



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL**

CARRERA DE TEGNOLOGÍAS DE LA INFORMACIÓN

Tema:

**APLICATIVO WEB UTILIZANDO FRAMEWORK OPEN SOURCE PARA
LA GESTIÓN DE ACTIVIDADES DE LA FUNDACIÓN DE RESCATE
ANIMAL PAQUITO DE LA CIUDAD DE AMBATO**

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la obtención del título de Ingeniera en Tecnologías de la Información.

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de Software

AUTOR: Freddy Santos Remache Gañay

TUTOR: Ing. Hernán Fabricio Naranjo Ávalos, Mg.

Ambato - Ecuador

febrero – 2024

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: APLICATIVO WEB UTILIZANDO FRAMEWORK OPEN SOURCE PARA LA GESTIÓN DE ACTIVIDADES DE LA FUNDACIÓN DE RESCATE ANIMAL PAQUITO DE LA CIUDAD DE AMBATO, desarrollado bajo la modalidad Proyecto de Investigación por el señor Freddy Santos Remache Gañay, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, febrero 2024.

Ing. Hernán Fabricio Naranjo Ávalos, Mg.

TUTOR

AUTORÍA

El presente trabajo de titulación con el tema: APLICATIVO WEB UTILIZANDO FRAMEWORK OPEN SOURCE PARA LA GESTIÓN DE ACTIVIDADES DE LA FUNDACIÓN DE RESCATE ANIMAL PAQUITO DE LA CIUDAD DE AMBATO es absolutamente original, auténtico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, febrero 2024.



Freddy Santos Remache Gañay

C.C. 0604833525

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, febrero 2024.



Freddy Santos Remache Gañay

C.C. 0604833525

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por el señor Freddy Santos Remache Gañay, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado **APLICATIVO WEB UTILIZANDO FRAMEWORK OPEN SOURCE PARA LA GESTIÓN DE ACTIVIDADES DE LA FUNDACIÓN DE RESCATE ANIMAL PAQUITO DE LA CIUDAD DE AMBATO**, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, febrero 2024.

Ing. Elsa Pilar Urrutia Urrutia, Mg.
PRESIDENTE DEL TRIBUNAL

Ing. Carlos Nuñez Miranda, Mg
PROFESOR CALIFICADOR

Ing. Franklin Mayorga Mayorga, Mg
PROFESOR CALIFICADOR

DEDICATORIA

El presente proyecto se encuentre dedicado a mis padres, María Gañay y Santos Remache, cuya inquebrantable asistencia ha sido fundamental a lo largo de mi trayectoria. Con su constante apoyo, afecto y respaldo, me han infundido la energía necesaria para alcanzar mis metas.

Así mismo, hacia mis hermanas quienes han sido mis guías incansables en cada etapa del recorrido. Su cariño y paciencia me han brindado la oportunidad de aprender y crecer en todos los aspectos de mi vida.

Esta dedicatoria también se extiende a todas aquellas personas que han sido un pilar en mi camino. A aquellos que me acompañaron tanto en mis momentos más brillantes como en los desafíos más arduos. A quienes, con su inquebrantable paciencia, han sabido escuchar y orientarme de manera valiosa.

Freddy Santos Remache Gañay

AGRADECIMIENTO

En primer lugar, deseo expresar mi más sincero agradecimiento a mi familia, por estar en las buenas y en las malas junto a mí, apoyándome y alentándome.

También debo reconocer y agradecer a mi distinguido profesor y mentor, el Ingeniero Hernán Naranjo. Su apoyo y enseñanzas tanto en el aula como en el desarrollo de este proyecto han sido de inmenso valor.

Finalmente, extiendo mi gratitud a mis amigos que han compartido experiencias en la universidad. Cada uno de ustedes ha dejado una huella imborrable en mi memoria, y estoy agradecido por todas las vivencias compartidas.

Freddy Santos Remache Gañay

ÍNDICE GENERAL DE CONTENIDOS

PORTADA.....	i
APROBACIÓN DEL TUTOR.....	ii
AUTORÍA.....	iii
DERECHOS DE AUTOR	iv
APROBACIÓN DEL TRIBUNAL DE GRADO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
ÍNDICE GENERAL DE CONTENIDOS.....	viii
ÍNDICE DE TABLAS	x
ÍNDICE DE FIGURAS.....	xiv
ÍNDICE DE ANEXOS.....	xvii
RESUMEN EJECUTIVO	xviii
ABSTRACT.....	xix
CAPÍTULO I. MARCO TEÓRICO.....	1
1.1 Tema de investigación.....	1
1.1.1 Planteamiento del problema.....	1
1.2 Antecedentes investigativos	2
1.3 Fundamentación teórica	4
1.4 Objetivos	11
1.4.1 Objetivo general	11
1.4.2 Objetivos específicos	11
CAPÍTULO II. METODOLOGÍA.....	13
2.1 Materiales.....	13
2.2 Métodos.....	17
2.2.1 Modalidad de la investigación	17
2.2.2 Población y muestra	17
2.2.3 Recolección de información.....	18
2.2.4 Procesamiento y análisis de datos	32

CAPÍTULO III. RESULTADOS Y DISCUSIÓN.....	34
3.1 Análisis y discusión de los resultados	34
3.1.1 Modelado de procesos.....	34
3.2.1 Tecnologías open source para el desarrollo frontend en aplicativos web 37	
3.1.2 Tecnologías open source para el desarrollo backend en aplicativos web 40	
3.1.3 Determinación de metodología para desarrollo del aplicativo web	42
3.1.4 Arquitectura del aplicativo web	45
3.2 Desarrollo de la propuesta.....	45
3.2.1 Fase I: Planificación	46
3.2.2 Fase 2: Diseño	87
3.2.3 Fase 3: Codificación.....	107
3.2.4 Fase 4: Pruebas.....	128
3.2.5 Fase 5: Lanzamiento	141
3.3 Análisis de tiempos estimados antes y después de la automatización de procesos.	142
CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES.....	144
4.1 Conclusiones	144
4.2 Recomendaciones.....	144
REFERENCIAS BIBLIOGRÁFICAS.....	146
ANEXOS	150

ÍNDICE DE TABLAS

Tabla 1. Población de Estudio.....	17
Tabla 2. Coeficiente Kuder-Richardson en la encuesta para personas que están en proceso de adopción de mascotas.	19
Tabla 3. Interpretación de la escala de consistencia de Kuder-Richardson.	19
Tabla 4. Confiabilidad Alfa de Cronbach en la encuesta para personas que están en proceso de adopción de mascotas.	20
Tabla 5. Confiabilidad Alfa de Cronbach en la encuesta para personas que están en proceso de adopción de mascotas.	20
Tabla 6. Resultados pregunta 1.	23
Tabla 7. Resultados pregunta 2.	24
Tabla 8. Resultados pregunta 3.	25
Tabla 9. Resultados pregunta 4.	26
Tabla 10. Resultados pregunta 5.	27
Tabla 11. Resultados pregunta 6.	28
Tabla 12. Resultados pregunta 7.	29
Tabla 13. Resultados pregunta 8.	30
Tabla 14. Resultados pregunta 9.	30
Tabla 15. Resultados pregunta 10.	31
Tabla 16. Comparación de tecnologías open source de desarrollo frontend.....	38
Tabla 17. Comparación de tecnologías open source de desarrollo backend.....	40
Tabla 18. Comparación de metodologías ágiles.	42
Tabla 19. Asignación de roles XP.....	47
Tabla 20. Requerimientos de implementación.....	47
Tabla 21. Requerimientos de implementación del aplicativo web.	49
Tabla 22. Modelo de historia de usuario.....	51
Tabla 23. Historia de usuario 1: Creación de la estructura del aplicativo web.....	51
Tabla 24. Actividad 1: Diseño conceptual de la base de datos del aplicativo web....	52
Tabla 25. Actividad 2: Creación del proyecto en backend.	52
Tabla 26. Actividad 3: Creación del proyecto en frontend.	52
Tabla 27. Historia de usuario 2: Formulario de registro.	52
Tabla 28. Actividad 4: Diseño de la interfaz de usuario.	53
Tabla 29. Actividad 5: Creación del modelo entidad relación del formulario de registro.....	53
Tabla 30. Actividad 6: Codificación del formulario de registro.	54
Tabla 31. Actividad 7: Pruebas Unitarias.	54
Tabla 32. Historia de usuario 3: Interfaz de autenticación y autorización.....	55
Tabla 33. Actividad 8: Diseño de la interfaz de usuario.	55
Tabla 34. Actividad 9: Creación del modelo entidad relación de la interfaz de autenticación y autorización.....	55
Tabla 35. Actividad 10: Codificación de la interfaz de autenticación y autorización.	56
Tabla 36. Actividad 11: Pruebas Unitarias.	56
Tabla 37. Historia de usuario 4: Interfaz para actualizar información del usuario. ...	57

Tabla 38. Actividad 12: Diseño de la interfaz de usuario.....	58
Tabla 39. Actividad 13: Creación del modelo entidad relación para actualizar información del usuario.....	58
Tabla 40. Actividad 14: Codificación de la interfaz para actualizar información del usuario.....	58
Tabla 41. Actividad 15: Pruebas Unitarias.	58
Tabla 42. Historia de usuario 5: Interfaz para asignar perfiles de usuario.....	59
Tabla 43. Actividad 16: Diseño de la interfaz de usuario.....	60
Tabla 44. Actividad 17: Codificación de la interfaz para asignar perfiles de usuario.	60
Tabla 45. Actividad 18: Pruebas Unitarias.	60
Tabla 46. Historia de usuario 6: Interfaz para gestionar información de las mascotas.	61
Tabla 47. Actividad 19: Diseño de la interfaz de usuario.....	62
Tabla 48. Actividad 20: Creación del modelo entidad relación para la gestión de la información de las mascotas.	62
Tabla 49. Actividad 21: Codificación del formulario.....	62
Tabla 50. Actividad 22: Pruebas Unitarias.	63
Tabla 51. Historia de usuario 7: Formulario para registrar información médica de las mascotas.	63
Tabla 52. Actividad 23: Diseño de la interfaz de usuario.....	64
Tabla 53. Actividad 24: Creación del modelo entidad relación del formulario para registrar información médica de las mascotas.	64
Tabla 54. Actividad 25: Codificación del formulario.....	65
Tabla 55. Actividad 26: Pruebas Unitarias.	65
Tabla 56. Historia de usuario 8: Catálogo de mascotas.	65
Tabla 57. Actividad 27: Diseño de la interfaz del catálogo.....	66
Tabla 58. Actividad 28: Codificación del catálogo de mascotas.	66
Tabla 59. Actividad 29: Pruebas Unitarias.	67
Tabla 60. Actividad 30: Diseño de la interfaz de usuario.....	67
Tabla 61. Historia de usuario 9: Formulario de solicitud de adopción.....	67
Tabla 62. Actividad 31: Creación del modelo entidad relación del formulario de solicitud de adopción.....	68
Tabla 63. Actividad 32: Codificación del formulario.....	68
Tabla 64. Actividad 33: Pruebas Unitarias.	69
Tabla 65. Historia de usuario 10: Interfaz para gestionar solicitudes de adopciones.....	69
Tabla 66. Actividad 34: Diseño de la interfaz de usuario.....	70
Tabla 67. Actividad 35: Creación del modelo entidad relación para gestionar solicitudes de adopciones.....	70
Tabla 68. Actividad 36: Codificación del formulario.....	71
Tabla 69. Actividad 37: Pruebas Unitarias.	71
Tabla 70. Historia de usuario 11: Desarrollo de la notificación de resolución de solicitud de adopción.....	72
Tabla 71. Actividad 38: Diseño de la interfaz de usuario.....	72
Tabla 72. Actividad 39: Codificación de la notificación de resolución de solicitud de adopción.....	72

Tabla 73. Actividad 40: Pruebas Unitarias.	73
Tabla 74. Historia de usuario 12: Acta de adopción.	73
Tabla 75. Actividad 41: Diseño del acta de adopción.	74
Tabla 76. Actividad 42: Codificación del acta de adopción.	74
Tabla 77. Actividad 43: Pruebas Unitarias.	74
Tabla 78. Historia de usuario 13: Interfaz de reportes.	75
Tabla 79. Actividad 44: Diseño de la interfaz de usuario.	76
Tabla 80. Actividad 45: Codificación de la interfaz de reportes.	76
Tabla 81. Actividad 46: Pruebas Unitarias.	76
Tabla 82. Historia de usuario 14: Formulario para crear actividades de voluntariado.	77
Tabla 83. Actividad 47: Diseño de la interfaz de usuario.	78
Tabla 84. Actividad 48: Creación del modelo entidad relación del formulario para crear actividades de voluntariado.	78
Tabla 85. Actividad 49: Codificación del formulario.	78
Tabla 86. Actividad 50: Pruebas Unitarias.	79
Tabla 87. Historia de usuario 15: Catálogo de actividades para voluntariado.	79
Tabla 88. Actividad 51: Diseño de la interfaz de usuario.	80
Tabla 89. Actividad 52: Codificación del catálogo de actividades para voluntariado.	80
Tabla 90. Actividad 53: Pruebas Unitarias.	80
Tabla 91. Historia de usuario 16: Automatización de recepción de donaciones de dinero.	81
Tabla 92. Actividad 54: Diseño de la interfaz de usuario.	82
Tabla 93. Actividad 55: Codificación de la recepción de donaciones de dinero con PayPal.	82
Tabla 94. Actividad 56: Pruebas Unitarias.	82
Tabla 95. Estimación de tiempo de historias de usuario.	83
Tabla 96. Plan de entregas del aplicativo web.	86
Tabla 97. Historias de usuario de Iteración N° 1.	87
Tabla 98. Historias de usuario de Iteración N° 2.	93
Tabla 99. Historias de usuario de Iteración N° 3.	99
Tabla 100. Historias de usuario de Iteración N° 4.	101
Tabla 101. Historias de usuario de Iteración N° 5.	103
Tabla 102. Modelo de tarjeta CRC.	106
Tabla 103. Tarjeta CRC- Usuario.	106
Tabla 104. Tarjeta CRC- Mascota.	106
Tabla 105. Tarjeta CRC- Voluntariado.	106
Tabla 106. Tarjeta CRC- Adopción.	106
Tabla 107. Tarjeta CRC- Solicitud.	107
Tabla 108. Tarjeta CRC- Actividad.	107
Tabla 109. Tarjeta CRC- Enfermedad.	107
Tabla 110. Modelo de prueba de aceptación.	128
Tabla 111. Prueba de aceptación 1: Creación de la estructura del aplicativo web. .	129
Tabla 112. Prueba de aceptación 2: Formulario de registro.	129
Tabla 113. Prueba de aceptación 3: Interfaz de autenticación y autorización.	130

Tabla 114. Prueba de aceptación 4: Interfaz para actualizar información del usuario.	130
Tabla 115. Prueba de aceptación 5: Interfaz para asignar perfiles de usuario.	131
Tabla 116. Prueba de aceptación 6: Interfaz para gestionar información de las mascotas.	131
Tabla 117. Prueba de aceptación 7: Formulario para registrar información médica de las mascotas.	132
Tabla 118. Prueba de aceptación 8: Catálogo de mascotas.	132
Tabla 119. Prueba de aceptación 9: Formulario de solicitud de adopción.	133
Tabla 120. Prueba de aceptación 10: Interfaz para gestionar solicitudes de adopciones.	133
Tabla 121. Prueba de aceptación 11: Desarrollo de la notificación de resolución de solicitud de adopción.	134
Tabla 122. Prueba de aceptación 12: Acta de adopción.	134
Tabla 123. Prueba de aceptación 13: Interfaz de reportes.	135
Tabla 124. Prueba de aceptación 14: Formulario para crear actividades de voluntariado.	135
Tabla 125. Prueba de aceptación 15: Catálogo de actividades para voluntariado. ..	136
Tabla 126. Prueba de aceptación 16: Automatización de recepción de donaciones de dinero.	136
Tabla 127. Escala de Likert.	138
Tabla 128. Cuestionario de utilidad percibida.	138
Tabla 129. Resultados del cuestionario de utilidad percibida.	139
Tabla 130. Cuestionario de utilidad percibida.	140
Tabla 131. Resultados del cuestionario de facilidad percibida.	140
Tabla 132. Análisis de tiempos estimados en los procesos.	143

ÍNDICE DE FIGURAS

Figura 1. Resultados pregunta 1.....	23
Figura 2. Resultados pregunta 2.....	24
Figura 3. Resultados pregunta 3.....	25
Figura 4. Resultados pregunta 4.....	26
Figura 5. Resultados pregunta 5.....	27
Figura 6. Resultados pregunta 6.....	28
Figura 7. Resultados pregunta 7.....	29
Figura 8. Resultados pregunta 8.....	30
Figura 9. Resultados pregunta 9.....	31
Figura 10. Resultados pregunta 10.....	32
Figura 11. Diagrama del proceso de rescate de mascotas.....	34
Figura 12. Diagrama del proceso de adopción de mascotas.	35
Figura 13. Diagrama del proceso de voluntariado.	36
Figura 14. Diagrama del proceso de donación de dinero.....	37
Figura 15. Arquitectura del aplicativo web.....	45
Figura 16. Diagrama de fases de la metodología Extreme Programming.....	46
Figura 17. Diseño preliminar de la interfaz para Registrarse.....	54
Figura 18. Diseño preliminar la interfaz de autenticación y autorización.	57
Figura 19. Diseño preliminar del formulario de usuarios.	59
Figura 20. Diseño preliminar de la interfaz para asignar perfiles de usuario.	61
Figura 21. Diseño preliminar de la interfaz para gestionar información de las mascotas.....	63
Figura 22. Diseño preliminar de la interfaz para crear un reconocimiento y enfermedad.....	65
Figura 23. Diseño preliminar del catálogo de mascotas.	67
Figura 24. Diseño preliminar de la interfaz del formulario de adopción.	69
Figura 25. Diseño preliminar de la interfaz para gestionar solicitudes de adopciones.	72
Figura 26. Diseño preliminar de la notificación de resolución de solicitud de adopción.....	73
Figura 27. Diseño preliminar del acta de adopción.....	75
Figura 28. Diseño preliminar de la interfaz de reportes.....	77
Figura 29. Diseño preliminar de la interfaz del formulario para agregar una actividad.	79
Figura 30. Diseño preliminar del catálogo de actividades para voluntariado.	81
Figura 31. Diseño preliminar de la interfaz para realizar donaciones.....	83
Figura 32. Diagrama entidad-relación rescate de mascotas.....	88
Figura 33. Diagrama entidad-relación adopciones.....	89
Figura 34. Diagrama entidad-relación voluntariado.	90
Figura 35. Formulario para el registro de usuarios.	91
Figura 36. Ventana modal para el mensaje de confirmación.	91
Figura 37. Login.....	92
Figura 38. Página principal.	92

Figura 39. Tabla de los usuarios existentes.....	93
Figura 40. Formulario para editar información de los usuarios.....	93
Figura 41. Ventana modal para asignar un rol de usuario.....	94
Figura 42. Tabla de las mascotas existentes.....	95
Figura 43. Formulario para ingresar datos de una mascota.....	95
Figura 44. Sección para agregar fotos de las mascotas.....	96
Figura 45. Formulario para editar información de las mascotas.....	96
Figura 46. Formulario para agregar información del Reconocimiento de la mascota.	97
Figura 47. Formulario para agregar vacunas.....	97
Figura 48. Formulario para agregar reconocimientos de las mascotas.....	98
Figura 49. Formulario para agregar enfermedad o alergia.....	98
Figura 50. Catálogo de mascotas.....	99
Figura 51. Formulario de solicitud de adopción.....	100
Figura 52. Formato PDF del formulario de solicitud de adopción.....	100
Figura 53. Tabla de solicitudes de adopción.....	101
Figura 54. Mensaje de resolución de solicitud de adopción.....	101
Figura 55. Acta de adopción en formato PDF.....	102
Figura 56. Acta de adopción en formato PDF.....	103
Figura 57. Tabla de actividades disponibles para voluntariado.....	104
Figura 58. Formulario para agregar una nueva actividad.....	104
Figura 59. Catálogo de actividades.....	105
Figura 60. Cuenta empresarial de PayPal asociada a el aplicativo web.....	105
Figura 61. Cadena de conexión a la base de datos.....	108
Figura 62. Entidades que componen el aplicativo web.....	108
Figura 63. Capa de acceso a los datos del aplicativo web.....	109
Figura 64. Capa de lógica de negocios del aplicativo web.....	109
Figura 65. Clase controllers del aplicativo web.....	110
Figura 66. Archivo login.component.ts.....	111
Figura 67. Archivo login.service.ts.....	111
Figura 68. Archivo registrar.component.ts.....	112
Figura 69. Archivo registrar.service.ts.....	112
Figura 70. Archivo actualizarUsuario.component.ts.....	114
Figura 71. Archivo tipoPermiso.component.ts.....	115
Figura 72. Archivo usuario.service.ts.....	115
Figura 73. Archivo crearMascota.component.ts.....	116
Figura 74. Archivo crearEnfermedad.component.ts.....	117
Figura 75. Archivo crearVacuna.component.ts.....	118
Figura 76. Archivo crearReconocimiento.component.ts.....	118
Figura 77. Archivo mascota.service.ts.....	119
Figura 78. Archivo catálogo.component.ts.....	120
Figura 79. Archivo catálogo.service.ts.....	120
Figura 80. Archivo formulario-adopcion.component.ts.....	121
Figura 81. Archivo formulario.service.ts.....	122
Figura 82. Archivo formulario-adopcion.component.ts.....	123
Figura 83. Archivo datalleSolicitud.service.ts.....	124

Figura 84. Archivo dashboard.component.ts.	125
Figura 85. Archivo dashboard.service.ts.	126
Figura 86. Archivo crearActividad.component.ts.	127
Figura 87. Archivo voluntariado.service.ts.	128
Figura 88. Resultados del cuestionario de utilidad percibida.	139
Figura 89. Resultados del cuestionario de facilidad percibida.	141

ÍNDICE DE ANEXOS

Anexo A. Formato de encuesta aplicada a personas que está en proceso de adopción de mascotas.	150
Anexo B Manual de usuario.....	155

RESUMEN EJECUTIVO

El rescate y cuidado de animales se ha convertido en uno de los sectores más destacados en cuanto a obra social y generación de interés público. Es por esta razón que las fundaciones comprometidas con esta noble causa promueven el cuidado responsable de los animales, colocando un énfasis significativo en la importancia de asegurar su bienestar y fomentar prácticas responsables en su atención y adopción.

En este contexto, sin embargo, una de las limitaciones que enfrentan estas fundaciones es el acceso limitado a herramientas tecnológicas que les permitan gestionar de manera eficaz la información relacionada con sus actividades. La falta de recursos tecnológicos puede dificultar la coordinación de rescates, adopciones y voluntariado, así como el seguimiento detallado de la salud y necesidades individuales de los animales bajo su cuidado.

El presente proyecto de investigación se planteó con la finalidad de desarrollar un aplicativo web diseñado para ayudar a la Fundación de Rescate Animal Paquito. Este aplicativo web permite la gestión de los procesos involucrados en las actividades de rescate, adopción, voluntariado y recepción de donaciones monetarias.

En la creación de la aplicación, se optó por aplicar la metodología XP con el objetivo de adaptarse de manera flexible a los cambios que pudieran surgir en el aplicativo. Para el desarrollo del Frontend y la interfaz de usuario, se empleó la tecnología Angular, mientras que el backend se construyó con el framework .NET Core, incluyendo una API Rest que contiene los métodos utilizados por el aplicativo. La base de datos MySQL fue la elección para almacenar los datos.

Palabras clave: Aplicativo web, gestión de actividades, Angular, APS.NET Core.

ABSTRACT

The rescue and care of animals have become one of the most prominent sectors in terms of social work and public interest generation. For this reason, foundations committed to this noble cause promote responsible care of animals, placing significant emphasis on ensuring their well-being and fostering responsible practices in their care and adoption.

In this context, however, one of the limitations faced by these foundations is the limited access to technological tools that would allow them to effectively manage information related to their activities. The lack of technological resources can hinder the coordination of rescues, adoptions, and volunteer work, as well as the detailed monitoring of the health and individual needs of the animals under their care.

The present research project was conceived with the aim of developing a web application designed to assist the Paquito Animal Rescue Foundation. This web application enables the management of processes involved in rescue activities, adoption, volunteering, and the reception of monetary donations.

In the development of the application, the decision was made to apply the XP methodology with the goal of adapting flexibly to changes that may arise in the application. For the development of the Frontend and user interface, Angular technology was employed, while the backend was built with the .NET Core framework, including a Rest API that contains the methods used by the application. MySQL was the chosen database to store the data.

Keywords: Web application, activity management, Angular, APS.NET Core.

CAPÍTULO I. MARCO TEÓRICO

1.1 Tema de investigación

APLICATIVO WEB UTILIZANDO FRAMEWORK OPEN SOURCE PARA LA GESTIÓN DE ACTIVIDADES DE LA FUNDACIÓN DE RESCATE ANIMAL PAQUITO DE LA CIUDAD DE AMBATO.

1.1.1 Planteamiento del problema

En el Ecuador según el Sistema Unificado de Información de las Organizaciones Sociales (SUIOS) se encuentran registradas 12,743 fundaciones tanto públicas como privadas [1]. De acuerdo a los datos de la Secretaría Técnica de Cooperación Internacional se calcula que existen 208 organizaciones no gubernamentales internacionales en el país [2]. Gran parte de las fundaciones internacionales reciben recursos económicos y tecnológicos desde sus países de origen. En contraposición gran parte de las fundaciones ecuatorianas no reciben ayuda económica interna ni externa. Por tal motivo, las fundaciones se ven en la necesidad de planificar nuevas actividades para recaudar fondos e invertir en tecnología, personal e infraestructura para de esta manera llevar a cabo los proyectos planteados que van dirigidos a los beneficiarios [3].

En el cantón Ambato de la provincia de Tungurahua existen 214 fundaciones registradas según el Ministerio de Inclusión Económica y Social. Para estas fundaciones el uso de los recursos tecnológicos puede ser un desafío y resulta un objetivo de difícil acceso, ya que no todas cuentan con los recursos económicos para adquirirlos. Como resultado, estas fundaciones pueden enfrentar ciertas limitaciones debido a la escasez de tecnología adecuada, lo que puede afectar el desarrollo de sus actividades y disminuir la rapidez y efectividad en la ejecución de los proyectos que llevan a cabo. Sin embargo, cada vez más fundaciones buscan alternativas y medios de financiamiento con el propósito de adoptar soluciones tecnológicas para automatizar sus procesos de gestión [2].

La fundación de rescate animal “Paquito” de la ciudad de Ambato desarrolla continuamente proyectos en beneficio de animales abandonados o extraviados, los

cuales buscan alimento y refugio en las calles de la ciudad. Sin embargo, con el incremento de cifras relacionadas al patrocinio y auspicio en favor de este grupo de animales, la fundación se enfrenta a un importante desafío debido al escaso financiamiento tanto interno y externo, lo que limita sus recursos para la ejecución de proyectos.

Actualmente, la fundación no cuenta con una herramienta de software que automatice las actividades de rescate, el seguimiento y cuidado de los animales rescatados, la búsqueda de hogares definitivos para los animales, la recolección de donaciones y la coordinación de voluntarios. Por tal motivo, se cree necesario desarrollar un aplicativo web para la gestión de actividades que la fundación lleva a cabo.

1.2 Antecedentes investigativos

Por medio de la investigación realizada en los repositorios, sitios académicos, artículos científicos, entre otras fuentes se pueden citar los siguientes aportes como antecedentes al presente proyecto.

En el proyecto de investigación presentado por Chicaiza Villegas [4], supo concluir que: “La combinación de las tecnologías Open Source ASP.NET Core y Angular ha sido clave en el desarrollo de la aplicación, ya que ha permitido trabajar de manera independiente el backend y frontend facilitando el mantenimiento y escalabilidad de la aplicación web. Además, la facilidad con la que se integran librerías como: AutoMapper, RestSharp y las que comprenden Microsoft.EntityFrameworkCore de lado del servidor y librerías del lado del cliente como: Angular Material y MapBox, ha dado lugar a la reducción en el tiempo de desarrollo de la aplicación. El autor también menciona que para el desarrollo de la aplicación ha utilizado la metodología Extreme Programming (XP) por su eficiencia, flexibilidad y control en el cumplimiento de procesos mediante una buena gestión de actividades.

En la investigación de Manzano Rojas [5], se menciona que: “Para la creación del sistema web, se empleó el framework Angular en su versión 15.0.0. La elección de Angular se basó en su integración del lenguaje TypeScript, que aporta diversas funcionalidades para mejorar la versatilidad y robustez del sistema. Este framework

también ofrece una estructura modular y consistente al código, permitiendo trabajar con módulos que facilitan la organización de la aplicación. La capacidad de separar estos módulos en componentes reutilizables agiliza significativamente el proceso de desarrollo. Además, con la utilización de librerías de Angular se facilitó la gestión de las Api-REST. El autor también menciona que la implementación del sistema favoreció a la Empresa con la reducción del tiempo empleado en el proceso de obtención de Insumos Primarios debido a que se automatizó los procesos que se realizaban de forma manual”.

Según Hurtado Samaniego [6], determinó que: “El Framework .Net Core permitió agilizar de manera sustancial el proceso de desarrollo, permitiendo crear código limpio, óptimo y reutilizable, lo cual garantiza tener un impacto mínimo en tiempos de desarrollo cuando se presenten cambios o se corrijan errores y fallos. La autora menciona, que para el desarrollo de la aplicación web propuesta se utilizó el Framework .Net Core versión 5, utilizando Clean Architecture para la abstracción de la lógica de la aplicación. En resumen, se ha logrado desarrollar una aplicación interactiva que cumple con los requisitos identificados durante la fase de recolección y análisis de información, resultando en un producto adaptable y fácilmente utilizable según las necesidades del usuario”.

En el proyecto de investigación de Toaza Cepeda [7], menciona que: “La aplicación web basada en tecnología Open Source fue desarrollada para dar solución a la problemática planteada en su proyecto, y como resultado, el software cumplió con el objetivo propuesto desde el inicio del proyecto, lo que permitió que los procesos sean automatizados facilitando al usuario la obtención de información de manera más eficiente. El autor también menciona que la elección de la tecnología Open Source se fundamenta en la ausencia de costos asociados a licencias. Esta característica posibilita que cualquier persona pueda aprovechar la herramienta para gestionar su empresa o negocio de manera eficiente”.

En su trabajo de investigación Triviño Zambrano [8], concluyó que: “La aplicación desarrollada a través del Framework Angular ofrece la funcionalidad planeada, con interfaces que fueron calificadas como intuitivas, atractivas y fáciles de usar. De tal manera, da respuesta a la necesidad de la Fundación Rescate Animal de gestionar la

información relacionada con los casos de animales rescatados, tratados y listos para ser adoptados por una familia que les brinde un hogar y un ambiente adecuado para su bienestar”.

Según Pajuelo Carlevarino [9], menciona que: “La gestión de la información surge como el manejo de inteligencia corporativa de una organización, que permite reaccionar ante cambios de su entorno apoyándose en el uso de la información y de los recursos de información disponibles. Los elementos involucrados en la gestión de la información pueden ser resumidos en tres grupos: los que competen a la información como fuente/recurso (procesos productivos al interior de las organizaciones), los relacionados con los usuarios y servicios de información y los que conforman el canal de comunicación entre usuarios y la fuente”.

1.3 Fundamentación teórica

A continuación, se aborda la fundamentación teórica sobre la cual se construye la investigación, brindando contexto y fundamentos sólidos para el desarrollo del proyecto.

Ingeniería de Software

La ingeniería de software es un campo donde los profesionales emplean ciencias exactas como matemáticas, física y química, junto con conocimientos tecnológicos, para abordar problemas y desarrollar productos. Además, engloba diversas disciplinas interrelacionadas, como la arquitectura de software, la aplicación de metodologías, el uso de herramientas, técnicas de desarrollo, documentación y otros aspectos afines [10].

Para asegurar que un software cumpla con altos estándares de calidad, es esencial que tenga características como escalabilidad, facilidad de uso, seguridad y flexibilidad, entre otras. Estas cualidades garantizan que el software sea confiable, adaptable y satisfaga las necesidades de los usuarios [10].

Desarrollo Web

El desarrollo de aplicaciones web se conforma de la capa de presentación (FrontEnd) y una capa de acceso a datos (BackEnd). En el ámbito del desarrollo web, el FrontEnd está conformado por todas aquellas tecnologías que corren del lado del cliente, es decir, todas aquellas tecnologías que corren del lado del navegador web, generalizándose más que nada en tres lenguajes, Html (HyperText Markup Language), CSS (Cascading Style Sheets) y JavaScript. La persona encargada del FrontEnd, trabaja con estas tres tecnologías, aunque esto no significa que desconozca como trabaja el BackEnd (lado servidor), ya que es necesario para que pueda consumir datos y pueda estructurar correctamente un maquetado en HTML y CSS para su mejor comodidad [11].

FrontEnd

El término "frontend" o "desarrollo del lado del cliente" hace referencia a la práctica de crear HTML, CSS y JavaScript, siendo estos tres componentes los responsables de dar forma y estructura a la interfaz visual de un sitio web o aplicación. El frontend desempeña un papel esencial en la creación de la interfaz de un sitio web, abarcando desde la estructura hasta los estilos, como colores, texturas, tipografías y secciones, entre otros aspectos. Su importancia radica en proporcionar una experiencia positiva al usuario al interactuar con el sitio o aplicación, influyendo directamente en la percepción y usabilidad del producto [12].

Backend

Se encarga de la manipulación de los datos, un Backend no sirve de mucho si no existe un FrontEnd de por medio, el desarrollador Backend debe de conocer de bases de datos, frameworks y aspectos de seguridad. Él debe encargarse de que la información que llega desde el FrontEnd, sea almacenada a una base de datos. Así mismo se encarga de crear API's(Application Programming Interface) para que sus datos puedan consumirse de manera cómoda y pueda mejorar la experiencia del usuario [12].

Aplicativo Web

Un Aplicativo Web (web-based application) es un tipo especial de aplicación cliente/servidor, donde tanto el cliente (el navegador, explorador o visualizador) como el servidor (el servidor web) y el protocolo mediante el que se comunican HTTP (Hypertext Transfer Protocol) están estandarizados y no han de ser creados por el programador de aplicaciones [13].

Framework

Un framework es un esquema o marco de trabajo que proporciona una estructura base para la creación de un proyecto con objetivos específicos. Se podría describir como una plantilla que sirve como punto de partida para la organización y desarrollo de software. Por lo general, los programadores utilizan frameworks para agilizar el trabajo, fomentar la colaboración, minimizar errores y lograr resultados de mayor calidad [14].

Framework Open Source

Un framework Open Source es un conjunto de herramientas, bibliotecas y componentes de software predefinidos y accesibles públicamente, cuyo código fuente está disponible para el público en general. Estos frameworks ofrecen una estructura básica para el desarrollo de aplicaciones o proyectos de software, permitiendo a los desarrolladores aprovechar soluciones ya existentes, acelerar el proceso de desarrollo y fomentar la colaboración en la comunidad de código abierto [15].

Angular

Angular es un marco de JavaScript de código abierto desarrollado en TypeScript, con el respaldo de Google y orientado a la creación de aplicaciones de una sola página (SPA). Al destacar notables ventajas durante su desarrollo, proporciona una estructura estandarizada que facilita el trabajo de los desarrolladores. Asimismo, permite la construcción eficiente de aplicaciones a gran escala [16].

Vue

Vue es un marco de JavaScript para crear interfaces de usuario. Se basa en HTML, CSS y JavaScript estándar y proporciona un modelo de programación declarativo y

basado en componentes que le ayuda a desarrollar interfaces de usuario de manera eficiente, ya sean simples o complejas. Desarrollado por Evan You, Vue. se centra en la capa de vista de una aplicación, proporcionando un enfoque reactivo y fácil de integrar con otras bibliotecas o proyectos [17].

Vue es conocido por su flexibilidad, simplicidad y curva de aprendizaje gradual, lo que lo hace adecuado tanto para pequeños proyectos como para aplicaciones empresariales más complejas. Además, su creciente comunidad y documentación extensa lo convierten en una opción popular para el desarrollo web [17].

React

React es una biblioteca de JavaScript de código abierto desarrollada y mantenida por Facebook. Aunque comúnmente se le llama "framework", React se enfoca específicamente en la creación de interfaces de usuario (UI) interactivas para aplicaciones de una sola página (SPA). React utiliza un paradigma de programación declarativa y se centra en la creación de componentes reutilizables, lo que facilita la construcción de interfaces de usuario complejas [18].

Net Core

ASP.NET Core, la versión de código abierto de ASP.NET, es compatible con macOS, Linux y Windows. Su lanzamiento inicial en 2016 marcó un rediseño respecto a las versiones anteriores, que estaban limitadas a Windows. Este marco está diseñado para permitir una rápida evolución de componentes en tiempo de ejecución, API, compiladores y lenguajes, al mismo tiempo que proporciona una plataforma estable y compatible para mantener la continuidad de las aplicaciones en ejecución [19].

Es posible tener varias versiones de .NET Core ejecutándose simultáneamente en el mismo servidor. Esto implica que una aplicación puede adoptar la versión más reciente, mientras otras aplicaciones siguen funcionando con la versión en la que fueron probadas. .NET Core ofrece diversas opciones de soporte técnico de ciclo de vida para adaptarse a las necesidades de la aplicación. Los usuarios pueden optar por una versión con soporte técnico a largo plazo o ejecutarla con la versión más reciente, siempre que se comprometan a realizar actualizaciones con mayor frecuencia [19].

API REST

Es una interfaz de programación de aplicaciones web que se ajusta a los límites de la arquitectura Representational State Transfer (REST) el cual es un estilo arquitectónico que define un conjunto de directrices para la construcción de servicios web mediante la comunicación cliente - servidor. Las solicitudes se realizan con el uso del protocolo HTTP, los mensajes enviados o recibidos pueden estar en formato XML o JSON [20].

Bases de Datos

Una base de datos, en términos generales, es un conjunto organizado de información. De manera más específica, se trata de un sistema electrónico que facilita el acceso, manipulación y actualización eficientes de datos. En esencia, las organizaciones utilizan bases de datos como un medio para almacenar, administrar y recuperar información de manera efectiva. Estos conjuntos de datos contemporáneos son gestionados a través de sistemas especializados conocidos como sistemas de gestión de bases de datos (DBMS) [21].

MySQL

MySQL (My Structured Query Language) es un sistema gestor de base de datos de código libre. Se caracteriza por su adaptación en varios entornos de desarrollo y es compatible con los lenguajes de programación más utilizados como PHP, Perl, Java, entre otros. MySQL recibe continuamente actualizaciones gracias a su libre distribución, lo que la convierte en uno de los motores de bases de datos más utilizados en el desarrollo web [22].

Arquitectura cliente/servidor (C/S)

La arquitectura clásica de cliente/servidor, también denominada arquitectura de dos capas, implica la necesidad de una interfaz de usuario instalada y en ejecución en un equipo o estación de trabajo. Esta interfaz de usuario se encarga de enviar solicitudes al servidor para llevar a cabo operaciones más complejas [23].

Gestión de Actividades

La Gestión de Actividades incluye la identificación de las actividades necesarias para lograr los objetivos de la organización, la asignación de los recursos necesarios para llevar a cabo esas actividades, la definición de los plazos y el seguimiento del progreso para asegurarse de que se estén cumpliendo [24]. Por el tipo de actividades las podemos clasificar en:

- Gestión de actividades operativas: Aquellas relacionadas con las tareas diarias y rutinarias necesarias para el funcionamiento continuo de la organización [24].
- Gestión de actividades estratégicas: Aquellas relacionadas con la planificación a largo plazo, la toma de decisiones y la implementación de estrategias para alcanzar los objetivos organizacionales [24].
- Gestión de actividades de apoyo: Aquellas que brindan soporte a las operaciones principales de la organización, como recursos humanos, finanzas, tecnología de la información, entre otros [24].

Metodologías de desarrollo de software

Una metodología de desarrollo de software es un marco de trabajo que se usa para estructurar, planificar y controlar el proceso de desarrollo de sistemas de información. Una gran variedad de estos marcos de trabajo ha evolucionado durante los años, cada uno con sus propias fortalezas y debilidades. Una metodología de desarrollo de sistemas no tiene que ser necesariamente adecuada para usarla en todos los proyectos. Cada una de las metodologías disponibles es más adecuada para tipos específicos de proyectos, basados en consideraciones técnicas, organizacionales, de proyecto y de equipo [25].

Metodologías tradicionales

Se distingue por su enfoque en la documentación exhaustiva de todos los procesos involucrados en el desarrollo de software, desde la recopilación de requisitos hasta la planificación y el control del proyecto. Impone un flujo de trabajo meticuloso que incluye una definición detallada de roles, actividades, herramientas y recursos necesarios para garantizar un producto final de calidad [26].

Algunas características distintivas de esta metodología incluyen la participación de grupos grandes en el proyecto, una falta de escalabilidad, ciclos de entrega cortos, la participación ocasional del cliente a través de reuniones, resistencia del equipo a los cambios y una diversidad de roles dentro del proyecto [26].

Metodologías Ágiles

La metodología ágil es un enfoque de gestión de proyectos que implica la división del proyecto en fases, destacando la colaboración y la mejora continua. Utilizada tanto en el desarrollo de software como en proyectos de alto rendimiento, la metodología ágil se centra en establecer rápidamente un equipo eficiente y flexible para planificar el flujo de trabajo [27].

Extreme Programming (XP)

Extreme Programming (XP) es un marco ágil de desarrollo de software diseñado para producir software de alta calidad y mejorar la eficiencia del equipo de desarrollo. Se centra en la implementación de prácticas de ingeniería que promueven la creación efectiva de software. La metodología fue formulada por Kent Beck, autor del influyente libro "Extreme Programming Explained: Embrace Change", publicado en 1999 [28].

Kanban

Kanban es una técnica que se centra en el control eficiente del trabajo, priorizando la finalización de las actividades en un flujo de trabajo específico. Se basa en un tablero visual donde se representan las diferentes etapas del proceso, lo que facilita la visualización, limitación y medición del flujo de trabajo para un mejor control del proyecto [29].

Una de las características distintivas de Kanban es su enfoque en la calidad de los procesos, destacando las actividades fundamentales para mejorar el rendimiento y alcanzar los objetivos establecidos. Esta metodología ofrece varios beneficios, incluyendo una mayor transparencia en el progreso del trabajo, una capacidad de

respuesta mejorada a los cambios y una optimización continua de los procesos para una mayor eficiencia [29].

Scrum

Esta metodología ofrece una forma eficaz de gestionar el proceso de desarrollo de software mediante un enfoque incremental e iterativo. Su principal característica radica en la optimización de los tiempos de desarrollo y la mejora de la comunicación entre los miembros del equipo, lo que resulta en una mayor productividad y una reducción de costos derivada de la mejora en la calidad del producto gracias a la comunicación constante [30].

Scrum se compone de tres roles principales: el Scrum Master, quien actúa como facilitador y líder del equipo, el Product Owner, encargado de establecer las prioridades del proyecto y garantizar su alineación con los objetivos del cliente, y el Scrum Team, el equipo de desarrollo responsable de ejecutar las tareas necesarias para cumplir con los requisitos del proyecto. Además, el Product Owner evalúa constantemente la metodología y su adaptabilidad para abordar cualquier posible contratiempo o desafío que pueda surgir durante el desarrollo del proyecto [30].

1.4 Objetivos

Para el presente proyecto se han establecido los siguientes objetivos:

1.4.1 Objetivo general

Implantar un aplicativo web utilizando framework open source para la gestión de actividades de la fundación de rescate animal “Paquito”.

1.4.2 Objetivos específicos

- Analizar los procesos que se llevan a cabo en la fundación en cuanto a la gestión de actividades de rescate, adopción y recolección de donaciones.
- Definir las tecnologías open source adecuadas con las que se desarrollará el aplicativo.

- Desarrollar un aplicativo web utilizando framework open source para la gestión de actividades en la fundación de rescate animal “Paquito”.

CAPÍTULO II. METODOLOGÍA

2.1 Materiales

Para el desarrollo del presente proyecto de investigación se aplicó una entrevista compuesta de 11 preguntas dirigida al presidente y vicepresidente de la fundación. Además, se aplicó una encuesta compuesta por 10 preguntas. Estas preguntas incluyeron opciones cerradas y el empleo de la escala de Likert. Los cuestionarios fueron dirigidos a personas que están en proceso de adopción de mascotas en la fundación “Paquito”.

Entrevista dirigida al presidente y vicepresidente de la fundación

Pregunta 1: ¿Cómo gestionan actualmente los datos de las mascotas rescatadas?

Pregunta 2: ¿Cuál es el proceso que siguen para dar en adopción a una mascota?

Pregunta 3: ¿Cómo se le comunica la resolución de una solicitud de adopción al adoptante?

Pregunta 4: ¿De qué manera realizan actualmente el proceso de recaudación de donaciones de dinero en la fundación?

Pregunta 5: ¿Por qué medio dan a conocer las actividades que realizan dentro de la fundación?

Pregunta 6: ¿Creen que la manera en que se gestionan las actividades actualmente es adecuada?

Pregunta 7: ¿Cuentan actualmente con recursos tecnológicos en la fundación?

Pregunta 8: ¿Están utilizando actualmente algún programa informático para gestionar las actividades?

Pregunta 9: ¿Cuáles son los principales problemas o desafíos que enfrentan como fundación actualmente en la gestión de las actividades que realizan?

Pregunta 10: ¿Qué beneficios creen que traería la implementación de un aplicativo web para gestionar las actividades que llevan a cabo?

Pregunta 11: ¿Cuál es el número promedio de adopciones de mascotas que la fundación realiza anual y mensualmente?

Encuesta dirigida a personas que están en proceso de adopción de mascotas

Cómo usuario externo, ¿tiene usted conocimiento del proceso a seguir para efectuar una adopción de mascotas en la fundación de rescate animal “Paquito”?

- Si
- No

En base a su experiencia, ¿cómo considera el proceso de adopción de mascotas con el que cuenta la fundación de rescate animal “Paquito”?

- Pésima
- Mala
- Regular
- Buena
- Excelente

¿En qué medida considera usted el nivel de eficiencia del proceso de solicitud de adopción de mascotas?

- Muy Ineficiente
- Ineficiente
- Regular
- Eficiente

Muy Eficiente

En su opinión, ¿considera usted que la fundación de rescate animal “Paquito” proporciona información necesaria y detallada de las mascotas que están en adopción?

Totalmente en desacuerdo

En desacuerdo

Neutral

De acuerdo

Totalmente de acuerdo

¿Por favor, indique en qué medida considera importante que la fundación de rescate animal “Paquito” realice un adecuado seguimiento de las mascotas que han sido dadas en adopción?

Nada importante

De poca importancia

Moderadamente importante

Importante

Muy importante

¿Defina en base a la siguiente escala el nivel de recursos económicos y el espacio físico idóneo con el que usted cuenta para proporcionar un hogar adecuado a una mascota adoptada?

Muy Insuficiente

Insuficiente

Regular

Suficiente

Abundante

¿Conoce usted de aplicaciones o sitios web donde se pueda realizar adopciones de mascotas?

- Si
- No

¿En base a sus aptitudes adquiridas, defina el nivel de dominio con el que usted cuenta en base al uso de la tecnología?

- Muy Difícil
- Difícil
- Regular
- Fácil
- Muy Fácil

¿A través de qué medio considera que la difusión de adopciones de mascotas puede alcanzar a más personas interesadas en realizar una adopción?

- Redes Sociales
- Aplicativos Web
- Radio
- Televisión
- Carteles exteriores

En general, ¿cómo calificaría su experiencia al adoptar una mascota en la fundación “Paquito”?

- Totalmente Insatisfecho
- Insatisfecho
- Algo Satisfecho
- Satisfecho
- Totalmente Satisfecho

2.2 Métodos

La investigación tiene un enfoque cuantitativo, ya que se analizarán de forma estadística los datos obtenidos mediante la encuesta aplicada a la muestra de la población objeto de estudio.

2.2.1 Modalidad de la investigación

Las modalidades de investigación del presente proyecto son: bibliográfica-documental.

Investigación bibliográfica - documental

Se llevó a cabo una investigación bibliográfica-documental, empleando fuentes como libros, documentos, artículos y revistas para construir la contextualización y el marco teórico. Además, se analizaron los resultados de la investigación con el fin de proponer una solución al problema planteado.

2.2.2 Población y muestra

Población: Para la presente investigación la población identificada fueron el presidente, vicepresidente y usuarios que están en proceso de adopción de mascotas de la fundación de rescate animal Paquito de la ciudad de Ambato. El número de usuarios que se encuentra actualmente en proceso de adopción de mascotas es 20. Por lo tanto, se ha elegido a este grupo de usuarios para la aplicación de los cuestionarios.

Tabla 1. Población de Estudio.

Tipo de Población	Número	Porcentaje
Presidente	1	4,55 %
Vicepresidente	1	4,55 %
Usuarios	20	90,90 %
Total	22	100 %

Muestra: Dado que la población a estudiar en la presente investigación no supera los cien elementos no se trabajó con la muestra representativa y se trabajó con la totalidad de la población.

2.2.3 Recolección de información

Se recolectó información a través de una entrevista aplicada al presidente y vicepresidente de la fundación. Además, se aplicó una encuesta dirigida a personas que están en proceso de adopción de mascotas en la fundación, esto permitirá obtener de manera específica la información que necesitamos.

Validación del instrumento

Kuder-Richardson en la encuesta dirigida a personas que están en proceso de adopción de mascotas

Es una medida estadística utilizada para evaluar la confiabilidad de una prueba o cuestionario con variables binarias (es decir, respuestas correctas o incorrectas) que mide la consistencia interna de las respuestas.

Para verificar la validez y confiabilidad de los datos de la encuesta dirigida a personas que están en proceso de adopción de mascotas se aplicó el coeficiente de Kuder-Richardson, aplicado a 2 de las 10 preguntas, las cuales se identifican como dicotómicas es decir que ofrecen dos opciones de respuesta tanto si como no.

Para el cálculo de Kuder-Richardson se aplicó la siguiente fórmula:

$$r_{kr20} = \left(\frac{k}{k-1} \right) \left(1 - \frac{\sum pq}{\sigma^2} \right)$$

Como herramienta de cálculo para evaluar los resultados se utilizó Microsoft Excel, empleando una tabla que incorporó las respuestas obtenidas previamente.

Tabla 2. Coeficiente Kuder-Richardson en la encuesta para personas que están en proceso de adopción de mascotas.

Simbología	Valor
k (número de ítems)	2
p (porcentaje de personas que responde Si cada ítem)	0,65 y 0,50
q (porcentaje de personas que responden No cada ítem)	0,35 y 0,50
σ^2 (varianza total del instrumento)	0,77
KR-20	0,75

Tabla 3. Interpretación de la escala de consistencia de Kuder-Richardson.

KR-20	Interpretación
0,9 - 1	Excelente
0,8 - 0,9	Buena
0,7 - 0,8	Aceptable
0,6 - 0,7	Débil
0,5 - 0,6	Pobre
< 0,5	Inaceptable

El coeficiente Kuder-Richardson da un valor de 0.75, indicando que las respuestas de las personas se sitúan en un nivel aceptable de confiabilidad.

Alfa de Cronbach en la encuesta dirigida a personas que están en proceso de adopción de mascotas

El coeficiente alfa de Cronbach es una medida estadística empleada para evaluar la confiabilidad interna o consistencia de un conjunto de ítems en un cuestionario o prueba. En el proceso de validación de las preguntas formuladas en una escala de Likert, se utiliza este coeficiente para asegurarse de que las preguntas sean consistentes y fiables.

Para el cálculo de alfa de Cronbach se aplicó la siguiente fórmula:

$$a = \frac{K}{K - 1} \left[1 - \frac{\sum v_i}{vt} \right]$$

Como herramienta de cálculo para evaluar los resultados se utilizó Microsoft Excel, empleando una tabla que incorporó las respuestas obtenidas previamente.

Tabla 4. Confiabilidad Alfa de Cronbach en la encuesta para personas que están en proceso de adopción de mascotas.

Simbología	Valor
α (coeficiente de confiabilidad)	0,73
k (número de ítems)	8
$\sum v_i$ (sumatoria de varianzas de los ítems)	5,58
vt (varianza total)	15,46

Tabla 5. Confiabilidad Alfa de Cronbach en la encuesta para personas que están en proceso de adopción de mascotas.

Alfa de Cronbach	Interpretación
0,9 - 1	Excelente
0,8 - 0,9	Buena
0,7 - 0,8	Aceptable
0,6 - 0,7	Débil
0,5 - 0,6	Pobre
< 0,5	Inaceptable

Como resultado el Alfa de Cronbach da un valor de 0,73 indicando que las respuestas de las personas se sitúan en un nivel aceptable de confiabilidad.

Resultados de la entrevista

Los resultados de la entrevista realizada al presidente y vicepresidente de la fundación fueron los siguientes:

Pregunta 1: ¿Cómo gestionan actualmente los datos de las mascotas rescatadas?

Respuesta: Actualmente, en la fundación, se gestionan los datos de las mascotas rescatadas de forma manual y mediante el uso Excel. Cuando rescatamos un animal, creamos una ficha de registro.

Esta ficha se almacena en formato físico y también se registra en una hoja de cálculo de Excel dedicada a la gestión de adopciones y rescates. Utilizamos Excel para llevar un seguimiento de las adopciones, incluyendo detalles sobre los adoptantes, fechas de adopción y la información de contacto necesaria.

Pregunta 2: ¿Cuál es el proceso que siguen para dar en adopción a una mascota?

Respuesta: El proceso de adopción de un animalito involucra una serie de pasos importantes. En primer lugar, se lleva a cabo una conversación para conocer cómo se llevará a cabo el cuidado del animal y para llegar a un compromiso mutuo. Este compromiso se formaliza mediante la presentación de una solicitud de adopción, la cual se revisa y se da la aprobación o negación de la misma.

Pregunta 3: ¿Cómo se le comunica la resolución de una solicitud de adopción al adoptante?

Respuesta: Se le llama telefónicamente al número de contacto al adoptante y se le entrega la resolución personalmente en la fundación.

Pregunta 4: ¿De qué manera realizan actualmente el proceso de recaudación de donaciones de dinero en la fundación?

Respuesta: A través de depósitos en la cuenta bancaria de la Fundación, así como mediante contribuciones directas en las instalaciones de la fundación.

Pregunta 5: ¿Por qué medio dan a conocer las actividades que realizan dentro de la fundación?

Respuesta: A través de redes sociales, periódico, radio.

Pregunta 6: ¿Creen que la manera en que se gestionan las actividades actualmente es adecuada?

Respuesta: Sí, porque la gestionamos de la manera más ordenada posible. Aunque siempre estamos abiertos a nuevas ideas que nos puedan ayudar a mejorar.

Pregunta 7: ¿Cuentan actualmente con recursos tecnológicos en la fundación?

Respuesta: Actualmente contamos con una computadora con acceso a internet.

Pregunta 8: ¿Están utilizando actualmente algún programa informático para gestionar las actividades?

Respuesta: Si, utilizamos mayormente Microsoft Excel y Word.

Pregunta 9: ¿Cuáles son los principales problemas o desafíos que enfrentan como fundación actualmente en la gestión de las actividades que realizan?

Respuesta: La falta de recursos humanos altamente capacitados, la disponibilidad de recursos económicos sustanciales y la escasa tecnología que ayude a mantener la información mejor distribuida y segura.

Pregunta 10: ¿Qué beneficios creen que traería la implementación de un aplicativo web para gestionar las actividades que llevan a cabo?

Respuesta: Primeramente, nos ayudaría a ser más visibles para entidades extranjeras que ayuden a mantener nuestra labor, también nos ayudaría a integrar toda la información y actividades que realizamos en un solo sitio.

Pregunta 11: ¿Cuál es el número promedio de adopciones de mascotas que la fundación realiza anualmente y mensualmente?

Respuesta: El número promedio de adopciones de mascotas que nuestra fundación realiza anualmente es de aproximadamente 100 adopciones al año. Si lo desglosamos por mes tenemos un promedio de alrededor de 9 adopciones mensuales.

Conclusión de la entrevista: La mayoría de las actividades son realizadas de manera presencial con los adoptantes, y a las herramientas informáticas que utilizan actualmente que están diseñadas en Excel y Word solo se puede acceder desde la computadora que posee la fundación, lo que puede ocasionar problemas al subir o consultar la información.

La comunicación con el adoptante se lleva a cabo por teléfono, lo cual puede generar demoras en el proceso, ya que no siempre el adoptante dispone de tiempo para atender llamadas.

La fundación al recibir únicamente donaciones de dinero de forma presencial o a través de depósitos bancarios, se restringe el acceso a ayuda internacional, tanto de personas como de instituciones.

Resultados de la encuesta aplicada a personas que están en proceso de adopción de mascotas.

A continuación, para cada una de las preguntas de la encuesta se representará estadísticamente de manera gráfica y tabular las respuestas entregadas por las personas encuestadas.

Pregunta 1: Como usuario externo, ¿tiene usted conocimiento del proceso a seguir para efectuar una adopción de mascotas en la fundación de rescate animal “Paquito”.

Tabla 6. Resultados pregunta 1.

Indicador	Frecuencia	Porcentaje
Si	13	65%
No	7	35%
Total	20	100%

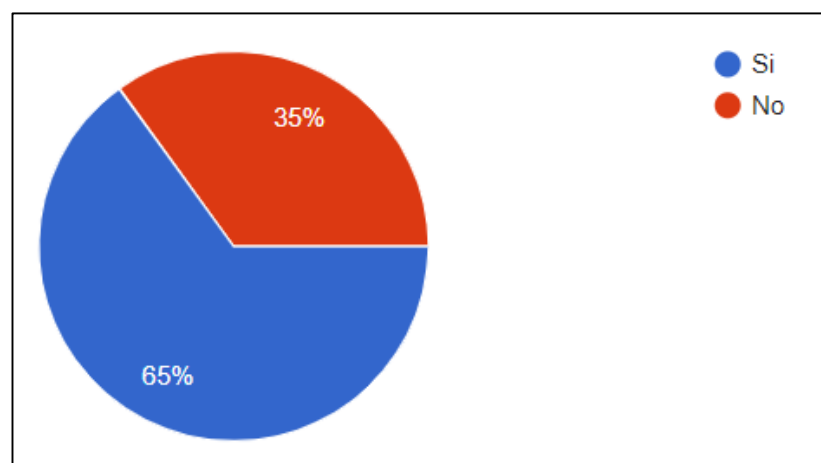


Figura 1. Resultados pregunta 1.

Análisis e interpretación de resultados

Según los datos presentados en la Figura 1, se observa que el 65% afirma tener conocimiento del proceso para efectuar una adopción de mascotas en la fundación, mientras que el 35% desconocen el proceso a seguir para efectuar una adopción. Se puede definir que existe un grupo considerable que no tiene conocimiento del proceso, lo que puede ser una oportunidad para mejorar la difusión de información sobre el proceso de adopción y facilitar el acceso al mismo.

Pregunta 2: En base a su experiencia, ¿cómo considera el proceso de adopción de mascotas con el que cuenta la fundación de rescate animal “Paquito”?

Tabla 7. Resultados pregunta 2.

Indicador	Frecuencia	Porcentaje
Pésima	0	0%
Mala	0	0%
Regular	7	35%
Buena	10	50%
Excelente	3	15%
Total	20	100%

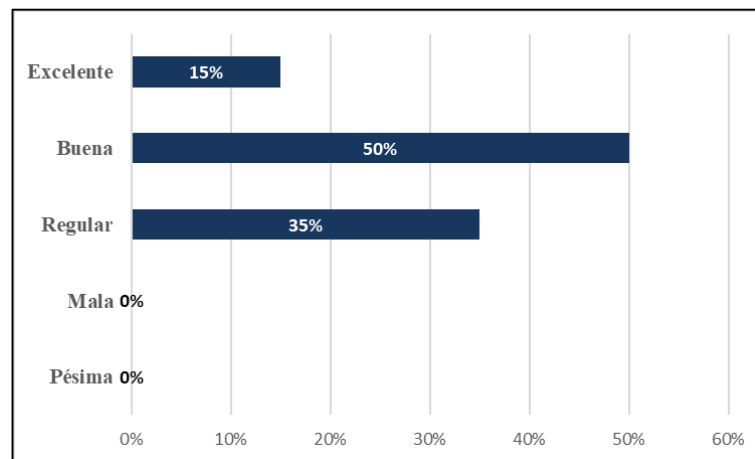


Figura 2. Resultados pregunta 2.

Análisis e interpretación de resultados

Con los resultados obtenidos en la Figura 2, se observa que el 50% de las personas considera que el proceso de adopción de mascotas con el que cuenta la fundación es

bueno, el 35% regular, mientras que un 15% considera que es excelente. Como resultado se evidencia que la mayoría de los encuestados tienen una opinión favorable sobre el proceso de adopción de mascotas. Sin embargo, hay un porcentaje importante de personas que tuvieron una experiencia regular durante el proceso de adopción. Estos resultados pueden ser útiles para la fundación al identificar áreas de mejora y mantener la calidad del proceso de adopción.

Pregunta 3: ¿En qué medida considera usted el nivel de eficiencia del proceso de solicitud de adopción de mascotas?

Tabla 8. Resultados pregunta 3.

Indicador	Frecuencia	Porcentaje
Muy Ineficiente	0	0%
Ineficiente	0	0%
Regular	5	25%
Eficiente	11	55%
Muy Eficiente	4	20%
Total	20	100%

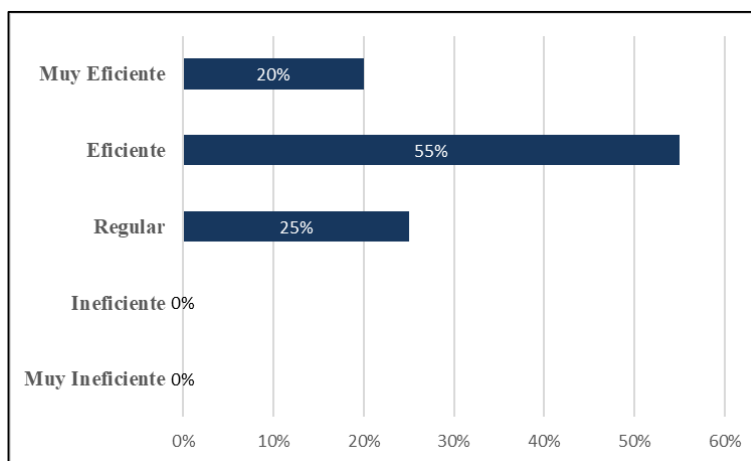


Figura 3. Resultados pregunta 3.

Análisis e interpretación de resultados

Con los resultados obtenidos en la Figura 3, el 55% de las personas considera que el proceso de solicitud de adopción de mascotas es eficiente, el 25% considera que es regular, el 20% que es muy eficiente. Por lo tanto, estos resultados indican que el

proceso de solicitud de adopción generalmente es bien recibido por la mayoría los usuarios.

Pregunta 4: En su opinión, ¿considera usted que la fundación de rescate animal “Paquito” proporciona información necesaria y detallada de las mascotas que están en adopción?

Tabla 9. Resultados pregunta 4.

Indicador	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	2	10%
Neutral	1	5%
De acuerdo	15	75%
Totalmente de acuerdo	2	10%
Total	20	100%

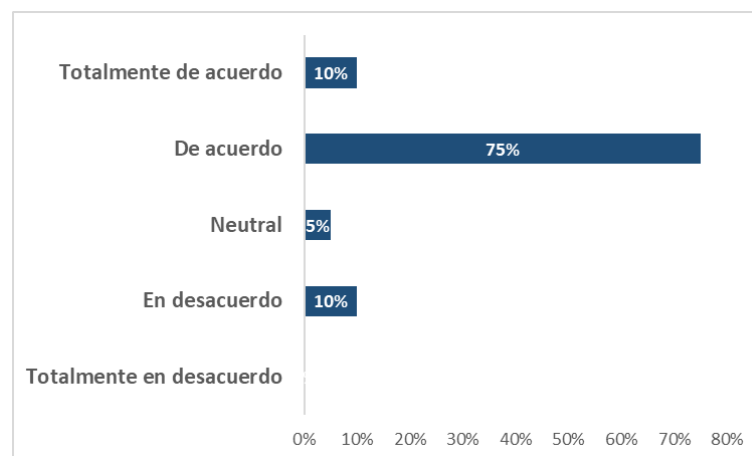


Figura 4. Resultados pregunta 4.

Análisis e interpretación de resultados

Con los resultados obtenidos en la Figura 4, el 75% de las personas está de acuerdo con la pregunta, el 5% se muestra neutral ante esta interrogante, el 10% está en desacuerdo y finalmente el otro 10% está totalmente de acuerdo. Se puede concluir que la mayoría de los encuestados opina que la fundación proporciona información necesaria y detallada sobre las mascotas en adopción. No obstante, un grupo minoritario de encuestados expresa desacuerdo ante esta interrogante. Es recomendable explorar las áreas en las que los encuestados puedan sentir que la

información no es suficiente o detallada, con el objetivo de mejorar la comunicación y satisfacer mejor las necesidades de quienes estén interesados en la adopción de mascotas.

Pregunta 5: ¿Por favor, indique en qué medida considera importante que la fundación de rescate animal “Paquito” realice un adecuado seguimiento de las mascotas que han sido dadas en adopción?

Tabla 10. Resultados pregunta 5.

Indicador	Frecuencia	Porcentaje
Nada importante	0	0%
De poca importancia	0	0%
Moderadamente importante	3	15%
Importante	5	25%
Muy importante	12	60%
Total	20	100%

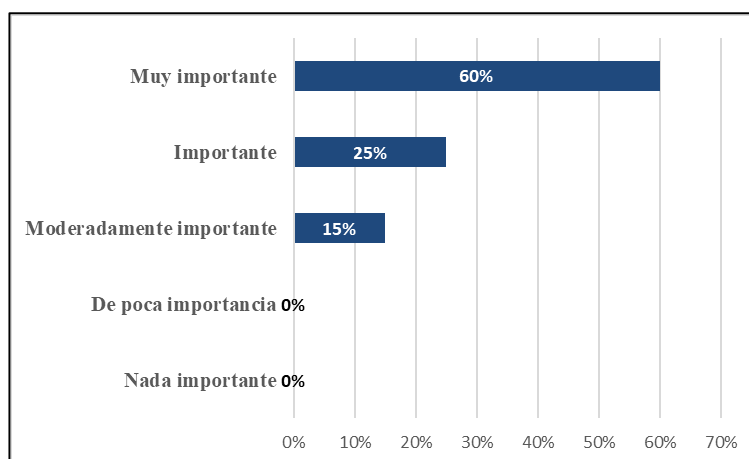


Figura 5. Resultados pregunta 5.

Análisis e interpretación de resultados

Con los resultados representados en la Figura 5, el 60% de personas considera muy importante que la fundación realice un adecuado seguimiento de las mascotas que han sido dadas en adopción, el 25% lo considera importante, mientras que al 15% moderadamente importante. Como consecuencia, se puede inferir que estos resultados demuestran el interés de las personas por el bienestar y la seguridad de las mascotas

después de la adopción y sugieren que la fundación debería dar seguimiento de las mascotas que han sido dadas en adopción.

Pregunta 6: ¿Defina en base a la siguiente escala el nivel de recursos económicos y el espacio físico idóneo con el que usted cuenta para proporcionar un hogar adecuado a una mascota adoptada?

Tabla 11. Resultados pregunta 6.

Indicador	Frecuencia	Porcentaje
Muy insuficiente	0	0%
Insuficiente	0	0%
Regular	7	35%
Suficiente	9	45%
Abundante	4	20%
Total	20	100%

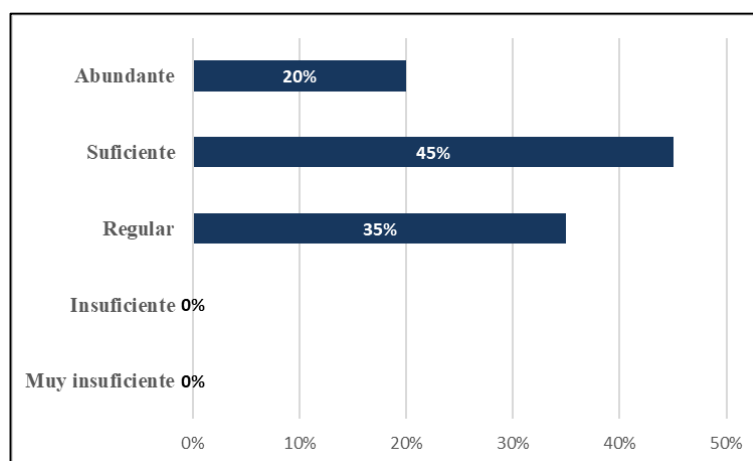


Figura 6. Resultados pregunta 6.

Análisis e interpretación de resultados

Con los resultados obtenidos en la *Figura 6*, donde el 45% de las personas encuestadas considera que posee los suficientes recursos económicos y el espacio físico idóneo para proporcionar un hogar adecuado a una mascota adoptada, el 35% contesta que regular, mientras el 20% responde que sus recursos son abundantes. En consecuencia, la mayoría de los encuestados se siente en condiciones de proporcionar un hogar

adecuado a una mascota adoptada. Estos resultados son útiles para evaluar de mejor manera la idoneidad de los adoptantes potenciales.

Pregunta 7: ¿Conoce usted de aplicaciones o sitios web donde se pueda realizar adopciones de mascotas?

Tabla 12. Resultados pregunta 7.

Indicador	Frecuencia	Porcentaje
Si	9	45%
No	11	55%
Total	20	100%

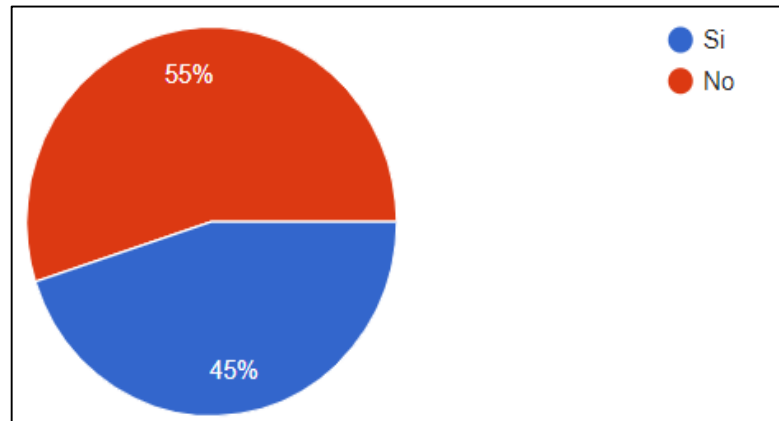


Figura 7. Resultados pregunta 7.

Análisis e interpretación de resultados

Con los resultados obtenidos en la Figura 7, se observa que el 55% de las personas encuestadas desconocen de aplicaciones o sitios web donde se pueda realizar adopciones de mascotas, mientras que el 45% tienen conocimiento de herramientas tecnológicas. Como resultado se evidencia el desconocimiento de aplicaciones y sitios web por parte de las personas, aumentando oportunamente el mercado para que las herramientas tecnológicas puedan favorecer en que exista más adopciones de mascotas.

Pregunta 8: ¿En base a sus aptitudes adquiridas, defina el nivel de dominio con el que usted cuenta en base al uso de la tecnología?

Tabla 13. Resultados pregunta 8.

Indicador	Frecuencia	Porcentaje
Muy difícil	0	0%
Difícil	0	0%
Regular	3	15%
Fácil	12	60%
Muy Fácil	5	25%
Total	20	100%

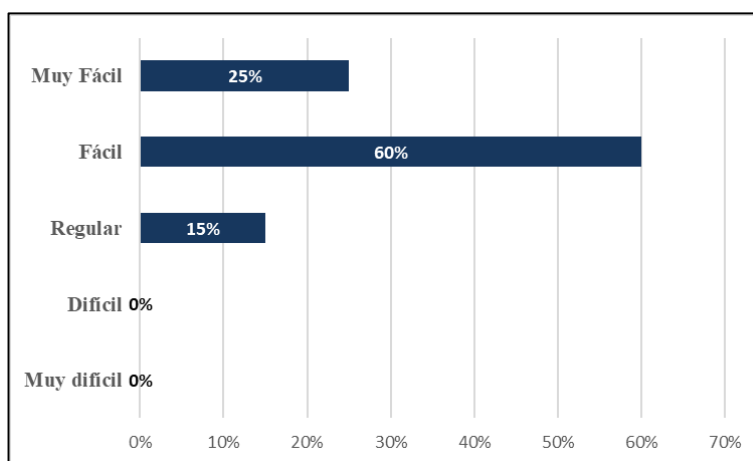


Figura 8. Resultados pregunta 8.

Análisis e interpretación de resultados

Con los resultados obtenidos en la Figura 8, al 60% de las personas encuestadas se les torna fácil el uso de la tecnología, al 25% muy fácil, al 15% regular. Estos resultados son útiles para comprender la preparación de la población encuestada tanto para el uso de tecnología como para la utilización del aplicativo web.

Pregunta 9: ¿A través de qué medio considera que la difusión de adopciones de mascotas puede alcanzar a más personas interesadas en realizar una adopción?

Tabla 14. Resultados pregunta 9.

Indicador	Frecuencia	Porcentaje
Redes Sociales	2	10%
Aplicativos Web	6	30%
Radio	2	10%
Televisión	2	10%

Carteles Exteriores	8	40%
Total	20	100%

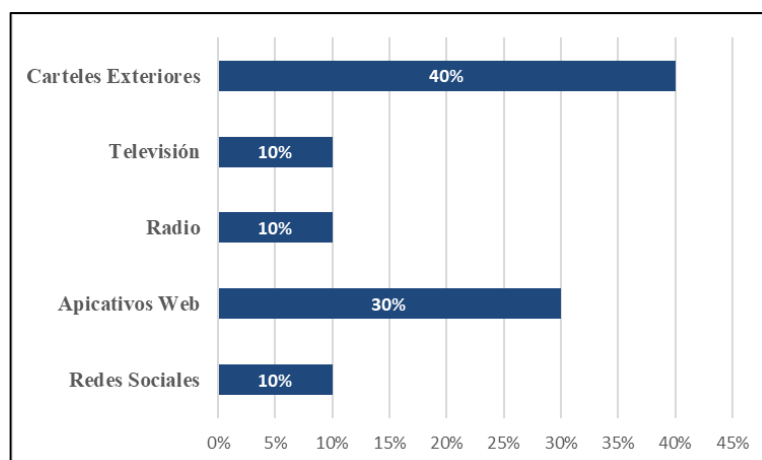


Figura 9. Resultados pregunta 9.

Análisis e interpretación de resultados

Los resultados de la pregunta formulada se han analizado en la Figura 9, donde el 10% de los encuestados considera que mediante redes sociales la difusión de adopciones de mascotas puede alcanzar a más personas interesadas en realizar una adopción, el 10% considera que, por medio de radio, el 30% piensa que, mediante aplicativos webs, el 40% considera que, mediante carteles exteriores, mientras el otro 10% cree que por televisión. Por lo tanto, el uso de plataformas tecnológicas predomina en este caso, siendo la alternativa más accesible para la difusión de adopciones de mascotas.

Pregunta 10: En general, ¿cómo calificaría su experiencia al adoptar una mascota en la fundación “Paquito”?

Tabla 15. Resultados pregunta 10.

Indicador	Frecuencia	Porcentaje
Totalmente Insatisfecho	0	0%
Insatisfecho	0	0%
Algo Satisfecho	2	10%
Satisfecho	7	35%
Totalmente Satisfecho	11	55%
Total	20	100%

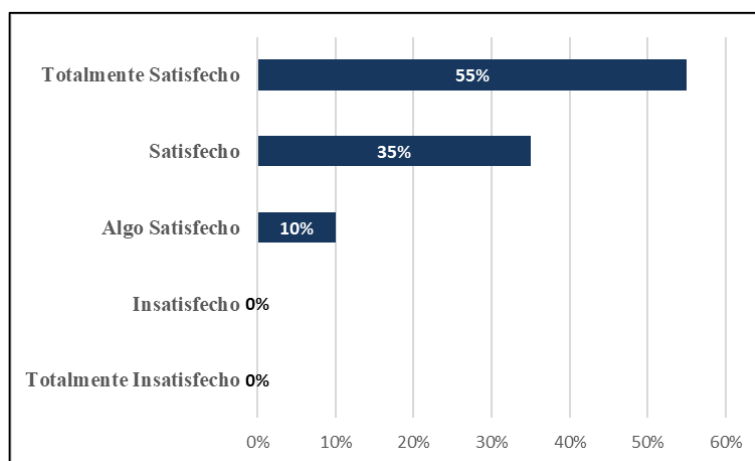


Figura 10. Resultados pregunta 10.

Análisis e interpretación de resultados

En base a los resultados representados en la Figura 10, el 55% de las personas encuestadas se sienten totalmente satisfechos con su experiencia al adoptar una mascota en la fundación, el 35% mencionan que se sienten satisfechos, el otro 10% se consideran algo satisfechos. En consecuencia, estos resultados son útiles para evaluar la satisfacción general de los adoptantes y proporciona información sobre la calidad de los servicios de la fundación. Se puede concluir que la experiencia de adopción aún tiene espacio para mejorar.

2.2.4 Procesamiento y análisis de datos

Con la información obtenida de la entrevista y las encuestas se determinó que.

- La mayoría de las personas no tienen el suficiente conocimiento acerca de los sitios web disponibles para la adopción de mascotas.
- Debido a que las personas no tienen el suficiente acercamiento hacia la tecnología se ha dinamizado dar a conocer mascotas en adopción por medio de carteles exteriores, siendo el método más conocido y tradicional. Pero con una limitación en cuanto al alcance de posibles adoptantes, una limitación que podría superarse mediante una herramienta tecnológica.

- Es importante que la fundación mejore la información que brinda acerca de las mascotas disponibles para ser adoptadas. Brindar información más detallada incentivaría de manera más efectiva la adopción de mascotas.
- Los insuficientes recursos económicos o falta de espacios físicos son una limitante para que las personas puedan realizar una adopción de mascotas.
- La mayor parte de las personas tienen conocimientos del uso de tecnologías lo cual nos indica que están capacitados para utilizar el aplicativo web.
- El proceso de solicitud de adopción de mascotas necesita mejoras debido a que las personas encuestadas manifestaron no estar completamente satisfechas con el proceso que tiene actualmente la fundación.
- Se considera necesario la implementación de un aplicativo web que permitirá a la fundación mantener un registro de todas las actividades relacionadas con el rescate de los animales, así como de las personas que están dispuestas a brindarles un hogar adecuado. Además, el aplicativo web permitirá mejorar el proceso de donaciones.

CAPÍTULO III. RESULTADOS Y DISCUSIÓN

3.1 Análisis y discusión de los resultados

Acorde a los resultados obtenidos en la recolección de datos se elaboró el modelado de los procesos de rescate y adopción de mascotas, voluntariado y donaciones de dinero. Además, se realiza una comparativa de las tecnologías open source de desarrollo frontend y backend y de las metodologías ágiles.

3.1.1 Modelado de procesos

A continuación, se realiza el modelado de los procesos que se automatizarán en la fundación.

3.1.1.1 Modelado del proceso de rescate de mascotas.

En la Figura 11, se detalla el proceso de rescate de mascotas que se ha diseñado para la fundación de rescate animal “Paquito”.

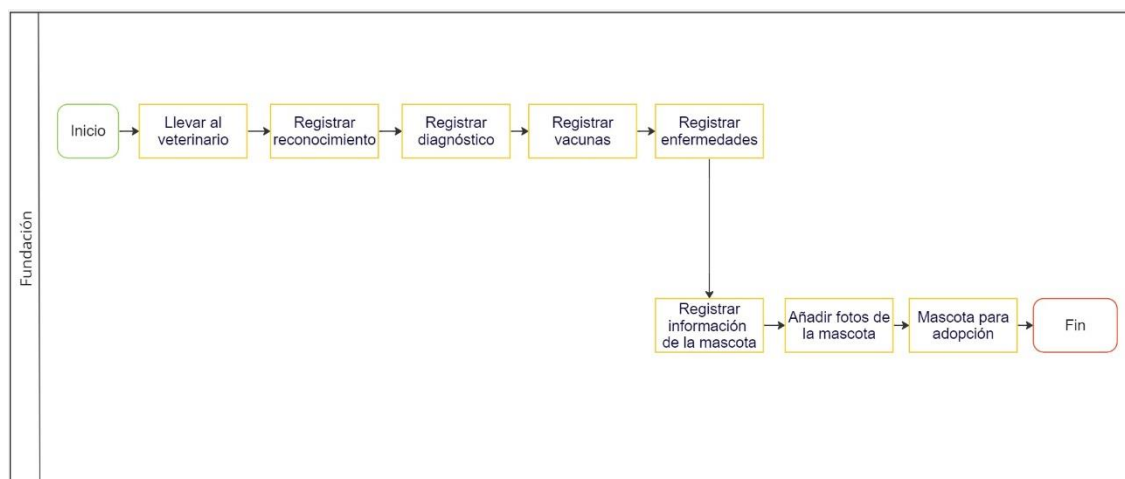


Figura 11. Diagrama del proceso de rescate de mascotas.

Descripción de las fases del proceso de rescate de mascotas.

1. Se lleva a la mascota rescatada al veterinario.
2. Una vez que la mascota ha sido revisada por el veterinario, se registra el reconocimiento el cual consta de datos como: el peso, las pruebas veterinarias realizadas.

3. Se realiza el registro del diagnóstico de las pruebas veterinarias que se ha realizado a las mascotas.
4. En caso de ser necesario se registra las vacunas que se le deben aplicar a las mascotas.
5. Se registra las enfermedades o alergias que puedan presentar las mascotas.
6. Se registra información general de las mascotas como: nombre, raza, sexo, tamaño, edad aproximada, dirección donde fue rescatada la mascota.
7. Se agrega fotos de la mascota.
8. Finalmente, la mascota es puesta en adopción.

3.1.1.2 Modelado del proceso de adopción de mascotas.

En la Figura 12, se detalla el proceso de adopción de mascotas que se ha diseñado para la fundación de rescate animal “Paquito”.

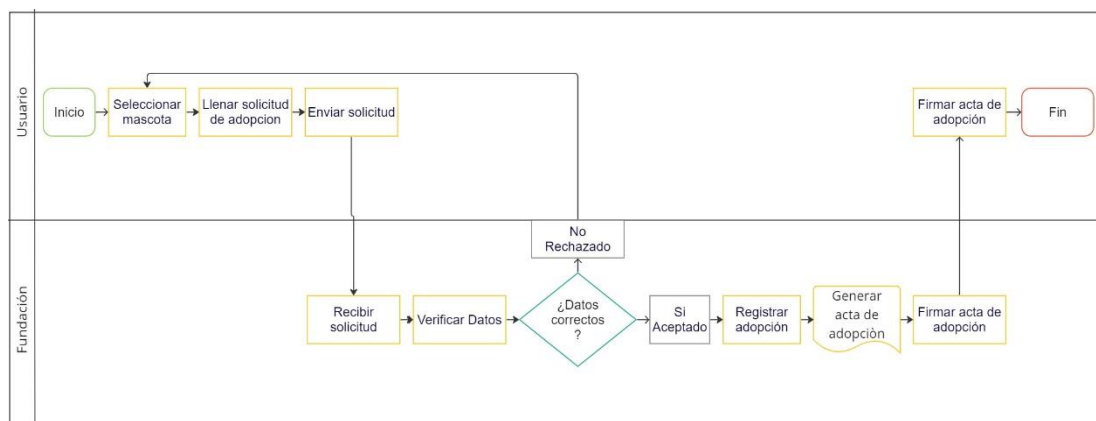


Figura 12. Diagrama del proceso de adopción de mascotas.

Descripción de las fases del proceso de adopción de mascotas.

1. El usuario selecciona la mascota que desea adoptar.
2. Llena una solicitud de adopción donde se realiza preguntas socioeconómicas y de relación con los animales.
3. Una vez completada la información el usuario envía la solicitud.
4. El administrador recibe la solicitud y evalúa la información para decidir si dar o no en adopción a la mascota.

5. Si la información ingresada es idónea el administrador procede a Aprobar la solicitud y genera el acta de adopción.
6. Si la solicitud no es aprobada el administrador Cancela la solicitud e informa al usuario el motivo por el cual su solicitud ha sido rechazada.
7. Finalmente, en caso de que se haya aprobado la solicitud se procede a firmar el acta de adopción entre el usuario y el administrador.

3.1.1.3 Modelado del proceso voluntariado.

En la Figura 13, se detalla el proceso de voluntariado que se ha diseñado para la fundación de rescate animal “Paquito”.

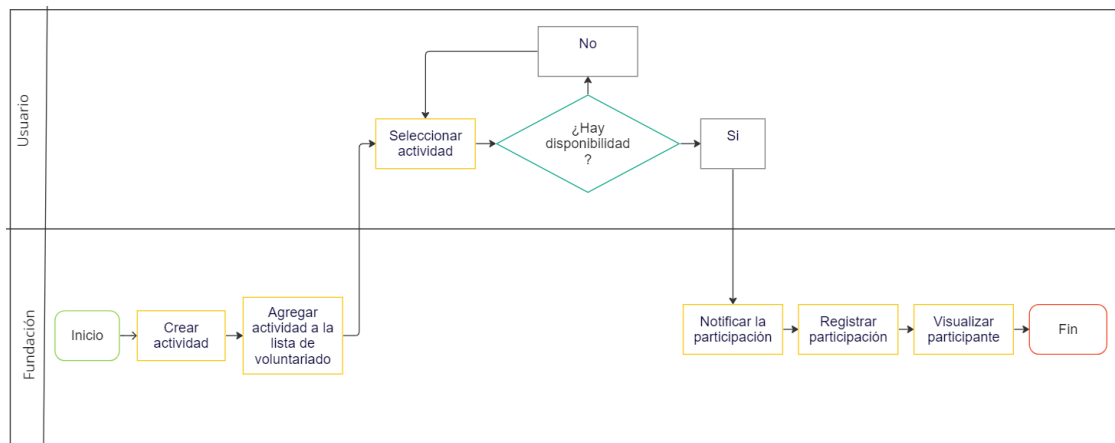


Figura 13. Diagrama del proceso de voluntariado.

Descripción de las fases del proceso de voluntariado.

1. El administrador crea una actividad de voluntariado indicando la fecha y hora en la cual se realizará.
2. Una vez creada la actividad se agrega a la lista de actividades para voluntariado.
3. El usuario visualizará la lista de actividades disponibles y seleccionará la actividad en la que desea participar.
4. El administrador visualizará los participantes que estén en cada actividad.

3.1.1.4 Modelado del proceso de donación de dinero.

En la Figura 14, se detalla el proceso de donación de dinero que se ha diseñado para la fundación de rescate animal “Paquito”.

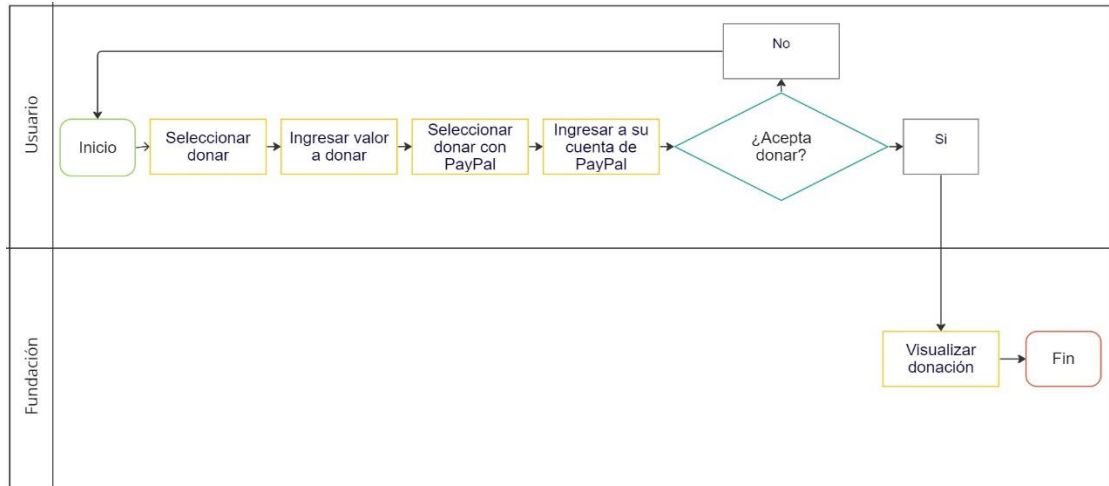


Figura 14. Diagrama del proceso de donación de dinero.

Descripción de las fases del proceso de donación de dinero.

1. El usuario ingresará a la opción de donar.
2. Ingresará el valor de dinero que desea donar.
3. Deberá seleccionar la opción de donar con PayPal y se le pedirá que ingrese a su cuenta de Papal para que pueda realizar la donación.
4. La donación se registrará únicamente si el usuario finaliza la donación.
5. El administrador podrá visualizar las donaciones.

3.2.1 Tecnologías open source para el desarrollo frontend en aplicativos web

Las tecnologías open source para el desarrollo frontend en aplicativos webs son conjuntos de herramientas, lenguajes de programación y recursos que los desarrolladores utilizan para crear la interfaz de usuario de una aplicación web. Estas tecnologías se enfocan en la parte del cliente de la aplicación y son esenciales para crear experiencias de usuario atractivas y funcionales.

A continuación, en la Tabla 16 se realiza una comparativa entre las tecnologías open source para el desarrollo frontend: Angular, React y Vue. El objetivo es analizar sus

características y seleccionar la tecnología que mejor se adapte a las necesidades de este proyecto. Para realizar la tabla se ha tomado como referencia las siguientes citas bibliográficas: [31], [32], [33], [34], [35].

Tabla 16. Comparación de tecnologías open source de desarrollo frontend.

Características	Angular	Vue	React
Arquitectura	Modelo-Vista-Controlador (MVC).	Modelo-Vista-Vista-Modelo (MVVM).	Basada en componentes.
Documentación	https://angular.io/cli	https://cli.vuejs.org/	https://es.react.dev/
Estadísticas de uso	22.96% de uso a nivel mundial.	18.97% de uso a nivel mundial.	40.14% de usos a nivel mundial.
Creadores	Google	Evan You	Facebook
Reutilización de componentes	Se enfoca en la encapsulación y la inyección de dependencias para la reutilización.	Prioriza la creación de componentes y utiliza slots y mixins para aumentar la flexibilidad de reutilización.	Se centra en la creación de componentes, y utiliza HOC y Hooks para reutilizar lógica entre componentes.
Curva de aprendizaje	Tiene una curva de aprendizaje alta debido a conceptos avanzados. Sin embargo, la	Tiene una curva de aprendizaje baja, debido a que no requiere nada más que tecnologías básicas del lado del	Tiene una curva de aprendizaje baja debido a que utiliza JSX que puede ser inusual al principio, pero

	documentación oficial y los recursos de la comunidad pueden ser valiosos para reducir la curva de aprendizaje.	cliente, como HTML, CSS y JavaScript	rápidamente se vuelve intuitivo y fácil de utilizar.
Lenguaje de programación	Utiliza TypeScript de manera nativa.	Utiliza JavaScript.	Utiliza JSX (JavaScript XML)
Compatibilidad con SEO (Search Engine Optimization)	Ofrece una buena compatibilidad con SEO de forma nativa gracias a Angular Universal.	Ofrece una buena compatibilidad con SEO de forma nativa debido a su representación del lado del servidor.	Requiere configuración adicional para SEO debido a la representación del lado del cliente.
Rendimiento	Angular tiene un sistema de detección de cambios que es eficiente en la actualización del DOM.	Actualiza solo las partes del DOM que cambian, lo que contribuye a la velocidad de carga.	En lugar de realizar cambios directamente en el DOM del navegador, React crea una representación virtual del DOM y compara los cambios antes de actualizar el DOM real.
Flexibilidad	Impone una estructura específica para las aplicaciones, lo que puede ser	Flexible, fácil de incorporar en proyectos existentes.	Altamente flexible, permite integración sencilla con otras

	beneficioso para proyectos grandes.		bibliotecas y frameworks.
--	-------------------------------------	--	---------------------------

En base al análisis comparativo que se presenta en la Tabla 16 entre las tecnologías open source para desarrollo del frontend, se ha seleccionado Angular. Esto debido a que tiene un sistema de detección de cambios que es eficiente en la actualización del DOM. Además, ofrece compatibilidad con SEO para mejorar el posicionamiento (la visibilidad) en buscadores del aplicativo web en Internet. Adicionalmente posee soporte extenso de herramientas y documentación, características esenciales para desarrollar un proyecto web robusto, mantenible y escalable.

3.1.2 Tecnologías open source para el desarrollo backend en aplicativos web

Las tecnologías open source para el desarrollo backend en aplicativo web son conjuntos de herramientas, lenguajes de programación y frameworks que los desarrolladores utilizan para construir la parte del servidor de una aplicación web. Esta parte del sistema se encarga de procesar las solicitudes de los usuarios, gestionar la lógica de negocio, interactuar con bases de datos y enviar las respuestas al frontend, que luego se presenta al usuario.

A continuación, en la Tabla 17, se realiza una comparativa entre las tecnologías open source para el desarrollo backend: .Net Core, Nodejs, Spring Boot. Y mediante esto seleccionar la tecnología que mejor se adapte a las necesidades de este proyecto. Para realizar la tabla se ha tomado como referencia las siguientes citas bibliográficas: [36], [37], [38], [39], [40], [41], [42].

Tabla 17. Comparación de tecnologías open source de desarrollo backend.

Características	.Net Core	Nodejs	Spring Boot
Lenguaje de programación	C#	JavaScript	Java
Arquitectura	MVC, Microservicios, API RESTful	Basada en módulos, API RESTful	MVC, Microservicios, API RESTful

Multiplataforma	Windows. Linux, MacOS	Windows. Linux, MacOS	Windows. Linux, MacOS
Documentación	https://learn.microsoft.com/en-us/aspnet/core/?view=aspnetcore-8.0	https://nodejs.org/en/docs	https://spring.io/projects/spring-boot
Creadores	Microsoft	Ryan Lienhart Dahl	VMware
Estadísticas de uso	16.57% de uso a nivel mundial.	42.65% de uso a nivel mundial.	11.95% de uso a nivel mundial.
Rendimiento	Aproximadamente 900 mil peticiones por minuto.	Aproximadamente 600 mil peticiones por minuto.	Aproximadamente 400 mil peticiones por minuto.
Gestión de paquetes	Utiliza NuGet para gestionar paquetes.	npm es el gestor de paquetes.	Utiliza Maven o Gradle para gestionar paquetes.
Curva de aprendizaje	Cuenta con una guía de aprendizaje muy detallada y bien organizada. Es fácil de entender debido a su documentación clara y completa.	Se basa en el lenguaje Javascript por lo que su curva de aprendizaje es baja.	Utiliza Java que tiene una curva de aprendizaje relativamente sencilla.
Eficiencia	Gracias a su técnica de compilación Just-in-Time (JIT) las aplicaciones pueden ejecutarse de manera más rápida y utilizar de	Aprovecha el motor V8 de para compilar y ejecutar código JavaScript a alta velocidad.	Permite crear aplicaciones autónomas que se ejecutan por sí solas, sin depender de un servidor web externo.

	manera más eficiente los recursos del sistema.		
Seguridad	Posee Autorización integrada, Autenticación, aplicación forzada de HTTPS.	Requiere APIs de terceros para configurar mecanismos de seguridad de manera exitosa y confiable.	Puede utilizar la seguridad de Liberty, CICS o la seguridad Java™ EE.

Una vez analizada la Tabla 17 de tecnologías para desarrollo backend se ha seleccionado .Net Core. Debido a que es un framework multiplataforma de código abierto que permite el desarrollo y la ejecución en diferentes sistemas operativos. Cuenta con una guía de aprendizaje muy detallada y bien organizada y la documentación es clara y detallada, proporcionando una guía fácil de seguir para los usuarios. Además, nos ofrece un alto rendimiento gracias a la compilación en tiempo de ejecución.

3.1.3 Determinación de metodología para desarrollo del aplicativo web

A continuación, en la Tabla 18 se realiza una comparativa entre las metodologías ágiles utilizadas en el desarrollo de proyectos: XP, KANBAN y SCRUM. Esta tabla tiene como objetivo proporcionar una comprensión más detallada de las características de cada metodología y de su aplicación práctica. Para realizar la tabla se ha tomado como referencia las siguientes citas bibliográficas: [43], [44], [45], [46], [47], [48], [49], [50], [51].

Tabla 18. Comparación de metodologías ágiles.

Características	XP	Kanban	Scrum
Tamaño del proyecto	Pequeños y medianos.	Pequeños, medianos y grandes.	Proyectos de cualquier tamaño y complejidad

Tamaño del equipo	5 integrantes o menos.	Puede tener de 5 a 14 integrantes.	De 5 a 9 integrantes
Enfoque	Se enfoca en la calidad del producto y la satisfacción del cliente.	Se enfoca en las mejoras continuas del proyecto.	Se enfoca en el trabajo en equipo.
Documentación oficial	www.extremeprogramming.org	www.kanban.university/kanban-guide/	www.scrum.org/resources/scrum-guide
Roles	Programador, Cliente, Tester, Coach, Encargado de Seguimiento	No hay roles específicos, algunos equipos pueden contar con la ayuda de un coach.	Product Owner, Scrum Master, Equipo de desarrollo
Ciclo de Vida	Exploración, Planificación, Iteraciones, Producción, Monitoreo	Inicio (To-Do), Trabajo en progreso (Doing), Listo para revisión (Done)	Sprint, Sprint Planning, Daily Scrum, Sprint Review, Sprint Retrospective
Ciclo de entrega	Se enfoca en realizar entregas frecuentes y pequeñas a medida que se desarrollan nuevas funcionalidades.	Las tareas avanzan a lo largo del tablero Kanban a medida que avanzan en el proceso de desarrollo, y una vez que una tarea está completada, se entrega al cliente.	Al final de cada Sprint, se lleva a cabo una revisión y se entrega el incremento del producto al cliente o usuario.

Gestión de cambios	Se destaca por su capacidad de respuesta a los cambios. Los cambios son bienvenidos en cualquier momento, ya que XP se basa en la retroalimentación continua.	Los cambios se pueden abordar tan pronto como se identifiquen y prioricen.	Si se identifica un cambio que debe abordarse se cancela el Sprint actual y comienza uno nuevo con los cambios necesarios.
Enfoque en la calidad	Las pruebas unitarias, la integración continua y la refactorización son prácticas clave en XP que ayudan a garantizar la calidad del software.	La detección temprana de problemas y la resolución inmediata son esenciales para mantener la calidad. Si se identifican defectos, se priorizan y se corrigen	Durante un Sprint, el equipo trabaja en la implementación y prueba de las funcionalidades planificadas. Se espera que cada funcionalidad cumpla con los criterios de aceptación definidos.

Después de realizar una evaluación comparativa de diversas metodologías ágiles, se ha determinado que la metodología XP es la elección más adecuada para la ejecución del proyecto. La metodología destaca por su capacidad para adaptarse a cambios constantes en los requisitos del proyecto. Además, su enfoque en la retroalimentación continua garantiza la alineación precisa del aplicativo web con las expectativas reales de la fundación. La colaboración activa de la fundación y el énfasis en el desarrollo incremental y entregas frecuentes garantiza aún más la calidad final del proyecto.

3.1.4 Arquitectura del aplicativo web

El aplicativo web está construido utilizando Angular para crear la parte visible y de interacción del aplicativo (frontend), mientras que .Net Core se emplea para desarrollar la parte lógica y funcional del sistema (backend). Se implementa una API REST para acceder y gestionar los datos, y el sistema de gestión de bases de datos MySQL se utiliza para almacenar la información.

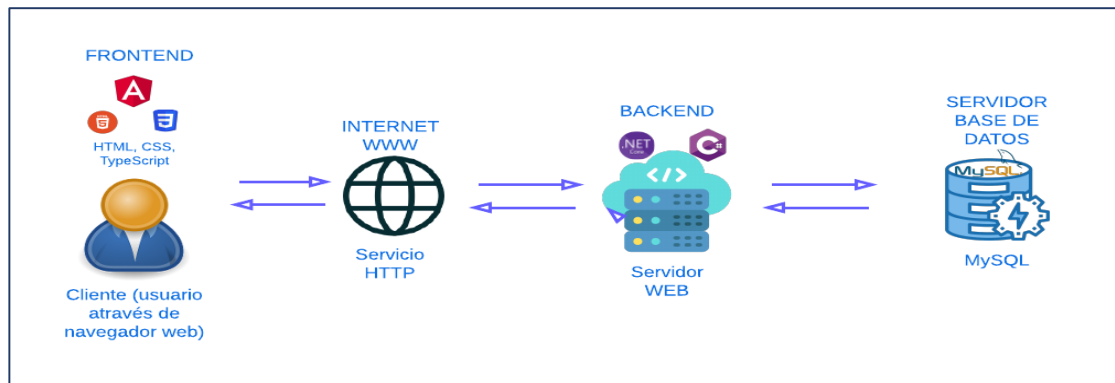


Figura 15. Arquitectura del aplicativo web.

3.2 Desarrollo de la propuesta

De acuerdo con la metodología XP, es necesario seguir una serie de fases, que incluyen planificación, diseño, codificación, pruebas y lanzamiento. Cada una de estas etapas requiere cumplir con una serie de pasos específicos para garantizar un desarrollo de software organizado.

En la Figura 16, se detalla las fases de la metodología Extreme Programming.

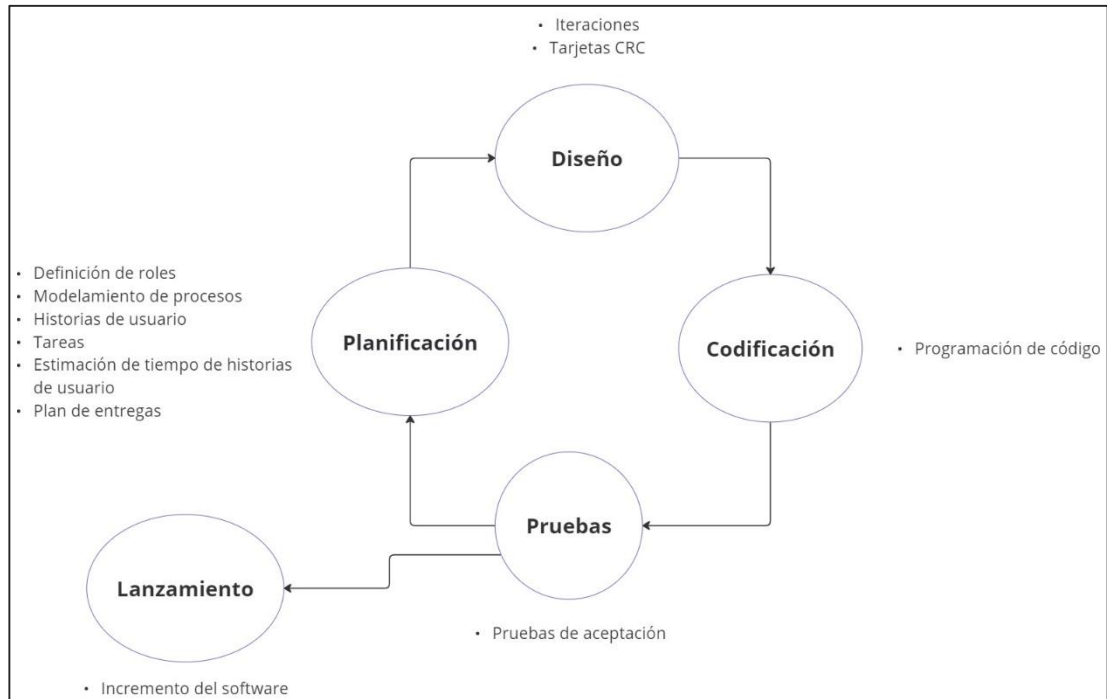


Figura 16. Diagrama de fases de la metodología Extreme Programming.

3.2.1 Fase I: Planificación

La planificación es la fase donde el cliente, los gerentes y el grupo desarrolladores acuerdan el orden en que se implementarán las historias usuario y también las entregas.

3.2.1.1 Grupos de interés (Stakeholders)

- Usuarios del aplicativo web: Personal administrativo, personas interesadas en adoptar una mascota, realizar voluntariado o donar dinero.
- Desarrollador: Autor del presente proyecto.

3.2.1.2 Roles del Proyecto

La asignación de roles es clave para una organización efectiva entre los miembros del equipo y el cliente. En el presente proyecto se han determinado los siguientes roles en base a la metodología escogida (XP).

Tabla 19. Asignación de roles XP.

Miembro del equipo	Rol	Función
Freddy Remache	Programador y encargado de pruebas	Responsable de la planificación, codificación, implementación y pruebas del sistema.
Ing. Hernán Naranjo	Coach	Acompañamiento y retroalimentación.
Fundación Paquito	Cliente	Otorga la descripción de los requisitos del aplicativo web.

Requerimientos de implementación del aplicativo web.

En la Tabla 20 se especifican los requerimientos del aplicativo web.

Tabla 20. Requerimientos de implementación.

N°	Historia	Descripción	Prioridad
1	Formulario para registrar información médica de las mascotas.	El formulario permitirá registrar información médica relevante de las mascotas, como historial de vacunación, enfermedades o alergias y condiciones de salud.	Alta
2	Acta de adopción.	El acta será documento en formato PDF, contendrá información detallada sobre el adoptante, la mascota y los términos de la adopción.	Media
3	Catálogo de mascotas.	Será un catálogo visual y organizado que presente información sobre las mascotas disponibles para adopción. Este catálogo facilitará a los usuarios la búsqueda y selección de mascotas.	Alta
4	Interfaz para actualizar información del usuario.	La interfaz permitirá a los usuarios actualizar su información personal, como dirección, número de teléfono, estado civil, correo.	Alta
5	Interfaz para la asignación de perfiles de usuario.	Permitirá asignar perfiles de usuario con diferentes niveles de acceso y privilegios dentro del aplicativo web, garantizando la seguridad y la adecuada gestión de funcionalidades.	Alta

6	Interfaz para gestionar información de las mascotas.	La interfaz permitirá a los usuarios autorizados gestionar y mantener actualizada la información detallada de las mascotas.	Alta
7	Creación de la estructura del aplicativo web.	Se creará la estructura del aplicativo web, estableciendo el diseño de la base de datos, la arquitectura base y los componentes necesarios para el desarrollo de las funcionalidades.	Alta
8	Interfaz de autenticación y autorización.	La interfaz deberá ser la pantalla de inicio del aplicativo web permitirá ingresar un correo y una contraseña para que el usuario pueda autenticarse.	Alta
9	Formulario de solicitud de adopción.	El formulario permitirá a los usuarios interesados en adoptar una mascota completar y enviar su solicitud. Este formulario capturará información relevante para evaluar la idoneidad del adoptante.	Media
10	Catálogo de actividades para voluntariado.	Se desarrollará un catálogo que muestre y describa las actividades disponibles para voluntariado, proporcionando detalles sobre las tareas, fechas.	Baja
11	Desarrollo de la notificación de resolución de solicitud de adopción.	Mostrará una notificación que informe a los solicitantes sobre la resolución de sus solicitudes de adopción, ya sea aprobadas o rechazada.	Media
12	Formulario de registro.	El formulario permitirá a nuevos usuarios registrarse en el aplicativo web, proporcionando la información necesaria para crear una cuenta.	Alta
13	Interfaz de reportes.	Se generará y visualizará gráficas estadísticas relacionadas con el proceso de adopción y actividades de voluntariado.	Baja
14	Formulario para crear actividades de voluntariado.	Se creará un formulario que facilite la creación de nuevas actividades de voluntariado, permitiendo describir la actividad, establecer fecha y duración de la actividad.	Baja
15	Interfaz para gestionar solicitudes de adopciones.	La interfaz permitirá revisar, evaluar y gestionar las solicitudes de adopción recibidas, asegurando un proceso eficiente y transparente.	Media

16	Automatización de recepción de donaciones de dinero.	Se trabajará en la automatización del proceso de recepción de donaciones de dinero, facilitando y agilizando la contribución financiera.	Baja
----	--	--	------

3.2.1.3 Historias de Usuario

En base a la prioridad y necesidades del proyecto se implementará las historias de usuario en el siguiente orden:

Tabla 21. Requerimientos de implementación del aplicativo web.

N°	Historia	Descripción	Prioridad
1	Creación de la estructura del aplicativo web	Se creará la estructura del aplicativo web, estableciendo el diseño de la base de datos, la arquitectura base y los componentes necesarios para el desarrollo de las funcionalidades.	Alta
2	Formulario de registro	El formulario permitirá a nuevos usuarios registrarse en el aplicativo web, proporcionando la información necesaria para crear una cuenta.	Alta
3	Interfaz de autenticación y autorización	La interfaz deberá ser la pantalla de inicio del aplicativo web permitirá ingresar un correo y una contraseña para que el usuario pueda autenticarse.	Alta
4	Interfaz para actualizar información del usuario	La interfaz permitirá a los usuarios actualizar su información personal, como dirección, número de teléfono, estado civil, correo.	Alta
5	Interfaz para la asignación de perfiles de usuario.	Permitirá asignar perfiles de usuario con diferentes niveles de acceso y privilegios dentro del aplicativo web, garantizando la seguridad y la adecuada gestión de funcionalidades.	Alta
6	Interfaz para gestionar información de las mascotas	La interfaz permitirá a los usuarios autorizados gestionar y mantener actualizada la información detallada de las mascotas.	Alta
7	Formulario para registrar información médica de las mascotas	El formulario permitirá registrar información médica relevante de las mascotas, como historial de	Alta

		vacunación, enfermedades o alergias y condiciones de salud.	
8	Catálogo de mascotas.	Será un catálogo visual y organizado que presente información sobre las mascotas disponibles para adopción. Este catálogo facilitará a los usuarios la búsqueda y selección de mascotas.	Alta
9	Formulario de solicitud de adopción.	El formulario permitirá a los usuarios interesados en adoptar una mascota completar y enviar su solicitud. Este formulario capturará información relevante para evaluar la idoneidad del adoptante.	Media
10	Interfaz para gestionar solicitudes de adopciones.	La interfaz permitirá revisar, evaluar y gestionar las solicitudes de adopción recibidas, asegurando un proceso eficiente y transparente.	Media
11	Desarrollo de la notificación de resolución de solicitud de adopción	Mostrará una notificación que informe a los solicitantes sobre la resolución de sus solicitudes de adopción, ya sea aprobadas o rechazada.	Media
12	Acta de adopción	El acta será documento en formato PDF, contendrá información detallada sobre el adoptante, la mascota y los términos de la adopción.	Media
13	Interfaz de reportes	Se generará y visualizará gráficas estadísticas relacionadas con el proceso de adopción y actividades de voluntariado.	Baja
14	Formulario para crear actividades de voluntariado.	Se creará un formulario que facilite la creación de nuevas actividades de voluntariado, permitiendo describir la actividad, establecer fecha y duración de la actividad.	Baja
15	Catálogo de actividades para voluntariado	Se desarrollará un catálogo que muestre y describa las actividades disponibles para voluntariado, proporcionando detalles sobre las tareas, fechas.	Baja
16	Automatización de recepción de donaciones de dinero	Se trabajará en la automatización del proceso de recepción de donaciones de dinero, facilitando y agilizando la contribución financiera.	Baja

Para describir los requisitos funcionales del aplicativo web se utilizarán historias de usuario. El modelo de las historias de usuario se detalla en la Tabla 22.

Tabla 22. Modelo de historia de usuario.

Historia de Usuario	
Número <Identificador de la historia>	Usuario <Persona que proporciona la información>
Nombre de historia <Título designado a la historia de usuario>	
Prioridad en negocio <Cuantificador de la necesidad de negocio (Alta, Media, Baja)>	Riesgo en desarrollo <Cuantificador del impacto en caso de fallo (Alta, Media, Baja)>
Puntos estimados <Días estimados designados para el desarrollo>	Iteración asignada <Número de la iteración de la historia>
Programador responsable <Persona encargada del desarrollo de la historia>	
Descripción <Descripción de la historia con un lenguaje natural>	
Criterios de Aceptación	
< Se describe las condiciones necesarias que se ejecute cada historia de usuario>	

Tabla 23. Historia de usuario 1: Creación de la estructura del aplicativo web.

Historia de Usuario	
Número: 1	Usuario: Personal administrativo
Nombre de historia: Creación de la estructura del aplicativo web.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Freddy Remache	
Descripción: Como personal administrativo quiero almacenar de manera estructura toda la información de los procesos que la fundación lleva a cabo para automatizar dichos procesos y evitar redundancia y pérdida de información.	
Criterios de Aceptación	
Criterio de aceptación 1: Dado que el sistema almacenará la información de las actividades de rescate, adopción, y donaciones de dinero cuando se necesite llevar un registro de los datos	

entonces diseñar una base de datos que incluya las tablas y campos que permitan almacenar la información.

Tabla 24. Actividad 1: Diseño conceptual de la base de datos del aplicativo web.

Tarea	
Número: 1	Número de historia: 1
Nombre de tarea: Diseño conceptual de la base de datos del aplicativo web.	
Tipo de tarea: Esquematación de base de datos.	Puntos estimados: 1
Fecha inicio: 27/08/2023	Fecha finalización: 27/08/2023
Descripción: Modelar el esquema conceptual de la base de datos con las entidades que intervienen en las actividades de adopción, voluntariado y donaciones de dinero de la fundación.	

Tabla 25. Actividad 2: Creación del proyecto en backend.

Tarea	
Número: 2	Número de historia: 1
Nombre de tarea: Creación del proyecto en backend.	
Tipo de tarea: Desarrollo Backend.	Puntos estimados: 1
Fecha inicio: 28/08/2023	Fecha finalización: 28/08/2023
Descripción: Creación del proyecto con bibliotecas de clases y modelado de las capas en .Net Core.	

Tabla 26. Actividad 3: Creación del proyecto en frontend.

Tarea	
Número: 3	Número de historia: 1
Nombre de tarea: Creación del proyecto en frontend.	
Tipo de tarea: Desarrollo Frontend.	Puntos estimados: 1
Fecha inicio: 29/08/2023	Fecha finalización: 29/08/2023
Descripción: Creación del proyecto utilizando el patrón MVC (Modelo-Vista-Controlador) en Angular.	

Tabla 27. Historia de usuario 2: Formulario de registro.

Historia de Usuario	
Número: 2	Usuario: Usuario del aplicativo

Nombre de historia: Formulario de registro.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 8	Iteración asignada: 1
Programador responsable: Freddy Remache	
Descripción: Como usuario del aplicativo web quiero registrar mis datos de manera segura para utilizar las diferentes funcionalidades del aplicativo web.	
Criterios de Aceptación	
Criterio de aceptación 1: Dado que el usuario complete los campos del formulario de manera correcta cuando presione el botón “Registrarse” entonces el aplicativo notificará que el usuario ha sido creado correctamente.	
Criterio de aceptación 2: Dado que el usuario no haya completado todos los campos del formulario cuando presione el botón “Registrarse” entonces el aplicativo emitirá un mensaje de advertencia indicando que complete todos los campos del formulario.	

Tabla 28. Actividad 4: Diseño de la interfaz de usuario.

Tarea	
Número: 4	Número de historia: 2
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 30/08/2023	Fecha finalización: 30/08/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica del formulario de registro.	

Tabla 29. Actividad 5: Creación del modelo entidad relación del formulario de registro.

Tarea	
Número: 5	Número de historia: 2
Nombre de tarea: Creación del modelo entidad relación del formulario de registro.	
Tipo de tarea: Diseño del modelo entidad relación.	Puntos estimados: 2
Fecha inicio: 31/08/2023	Fecha finalización: 01/09/2023

Descripción: Identificar las entidades que intervienen, la determinación de sus atributos, la definición de las relaciones entre ellas y la especificación de las claves primarias y foráneas.

Tabla 30. Actividad 6: Codificación del formulario de registro.

Tarea	
Número: 6	Número de historia: 2
Nombre de tarea: Codificación del formulario de registro.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 4
Fecha inicio: 02/09/2023	Fecha finalización: 05/09/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad del formulario.	

Tabla 31. Actividad 7: Pruebas Unitarias.

Tarea	
Número: 7	Número de historia: 2
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 2
Fecha inicio: 06/09/2023	Fecha finalización: 06/09/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

Regístrase

Cédula*

Nombres*

Apellidos*

Celular*

Día* Mes* Año*

Estado Civil *

Correo Electrónico*

Contraseña*

Repite contraseña*

Acepto los términos y condiciones

Regístrame

Logotipo

Figura 17. Diseño preliminar de la interfaz para Registrarse.

Tabla 32. Historia de usuario 3: Interfaz de autenticación y autorización.

Historia de Usuario	
Número: 3	Usuario: Todos
Nombre de historia: Interfaz de autenticación y autorización.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 8	Iteración asignada: 1
Programador responsable: Freddy Remache	
Descripción: Como usuario del aplicativo web quiero ingresar al aplicativo web de manera segura para tener acceso a las funcionalidades.	
Criterios de Aceptación	
Criterio de aceptación 1: Dado un usuario que ingrese su correo y contraseña cuando estos sean correctos entonces el aplicativo permite el ingreso del usuario a las funciones pertinentes a su rol.	
Criterio de aceptación 2: Dado un usuario presione el botón de Iniciar cuando no se haya completado datos en correo y contraseña entonces el aplicativo notificará que se deben completar los campos de correo y contraseña.	

Tabla 33. Actividad 8: Diseño de la interfaz de usuario.

Tarea	
Número: 8	Número de historia: 3
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 07/09/2023	Fecha finalización: 07/09/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica de autenticación y autorización.	

Tabla 34. Actividad 9: Creación del modelo entidad relación de la interfaz de autenticación y autorización

Tarea	
Número: 9	Número de historia: 3

Nombre de tarea: Creación del modelo entidad relación de la interfaz de autenticación y autorización.	
Tipo de tarea: Diseño del modelo entidad relación.	Puntos estimados: 2
Fecha inicio: 08/09/2023	Fecha finalización: 09/09/2023
Descripción: Identificar las entidades que intervienen, la determinación de sus atributos, la definición de las relaciones entre ellas y la especificación de las claves primarias y foráneas.	

Tabla 35. Actividad 10: Codificación de la interfaz de autenticación y autorización.

Tarea	
Número: 10	Número de historia: 3
Nombre de tarea: Codificación de la interfaz de autenticación y autorización.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 4
Fecha inicio: 10/09/2023	Fecha finalización: 13/09/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad la interfaz de autenticación y autorización.	

Tabla 36. Actividad 11: Pruebas Unitarias.

Tarea	
Número: 11	Número de historia: 3
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 14/09/2023	Fecha finalización: 14/09/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

Figura 18. Diseño preliminar la interfaz de autenticación y autorización.

Tabla 37. Historia de usuario 4: Interfaz para actualizar información del usuario.

Historia de Usuario	
Número: 4	Usuario: Todos
Nombre de historia: Interfaz para actualizar información del usuario..	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 6	Iteración asignada: 1
Programador responsable: Freddy Remache	
Descripción: <p>Como usuario del aplicativo web quiero tener la posibilidad de tener acceso a la información registrada en el aplicativo web para poder actualizar información en caso de ser necesario.</p>	
Criterios de Aceptación	
Criterio de aceptación 1: <p>Dado que el usuario complete los campos del formulario que desee actualizar de manera correcta cuando presione el botón “Guardar” entonces el aplicativo notificará que los datos han sido actualizados.</p>	
Criterio de aceptación 2: <p>Dado que el usuario no haya completado los campos que desea actualizar cuando presione el botón “Guardar” entonces el aplicativo emitirá un mensaje de advertencia indicando que complete todos los campos.</p>	

Tabla 38. Actividad 12: Diseño de la interfaz de usuario.

Tarea	
Número: 12	Número de historia: 4
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 15/09/2023	Fecha finalización: 15/09/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica de autenticación y autorización.	

Tabla 39. Actividad 13: Creación del modelo entidad relación para actualizar información del usuario.

Tarea	
Número: 13	Número de historia: 4
Nombre de tarea: Creación del modelo entidad relación para actualizar información del usuario.	
Tipo de tarea: Diseño del modelo entidad relación.	Puntos estimados: 1
Fecha inicio: 16/09/2023	Fecha finalización: 16/09/2023
Descripción: Identificar las entidades que intervienen, la determinación de sus atributos, la definición de las relaciones entre ellas y la especificación de las claves primarias y foráneas.	

Tabla 40. Actividad 14: Codificación de la interfaz para actualizar información del usuario.

Tarea	
Número: 14	Número de historia: 4
Nombre de tarea: Codificación de la interfaz para actualizar información del usuario.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 3
Fecha inicio: 19/09/2023	Fecha finalización: 19/09/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad de la interfaz para actualizar información del usuario.	

Tabla 41. Actividad 15: Pruebas Unitarias.

Tarea	
Número: 15	Número de historia: 4

Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 20/09/2023	Fecha finalización: 20/09/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

Modificar Usuario

Figura 19. Diseño preliminar del formulario de usuarios.

Tabla 42. Historia de usuario 5: Interfaz para asignar perfiles de usuario.

Historia de Usuario	
Número: 5	Usuario: Personal administrativo
Nombre de historia: Interfaz para asignar perfiles de usuario.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 6	Iteración asignada: 2
Programador responsable: Freddy Remache	
Descripción:	
<p>Como personal administrativo de la fundación quiero asignar los perfiles de usuarios del aplicativo web con el fin de asignar roles y permisos a cada usuario para que solo puedan acceder a las funcionalidades que les corresponda.</p>	
Criterios de Aceptación	

Criterio de aceptación 1:

Dado que el administrador desee asignar perfiles de usuario a los usuarios registrados **cuando** asigne el perfil de “Administrador” a un usuario **entonces** se permitirá el acceso a todas las funcionalidades del aplicativo.

Criterio de aceptación 2:

Dado que el administrador desee asignar perfiles de usuario a los usuarios registrados **cuando** asigne el perfil de “Usuario Normal” a un usuario **entonces** se permitirá el acceso a las funciones básicas del aplicativo.

Tabla 43. Actividad 16: Diseño de la interfaz de usuario.

Tarea	
Número: 16	Número de historia: 5
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 21/09/2023	Fecha finalización: 21/09/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica para asignar perfiles de usuario.	

Tabla 44. Actividad 17: Codificación de la interfaz para asignar perfiles de usuario.

Tarea	
Número: 17	Número de historia: 5
Nombre de tarea: Codificación <i>de la</i> interfaz para asignar perfiles de usuario.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 4
Fecha inicio: 22/09/2023	Fecha finalización: 25/09/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad de la interfaz para asignar perfiles de usuario.	

Tabla 45. Actividad 18: Pruebas Unitarias.

Tarea	
Número: 18	Número de historia: 5
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 26/09/2023	Fecha finalización: 26/09/2023

Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.

Tipos de Usuarios	
Administrador	
<input type="radio"/>	Puede usar todas las funciones del sistema, incluidas las configuraciones
Usuario Normal	
<input type="radio"/>	Solo puede usar las funciones básicas del sistema

Figura 20. Diseño preliminar de la interfaz para asignar perfiles de usuario.

Tabla 46. Historia de usuario 6: Interfaz para gestionar información de las mascotas.

Historia de Usuario	
Número: 6	Usuario: Personal administrativo
Nombre de historia: Interfaz para gestionar información de las mascotas.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 8	Iteración asignada: 2
Programador responsable: Freddy Remache	
Descripción:	
<p>Como personal administrativo de la fundación quiero gestionar la información de las mascotas rescatadas para tener los datos de cada mascota organizados y de esta manera puedan ser dadas en adopción.</p>	
Criterios de Aceptación	
Criterio de aceptación 1:	
<p>Dado que el administrador complete los campos del formulario para registrar o actualizar información de una mascota cuando presione el botón Crear entonces el aplicativo notificará que los datos han sido guardados o actualizados según sea el caso.</p>	
Criterio de aceptación 2:	

Dado que el administrador no haya completado todos los campos del formulario **cuando** presione el botón “Crear” **entonces** el aplicativo emitirá un mensaje de advertencia indicando que complete todos los campos del formulario.

Tabla 47. Actividad 19: Diseño de la interfaz de usuario.

Tarea	
Número: 19	Número de historia: 6
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 27/09/2023	Fecha finalización: 27/09/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica para la gestión de la información de las mascotas.	

Tabla 48. Actividad 20: Creación del modelo entidad relación para la gestión de la información de las mascotas.

Tarea	
Número: 20	Número de historia: 6
Nombre de tarea: Creación del modelo entidad relación para la gestión de la información de las mascotas.	
Tipo de tarea: Diseño del modelo entidad relación.	Puntos estimados: 2
Fecha inicio: 28/09/2023	Fecha finalización: 29/09/2023
Descripción: Identificar las entidades que intervienen, la determinación de sus atributos, la definición de las relaciones entre ellas y la especificación de las claves primarias y foráneas.	

Tabla 49. Actividad 21: Codificación del formulario.

Tarea	
Número: 21	Número de historia: 6
Nombre de tarea: Codificación del formulario.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 3
Fecha inicio: 30/09/2023	Fecha finalización: 02/10/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad del formulario para la gestión de la información de las mascotas.	

Tabla 50. Actividad 22: Pruebas Unitarias.

Tarea	
Número: 22	Número de historia: 6
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 2
Fecha inicio: 03/10/2023	Fecha finalización: 03/10/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

Crear Mascota

Nombre de la mascota*

Tipo de Mascota*

Raza *

Sexo *

Tamaño *

Día* Mes* Año*

Dirección *

Esterilizado

Figura 21. Diseño preliminar de la interfaz para gestionar información de las mascotas

Tabla 51. Historia de usuario 7: Formulario para registrar información médica de las mascotas.

Historia de Usuario	
Número: 7	Usuario: Personal administrativo
Nombre de historia: Formulario para registrar información médica de las mascotas.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 7	Iteración asignada: 2
Programador responsable: Freddy Remache	

<p>Descripción:</p> <p>Como personal administrativo de la fundación quiero registrar la información médica de las mascotas rescatadas para tener un registro de las vacunas y del estado de salud de las mascotas.</p>
Criterios de Aceptación
<p>Criterio de aceptación 1:</p> <p>Dado que el administrador complete los campos del formulario de manera correcta cuando presione el botón “Guardar” entonces el aplicativo registrará la información.</p> <p>Criterio de aceptación 2:</p> <p>Dado que el administrador no haya completado todos los campos del formulario cuando presione el botón “Guardar” entonces el aplicativo emitirá un mensaje de advertencia indicando que complete todos los campos del formulario.</p>

Tabla 52. Actividad 23: Diseño de la interfaz de usuario.

Tarea	
Número: 23	Número de historia: 7
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 04/10/2023	Fecha finalización: 04/10/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica del formulario para registrar información médica de las mascotas.	

Tabla 53. Actividad 24: Creación del modelo entidad relación del formulario para registrar información médica de las mascotas.

Tarea	
Número: 24	Número de historia: 7
Nombre de tarea: Creación del modelo entidad relación del formulario para registrar información médica de las mascotas.	
Tipo de tarea: Diseño del modelo entidad relación.	Puntos estimados: 2
Fecha inicio: 05/10/2023	Fecha finalización: 06/10/2023
Descripción: Identificar las entidades que intervienen, la determinación de sus atributos, la definición de las relaciones entre ellas y la especificación de las claves primarias y foráneas.	

Tabla 54. Actividad 25: Codificación del formulario.

Tarea	
Número: 25	Número de historia: 7
Nombre de tarea: Codificación del formulario.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 3
Fecha inicio: 07/10/2023	Fecha finalización: 09/10/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad del formulario para registrar información médica de las mascotas.	

Tabla 55. Actividad 26: Pruebas Unitarias.

Tarea	
Número: 26	Número de historia: 7
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 10/10/2023	Fecha finalización: 10/10/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

Enfermedad

Enfermedad *

Descripción

Guardar

Figura 22. Diseño preliminar de la interfaz para crear un reconocimiento y enfermedad.

Tabla 56. Historia de usuario 8: Catálogo de mascotas.

Historia de Usuario	
Número: 8	Usuario: Usuario del Aplicativo.
Nombre de historia: Catálogo de mascotas.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 6	Iteración asignada: 3
Programador responsable: Freddy Remache	
Descripción: Como cliente del aplicativo web quiero visualizar las mascotas que se encuentran disponibles en adopción para poder tomar la decisión de por cuál de las mascotas aplicar para una posible adopción.	
Criterios de Aceptación	
Criterio de aceptación 1: Dado que el usuario ingrese al catálogo y no se encuentre registrado en el aplicativo cuando presione en el botón “Adoptar” entonces el aplicativo lo dirigirá al formulario de registro.	
Criterio de aceptación 2: Dado que el usuario ingrese al catálogo y se encuentre registrado en el aplicativo cuando presione en el botón “Adoptar” entonces el aplicativo lo dirigirá al formulario de solicitud de adopción.	

Tabla 57. Actividad 27: Diseño de la interfaz del catálogo.

Tarea	
Número: 27	Número de historia: 8
Nombre de tarea: Diseño de la interfaz del catálogo.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 11/10/2023	Fecha finalización: 11/10/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica del catálogo de mascotas.	

Tabla 58. Actividad 28: Codificación del catálogo de mascotas.

Tarea	
Número: 28	Número de historia: 8
Nombre de tarea: Codificación del catálogo de mascotas.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 4
Fecha inicio: 12/10/2023	Fecha finalización: 15/10/2023

Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad del catálogo de mascotas.

Tabla 59. Actividad 29: Pruebas Unitarias.

Tarea	
Número: 29	Número de historia: 8
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 16/10/2023	Fecha finalización: 16/10/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

Tabla 60. Actividad 30: Diseño de la interfaz de usuario.

Tarea	
Número: 30	Número de historia: 9
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 17/10/2023	Fecha finalización: 17/10/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica del formulario de solicitud de adopción.	



Figura 23. Diseño preliminar del catálogo de mascotas.

Tabla 61. Historia de usuario 9: Formulario de solicitud de adopción.

Historia de Usuario

Número: 9	Usuario: Usuario del aplicativo
Nombre de historia: Formulario de solicitud de adopción.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 7	Iteración asignada: 3
Programador responsable: Freddy Remache	
Descripción: Como usuario del aplicativo web quiero tener un formulario mediante el cual pueda solicitar la adopción de una mascota para que la fundación pueda evaluar si soy o no un candidato idóneo para adoptar una mascota.	
Criterios de Aceptación	
Criterio de aceptación 1: Dado que el usuario complete los campos del formulario de manera correcta cuando presione el botón “Enviar solicitud de adopción” entonces el aplicativo registrará la solicitud de adopción.	
Criterio de aceptación 2: Dado que el usuario no haya completado todos los campos del formulario cuando presione el botón “Enviar solicitud de adopción” entonces el aplicativo emitirá un mensaje de advertencia indicando que complete todos los campos del formulario.	

Tabla 62. Actividad 31: Creación del modelo entidad relación del formulario de solicitud de adopción.

Tarea	
Número: 31	Número de historia: 9
Nombre de tarea: Creación del modelo entidad relación del formulario de solicitud de adopción.	
Tipo de tarea: Diseño del modelo entidad relación.	Puntos estimados: 1
Fecha inicio: 18/10/2023	Fecha finalización: 18/10/2023
Descripción: Identificar las entidades que intervienen, la determinación de sus atributos, la definición de las relaciones entre ellas y la especificación de las claves primarias y foráneas.	

Tabla 63. Actividad 32: Codificación del formulario.

Tarea	
Número: 32	Número de historia: 9
Nombre de tarea: Codificación del formulario.	

Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 4
Fecha inicio: 19/10/2023	Fecha finalización: 22/10/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad del formulario de solicitud de adopción.	

Tabla 64. Actividad 33: Pruebas Unitarias.

Tarea	
Número: 33	Número de historia: 9
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 23/10/2023	Fecha finalización: 23/10/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

Formulario de adopción

Situación Personal

Número de Cédula: Nombres Completos:

Dirección: Teléfono:

Email: Edad: Estado Civil:

Trabaja ?

Sí No

Nombre del sitio de trabajo: Actividad:

Ingreso mensual aproximado

0 - 200 201 - 500 501 - más otro

Convivencia Interspecifica

Gatos Perros Granja Otros

Tipo de Vivienda Casa Departamento Arrenda Propio Cerramiento de malla Cerramiento de pared

Relación con los animales

La mascota adoptada será para

Compañía Cuidar casa Juguete de niños otro

En caso de viajes largos su mascota

Viaja con usted Hospedaje Familiar u otro amigo otro

La adopción fue discutida con su familia

Sí No

Figura 24. Diseño preliminar de la interfaz del formulario de adopción.

Tabla 65. Historia de usuario 10: Interfaz para gestionar solicitudes de adopciones.

Historia de Usuario	
Número: 10	Usuario: Personal administrativo
Nombre de historia: Interfaz para gestionar solicitudes de adopciones.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 9	Iteración asignada: 3

Programador responsable: Freddy Remache
Descripción: Como personal administrativo de la fundación quiero llevar un registro y control de cada una de las solicitudes de adopción que se recibe para poder analizar los datos del posible adoptante y tomar la decisión de dar o no una mascota en adopción.
Criterios de Aceptación
Criterio de aceptación 1: Dado que el administrador decida aprobar una solicitud de adopción cuando seleccione la opción “APROBADO” entonces el aplicativo registrará la adopción.
Criterio de aceptación 2: Dado que el Administrador decida rechazar una solicitud de adopción cuando seleccione la opción “CANCELAR” entonces deberá ingresar la razón por la cual fue rechazada la solicitud.

Tabla 66. Actividad 34: Diseño de la interfaz de usuario.

Tarea	
Número: 34	Número de historia: 10
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 24/10/2023	Fecha finalización: 24/10/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica para gestionar solicitudes de adopciones.	

Tabla 67. Actividad 35: Creación del modelo entidad relación para gestionar solicitudes de adopciones.

Tarea	
Número: 35	Número de historia: 10
Nombre de tarea: Creación del modelo entidad relación para gestionar solicitudes de adopciones.	
Tipo de tarea: Diseño del modelo entidad relación.	Puntos estimados: 2
Fecha inicio: 25/10/2023	Fecha finalización: 26/10/2023

Descripción: Identificar las entidades que intervienen, la determinación de sus atributos, la definición de las relaciones entre ellas y la especificación de las claves primarias y foráneas.

Tabla 68. Actividad 36: Codificación del formulario.

Tarea	
Número: 36	Número de historia: 10
Nombre de tarea: Codificación del formulario.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 5
Fecha inicio: 27/10/2023	Fecha finalización: 31/10/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad para gestionar solicitudes de adopciones	

Tabla 69. Actividad 37: Pruebas Unitarias.

Tarea	
Número: 37	Número de historia: 10
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 01/11/2023	Fecha finalización: 01/11/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

Solicitudes

▾

Activas	Aprobadas	Canceladas	
FECHA	BENEFICIARIO	MASCOTA	ETAPA

Figura 25. Diseño preliminar de la interfaz para gestionar solicitudes de adopciones.

Tabla 70. Historia de usuario 11: Desarrollo de la notificación de resolución de solicitud de adopción.

Historia de Usuario	
Número: 11	Usuario: Personal administrativo
Nombre de historia: Desarrollo de la notificación de resolución de solicitud de adopción.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 6	Iteración asignada: 4
Programador responsable: Freddy Remache	
Descripción:	
<p>Como personal administrativo de la fundación quiero dar a conocer la aceptación o negación de una solicitud de adopción para que el usuario que ha enviado la solicitud pueda conocer la resolución.</p>	
Criterios de Aceptación	
Criterio de aceptación 1:	
<p>Dado que el usuario ha ingresado a revisar el estado de su solicitud de adopción cuando su solicitud haya sido aprobada entonces el aplicativo mostrará al usuario una notificación de que la solicitud ha sido aprobada.</p>	
Criterio de aceptación 2:	
<p>Dado que el usuario ha ingresado a revisar el estado de su solicitud de adopción cuando solicitud haya sido rechazada entonces el aplicativo mostrará al usuario un mensaje con la razón de porque fue rechazada la solicitud.</p>	

Tabla 71. Actividad 38: Diseño de la interfaz de usuario.

Tarea	
Número: 38	Número de historia: 11
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 02/11/2023	Fecha finalización: 02/11/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica de la notificación de resolución de solicitud de adopción.	

Tabla 72. Actividad 39: Codificación de la notificación de resolución de solicitud de adopción.

Tarea	
Número: 39	Número de historia: 11
Nombre de tarea: Codificación de la notificación de resolución de solicitud de adopción.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 4
Fecha inicio: 03/11/2023	Fecha finalización: 06/11/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad de la notificación de resolución de solicitud de adopción	

Tabla 73. Actividad 40: Pruebas Unitarias.

Tarea	
Número: 40	Número de historia: 11
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 07/11/2023	Fecha finalización: 07/11/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

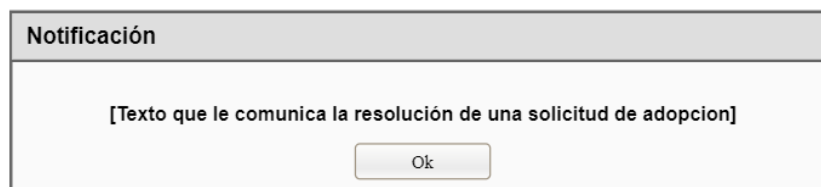


Figura 26. Diseño preliminar de la notificación de resolución de solicitud de adopción.

Tabla 74. Historia de usuario 12: Acta de adopción.

Historia de Usuario	
Número: 12	Usuario: Personal administrativo
Nombre de historia: Acta de adopción.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 7	Iteración asignada: 4
Programador responsable: Freddy Remache	

<p>Descripción:</p> <p>Como personal administrativo de la fundación quiero poder generar una acta física de adopción para posteriormente firmar el documento.</p>
Criterios de Aceptación
<p>Criterio de aceptación 1:</p> <p>Dado que el administrador ha aprobado una solicitud de adopción cuando la solicitud haya sido registrada entonces el aplicativo generará un archivo en formato PDF.</p> <p>Criterio de aceptación 2:</p> <p>Dado que el administrador ha rechazado una solicitud de adopción entonces el aplicativo no generará el archivo en formato PDF.</p>

Tabla 75. Actividad 41: Diseño del acta de adopción.

Tarea	
Número: 41	Número de historia: 12
Nombre de tarea: Diseño del acta de adopción.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 2
Fecha inicio: 08/11/2023	Fecha finalización: 09/11/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica del acta de adopción.	

Tabla 76. Actividad 42: Codificación del acta de adopción.

Tarea	
Número: 42	Número de historia: 12
Nombre de tarea: Codificación del formulario.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 4
Fecha inicio: 10/11/2023	Fecha finalización: 13/11/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad del acta de adopción.	

Tabla 77. Actividad 43: Pruebas Unitarias.

Tarea	
Número: 43	Número de historia: 12
Nombre de tarea: Pruebas Unitarias.	

Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 14/11/2023	Fecha finalización: 14/11/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

PAQUITO		ACTA DE ADOPCIÓN		Fecha elaboración:	
Elaborado por:		Revisado por:		Última aprobación:	
				Revisión:	
				Aprobado por:	
A los días del mes de de 20..... Comparecen a la suscripción de la presente acta, por una parte el albergue municipal de animales domésticos, debidamente representado por y por otra, el/la señor/a en calidad de ADOPTANTE, a quien de ahora en adelante se le conocerá como el/la ADOPTANTE de la mascota motivo de la presente acta.					
La Fundación Paquito "Rescatistas Independientes de Ambato", declara haber recogido en estado abandono un con las siguientes características:					
Raza:	Sexo:	Edad aproximada:	Tamaño:		
ID:	Rasgos característicos:				
Mismo que ha recibido los siguientes tratamientos profilácticos y/o quirúrgicos:					
Fecha:	Vacuna:	Disparatización:	Fecha de revacunación sugerida:	Esterilización:	
Y por su parte el/la señor/a con C.C. No. Domiciliado/a en de la ciudad de teléfono # declara tener condiciones suficientes para adoptar a la mascota en cuestión, por lo que La Fundación de Rescate Paquito a través del albergue de Animales Domésticos, le entrega en ADOPCIÓN, y por la presente acta el custodio se compromete a:					
<ul style="list-style-type: none"> • alimentarla adecuadamente • ubicarla en su domicilio en un espacio adecuado y abrigado • darle atención veterinaria • mantener contacto humano y cercano con ella • respetar todas sus necesidades físico-mentales. 					
El/la CUSTODIO/A se compromete además a permitir la visita de un representante del La Fundación Paquito, a fin de verificar el buen estado de la mascota y el cumplimiento del presente convenio, así como de la ordenanza municipal que regula el cuidado de mascotas.					
En caso de incumplimiento de cualquiera de las obligaciones del presente convenio, La Fundación de Rescate Paquito, queda facultado para retirar a la mascota y denunciar ante la autoridad pertinente si se hubiere cometido cualquier clase de atentado en contra de su bienestar físico o mental.					
Para constancia de todo lo expuesto, firman las partes en unidad de acto.					
Albergue Municipal de Animales Domésticos:			ADOPTANTE:		
Representante:			Sr/a		
			Cédula No.		

Figura 27. Diseño preliminar del acta de adopción.

Tabla 78. Historia de usuario 13: Interfaz de reportes.

Historia de Usuario	
Número: 13	Usuario: Personal administrativo
Nombre de historia: Interfaz de Reportes.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Alto
Puntos estimados: 6	Iteración asignada: 4
Programador responsable: Freddy Remache	
Descripción:	
Como personal administrativo de la fundación quiero conocer el número de solicitudes de adopciones recibidas, las solicitudes aprobadas por mascota, el número de actividades realizadas por la fundación para conocer estadísticamente los datos de la fundación.	
Criterios de Aceptación	

Criterio de aceptación 1:

Dado que el administrador ingrese al Dashboard **cuando** desee conocer datos estadísticos de las adopciones **entonces** el aplicativo mostrará el número total de solicitudes de adopciones, solicitudes activas, solicitudes canceladas y solicitudes aprobadas.

Criterio de aceptación 2:

Dado que el administrador ingrese al Dashboard **cuando** desee conocer datos estadísticos de las actividades de voluntariado **entonces** el aplicativo mostrará el número total de actividades, actividades realizadas y actividades sin participantes.

Tabla 79. Actividad 44: Diseño de la interfaz de usuario.

Tarea	
Número: 44	Número de historia: 13
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 15/11/2023	Fecha finalización: 15/11/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica de la interfaz de reportes.	

Tabla 80. Actividad 45: Codificación de la interfaz de reportes.

Tarea	
Número: 45	Número de historia: 13
Nombre de tarea: Codificación de la interfaz de reportes.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 4
Fecha inicio: 16/11/2023	Fecha finalización: 19/11/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad de la interfaz de reportes.	

Tabla 81. Actividad 46: Pruebas Unitarias.

Tarea	
Número: 46	Número de historia: 13
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 20/11/2023	Fecha finalización: 20/11/2023

Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.

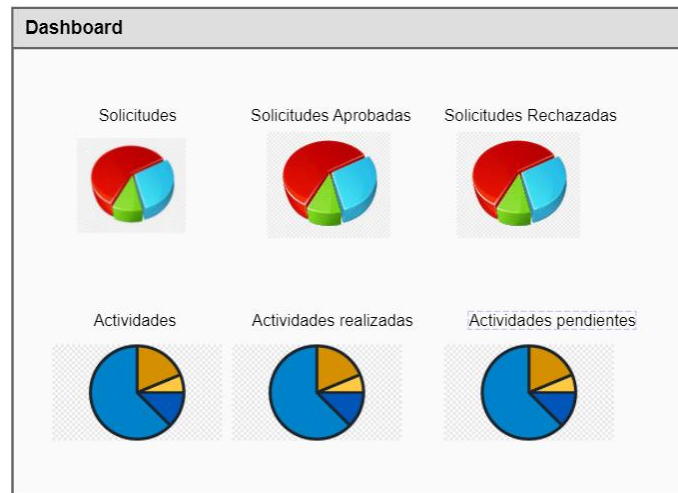


Figura 28. Diseño preliminar de la interfaz de reportes.

Tabla 82. Historia de usuario 14: Formulario para crear actividades de voluntariado.

Historia de Usuario	
Número: 14	Usuario: Personal administrativo
Nombre de historia: Formulario para crear actividades de voluntariado.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Alto
Puntos estimados: 9	Iteración asignada: 5
Programador responsable: Freddy Remache	
Descripción:	
<p>Como personal administrativo de la fundación quiero gestionar la información de las diferentes actividades que realiza la fundación para incitar a las personas a ayudar en algunas de las actividades.</p>	
Criterios de Aceptación	
Criterio de aceptación 1:	
<p>Dado que el administrador complete los campos del formulario para registrar información de una actividad cuando presione el botón Crear Actividad entonces el aplicativo notificará que la actividad ha sido creada correctamente.</p>	
Criterio de aceptación 2:	

Dado que el administrador no haya completado todos los campos del formulario **cuando** presione el botón Crear Actividad **entonces** el aplicativo emitirá un mensaje de advertencia indicando que complete todos los campos del formulario.

Tabla 83. Actividad 47: Diseño de la interfaz de usuario.

Tarea	
Número: 47	Número de historia: 14
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 21/11/2023	Fecha finalización: 21/11/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica del formulario para crear actividades de voluntariado.	

Tabla 84. Actividad 48: Creación del modelo entidad relación del formulario para crear actividades de voluntariado.

Tarea	
Número: 48	Número de historia: 14
Nombre de tarea: Creación del modelo entidad relación del formulario para crear actividades de voluntariado.	
Tipo de tarea: Diseño del modelo entidad relación.	Puntos estimados: 2
Fecha inicio: 22/11/2023	Fecha finalización: 23/11/2023
Descripción: Identificar las entidades que intervienen, la determinación de sus atributos, la definición de las relaciones entre ellas y la especificación de las claves primarias y foráneas.	

Tabla 85. Actividad 49: Codificación del formulario.

Tarea	
Número: 49	Número de historia: 14
Nombre de tarea: Codificación del formulario.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 5
Fecha inicio: 24/11/2023	Fecha finalización: 28/11/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad del formulario para crear actividades de voluntariado.	

Tabla 86. Actividad 50: Pruebas Unitarias.

Tarea	
Número: 50	Número de historia: 14
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 29/11/2023	Fecha finalización: 29/11/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

The image shows a web form titled "Nueva Actividad". It contains the following elements from top to bottom: a "Tipo" dropdown menu; a text input field for "Actividad"; a "Cuando:" label; a calendar widget for August 2016; two dropdown menus for "Desde" and "Hasta"; and a "Crear Actividad" button.

Figura 29. Diseño preliminar de la interfaz del formulario para agregar una actividad.

Tabla 87. Historia de usuario 15: Catálogo de actividades para voluntariado.

Historia de Usuario	
Número: 15	Usuario: Cliente
Nombre de historia: Catálogo de actividades para voluntariado.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Alto
Puntos estimados: 6	Iteración asignada: 5
Programador responsable: Freddy Remache	
Descripción: Como cliente del aplicativo web quiero conocer las actividades que se llevan a cabo dentro de la fundación para participar como voluntario.	
Criterios de Aceptación	

<p>Criterio de aceptación 1:</p> <p>Dado que el usuario se encuentre registrado en el aplicativo cuando presione en el botón “Participar” en una actividad para realizar en voluntariado entonces el aplicativo registrará su participación.</p> <p>Criterio de aceptación 2:</p> <p>Dado que el usuario desee participar en una actividad de voluntariado cuando no se encuentre registrado en el aplicativo entonces el aplicativo lo dirigirá al formulario de registro.</p>

Tabla 88. Actividad 51: Diseño de la interfaz de usuario.

Tarea	
Número: 51	Número de historia: 15
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 3
Fecha inicio: 30/11/2023	Fecha finalización: 30/11/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica del catálogo de actividades para voluntariado.	

Tabla 89. Actividad 52: Codificación del catálogo de actividades para voluntariado.

Tarea	
Número: 52	Número de historia: 15
Nombre de tarea: Codificación del catálogo de actividades para voluntariado.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 4
Fecha inicio: 01/12/2023	Fecha finalización: 04/12/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad del catálogo de actividades para voluntariado.	

Tabla 90. Actividad 53: Pruebas Unitarias.

Tarea	
Número: 53	Número de historia: 15
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 05/12/2023	Fecha finalización: 05/12/2023

Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.

Actividades para Voluntariado				
ACTIVIDAD	DESCRIPCIÓN	FECHA	HORARIO	
				Participar
				Participar
				Participar
				Participar

Figura 30. Diseño preliminar del catálogo de actividades para voluntariado.

Tabla 91. Historia de usuario 16: Automatización de recepción de donaciones de dinero.

Historia de Usuario	
Número: 16	Usuario: Personal administrativo
Nombre de historia: Automatización de recepción de donaciones de dinero.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Alto
Puntos estimados: 6	Iteración asignada: 5
Programador responsable: Freddy Remache	
Descripción:	
Como personal administrativo de la fundación quiero que la recolección de donaciones de dinero se automatice para que el dinero se vaya directamente a nuestra cuenta de PayPal.	
Criterios de Aceptación	
Criterio de aceptación 1:	
Dado que el usuario haya ingresado un valor de dinero para donar cuando presione en el botón “Donar con PayPal” entonces el aplicativo lo dirigirá hacia el Login de su cuenta de PayPal.	
Criterio de aceptación 2:	

Dado que el usuario no haya ingresado un valor de dinero para donar **cuando** presione en el botón “Donar con PayPal” **entonces** el aplicativo le mostrara un mensaje de advertencia donde se le indica que debe ingresar un valor.

Tabla 92. Actividad 54: Diseño de la interfaz de usuario.

Tarea	
Número: 54	Número de historia: 16
Nombre de tarea: Diseño de la interfaz de usuario.	
Tipo de tarea: Diseño de interfaz.	Puntos estimados: 1
Fecha inicio: 06/12/2023	Fecha finalización: 06/12/2023
Descripción: Diseñar y describir cómo se estructura y presenta la interfaz gráfica de la recepción de donaciones de dinero con PayPal.	

Tabla 93. Actividad 55: Codificación de la recepción de donaciones de dinero con PayPal.

Tarea	
Número: 55	Número de historia: 16
Nombre de tarea: Codificación de la recepción de donaciones de dinero con PayPal.	
Tipo de tarea: Desarrollo Frontend y Backend.	Puntos estimados: 4
Fecha inicio: 07/12/2023	Fecha finalización: 07/12/2023
Descripción: Creación de código, siguiendo la estructura y la sintaxis requeridas para lograr la funcionalidad de la recepción de donaciones de dinero con PayPal.	

Tabla 94. Actividad 56: Pruebas Unitarias.

Tarea	
Número: 56	Número de historia: 16
Nombre de tarea: Pruebas Unitarias.	
Tipo de tarea: Pruebas de funcionamiento.	Puntos estimados: 1
Fecha inicio: 08/12/2023	Fecha finalización: 08/12/2023
Descripción: Verificar que cada bloque de código funcione correctamente según lo esperado.	

Se ha decidido utilizar PayPal para automatizar las donaciones de dinero porque es un servicio gratuito y seguro de pago online que permite a particulares y empresas realizar transferencias electrónicas de dinero.

Figura 31. Diseño preliminar de la interfaz para realizar donaciones.

3.2.1.4 Valoración de historias de usuario

A partir del análisis de los requerimientos de la propuesta por medio de las historias de usuario, se procede con la estimación del tiempo de desarrollo de cada módulo del sistema. Para esto, se tomará en consideración el siguiente aspecto: 1 punto estimado equivale a 8 horas diarias.

En la Tabla 95 se realiza la estimación de tiempo de cada historia de usuario:

Tabla 95. Estimación de tiempo de historias de usuario.

N°	Historia de Usuario	Tareas	Puntos estimados	Horas	Total
1	Creación de la estructura del aplicativo web.	Diseño conceptual de la base de datos del aplicativo web.	1	8	24
		Creación del proyecto en backend.	1	8	
		Creación del proyecto en frontend.	1	8	
2		Diseño de la interfaz de usuario.	1	8	64

	Formulario de registro.	Creación del modelo entidad relación del formulario de registro	2	16	
		Codificación del formulario de registro.	4	32	
		Pruebas Unitarias.	1	8	
3	Interfaz de autenticación y autorización	Diseño de la interfaz de usuario.	1	8	64
		Creación del modelo entidad relación de la interfaz de autenticación y autorización.	2	16	
		Codificación de la interfaz de autenticación y autorización.	4	32	
		Pruebas Unitarias.	1	8	
4	Interfaz para actualizar información del usuario	Diseño de la interfaz de usuario.	1	8	48
		Creación del modelo entidad relación para actualizar información del usuario.	1	8	
		Codificación de la interfaz para actualizar información del usuario.	3	24	
		Pruebas Unitarias.	1	8	
5	Interfaz para la asignación de perfiles de usuario.	Diseño de la interfaz de usuario.	1	8	48
		Codificación de la interfaz para la asignación de perfiles de usuario.	4	32	
		Pruebas Unitarias.	1	8	
6	Interfaz para gestionar información de las mascotas	Diseño de la interfaz de usuario.	1	8	64
		Creación del modelo entidad relación para la gestión de la información de las mascotas.	2	16	
		Codificación de la interfaz.	4	32	
		Pruebas Unitarias.	1	8	
7	Formulario para registrar información médica de las mascotas	Diseño de la interfaz de usuario.	1	8	56
		Creación del modelo entidad relación para la gestión de la información de las mascotas.	2	16	
		Codificación del formulario.	3	24	
		Pruebas Unitarias.	1	8	
8	Catálogo de mascotas.	Diseño de la interfaz de usuario.	1	8	48
		Codificación del catálogo de mascotas.	4	32	
		Pruebas Unitarias.	1	8	

9	Formulario de solicitud de adopción.	Diseño de la interfaz de usuario.	1	8	56
		Creación del modelo entidad relación del formulario de solicitud de adopción.	1	8	
		Codificación del formulario.	4	32	
		Pruebas Unitarias.	1	8	
10	Interfaz para gestionar solicitudes de adopciones.	Diseño de la interfaz de usuario.	1	8	72
		Creación del modelo entidad relación para gestionar solicitudes de adopciones.	2	16	
		Codificación de la interfaz.	5	40	
		Pruebas Unitarias.	1	8	
11	Desarrollo de la notificación de resolución de solicitud de adopción	Diseño de la interfaz de usuario.	1	8	48
		Codificación de la notificación de resolución de solicitud de adopción.	4	32	
		Pruebas Unitarias.	1	8	
12	Generar acta de adopción	Diseño del acta de adopción.	2	16	56
		Codificación del acta de adopción.	4	32	
		Pruebas Unitarias.	1	8	
13	Interfaz de reportes	Diseño de la interfaz de usuario.	1	8	48
		Codificación de la interfaz de reportes.	4	32	
		Pruebas Unitarias.	1	8	
14	Formulario para crear actividades de voluntariado.	Diseño de la interfaz de usuario.	1	8	72
		Creación del modelo entidad relación del formulario para crear actividades de voluntariado.	2	16	
		Codificación del formulario.	5	40	
		Pruebas Unitarias.	1	8	
15	Catálogo de actividades para voluntariado.	Diseño de la interfaz de usuario.	1	8	48
		Codificación del catálogo de actividades para voluntariado.	4	32	
		Pruebas Unitarias.	1	8	
16	Automatización de recepción de donaciones de dinero.	Diseño de la interfaz de usuario.	1	8	48
		Codificación de la recepción de donaciones de dinero con PayPal.	4	32	
		Pruebas Unitarias.	1	8	

Total	108	824	864
--------------	-----	-----	-----

3.2.1.5 Plan de entregas

Una vez que se ha realizado la estimación de las historias de usuario, se elabora el plan de entrega para la elaboración del proyecto detallando su iteración.

Tabla 96. Plan de entregas del aplicativo web.

N°	Historia de Usuario	Tiempo Estimado		Iteración Asignada				
		Días	Horas	1	2	3	4	5
1	Creación de la estructura del aplicativo web.	3	24	X				
2	Formulario de registro.	8	64	X				
3	Interfaz de autenticación y autorización.	8	64	X				
4	Interfaz para actualizar información del usuario.	6	48	X				
5	Interfaz para asignar perfiles de usuario.	6	48		X			
6	Interfaz para gestionar información de las mascotas.	8	64		X			
7	Formulario para registrar información médica de las mascotas.	7	56		X			
8	Catálogo de mascotas.	6	48			X		
9	Formulario de solicitud de adopción.	7	56			X		
10	Interfaz para gestionar solicitudes de adopciones.	9	72			X		
11	Desarrollo de la notificación de resolución de solicitud de adopción.	6	48				X	
12	Acta de adopción	7	56				X	
13	Interfaz de reportes.	6	48				X	
14	Formulario para crear actividades de voluntariado.	9	72					X
15	Catálogo de actividades para voluntariado.	6	48					X
16	Automatización de recepción de donaciones de dinero.	6	48					X

3.2.2 Fase 2: Diseño

Durante esta fase, se trabaja en incrementos pequeños y manejables del software, conocidos como "iteraciones" o "ciclos".

3.2.2.1. Iteraciones por entregas

El desarrollo del aplicativo web se lleva a cabo en ciclos repetitivos, y al final de cada ciclo se produce una entrega o versión funcional del producto.

3.2.2.1.1. Iteración N^a 1

En la Tabla 97 se describen las historias de usuario que deben cumplirse en el primer periodo de entrega.

Tabla 97. Historias de usuario de Iteración N^o 1.

N ^o	Historia de Usuario	Prioridad en negocio	Riesgo de desarrollo
1	Creación de la estructura del aplicativo web.	Alta	Alto
2	Formulario de registro.	Alta	Alto
3	Interfaz de autenticación y autorización.	Alta	Alto
4	Interfaz para actualizar información del usuario.	Alta	Alto

Desarrollo de la historia de usuario N^o 1: Creación de la estructura del aplicativo web.

En la Figura 32 se visualiza el diseño de la base de datos que consta de la ilustración de las entidades y campos que conforman la base de datos con sus respectivas relaciones. Para este diseño se utilizó el gestor de base de datos MySQL.

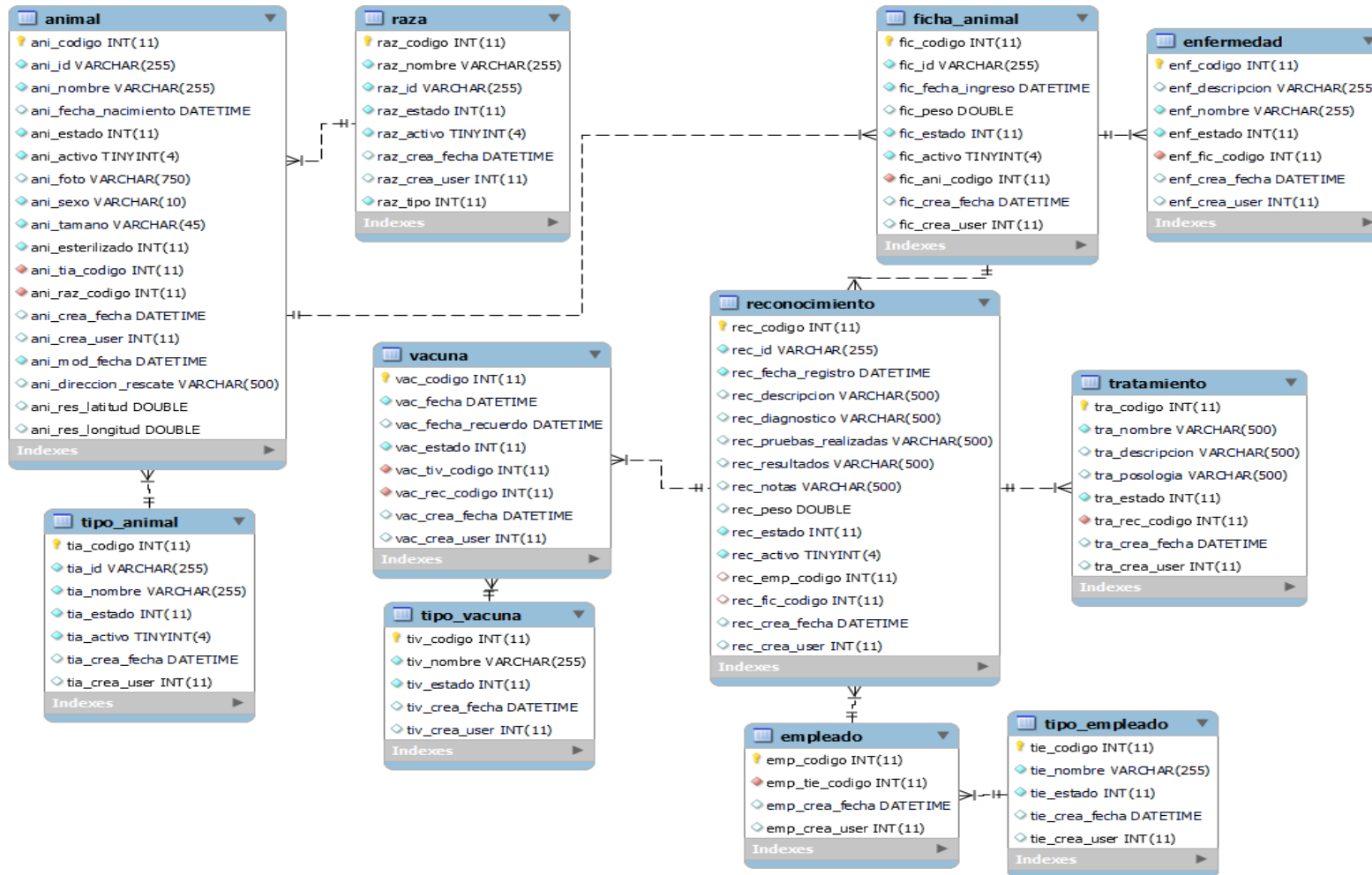


Figura 32. Diagrama entidad-relación rescate de mascotas.

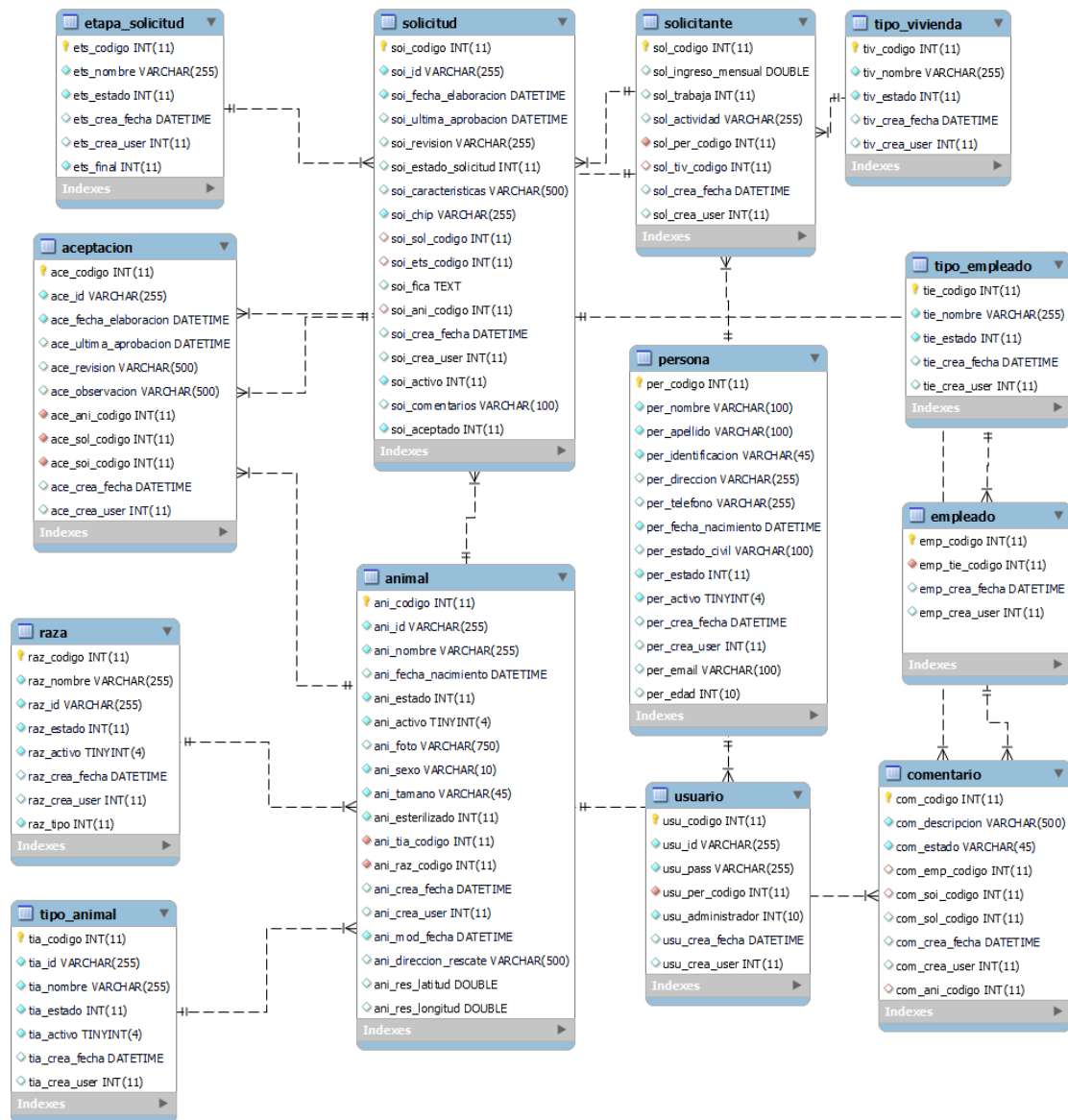


Figura 33. Diagrama entidad-relación adopciones.

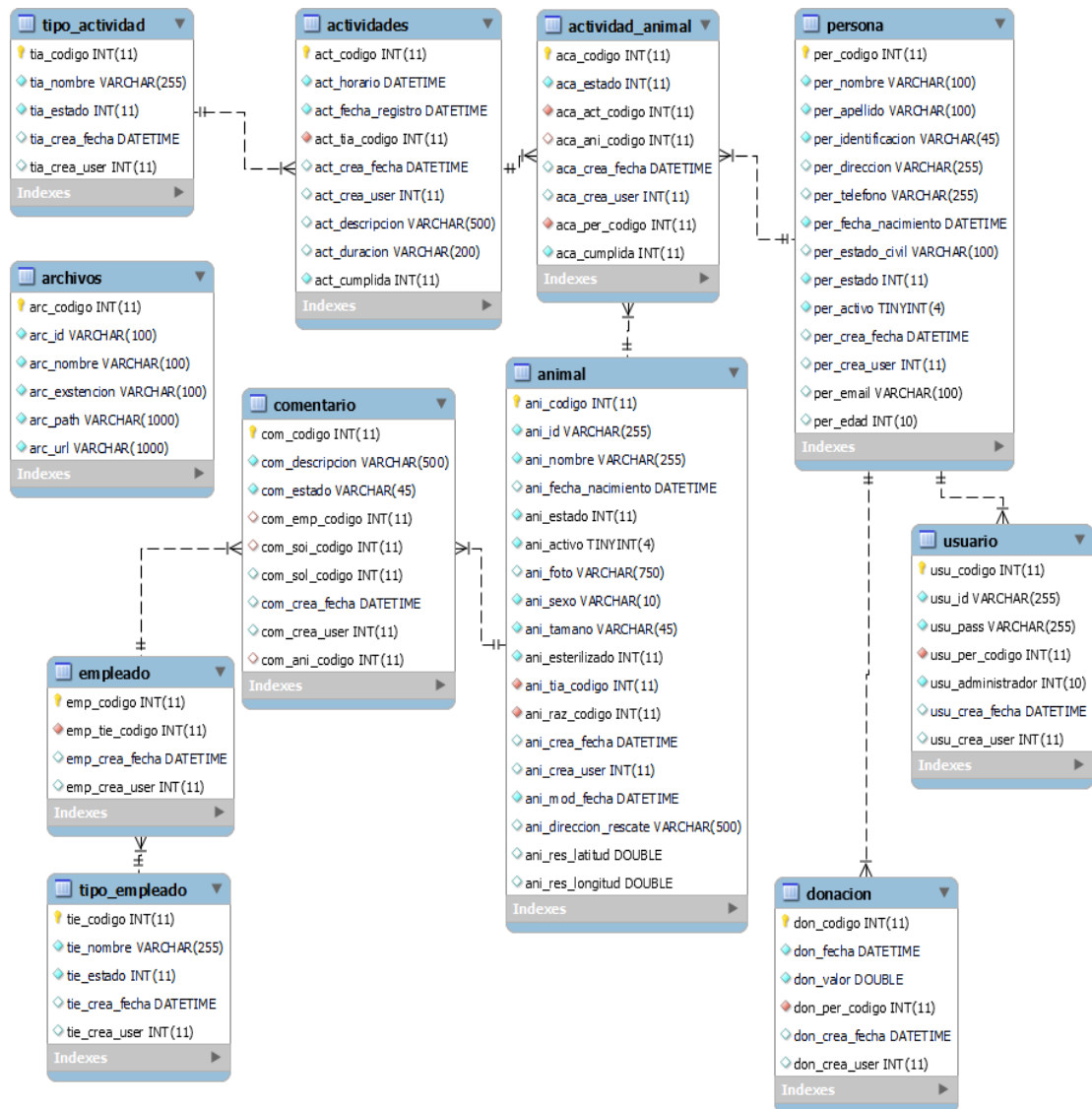


Figura 34. Diagrama entidad-relación voluntariado.

Desarrollo de la historia de usuario N° 2: Formulario de registro.

En la *Figura 35* se observa el formulario de registro que permite a los usuarios registrar su información para acceder a las opciones de adoptar una mascota o realizar voluntariado.

Regístrate

y adopta a tu primera mascota



Figura 35. Formulario para el registro de usuarios.

Cuando los datos son completados correctamente aparecerá el mensaje que se visualiza en la Figura 36.

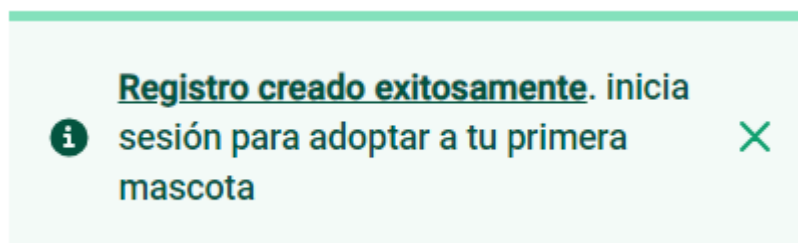


Figura 36. Ventana modal para el mensaje de confirmación.

Desarrollo de la historia de usuario N° 3: Interfaz de autenticación y autorización.

En la Figura 37 observamos la interfaz para ingresar al aplicativo web se debe autenticar los datos de usuario y contraseña en el Login. Una vez ingresado las credenciales correctamente se ingresará a los módulos del sistema.

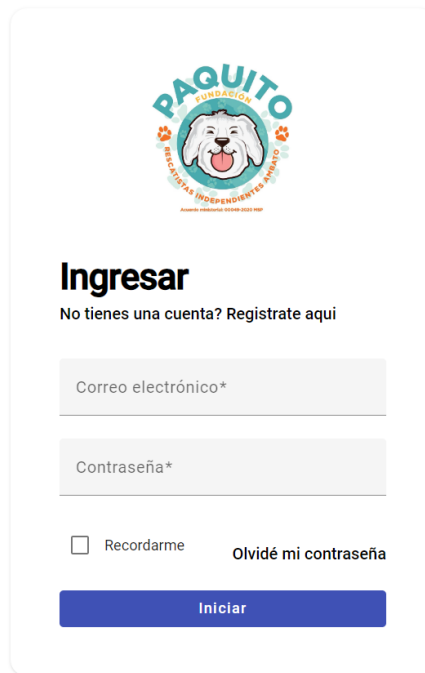


Figura 37. Login.

En la Figura 38 se visualiza la página principal del aplicativo web.

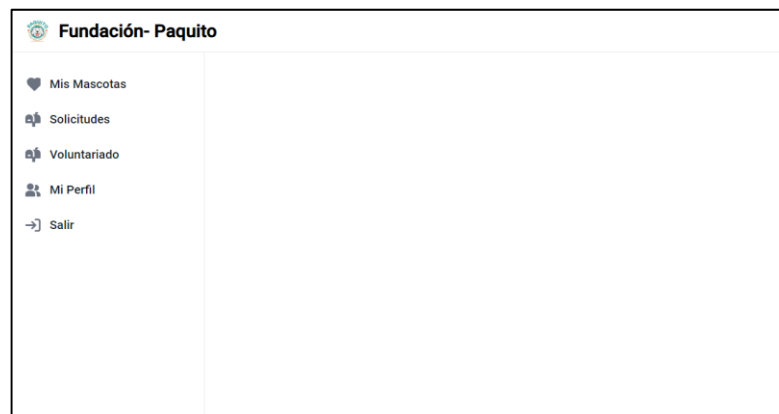


Figura 38. Página principal.

Desarrollo de la historia de usuario N° 4: Interfaz para actualizar información del usuario.

En la Figura 39 podemos visualizar la interfaz de usuarios, se constituye de una sección, donde: consiste en una lista de los usuarios existentes mediante una tabla, en la que se puede filtrar los usuarios y tiene la acción de editar.

Usuarios

CODIGO	ID	NOMBRES	CORREO	CREADO	TIPO
21	marcosvargas@gmail.com	Marcos Vargas	marcosvargas@gmail.com	11 ene 2024	Usuario
20	juanreyes22@gmail.com	Juan Reyes	juanreyes22@gmail.com	9 ene 2024	Usuario
19	juanaperez@gmail.com	Juana Perez	juanaperez@gmail.com	3 ene 2024	Usuario
18	isabeltorres@hotmail.com	Isabel Torres	isabeltorres@hotmail.com	1 ene 2024	Usuario
17	veronica@gmail.com	Veronica Maldonado	veronica@gmail.com	24 oct 2023	Usuario

Figura 39. Tabla de los usuarios existentes.

El formulario para editar la información de los usuarios se observa en la Figura 40.

marcosvargas@gmail.com

Estado civil
Divorciado

Fecha de nacimiento

Día: 4 Mes: Marzo Año: 1903

Dirección
San Blas

Guardar

Figura 40. Formulario para editar información de los usuarios.

3.2.2.1.2. Iteración N^a 2

En la Tabla 98 se describen las historias de usuario que deben cumplirse en el segundo periodo de entrega.

Tabla 98. Historias de usuario de Iteración N^o 2.

N ^o	Historia de Usuario	Prioridad en negocio	Riesgo de desarrollo
----------------	---------------------	----------------------	----------------------

5	Interfaz para asignar perfiles de usuario.	Alta	Alto
6	Interfaz para gestionar información de las mascotas.	Alta	Alto
7	Formulario para registrar información médica de las mascotas.	Alta	Alto

Desarrollo de la historia de usuario N° 5: Interfaz para asignar perfiles de usuario.

En la Figura 41 se representa la interfaz para asignar de perfiles de usuario a cada usuario que se encuentre registrado en el aplicativo web. Los roles de usuario que se pueden asignar es el rol de Administrador que puede usar todas las funciones del aplicativo incluidas las configuraciones y el rol de Usuario que únicamente puede usar las funciones básicas del aplicativo web.

The image shows a modal window with the title "Tipos de usuario". It contains two radio button options: "Administrador" (unselected) and "Usuario Normal" (selected). Below the options is a button labeled "Cambiar el tipo de usuario".

Figura 41. Ventana modal para asignar un rol de usuario.

Desarrollo de la historia de usuario N° 6: Interfaz para gestionar información de las mascotas.

En la Figura 42 se visualiza el módulo de gestión de mascotas, se constituye de una sección que consiste en una lista de las mascotas existentes mediante una tabla, en la que se puede filtrar las mascotas y tiene la acción de crear y editar.

♥ Mis mascotas Crear una mascota

TIPO	NOMBRE	RAZA	ESTADO
Perro	Guido	Balkan Hound	Adoptado
Gato	Hugo	Korat	Adoptado
Perro	Lazy	Australian silky terrier	Adoptado
Gato	Loba	Abisinio	Adoptado
Perro	Bobby	Ariegeois	Adoptado
Gato	Ruby	Abisinio	Adoptado
Perro	waco	Pastor Alemán	
Gato	Toby	Africano doméstico	
Gato	Leo3435	Abisinio	Adoptado

Figura 42. Tabla de las mascotas existentes.

La interfaz para ingresar datos de una mascota se presenta en la Figura 43.

Crear Mascota

Nombre de la mascota

Tipo de mascota

Raza

Sexo

Tamaño

Fecha de nacimiento

Figura 43. Formulario para ingresar datos de una mascota.

En la Figura 44 se observa la sección para agregar fotos de las mascotas.

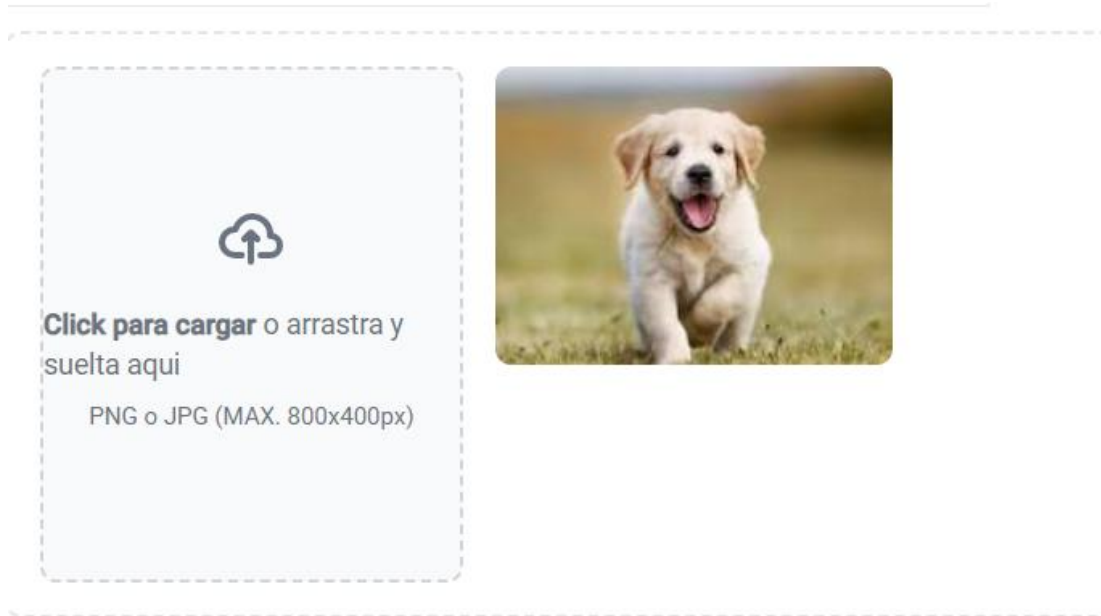


Figura 44. Sección para agregar fotos de las mascotas.

Mediante el formulario que se presenta en la Figura 45 podemos editar información de las mascotas.

Un formulario de edición de datos para una mascota. Incluye un menú desplegable para "Tamaño" con "Mediano" seleccionado. Una sección "Fecha de nacimiento" con campos para "Día" (24), "Mes" (Julio) y "Año" (2016). Un campo de texto para "Dirección" con "Cevallos" ingresado. Una casilla de verificación marcada para "Esterilizado". Un botón rojo "Guardar Mascota" al final.

Figura 45. Formulario para editar información de las mascotas.

Desarrollo de la historia de usuario N° 7: Formulario para registrar información médica de las mascotas.

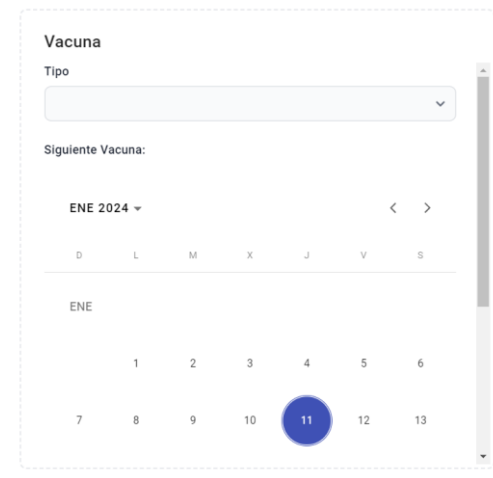
En la Figura 46 se observa el formulario para registrar información médica de las mascotas, se constituye de una sección que tiene la acción de crear y editar información médica como: enfermedades y alergias de cada una de las mascotas.



The image shows a web form titled "Nuevo Reconocimiento" (New Recognition). It contains five input fields, each with a label above it: "Peso (kg)", "Descripción del reconocimiento", "Diagnostico", "Pruebas realizadas", and "Resultados". The form is enclosed in a dashed border and has a vertical scrollbar on the right side.

Figura 46. Formulario para agregar información del Reconocimiento de la mascota.

En la Figura 47 se visualiza el formulario para agregar vacunas.



The image shows a web form titled "Vacuna" (Vaccine). It includes a dropdown menu for "Tipo" (Type). Below it is a date picker labeled "Siguiete Vacuna:" (Next Vaccine:) showing "ENE 2024". Underneath is a calendar grid for the month of "ENE" (January) with days 1 through 13. The date "11" is highlighted with a blue circle. There are navigation arrows (< and >) above the calendar grid.

Figura 47. Formulario para agregar vacunas.

En la Figura 48 se presenta un formulario para agregar reconocimientos de las mascotas.

Nuevo Tratamiento

Nombre

Descripción del tratamiento

Diagnostico

Guardar Tratamiento

Figura 48. Formulario para agregar reconocimientos de las mascotas.

En la Figura 49 se visualiza el formulario para agregar enfermedades o alergias.

Enfermedad

Nombre

Descripción

Escribe aquí una descripción de la enfermedad...

Crear enfermedad

Figura 49. Formulario para agregar enfermedad o alergia.

3.2.2.1.3. Iteración N^a 3

En la

Tabla 99 se describen las historias de usuario que deben cumplirse en el tercer periodo de entrega.

Tabla 99. Historias de usuario de Iteración N° 3.

N°	Historia de Usuario	Prioridad en negocio	Riesgo de desarrollo
8	Catálogo de mascotas.	Alta	Alto
9	Formulario de solicitud de adopción.	Alta	Alto
10	Interfaz para gestionar solicitudes de adopciones.	Alta	Alto

Desarrollo de la historia de usuario N° 8: Catálogo de mascotas.

En la *Figura 50* se observa el catálogo de mascotas que permite al usuario visualizar las mascotas que están registradas en la fundación y disponibles para ser adoptadas.

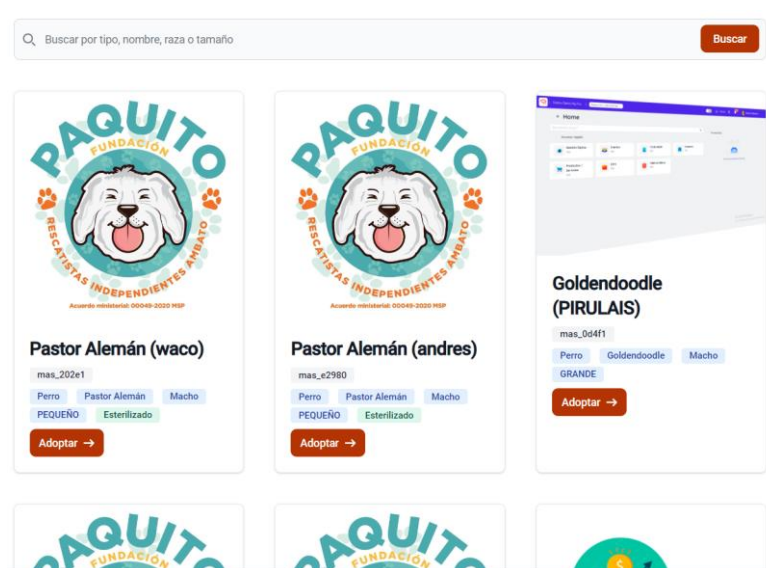


Figura 50. Catálogo de mascotas.

Desarrollo de la historia de usuario N° 9: Formulario de solicitud de adopción.

En la *Figura 51* se visualiza el formulario de solicitud de adopción que permite al usuario ingresar los datos necesarios para solicitar una mascota en adopción. Del mismo modo, se podrá generar una vez lleno el formulario un formato pdf en base una plantilla establecida por la fundación.

PASTOR ALEMÁN (waco)
 Perro Macho PEQUEÑO Esterilizado
 Cambiar →

Situación Personal

Número de Cédula: 0104358839
 Nombres Completos: Admin Admin

Dirección: Cuenca
 Teléfono: 0998763543

Email: pablo.ab34@gmail.com
 Edad: 0
 Estado Civil: Soltero/a

¿Trabaja?
 SI
 NO

Nombre del sitio de trabajo: _____
 Actividad: _____

Figura 51. Formulario de solicitud de adopción.

En la *Figura 52* se observa el formato PDF del formulario de solicitud de adopción.

PASTOR ALEMÁN (waco)
 SOLICITUD DE ADOPCION sol_76aoc
 Fecha elaboración: viernes, 8 de septiembre de 2023
 Última aprobación: viernes, 8 de septiembre de 2023
 Revisión: viernes, 8 de septiembre de 2023

Elaborado por: _____ Revisado Por: _____ Aprobado por: _____

Fecha: viernes, 8 de septiembre de 2023
 Mascota Escogida: Perro
 Código/Chip: _____ Negado: _____
 Características: Pastor Alemán - MEDIANO - M Aprobado: _____

Nuestro principal compromiso es asegurar que la mascota a ser adoptada llegue a un hogar que presente las mejores condiciones en cuanto a su bienestar.
 Esta solicitud será aprobada o rechazada de manera inmediata y en casos necesarios en un plazo máximo de 48 horas. Luego de ello nosotros nos contactamos con usted.
 La fundación paquito tiene el derecho absoluto sobre los animales y se reserva el derecho de proporcionar información sobre la aprobación o rechazo de la solicitud.
 Todas las mascotas entregadas en adopción, LA Fundación Paquito lo harán previamente esterilizadas respetando la normativa legal vigente, política interna y como parte de un manejo responsable de mascotas para evitar su reproducción.

USTED DESEA ADOPTAR

CANINO		FELINO	
Macho	<input checked="" type="checkbox"/> Cachorro	Macho	<input type="checkbox"/> Cachorro
Hembra	<input type="checkbox"/> Adulto	Hembra	<input type="checkbox"/> Adulto
Tamaño	MEDIANO	Tamaño	

SITUACION PERSONAL

Nombre completo: Admin Admin CC: 0104358839
 Dirección: cda uncovia Teléfono: 0998763543
 Email: pablo.ab34@gmail.com Edad: 30 Estado civil: Soltero

¿Trabaja? Si Nombre del sitio de trabajo gvastudios Actividad programador

Ingreso mensual aproximado(Opcional)	Personas con las que vive (Opcional)	Parentesco	Convivencia Interspecifica
0 - 200			Perros
201 - 500	<input checked="" type="checkbox"/>		Gatos
501 - mas			Granjas
Otro			Otros

Tipo de vivienda: _____ Permanencia de la mascota: _____

Figura 52. Formato PDF del formulario de solicitud de adopción.

Desarrollo de la historia de usuario N° 10: Interfaz para gestionar solicitudes de adopciones.

En la *Figura 53* se presenta el módulo de gestión de solicitudes de adopciones, el cual se constituye de una sección, donde: consiste en una lista de solicitudes de adopciones existentes mediante una tabla, en la que se puede filtrar las solicitudes y tiene la acción de Aprobar o Cancelar una solicitud.

Solicitudes de Adopción Crear una solicitud de Adopción

Activas Aprobadas Canceladas

FECHA	BENEFICIARIO	MASCOTA	ETAPA
8 sept 2023, 20:15:27	Admin Admin	Tomás - Pastor Alemán	RECIBIDO
8 sept 2023, 19:59:32	Admin Admin	Angelito - Pastor Alemán	RECIBIDO
6 sept 2023, 18:37:22	Admin Admin	Lucesita - Goldendoodle	RECIBIDO
6 sept 2023, 18:37:07	Admin Admin	Tomás - Pastor Alemán	RECIBIDO
6 sept 2023, 17:51:06	Admin Admin	Roco - Mixta	PREAPROBADO

Figura 53. Tabla de solicitudes de adopción.

3.2.4.1.4. Iteración N^a 4

En la Tabla 100 se describen las historias de usuario que deben cumplirse en el cuarto periodo de entrega.

Tabla 100. Historias de usuario de Iteración N^o 4.

N ^o	Historia de Usuario	Prioridad en negocio	Riesgo de desarrollo
11	Desarrollo de la notificación de resolución de solicitud de adopción.	Alta	Alto
12	Acta de adopción	Alta	Alto
13	Interfaz de reportes.	Alta	Alto

Desarrollo de la historia de usuario N^o 11: Desarrollo de la notificación de resolución de solicitud de adopción.

En la *Figura 54* se visualiza la notificación con la cual el usuario será notificado de la resolución de una solicitud ya sea Aprobada o Cancelada. En caso de ser Cancelada también se le enviará el motivo por el cual ha sido cancelada su solicitud.

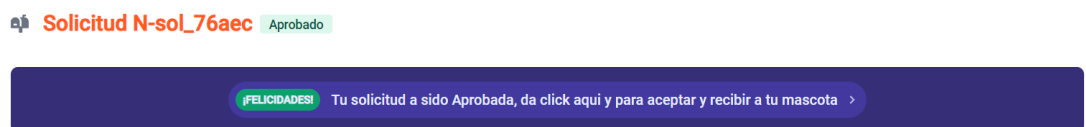


Figura 54. Mensaje de resolución de solicitud de adopción.

Desarrollo de la historia de usuario N° 12: Acta de adopción

Si una solicitud de adopción ha sido aprobada se generará automáticamente el Acta de adopción en formato PDF según plantilla proporcionada por la fundación, la cual se observa en la *Figura 55*.

PAQUITO		ACTA DE ADOPCION		Fecha elaboración: martes, 5 de septiembre de 2023	
				Última aprobación martes, 5 de septiembre de 2023	
				Revisión: martes, 5 de septiembre de 2023	
Elaborado por:		Revisado Por:		Aprobado por:	

A los 11 días del mes enero del 2024 Comparecen a la suscripción de la presente acta, por parte el albergue municipal de animales domésticos, debidamente representado por Fundación Paquito y por otra, el/la señor/a Admin Admin en calidad de ADOPTANTE a quien de ahora en adelante se le conocerá como el/la ADOPTANTE de la mascota motivo de la presente acta.

La Fundación Paquito "Rescatistas Independientes de Ambato", declara haber recogido en estado de abandono un Perro con las siguientes características:

Raza:	Sexo:	Edad aproximada:	Tamaño:
Pastor Alemán	M	26 Meses	MEDIANO

ID: pe_2d4f69h Rasgos característicos: -

Mismo que a recibido los siguientes tratamientos profilácticos y/o quirúrgicos

Fecha	Vacuna	Desparasitación	Fecha de revacunación sugerida	Esterilización

Y por su parte el/la señor/a Admin Admin con CC: 0104358839 Domiciliado en ciudadela uncovia de la ciudad de ciudadela uncovia teléfono 0998763543 declara tener las condiciones suficientes para adoptar a la mascota en cuestión, por lo que la Fundación de rescate Paquito a través del albergue de animales domésticos, le entrega en ADOPCION, y por la presente acta de custodia se compromete a:

- Alimentarla adecuadamente
- Ubicarla en su domicilio en un espacio adecuado y abrigado
- Darle atención veterinaria
- Mantener contacto humano y cercano con ella
- Respetar todas sus necesidades físico-mentales

El/La CUSTODIO/A se compromete además a permitir la visita de un representante de la Fundación Paquito, a fin de verificar el buen estado de la mascota y el cumplimiento del presente convenio, así como de la ordenanza municipal que regula el cuidado de las mascotas.

En caso de incumplimiento de cualquiera de las obligaciones del presente convenio, la Fundación de rescate

Figura 55. Acta de adopción en formato PDF.

Desarrollo de la historia de usuario N° 13: Interfaz de reportes.

En la *Figura 56* se visualiza la interfaz de reportes que permitirá al administrador visualizar datos estadísticos donde se podrán visualizar un resumen de las adopciones y de voluntariado.



Figura 56. Acta de adopción en formato PDF.

3.2.2.1.5. Iteración Nª 5

En la Tabla 101 se describen las historias de usuario que deben cumplirse en el cuarto periodo de entrega.

Tabla 101. Historias de usuario de Iteración N° 5.

N°	Historia de Usuario	Prioridad en negocio	Riesgo de desarrollo
14	Formulario para crear actividades de voluntariado.	Alta	Alto
15	Catálogo de actividades para voluntariado.	Alta	Alto
16	Automatización de recepción de donaciones de dinero.	Alta	Alto

Desarrollo de la historia de usuario N° 14: Formulario para crear actividades de voluntariado.

En la Figura 57 se observa el módulo de gestión de voluntariado, se constituye de una sección, donde: consiste en una lista de actividades para voluntariado existentes mediante una tabla, en la que se puede filtrar las actividades y tiene la acción Crear una actividad para voluntariado.

FECHA	TIPO	ACTIVIDAD	DURACIÓN	PARTICIPANTES	ACCIÓN
9 ene 2024	Paseos, juegos y socialización	Paseo de cachorros	10/1/24, 7:30 10/1/24, 8:30 1 horas	1 Personas	Realizada
3 ene 2024	Limpieza de áreas comunales	Limpieza de las casa de perros	9/1/24, 7:30 9/1/24, 8:30 1 horas	2 Personas	Participar
3 ene 2024	Juegos y acompañamiento a gatos	Jugar con gatos	11/1/24, 8:15 11/1/24, 9:30 1 horas	1 Personas	Participar
30 nov 2023	Limpieza de áreas comunales	Nueva	1 horas	0 Personas	Participar
29 nov 2023	Caminata dirigida	caminar con perros	2 horas	1 Personas	Participar

Figura 57. Tabla de actividades disponibles para voluntariado.

En la Figura 58 se presenta la interfaz para agregar una nueva actividad para voluntariado.

Nueva Actividad

Tipo:

Actividad:

Cuando:

ENE 2024 < >

D L M X J V S

ENE

1 2 3 4 5 6

Figura 58. Formulario para agregar una nueva actividad.

Desarrollo de la historia de usuario N° 15: Catálogo de actividades para voluntariado.

En la Figura 59 se observa el catálogo de actividades que permite al usuario visualizar las actividades que están registradas y disponibles para participar como voluntario.

30 nov 2023	Limpieza de áreas comunales	Nueva	1 horas	0 Personas	Participar
29 nov 2023	Caminata dirigida	caminar con perros	2 horas	1 Personas	Participar
29 nov 2023	Paseos, juegos y socialización	limpieza de jaulas	30/11/23, 10:00 30/11/23, 10:15 15 minutos	2 Personas	Participar
29 nov 2023	Caminata dirigida	paseo por el parque calderon	30/11/23, 8:30 30/11/23, 9:30 1 horas	0 Personas	Participar
29 nov 2023	Limpieza, baño y cepillado de los perros y gatos	paseo por el parque calderon	30/11/23, 23:45 1/12/23, 1:15	0 Personas	Participar
6 nov 2023	Paseos, juegos y socialización	juegos	Tres horas	1 Personas	Participar

Figura 59. Catálogo de actividades.

Desarrollo de la historia de usuario N° 16: Automatización de recepción de donaciones de dinero.

En la *Figura 60* se visualiza la configuración de una cuenta empresarial de PayPal que está asociada al aplicativo web para que la fundación pueda recibir donaciones de dinero.

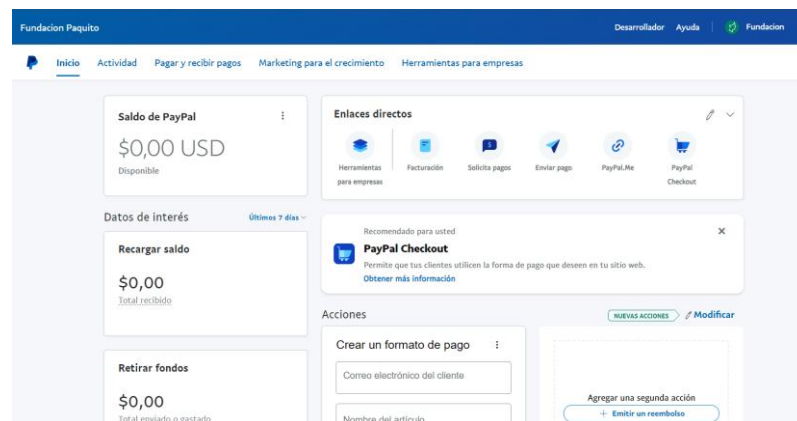


Figura 60. Cuenta empresarial de PayPal asociada a el aplicativo web.

3.2.2.2 Tarjetas CRC (Clase, Responsabilidad y Colaboración)

Las tarjetas CRC permiten diseñar y organizar un sistema de programación orientada a objetos. Estas tarjetas ayudan a identificar y definir las responsabilidades y colaboraciones entre las diferentes clases del sistema.

El modelo de tarjetas CRC se detalla en la tabla Tabla 102.

Tabla 102. Modelo de tarjeta CRC.

Clase	
Responsabilidades	Colaboradores
<Define las responsabilidades de las operaciones que tiene la clase>	<Clases asociadas con respecto a las clases de la tarjeta>

Tabla 103. Tarjeta CRC- Usuario.

Usuario	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> • Verifica las credenciales de inicio de sesión. • Proporciona información de los usuarios registrados. 	Actividad Solicitud

Tabla 104. Tarjeta CRC- Mascota.

Mascota	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> • Proporciona información de las mascotas registradas. • Gestiona información de las mascotas. 	Solicitud Enfermedad Vacuna Reconocimiento

Tabla 105. Tarjeta CRC- Voluntariado.

Voluntariado	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> • Gestiona información de voluntariado. • Gestiona la recepción de usuarios para voluntariado. • Proporciona información de voluntariado. 	Usuario Actividad

Tabla 106. Tarjeta CRC- Adopción.

Adopción	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> • Gestiona el proceso de adopción. • Notifica la resolución de una adopción. • Proporciona información de adopciones. 	Solicitud

Tabla 107. Tarjeta CRC- Solicitud.

Solicitud	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> • Recapta solicitudes de adopción. • Gestiona información de la solicitud. 	Usuario

Tabla 108. Tarjeta CRC- Actividad.

Actividad	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> • Gestiona información de las actividades. • Proporciona información de las actividades. 	Voluntariado. Usuario

Tabla 109. Tarjeta CRC- Enfermedad.

Enfermedad	
Responsabilidades	Colaboradores
<ul style="list-style-type: none"> • Gestiona información de las enfermedades y alergias. • Proporciona información de enfermedades y alergias. 	Mascota.

3.2.3 Fase 3: Codificación

La parte del backend del aplicativo web se lo realizó con .Net Core y la parte del frontend con el framework Angular.

Backend del aplicativo web

El backend se ocupa de la lógica de la aplicación, el acceso a la base de datos y otros aspectos que no están directamente relacionados con la interfaz de usuario.

Conexión a la base de datos

Para tener acceso a ella se debe tener las credenciales de usuario y contraseña. Además, se debe configurar el servicio de la conexión en el archivo Program.cs del proyecto.

```

"Logging": {
  "LogLevel": {
    "Default": "Information",
    "Microsoft.AspNetCore": "Warning"
  }
},
"AllowedHosts": "*",
"ConnectionStrings": {
  "MySQLConnectionString": "Server=159.89.85.43;Port=3309;User ID=root;Password=fpaquitodb;Database=fpdb"
}

```

Figura 61. Cadena de conexión a la base de datos.

Entidades

Las entidades son representadas como tablas en una base de datos, donde cada fila de la tabla corresponde a una instancia específica de esa entidad y cada columna representa un atributo o característica asociada a la entidad.

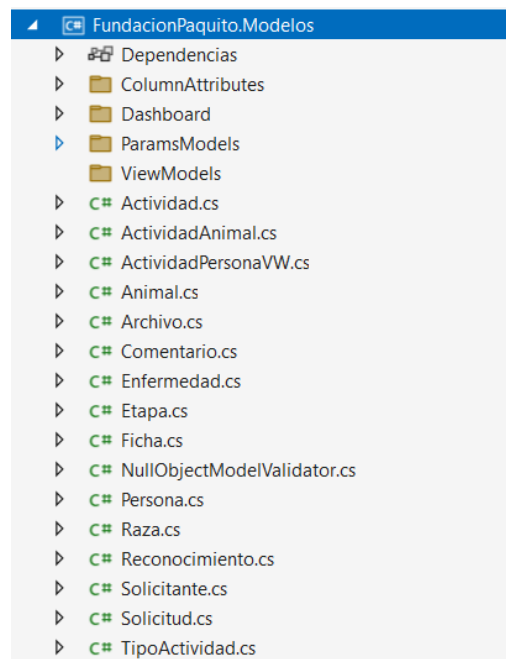


Figura 62. Entidades que componen el aplicativo web.

La capa de acceso a datos (DAL)

La capa de acceso a datos separa limpiamente la lógica de acceso a datos de la lógica de presentación.

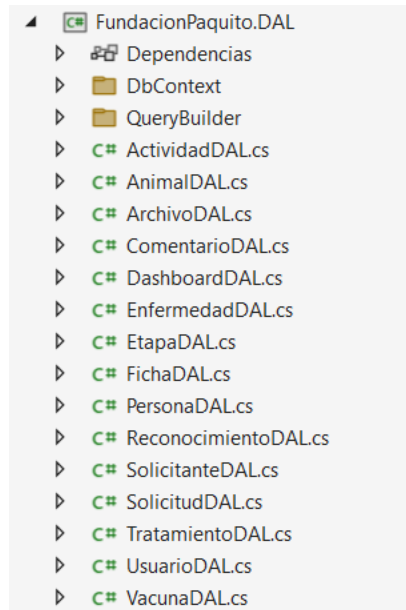


Figura 63. Capa de acceso a los datos del aplicativo web.

Capa de lógica de negocios (BLL)

La capa de lógica de negocios actúa como intermediario para el intercambio de datos entre la capa de presentación y la DAL.



Figura 64. Capa de lógica de negocios del aplicativo web.

Controllers

La clase de controlador es una parte fundamental que maneja las solicitudes entrantes, procesa la lógica de negocios correspondiente y devuelve las respuestas adecuadas.

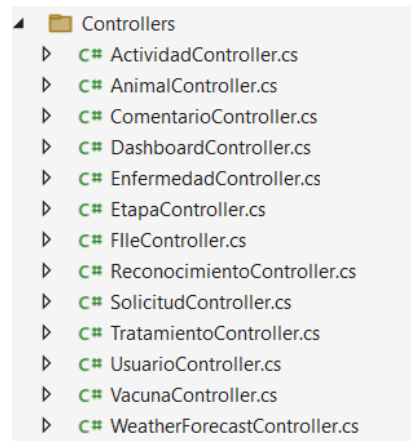


Figura 65. Clase controllers del aplicativo web.

Frontend del aplicativo web.

Es la interfaz de usuario visible y accesible que permite a los usuarios interactuar con el aplicativo web.

Login

Este módulo del aplicativo web permitirá ingresar un correo y una contraseña para que el usuario pueda autenticarse.

En la Figura 66 se visualiza el código realizado dentro del archivo login.component.ts mediante el cual se permite o se deniega el acceso al aplicativo web.

```

signIn(){
  if(!this.signInForm.valid)
    return
  this.cargando = true
  this.loginservice.login(this.signInForm.get('email')?.value,this.signInForm.get('password')?.value).subscribe(d => {
    this.auth.setUser(d.usuario,d.tk)
    console.log(this.redirectUrl)
    if(this.redirectUrl.length == 0)
    {
      this.router.navigate(['/home'])
    }
    else
    {
      document.location.href = this.redirectUrl
    }
  })
  this.cargando = false
}, e => {
  this.toast.error('Usuario o contraseña incorrectos')
  this.cargando = false
})
}

```

Figura 66. Archivo login.component.ts.

En la Figura 67 se visualiza el código realizado dentro del archivo login.service.ts.

```

export class LoginService {
  constructor(private http:HttpService) { }
  login(usuario:string, password:string){
    return this.http.post('usuario/login', {usuario,password})
  }
}

```

Figura 67. Archivo login.service.ts.

Formulario de registro

El formulario permitirá a nuevos usuarios registrarse en el aplicativo web, proporcionando la información necesaria para crear una cuenta.

En la Figura 68 se visualiza el código realizado dentro del archivo registrar.component.ts mediante el cual se recibe los datos del usuario para registrarlo en el aplicativo.

```

registrar() {
  this.validarForm(this.registrarForm)

  if(!this.registrarForm.get('recordarme').value) {
    this.toast.warning('Acepta los términos y condiciones')
    return
  }

  if(!this.registrarForm.valid) {
    this.toast.warning('Ingresa todos los datos obligatorios')
    return
  }

  if(this.registrarForm.get('password').value !== this.registrarForm.get('password2').value) {
    this.toast.warning('Las contraseñas no coincide, intentalo de nuevo')
    return
  }

  console.log(this.registrarForm.value)
  let usuario = {
    usu_id: this.registrarForm.get('email').value,
    usu_pass: this.registrarForm.get('password').value,
    persona: {
      per_nombre: this.registrarForm.get('nombres').value,
      per_apellido: this.registrarForm.get('apellidos').value,
      per_identificacion: this.registrarForm.get('cedula').value,
      per_telefono: this.registrarForm.get('celular').value,
      per_fecha_nacimiento: new Date(this.registrarForm.get('anio_nacimiento').value, this.registrarForm.get('mes_nacimiento').value, this.registrarForm.get('dia_nacimiento').value),
      per_email: this.registrarForm.get('email').value,
      per_estado_civil: this.registrarForm.get('estado_civil').value,
    }
  }

  this.error = null
  this.registrando = true
  this.registrarservice.Registrar(usuario).subscribe(d => {
    console.log(d)
    this.registrando = false
    this.router.navigate(['/login'], {queryParams:{nuevo: 1}})
  }, e => {
    this.registrando = false
    console.log(e)
    this.error = e.error ? e.error : "Ha ocurrido un error"
  })
}

validarForm(form:FormGroup) {
  Object.values(form.controls).forEach(control => {
    if (control.invalid) {
      control.markAsDirty()
      control.updateValueAndValidity({ onlySelf: true })
    }
  })
}
}

```

Figura 68. Archivo registrar.component.ts.

En la Figura 69 se visualiza el código realizado dentro del archivo registrar.service.ts.

```

export class RegistrarService {
  constructor(private http:HttpService) { }

  Registrar(usuario:any):Observable<any>{
    return this.http.post(`usuario/registrar`, usuario)
  }
}

```

Figura 69. Archivo registrar.service.ts.

Actualizar información del usuario

Permitirá a los usuarios actualizar su información personal, como dirección, número de teléfono, estado civil, correo.

En la Figura 70 se visualiza el código realizado dentro del archivo `actualizarUsuario.component.ts` mediante el cual se actualiza los datos del usuario.

```

obtenerUsuario(){
  this.usuarioservice.obtenerUsuario(this.data.usu_codigo).subscribe(d => {
    console.log(d);

    this.form.get('cedula')?.setValue(d.persona.per_identificacion),
    this.form.get('nombres')?.setValue(d.persona.per_nombre),
    this.form.get('apellidos')?.setValue(d.persona.per_apellido),
    this.form.get('celular')?.setValue(d.persona.per_telefono),
    this.form.get('email')?.setValue(d.persona.per_email),
    this.form.get('estado_civil')?.setValue(d.persona.per_estado_civil),
    this.form.get('dia')?.setValue(new Date(d.persona.per_fecha_nacimiento).getDate()),
    this.form.get('mes')?.setValue(new Date(d.persona.per_fecha_nacimiento).getMonth()),
    this.form.get('anio')?.setValue(new Date(d.persona.per_fecha_nacimiento).getFullYear()),
    this.form.get('direccion')?.setValue(d.persona.per_direccion)
  })
}

submitForm(){
  if(!this.form.valid){
    this.toast.warning('Ingrese todos los campos obligatorios')
    return
  }

  let u = {
    usu_codigo: this.data.usu_codigo,
    usu_id: this.data.usu_id,
    usu_per_codigo: this.data.usu_per_codigo,

    persona: {
      per_codigo: this.data.usu_per_codigo,
      per_nombre: this.form.get('nombres')?.value,
      per_apellido: this.form.get('apellidos')?.value,
      per_identificacion: this.form.get('cedula')?.value,
    }
  }

  this.usuarioservice.actualizar(u).subscribe(d => {
    this.toast.success("Datos actualizados correctamente")
    this.dialogRef.close({status:true})
  }, e => {
    this.toast.error("Ha ocurrido un error")
  })
}

```

Figura 70. Archivo actualizarUsuario.component.ts.

Asignar perfiles de usuario

Permite asignar perfiles de usuario con diferentes niveles de acceso y privilegios dentro del aplicativo web, garantizando la seguridad y la adecuada gestión de funcionalidades.

En la Figura 71 se visualiza el código realizado dentro del archivo tipoPermiso.component.ts mediante el cual se asigna un perfil de usuario.

```
ngOnInit() {
  if(this.data.usu_administrador == 1){
    this.form.get('permiso')?.setValue("1")
  }
  else{
    this.form.get('permiso')?.setValue("2")
  }
}

submitForm(){
  if(!this.form.valid){
    this.toast.warning('Seleccione un tipo')
    return
  }

  this.usuarioservice.cambiarTipo(this.data.usu_codigo, this.form.get('permiso')?.value).subscribe(d => {
    this.toast.success("Permiso cambiado correctamente")
    this.dialogRefP.close({status:true})
  })
}
}
```

Figura 71. Archivo tipoPermiso.component.ts.

En la Figura 72 se visualiza el código realizado dentro del archivo usuario.service.ts.

```
export class UsuarioService {

  constructor(private http:HttpService) { }

  obtenerUsuarios():Observable<any>{
    return this.http.get(`usuario/lista`)
  }

  obtenerUsuario(usuario:number):Observable<any>{
    return this.http.get(`usuario/${usuario}`)
  }

  actualizar(usuario:any):Observable<any>{
    return this.http.post(`usuario/actualizar`,usuario)
  }

  cambiarTipo(usuario:number, tipo: number):Observable<any>{
    return this.http.post(`usuario/cambiar_tipo/${usuario}/${tipo}`)
  }
}
```

Figura 72. Archivo usuario.service.ts.

Gestionar información de las mascotas

La interfaz permitirá a los usuarios autorizados gestionar y mantener actualizada la información detallada de las mascotas.

En la Figura 73 se visualiza el código realizado dentro del archivo crearMascota.component.ts mediante el cual se registra o edita información de las mascotas.

```
obtenerMascota(){
  this.mascotaservice.obtenerMascota(this.data.id).subscribe(d => {
    this.mascota = d
    this.mascotaForm.get('nombre')?.setValue(d.ani_nombre)
    this.mascotaForm.get('tipo')?.setValue(d.ani_tia_codigo)
    this.mascotaForm.get('raza')?.setValue(d.ani_raz_codigo)
    this.mascotaForm.get('sexo')?.setValue(d.ani_sexo)
    this.mascotaForm.get('tamano')?.setValue(d.ani_tamano)
    this.mascotaForm.get('esterilizado')?.setValue(d.ani_esterilizado)
    if(d.ani_fecha_nacimiento){
      this.mascotaForm.get('dia')?.setValue(new Date(d.ani_fecha_nacimiento).getDate())
      this.mascotaForm.get('mes')?.setValue(new Date(d.ani_fecha_nacimiento).getMonth())
      this.mascotaForm.get('anio')?.setValue(String(new Date(d.ani_fecha_nacimiento).getFullYear()))
    }
    else{
      this.mascotaForm.get('dia')?.setValue(null)
      this.mascotaForm.get('mes')?.setValue(null)
      this.mascotaForm.get('anio')?.setValue(null)
    }
    this.mascotaForm.get('direccion')?.setValue(d.ani_direccion_rescate)
  })
}

SubmitMascotaForm(){
  if(!this.mascotaForm.valid){
    this.toast.warning('Ingrese todos los campos obligatorios')
    return
  }

  let form = this.mascotaForm.value
  console.log(form)

  let ani_fecha_nacimiento = null

  if(form.dia && form.mes && form.anio){
    ani_fecha_nacimiento = new Date(form.anio, form.mes, form.dia)
  }

  let mascota:any = {
    ani_nombre: form.nombre,
    ani_sexo: form.sexo,
    ani_tamano: form.tamano,
    ani_esterilizado: form.esterilizado ? 1: 0,
    ani_tia_codigo: form.tipo,
    ani_raz_codigo: form.raza,
    ani_fecha_nacimiento,
    ani_direccion_rescate: form.direccion
  }

  if(this.modo == 'editar'){
    mascota['ani_codigo'] = this.mascota.ani_codigo
  }

  this.loading = true
  this.mascotaservice.GuardarMascota(mascota).subscribe(d => {
    this.toast.success('Mascota creada correctamente')
    this.loading = false
    this.dialogRef.close({status: true})
  }, e => {
    this.loading = false
    this.toast.error('Ha ocurrido un error!')
  })
}
```

Figura 73. Archivo crearMascota.component.ts.

Formulario para registrar información médica de las mascotas

El formulario permitirá registrar información médica relevante de las mascotas, como historial de vacunación, enfermedades o alergias y condiciones de salud.

En la Figura 74 se visualiza el código realizado dentro del archivo crearEnfermedad.component.ts mediante el cual se registra enfermedades o alergías que tenga la mascota.

```
SubmitForm(){
  if(!this.form.valid){
    this.toast.warning('Ingresa todos los campos requeridos')
    return
  }

  let enfermedad = {
    enf_nombre : this.form.get('nombre)?.value,
    enf_descripcion :this.form.get('descripcion)?.value,
    enf_fic_codigo: this.data.mascota.ficha
  }

  this.mascotaService.crearEnfermedad(enfermedad).subscribe(d => {
    this.dialogRef.close({status: true})
  }, e=> {
    this.toast.error('Error al guardar enfermedad')
  })
}
```

Figura 74. Archivo crearEnfermedad.component.ts.

En la Figura 75 se visualiza el código realizado dentro del archivo crearVacuna.component.ts mediante el cual se registra vacunas que se le hayan aplicado a las mascotas.


```

SubmitForm(){
  if(!this.form.valid){
    this.toast.warning('Ingrese todos los campos requeridos')
    return
  }

  let vacuna = {
    vac_fecha_reuerdo: this.selected,
    vac_tiv_codigo : this.form.get('tipo')?.value,
    vac_rec_codigo: this.data.rec.rec_codigo
  }

  this.mascotaService.crearVacuna(vacuna).subscribe(d => {
    this.dialogRefV.close({status: true})
  })
}

```

Figura 75. Archivo crearVacuna.component.ts.

En la Figura 76 se visualiza el código realizado dentro del archivo crearReconocimiento.component.ts mediante el cual se registra datos como el peso, datos de pruebas realizadas por un veterinario que se le haya realizado a las mascotas.

```

SubmitForm(){
  if(!this.form.valid){
    this.toast.warning('Ingrese todos los valores requeridos')
    return
  }

  let form = this.form.value

  let reconocimiento = {
    rec_descripcion: form.descripcion,
    rec_diagnostico: form.diagnostico,
    rec_pruebas_realizadas: form.pruebas,
    rec_resultados: form.resultados,
    rec_notas: '',
    rec_peso: form.peso,
    rec_fic_codigo: this.data.ficha
  }

  this.mascotaService.crearReconocimiento(reconocimiento).subscribe(d => {
    console.log(d)
    this.dialogRef.close({status: true})
  }, e=> {
    this.toast.error('Error al crear un reconocimiento')
  })
}

```

Figura 76. Archivo crearReconocimiento.component.ts.

En la Figura 77 se visualiza el código realizado dentro del archivo mascota.service.ts.

```

export class MascotaService {
  constructor(private http:HttpService) { }

  GuardarMascota(mascota:any):Observable<any>{
    return this.http.post(`animal`, mascota)
  }

  ObtenerMascotas():Observable<any>{
    return this.http.get(`animal/animales`)
  }

  obtenerMascota(id:string):Observable<any>{
    return this.http.get(`animal?id=${id}`)
  }

  crearReconocimiento(reconocimiento:any):Observable<any>{
    return this.http.post(`reconocimiento`, reconocimiento)
  }

  obtenerReconocimientos(ficha:number):Observable<any>{
    return this.http.get(`reconocimiento/${ficha}`)
  }

  crearVacuna(vacuna:any):Observable<any>{
    return this.http.post(`vacuna`, vacuna)
  }

  obtenerTiposVacunas():Observable<any>{
    return this.http.get(`vacuna/tipos`)
  }

  obtenerVacunas(reconocimiento:number):Observable<any>{
    return this.http.get(`vacuna/${reconocimiento}`)
  }

  crearEnfermedad(enfermedad:any):Observable<any>{
    console.log(enfermedad)
    return this.http.post(`enfermedad`, enfermedad)
  }

  obtenerEnfermedades(ficha:number):Observable<any>{
    return this.http.get(`enfermedad/${ficha}`)
  }
}

```

Figura 77. Archivo mascota.service.ts.

Catálogo de mascotas

Presenta información sobre las mascotas disponibles para adopción. Este catálogo facilitará a los usuarios la búsqueda y selección de mascotas.

En la Figura 78 se visualiza el código realizado dentro del archivo catálogo.component.ts mediante el cual se obtiene información de las mascotas en forma de galería.

```

ngOnInit() {
  this.catalogoservice.GetCatalogo().subscribe(data => {
    let lista:any[] = data
    lista.forEach(x => {
      if(x.ani_foto && x.ani_foto.length > 0){
        let arr = x.ani_foto.split(',')
        x.fotos = arr
      }
      else{
        x.fotos = []
        x.fotos.push('https://firebasestorage.googleapis.com/v0/b/fundacion-paquito-app.appspot.com/o/LOGOTIPO%20(1).jpeg?alt=')
      }
    })
    this.list = lista
  })
}

```

Figura 78. Archivo catálogo.component.ts.

En la Figura 79 se visualiza el código realizado dentro del archivo catálogo.service.ts.

```

export class CatalogoService {
  constructor(private http:HttpClient) { }

  GetCatalogo():Observable<any>{
    return this.http.get('animales')
  }
}

```

Figura 79. Archivo catálogo.service.ts.

Formulario de adopción

Permite a los usuarios interesados en adoptar una mascota completar y enviar su solicitud. Este formulario capturará información relevante para evaluar la idoneidad del adoptante.

En la Figura 80 se visualiza el código realizado dentro del archivo formulario-adopcion.component.ts mediante el cual los usuarios podrán enviar los datos necesarios para solicitar la adopción de una mascota.

```

ngOnInit() {
  this.animalId = this.route.snapshot.queryParamMap.get('id');
  this.getAnimal();

  this.auth.user.subscribe((u:any) => {
    this.user = u

    this.formSituacionPersonal.get('cedula')?.setValue(this.user.persona.per_identificacion)
    this.formSituacionPersonal.get('nombres')?.setValue(this.user.persona.per_nombre + ' ' + this.user.persona.per_apellido)
    this.formSituacionPersonal.get('direccion')?.setValue(this.user.persona.per_direccion)
    this.formSituacionPersonal.get('telefono')?.setValue(this.user.persona.per_telefono)
    this.formSituacionPersonal.get('estado_civil')?.setValue(this.user.persona.per_estado_civil)
    this.formSituacionPersonal.get('email')?.setValue(this.user.persona.per_email)
    this.formSituacionPersonal.get('edad')?.setValue(this.user.persona.per_edad)

    this.formSituacionPersonal.get('cedula')?.disable()
    this.formSituacionPersonal.get('nombres')?.disable()
  })
}

getAnimal(){
  this.formularioService.obtenerAnimal(this.animalId).subscribe(d => {
    this.animalSeleccionado = d
    if(this.animalSeleccionado.ani_foto && this.animalSeleccionado.ani_foto.length > 0){
      let arr = this.animalSeleccionado.ani_foto.split(',')
      this.animalFotos = arr
    }
    else {
      this.animalFotos = []
      this.animalFotos.push('https://firebasestorage.googleapis.com/v0/b/fundacion-paquito-app.appspot.com/o/LOGOTIPO020(1).')
    }
  })
}

validarFormulario(){
  this.validateForm(this.formSituacionPersonal)
  this.validateForm(this.formRelacionAnimales)
  if(!this.formSituacionPersonal.valid || !this.formRelacionAnimales.valid){
    this.toast.warning('Ingresa todos los campos requeridos!');
    return
  }

  let form1 = this.formSituacionPersonal.value
  let form2 = this.formRelacionAnimales.value

  console.log(form1)
  console.log(form2)

  let nuevoformulario:any = {}
  nuevoformulario['personal_info'] = form1
  nuevoformulario['relacion_con_animales'] = form2
  nuevoformulario['personas_con_las_que_vive'] = this.personas_vive

  let solicitud = {
    soi_ani_codigo: this.animalSeleccionado.ani_codigo,
    soi_fica : JSON.stringify(nuevoformulario)
  }

  this.enviandoFormulario = true
  this.formularioService.guardarFormulario(solicitud).subscribe(d => {
    console.log(d)
    this.enviandoFormulario = false
  })
  validateForm(form:FormGroup){
    Object.values(form.controls).forEach(control => {
      if (control.invalid) {
        control.markAsDirty()
        control.updateValueAndValidity({ onlySelf: true })
      }
    })
  }
}

agregarPersona(){
  this.personas_vive.push({edad: 1, parentesco: 'papa'})
}

quitarPersona(index:number){
  this.personas_vive.splice(index,1);
}
}

```

Figura 80. Archivo formulario-adopcion.component.ts.

En la Figura 81 se visualiza el código realizado dentro del archivo formulario.service.ts.

```
export class FormularioService {  
  constructor(private http:HttpService) { }  
  
  guardarFormulario(formulario:any):Observable<any>{  
    return this.http.post('solicitud/crear',formulario)  
  }  
  
  obtenerAnimal(id:string):Observable<any>{  
    return this.http.get(`animal?id=${id}`)  
  }  
  
  obtenerPersona(id:number):Observable<any>{  
    return this.http.get(`usuario/persona/${id}`)  
  }  
}
```

Figura 81. Archivo formulario.service.ts.

Elaborado por: El Investigador.

Gestionar solicitudes de adopción

La interfaz permitirá revisar, evaluar y gestionar las solicitudes de adopción recibidas, asegurando un proceso eficiente y transparente.

En la Figura 82 se visualiza el código realizado dentro del archivo detalleSolicitud.component.ts mediante el cual se gestionan las solicitudes recibidas.

```

ngOnInit() {
  this.getEtapas()
  this.auth.user.subscribe((x:any) => {
    this.user = x
  })
  this.sol_id = this.route.snapshot.paramMap.get('id')
  this.sub = this.route.queryParams.subscribe(params => {
    this.tab = params['tab']
    this.obtenerDetalle()
    this.obtenerComentarios()
  })
}

obtenerDetalle(){
  this.detalleservice.obtenerDetalle(this.sol_id).subscribe(d => {
    this.detalle = d
    this.etapaseleccionada = this.detalle.sol_ets_codigo
    this.animalseleccionado = d.animal
    if(d.sol_fica){
      this.formulario = JSON.parse(d.sol_fica)
    }

    if(this.animalseleccionado.ani_foto && this.animalseleccionado.ani_foto.length > 0){
      let arr = this.animalseleccionado.ani_foto.split(',')
      this.animalFotos = arr
    }
    else{
      this.animalFotos = []
      this.animalFotos.push("https://firebasestorage.googleapis.com/v0/b/fundacion-paquito-app.appspot.com/o/LOGOTIPOX20(1).")
    }
  })
}

guardarComentario(){
  if(!this.comentarioForm.valid){
    this.toast.warning('Escribe un comentario')
    return
  }

  let comentario:any = {
    com_sol_codigo: this.detalle.sol_codigo,
    com_descripcion: this.comentarioForm.get('comentario')?.value
  }

  this.detalleservice.guardarComentario(this.sol_id,comentario).subscribe(d => {
    this.toast.success('Comentario creado')
    this.obtenerComentarios()
  })
}

obtenerComentarios(){
  this.detalleservice.comentarios(this.sol_id).subscribe(d => {
    this.comentarios = d
  })
}

cancelar(comentario:string){
  this.cancelando = true
  this.detalleservice.cancelar(this.sol_id,comentario).subscribe(d => {
    this.toast.success('Solicitud aprobada')
    this.cancelando = false
    this.obtenerDetalle()
  },e => {
    this.cancelando = false
  })
}

cambiarEtapa(){
  this.cambiandoEtapa = true
  this.detalleservice.cambiarEtapa(this.sol_id, this.etapaseleccionada).subscribe(d => {
    this.toast.success('Se ha cambiado de etapa')
    this.cambiandoEtapa = false
    this.obtenerDetalle()
  }, e => {
    this.cambiandoEtapa = false
    this.toast.error("error al cambiar de etapa")
  })
}

getEtapas(){
  this.detalleservice.etapas().subscribe(d => {
    this.etapas = d
  })
}

seleccionaretape(ori:any){
  this.cambiarEtapa()
}

abrirCancelarDialog(){
  const dialogref = this.dialog.open(CancelarFormComponent,{
    width: '600px',
  })
  dialogref.afterClosed().subscribe(
    (res)->{
      this.cancelar(res.comentario)
    }
  )
}
}

```

Figura 82. Archivo formulario-adopcion.component.ts.

Elaborado por: El Investigador.

En la Figura 83 se visualiza el código realizado dentro del archivo detalleSolicitud.service.ts.

```

export class DetalleSolicitudService {

  constructor(private http:HttpService) { }

  ObtenerDetalle(id:string):Observable<any>{
    return this.http.get(`solicitud?id=${id}`)
  }

  cancelar(solicitud_id:string, comentario:string):Observable<any>{
    return this.http.post(`solicitud/cancelar/${solicitud_id}`, {comentario})
  }

  etapas():Observable<any>{
    return this.http.get(`etapa/lista`)
  }

  cambiarEtapa(solicitud_id:string, etapa:number):Observable<any>{
    return this.http.post(`solicitud/cambiar_etapa/${solicitud_id}/${etapa}`,{})
  }

  AceptarMascota(solicitud_id:string):Observable<any>{
    return this.http.post(`solicitud/aceptar_mascota/${solicitud_id}`,{})
  }
}

```

Figura 83. Archivo datalleSolicitud.service.ts.

Elaborado por: El Investigador.

Reportes

Se generará y visualizará gráficas estadísticas relacionadas con el proceso de adopción y actividades de voluntariado.

En la Figura 84 se visualiza el código realizado dentro del archivo dashboard.component.ts mediante el cual se visualizarán gráficas estadísticas.

```
ngOnInit() {
  this.getTotalPorTipos()
  this.getTotalPorMascotaSolicitud()
  this.getTotalPorSolicitudes()
  this.getTotalPorSegmentos()
}

getTotalPorTipos(){
  this.dashboardservice.getTotalPorTipos().subscribe(d => {
    let data:any[] = d
    let series:any = []
    let labels:any = []
    data.forEach(x => {
      series.push(x.total)
      labels.push(x.tia_nombre)
    })

    this.chartOptions = {
      series: series,
      chart: {
        type: "pie"
      },
      labels: labels,
      responsive: [
        {
          breakpoint: 480,
          options: {
            chart: {
              width: 200
            },
            legend: {
              position: "bottom"
            }
          }
        }
      ]
    }
  });
}

getTotalPorMascotaSolicitud(){
  let series:any[] = []
  let categorias:any[] = []

  this.dashboardservice.getTotalPorMascotaSolicitud().subscribe(d => {
    let data:any[] = d
    series.push({name: 'Activas', data: data.map(x => x.tia_num_activas)})
    series.push({name: 'Aceptadas', data: data.map(x => x.tia_num_aceptadas)})
    series.push({name: 'Canceladas', data: data.map(x => x.tia_num_canceladas)})

    categorias = data.map(x => x.tia_nombre)

    this.chartOptions2 = {
      series: series,
      chart: {
        type: "bar",
        stacked: true,
        stacktype: "100%"
      },
      responsive: [
        {
          breakpoint: 480,
          options: {
            legend: {
              position: "bottom",
              offsetX: -50,
              offsetY: 0
            }
          }
        }
      ],
      series: [
        {
          categories: categorias
        }
      ],
      fill: {
        opacity: 1
      },
      legend: {
        position: "top",
      }
    }
  });
}
```

Figura 84. Archivo dashboard.component.ts.

En la Figura 85 se visualiza el código realizado dentro del archivo detalleSolicitud.service.ts.


```

export class DashboardService {

  constructor(private http:HttpService) { }

  getTotalPorTipos():Observable<any>{
    return this.http.get(`Dashboard/total_por_tipos`)
  }
  getTotalPorMascotaSolicitud():Observable<any>{
    return this.http.get(`Dashboard/total_por_tipos_mascota_solicitud`)
  }

  getTotalPorActividades():Observable<any>{
    return this.http.get(`Dashboard/total_por_actividades_activas`)
  }

  getTotalPorSolicitudes():Observable<any>{
    return this.http.get(`Dashboard/total_por_solicitudes`)
  }
}

```

Figura 85. Archivo dashboard.service.ts.

Formulario de actividades para voluntariado.

Se creará un formulario que facilite la creación de nuevas actividades de voluntariado, permitiendo describir la actividad, establecer fecha y duración de la actividad.

En la Figura 86 se visualiza el código realizado dentro del archivo crearActividad.component.ts mediante el cual se crean actividades para voluntariado.

```

ngOnInit() {
  this.obtenerTipos()
  this.fechaSeleccionada()
}

fechaSeleccionada(){

  var h = this.generarListaHoras(this.dateSelected, '01:00', '23:00')
  console.log(h)
  this.horas = h
}

generarListaHoras(fecha:Date,horaInicial: string, horaFinal: string): string[] {

  let fech = new Date(fecha)
  const listaHoras: any[] = [];

  // Parsea las horas iniciales y finales
  const horaInicio = new Date('1970-01-01T' + horaInicial + ':00');
  const horaFin = new Date('1970-01-01T' + horaFinal + ':00');

  // Verifica si las horas son válidas
  if (isNaN(horaInicio.getTime()) || isNaN(horaFin.getTime())) {
    console.error('Horas proporcionadas no son válidas');
    return listaHoras;
  }

  // Configura la fecha con las horas iniciales
  fech.setHours(horaInicio.getHours());
  fech.setMinutes(horaInicio.getMinutes());

  let fechaFin:Date = new Date(fech)
  fechaFin.setHours(horaFin.getHours())
  fechaFin.setMinutes(horaFin.getMinutes())

  // Itera sobre las horas y agrega a la lista
  while (fech <= fechaFin) {
    const horaFormateada = this.formatearHora(fech);
    listaHoras.push({date: new Date(fech), label: horaFormateada});
    fech.setMinutes(fech.getMinutes() + 15);
  }

  return listaHoras;
}

private formatearHora(hora: Date): string {
  return hora.getHours().toString().padStart(2, '0') + ':' + hora.getMinutes().toString().padStart(2, '0');
}

obtenerTipos(){
  this.voluntariadoservice.obtenerTiposActividad().subscribe(d => {
    this.tipos = d
  })
}

obtenerDiferenciaEnTiempo(fechaInicio: Date, fechaFin: Date): string {
  const diferenciaEnMinutos = differenceInMinutes(fechaFin, fechaInicio);

  if (diferenciaEnMinutos < 60) {
    return `${diferenciaEnMinutos} minutos`;
  } else {
    const diferenciaEnHoras = differenceInHours(fechaFin, fechaInicio);

    if (diferenciaEnHoras < 24) {
      const diferenciaEnDias = Math.floor(diferenciaEnHoras / 24);
      return `${diferenciaEnDias} días`;
    }
  }
}
}

```

Figura 86. Archivo crearActividad.component.ts.

En la Figura 87 se visualiza el código realizado dentro del archivo crearActividad.service.ts.

```

obtenerTiposActividad():Observable<any>{
  return this.http.get(`actividad/tipos`)
}

crearActividad(actividad:any):Observable<any>{
  return this.http.post(`actividad`,actividad)
}

obtenerActividades():Observable<any>[
  return this.http.get(`actividad/lista`)
]

obtenerPersonasActividad(actividad:number):Observable<any>{
  return this.http.get(`actividad/personas/${actividad}`)
}

obtenerMascotasAsignar():Observable<any>{
  return this.http.get(`actividad/mascotas`)
}

asignarMascota(actividadpersona:any):Observable<any>{
  return this.http.post(`actividad/asignar_mascota`, actividadpersona)
}

```

Figura 87. Archivo voluntariado.service.ts.

3.2.4 Fase 4: Pruebas

En la fase de pruebas se procede a realizar las pruebas de aceptación de los requerimientos descritos en las historias de usuario. El modelo de las pruebas de aceptación se describe en la Tabla 110.

Tabla 110. Modelo de prueba de aceptación.

Prueba de aceptación	
Número: [Número de prueba de aceptación]	Historia de usuario: [Número de historia de usuario]
Nombre de historia: [Título de historia de usuario]	
Descripción: [Se describe con un lenguaje natural sobre lo que debe hacer cada historia]	
Condiciones de ejecución: [Se describen las condiciones previas que deben cumplirse para ejecutar la historia de usuario]	
Entrada / Pasos de ejecución: [Descripción de los pasos que realiza el usuario para realizar la historia de usuario]	
Resultado esperado: [Descripción del resultado esperado en base a los criterios de aceptación de la historia de usuario]	
Evaluación de prueba: [Resultado final de la prueba]	

Tabla 111. Prueba de aceptación 1: Creación de la estructura del aplicativo web.

Prueba de aceptación	
Número: 1	Historia de usuario: 1
Nombre de historia: Creación de la estructura del aplicativo web.	
Descripción: Como personal administrativo quiero almacenar de manera estructura toda la información de los procesos que la fundación lleva a cabo para automatizar dichos procesos y evitar redundancia y pérdida de información.	
Condiciones de ejecución: Ser parte del personal administrativo.	
Entrada / Pasos de ejecución: Ninguna	
Resultado esperado: Dado que el aplicativo manejará la información del proceso de adopción de mascotas, voluntariado y donaciones Cuando se necesite llevar un registro Entonces diseñar una base de datos que incluya las tablas y campos que permitan almacenar la información correspondiente.	
Evaluación de prueba: Satisfactoria.	

Tabla 112. Prueba de aceptación 2: Formulario de registro.

Prueba de aceptación	
Número: 2	Historia de usuario: 2
Nombre de historia: Formulario de registro.	
Descripción: Como usuario del aplicativo web quiero registrar mis datos de manera segura para utilizar las diferentes funcionalidades del aplicativo web.	
Condiciones de ejecución: Ser un usuario que vaya a hacer una adopción o voluntariado.	
Entrada / Pasos de ejecución: El usuario ingresa a la opción de registrarse y llena todos los campos.	
Resultado esperado: Dado que el usuario ingresa a la opción de registro Cuando llene todos los datos Entonces el aplicativo web registrará sus datos y lo dirigirá a al Inicio.	
Evaluación de prueba: Satisfactoria.	

Tabla 113. Prueba de aceptación 3: Interfaz de autenticación y autorización.

Prueba de aceptación	
Número: 3	Historia de usuario: 3
Nombre de historia: Interfaz de autenticación y autorización.	
Descripción: Como usuario del aplicativo web quiero ingresar al aplicativo web de manera segura para tener acceso a las funcionalidades.	
Condiciones de ejecución: Ser un usuario que se encuentre previamente registrado en el aplicativo.	
Entrada / Pasos de ejecución: El usuario ingresa a la opción de Inicio e ingresa su contraseña y correo.	
Resultado esperado: Dado que el usuario ingresa a la opción de Inicio Cuando llene sus credenciales correctamente Entonces el aplicativo web permitirá el acceso a las funcionalidades.	
Evaluación de prueba: Satisfactoria.	

Tabla 114. Prueba de aceptación 4: Interfaz para actualizar información del usuario.

Prueba de aceptación	
Número: 4	Historia de usuario: 4
Nombre de historia: Interfaz para actualizar información del usuario.	
Descripción: Como usuario del aplicativo web quiero tener la posibilidad de tener acceso a la información registrada en el aplicativo web para poder actualizar información en caso de ser necesario.	
Condiciones de ejecución: Ser parte del personal administrativo o un usuario registrado.	
Entrada / Pasos de ejecución: El administrador o usuario debe haber ingresado al aplicativo web.	
Resultado esperado: Dado que el administrador o usuario ingresa a la opción de Usuarios Cuando necesite actualizar algún dato de la información Entonces el aplicativo web permitirá la actualización de los datos en la base de datos.	
Evaluación de prueba: Satisfactoria.	

Tabla 115. Prueba de aceptación 5: Interfaz para asignar perfiles de usuario.

Prueba de aceptación	
Número: 5	Historia de usuario: 5
Nombre de historia: Interfaz para asignar perfiles de usuario.	
Descripción: Como personal administrativo de la fundación quiero gestionar los perfiles de usuarios del aplicativo web con el fin de asignar roles y permisos a cada usuario para que solo puedan acceder a las funcionalidades que les corresponda.	
Condiciones de ejecución: Ser parte del personal administrativo.	
Entrada / Pasos de ejecución: El administrador debe haber ingresado al aplicativo y dirigirse a la opción Usuarios.	
Resultado esperado: Dado que el administrador ingresa a la opción de Usuarios Cuando necesite cambiar de rol a un usuario registrado en el aplicativo web Entonces el aplicativo web permitirá la actualización de rol de usuario en la base de datos.	
Evaluación de prueba: Satisfactoria.	

Tabla 116. Prueba de aceptación 6: Interfaz para gestionar información de las mascotas.

Prueba de aceptación	
Número: 6	Historia de usuario: 6
Nombre de historia: Interfaz para gestionar información de las mascotas.	
Descripción: Como personal administrativo de la fundación quiero gestionar la información de las mascotas rescatadas para tener los datos de cada mascota organizados y de esta manera puedan ser dadas en adopción.	
Condiciones de ejecución: Ser parte del personal administrativo.	
Entrada / Pasos de ejecución: El administrador debe haber ingresado al aplicativo y dirigirse a la opción Mis mascotas.	
Resultado esperado: Dado que el administrador ingresa a la opción de Mis mascotas Cuando necesite ingresar o modificar los datos de una mascota Entonces el aplicativo web permitirá el registro o actualización de los datos en la base de datos.	

Evaluación de prueba: Satisfactoria.

Tabla 117. Prueba de aceptación 7: Formulario para registrar información médica de las mascotas.

Prueba de aceptación	
Número: 7	Historia de usuario: 007
Nombre de historia: Formulario para registrar información médica de las mascotas.	
Descripción: Como personal administrativo de la fundación quiero gestionar la información médica de las mascotas rescatadas para tener un control de las vacunas y del estado de salud de las mascotas.	
Condiciones de ejecución: Ser parte del personal administrativo.	
Entrada / Pasos de ejecución: El administrador debe haber ingresado al aplicativo y seleccionar una de las mascotas en la tabla de Mis mascotas.	
Resultado esperado: Dado que el administrador ingresa a la opción de Mis mascotas y selecciona una mascota Cuando necesite ingresar o modificar los datos médicos de una mascota Entonces el aplicativo web permitirá el registro o actualización de los datos en la base de datos.	
Evaluación de prueba: Satisfactoria.	

Tabla 118. Prueba de aceptación 8: Catálogo de mascotas.

Prueba de aceptación	
Número: 8	Historia de usuario: 8
Nombre de historia: Catálogo de mascotas.	
Descripción: Como cliente del aplicativo web quiero visualizar las mascotas que se encuentran disponibles en adopción para poder tomar la decisión de por cuál de las mascotas aplicar para una posible adopción.	
Condiciones de ejecución: Ser parte del personal administrativo o usuario.	
Entrada / Pasos de ejecución: El administrador o usuario debe dirigirse a la opción de Adoptar.	

Resultado esperado: Dado que el administrador o usuario ingresa a la opción de Adoptar Cuando quiera adoptar una mascota Entonces el aplicativo web mostrará un catálogo con todas las mascotas que han sido registradas.
Evaluación de prueba: Satisfactoria.

Tabla 119. Prueba de aceptación 9: Formulario de solicitud de adopción.

Prueba de aceptación	
Número: 9	Historia de usuario: 9
Nombre de historia: Formulario de solicitud de adopción.	
Descripción: Como cliente del aplicativo web quiero tener un formulario mediante el cual pueda solicitar la adopción de una mascota para que la fundación pueda evaluar si soy o no un candidato idóneo para adoptar una mascota.	
Condiciones de ejecución: Ser parte del personal administrativo o usuario.	
Entrada / Pasos de ejecución: El administrador o usuario debe haber seleccionado una mascota del catálogo.	
Resultado esperado: Dado que el administrador o usuario selecciona una mascota del catálogo Cuando quiera adoptar una mascota Entonces el aplicativo web mostrará un formulario el cual se debe completar completamente antes de enviarlo para que el aplicativo registre los datos en la base de datos..	
Evaluación de prueba: Satisfactoria.	

Tabla 120. Prueba de aceptación 10: Interfaz para gestionar solicitudes de adopciones.

Prueba de aceptación	
Número: 10	Historia de usuario: 10
Nombre de historia: Interfaz para gestionar solicitudes de adopciones.	
Descripción: Como personal administrativo de la fundación quiero llevar un registro y control de cada una de las solicitudes de adopción que se recibe para poder analizar los datos del posible adoptante y tomar la decisión de dar o no una mascota en adopción.	
Condiciones de ejecución: Ser parte del personal administrativo.	

Entrada / Pasos de ejecución: El administrador debe haber ingresado al aplicativo y seleccionado la opción de Solicitudes.
Resultado esperado: Dado que el administrador selecciona la opción de solicitudes y elige una de las solicitudes Cuando quiera revisar los datos y aceptar o cancelar la solicitud Entonces el aplicativo web registra la resolución en la base de datos.
Evaluación de prueba: Satisfactoria.

Tabla 121. Prueba de aceptación 11: Desarrollo de la notificación de resolución de solicitud de adopción.

Prueba de aceptación	
Número: 11	Historia de usuario: 11
Nombre de historia: Desarrollo de la notificación de resolución de solicitud de adopción.	
Descripción: Como personal administrativo de la fundación quiero dar a conocer la aceptación o negación de una solicitud de adopción para que el cliente que ha enviado la solicitud pueda conocer la resolución	
Condiciones de ejecución: Ser parte del personal administrativo.	
Entrada / Pasos de ejecución: El administrador debe haber ingresado al aplicativo y seleccionado la opción de Solicitudes.	
Resultado esperado: Dado que el administrador selecciona Aprobar o Cancelar una resolución Cuando haya revisado los datos del posible adoptante Entonces el aplicativo web registra la resolución en la base de datos y le notifica al posible adoptante la resolución.	
Evaluación de prueba: Satisfactoria.	

Tabla 122. Prueba de aceptación 12: Acta de adopción.

Prueba de aceptación	
Número: 12	Historia de usuario: 12
Nombre de historia: Acta de adopción.	

Descripción: Como personal administrativo de la fundación quiero poder generar una acta física de adopción para posteriormente firmar el documento.
Condiciones de ejecución: Ser parte del personal administrativo.
Entrada / Pasos de ejecución: El administrador debe haber ingresado al aplicativo y seleccionado la opción de Solicitudes.
Resultado esperado: Dado que el administrador Aprueba una resolución Cuando haya decidido dar en adopción una mascota Entonces el aplicativo web generará una acta de adopción en formato PDF.
Evaluación de prueba: Satisfactoria.

Tabla 123. Prueba de aceptación 13: Interfaz de reportes.

Prueba de aceptación	
Número: 13	Historia de usuario: 13
Nombre de historia: Interfaz de reportes.	
Descripción: Como personal administrativo de la fundación quiero conocer el número de solicitudes de adopciones recibidas, las solicitudes aprobadas por mascota, el número de actividades realizadas por la fundación para conocer cuantitativamente los datos de la fundación.	
Condiciones de ejecución: Ser parte del personal administrativo.	
Entrada / Pasos de ejecución: El administrador debe haber ingresado al aplicativo y seleccionado la opción Dashboard.	
Resultado esperado: Dado que el administrador ingresa a la opción de Dashboard Cuando quiera conocer datos estadísticos de las adopciones y voluntariado Entonces el aplicativo web mostrará mediante gráficos los datos estadísticos..	
Evaluación de prueba: Satisfactoria.	

Tabla 124. Prueba de aceptación 14: Formulario para crear actividades de voluntariado.

Prueba de aceptación	
Número: 14	Historia de usuario: 14
Nombre de historia: Formulario para crear actividades de voluntariado.	

Descripción: Como personal administrativo de la fundación quiero gestionar la información de las diferentes actividades que realiza la fundación para incitar a las personas a ayudar en algunas de las actividades.
Condiciones de ejecución: Ser parte del personal administrativo.
Entrada / Pasos de ejecución: El administrador debe haber ingresado al aplicativo y seleccionado la opción Voluntariado.
Resultado esperado: Dado que el administrador ingresa a la opción de Voluntariado Cuando quiera registrar una nueva actividad para voluntariado Entonces el aplicativo web registrará los datos en la base de datos.
Evaluación de prueba: Satisfactoria.

Tabla 125. Prueba de aceptación 15: Catálogo de actividades para voluntariado.

Prueba de aceptación	
Número: 15	Historia de usuario: 15
Nombre de historia: Catálogo de actividades para voluntariado.	
Descripción: Como cliente del aplicativo web quiero conocer las actividades que se llevan a cabo dentro de la fundación para participar como voluntario.	
Condiciones de ejecución: Ser parte del personal administrativo o usuario registrado.	
Entrada / Pasos de ejecución: El administrador o usuario debe haber ingresado al aplicativo y seleccionado la opción Voluntariado.	
Resultado esperado: Dado que el administrador o usuario ingresa a la opción de Voluntariado Cuando quiera visualizar las actividades disponibles para voluntariado Entonces el aplicativo web mostrará en un listado todas las actividades que el administrador haya registrado.	
Evaluación de prueba: Satisfactoria.	

Tabla 126. Prueba de aceptación 16: Automatización de recepción de donaciones de dinero.

Prueba de aceptación	
Número: 16	Historia de usuario: 16

Nombre de historia: Automatización de recepción de donaciones de dinero.
Descripción: Como personal administrativo de la fundación quiero que la recolección de donaciones de dinero se automatice para que el dinero se vaya directamente a nuestra cuenta de PayPal.
Condiciones de ejecución: Ser parte del personal administrativo..
Entrada / Pasos de ejecución: El administrador debe ingresar sus credenciales de PayPal.
Resultado esperado: Dado que el administrador ingresar a su cuenta de PayPal Cuando quiera administrar las donaciones de dinero recibidas Entonces el aplicativo web le dirigirá a la cuenta de PayPal de la fundación.
Evaluación de prueba: Satisfactoria.

Modelo de aceptación tecnológica TAM

El Modelo de Aceptación de la Tecnología (TAM, por sus siglas en inglés: Technology Acceptance Model) es un marco teórico ampliamente utilizado en el campo de la investigación en sistemas de información y tecnología de la información. TAM se centra en la utilización y aceptación de una herramienta tecnología por parte de los usuarios [52].

La aceptación de la tecnología se fundamenta en dos criterios: la utilidad percibida y la facilidad de uso, los cuales constituyen componentes fundamentales del modelo [53].

Utilidad percibida (UP): Se refiere al grado en que una persona estima que el uso de una tecnología específica mejora su rendimiento laboral o la satisfacción con un servicio [53].

Facilidad de uso percibida (FUP): Se refiere a la percepción del usuario sobre la facilidad con la que puede aprender a usar la tecnología y la facilidad con la que puede interactuar con ella [53].

Cuestionarios de evaluación TAM

Los cuestionarios serán aplicados a un total de 4 miembros de la fundación de las cuales 2 miembros conforman el personal administrativo y 2 asistentes. Las respuestas establecidas para las preguntas estarán basadas en una escala Likert, tal y como se plantea en la Tabla 127.

Tabla 127. Escala de Likert.

Indicador	Valor asignado
Totalmente en desacuerdo	1
En desacuerdo	2
Neutral	3
De acuerdo	4
Totalmente de acuerdo	5

Cuestionario de utilidad percibida

En la Tabla 128, se presenta un cuestionario de 5 preguntas mediante el cual se recopila información para la evaluación del criterio de utilidad percibida.

Tabla 128. Cuestionario de utilidad percibida.

UP	Utilidad percibida
UP1	¿El aplicativo web me ayuda a gestionar de manera más eficiente la información sobre las mascotas rescatadas?
UP2	¿El aplicativo web me permite coordinar las actividades de voluntariado de manera más efectiva?
UP3	¿El aplicativo web me facilita la recepción de donaciones de dinero?
UP4	¿El aplicativo web permite gestionar el proceso de adopción de mascotas?
UP5	¿El aplicativo web me permite obtener reportes de maneras eficiente?

Resultados

En la Tabla 129, se presenta los resultados del cuestionario de utilidad percibida.

Tabla 129. Resultados del cuestionario de utilidad percibida.

Resultados de la utilidad percibida					
UP	Totalmente en desacuerdo	En desacuerdo	Neutral	De acuerdo	Totalmente de acuerdo
UP1	0%	0%	0%	25%	75%
UP2	0%	0%	25%	50%	25%
UP3	0%	0%	0%	50%	50%
UP4	0%	0%	0%	25%	75%
UP5	0%	0%	25%	50%	25%

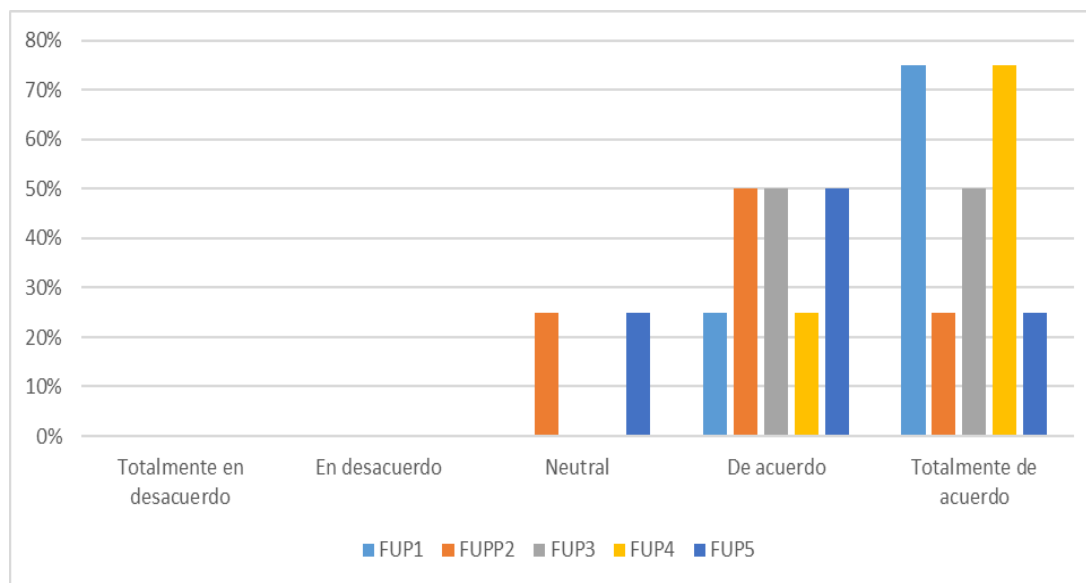


Figura 88. Resultados del cuestionario de utilidad percibida.

Conclusión

En general, las respuestas muestran una percepción positiva hacia el aplicativo web en cuanto al criterio de utilidad percibida. La mayoría de los encuestados están de acuerdo o totalmente de acuerdo en que el aplicativo es útil para la gestión de información

sobre mascotas rescatadas, la coordinación de actividades de voluntariado, la recepción de donaciones de dinero, la gestión del proceso de adopción y la generación de reportes. Esto sugiere que el aplicativo es bien recibido y visto como una herramienta eficiente para la fundación.

Cuestionario de facilidad de uso percibida

En la Tabla 130, se presenta un cuestionario de 5 preguntas mediante el cual se recopila información para la evaluación del criterio de facilidad de uso percibida.

Tabla 130. Cuestionario de facilidad de uso percibida.

FUP	Facilidad de uso percibida
FUP1	¿El aplicativo web es fácil, sencilla y comprensible para usar.?
FUP2	¿El registro de usuarios en el aplicativo web es intuitivo y sencillo?
FUP3	¿La gestión de adopciones de mascotas mediante el aplicativo web es fácil y comprensible?
FUP4	¿El registro de actividades para voluntariado es sencillo de realizarlo?
FUP5	¿La gestión de la información de las mascotas mediante el aplicativo web es comprensible y fácil?

Resultados

En la Tabla 131, se muestran los resultados del cuestionario aplicado de la facilidad de uso percibida.

Tabla 131. Resultados del cuestionario de facilidad de uso percibida.

Resultados de la Facilidad de uso percibida					
FUP	Totalmente en desacuerdo	En desacuerdo	Neutral	De acuerdo	Totalmente de acuerdo
FUP1	0%	0%	0%	25%	75%
FUP2	0%	0%	0%	75%	25%

FUP3	0%	0%	0%	50%	50%
FUP4	0%	0%	0%	75%	25%
FUP5	0%	0%	0%	50%	50%

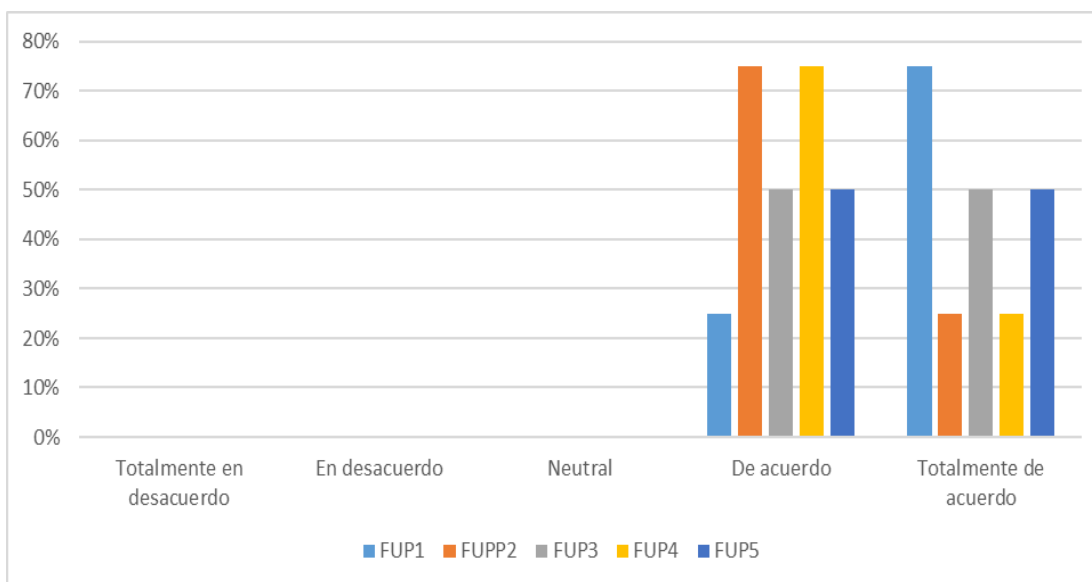


Figura 89. Resultados del cuestionario de facilidad de uso percibida.

Conclusión

En general, los resultados reflejan una percepción favorable en cuanto a la facilidad de uso del aplicativo web. La mayoría de los encuestados están de acuerdo o totalmente de acuerdo en que el aplicativo es fácil de usar, intuitivo en el registro de usuarios, sencillo en la gestión de adopciones y actividades de voluntariado, así como comprensible en la gestión de la información de las mascotas. Esto sugiere que el aplicativo ha sido bien diseñado en términos de usabilidad, lo que contribuye a una experiencia positiva para los usuarios.

3.2.5 Fase 5: Lanzamiento

Dadas las pruebas de aceptación de forma satisfactoria el aplicativo web se pasa a pruebas de testeo por medio del personal administrativo de la fundación de rescate animal Paquito, que son el personal encargado de dar uso al aplicativo web.

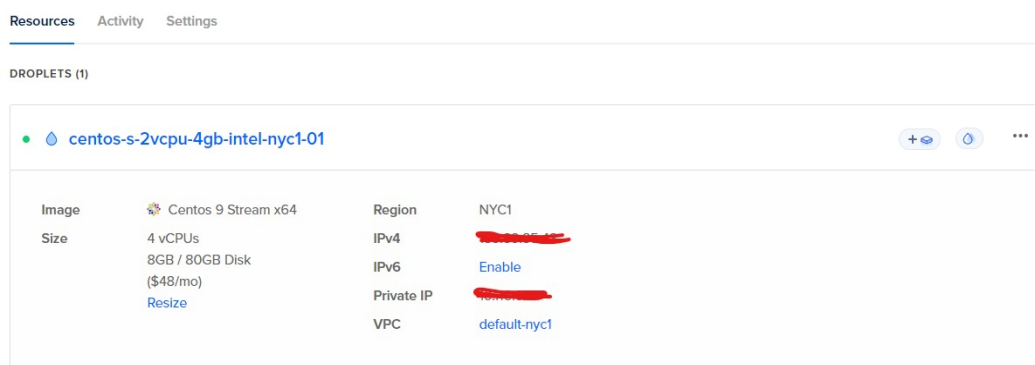
La entrega del proyecto se realiza al personal administrativo de la fundación de rescate animal Paquito en conjunto con la capacitación del manejo del aplicativo web. Se

capacita además sobre el uso de cada una de las funcionalidades que tiene el aplicativo y el ingreso al mismo, además de los roles que se manejan y los módulos de cada uno.

Una vez el proyecto fue entregado a la fundación de rescate animal Paquito, se procede a realizar la entrega de la documentación relacionada con el proyecto.

El manual de usuario del aplicativo web se encuentra en el Anexo B

El aplicativo web y la base de datos se encuentra en un servidor en DigitalOcean.



La dirección del aplicativo web es: <http://app.fundacionpaquito.com/login>

3.3 Análisis de tiempos estimados antes y después de la automatización de procesos.

En la Tabla 132 se realiza un análisis de los tiempos estimados en los procesos de rescate, adopción de mascotas, voluntariado y donaciones de dinero, comparando el antes y después de automatizarlos. Se muestran los tiempos estimados de cada proceso y el mejoramiento logrado con la automatización.

Para determinar el porcentaje del mejoramiento se usó a siguiente fórmula:

Mejora porcentual = (Tiempo estimado proceso sin automatizar – Tiempo estimado proceso automatizado) / Tiempo estimado proceso sin automatizar * 100.

Tabla 132. Análisis de tiempos estimados en los procesos.

N°	Proceso	Tiempo estimado proceso sin automatizar	Tiempo estimado proceso automatizado	Cantidad	Porcentaje de mejoramiento
1	Rescate de mascotas	5 horas	2 horas	1 rescate	60 %
2	Adopción de mascotas	48 horas	12 horas	1 adopción	75 %
3	Voluntariado	3 horas	0.20 horas	1 voluntariado	93.33 %
4	Donación de dinero	2 horas	0.10 horas	1 donación	95 %

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- Se determinaron los requerimientos funcionales para el desarrollo del aplicativo web mediante una encuesta a las personas que están en proceso de adopción de mascotas y una entrevista al personal administrativo de la fundación. Además, se obtuvo información necesaria para entender los procesos involucrados en las actividades que realiza la fundación y de esta manera realizar el modelado de los procesos para posteriormente ejecutar el desarrollo del aplicativo web.
- Se realizó una comparativa entre diferentes tecnologías open source para el desarrollo del aplicativo web, de las que se seleccionó Angular como framework para la creación del frontend y el framework .NET Core para el backend. Esta elección se fundamenta en la robustez y flexibilidad que ofrecen ambas herramientas. Angular facilita la construcción de interfaces de usuario dinámicas y altamente interactivas, mientras que .NET Core brinda una arquitectura escalable y eficiente para gestionar las operaciones del servidor.
- A través de la implantación del aplicativo web se pudo cumplir con las necesidades de la fundación en cuanto a la gestión de rescate, adopciones, voluntariado y la automatización de donaciones de dinero. Esta implementación ha permitido una mayor eficiencia en los procesos vinculados a estas actividades.

4.2 Recomendaciones

- Al tratarse de una aplicación web, se recomienda mantener los navegadores actualizados para asegurar una experiencia óptima en todo momento. Asimismo, es recomendable mantener actualizadas las herramientas de desarrollo, ya que los desarrolladores suelen corregir errores, optimizar tareas y resolver problemas de compatibilidad de manera constante.

- Se recomienda llevar a cabo mantenimientos periódicos en la base de datos, junto con la realización de respaldos, dado que la fundación gestiona una considerable cantidad de información.
- Se sugiere seguir evaluando en el futuro la gestión de otros procesos como el de venta de productos para mascotas, la localización de mascotas perdidas, entre otros. Para, de ser el caso agregar estos posibles módulos al aplicativo web ya que sería de gran ayuda gestionar sus actividades a partir de un solo aplicativo.

REFERENCIAS BIBLIOGRÁFICAS

- [1] P. D. E. Titulación, L. María, and S. Viteri, “Desarrollo De Un Sistema Web De Subastas Dinámicas De Los Objetos Manuales Elaborados Por Adultos Mayores Y Personas Discapacitadas De La Fundación Fe Y Acción,” *Univ. GUAYAQUIL Fac.*, 2020.
- [2] S. Yildırım, “Marketing Social para la Fundación Cuesta Holguín,” no. 21, pp. 1–9, 2021.
- [3] V. A. C. RIOFRÍO, “ESTUDIO DE LAS ESTRATEGIAS DE MARKETING Y SU INCIDENCIA EN EL FORTALECIMIENTO INSTITUCIONAL DE LA FUNDACIÓN FUDIS ALBERGUE, EN LA CIUDAD DE QUITO.,” *Ekp*, vol. 13, 2019.
- [4] J. A. C. Villegas, “APLICACIÓN WEB PARA LA GESTIÓN DE SERVICIOS, NOTIFICACIONES DE PAGO Y GEOLOCALIZACIÓN DE VIVIENDAS DE LOS CLIENTES EN LA EMPRESA OPTYNET,” *Univ. Tec. Ambato*, pp. 1–23, 2023.
- [5] E. J. M. Rojas, “SISTEMA WEB BASADO EN ARQUITECTURA MVC PARA LA GESTIÓN DE INSUMOS PRIMARIOS PARA LA PLANIFICACIÓN DEL SISTEMA ELÉCTRICO DE DISTRIBUCIÓN DE LA EMPRESA ELÉCTRICA AMBATO (EEASA),” *Univ. Tec. Ambato*, pp. 2013–2015, 2022.
- [6] S. Hurtado, “Aplicación Web Usando El Framework ModularMultiplataforma .Net Core Para El Refuerzo Académico De La Educación General Básica Virtual En La Unidad Educativa ‘González Suárez’ De La Ciudad De Ambato,” 2022, [Online]. Available: <https://repositorio.uta.edu.ec/jspui/handle/123456789/34829>
- [7] E. Armando and T. Cepeda, “APLICACIÓN WEB CON TECNOLOGÍAS OPEN SOURCE PARA MEJORAR LA GESTIÓN DEL CONSULTORIO DENTAL ‘ODONTOLOGICA ORAL FAMILY,’” *Univ. Guayaquil*, 2021.
- [8] T. Z. O. FARITH, “APLICACIÓN WEB-MÓVIL PARA LA GESTIÓN DE ADOPCIONES DE ANIMALES RESCATADOS EN LA FUNDACIÓN RESCATE ANIMAL,” *Energies*, vol. 6, no. 1, pp. 1–8, 2018, [Online]. Available: <http://journals.sagepub.com/doi/10.1177/1120700020921110%0Ahttps://doi.org/10.1016/j.reuma.2018.06.001%0Ahttps://doi.org/10.1016/j.arth.2018.03.044%0Ahttps://reader.elsevier.com/reader/sd/pii/S1063458420300078?token=C039B8B13922A2079230DC9AF11A333E295FCD8>
- [9] J. Pajuelo, “Aplicación web para la gestión de la información de los Programas Sociales en la Municipalidad Provincial del Calla,” pp. 1–149, 2019.
- [10] S. Pérez, J. Quispe, F. Mullicundo, and D. Lamas, “Herramientas Y Tecnologías Para El Desarrollo Web Desde El Frontend Al Backend,” *XXIII Work. Investig. en Ciencias la Comput.*, pp. 347–350, 2021, [Online]. Available: <http://sedici.unlp.edu.ar/handle/10915/120476>
- [11] E. P. R. TAPIERO, “APLICATIVO WEB DE GESTION DE INFORMACION PARA EL SEMILLERO ‘KERVERUX,’” no. 21, pp. 1–9, 2018.
- [12] D. R. Mancilla, *Gestion Organizacional elemtos para su estudio*. Universidad Iberoamericana. [Online]. Available: <https://books.google.es/books?id=ckXS4Eg-UI0C&lpg=PA11&ots=uddiSbh4-k&dq=gestión organizacional concepto&lr&hl=es&pg=PP1#v=onepage&q=gestión organizacional concepto&f=false>

- [13] M. Armijo, *Planificación Estratégica e indicadores de desempeño en el sector público*. ILPES. [Online]. Available: <https://n9.cl/haicx>
- [14] E. Saks, “JavaScript Frameworks: Angular vs React vs Vue.,” *Univ. Appl. Sci.*, p. 42, 2019.
- [15] H. Zuse, *A Framework of Software Measurement*. Walter de Gruyter. [Online]. Available: <https://books.google.es/books?hl=es&lr=&id=8a4fAAAAQBAJ&oi=fnd&pg=PA1&dq=framework++software&ots=HIpYcIsZ5y&sig=pSk7UWv3O3LFIETH4PbbZ5tPxTg#v=onepage&q&f=false>
- [16] J. Feller, “paradigm SOURCE DEVELOPMENT PARADIGM,” pp. 58–69, 2000.
- [17] A. Antamba, “Desarrollo Del Sistema Web Para La Gestión Académica De La Unidad Educativa ‘Modesto a. Peñaherrera’. Utilizando Las Herramientas Vue.Js Y Spring Framework.,” pp. 1–99, 2018, [Online]. Available: https://www.researchgate.net/profile/Pedro-Larranaga/publication/268291232_Tema_8_Neutral_Neural_Neural/links/55b7b5c408ae9289a08c0c68/Tema-8-Redes-Neurales.pdf
- [18] P. Rawat and A. N. Mahajan, “ReactJS: A Modern Web Development Framework,” *Int. J. Innov. Sci. Res. Technol.*, vol. 5, no. 11, pp. 698–702, 2020.
- [19] A. Freeman, *Pro ASP.NET Core*, 6th ed. Apress, 2017. [Online]. Available: <https://books.google.es/books?hl=es&lr=&id=iEQWDQAAQBAJ&oi=fnd&pg=PR5&dq=asp.net+core&ots=tMSNboD4gi&sig=dx6pcvCVMgTbTrSyD-aKwnz8wDE#v=onepage&q=asp.net+core&f=false>
- [20] A. Arsaute, F. Zorzan, M. Daniele, A. González, and M. Frutos, “Generación automática de API REST a partir de API Java, basada en transformación de Modelos (MDD),” *Work. Investig. en Ciencias la Comput.*, no. Mdd, pp. 629–633, 2018, [Online]. Available: http://sedici.unlp.edu.ar/bitstream/handle/10915/67777/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y
- [21] Óscar Gómez, “Bases de datos Versión 1.1,” 2019, [Online]. Available: https://d1wqtxts1xzle7.cloudfront.net/52350086/Base_de_Datos_Oscar_Gomez_2012-libre.pdf?1490754368=&response-content-disposition=inline%3B+filename%3DBases_de_datos.pdf&Expires=1690433982&Signature=N5t0i8ZH1aqEUfqF~aALBxIy7p5OlbY8Cq1if~I0gotP~81Rx5UUxsGOW
- [22] G. Rodriguez, J and Guardo, *MySQL introduces the essential concept*, vol. 3, no. 1. 2021.
- [23] C. Giménez, G. Braun, L. Cecchi, and P. R. Fillottrani, “Una arquitectura cliente-servidor para modelado conceptual asistido por razonamiento automático,” *XVIII Work. Investig. en Ciencias la Comput.*, pp. 692–696, 2016, [Online]. Available: <http://sedici.unlp.edu.ar/handle/10915/53044>
- [24] J. E. Serrano, *Gestión Logística y Comercial*. [Online]. Available: https://books.google.es/books?id=zQv_AAAAQBAJ&lpg=PR7&ots=NDDI8FRwcy&dq=sistema+gestion+de+actividades&lr=&hl=es&pg=PP1#v=onepage&q=sistema+gestion+de+actividades&f=false
- [25] E. Maida and J. Pacienza, “Metodologías de desarrollo de software,” *Bibl. Digit. la Univ. Católica Argentina*, p. 117, 2018, [Online]. Available: <http://bibliotecadigital.uca.edu.ar/repositorio/tesis/metodologias-desarrollo->

software.pdf

- [26] S. M. Velásquez, J. D. Vahos Montoya, M. E. Gómez Adasme, E. J. Restrepo Zapata, A. A. Pino, and S. Londoño Marín, “Una revisión comparativa de la literatura acerca de metodologías tradicionales y modernas de desarrollo de software,” *Rev. CINTEX*, vol. 24, no. 2, pp. 13–23, 2019, doi: 10.33131/24222208.334.
- [27] J. Canos, P. Letelier, and C. Penades, “Metodologías Agiles en el Desarrollo de Software,” *Taller Metodol. Agil. en el Desarro. Softw.*, p. 59, 2003, [Online]. Available: <https://s3.amazonaws.com/academia.edu.documents/34257802/actas.pdf>
- [28] F. De Ciencias, A. Calderón, S. Dámaris, V. Rebaza, and J. Carlos, “Universidad Nacional de Trujillo,” *Lexus*, vol. 4, no. None, p. 37, 2007.
- [29] L. P. Raut, R. B. Wakode, P. Talmale, and B. Steels Pvt Ltd, “Overview on Kanban Methodology and its Implementation Overview on Kanban Methodology and its Implementation,” *Int. J. Sci. Res. Dev.*, vol. 3, no. 02, pp. 2321–0613, 2015, [Online]. Available: www.ij srd.com
- [30] S. Sachdeva, “Scrum Methodology,” *Int. J. Eng. Comput. Sci.*, vol. 5, no. 16792, pp. 16792–16800, 2016, doi: 10.18535/ijecs/v5i6.11.
- [31] Wenqing Xu, “Benchmark Comparison of JavaScript Frameworks React, Vue, Angular and Svelte,” *A Res. Pap. Submitt. to Univ. Dublin, Partial fulfilment Requir. degree Master Sci. Interact. Digit. Media*, 2021, [Online]. Available: <https://www.scss.tcd.ie/publications/theses/diss/2021/TCD-SCSS-DISSERTATION-2021-020.pdf>
- [32] M. Levlin, “DOM benchmark comparison of the front-end JavaScript frameworks React, Angular, Vue, and Svelte,” p. 95, 2020.
- [33] Zadhid Powell, “Angular vs React: Una Comparación En Profundidad,” *kinsta*, 2023. <https://kinsta.com/es/blog/angular-vs-react/>
- [34] M. P. Jelica Cincović, “Comparison: Angular vs. React vs. Vue. Which framework is the best choice?,” *Jelica Cincović, Marija Punt*, 2020.
- [35] E. Wohlgethan, “Supporting Web Development Decisions by Comparing Three Major JavaScript Frameworks: Angular, React and Vue.js,” *Bachelorarbeit*, pp. 14–21, 2018.
- [36] E. Haro, T. Guarda, A. Omar, Z. Peñaherrera, and N. Quiña, “Desarrollo backend untuk aplikasi web , Layanan Web Restful : Node . js vs Spring Boot,” pp. 309–322, 2019.
- [37] H. K. Dhalla, “A Performance Comparison of RESTful Applications Implemented in Spring Boot Java and MS.NET Core,” *J. Phys. Conf. Ser.*, vol. 1933, no. 1, 2021, doi: 10.1088/1742-6596/1933/1/012041.
- [38] L. Pantoja and C. Pardo, “Publicaciones e investigación revista especializada en ingeniería : journal specializing in engineering,” *Publicaciones e Investig.*, vol. 10, no. 0, pp. 129–142, 2007, [Online]. Available: <http://hemeroteca.unad.edu.co/index.php/publicaciones-e-investigacion/article/view/1592/1924>
- [39] IBM, “Seguridad en aplicaciones Spring Boot.” <https://www.ibm.com/docs/es/cics-ts/6.1?topic=applications-security-in-spring-boot>
- [40] V. A. Dorokhin, “NET core technology in scientific tasks,” *CEUR Workshop Proc.*, vol. 2267, no. Grid, pp. 166–169, 2018.
- [41] IBM, “¿Qué es Java Spring Boot?” <https://www.ibm.com/es-es/topics/java-spring->

ANEXOS

Anexo A. Formato de encuesta aplicada a personas que está en proceso de adopción de mascotas.



ENCUESTA

freddyread8@gmail.com [Cambiar de cuenta](#) 

 No compartido

* Indica que la pregunta es obligatoria

Cómo usuario externo, ¿tiene usted conocimiento del proceso a seguir para *
efectuar una adopción de mascotas en la fundación de rescate animal "Paquito"?

Si

No

En base a su experiencia, ¿cómo considera el proceso de adopción de mascotas *
con el que cuenta la fundación de rescate animal "Paquito"?

- Pèsima
- Mala
- Regular
- Buena
- Excelente

¿En qué medida considera usted el nivel de eficiencia del proceso de solicitud de *
adopción de mascotas?

- Muy Ineficiente
- Ineficiente
- Regular
- Eficiente
- Muy Eficiente

En su opinión, ¿considera usted que la fundación de rescate animal "Paquito" proporciona información necesaria y detallada de las mascotas que están en adopción? *

- Totalmente en desacuerdo
- En desacuerdo
- Neutral
- De acuerdo
- Totalmente de acuerdo

¿Por favor, indique en qué medida considera importante que la fundación de rescate animal "Paquito" realice un adecuado seguimiento de las mascotas que han sido dadas en adopción? *

- Nada importante
- De poca importancia
- Moderadamente importante
- Importante
- Muy importante

¿Defina en base a la siguiente escala el nivel de recursos económicos y el espacio físico idóneo con el que usted cuenta para proporcionar un hogar adecuado a una mascota adoptada? *

- Muy Insuficiente
- Insuficiente
- Regular
- Suficiente
- Abundante

¿Conoce usted de aplicaciones o sitios web donde se pueda realizar adopciones de mascotas? *

- Si
- No

¿En base a sus aptitudes adquiridas, defina el nivel de dominio con el que usted cuenta en base al uso de la tecnología? *

- Muy Difícil
- Difícil
- Regular
- Fácil
- Muy Fácil

¿A través de qué medio considera que la difusión de adopciones de mascotas puede alcanzar a más personas interesadas en realizar una adopción? *

- Redes Sociales
- Aplicativos Web
- Radio
- Televisión
- Carteles exteriores

En general, ¿cómo calificaría su experiencia al adoptar una mascota en la fundación "Paquito"? *

- Totalmente Insatisfecho
- Insatisfecho
- Algo Satisfecho
- Satisfecho
- Totalmente Satisfecho

Enviar

Borrar formulario

Anexo B Manual de usuario.

Ingreso al aplicativo web

En la interfaz de inicio de sesión se solicita al usuario ingresar el correo y contraseña correctamente.



Ingresar

No tienes una cuenta? [Regístrate aquí](#)

 Recordarme[Olvidé mi contraseña](#)

Olvidé mi contraseña

Si el usuario no recuerda su contraseña, en la interfaz de cambio de contraseña se debe ingresar el correo con el cual está registrado, su número de cédula y la nueva contraseña. Si la información es válida se cambiará la contraseña.



Cambio de contraseña

Ingresa la información requerida a continuación, si la información es válida, se cambiara la contraseña

Regístrate aquí.

Para registrarse en el aplicativo web el usuario debe ingresar todos los datos que se solicita en el formulario y Aceptar los términos y condiciones, una vez llenado los datos debe dar clic en Registrarme.

Regístrate

y adopta a tu primera mascota




Acepto los [términos y condiciones](#)


Registrarme

Página principal


En la página principal se muestra el menú de lado izquierdo de la pantalla y el nombre del usuario en la parte superior derecha.


 **Fundación- Paquito**

Admin

 Dashboard

 Mis Mascotas

 Solicitudes

 Voluntariado

 Usuarios

 Salir

Dashboard

En esta interfaz se encuentra las gráficas estadísticas de solicitudes, número de solicitudes, activas, solicitudes aprobadas y solicitudes canceladas. Además, encontraremos estadísticas de las solicitudes de adopción y actividades de voluntariado.



Mis Mascotas

En esta interfaz encontraremos un listado de mascotas que están ingresadas en la fundación, además encontraremos la opción de ingresar información de una nueva mascota, y de adoptar una mascota.

Listado de mascotas

En el listado de mascotas encontraremos datos como tipo, nombre, raza y el estado ya sea adoptado o en proceso de adopción.

♥ Mis mascotas

Crear una mascota

Adoptar otra mascota

TIPO	NOMBRE	RAZA	ESTADO
Perro	Guido	Balkan Hound	Adoptado
Gato	Hugo	Korat	Adoptado
Perro	Lazy	Australian silky terrier	Adoptado
Gato	Loba	Abisinio	Adoptado
Perro	Bobby	Ariegeois	Adoptado
Gato	Ruby	Abisinio	Adoptado
Perro	waco	Pastor Alemán	
Gato	Toby	Africano doméstico	

Crear una mascota

En esta interfaz debemos ingresar todos los datos que se nos solicita de una mascota rescatada. Una vez ingresado todos los datos damos clic el botón Crear Mascota y de esta manera la mascota se registrará en el aplicativo.

Crear Mascota

Nombre de la mascota

Tipo de mascota

Perro

Raza

Sexo

Macho

Tamaño

Pequeño

Fecha de nacimiento

Día

Mes

Año

Dirección

Esterilizado

Crear Mascota

Seguimiento

En la opción de seguimiento podemos añadir cualquier observación respecto al rescate o adopción de la mascota.

Seguimiento Reconocimiento Enfermedades o alergias Fotos

Anota aquí cualquier novedad que suceda con tu mascota...

Escribe un comentario...

Guardar

Reconocimiento

En la interfaz de reconocimiento se agregarán datos de la revisión realizada por un veterinario a la mascota. Estos datos incluyen el peso, diagnóstico, pruebas realizadas y los resultados de la misma.

Nuevo Reconocimiento

Peso (kg)

Descripción del reconocimiento

Diagnostico

Pruebas realizadas

Resultados

Guardar Reconocimiento

Enfermedades o alergias

En esta interfaz ingresaremos las enfermedades o alergias que tenga la mascota. En caso de que no haya se dejará vacía esta opción.

Enfermedad

Enfermedad

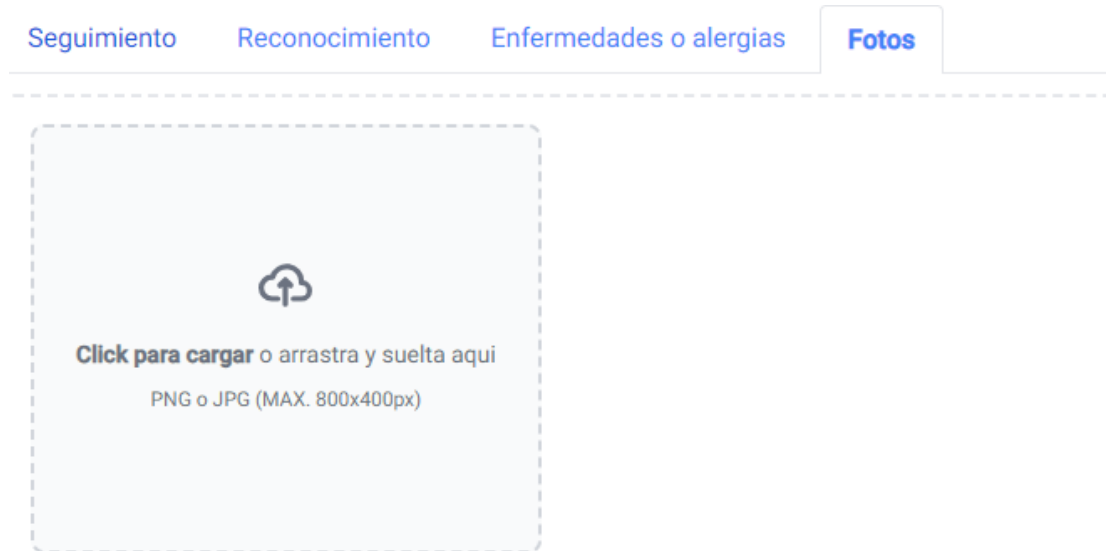
Descripción

Escribe aquí una descripción de la enfermedad...

Crear enfermedad

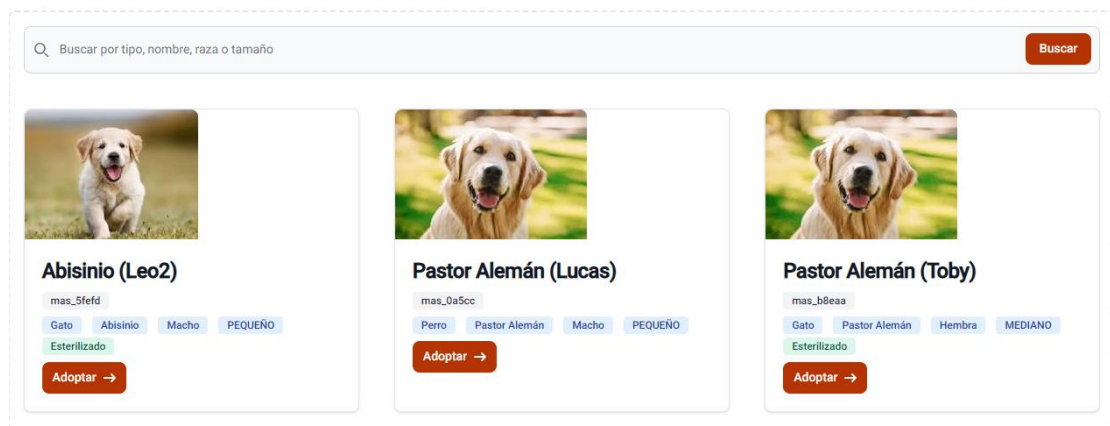
Fotos

En la sección de fotos podremos subir fotos de la mascota. Para hacerlo demos dar clic o arrastrar la foto para subirla al aplicativo.



Adoptar una mascota

En la interfaz de mascotas se nos muestra una galería con las mascotas que están ingresadas en la fundación. Para adoptar una de estas mascotas damos click en Adoptar.



Formulario de adopción

En esta interfaz demos ingresar todos los datos que se nos solicita para poder enviar la solicitud de adopción. Una vez ingresado todos los datos damos clic en Enviar solicitud de adopción.



Situación Personal

Número de Cédula	Nombres Completos	
<input type="text" value="0104358839"/>	<input type="text" value="Admin Admin"/>	
Dirección	Teléfono	
<input type="text" value="Cuenca"/>	<input type="text" value="0998763543"/>	
Email	Edad	Estado Civil
<input type="text" value="pablo.ab34@gmail.comm"/>	<input type="text" value="0"/>	<input type="text" value="Soltero/a"/>
¿Trabaja?		
<input type="radio"/> SI		
<input type="radio"/> NO		
Nombre del sitio de trabajo	Actividad	
<input type="text"/>	<input type="text"/>	

Solicitudes


En la interfaz de solicitudes podremos visualizar todas las solicitudes Activas, Aprobadas o Canceladas.

Solicitudes de Adopción

FECHA	BENEFICIARIO	MASCOTA	ETAPA
9 ene 2024, 21:40:06	Juan Reyes	Guido - Balkan Hound	APROBADO
4 ene 2024, 23:58:30	Juana Perez	topo - Pastor Alemán	APROBADO
4 ene 2024, 10:00:00	Freddy Remache	Leo3435 - Abisinio	APROBADO
3 ene 2024, 21:25:04	Juana Perez	Hugo - Korat	APROBADO
3 ene 2024, 19:57:09	Juana Perez	Lazy - Australian silky terrier	APROBADO
1 ene 2024, 22:44:19	Isabel Torres	Leo22 - Abisinio	APROBADO
14 dic 2023, 6:58:06	Freddy Remache	waco1 - Pastor Alemán	APROBADO
30 nov 2023, 23:38:19	Freddy Remache	Bobby - Ariegeois	APROBADO

Al dar clic sobre una de las solicitudes de la lista se nos desplegará la siguiente interfaz en donde encontraremos un resumen de la información de la mascota y del solicitante.

En esta interfaz además podremos cambiar el estado de una solicitud a Aprobado o Cancelado.


 **Solicitud N-sol_80dc7** RECIBIDO Cancelar

Observaciones

[Resumen](#) [Formulario](#)

Creado: 8 sept 2023

Mascota



Pastor Alemán (Angelito)

Perro Macho MEDIANO

Solicitante

Admin Admin
Cédula: 0104358839
Estado Civil: soltero
Teléfono: 0998763543

Comentarios

Escribe un comentario...

Guardar

Formulario de la solicitud.

En esta interfaz encontraremos la información que el solicitante a ingresado en el formulario de adopción. Además, en la parte superior derecha encontramos un botón de Imprimir que generará la solicitud en formato PDF.

Informacion personal Imprimir

Detalles del solicitante.

Cédula	0104358839
Nombres	Admin Admin
Dirección	Ciudadela Uncovia
Teléfono	4068754
Email	pablo.ab34@gmail.com
Edad	30 Años
Estado civil	Soltero

Voluntariado.

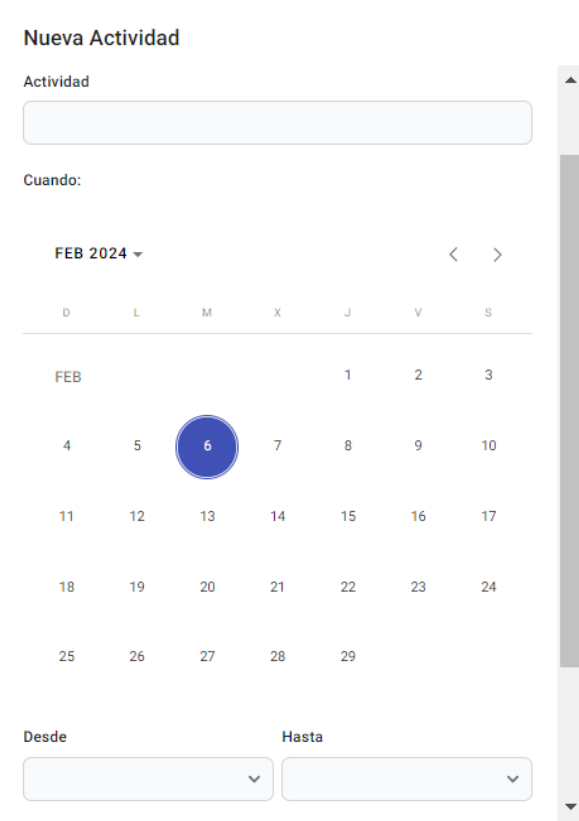
En la interfaz de voluntariado encontraremos una lista de actividades en las cuales los usuarios se pueden inscribir para realizar voluntariado. Para participar de una actividad únicamente se debe dar clic en Participar.

♥ **Voluntariado** Crear una actividad

FECHA	TIPO	ACTIVIDAD	DURACIÓN	PARTICIPANTES	ACCIÓN
2 feb 2024	Alimentación de las mascotas	test	16/2/24, 10:30 16/2/24, 9:45 -45 minutos	2 Personas max(100)	Realizada
2 feb 2024	Alimentación de las mascotas	paseo por el parque calderon	9/2/24, 10:30 9/2/24, 10:00 -30 minutos	1 Personas max(40)	Realizada
2 feb 2024	Alimentación de las mascotas	dar de comer	16/2/24, 9:45 16/2/24, 10:15 30 minutos	1 Personas max(10)	Realizada

Crear una actividad.

En la interfaz de crear una actividad debemos ingresar todos os datos requeridos para crear una nueva actividad para voluntariado. Una vez creada la actividad se añadirá automáticamente a la lista de actividades.



The screenshot shows a form titled "Nueva Actividad". It includes a text input field for "Actividad", a "Cuando:" section with a calendar for "FEB 2024", and two dropdown menus for "Desde" and "Hasta". The calendar shows the 6th of February selected. A vertical scrollbar is visible on the right side of the form.

Donaciones

Para realizar una donación debemos ingresar la cantidad que deseamos donar y presionar en el botón Donar con PayPal en caso de que queramos donar desde PayPal, en caso de que queramos donar con una tarjeta de débito o crédito demos dar clic en Donar con tarjeta de crédito o de débito.

Donar a
Fundacion Paquito

\$0
USD

Hacer esta donación mensualmente.

 **Donar con PayPal**

Donar con tarjeta de crédito o de débito