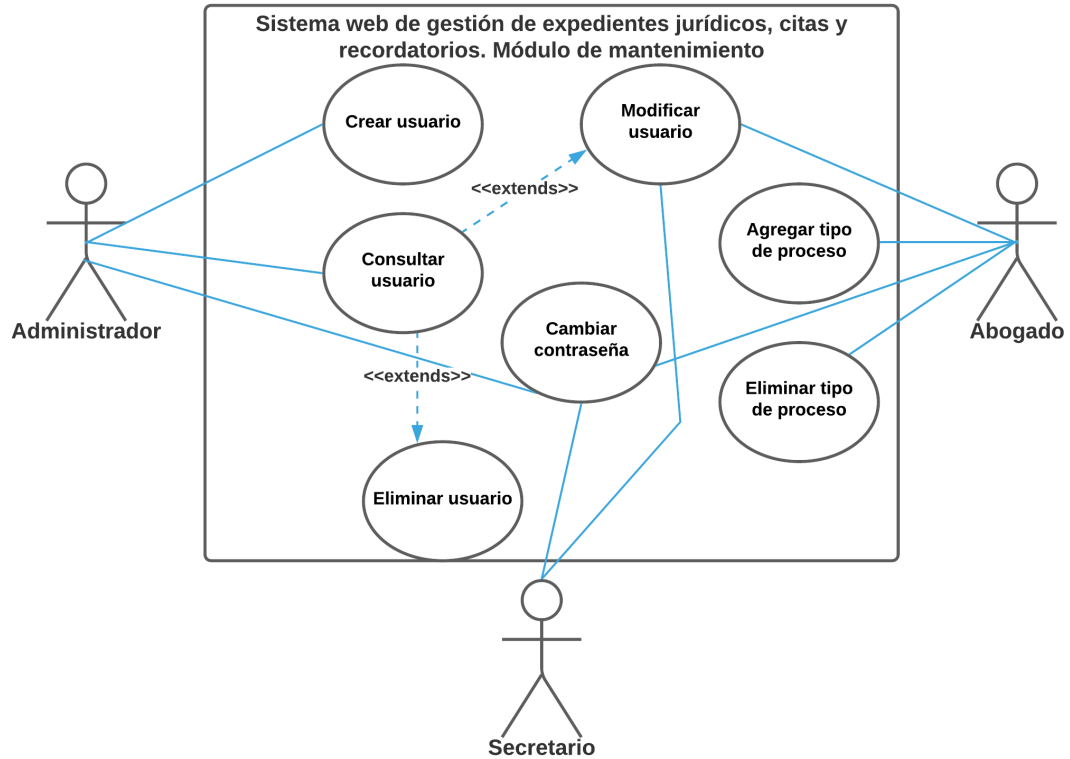


Figura 33. Diagrama de caso de uso 07, Módulo de mantenimiento.

Fuente: elaboración propia.

Código	CU-21	Nombre	Crear usuario
Actores	Administrador.		
Objetivo	Permitir al administrador crear un nuevo usuario en el sistema.		

Dependencias	<ul style="list-style-type: none"> • RQF02 Consultar usuario. • RQF03 Crear usuario. • RQF33 Enviar email al nuevo usuario.
Descripción	El actor crea un usuario nuevo.
Precondición	El actor tiene que haber iniciado sesión y tener rol de administrador.
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de mantenimiento. • El actor ingresa el número de identificación de usuario a registrar. • El sistema verifica que el usuario no esté registrado. • El actor ingresa la información del usuario a registrar. • El actor presiona el botón crear usuario. • El sistema agrega el usuario y crea el login del usuario. • El sistema envía un mensaje al correo del nuevo usuario con la información de inicio de sesión. • El sistema muestra un mensaje diciendo que el usuario se agregó correctamente.
Post condición	Se agrega el usuario.
Flujos alternos	<ol style="list-style-type: none"> 1. El usuario ya está agregado. <ol style="list-style-type: none"> 1.1 El actor presiona el botón buscar usuario. 1.2 El sistema muestra el formulario con la información del usuario lista para ser modificada. 2. El actor deja espacios en blanco. <ol style="list-style-type: none"> 2.1 El actor deja espacios en blanco. 2.2 El actor presiona el botón de agregar usuario. 2.3 El sistema muestra un mensaje diciendo que hacen falta espacios por llenar.

Tabla 87. Caso de uso 21, Crear usuario.

Fuente: elaboración propia.

Código	CU-22	Nombre	Consultar usuario
Actores	Administrador.		
Objetivo	Permitir al administrador buscar un usuario del sistema.		
Dependencias	<ul style="list-style-type: none"> • RQF02 Consultar usuario. 		
Descripción	El actor puede buscar algún usuario en el sistema.		
Precondición	<p>El actor debe haber iniciado sesión y tener rol de administrador.</p> <p>El usuario tiene que estar registrado.</p>		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor ingresa al módulo de mantenimiento. • El actor digita el usuario del usuario a buscar. • El sistema realiza la búsqueda. • El sistema muestra la información del usuario. 		
Post condición	Se muestra la información de la búsqueda.		
Flujos alternos	<ol style="list-style-type: none"> 1. El usuario no se encontró. <ol style="list-style-type: none"> 1.1 El sistema realiza la búsqueda. 1.2 El sistema muestra el formulario listo para que se registre el nuevo usuario. 		

Tabla 88. Caso de uso 22, Consultar usuario.

Fuente: elaboración propia.

Código	CU-23	Nombre	Modificar usuario
Actores	Administrador, Abogado.		
Objetivo	Cambiar información de su usuario.		
Dependencias	<ul style="list-style-type: none"> • RQF02 Consultar usuario. • RQF04 Modificar usuario. 		

Descripción	El actor modifica información de su usuario.
Precondición	El usuario debe haber iniciado sesión.
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de mantenimiento. • El actor se posiciona en la parte de modificación información de su usuario. • El actor cambia la información. • El sistema realiza los cambios. • El sistema muestra un mensaje diciendo que los cambios se realizaron con éxito.
Post condición	Los cambios se realizan.
Flujos alternos	<ol style="list-style-type: none"> 1. El actor es administrador. <ol style="list-style-type: none"> 1.1 El actor digita el usuario del usuario del que desea cambiar la información. 1.2 El sistema realiza la búsqueda. 1.3 El sistema muestra la información del usuario. 1.4 El actor hace los cambios en la información. 1.5 El sistema realiza los cambios. 1.6 El sistema muestra un mensaje diciendo que los cambios se realizaron con éxito.

Tabla 89. Caso de uso 23, Modificar usuario.

Fuente: elaboración propia.

Código	CU-24	Nombre	Eliminar usuario
Actores	Administrador.		
Objetivo	Eliminar un usuario del sistema.		
Dependencias	<ul style="list-style-type: none"> • RQF02 Consultar usuario. 		

	<ul style="list-style-type: none"> • RQF05 Eliminar usuario.
Descripción	El actor elimina un usuario del sistema.
Precondición	El actor debe haber iniciado sesión. El usuario debe estar registrado.
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de mantenimiento. • El actor realiza la búsqueda del usuario. • El actor presiona el botón de eliminar usuario. • El sistema cambia el estado del usuario de Autorizado a Denegado, para que ya no tenga permisos dentro de la aplicación. • El sistema muestra un mensaje diciendo que el usuario se eliminó.
Post condición	El usuario se elimina.
Flujos alternos	

Tabla 90. Caso de uso 24, Eliminar usuario.

Fuente: elaboración propia.

Código	CU-25	Nombre	Agregar tipo de proceso
Actores	Abogado.		
Objetivo	Agregar un tipo de proceso para seleccionar desde el módulo de expedientes.		
Dependencias	<ul style="list-style-type: none"> • RQF07 Agregar tipo de proceso. • RQF30 Consultar tipo de proceso. 		
Descripción	El actor agrega un nuevo tipo de proceso.		
Precondición	El actor debe haber iniciado sesión.		

Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de mantenimiento. • El actor se posiciona en la parte de agregar tipo de proceso. • El actor digita el nuevo tipo de proceso. • El actor presiona el botón agregar nuevo tipo de proceso. • El sistema agrega el tipo de proceso. • El sistema muestra un mensaje diciendo que el tipo de proceso se agregó con éxito.
Post condición	El tipo de proceso se agrega.
Flujos alternos	<ol style="list-style-type: none"> 1. El tipo de proceso ya está registrado. <ol style="list-style-type: none"> 1.1 El actor presiona el botón de agregar tipo de proceso. 1.2 El sistema muestra un mensaje diciendo que el tipo de proceso ya existe.

Tabla 91. Caso de uso 25, Agregar tipo de proceso.

Fuente: elaboración propia.

Código	CU-26	Nombre	Eliminar tipo de proceso
Actores	Abogado.		
Objetivo	Eliminar un tipo de proceso.		
Dependencias	<ul style="list-style-type: none"> • RQF08 Eliminar tipo de proceso. • RQF30 Consultar tipo de proceso. 		
Descripción	El actor elimina un tipo de proceso almacenado en el sistema.		
Precondición	El actor debe haber iniciado sesión.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de mantenimiento. • El actor se posiciona en la sección de tipos de proceso. • El actor selecciona desde la barra desplegable el tipo de proceso que quiere eliminar. 		

	<ul style="list-style-type: none"> • El actor presiona el botón eliminar. • El sistema elimina el tipo de proceso. • El sistema muestra un mensaje diciendo que el tipo de proceso se eliminó correctamente.
Post condición	El tipo de proceso se elimina.
Flujos alternos	

Tabla 92. Caso de uso 26, Eliminar tipo de proceso.

Fuente: elaboración propia.

Código	CU-27	Nombre	Cambiar contraseña
Actores	Abogado, Administrador.		
Objetivo	Cambiar la contraseña del actor.		
Dependencias	<ul style="list-style-type: none"> • RQF31 Cambiar contraseña. 		
Descripción	El actor cambia su propia contraseña.		
Precondición	El actor debe haber iniciado sesión.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de mantenimiento. • El actor se posiciona en la sección de cambio de contraseña. • El actor digita la contraseña actual y la nueva contraseña. • El sistema verifica que la contraseña actual almacenada en la base de datos coincida con la digitada. • El sistema realiza el cambio de contraseña. • El sistema muestra un mensaje diciendo que la contraseña se cambió correctamente. 		
Post condición	La contraseña se cambia.		

Flujos alternos	<ol style="list-style-type: none"> 1. La contraseña actual digitada no coincide con la almacenada. <ol style="list-style-type: none"> 1.1 El actor presiona el botón cambiar contraseña. 1.2 El sistema muestra un mensaje diciendo que la contraseña actual digitada es errónea.
------------------------	---

Tabla 93. Caso de uso 27, Cambiar contraseña.

Fuente: elaboración propia.

5.2.1.7 Módulo de Honorarios

Los casos de uso y su descripción se detallan a continuación en el siguiente diagrama y las siguientes tablas.

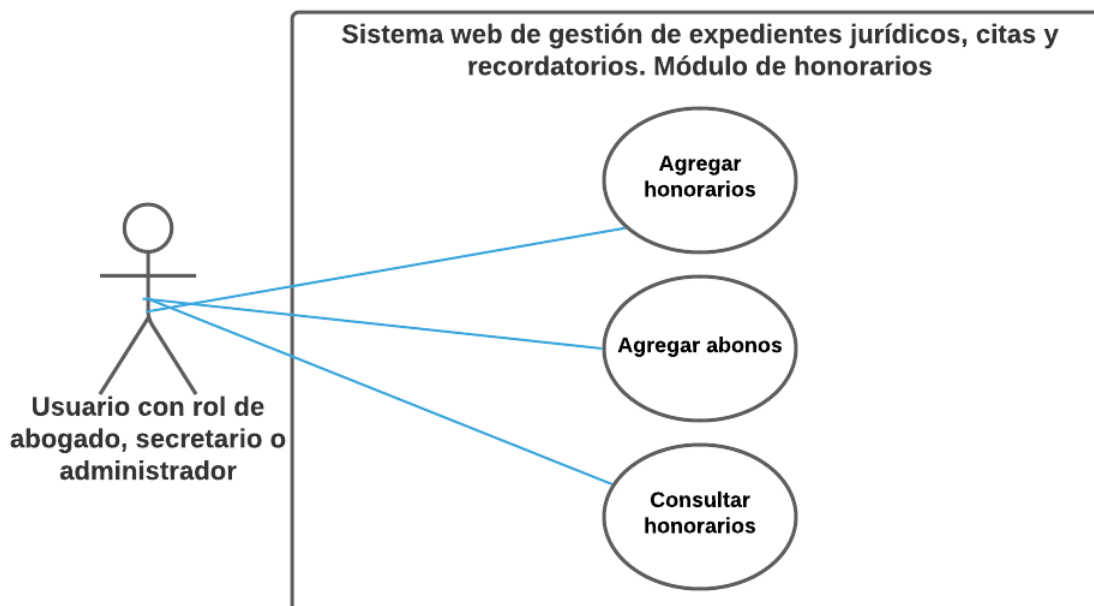


Figura 34. Diagrama de caso de uso 08, Módulo de honorarios.

Fuente: elaboración propia.

Código	CU-28	Nombre	Agregar honorarios
Actores	Administrador, Abogado.		
Objetivo	Agregar un honorario.		
Dependencias	<ul style="list-style-type: none"> • RQF35 Agregar honorario. • RQF34 Consultar honorarios. • CU-30 Consultar honorario. 		
Descripción	El actor agrega un honorario al registro.		
Precondición	El actor debe haber iniciado sesión.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor ingresa al módulo de honorarios. • El actor selecciona el botón de crear honorario. • El actor rellena el formulario y presiona el botón agregar honorario. • El sistema agrega el honorario a la base de datos. • El sistema muestra un mensaje diciendo que el honorario se agregó. 		
Post condición	El honorario se agrega.		
Flujos alternos			

Tabla 94. Caso de uso 28, Agregar honorarios.

Fuente: elaboración propia.

Código	CU-29	Nombre	Agregar abonos
Actores	Administrador, Abogado.		
Objetivo	Actualizar el registro de honorarios.		
Dependencias	<ul style="list-style-type: none"> • RQF34 Consultar honorarios. • RQF36 Agregar abonos. • CU-30 Consultar honorarios. 		

Descripción	El actor agrega abonos al honorario y actualiza el registro.
Precondición	El actor tiene que haber iniciado sesión. El honorario debe estar registrado.
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor ingresa al módulo de honorarios. • El actor busca el honorario que desea actualizar. • El actor selecciona el honorario y presiona el botón de agregar abono. • El actor rellena la información. • El actor presiona el botón agregar abono. • El abono se agrega.
Post condición	El honorario se actualiza.
Flujos alternos	

Tabla 95. Caso de uso 29, Agregar abonos.

Fuente: elaboración propia.

Código	CU-30	Nombre	Consultar honorarios
Actores	Administrador, Abogado.		
Objetivo	Buscar un honorario en los registros.		
Dependencias	<ul style="list-style-type: none"> • RQF34 Consultar honorarios. 		
Descripción	Se busca un honorario.		
Precondición	El actor debe haber iniciado sesión.		
Flujo principal/ normal de los eventos	<ul style="list-style-type: none"> • El actor entra al módulo de honorarios. • El actor digita el honorario a buscar por el número de expediente relacionado. 		

	<ul style="list-style-type: none"> • El sistema muestra los resultados de la búsqueda.
Post condición	Se muestran los resultados de la búsqueda.
Flujos alternos	

Tabla 96. Caso de uso 30, Consultar honorarios.

Fuente: elaboración propia.

5.2.2 Diagrama de clases

Un diagrama de clases muestra de forma gráfica las clases que posee el sistema, así como sus atributos, operaciones o funciones y las relaciones que existen entre ellas.

El proyecto cuenta con un grupo de clases que son generadas automáticamente por el Entity Framework al hacer la conexión con la base de datos, y cuenta con clases creadas de forma manual que se utilizan para manejar los datos que se necesitan en cada vista, a estas clases se les llama ViewModel. Los diagramas de clase se representan en las siguientes figuras.

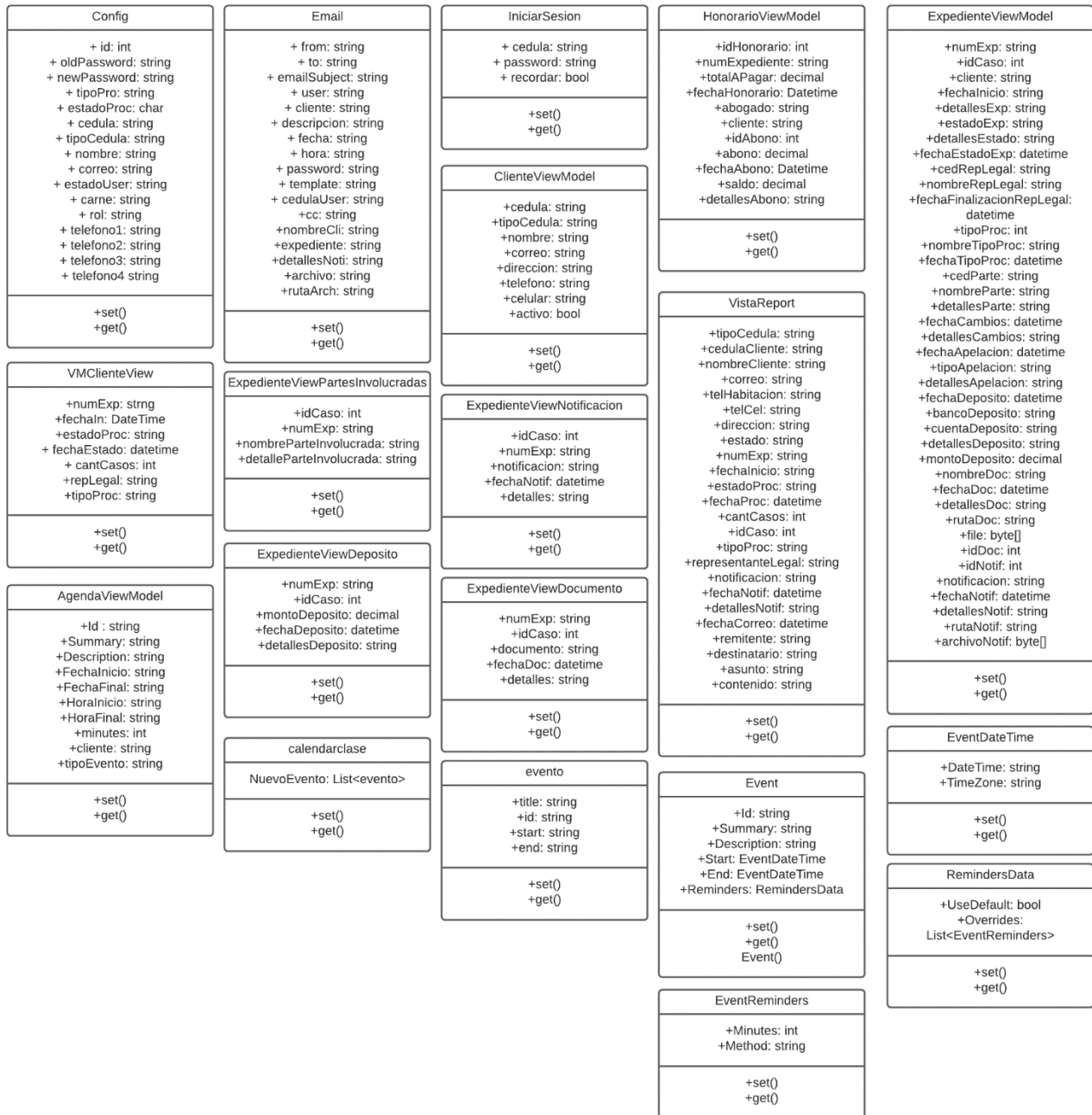


Figura 35. Diagrama de clases, ViewModel.

Fuente: elaboración propia.

5.2.3 Diagrama de actividades

Un diagrama de actividades es la representación gráfica de un proceso o un caso de uso del sistema que se está diseñando.

Los diagramas de actividad de cada módulo se muestran a continuación.

Manejos de sesión.

Iniciar sesión

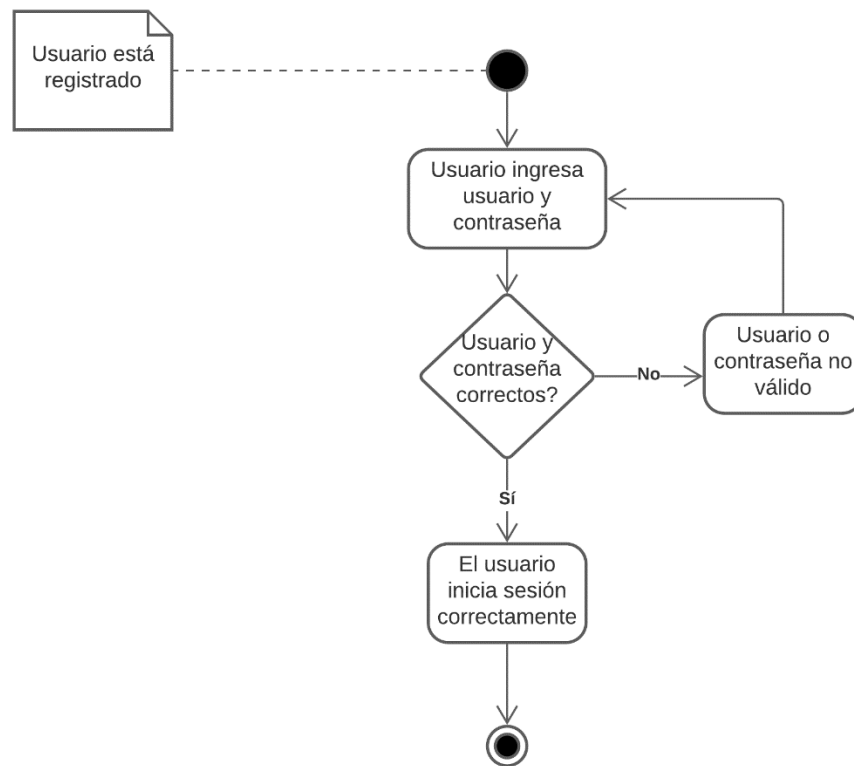


Figura 37. Diagrama de actividad, Iniciar sesión.

Fuente: elaboración propia.

Recuperar contraseña

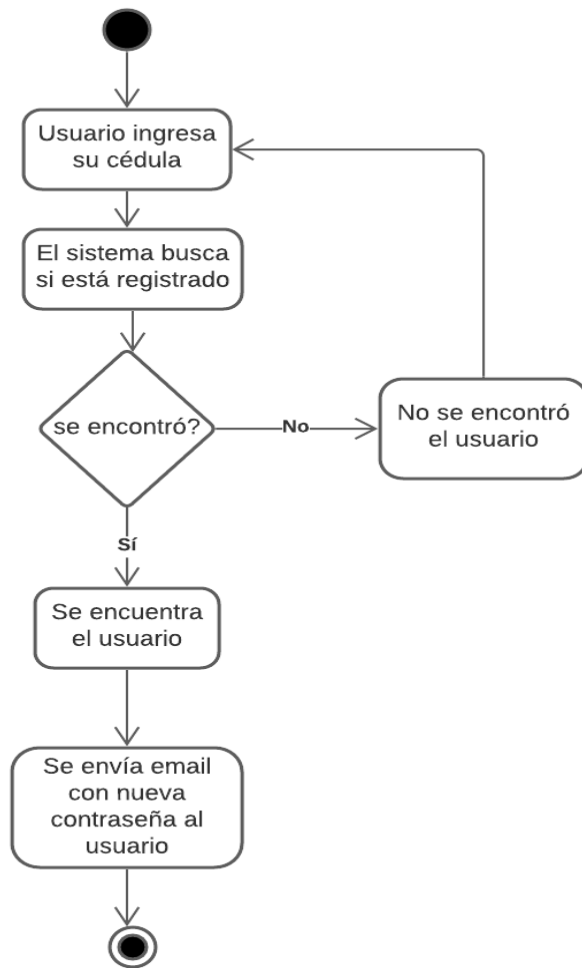


Figura 38. Diagrama de actividad, Recuperar contraseña.

Fuente: elaboración propia.

Cerrar sesión



Figura 39. Diagrama de actividad, Cerrar sesión.

Fuente: elaboración propia.

Módulo de clientes

Agregar cliente

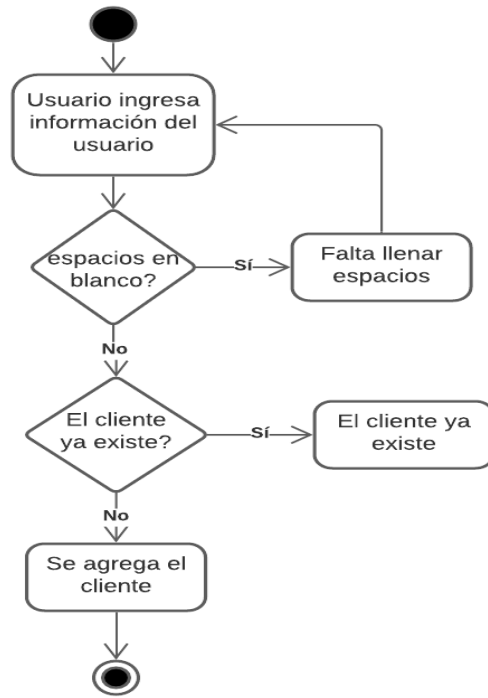


Figura 40. Diagrama de actividad, Agregar clientes.

Fuente: elaboración propia.

Consultar cliente

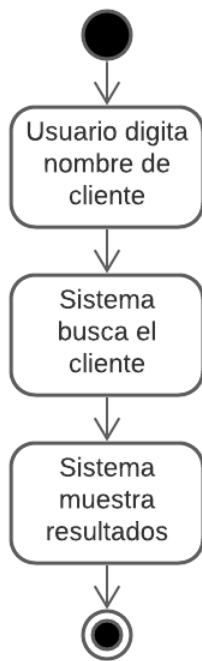


Figura 41. Diagrama de actividad, Consultar cliente.

Fuente: elaboración propia.

Modificar cliente

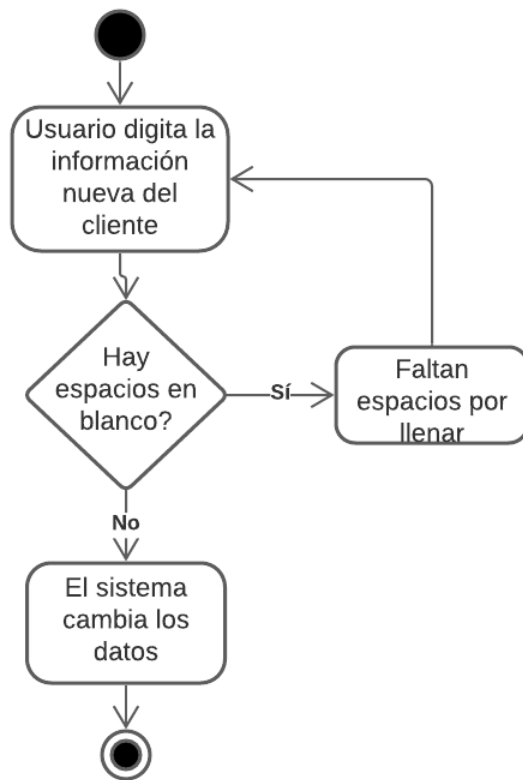


Figura 42. Diagrama de actividad, Modificar cliente.

Fuente: elaboración propia.

Eliminar cliente

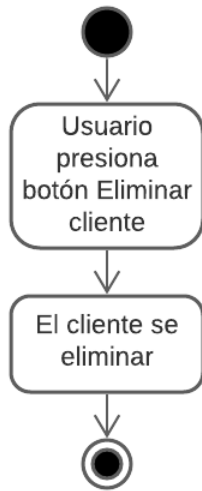


Figura 43. Diagrama de actividad, Eliminar cliente.

Fuente: elaboración propia.

Módulo de reportes

Generar reporte

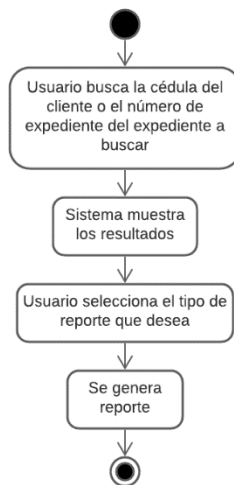


Figura 44. Diagrama de actividad, Generar reporte.

Fuente: elaboración propia.

Módulo de Honorarios

Agregar honorario

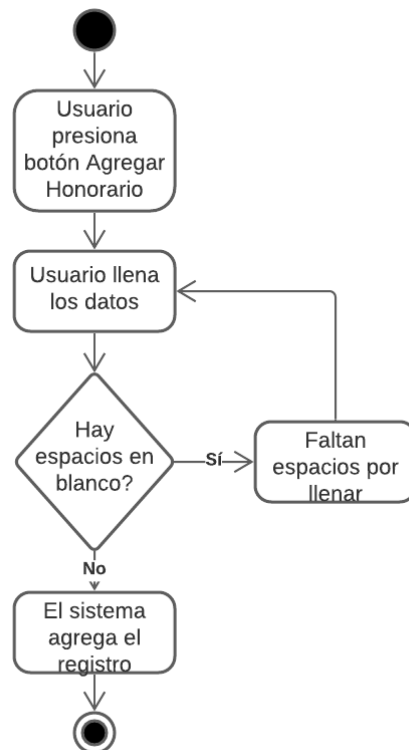


Figura 45. Diagrama de actividad, Agregar honorario.

Fuente: elaboración propia.

Agregar abonos

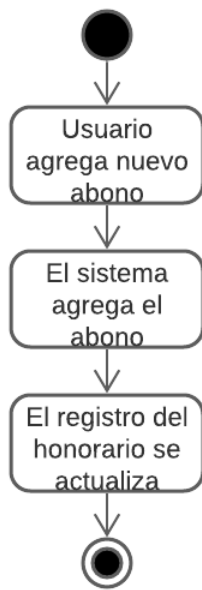


Figura 46. Diagrama de actividad, Agregar abonos.

Fuente: elaboración propia.

Consultar honorario

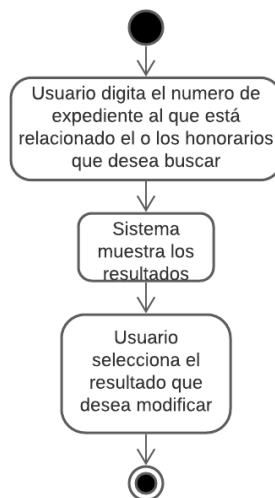


Figura 47. Diagrama de actividad, Consultar honorario.

Fuente: elaboración propia.

Módulo expedientes

Agregar expediente

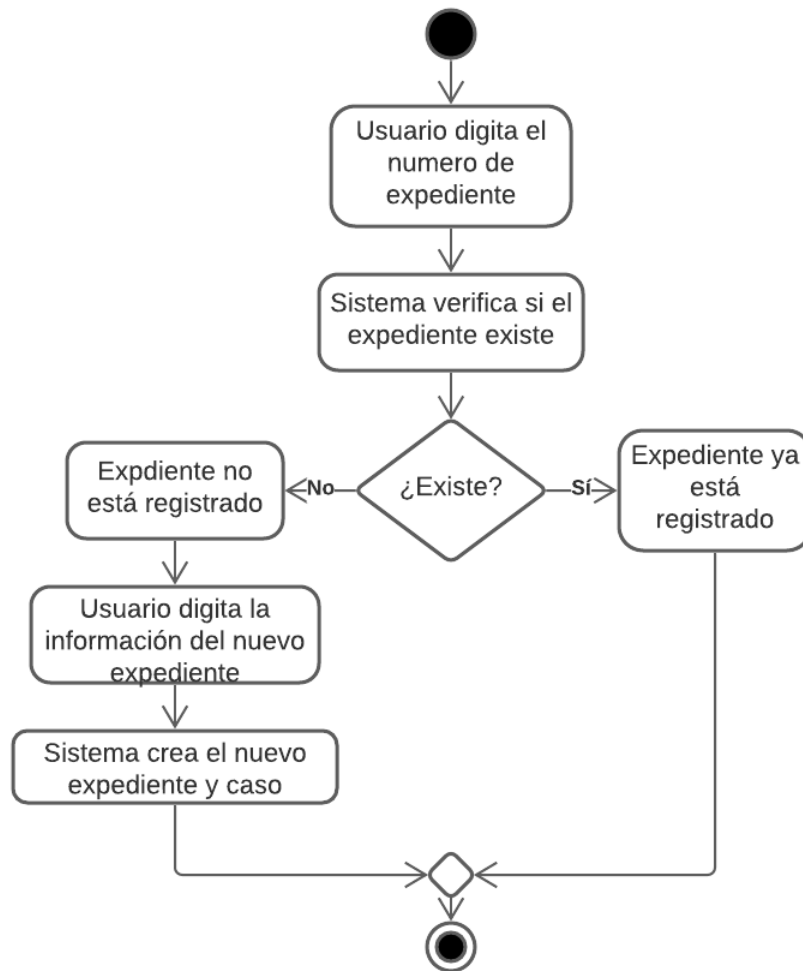


Figura 48. Diagrama de actividad, Agregar expediente.

Fuente: elaboración propia.

Modificar expediente

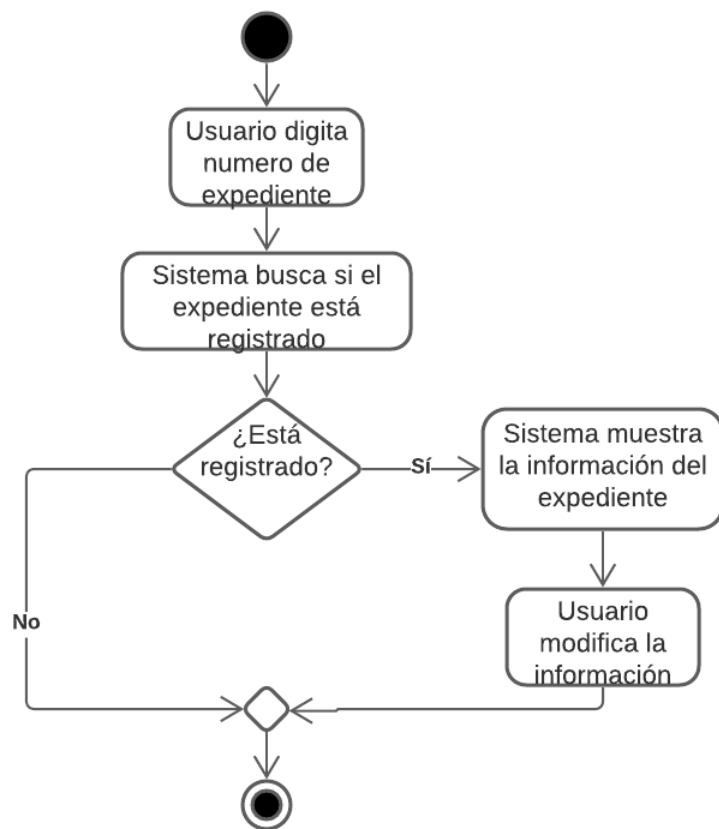


Figura 49. Diagrama de actividad, Modificar expediente.

Fuente: elaboración propia.

Agregar caso

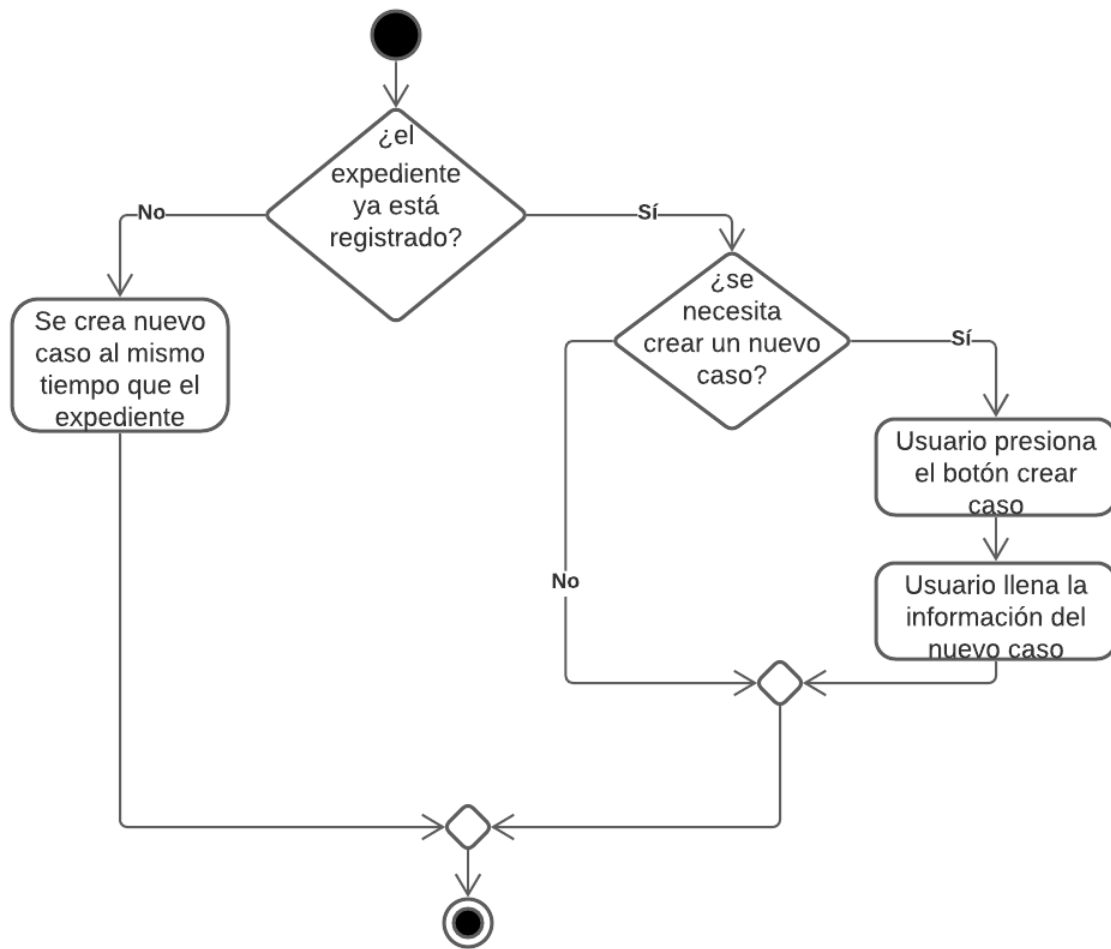


Figura 50. Diagrama de actividad, Agregar caso.

Fuente: elaboración propia.

Consultar expediente

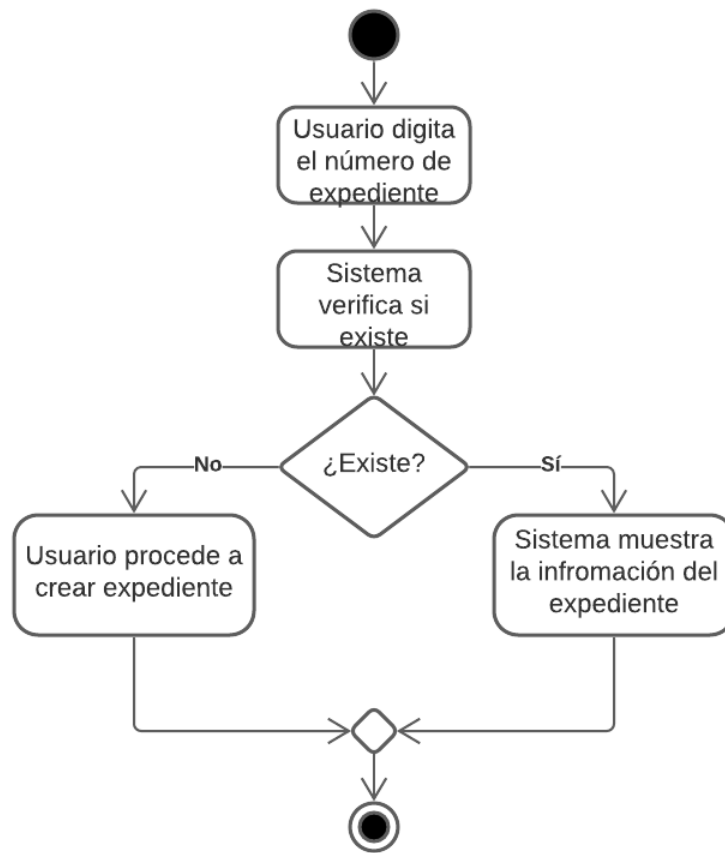


Figura 51. Diagrama de actividad, Consultar expediente.

Fuente: elaboración propia.

Módulo de mantenimiento

Crear usuario

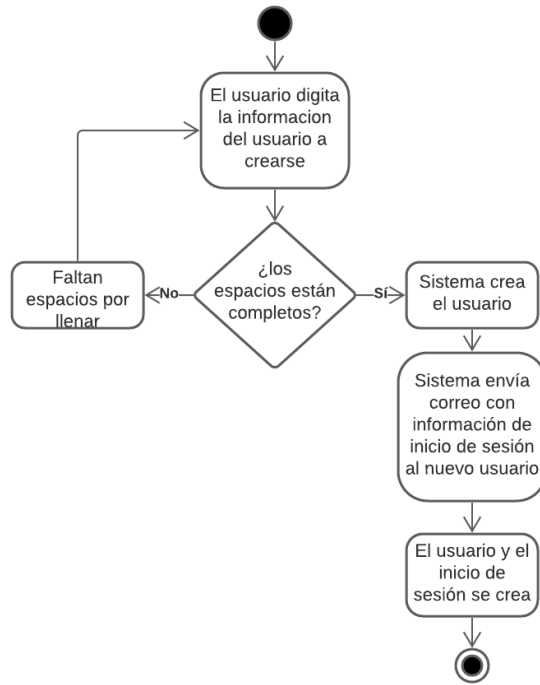


Figura 52. Diagrama de actividad, Crear usuario.

Fuente: elaboración propia.

Consultar usuario

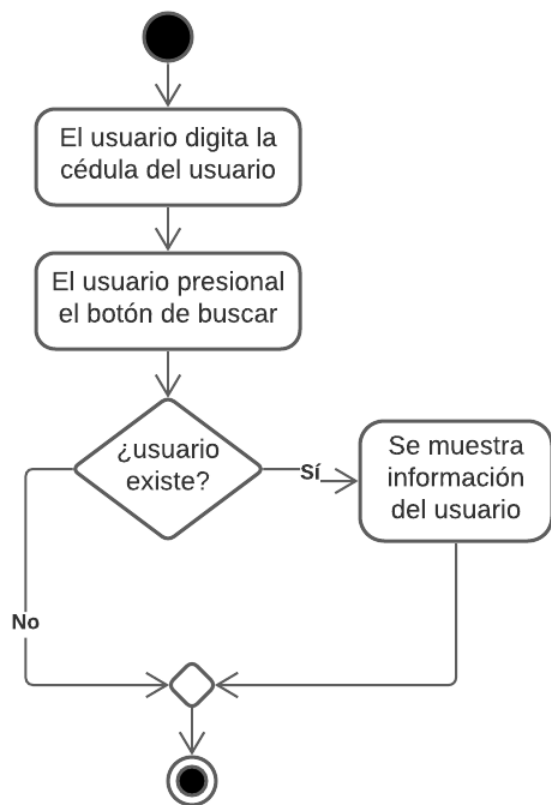


Figura 53. Diagrama de actividad, Consultar usuario.

Fuente: elaboración propia.

Modificar usuario

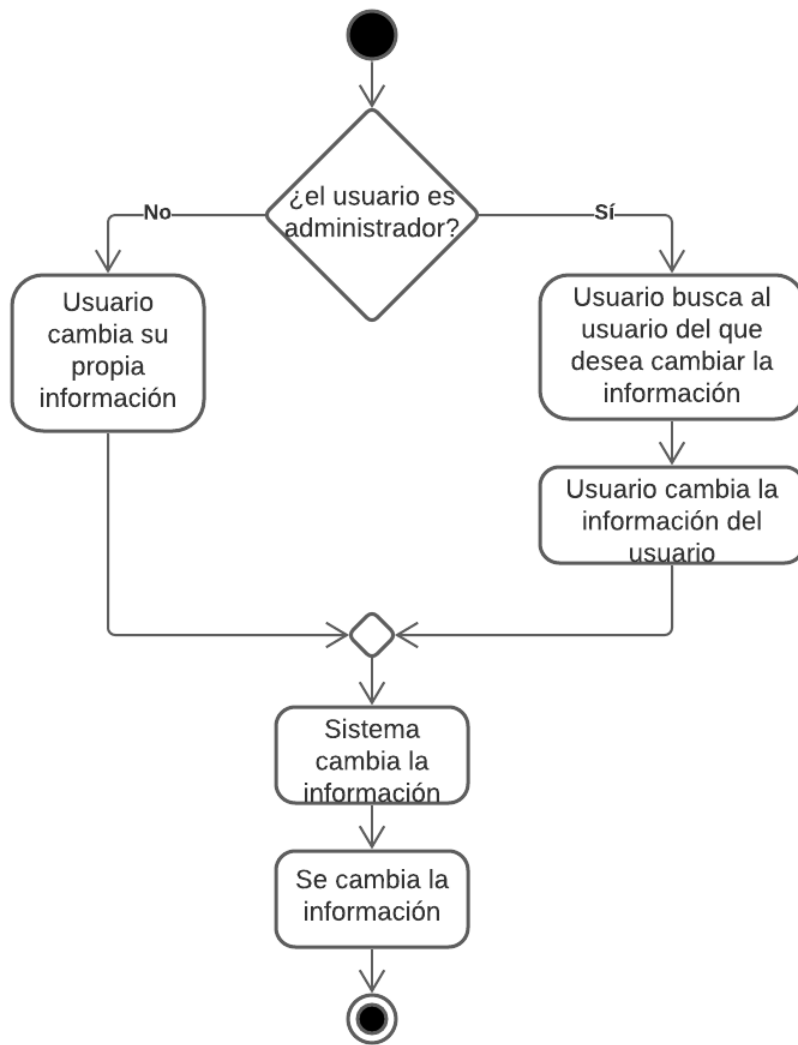


Figura 54. Diagrama de actividad, Modificar usuario.

Fuente: elaboración propia.

Eliminar usuario

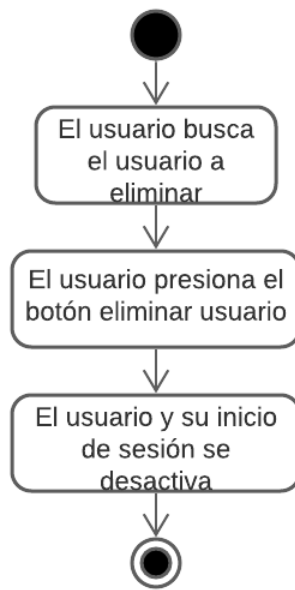


Figura 55. Diagrama de actividad, Eliminar usuario.

Fuente: elaboración propia.

Agregar tipo de proceso

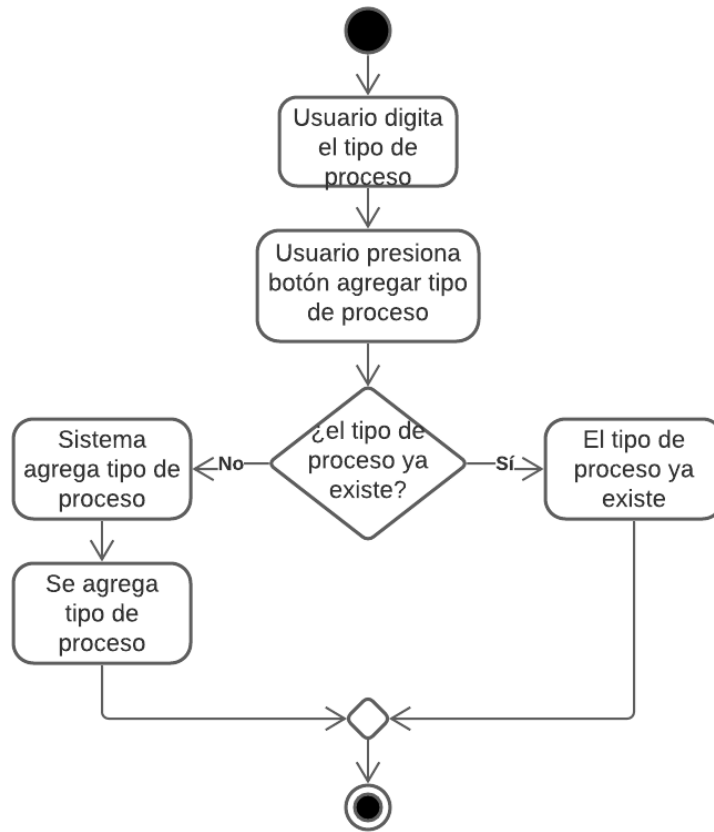


Figura 56. Diagrama de actividad, Agregar tipo de proceso.

Fuente: elaboración propia.

Eliminar tipo de proceso

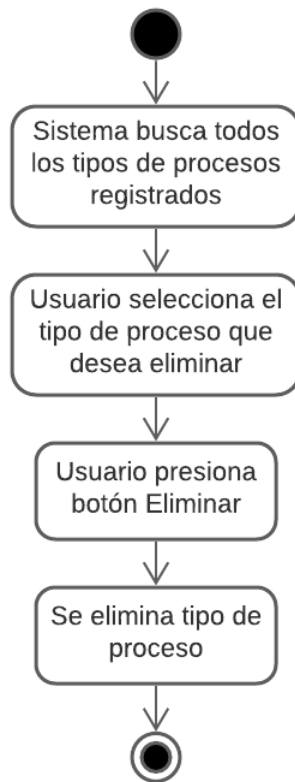


Figura 57. Diagrama de actividad, Consultar usuario.

Fuente: elaboración propia.

Cambiar contraseña

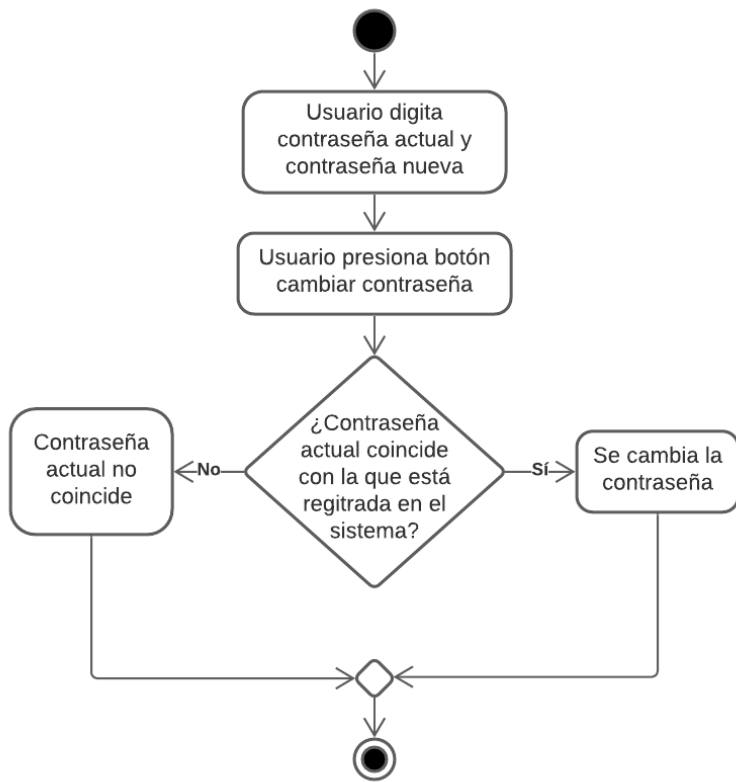


Figura 58. Diagrama de actividad, Consultar usuario.

Fuente: elaboración propia.

Módulo de agenda

Agregar cita

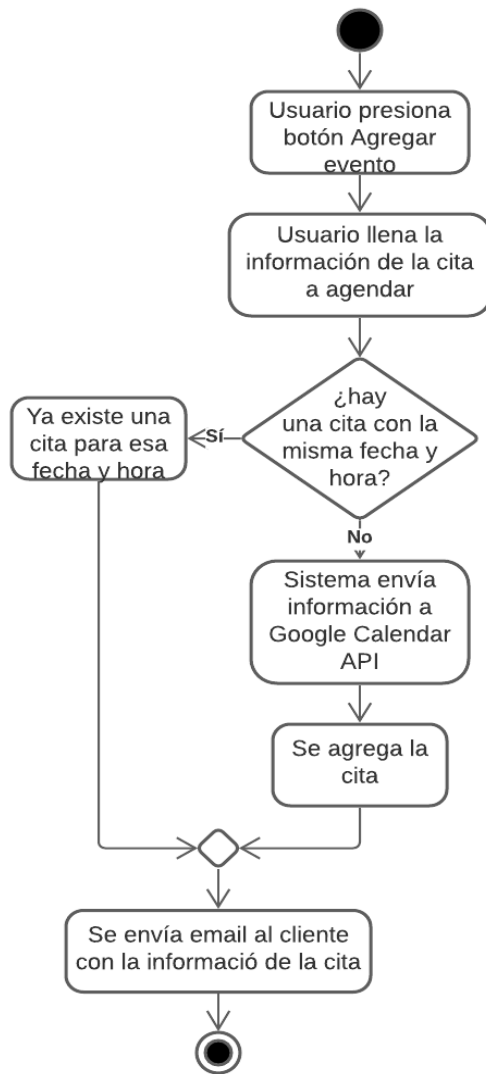


Figura 59. Diagrama de actividad, Agregar cita.

Fuente: elaboración propia.

Consultar cita

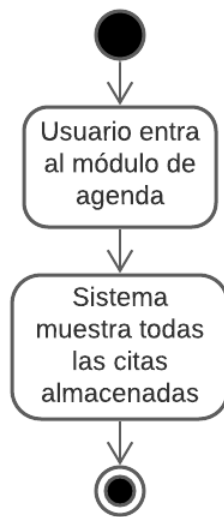


Figura 60. Diagrama de actividad, Consultar cita.

Fuente: elaboración propia.

Modificar cita

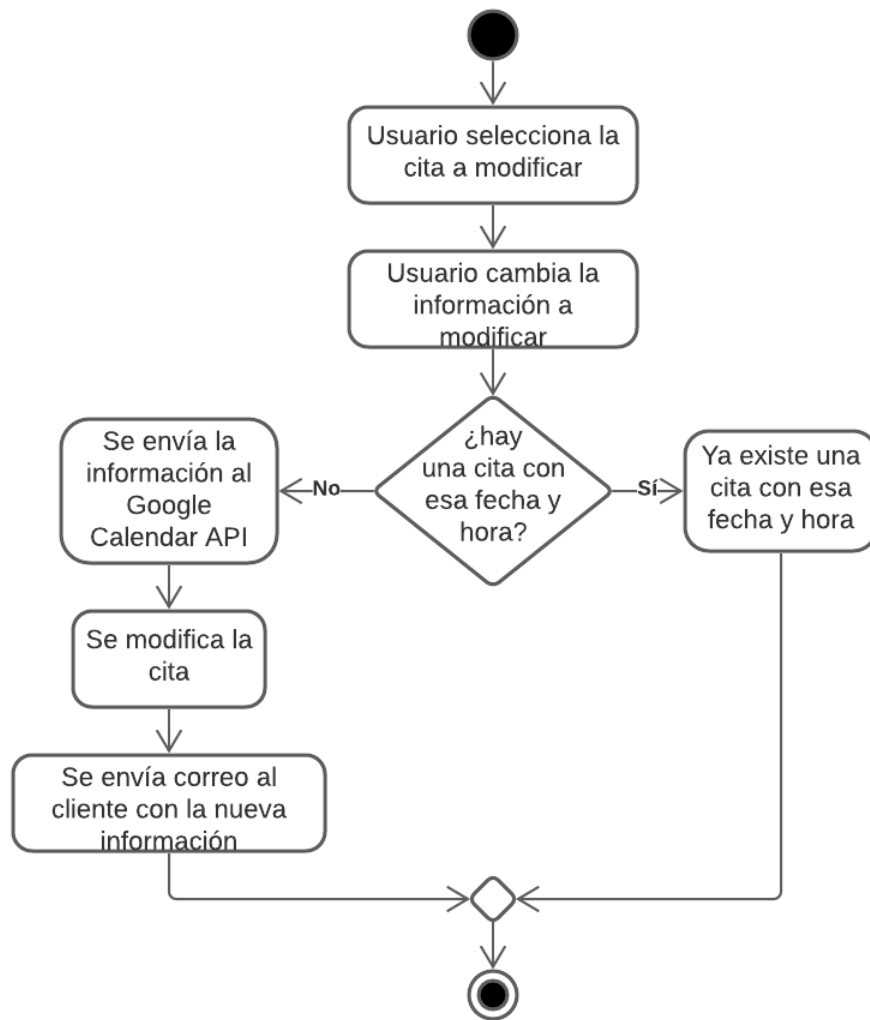


Figura 61. Diagrama de actividad, Modificar cita.

Fuente: elaboración propia.

Cancelar cita

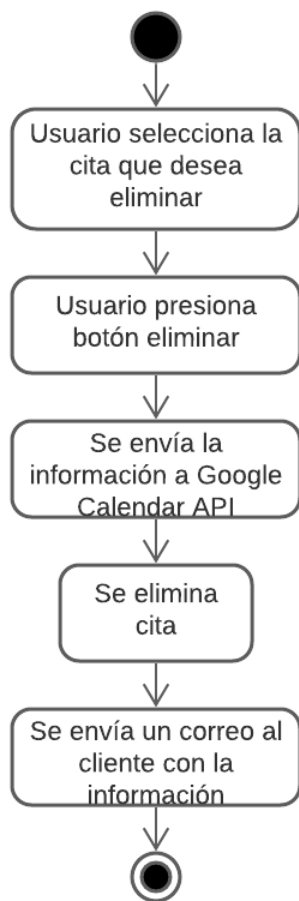


Figura 62. Diagrama de actividad, Cancelar cita.

Fuente: elaboración propia.

Agregar recordatorio

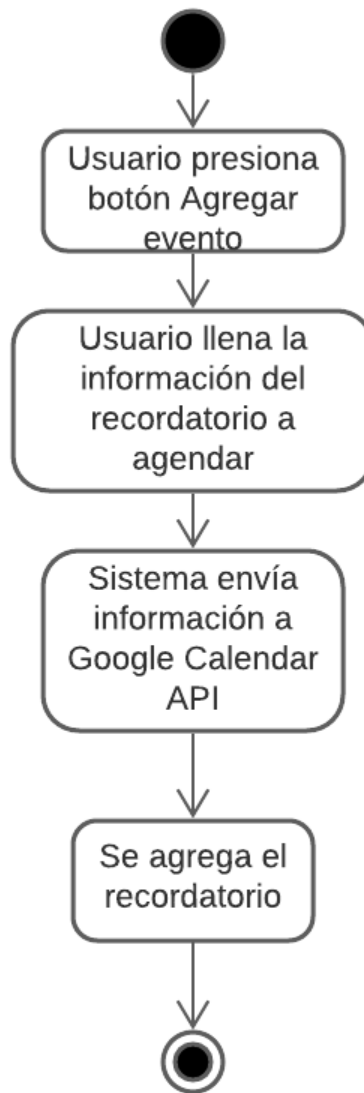


Figura 63. Diagrama de actividad, Agregar recordatorio.

Fuente: elaboración propia.

Consultar recordatorio

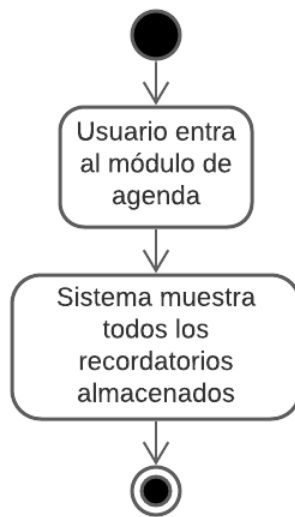


Figura 64. Diagrama de actividad, Consultar recordatorio.

Fuente: elaboración propia.

Modificar recordatorio

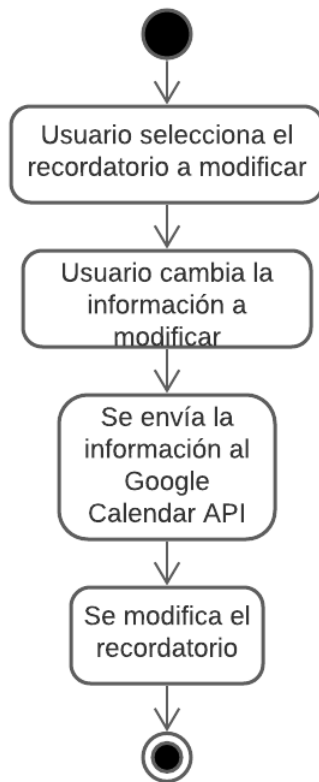


Figura 65. Diagrama de actividad, Modificar recordatorio.

Fuente: elaboración propia.

Eliminar recordatorio

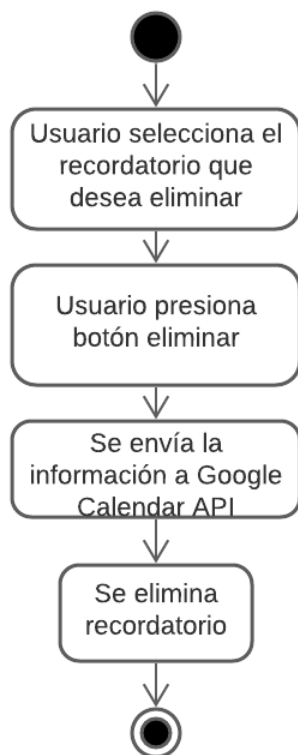


Figura 66. Diagrama de actividad, Eliminar recordatorio.

Fuente: elaboración propia.

5.2.4 Diagrama de secuencia

Este tipo de diagrama UML describe la interacción entre objetos del sistema, generalmente se utiliza para describir el proceso descrito en los casos de uso.

Los diagramas de secuencia de este proyecto son los siguientes:

Iniciar Sesion:

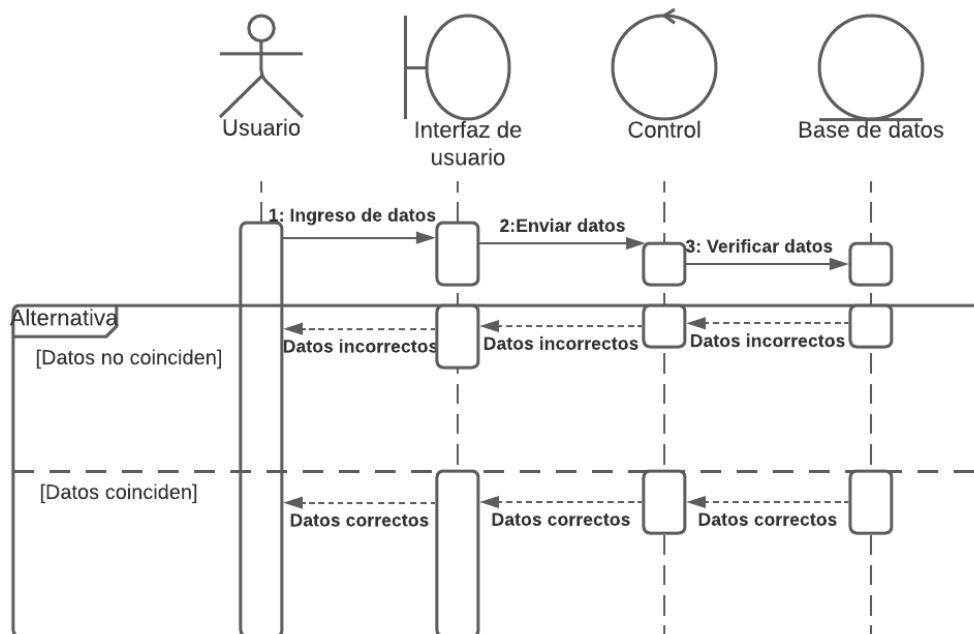


Figura 67. Diagrama de secuencia, Iniciar sesión.

Fuente: elaboración propia

Recuperar contraseña

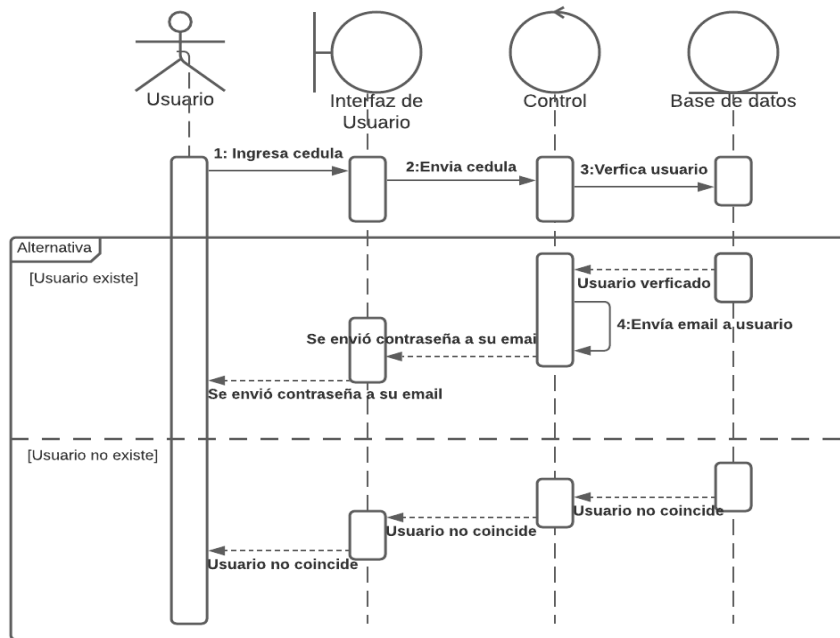


Figura 68. Diagrama de secuencia, Recuperar contraseña

Fuente: elaboración propia

Agregar cliente

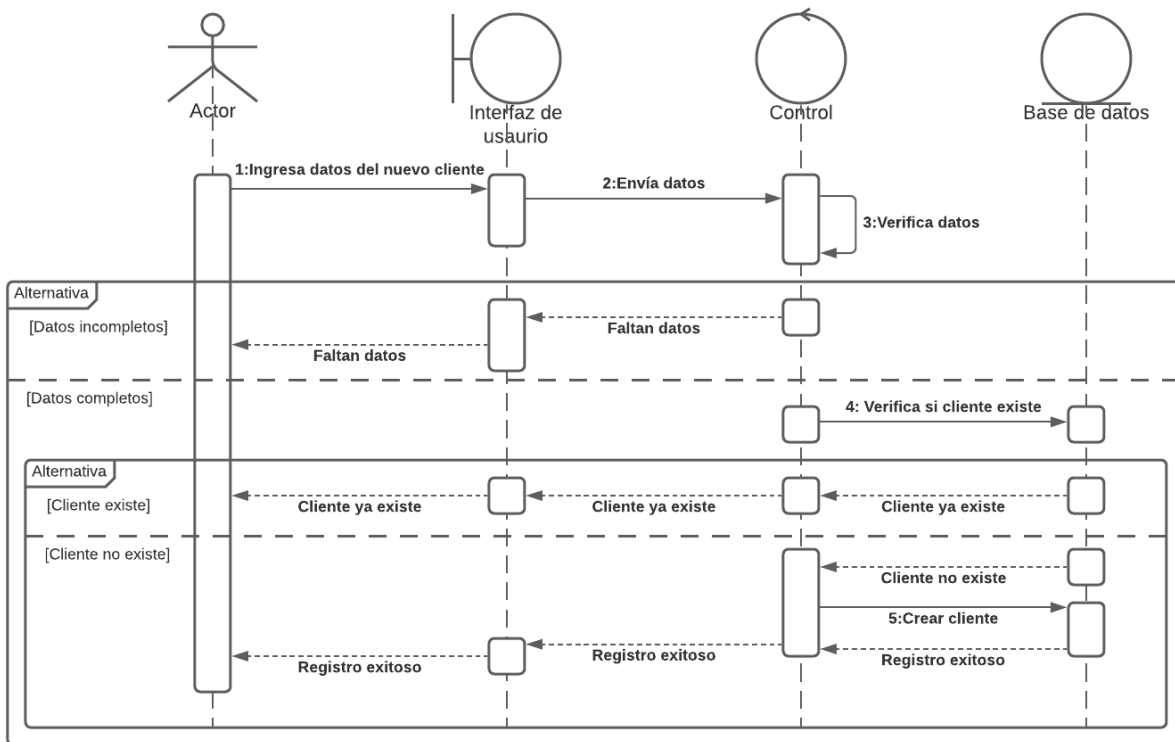


Figura 69. Diagrama de secuencia, Agregar cliente.

Fuente: elaboración propia

Buscar cliente

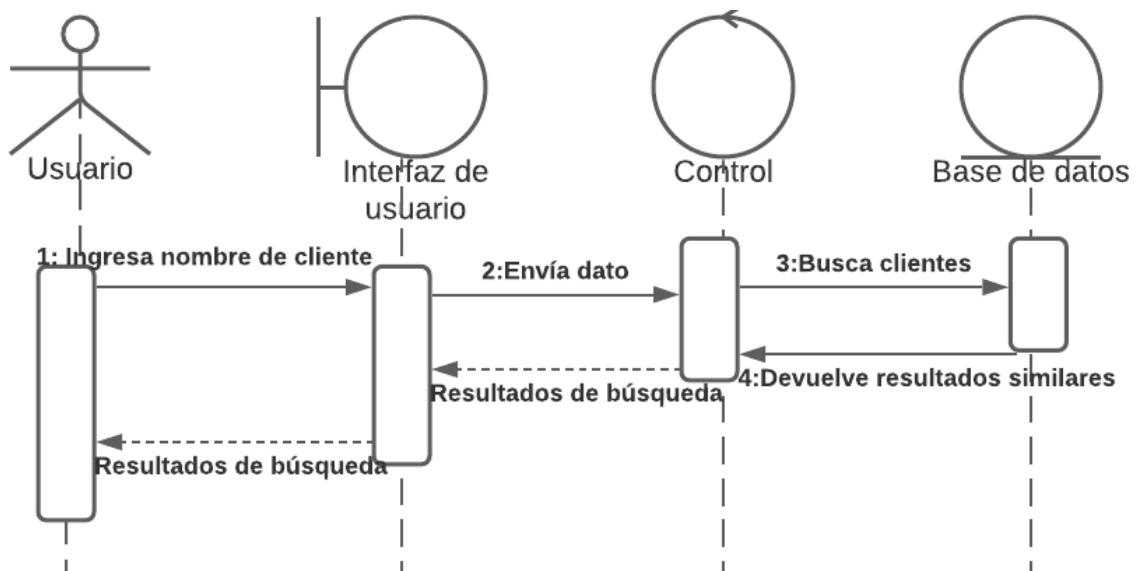


Figura 70. Diagrama de secuencia, Buscar cliente.

Fuente: elaboración propia.

Eliminar cliente

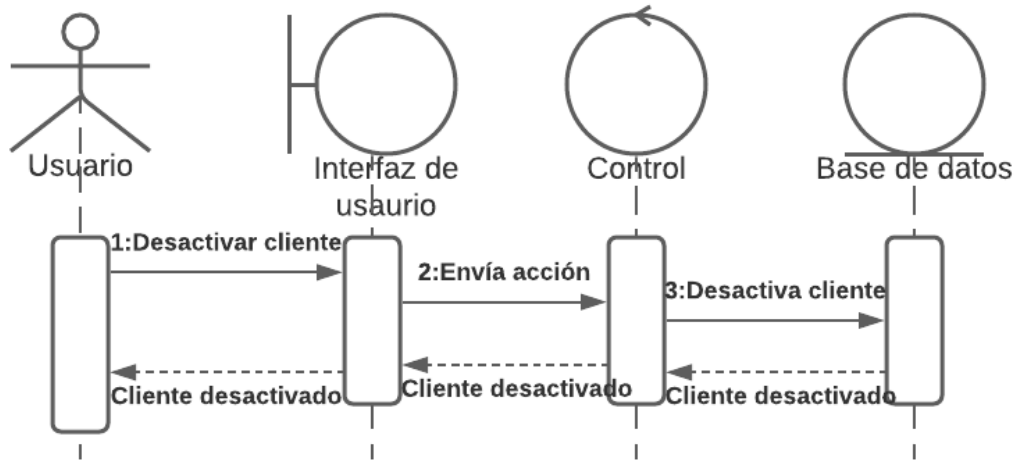


Figura 71. Diagrama de secuencia, Eliminar cliente.

Fuente: elaboración propia.

Modificar cliente

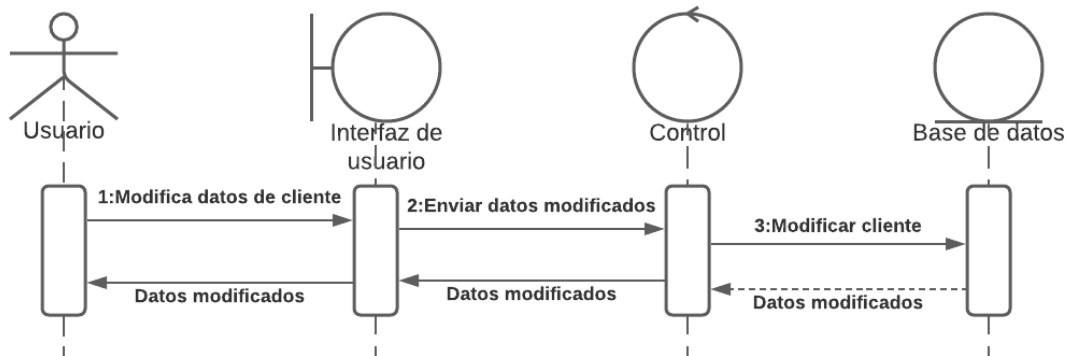


Figura 72. Diagrama de secuencia, Modificar cliente.

Fuente: elaboración propia.

Crear usuario

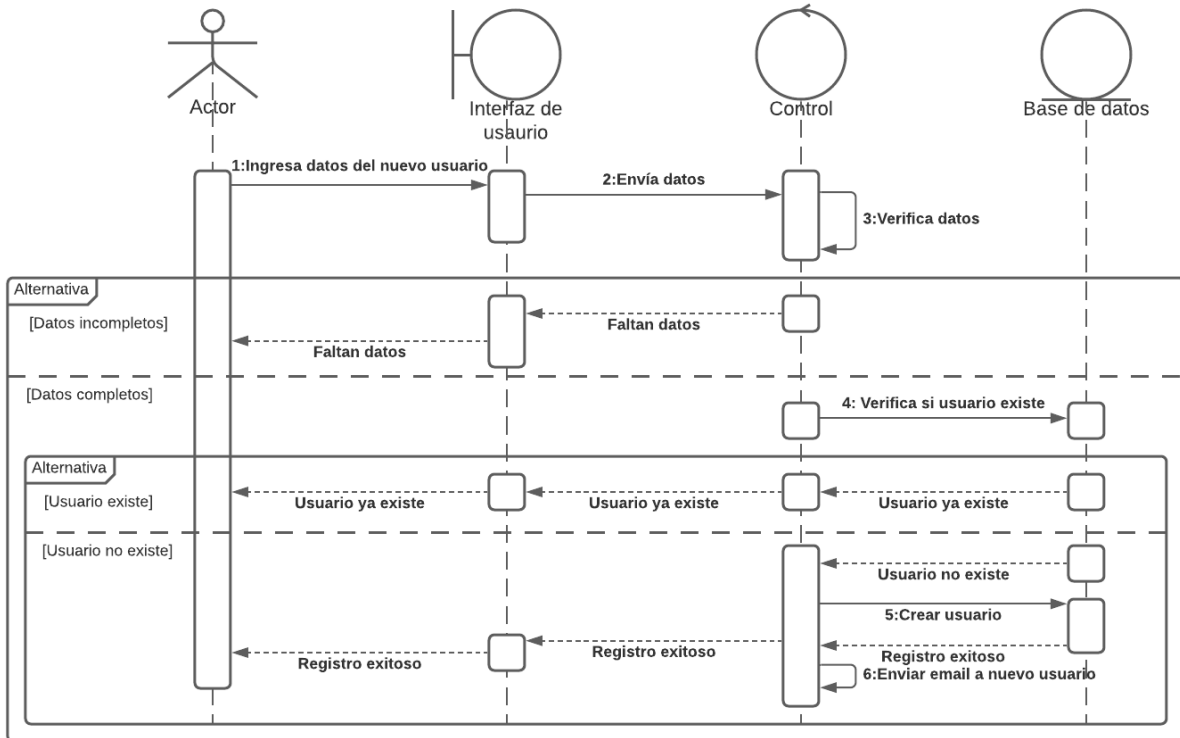


Figura 73. Diagrama de secuencia, Crear usuario.

Fuente: elaboración propia.

Consultar cliente

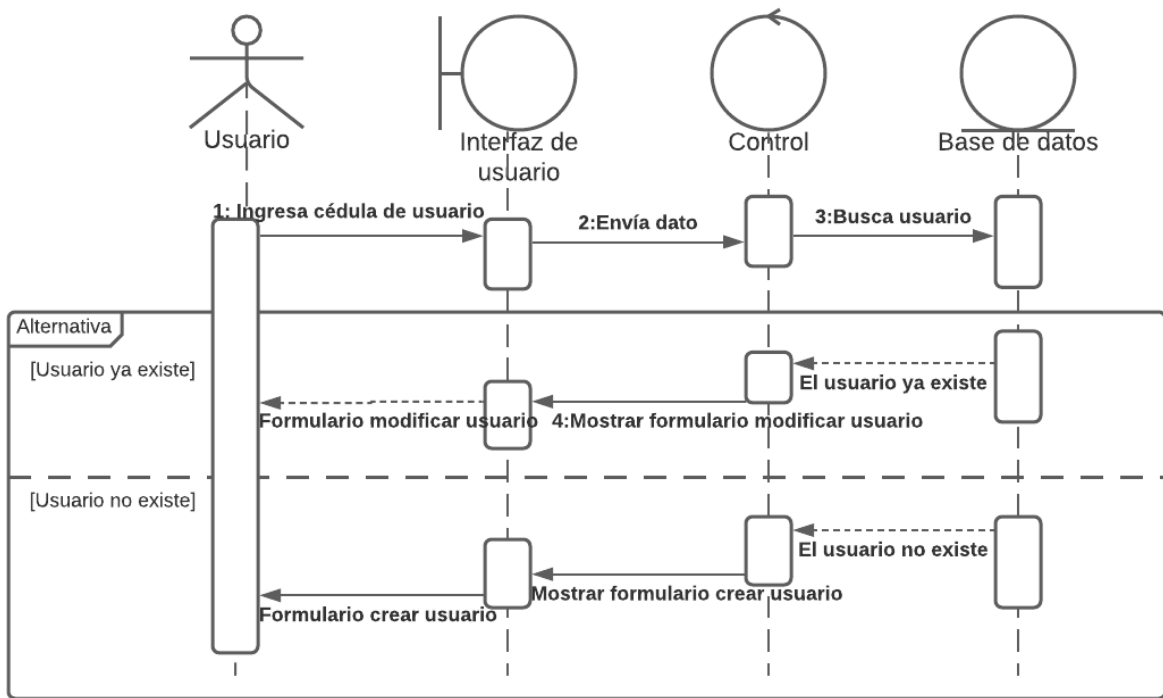


Figura 74. Diagrama de secuencia, Consultar usuario.

Fuente: elaboración propia

Eliminar cliente

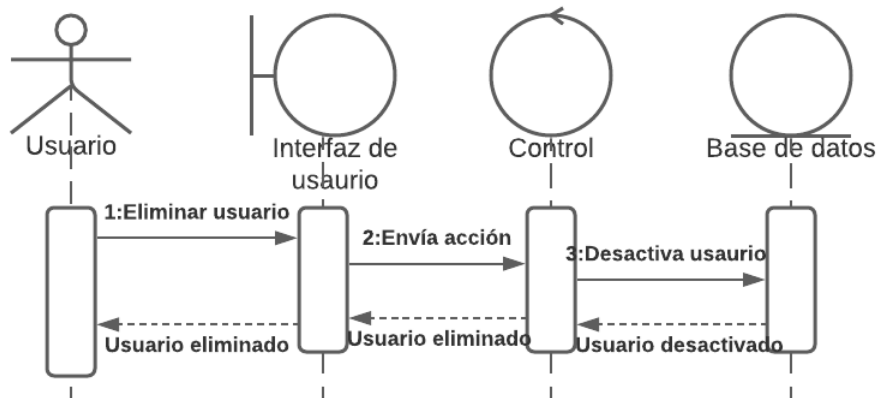


Figura 75. Diagrama de secuencia, Eliminar cliente.

Fuente: elaboración propia.

Modificar usuario

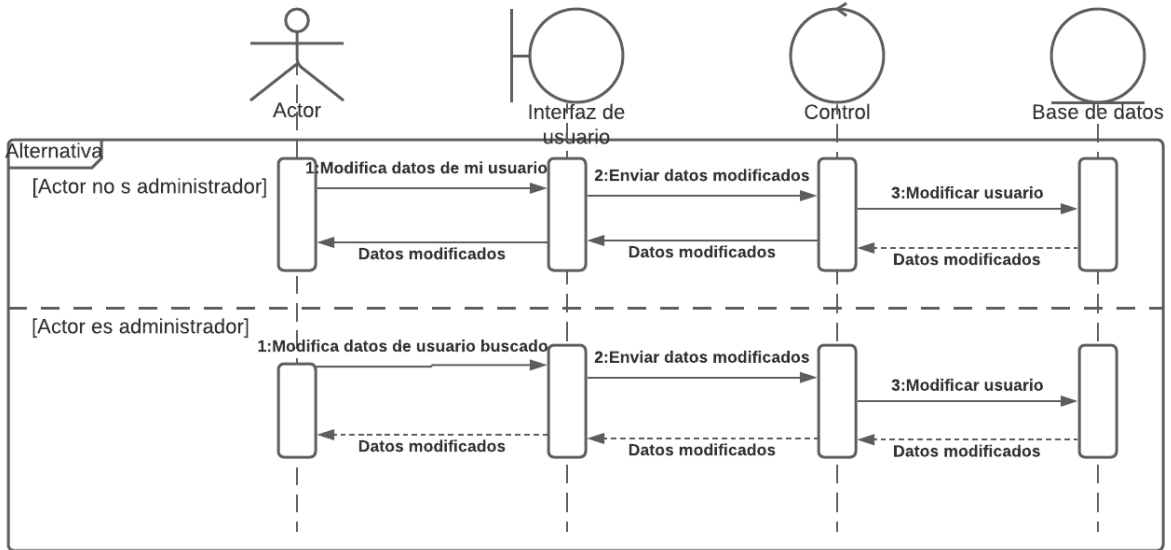


Figura 76. Diagrama de secuencia, Modificar usuario.

Fuente: elaboración propia.

Agregar tipo de proceso

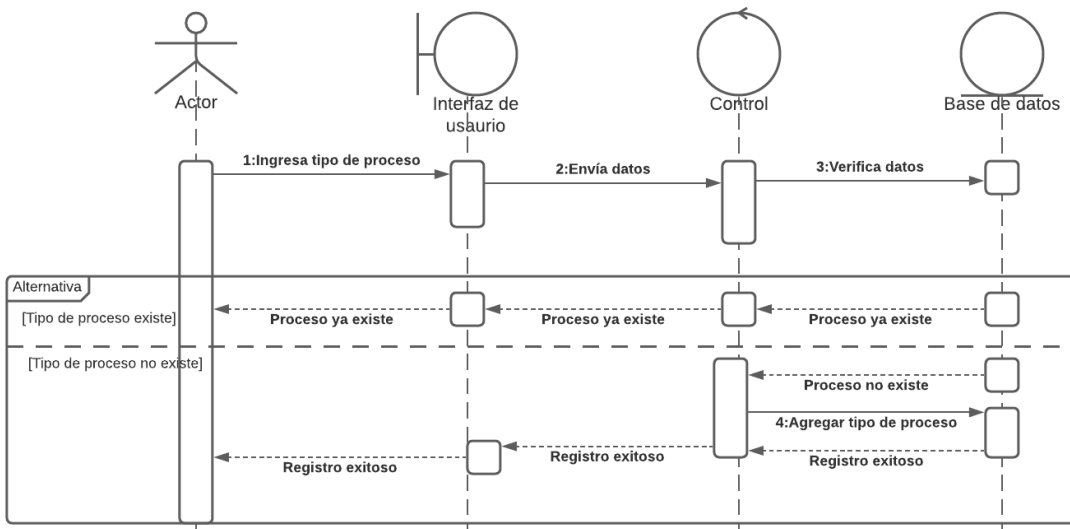


Figura 77. Diagrama de secuencia, Agregar tipo de proceso.

Fuente elaboración propia.

Eliminar tipo de proceso

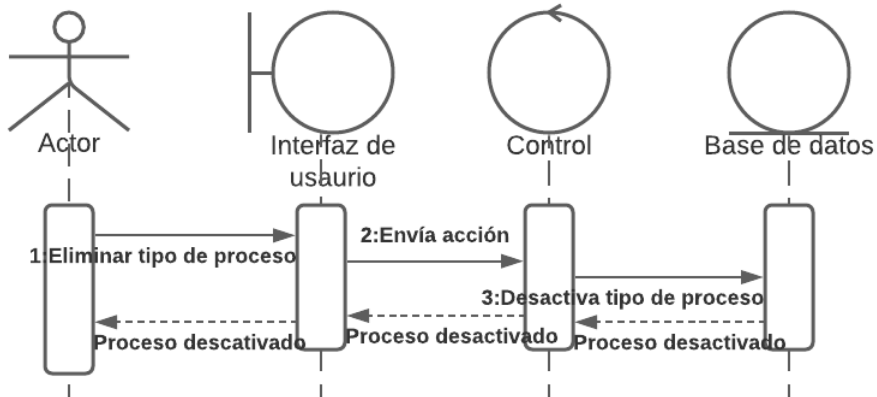


Figura 78. Diagrama de secuencia, Eliminar tipo de proceso.

Fuente: elaboración propia.

Cambiar contraseña

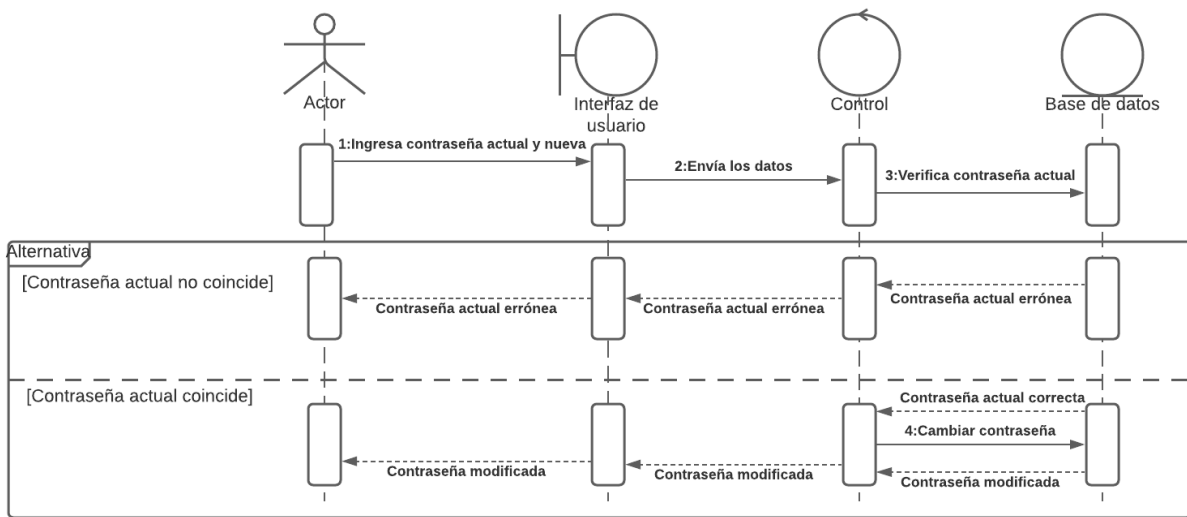


Figura 79. Diagrama de secuencia, Cambiar contraseña.

Fuente: elaboración propia.

Agregar honorario

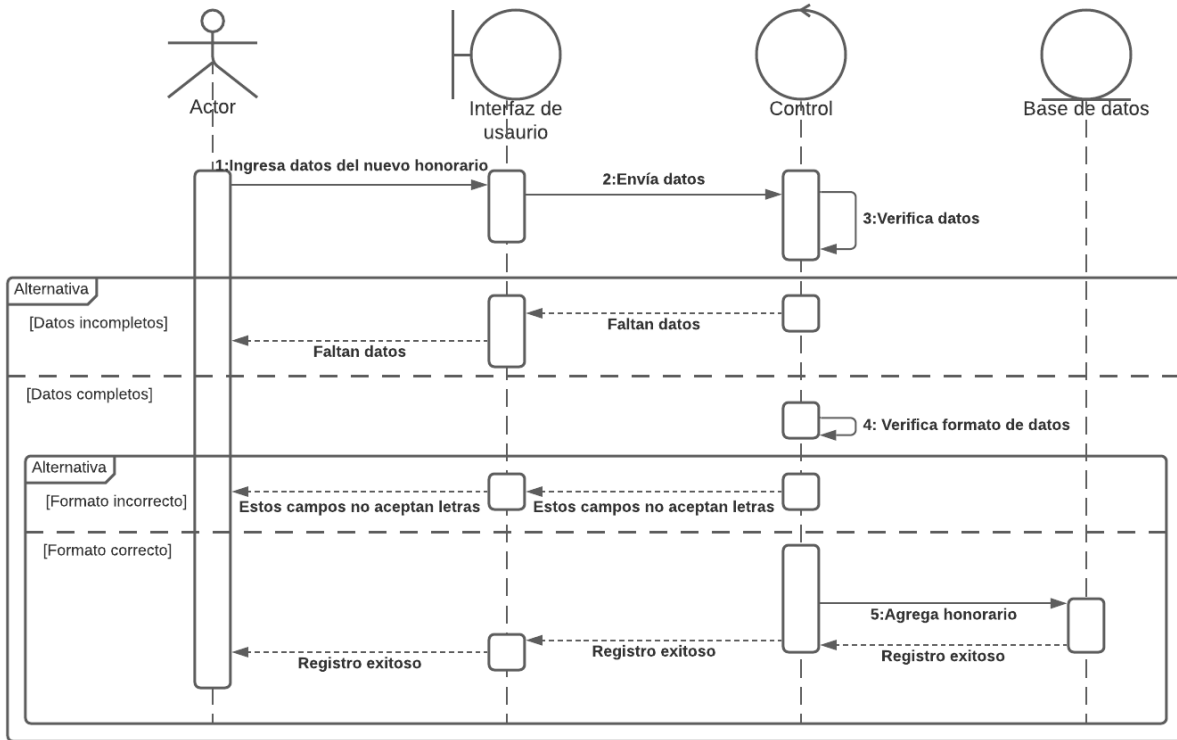


Figura 80. Diagrama de secuencia, Agregar honorario.

Fuente: elaboración propia

Agregar abonos

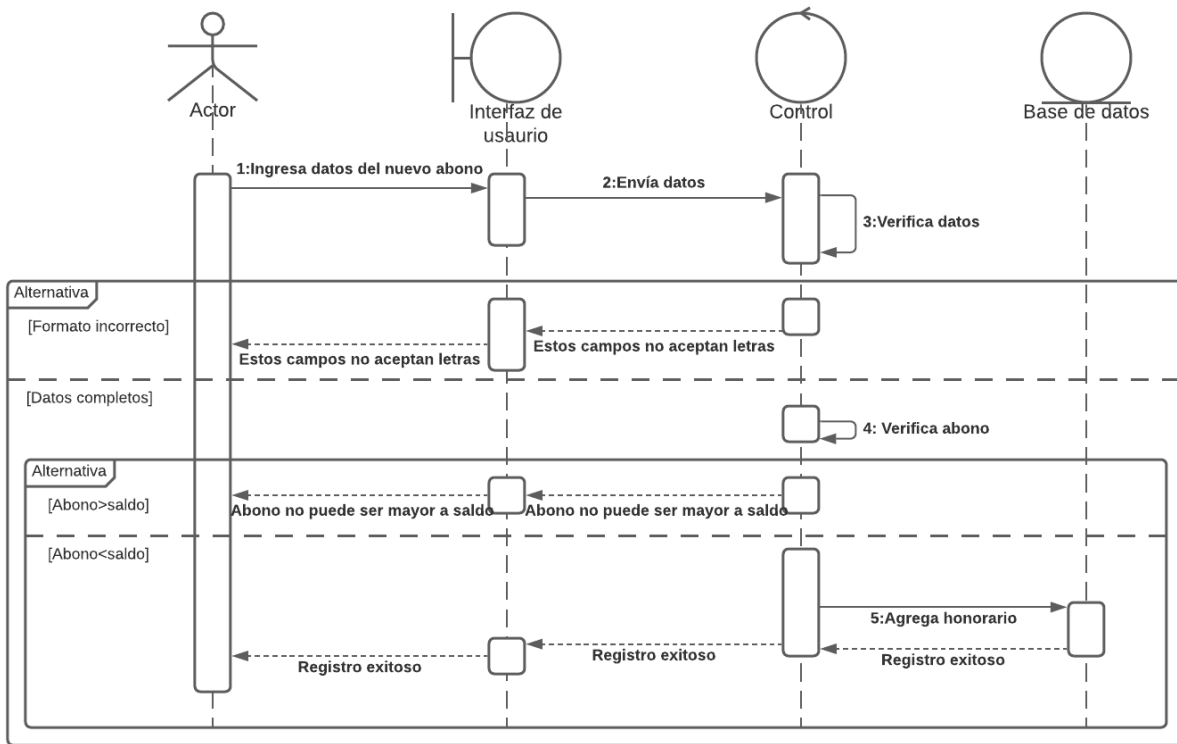


Figura 81. Diagrama de secuencia, Agregar abonos.

Fuente: elaboración propia.

Consultar honorario

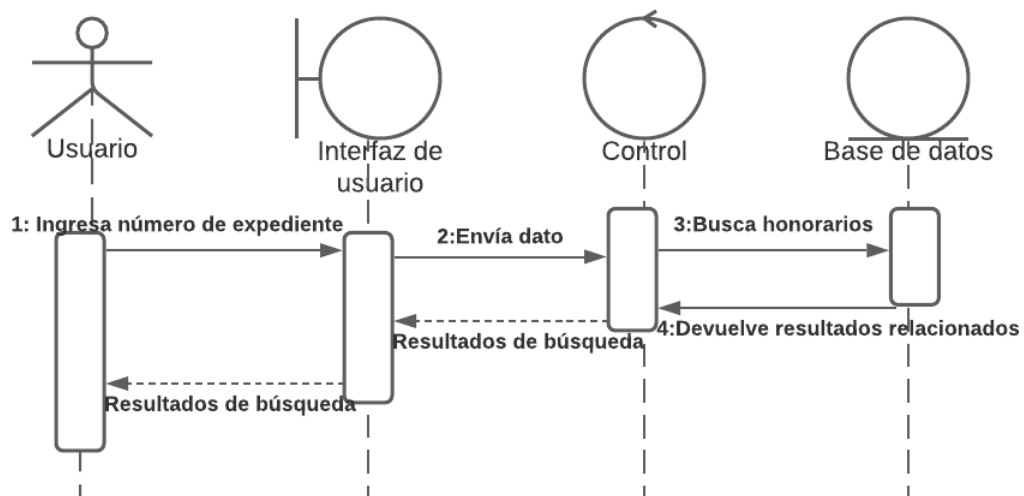


Figura 82. Diagrama de secuencia, Consultar honorario.

Fuente: elaboración propia.

Generar reportes

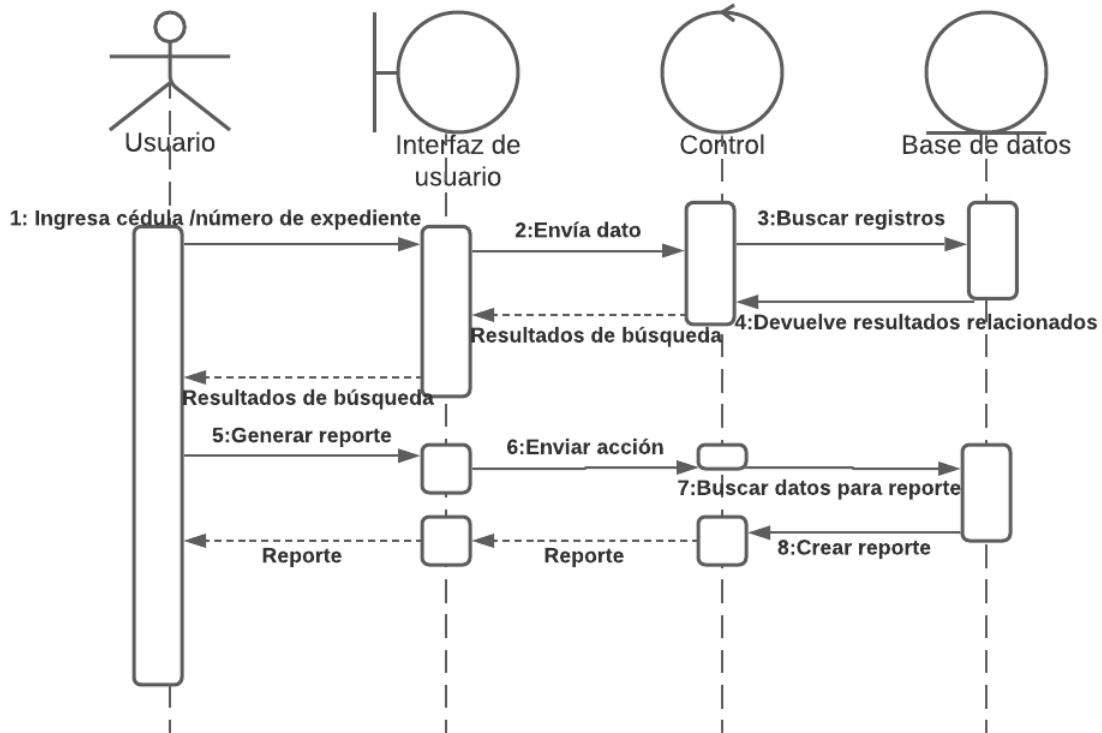


Figura 83. Diagrama de secuencia, Generar reporte.

Fuente: elaboración propia.

Agregar expediente

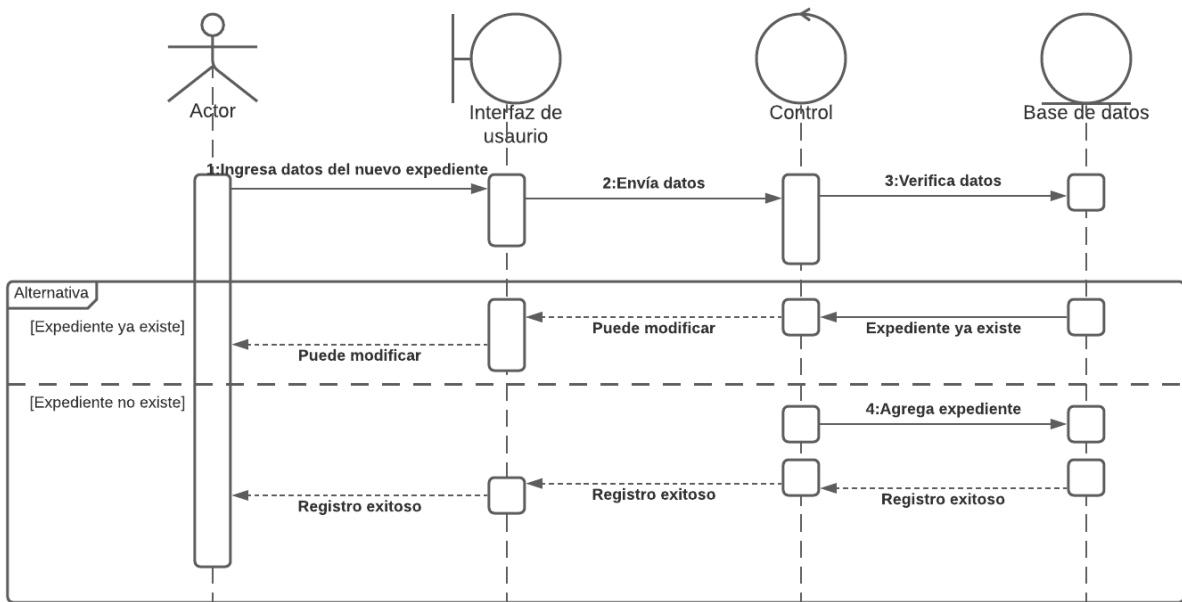


Figura 84. Diagrama de secuencia, Agregar expediente.

Fuente: elaboración propia.

Modificar expediente

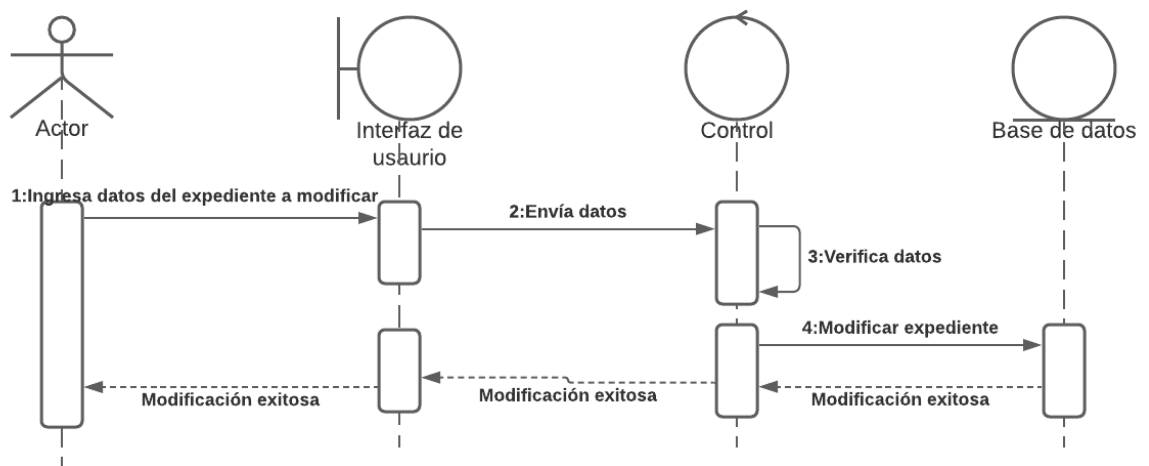


Figura 85. Diagrama de secuencia, Modificar expediente.

Fuente: elaboración propia.

Buscar expediente

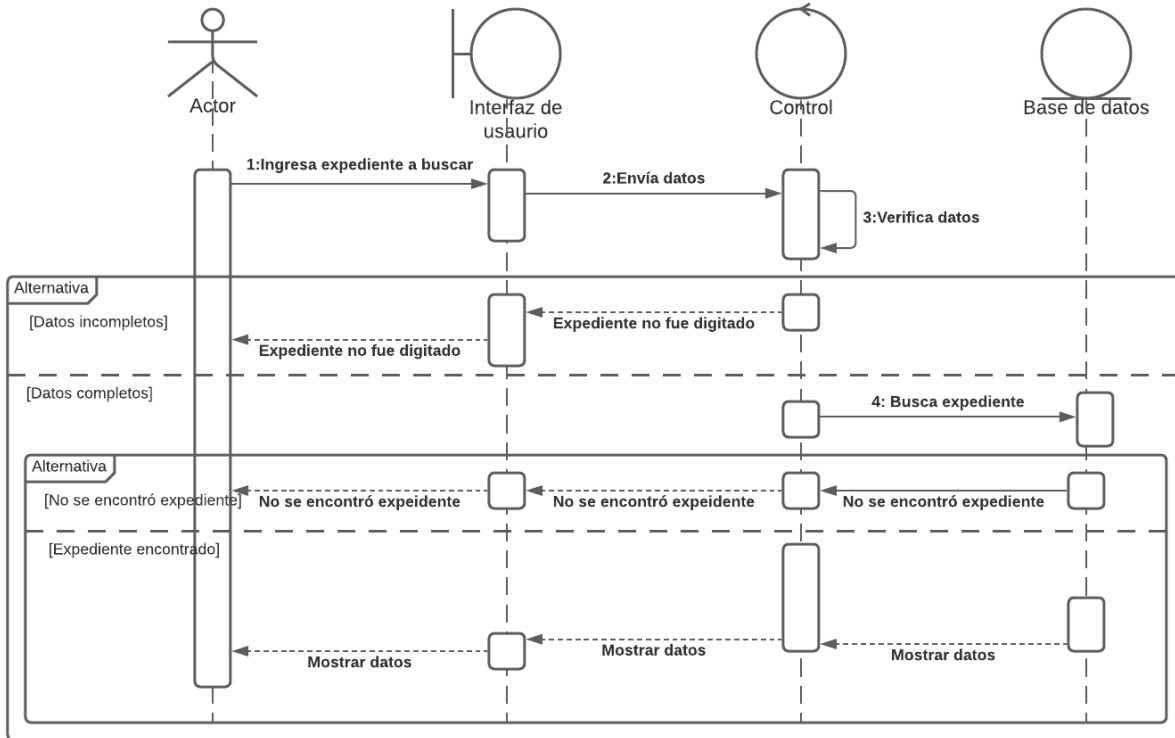


Figura 86. Diagrama de secuencia, Buscar expediente.

Fuente: elaboración propia.

Agregar caso

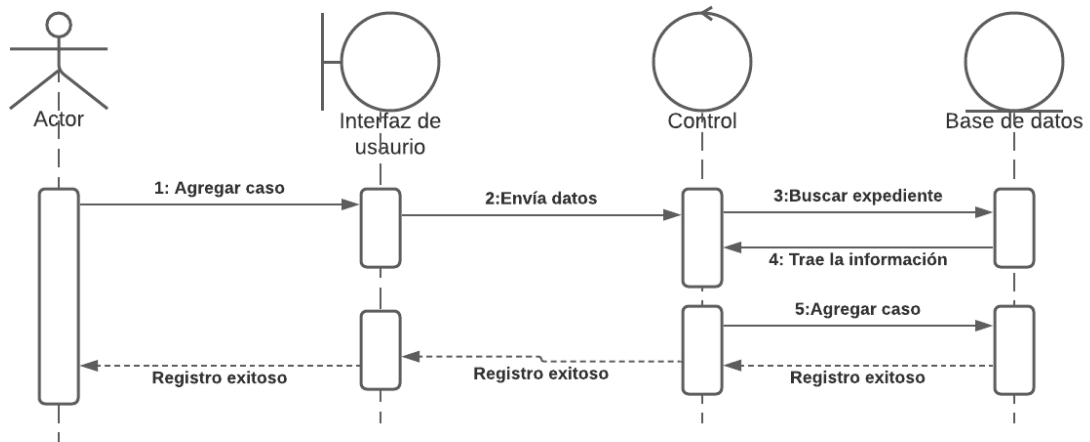


Figura 87. Diagrama de secuencia, Agregar caso.

Fuente: elaboración propia.

Agregar cita

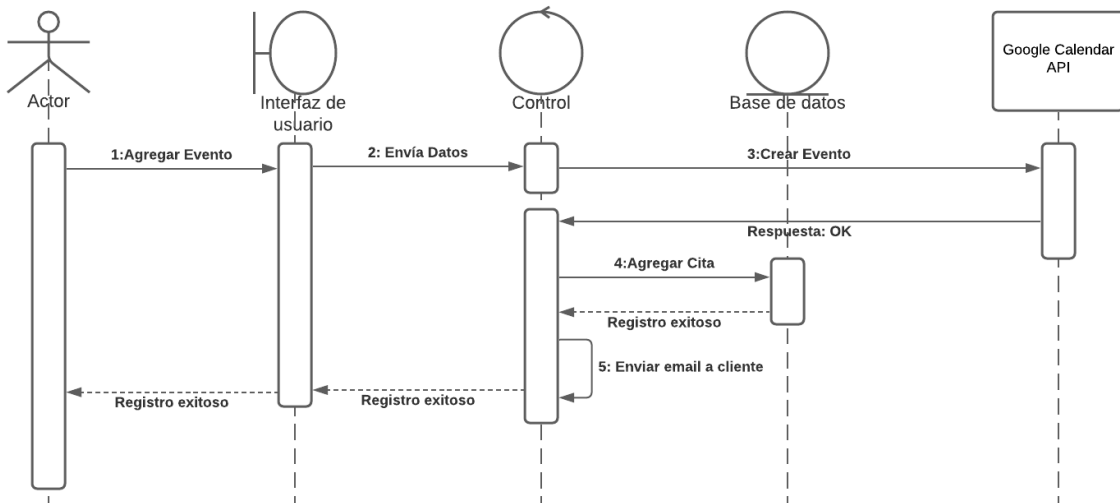


Figura 88. Diagrama de secuencia, agregar cita.

Fuente: elaboración propia.

Agregar recordatorio

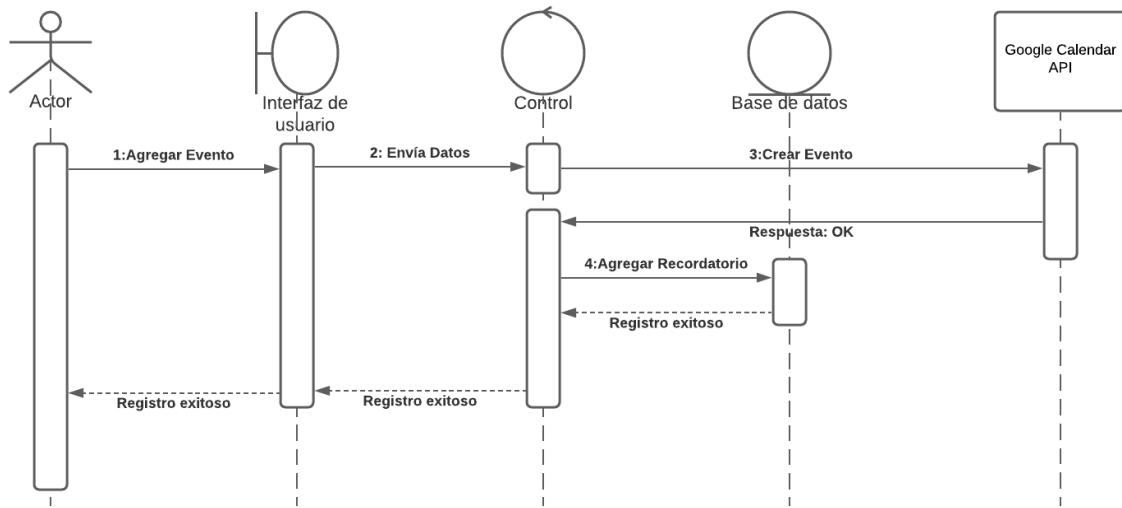


Figura 89. Diagrama de secuencia, agregar recordatorio.

Fuente: elaboración propia.

Consultar cita

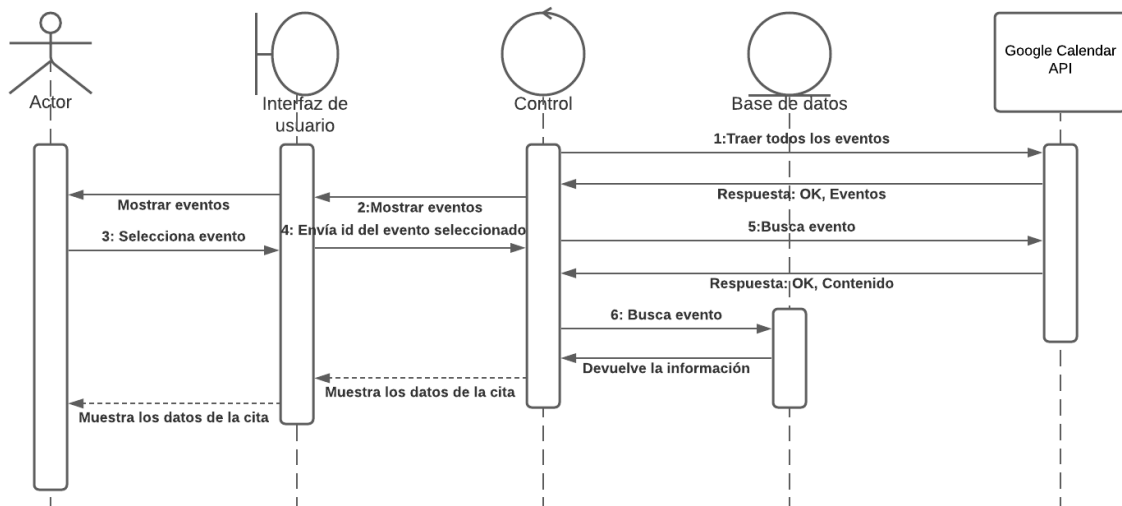


Figura 90. Diagrama de secuencia, consultar cita.

Fuente: elaboración propia.

Consultar recordatorio

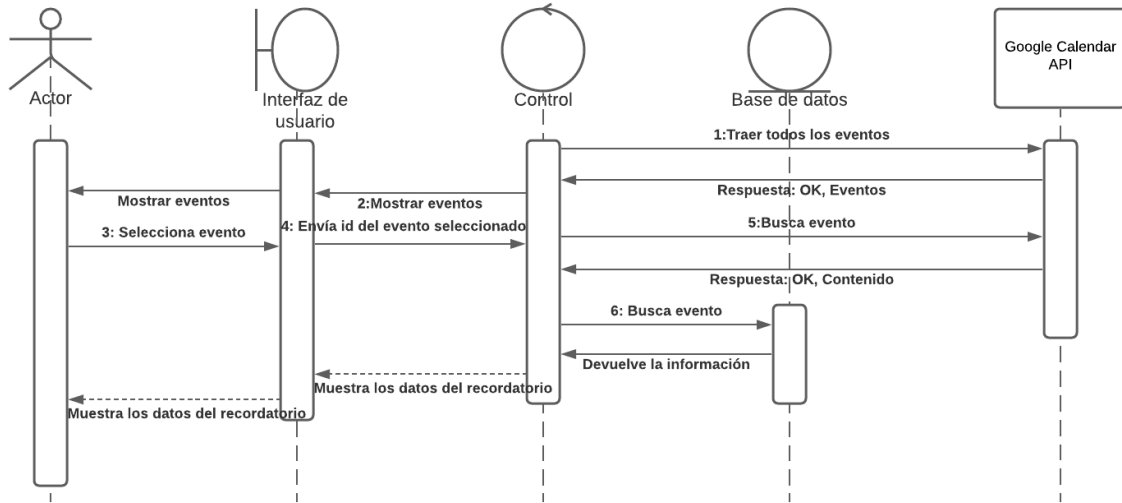


Figura 91. Diagrama de secuencia, consultar recordatorio.

Fuente: elaboración propia.

Modificar cita

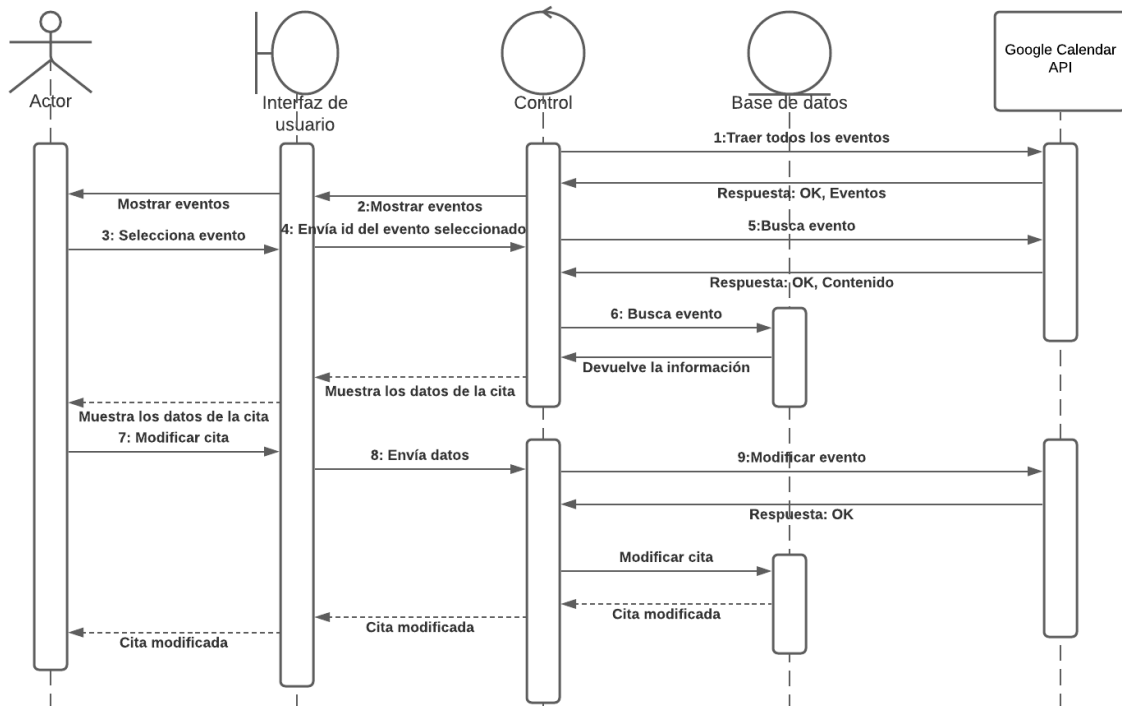


Figura 92. Diagrama de secuencia, modificar cita.

Fuente: elaboración propia.

Modificar recordatorio

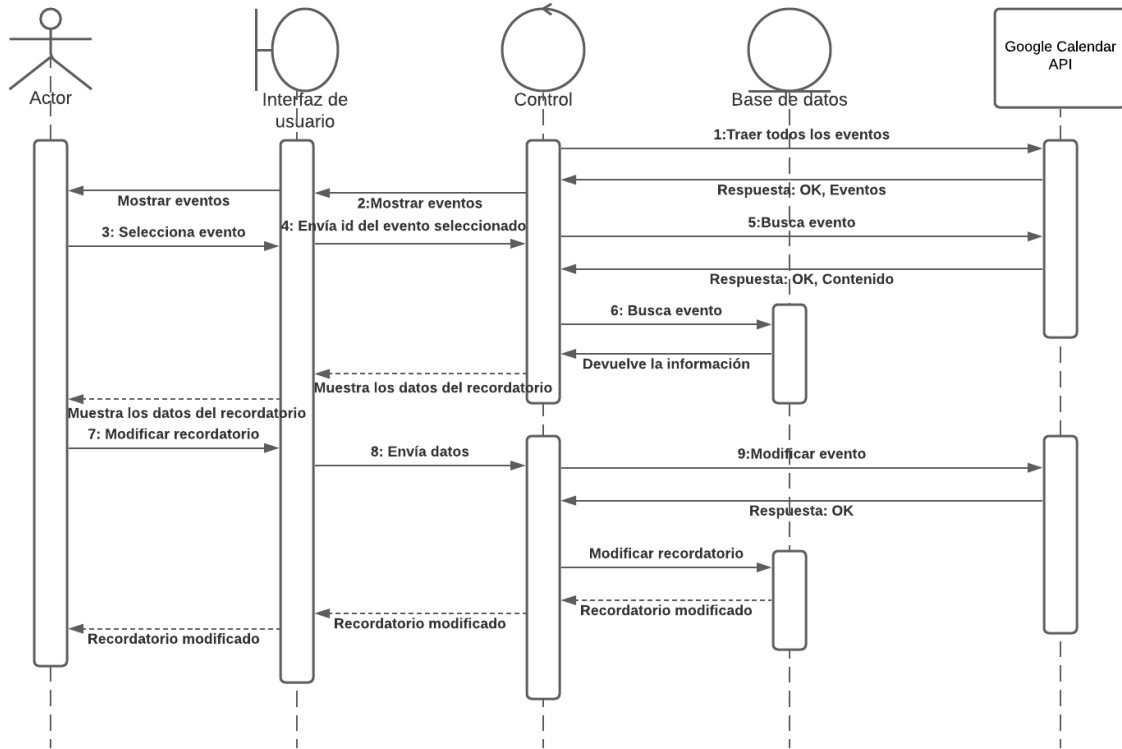


Figura 93. Diagrama de secuencia, modificar recordatorio.

Fuente: elaboración propia.

Eliminar cita

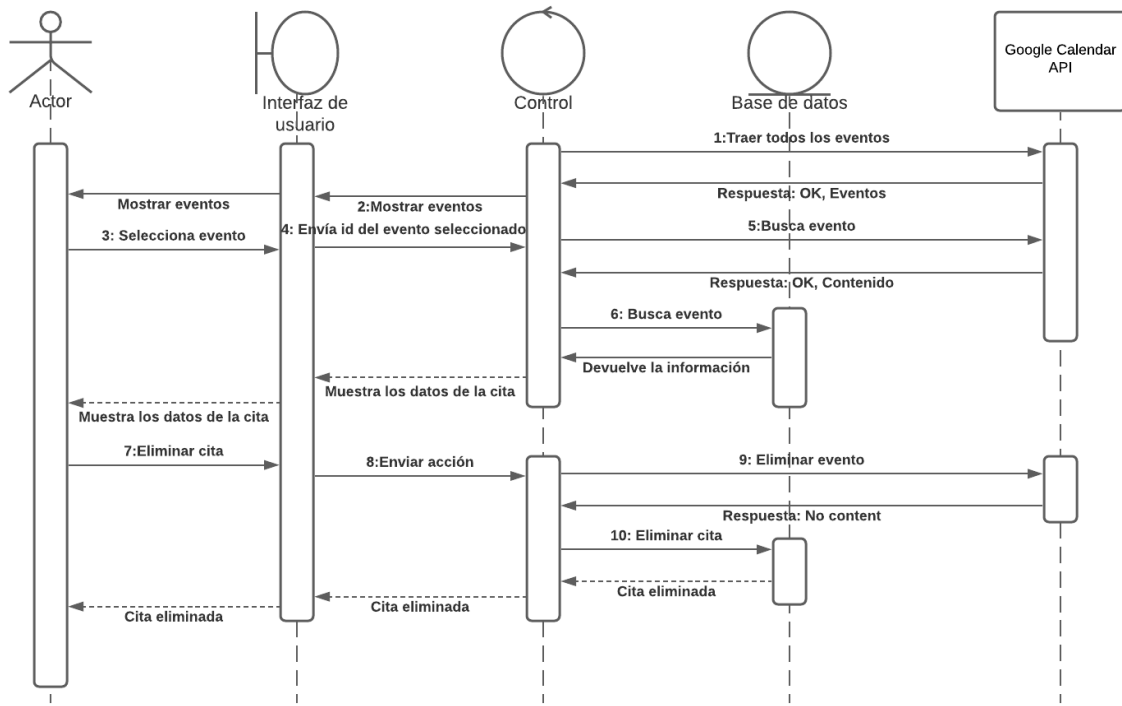


Figura 94. Diagrama de secuencia, eliminar cita.

Fuente: elaboración propia.

Eliminar recordatorio

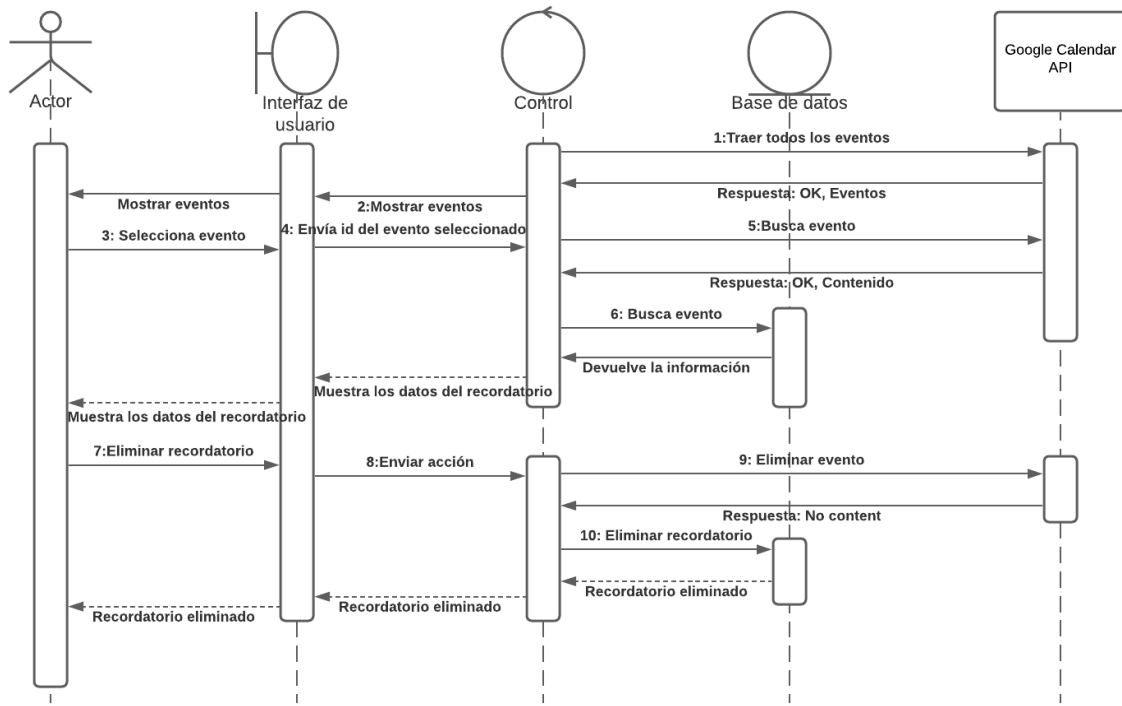


Figura 95. Diagrama de secuencia, eliminar recordatorio.

Fuente: elaboración propia.

5.2.5 Diagrama de entidad-relación

Un diagrama entidad relación es una representación gráfica que explica de forma sencilla las entidades del proyecto y cómo se relacionan entre sí.

La siguiente figura muestra el diagrama de entidad- relación del proyecto del sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados.

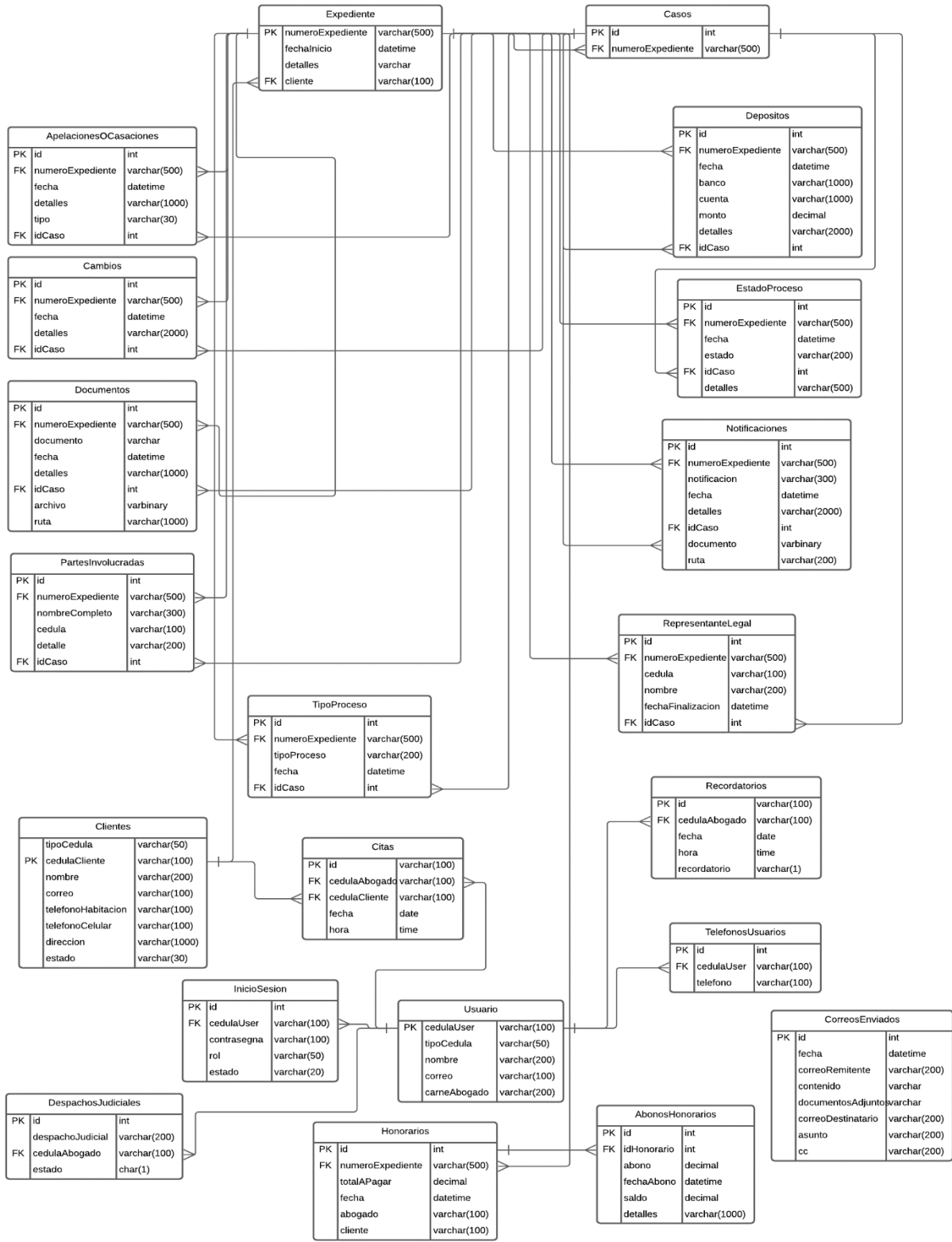


Figura 96. Diagrama entidad-relación.

Fuente: elaboración propia.

5.3 Desarrollo del proyecto

En esta sección del proyecto se describen las áreas del desarrollo. El diccionario de datos de la base de datos, y las pantallas del proyecto desarrollado.

5.3.1 Base de datos

La siguiente figura muestra el diagrama de a base de datos.

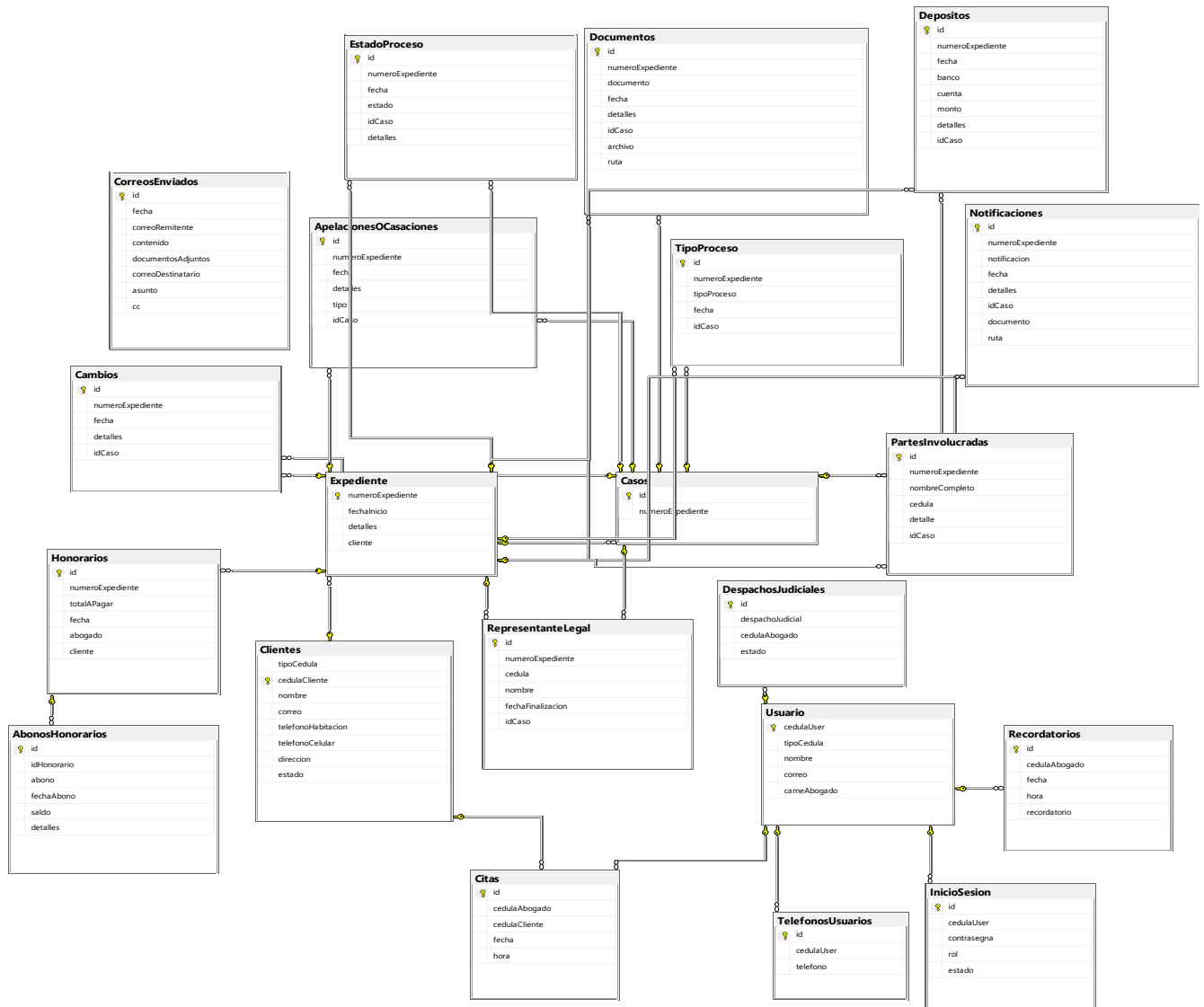


Figura 97. Diagrama de base de datos.

Fuente: elaboración propia.

Las siguientes tablas son el diccionario de datos, el cual presenta la descripción de cada campo en cada tabla de la base de datos.

Tabla de clientes.

Tabla: Clientes			
Descripción: Almacena la información de los clientes.			
Llave	Variable	Tipo	Descripción
PK	cedulaCliente	Varchar	Almacena el número de identificación del cliente.
	tipoCedula	Varchar	Almacena el tipo de identificación del cliente.
	nombre	Varchar	Almacena el nombre completo del cliente.
	correo	Varchar	Almacena el correo electrónico del cliente.
	telefonoHabitacion	Varchar	Almacena el número de teléfono del cliente.
	telefonoCelular	Varchar	Almacena el teléfono celular del cliente.
	direccion	varchar	Almacena la dirección exacta del cliente.

	estado	varchar	Almacena si el cliente es un cliente activo o inactivo.
--	--------	---------	---

Tabla 97. Tabla de clientes.

Fuente: elaboración propia.

Tabla de apelaciones o casaciones

Tabla: ApelacionesOCasaciones			
Descripción: Almacena alguna apelación o casación hecha, relacionada al expediente.			
Llave	Variable	Tipo	Descripción
PK	id	int	Identificador del registro.
FK	numeroExpediente	varchar	Almacena el número de expediente al que está relacionado.
	fecha	datetime	Almacena la fecha en la que se realiza la apelación.
	detalles	varchar	Almacena información acerca de la apelación.
	tipo	varchar	Almacena si es una apelación o casación.
FK	idCaso	int	Almacena el caso al que se relaciona la apelación o casación.

Tabla 98. Tabla de apelaciones.

Fuente: elaboración propia.

Tabla de citas

Tabla: Citas			
Descripción: Almacena las citas que se agendan.			
Llave	Variable	Tipo	Descripción
PK	id	varchar	Identificador del registro.
FK	cedulaAbogado	varchar	Almacena la cédula del abogado al que se le agenda la cita.
FK	cedulaCliente	varchar	Almacena la cédula del cliente que tiene la cita.
	fecha	Date	Almacena la fecha de la cita.
	hora	Time	Almacena la hora de la cita.

Tabla 99. Tabla de citas.

Fuente: elaboración propia.

Tabla de Cambios

Tabla: Cambios			
Descripción: Almacena detalles acerca de cambios o actualizaciones del expediente.			
Llave	Variable	Tipo	Descripción
PK	id	Int	Identificador del registro.
FK	numeroExpediente	Varchar	Almacena el número del expediente al que se le realizan los cambios.

	fecha	datetime	Almacena la fecha que se agregan los cambios.
	detalles	varchar	Almacena la descripción de los cambios realizados.
FK	idCaso	int	Almacena el id del caso al que se relaciona el cambio.

Tabla 100. Tabla de cambios.

Fuente: elaboración propia.

Tabla de tipo de proceso

Tabla: TipoProceso			
Descripción: Almacena el tipo de proceso o los tipos de proceso (también llamado despacho judicial), al que pertenece o perteneció el expediente.			
Llave	Variable	Tipo	Descripción
PK	id	Int	Identificador del registro.
FK	numeroExpediente	Varchar	Almacena el número de expediente al que se relaciona el tipo de proceso o despacho judicial.
	tipoProceso	varchar	Almacena el tipo de proceso al que pertenece el expediente.

	fecha	Datetime	Almacena la fecha en que se asigna el tipo de proceso.
FK	idCaso	int	Almacena el id del caso al que se relaciona el tipo de proceso.

Tabla 101. Tabla de tipo de proceso.

Fuente: elaboración propia.

Tabla de expedientes

Tabla: Expediente			
Descripción: Almacena los expedientes de los clientes.			
Llave	Variable	Tipo	Descripción
PK	numeroExpediente	varchar	Almacena el número de identificación del expediente.
FK	cliente	Varchar	Almacena la cédula del cliente relacionado con el expediente.
	fechaInicio	Datetime	Almacena la fecha que se crea el expediente.
	detalles	varchar	Almacena detalles adicionales sobre el expediente.

Tabla 102. Tabla de expedientes.

Fuente: elaboración propia.

Tabla de representante legal

Tabla: RepresentanteLegal			
Descripción: Almacena registro de antiguos de representantes legales que llevaron el caso antes del abogado actual.			
Llave	Variable	Tipo	Descripción
PK	id	Int	Identificador del registro.
FK	numeroExpediente	Varchar	Almacena el número del expediente al que se le asigna el representante legal.
	cedula	Varchar	Almacena la cédula del antiguo representante legal.
	nombre	Varchar	Almacena el nombre del antiguo representante legal.
	fechaFinalizacion	datetime	Almacena la fecha en la que el abogado dejó de representar el caso.
FK	idCaso	int	Almacena el id del caso al que se relaciona el representante legal.

Tabla 103. Tabla de representante legal.

Fuente: elaboración propia.

Tabla de depósitos

Tabla: Depositos			
Descripción: Almacena depósitos que se hayan realizado, relacionados al caso.			
Llave	Variable	Tipo	Descripción
PK	id	Int	Identificador de registro.
FK	numeroExpediente	Varchar	Almacena el número de expediente al que se le relacionan los depósitos.
	fecha	Datetime	Almacena la fecha que se realizó el depósito.
	banco	Varchar	Almacena el nombre del banco en el que está la cuenta que recibió el depósito.
	cuenta	Varchar	Almacena el número de cuenta al que se hizo el depósito.
	monto	Decimal	Almacena el monto por el cual se hizo el depósito.
	detalles	varchar	Almacena detalles extra acerca del depósito realizado.
FK	idCaso	int	Almacena el id del caso al que se relaciona el depósito.

Tabla 104. Tabla de depósitos.

Fuente: elaboración propia.

Tabla de documentos

Tabla: Documentos			
Descripción: Almacena los documentos subidos al sistema.			
Llave	Variable	Tipo	Descripción
PK	id	Int	Identificador de registro.
FK	numeroExpediente	Varchar	Almacena el número de identificación del expediente al que se le asignan los documentos.
	documento	Varchar	Almacena el nombre del documento que le asigna el abogado.
	fecha	Datetime	Almacena la fecha en que se sube el documento.
	detalles	varchar	Almacena detalles acerca del documento.
FK	idCaso	int	Almacena el id del caso al que se relaciona el documento.
	archivo	varbinary	Almacena el archivo en bytes.
	ruta	varchar	Almacena la ruta del archivo.

Tabla 105. Tabla de documentos.

Fuente: elaboración propia.

Tabla de partes involucradas

Tabla: PartesInvolucradas			
Descripción: Almacena las partes involucradas en un caso.			
Llave	Variable	Tipo	Descripción
PK	id	Int	Identificador del registro.
FK	numeroExpediente	Varchar	Almacena el número de expediente al que se le relacionan las personas involucradas.
	nombreCompleto	Varchar	Almacena el nombre de la parte involucrada en el caso.
	cedula	Varchar	Almacena la cedula de la persona involucrada en el caso.
	detalles	varchar	Almacena detalles adicionales sobre la persona involucrada al caso.
FK	idCaso	int	Almacena el id del caso al que se

			relacionan las partes involucradas.
--	--	--	-------------------------------------

Tabla 106. Tabla de partes involucradas.

Fuente: elaboración propia.

Tabla de despachos judiciales

Tabla: DespachosJudiciales			
Descripción: Almacena los tipos de proceso que el abogado va a manejar o de los que va a brindar servicios.			
Llave	Variable	Tipo	Descripción
PK	id	int	Identificador del registro.
FK	cedulaAbogado	Varchar	Almacena la cedula del abogado al que se le relacionará el tipo de proceso.
	despachoJudicial	Varchar	Almacena el tipo de proceso que se desea agregar.
	estado	Char	Almacena el estado en el que se encuentra ese tipo de proceso. Puede ser A, de Activo o I, de Inactivo.

Tabla 107. Tabla de despachos judiciales.

Fuente: elaboración propia.

Tabla de estado del proceso

Tabla: EstadoProceso			
Descripción: Almacena el estado en el que se encuentra el expediente.			
Llave	Variable	Tipo	Descripción
PK	id	int	Identificador del registro.
FK	numeroExpediente	varchar	Almacena el número del expediente al que se le asigna un estado.
	fecha	datetime	Almacena la fecha a la que se le asigna el estado.
	estado	varchar	Almacena el estado que se le asigna el estado.
	detalles	varchar	Almacena detalles del estado del proceso.
FK	idCaso	int	Almacena el id del caso al que se relaciona el estado del proceso.

Tabla 108. Tabla de estado del proceso.

Fuente: elaboración propia.

Tabla de notificaciones

Tabla: Notificaciones			
Descripción: Almacena las notificaciones que envía la corte relacionado con el expediente.			
Llave	Variable	Tipo	Descripción

PK	id	int	Identificador del registro.
FK	numeroExpediente	varchar	Almacena el número de expediente al que se le asigna la notificación nueva.
	notificacion	Varchar	Almacena el título de la notificación.
	fecha	Datetime	Almacena la fecha de la notificación.
	detalles	Varchar	Almacena detalles adicionales acerca de la notificación agregada.
	documento	varbinary	Almacena el documento que contiene la notificación, en bytes.
FK	idCaso	int	Almacena el id del caso al que se relaciona la notificación.
	ruta	varchar	Almacena la ruta del documento subido.

Tabla 109. Tabla de notificaciones.

Fuente: elaboración propia.

Tabla de recordatorios

Tabla: Recordatorios

Descripción: Almacena los recordatorios creados por el abogado.			
Llave	Variable	Tipo	Descripción
PK	id	varchar	Identificador del registro.
FK	cedulaAbogado	varchar	Almacena la cédula del abogado que registra el recordatorio.
	fecha	date	Almacena la fecha en la que se desea ser recordado.
	hora	time	Almacena la hora del recordatorio.
	recordatorio	varchar	Almacena el contenido del recordatorio.

Tabla 110. Tabla de recordatorios.

Fuente: elaboración propia.

Tabla de usuarios

Tabla: Usuario			
Descripción: Almacena los usuarios del sistema.			
Llave	Variable	Tipo	Descripción
PK	cedulaUser	varchar	Almacena la cédula del usuario.
	tipoCedula	varchar	Almacena el tipo de cédula del usuario.
	nombre	varchar	Almacena el nombre del usuario.

	correo	Varchar	Almacena el correo electrónico del usuario.
	carneAbogado	varchar	Almacena el número del carné de abogado, en caso de que el usuario tenga rol de abogado.

Tabla 111. Tabla de usuarios.

Fuente: elaboración propia.

Tabla de teléfonos de los usuarios

Tabla: TelefonosUsuarios			
Descripción: Almacena los números de teléfonos de los usuarios del sistema.			
Llave	Variable	Tipo	Descripción
PK	id	Int	Identificador del registro.
FK	cedulaUser	Varchar	Almacena la cédula del usuario al que se le asigna el número de teléfono.
	telefono	varchar	Almacena el número de teléfono.

Tabla 112. Tabla de teléfonos de los usuarios.

Fuente: elaboración propia.

Tabla de inicio de sesión

Tabla: InicioSesion

Descripción: Almacena los inicios de sesión de los usuarios del sistema.			
Llave	Variable	Tipo	Descripción
PK	id	int	Identificador del registro.
FK	cedulaUser	varchar	Almacena el número de cédula al que se le asigna las credenciales de inicio de sesión.
	contrasegna	varchar	Almacena la contraseña para iniciar sesión del usuario.
	estado	Varchar	Almacena el estado del usuario, esto determina si tiene permiso para utilizar el sistema. Puede ser: A, de Autorizado, o D, de Denegado
	rol	Varchar	Almacena el rol asignado al usuario dentro del sistema.

Tabla 113. Tabla de inicio de sesión.

Fuente: elaboración propia.

Tabla de correos enviados

Tabla: CorreosEnviados
Descripción: Almacena los correos enviados por el sistema.

Llave	Variable	Tipo	Descripción
PK	id	int	Identificador del registro.
	fecha	datetime	Almacena la fecha en que se envió el correo.
	correoRemitente	varchar	Almacena el correo del remitente, es decir, de quien envió el correo.
	correoDestinatario	varchar	Almacena el correo del destinatario, es decir el correo de a quien se le envió.
	contenido	varchar	Almacena el contenido que tenía el correo.
	documentosAdjuntos	varchar	Almacena los documentos adjuntos en el correo enviado.
	asunto	varchar	Almacena el asunto que se le asignó al correo.
	cc	varchar	Almacena el cc que se le asigna al correo enviado.

Tabla 114. Tabla de correos enviados.

Fuente: elaboración propia.

Tabla de casos

Tabla: Casos			
Descripción: Almacena los casos relacionados a un número de expediente.			
Llave	Variable	Tipo	Descripción
PK	id	Int	Identificador del registro.
FK	numeroExpediente	Varchar	Almacena el número de expediente al que se le asigna el caso.

Tabla 115. Tabla de casos.

Fuente: elaboración propia.

Tabla de honorarios

Tabla: Honorarios			
Descripción: Almacena los honorarios que se cobrarán por los servicios brindados.			
Llave	Variable	Tipo	Descripción
PK	id	Int	Identificador del registro.
FK	numeroExpediente	Varchar	Almacena el número de expediente por el que se cobran los honorarios.
	totalAPagar	decimal	Almacena el monto a pagar.
	fecha	datetime	Almacena la fecha en que se hace el registro.
	abogado	varchar	Almacena el abogado al que se le asignan los honorarios.

	cliente	varchar	Almacena el cliente al que se le cobran los honorarios.
--	---------	---------	---

Tabla 116. Tabla de honorarios.

Fuente: elaboración propia.

Tabla de abonos a honorarios

Tabla: AbonosHonorarios			
Descripción: Almacena los abonos hacia un honorario.			
Llave	Variable	Tipo	Descripción
PK	id	Int	Identificador del registro.
FK	idHonorario	int	Almacena el id del honorario al que se le asigna.
	abono	decimal	Almacena el número de teléfono.
	fechaAbono	datetime	Almacena la fecha en que se hace el abono.
	saldo	decimal	Almacena el saldo del total a pagar.
	detalles	varchar	Almacena detalles acerca del abono.

Tabla 117. Tabla de abonos a honorarios.

Fuente: elaboración propia.

5.3.2 Pantallas de la aplicación web

A continuación se presentan las pantallas de los diferentes módulos del sistema web.

Inicio de sesión.

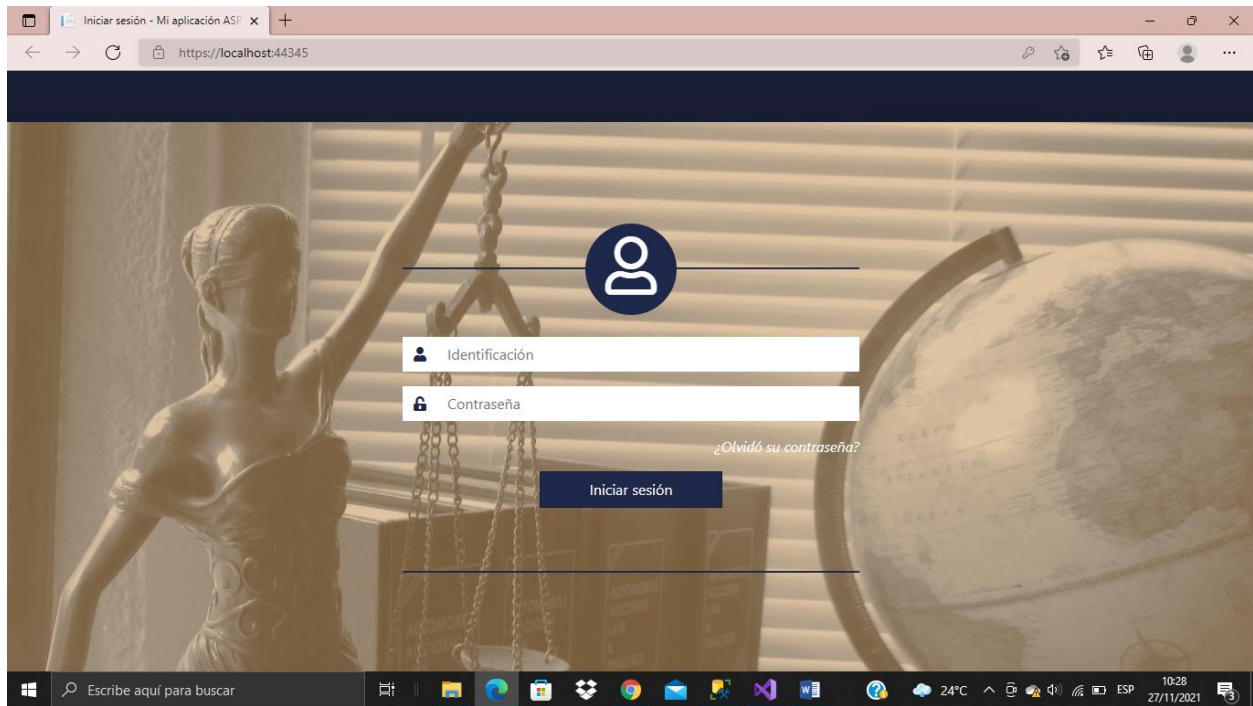


Figura 98. Pantalla de inicio de sesión.

Fuente: elaboración propia.

Recuperar contraseña

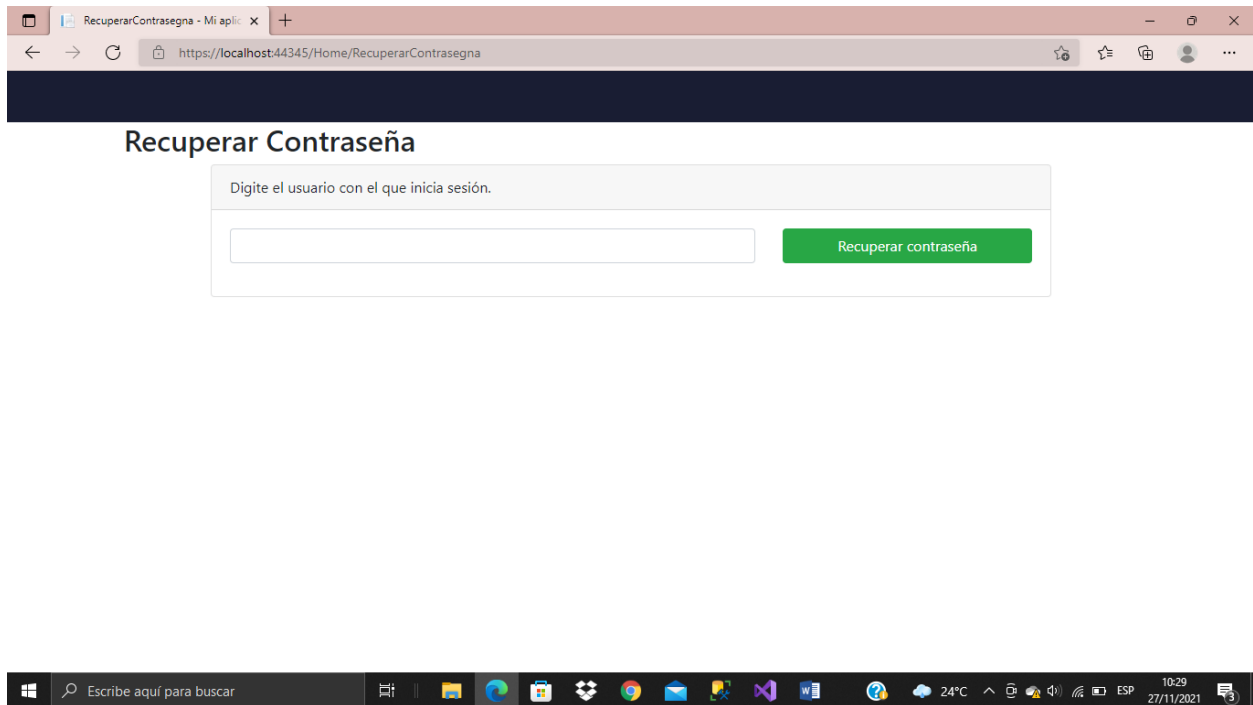


Figura 99. Pantalla de recuperación de contraseña.

Fuente: elaboración propia.

Pantalla principal y menú lateral para usuario abogado



Figura 100. Menú vertical de abogado.

Fuente: elaboración propia.

Pantalla principal y menú lateral para usuario administrador

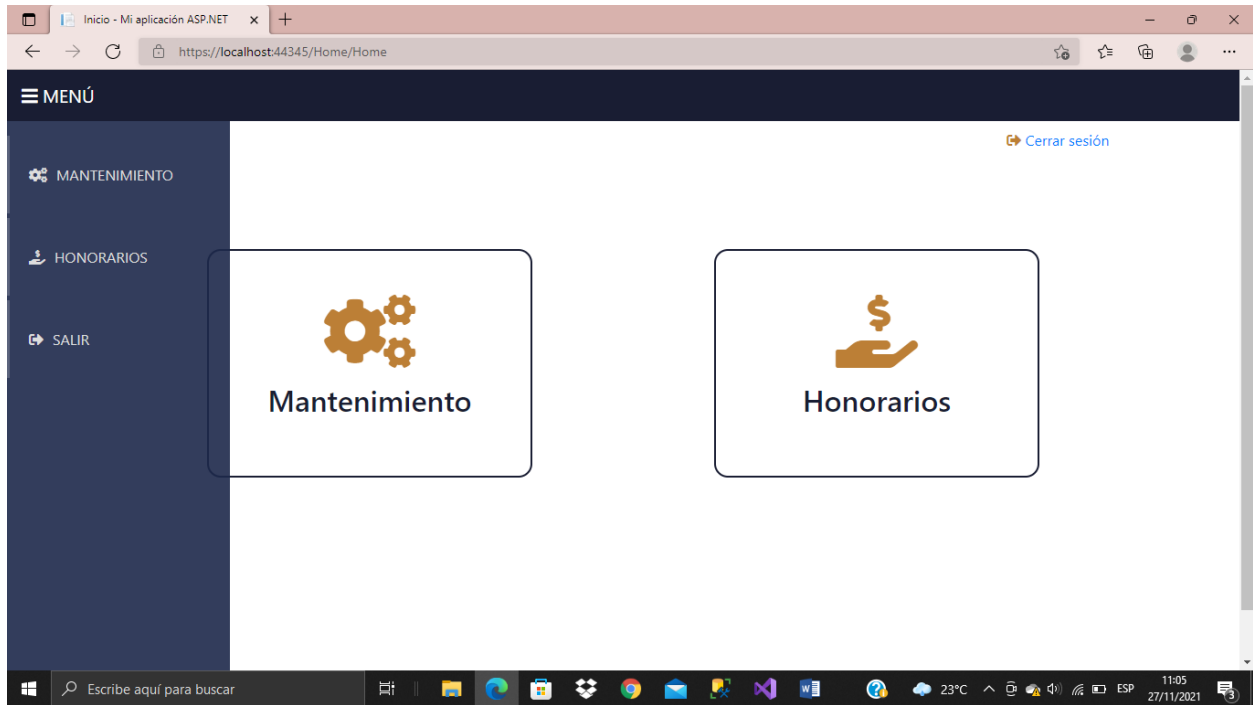


Figura 101. Pantalla principal de administrador.

Fuente: elaboración propia.

Módulo de reportes

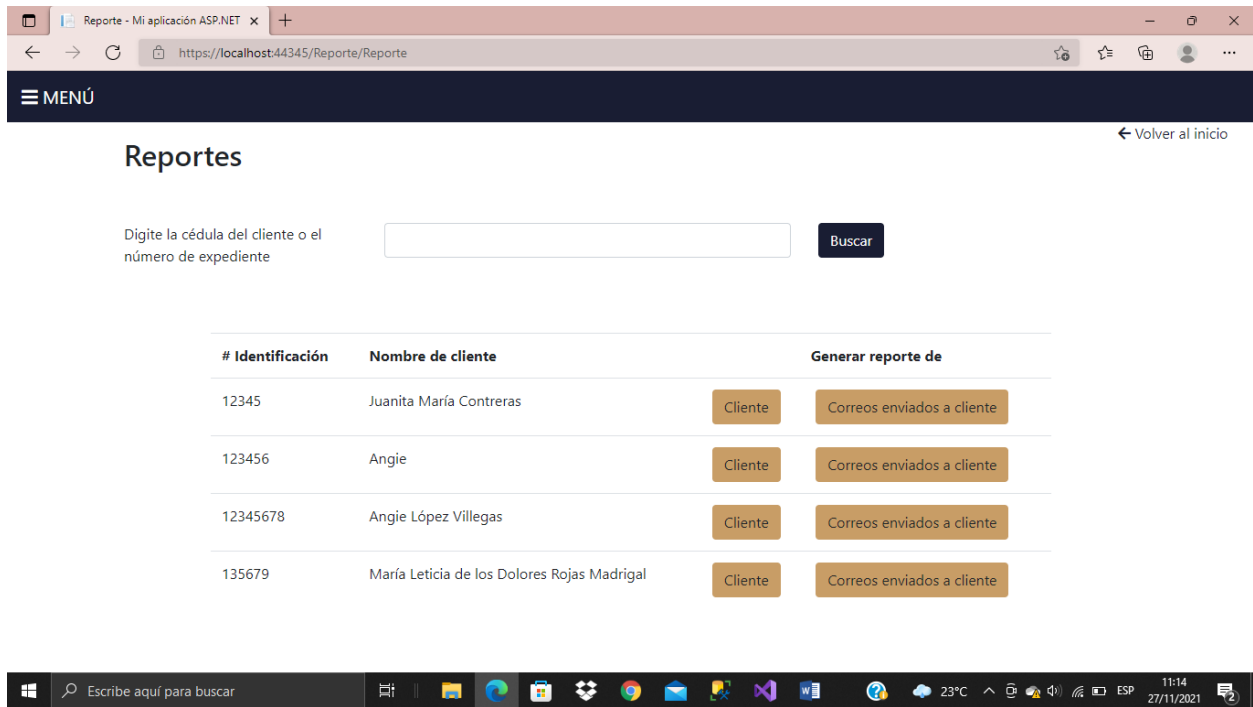


Figura 102. Pantalla de reportes.

Fuente: elaboración propia.

Módulo de Honorarios

Vista principal de honorarios

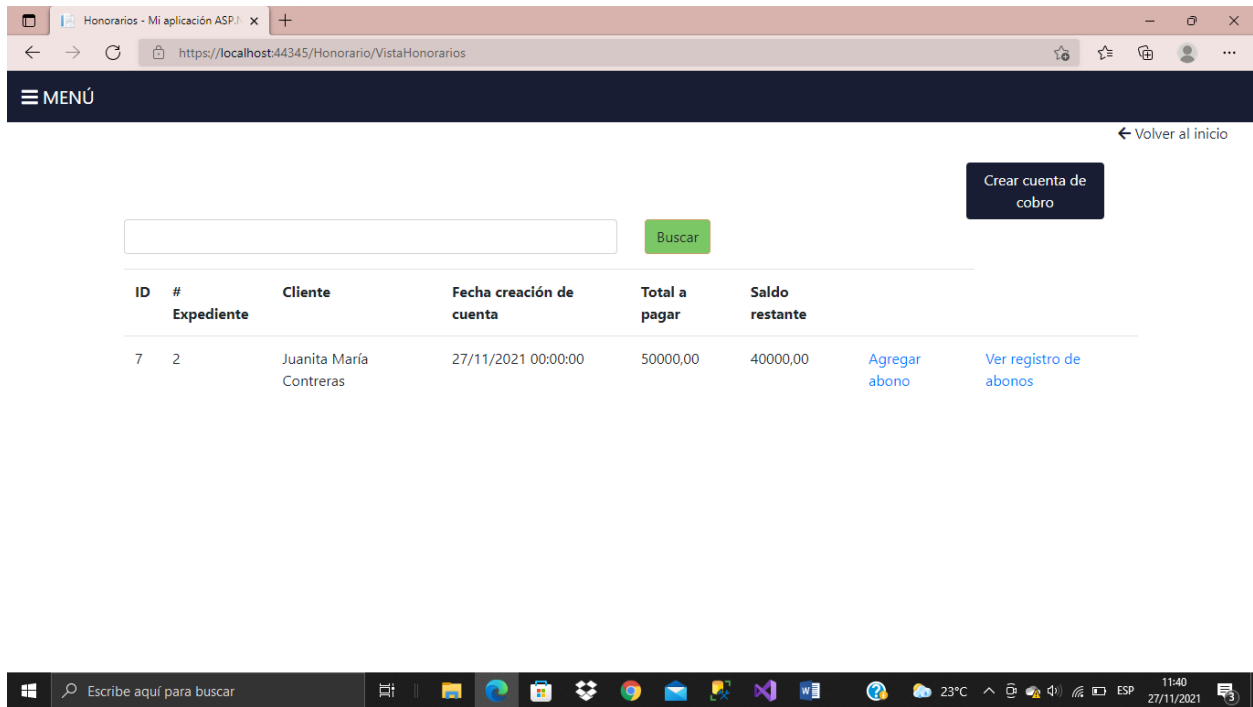


Figura 103. Vista principal Honorarios.

Fuente: elaboración propia.

Agregar honorario/ Crear cuenta de cobro para abogado

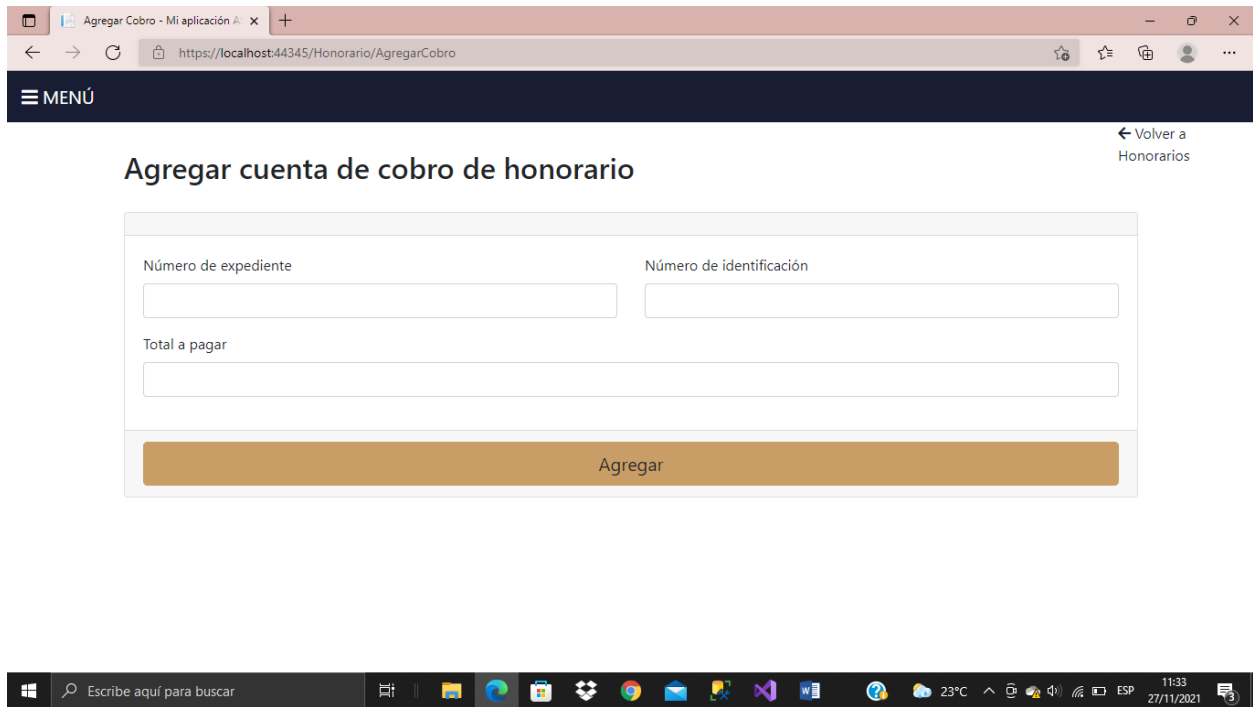


Figura 104. Pantalla agregar honorario para abogado.

Fuente: elaboración propia.

Agregar honorario/ Crear cuenta de cobro para administrador

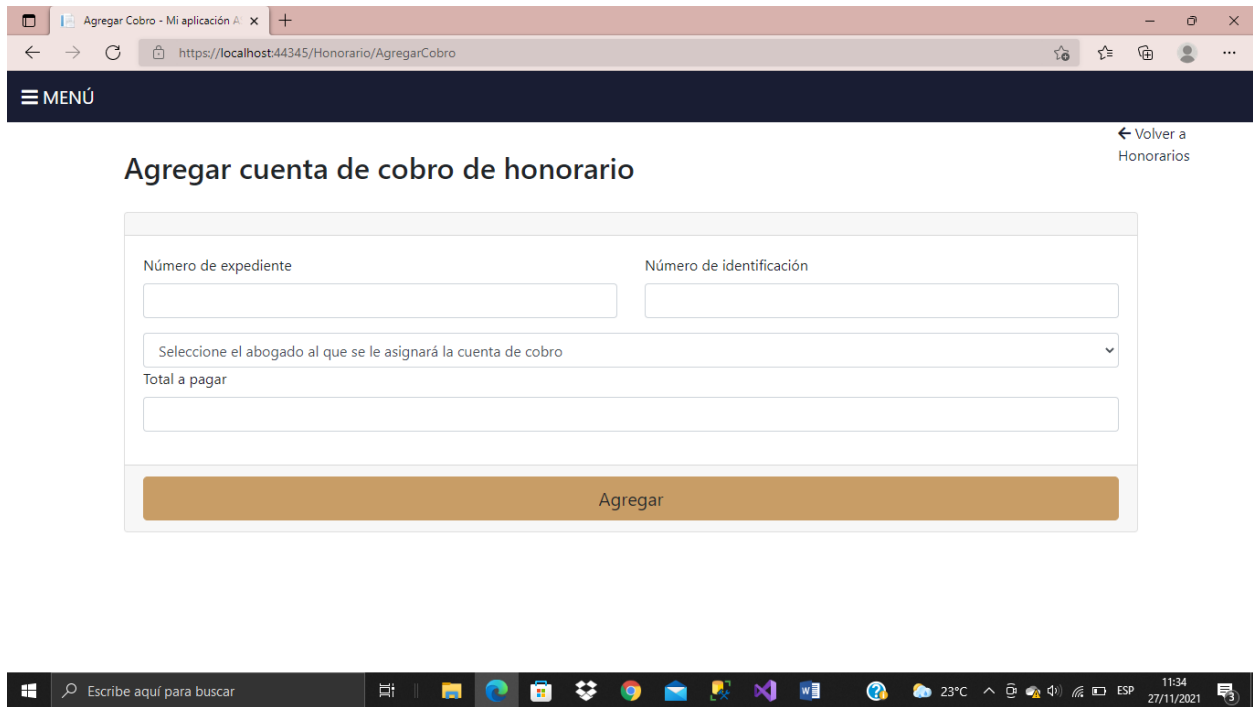


Figura 105. Agregar honorario para administrador.

Fuente: elaboración propia.

Agregar abono

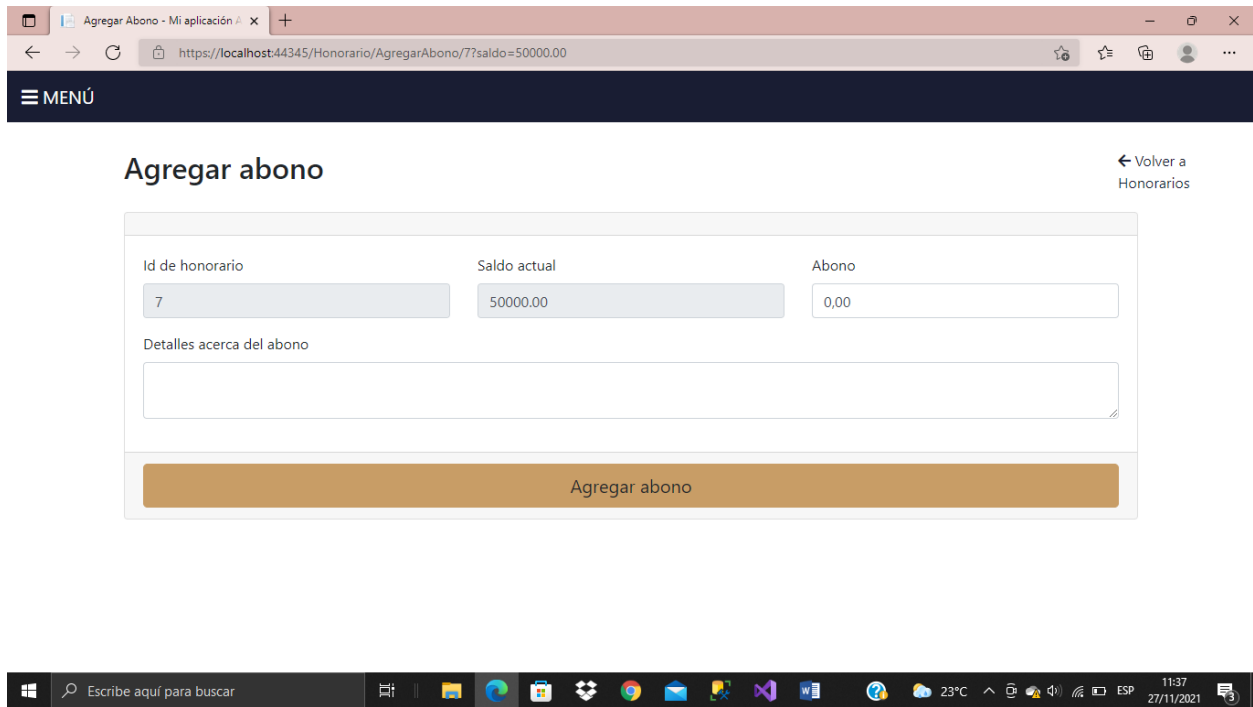


Figura 106. Agregar abono.

Fuente: elaboración propia.

Ver abonos.

Ver Abonos - Mi aplicación ASP | x +
 https://localhost:44345/Honorario/VerAbonos/7

MENÚ

Volver a Honorarios

Detalles de cuenta de cobro (Honorario)

ID de Honorario	# Expediente relacionado	Fecha de creación de cuenta de cobro	Abogado	Cliente	Total a pagar
7	2	27/11/2021 00:00:00	Angie López Villegas	Juanita María Contreras	50000,00

Detalles de abonos

ID	# Honorario	Fecha del abono	Abono	Saldo restante	Detalles sobre el abono
27	7	27/11/2021 00:00:00	10000,00	40000,00	abono de 10 000 colones

Escribe aquí para buscar

23°C 11:41 27/11/2021

Figura 107. Pantalla Ver abonos.

Fuente: elaboración propia.

Módulo de expediente

Expediente no registrado

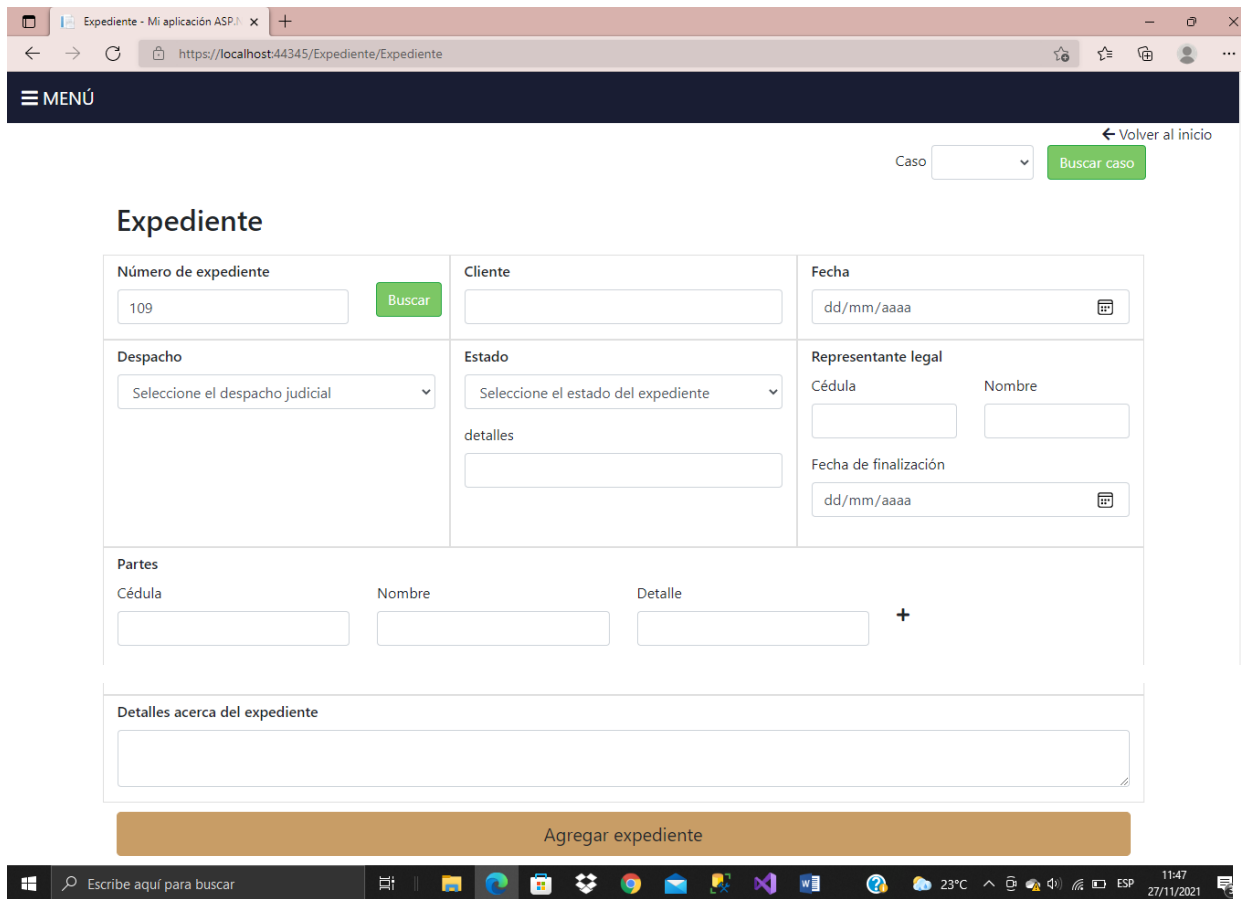


Figura 108. Pantalla expediente no registrado.

Fuente: elaboración propia.

Expediente registrado.

Caso

Expediente

Número de expediente <input type="text" value="2"/> <input type="button" value="Buscar"/>	Cliente <input type="text" value="12345"/>	Fecha <input type="text" value="03/02/2021"/>																											
Despacho <input type="text" value="Seleccione el despacho judicial"/>	Estado <input type="text" value="Seleccione el estado del expediente"/>	Representante legal Cédula <input type="text"/> Nombre <input type="text"/> Fecha de finalización <input type="text" value="01/01/0001"/>																											
Registros anteriores: <table border="1"> <tr> <th>Despacho</th> <th>Agregado</th> </tr> <tr> <td>familiar</td> <td>10/08/2021</td> </tr> </table>	Despacho	Agregado	familiar	10/08/2021	Registros anteriores: <table border="1"> <tr> <th>Estado</th> <th>Detalles</th> <th>Fecha</th> </tr> <tr> <td>Activo</td> <td></td> <td>25/11/2021</td> </tr> <tr> <td>activo</td> <td>esta en proceso bla bla bla</td> <td>07/07/2021</td> </tr> </table>	Estado	Detalles	Fecha	Activo		25/11/2021	activo	esta en proceso bla bla bla	07/07/2021	Registros anteriores: <table border="1"> <tr> <th>Cedula</th> <th>Nombre</th> <th>Fecha finalización</th> </tr> <tr> <td>23456</td> <td>maria jose gonzales</td> <td>10/10/2020</td> </tr> </table>	Cedula	Nombre	Fecha finalización	23456	maria jose gonzales	10/10/2020								
Despacho	Agregado																												
familiar	10/08/2021																												
Estado	Detalles	Fecha																											
Activo		25/11/2021																											
activo	esta en proceso bla bla bla	07/07/2021																											
Cedula	Nombre	Fecha finalización																											
23456	maria jose gonzales	10/10/2020																											
Partes <table border="1"> <tr> <th>Cédula</th> <th>Nombre</th> <th>Detalle</th> </tr> <tr> <td><input type="text"/></td> <td><input type="text"/></td> <td><input type="text"/></td> </tr> </table> Registros anteriores: <table border="1"> <tr> <th>Cedula</th> <th>Nombre</th> <th>Detalle</th> </tr> <tr> <td>123</td> <td>Pedro Paramo</td> <td>testigo</td> </tr> <tr> <td>64501234</td> <td>Ana rras</td> <td>demandada</td> </tr> </table>	Cédula	Nombre	Detalle	<input type="text"/>	<input type="text"/>	<input type="text"/>	Cedula	Nombre	Detalle	123	Pedro Paramo	testigo	64501234	Ana rras	demandada	Documentos Nombre del archivo <input type="text"/> Detalles del archivo <input type="text"/> Archivo <input type="text" value="Elegir archivo"/> No se eligió ningún archivo <input type="button" value="Subir documento"/>	Registros anteriores: <table border="1"> <tr> <th>Fecha</th> <th>Nombre</th> <th>Detalles</th> <th>Ver</th> </tr> <tr> <td>25/11/2021</td> <td>prueba de video</td> <td>prueba de video #1</td> <td>abigail_1ro_sem_1 oct.mp4</td> </tr> <tr> <td>25/11/2021</td> <td>prueba1</td> <td>esto es una prueba de pdf</td> <td>pdfCedula.docx</td> </tr> </table>	Fecha	Nombre	Detalles	Ver	25/11/2021	prueba de video	prueba de video #1	abigail_1ro_sem_1 oct.mp4	25/11/2021	prueba1	esto es una prueba de pdf	pdfCedula.docx
Cédula	Nombre	Detalle																											
<input type="text"/>	<input type="text"/>	<input type="text"/>																											
Cedula	Nombre	Detalle																											
123	Pedro Paramo	testigo																											
64501234	Ana rras	demandada																											
Fecha	Nombre	Detalles	Ver																										
25/11/2021	prueba de video	prueba de video #1	abigail_1ro_sem_1 oct.mp4																										
25/11/2021	prueba1	esto es una prueba de pdf	pdfCedula.docx																										
Notificaciones Título <input type="text"/> Detalles <input type="text"/> <input type="button" value="Elegir archivo"/> No se eligió ningún archivo <input type="button" value="Subir notificación"/>																													

Registros anteriores:

Fecha	Nombre	Detalles	Ver
27/11/2021	notificacion #1	prueba de notificación	informe13junio.pdf

Detalles acerca del expediente

expediente numero dos

Depósitos

Fecha: 01/01/0001

Banco:

Cuenta:

Monto: 0,00

Detalles del depósito:

Registros anteriores:

Fecha	Monto	Detalle	Banco	Cuenta
10/10/2021	50000,00	perito	BCR	1234567890098765432

Agregar detalles acerca de los cambios

Detalles de la modificación al expediente:

Registros anteriores:

Cambios	Fecha

Agregar apelación o casación

Tipo: Seleccione el tipo

Fecha: 01/01/0001

Detalles:

Registros anteriores:

Tipo	Detalle	Fecha

Modificar expediente

Figura 109. Pantalla expediente registrado.

Fuente: elaboración propia.

Módulo de Mantenimiento

Vista para abogado

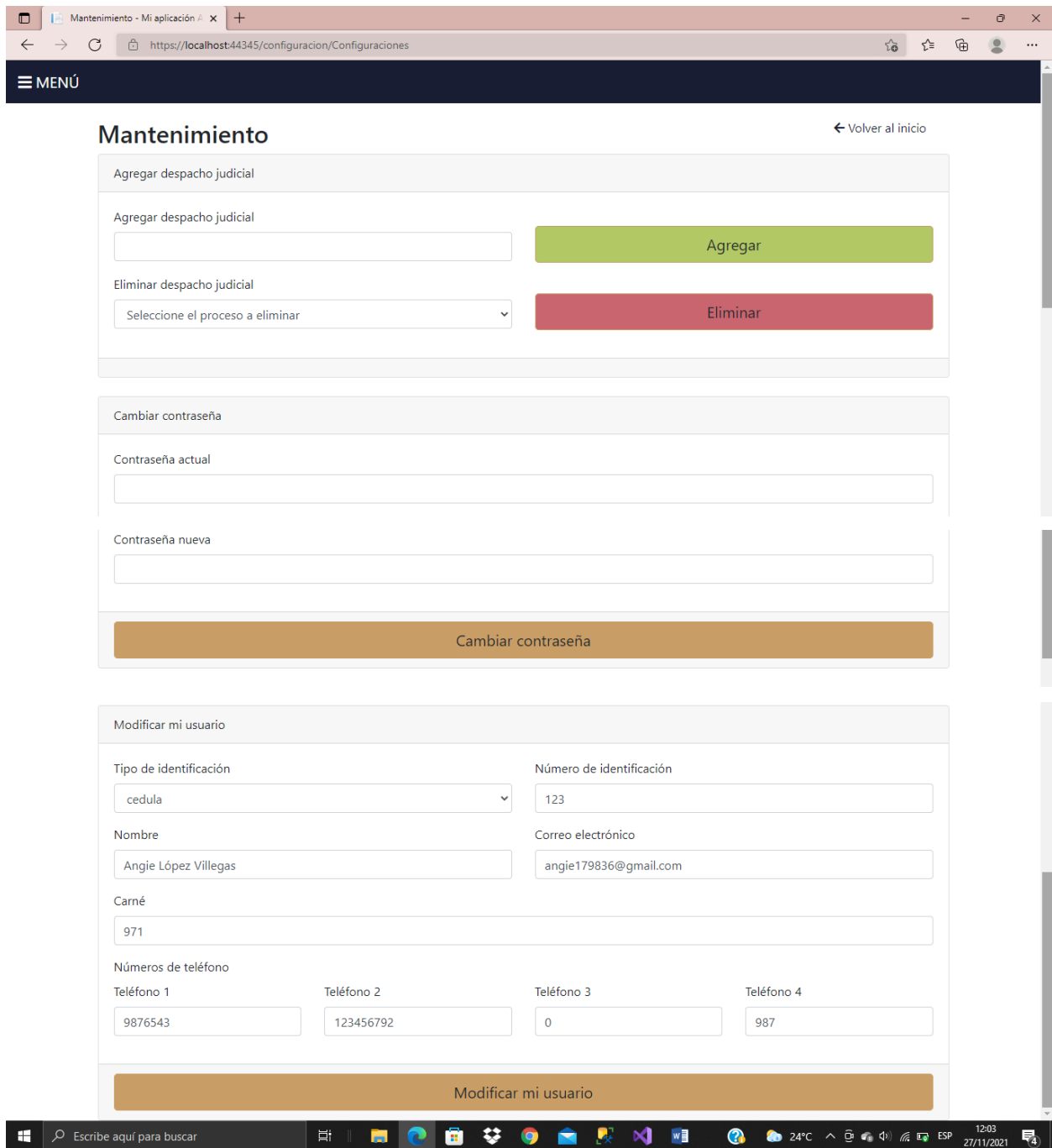


Figura 110. Pantalla de mantenimiento, vista para abogado.

Fuente: elaboración propia.

Vista para administrador

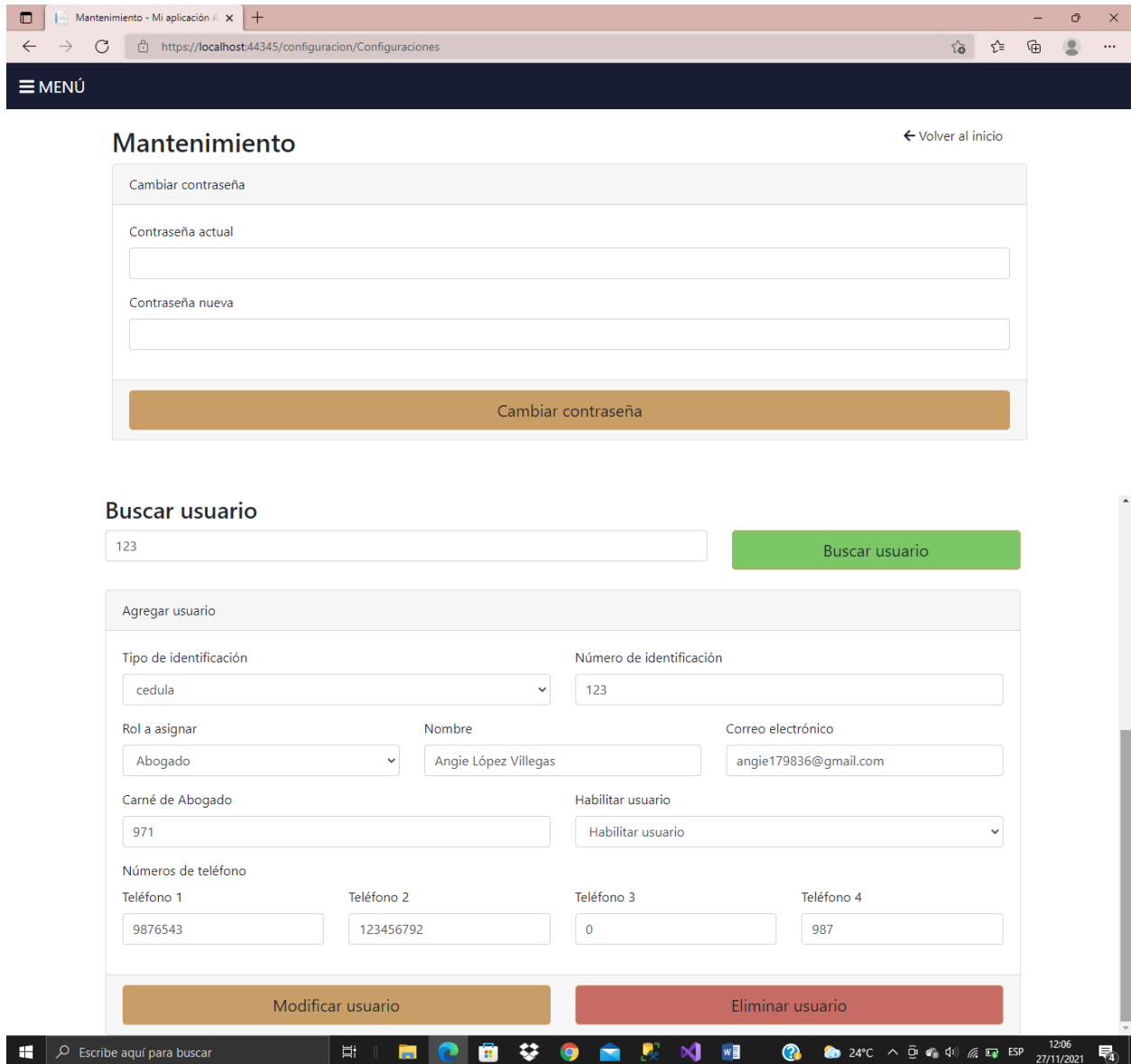


Figura 111. Pantalla mantenimiento para administrador.

Fuente: elaboración propia.

Módulo de clientes

Vista principal

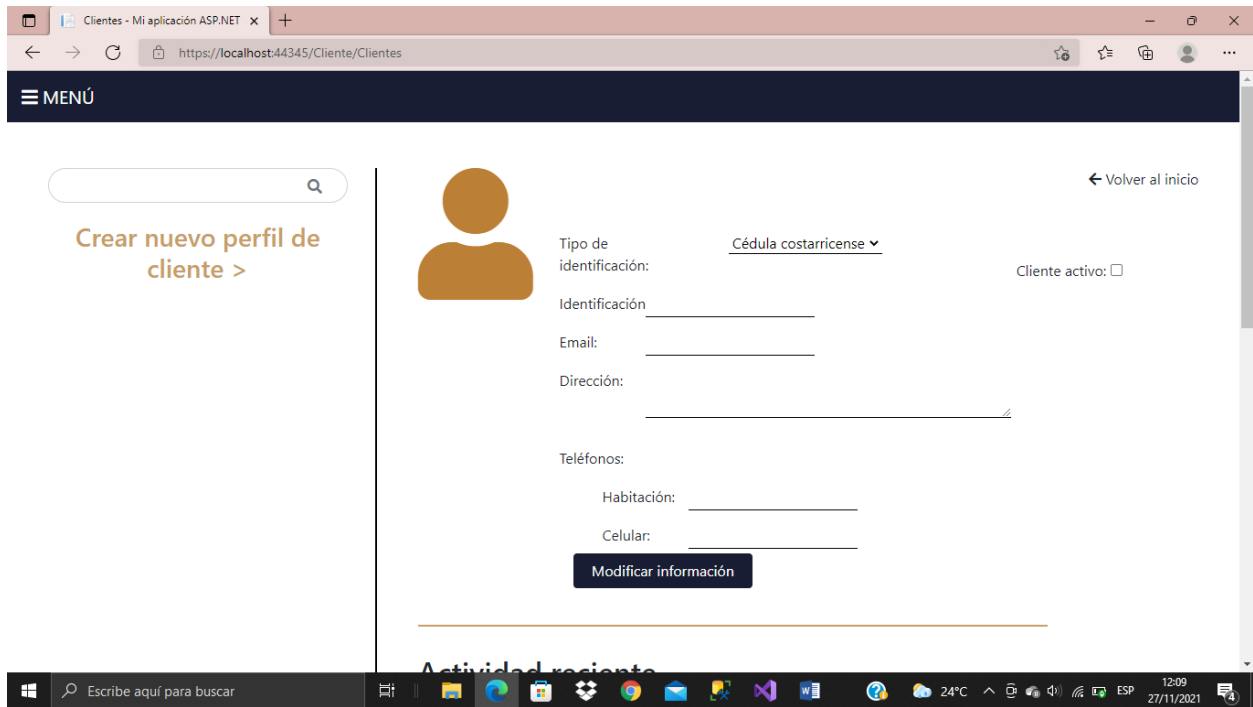


Figura 112. Pantalla de módulo de clientes, vista principal.

Fuente: elaboración propia.

Vista con búsqueda de cliente

Cientes - Mi aplicación ASP.NET x +
https://localhost:44345/Ciente/Cientes

MENÚ

a


Crear nuevo perfil de cliente >

Juanita María Contreras

Angie

Angie López Villegas

María Leticia de los Dolores Rojas Madrigal

 **Juanita María Contreras** [← Volver al inicio](#)

Tipo de identificación: Cédula costarricense Cliente activo:

Identificación: 12345

Email: angie179836@gmail.com

Dirección: sdfgbnm

Teléfonos:

Habitación: 0234

Celular: 1

Modificar información

Actividad reciente

Citas

Cita 1	Cita 2	Cita 3	Cita 4	Cita 5
Fecha: dd/mm/aaaa	Fecha: dd/mm/aaaa	Fecha: dd/mm/aaaa	Fecha: dd/mm/aaaa	Fecha: dd/mm/aaaa
Ver detalles	Ver detalles	Ver detalles	Ver detalles	Ver detalles

● ● ●

Emails enviados al cliente

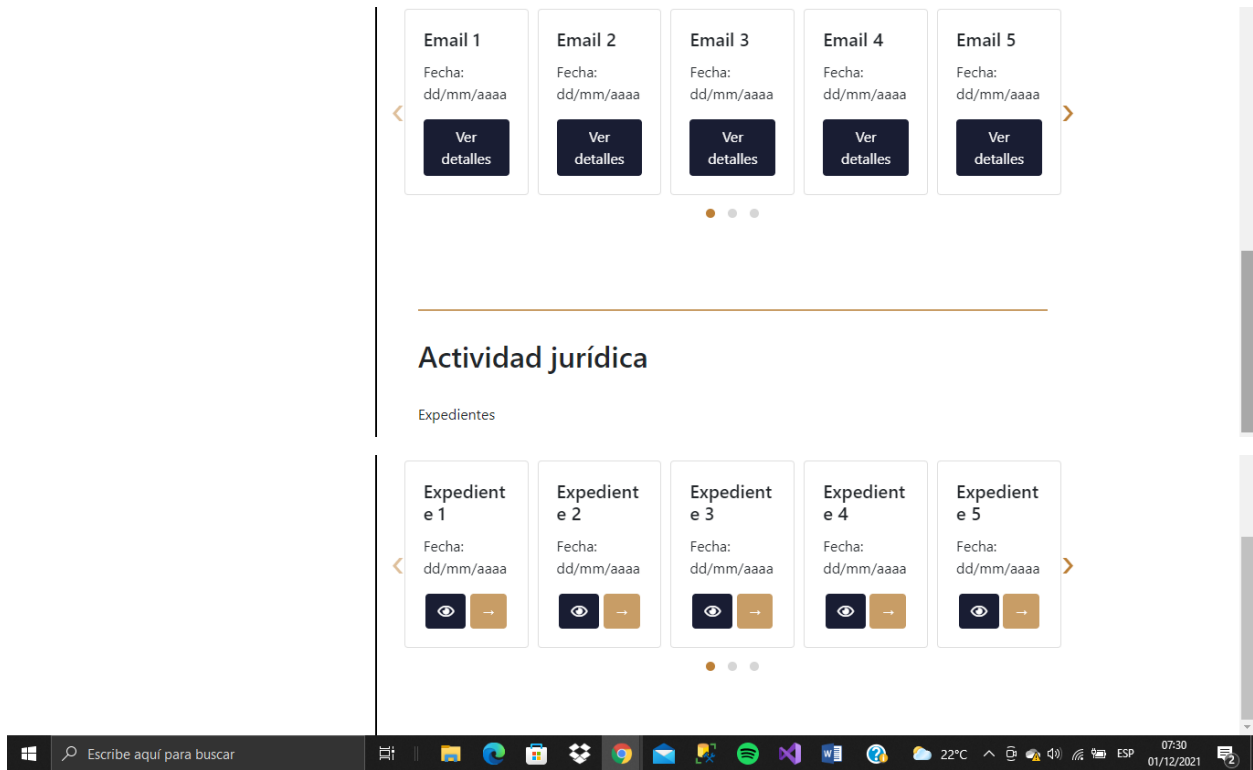


Figura 113. Pantalla de clientes, vista con cliente cargado.

Fuente: elaboración propia.

Crear cliente

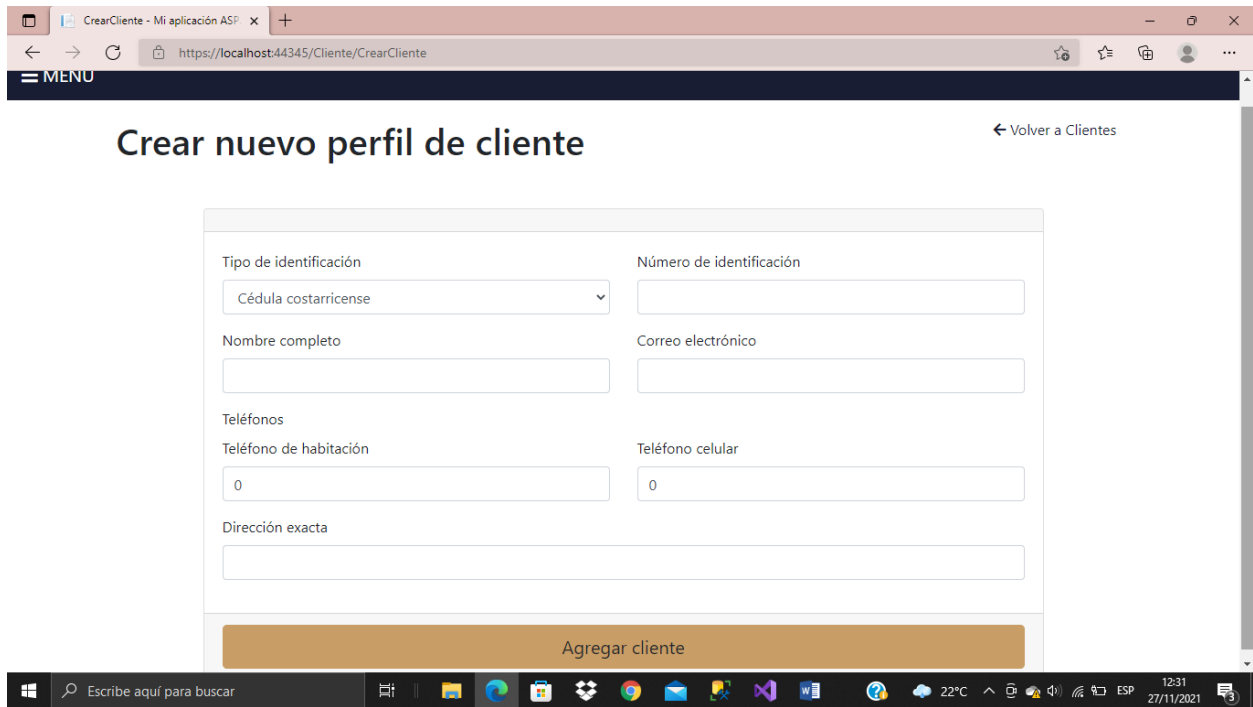


Figura 114. Pantalla crear cliente.

Fuente: elaboración propia.

Módulo de agenda

Pantalla principal

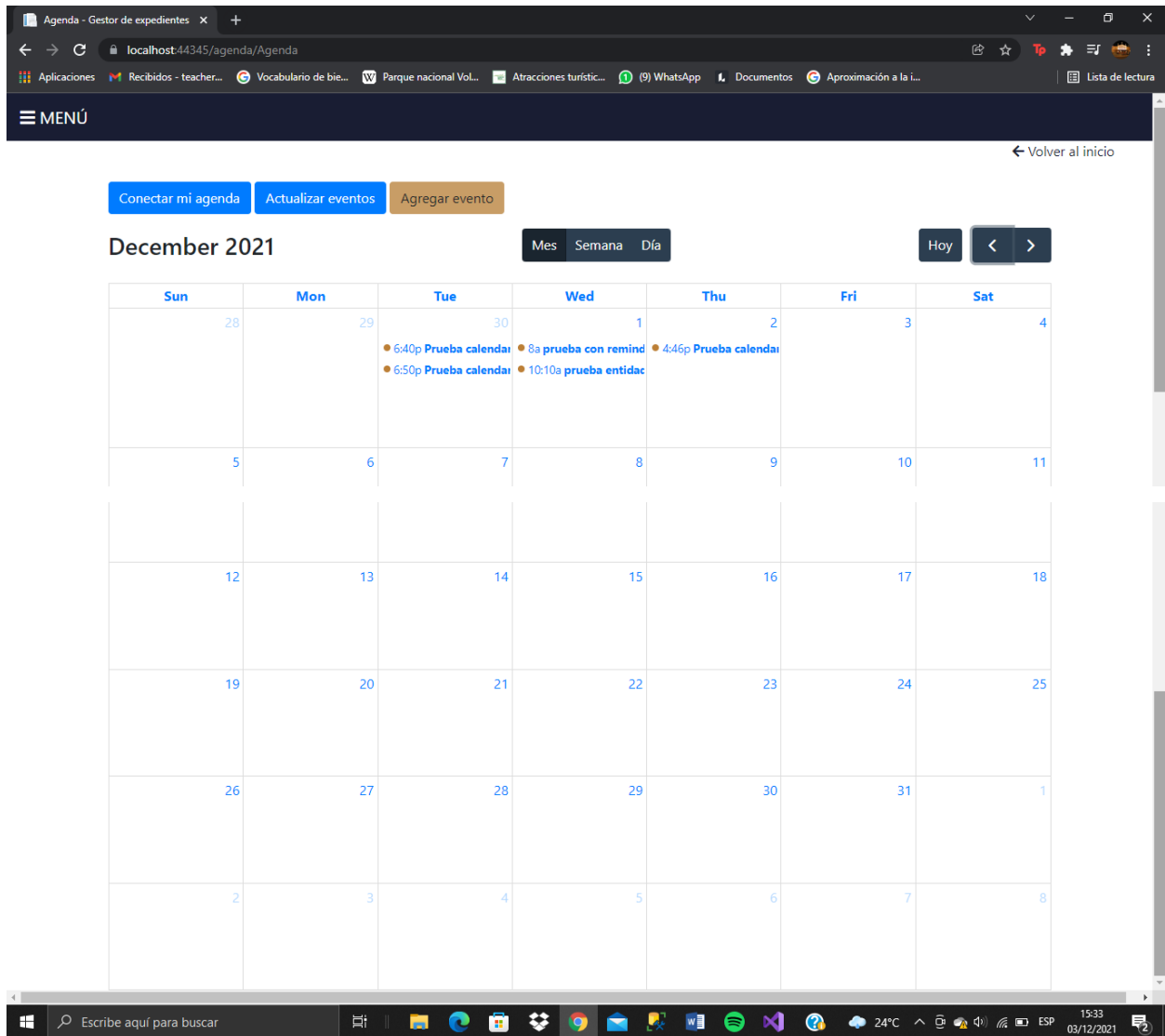


Figura 115. Pantalla principal.

Fuente: elaboración propia.

Pantalla crear evento

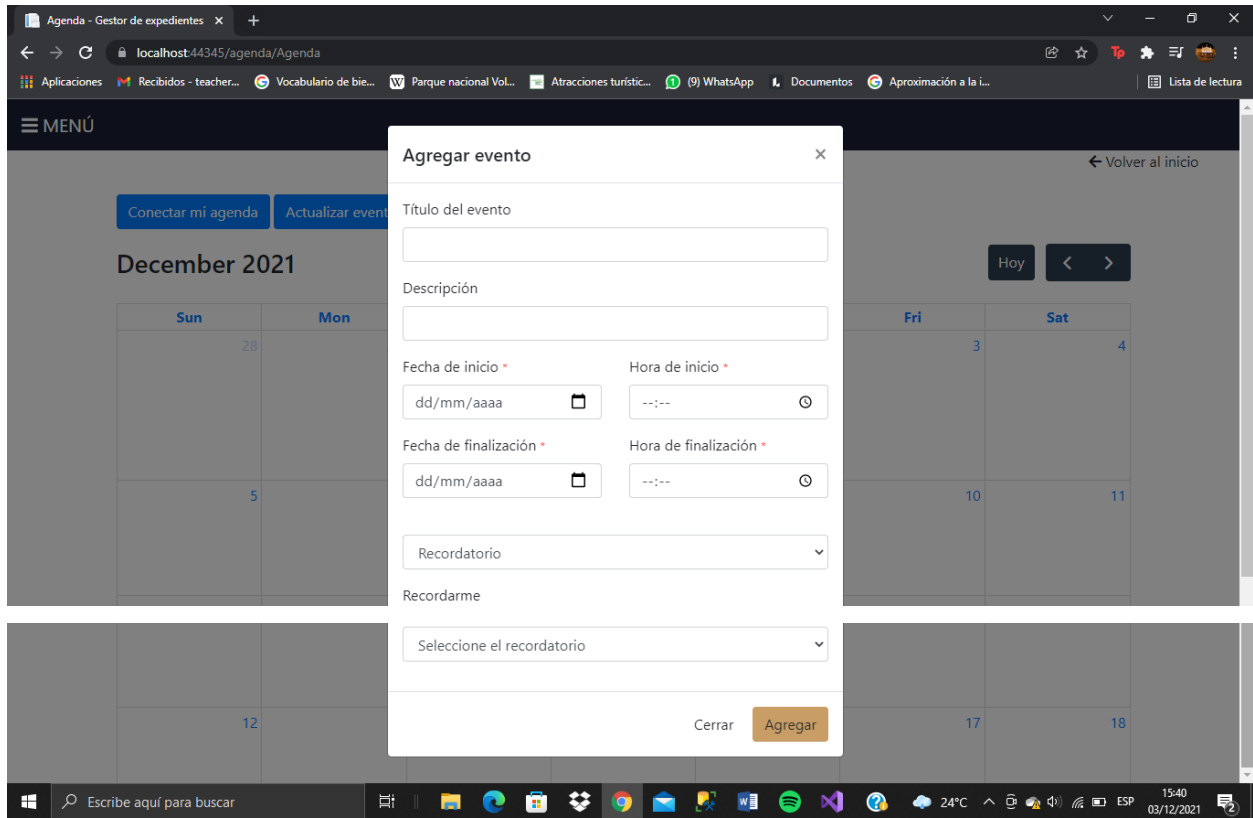


Figura 116. Pantalla agregar evento.

Fuente: elaboración propia.

Pantalla modificar evento

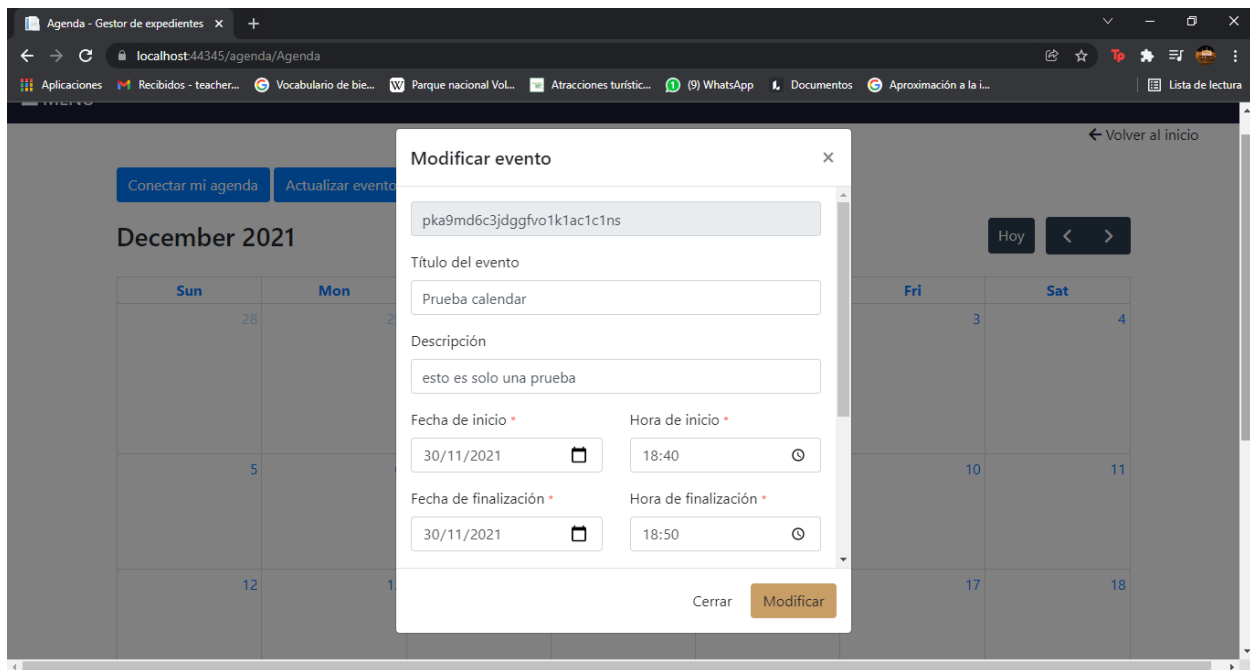


Figura 117. Pantalla de modificar evento, parte 1.

Fuente: elaboración propia.

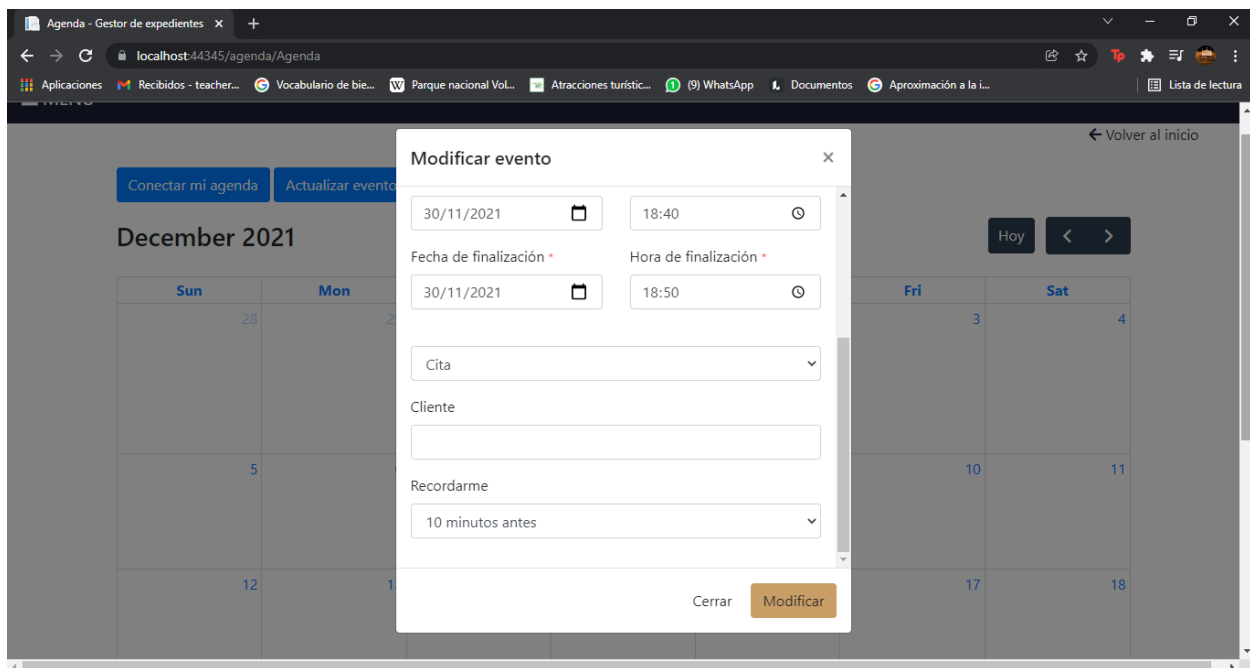


Figura 118. Pantalla de modificar evento, parte 2.

Fuente: elaboración propia.

5.4 Implementación del proyecto

En esta última parte del capítulo se habla acerca de la última fase, la implementación. Por lo que se hablará acerca de los detalles de esta.

5.4.1 Implementación

Para esta última fase del proyecto, luego del desarrollo exitoso del sistema de gestión de expedientes jurídicos, citas y recordatorios, se procede a alojar el sistema en un web hosting para que el cliente pueda hacer uso del mismo.

El proveedor seleccionado fue somee.com, que es un servicio de web hosting para Windows, por lo que la aplicación, que fue realizada en ASP.NET MVC, podrá ser alojada allí.

Las características que posee el web hosting seleccionado son las siguientes:

- SSL gratis.
- Capacidad de almacenamiento de 5GB.
- Dos bases de datos MS SQL de tamaño de 1GB cada una, y con tamaño de almacenamiento de backups de 4GB.
- Soporte técnico 24/7.
- Soporta ASP.NET 4.7 y MVC 5.
- Posee las licencias para IIS y MS SQL.

Luego de esto, se realiza una capacitación para el abogado sobre cómo utilizar el sistema. Esto se realiza con el fin de garantizar la adecuada utilización del sistema y evitar problemas o malentendidos con respecto al funcionamiento del sistema.

CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES

Introducción capítulo VI

En este capítulo se exponen las conclusiones a las que se llegaron durante la investigación realizada y las recomendaciones del caso.

6.1 Conclusiones

El objetivo general de este proyecto es: “Desarrollar e implementar un sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados, para la optimización del manejo de la información dentro del mismo.” Se concluye que el objetivo fue cumplido, el sistema desarrollado optimiza el manejo de información del bufete y cuenta con la aprobación del abogado y dueño del bufete. El éxito de este proyecto se debe al cumplimiento de cada uno de los objetivos específicos, las conclusiones de estos se describen a continuación.

- Se recolectó y analizó toda la información necesaria para la creación de los requerimientos funcionales y no funcionales del sistema. Esta información fue recolectada por medio de entrevistas con el abogado, adjuntadas en los anexos, y dieron como resultado los requerimientos funcionales y no funcionales para el sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados, paso fundamental para el desarrollo del mismo.
- Con ayuda de los requerimientos recolectados se diseñó el sistema por medio de diagramas UML, los diagramas que se hicieron fueron los de caso de uso, de clase, de secuencia y de actividades. Así mismo se realizó el diagrama de entidad-relación para la base de datos. Todo esto siendo la guía para el desarrollo del sistema.
- Se desarrolló el sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados. Se desarrollaron los módulos de clientes, expedientes, agenda, reportes, mantenimiento y honorarios, también se desarrolló la base de datos en SQL Server; utilizando como base y guía los requerimientos recolectados en la fase de recolección y análisis y los diagramas generados en la fase de diseño. Dando como resultado final el sistema web propuesto para solventar las necesidades del bufete.
- Se implementó el proyecto desarrollado del sistema web de gestión de expedientes jurídicos, citas y recordatorios para el bufete Fonseca Sequeira y asociados. Se subió a un web host y se le brindó la capacitación necesaria para el uso del sistema y gracias a eso el abogado puede acceder y hacer uso de él.

6.2 Recomendaciones

- Se recomienda, en caso de cualquier duda o problema con el sistema implementado, contactar en primera instancia con la desarrolladora del sistema.
- Se recomienda contratar un servicio de internet en la oficina de Jicaral, o aumentar el plan que tiene en su teléfono celular, para tener una mejor y más satisfactoria experiencia utilizando el sistema mientras se encuentra en esa oficina.
- Se recomienda considerar posibles funcionalidades que puedan ser implementadas en el sistema con el fin de tener un sistema cada vez más completo y robusto, siempre considerando que sean funcionalidades que solventen necesidades que posea el bufete.
- Se aconseja realizar respaldos de la base de datos periódicamente para evitar la pérdida de información almacenada en caso de un accidente o fallo. Estos respaldos se deben hacer, preferiblemente, fuera del horario laboral, para evitar entorpecer el desempeño del sistema y la productividad del bufete.
- Se recomienda realizar un manual de instrucciones con el paso a paso de cada una de las funciones del sistema, esto con la función de tener documentado cómo funciona el sistema en cuestión, y pueda servir de referencia al abogado y a futuros nuevos usuarios del sistema para entenderlo y lograr el buen uso del mismo.

CAPÍTULO VII: APÉNDICES Y ANEXOS

7.1 Referencias Bibliográficas

- Casillas, M., Martinell, A. R., Luna, M., & Marini, V. (2017). Ensayo de definición del perfil tecnológico del abogado. El abogado actual.
- Corrales, M., Fenwick, M., Haapio, H., & Vermeulen, E. P. M. (2019). Tomorrow's Lawyer Today? Platform-Driven Legaltech, Smart Contracts & the New World of Legal Design. *Journal of Internet Law*, 22(10), 3–12.
- Diccionario Usual del Poder Judicial - caso. (2021). Recuperado el 15 de julio 2021, de <https://dictionariusual.poder-judicial.go.cr/index.php/diccionario/52264:caso>
- Diccionario Usual del Poder Judicial - notificación. (2021). Recuperado el 15 de julio 2021, de <https://dictionariusual.poder-judicial.go.cr/index.php/diccionario/43915:notificaci%C3%B3n>
- Casado, L. (2016). *Diccionario de derecho*. Valleta Ediciones. Recuperado de <https://books.google.es/books?id=EiOFDwAAQBAJ>
- Pressman. (2021). *Ingeniería De Software* (7.^a ed.). MCGRAW HILL EDDUCATION. Recuperado de http://www.javier8a.com/umb/analisis/Libro_Pressman_7.pdf
- Mora-Pisco, L. L., Duran-Vasco, M. E., & Zambrano-Loor, J. G. (2016). Consideraciones actuales sobre gestión empresarial. *Dominio de las Ciencias*, 2(4), 511–520. Recuperado de <https://dominiodelasciencias.com/ojs/index.php/es/article/view/276>
- Noriega, R., Lainez, J. R., & Durango, A. (2017). *Curso de Ingeniería de Software: 2a Edición* (2.a ed.). Recuperado de https://books.google.es/books?id=G2Q4DgAAQBAJ&dq=software&lr=&hl=es&source=gbs_navlinks_s
- Palomo, S. R. G., & Gil, E. M. (2020). *Aproximación a la ingeniería del software*. Recuperado de https://books.google.es/books?id=8wnUDwAAQBAJ&dq=ciclo+de+vida+del+software&lr=&hl=es&source=gbs_navlinks_s
- Laudon, K. C., & Laudon, J. P. (2016). *Sistemas De Información Gerencial* (14.a ed.). Recuperado de http://cotana.informatica.edu.bo/downloads/ld-Sistemas_de_informacion_gerencial_14%20edicion.pdf
- Ávila, E. (2017). *Introducción a Internet*.

- Tsatsou, P. (2014). *Internet Studies : Past, Present and Future Directions*. Routledge.
Recuperado de <http://search.ebscohost.com/login.aspx?direct=true&db=e000xww&AN=790654&lang=es&site=eds-live&scope=site>.
- Torres, D. G. C., & Cuevas, D. G. L. C. (2008). *Diccionario juridico elemental / Legal Elemental Dictionary* (19.a ed.). Bs. As., Argentina: Heliasta.
- Marini, E. (2012). El modelo cliente/servidor. Recuperado el 22 de julio 2021, 5.
Recuperado de <https://www.linuxito.com/docs/el-modelo-cliente-servidor.pdf>
- Robbins, J. N. (2012). *Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics* (4.^a ed.).
- Duckett, J., Ruppert, G., & Moore, J. (2014). *Javascript & Jquery: Interactive Front-End Web Development*.
- Spurlock, J. (2013). *Bootstrap*. Recuperado de https://books.google.es/books?id=LZm7Cxgi3aQC&dq=bootstrap+spurlock&lr=&hl=es&source=gbs_navlinks_s
- Deutsch, D. (s. f.). Home | Owl Carousel | 2.3.4. Recuperado 22 de julio de 2021, de <https://owlcarousel2.github.io/OwlCarousel2/>
- Wulf, J., & Blohm, I. (2020). Fostering Value Creation with Digital Platforms: A Unified Theory of the Application Programming Interface Design. *Journal of Management Information Systems*, 37(1), 251–281. Recuperado de <https://doi.org/10.1080/07421222.2019.1705514>
- Sagdeo, P. (2018). Application Programming Interfaces and the Standardization-Value Appropriation Problem. *Harvard Journal of Law & Technology* (Harvard JOLT), 32(1), 235–264.
- Kendall, K. E., & Kendall, J. E. (2005). *Analisis Y Diseño De Sistemas* (6.^a ed.). Harlow, Reino Unido: Pearson Educación.
- Wiegers, K., & Beatty, J. (2013). *Software Requirements* (3.^a ed.). Recuperado de https://books.google.es/books?id=nbpCAwAAQBAJ&dq=software&lr=&hl=es&source=gbs_navlinks_s

- Talledo San Miguel, J. V. (2015). Implantación de aplicaciones web en entorno internet, intranet y extranet MF0493_3 (1.a ed.). Recuperado de https://books.google.es/books?id=RtESCgAAQBAJ&dq=dominio+web+concepto&lr=&hl=es&source=gbs_navlinks_s
- Menzinsky, A., López, G., Palacio, J., Sobrino, M. Á., Álvarez, R., & Rivas, V. (2018). Historias de Usuario. Ingeniería de requisitos ágil. Recuperado de https://scrummanager.net/files/scrum_manager_historias_usuario.pdf
- Arias Chaves, M. (2011). La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. InterSedes, 6(10). Recuperado de <https://revistas.ucr.ac.cr/index.php/intercedes/article/view/790>
- López, R. O., & Ramiro, L. V. (2011). Ingeniería del software. Centro de Estudios Financieros. Recuperado de <https://ocw.unican.es/pluginfile.php/1403/course/section/1792/is1-t02-trans.pdf>
- Aguilera, S. (2015). Diagrama Entidad Relación. Recuperado de http://repositorio.ub.edu.ar/bitstream/handle/123456789/5155/DER2_2014.pdf?sequence=1&isAllowed=y
- Hugon, J. (2018). C# 7: desarrolle aplicaciones Windows con Visual Studio 2017. Ediciones ENI. Recuperado de https://books.google.es/books?id=1e9dfieV4UEC&dq=c%23+hugon&lr=&hl=es&source=gbs_navlinks_s
- Hejlsberg, A., Torgersen, M., Wiltamuth, S., & Golde, P. (2008). The C# Programming Language. Recuperado de https://books.google.es/books?id=ICe7ea4RscUC&dq=c%23&lr=&hl=es&source=gbs_navlinks_s
- Suárez, E. M. (2008). ¿Qué es una base de datos relacional?. Universidad de murcia, Murcia, España.
- Santamaría, J., & Hernández, J. (2016). Microsoft SQL Server. SQL SER vs MY SQL, 1-6. Recuperado de <https://iessanvicente.com/colaboraciones/sqlserver.pdf>
- San Martín Cuenca, H. D., & Tusa Jumbo, E. A. (2015). Fundamentos de programación para ciencias e ingeniería.
- Singh, R. R. (2015). Mastering Entity Framework. Birmingham, UK: Packt Publishing.

- Chadwick, J. (2011). Programming Razor. Recuperado de https://books.google.es/books?id=z3QIDSjr0jYC&dq=mvc+razor&lr=&hl=es&source=gs_navlinks_s
- Fernández Romero, Y., & Díaz González, Y. (2012). Patrón Modelo-Vista-Controlador. Telemática, 11(1), 47–57. Recuperado de <https://revistatelematica.cujae.edu.cu/index.php/tele/article/view/15/10>
- Arias, A. (2015). Aprende a Programar ASP .NET y C#: 2a Edición (2.a ed.). Recuperado de https://books.google.es/books?id=9WnjCgAAQBAJ&dq=c%23&lr=&hl=es&source=gs_navlinks_s
- Palacio, L. G. (2009). Método para generar casos de prueba funcional en el desarrollo de software. Revista Ingenierías Universidad de Medellín, 8(15 Sup. 1), 29-36. Recuperado de <https://revistas.udem.edu.co/index.php/ingenierias/article/view/175/164>
- Menchaca, M. (2014). *Derecho informático*. Recuperado de <https://www.derechopenalened.com/libros/derecho-informatico-marcelo-menchaca.pdf>
- Sánchez Martínez, F. E. (2012). ANÁLISIS COMPARATIVO DE SERVIDORES WEB: APACHE VS MICROSOFT IIS. Recuperado de <http://www.repositorio.unacar.mx/jspui/handle/1030620191/438>
- Gamboa, J. Z. (2018). Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software. INNOVA Research Journal, 3(10), 20-33. Recuperado de <https://dialnet.unirioja.es/servlet/articulo?codigo=6777227>
- Gallardo Ruiz J. E., García López, C. M. (s.f.). Apuntes para la informática. Málaga: Universidad de Málaga.
- Microsoft. (2019, 18 septiembre). Tablas - SQL Server. Recuperado 2 de agosto de 2021, de <https://docs.microsoft.com/es-es/sql/relational-databases/tables/tables?view=sql-server-ver15>
- Daen, S. T. (2011). Tipos de investigación científica. Revista de Actualización Clínica Investiga Boliviana. Recuperado de http://www.revistasbolivianas.org.bo/pdf/raci/v12/v12_a11.pdf

- Guerrero Bejarano, M. A. (2016). La investigación cualitativa. Recuperado de <https://repositorio.uide.edu.ec/handle/37000/3645>
- Torres, M., Paz, K., y Salazar, F. G. (s.f.). Métodos de recolección de datos para una investigación. Recuperado de <http://biblioteca.udgvirtual.udg.mx/jspui/handle/123456789/2817>
- Caro, L. (2021). 7 Técnicas e instrumentos para la recolección de datos. Lifeder. Recuperado el 09 setiembre 2021 de: <https://www.lifeder.com/tecnicas-instrumentos-recoleccion-datos/>

7.2 Anexos

7.2.1 Entrevistas con el abogado

Entrevista #1

1. ¿Qué sistema necesita que desarrolle?

Yo lo que ocupo es un sistema personalizado. Que me permita llevar los expedientes judiciales de forma electrónica, porque se pierde mucho tiempo yendo a buscar el expediente físico en cada reunión con el cliente. Si se puede que lleve el control de las citas y reuniones que tengo, como una agenda, es algo extra pero sería bueno tenerlo.

También sería bueno que haya un número de usuario, que sea la cédula y una clave para poder usar el programa.

2. ¿Qué debe tener la parte de expedientes?

Puede que usted lo haga con un diferente tipo de formato pero es muy parecido a lo que se hace con Excel, con espacios para rellenar.

Necesita tener una casilla de número de expediente, fecha inicio, juzgado se puede poner como despacho judicial, partes: actor y demandado, estado del proceso, notificaciones, esto son las comunicaciones o informaciones actuales que se están dando, por ejemplo, que se notificó una resolución; depósitos porque en algunos procesos judiciales existen los depósitos como por ejemplo que hay que depositar los honorarios de un perito, o que alguien ya ganó un proceso y tiene que depositar por lo que fue condenada la persona. Cuando hablamos de depósitos se debe poner fecha de depósito, banco y número de cuenta.

Esto es lo básico que se necesita, con solo esta guía uno trabaja bien, porque yo no necesito mucho, yo no soy una empresa grande. Y he visto en varios formatos que manejan algunos colegas que básicamente esto es lo que manejan.

Con respecto al juzgado, este casi nunca cambia, pero hay veces donde se manda el caso a otro juzgado, entonces en ese caso deben guardarse los dos juzgados en los que ha estado el expediente, el anterior y el nuevo. Creo que sí sería bueno, como usted dice hacer la parte de agregar el tipo de proceso o juzgado en una sección aparte como de configuraciones porque si se

hace una pestañita que despliega opciones solo va a tener las que ya tiene predefinidas y si aparece un trabajo de otro tipo de proceso que no se encuentra en la lista entonces no se va a poder hacer.

Una casilla de información adicional o detalles también sería bueno.

Algo bueno para agregar sería también una casilla que diga representante legal, porque en la mayoría de los casos uno inicia un caso nuevo, pero hay otras veces donde uno no inicia el caso, sino que llegan clientes con un caso que estaban llevando con otro abogado y que por una u otra razón deciden ya no trabajar con él y vienen donde uno, entonces sería bueno para que quede una referencia de quién estuvo llevando el caso y quién lo está llevando ahora.

Otro punto sería documentos, otra casilla, porque entonces yo puedo agregar los documentos que tiene, y puedo agregar por ejemplo: “se aporta facturas” y los voy adjuntando.

¿Será posible que, como yo tengo el correo de los clientes, desde el mismo programa enviarles un correo a ellos cada vez que yo suba una notificación que llegó de la corte? Porque a mí me llega una notificación de la corte y yo la tengo que subir al programa para mantenerlo actualizado, pero me gustaría que se enviara al correo del cliente para que ellos también estén enterados.

Por lo general cada expediente está relacionado a un único caso, pero hay excepciones donde puede haber más de un caso. Por ejemplo en peleas de herencia, los hermanos no se llevan o desconfían del abogado que lleva el proceso y cada uno decide buscar un abogado entonces a la corte llegan varios expedientes con un mismo propósito y relacionados a una sola cosa, entonces lo que hace el juez es agrupar todos los casos en un único expediente, el primero que se generó. Entonces tal vez sería bueno agregar algo para eso.

3. Con respecto a la parte de las citas, ¿cómo lo prefiere?

Con Gmail. Hace poco empecé a usar el calendario del teléfono y éste usa un correo de Gmail, por lo que prefiero seguir utilizando ese correo.

4. ¿El sistema necesita algo más como imprimir o generar algún reporte?

Sería genial, porque si un cliente me dice que quiere que le de la información del expediente yo lo genero, lo imprimo y se lo doy.

5. Para el registro de los clientes, ¿qué información es necesaria?

No mucho, el nombre, la cédula, correo electrónico, teléfonos. Campos como sexo, estado civil y esas cosas no son realmente necesarias, siento que es hacer demasiado bulto.

Entrevista #2

1. Se muestra un avance del diseño de la aplicación web se explica cómo funcionaría y se pide opinión al respecto sobre si se ve amigable y entendible para él, o si hacen falta espacios por agregar en los módulos diseñados.

1.1 Sobre el módulo de clientes:

Para el tipo de identificación las opciones necesarias solo serían: cédula costarricense, pasaporte y residencia. También hace falta la casilla de dirección exacta; no es necesario agregar provincia, distrito y cantón, con solo la dirección exacta es suficiente.

1.2 Sobre el módulo de configuraciones o mantenimiento:

Todo se ve bastante bien, los campos también están bien, no hace falta nada.

1.3 Sobre el módulo de expedientes:

Se ve bien, está detallado, completo y accesible. En la parte de apelación deberías cambiarlo y poner “Agregar apelación o casación”, son los mismos campos por llenar, pero el nombre sería mejor así. En el caso de que alguien me pidiera imprimir el expediente de ellos, usted me dice que puedo entrar a la parte de reportes y generarlo desde ahí, pero me gustaría también tener desde la misma pestaña de reportes como un iconito de imprimir para nada más apretar el botoncito, guardarlo en PDF y ya

luego imprimirlo. También un ícono que diga salir, como hay en algunas páginas de los bancos, un ícono de salir y me devuelva al menú principal.

2. ¿Hay algo más que se necesite?

Sería bueno una parte más en el menú que sea de cobros o de honorarios. Esta parte solo llevaría el registro de los honorarios. Entonces lleva los siguientes campos: el total a cobrar, el abono, la fecha en que se da el abono y el saldo, que se va restando conforme se van agregando abonos.

3. Fuera de revisiones de a aplicación, actualmente, ¿cómo es el proceso operativo? ¿Qué se hace cada vez que llega un cliente?

Bueno, se podría explicar por pasos.

Paso 1: la persona llega, se saluda, se sienta.

Paso 2: la persona empieza a explicar la situación.

Paso 3: una vez la persona termina de contar toda la situación se comienza a dar la asesoría. Se trata de una asesoría bien dada, de acuerdo a la ley, uno busca algunos códigos para tener un marco de referencia y le explica al cliente con base a la ley qué se puede hacer y qué no se puede hacer.

Paso 4: si la persona, el cliente está de acuerdo con todo lo dicho entonces se inicia la negociación y se da el precio. Si se logra consensuar entonces se establecen formas de pago.

Paso 5: Luego de todo esto, se solicitan las pruebas, documentos, etc.

Paso 6: Puede ese el mismo día u otro, sentarse a redactar ya sea la demanda o defensa dependiendo de lo que amerite el caso.

4. ¿Cuántas oficinas tiene actualmente?

Dos. Una aquí en Puntarenas y otra en Jicaral.

5. Se hace acuerdo de que el abogado envíe fotos del equipo de cómputo que utiliza para trabajar y las agendas donde escribe las citas y recordatorios.

7.2.2 Carta del abogado

Lic. César Fonseca Sequeira
Abogado & Notario

Telefax: 2661-47-50/Cel: 8398-88-71

Correo Electrónico: cesarfonseca3@hotmail.com

Especialidad en: Derecho Penal, Familia, Laboral y Criminología

Dirección 25 metros Este del Estadio Lito Pérez contiguo a Panadería Musmanni y en
Jicaral Centro costado Oeste del Cen – Cinai

Puntarenas, 09 de diciembre de 2021

Señores:

Universidad Hispanoamericana. Sede Puntarenas

Carrera Bachillerato en Ingeniería Informática

Estimados señores:

Quien suscribe **Lic. César Fonseca Sequeira**, mayor, casado, abogado y notario, vecino de Puntarenas, con cédula de identidad número 602470588 y número de Carnet 11723, por medio de la presente me permito saludarle y a la vez indicarles que la joven **ANGIE SUGEY LÓPEZ VILLEGAS**, con cédula 604460167 me hizo entrega el Proyecto de Sistema Web de Gestión de expedientes jurídicos, citas y recordatorios para mi Bufete Fonseca Sequeira y Asociados, lo cual indico que me encuentro muy satisfecho del funcionamiento del Programa y se que será una herramienta muy útil y necesaria para eficiencia y agilidad en el manejo de expedientes electrónicos de mis clientes.

Sin otro particular se despide de usted muy atentamente.


Lic. César Fonseca Sequeira
Abogado & Notario



Dirección del Estadio Lito Pérez 25 metros al Este contiguo a Panadería Musmanni
Horario de Lunes a Viernes de 8:00 a.m. a 6:00 p.m.