

Sistema para el control de actividades orientadas a mascotas.



**UNIVERSIDAD DISTRICTAL  
FRANCISCO JOSÉ DE CALDAS**

Presentado por:  
Andres Mauricio Acosta Pulido  
Código Académico:  
20142020111

Universidad Distrital Francisco José de Caldas  
Facultad de Ingeniería  
Pregrado en Ingeniería de Sistemas  
Fundamentos de Ingeniería de Software  
Noviembre de 2017  
Bogotá D.C.

# Proyecto final

Sistema para el control de actividades orientadas a mascotas.



**UNIVERSIDAD DISTRITAL  
FRANCISCO JOSÉ DE CALDAS**

Presentado por:

Andres Mauricio Acosta Pulido

Presentado a:

Julio Barón Velandia

Doctor en Ingeniería Informática

Universidad Distrital Francisco José de Caldas

Facultad de Ingeniería

Pregrado en Ingeniería de Sistemas

Fundamentos de Ingeniería de Software

Noviembre de 2017

Bogotá D.C.

# Tabla de contenido

<b>Introducción</b>	<b>4</b>
<b>Objetivos</b>	<b>4</b>
Objetivo General	4
Objetivos Específicos	4
<b>Desarrollo</b>	<b>5</b>
Origen del proyecto	5
Selección de proyecto	7
Descripción del problema	7
Requisitos	7
Actores	10
Casos de Uso	10
Descripción Básica Casos de Uso	11
Matrices de trazabilidad	13
Diagrama de Casos de Uso Base	17
Diagramas de Casos de Uso	18
Descripción Detallada Casos de Uso Base	19
Diagramas	24
Diagramas de clases	24
Diagramas de estado	26
Diagramas de secuencia	27
Diagramas de actividades	28
Diagramas de componentes	30
Agenda de trabajo	30
<b>Tecnología a utilizar.</b>	<b>31</b>
MongoDB	31
Express	31
AngularJS	31
NodeJS	31
MEAN	32
PROTOTIPO SLIDE	32
COSTOS.	34
<b>Conclusiones</b>	<b>34</b>
<b>Bibliografía</b>	<b>34</b>
<b>Recursos Electrónicos.</b>	<b>34</b>

# Introducción

Las actividades que se realizan en la ciudad de Bogotá para el control de vacunación y planes para las mascotas se ve afectada por la falta de conocimiento de ellas, por ende en este documento se presenta el modelamiento de un sistema de software para la gestión de planes de animales, en el cual se busca la implementación adecuada de un proyecto de software utilizando elementos propios de la ingeniería de software como la aplicación del Proceso Unificado de Desarrollo Software (RUP).

Para esto se propone el siguiente proyecto, en el que se presenta una forma más ágil y funcional para esta problemática de falta de conocimiento de actividades para mascotas y así suplir las necesidades y obligaciones que refleja el tener una mascota en buenas condiciones de salud, todo esto plenamente con fines educativos, siendo como primera versión un software con licencia GNU Linux versión 3.0.

## Objetivos

### Objetivo General

Desarrollar la primera versión del aplicativo “*Sistema para el Control de actividades orientadas a mascotas*” que cumpla con los requerimientos establecidos en las fases de análisis de requisitos, diseño e implementación, para suplir las necesidades que actualmente se presentan en el desconocimiento de planes para vacunación de animales.

### Objetivos Específicos

1. Determinar los requerimientos y casos de uso del sistema, analizando las necesidades y falencias en el mismo, para establecer una priorización de los mismos y determinar el alcance de la solución
2. Modelar los componentes del sistema y sus características fundamentales, como lo son los estados, mensajes y relaciones, para completar una definición del sistema.
3. Implementar un sistema que permita a las personas realizar actividades para planes de mascotas, facilitando así el conocimiento de actividades que pueden haber para las mascotas.

# Desarrollo

## Origen del proyecto

Para determinar el proyecto a realizar se llevó a cabo una lluvia de ideas durante una de las sesiones de clase, con la presencia del docente y la mayoría de los equipos de trabajo. Cada equipo expuso su propuesta para que posteriormente mediante una votación se decidiera el proyecto a desarrollar por todos los grupos.

Las propuestas fueron:

Número de propuesta	Nombre	Descripción corta
1	Recolección de basuras	Implementación de código QR para premiar a los usuarios que reciclen he incentivar a los ciudadanos por una ciudad más limpia.
2	Clases privadas	Plataforma donde las personas que deseen tomar clases de forma personalizada en cualquier materia lo puedan realizar de forma más eficiente.
3	Estaciones turísticas- Realidad aumentada	Hacer uso de la realidad aumentada para que turistas puedan conocer lugares a lo cuales deseen ir, esto sin necesidad de realizar viajes extensos. Se da un alcance tecnológico a estos lugares a través de esta tecnología.
4	Problema huecos en las vías	Diseño de una aplicación para reportar huecos, basura y otros que se encuentre en las vías de la capital, los reportes serán atendidos por las entidades competentes quienes con uso de la aplicación para ser informados determinarán qué acciones realizar.
5	Puntos críticos de seguridad y accidentes	Aplicación que permita reportar lugares inseguros de la ciudad para determinar estadísticamente el nivel de inseguridad, con esta información, las autoridades a su vez y haciendo uso de ésta, determinarán las acciones a tomar para disminuir el índice de inseguridad.
6	Tiempos recorrido llegada empresa-estudio	Aplicación que permita determinar el tiempo que se tarda una persona en realizar un recorrido desde origen a destino en buses de transporte público, de esta forma mejorar el sistema de cobro en la ciudad y escoger mejores rutas
7	Control de electrodomésticos en forma remota	Controlar a través de un dispositivo los electrodomésticos de nuestro hogar, para mejorar la vida humana

8	Control actividades orientadas a las mascotas	A través de una aplicación informar a usuarios actividades que se realicen para mascotas, entre las que se encuentran: Vacunación, desparasitación, competencias, etc..
9	Seguimiento prenatal en zonas vulnerables (charlas, conferencias)	Software que permita comunicar y persuadir a personas de zonas vulnerables de la importancia del un control prenatal, ayudando a disminuir embarazos a temprana edad y familias numerosas no sustentables.
10	Sugerencias planes fines de semana	Software que permite a usuarios y organizadores recibir y emitir respectivamente información acerca de planes disponibles para realizar los fines de semana, estos incluirán fecha , lugar y precio, con la opción de armar el plan de acuerdo al presupuesto con el que se disponga.
11	Integración de emisoras en sitios públicos	Software que permita facilitar al usuario disfrutar de emisoras “libres” o “abiertas”.
12	Botón de pánico para seguridad	Aplicación que permita dar una alerta cuando el usuario se encuentre bajo condiciones de inseguridad o vulnerabilidad a su integridad en vías públicas, esto permitirá dar aviso a las autoridades y/o conocidos.

Con el fin de llegar a un acuerdo grupal entre todos los estudiantes se someten las anteriores propuestas a una votación las cuales fueron así:

Número de propuesta	Nombre	Número de votos
1	Recolección de basuras	2
2	Clases privadas	6
3	Estaciones turísticas- Realidad aumentada	1
4	Problema huecos en las vías	0
5	Puntos críticos de seguridad y accidentes	3
6	Tiempos recorrido llegada empresa-estudio	0
7	Control de electrodomésticos en forma remota	0
8	Control actividades orientadas a las mascotas	7
9	Seguimiento prenatal en zonas vulnerables (charlas,	0

	conferencias)	
10	Sugerencias planes fines de semana	7
11	Integración de emisoras en sitios públicos	1
12	Botón de pánico para seguridad	1

Debido a que se observa un empate del número de propuesta 8 y 10 se realiza nuevamente una votación contando un voto por grupo, esto para poder determinar el proyecto a realizar.

Número de propuesta	Nombre	Número de votos
8	Control actividades orientadas a las mascotas	5
10	Sugerencias planes fines de semana	4

Al final de la sesión de clases, al ver que los dos proyectos fueron acogidos por casi la mitad de los estudiantes cada uno, se da la posibilidad de trabajar en alguno de los dos proyectos, terminando así la sesión de clases a las 12:00pm.

### Selección de proyecto

El proyecto a escoger es el número 8, el cual es nombrado como “*Sistema para el Control de actividades orientadas a mascotas*”.

### Descripción del problema

El sistema permitirá a los usuarios registrar eventos de vacunación, desparasitación o cualquier tipo de actividad relacionada a mascotas con el fin de mejorar la falta de conocimiento de planes para vacunación y actividades de animales domésticos (perros y gatos).

El sistema trabajará a partir de la información generada por los usuarios, ya que estos son los que colocarán, modificarán o buscarán eventos de planes de mascotas.

### Requisitos

En la Tabla 1 se listan los requisitos del sistema. Se establece su prioridad en un rango del 1-5, donde 1 es poco relevante y 5 es fundamental. Se determina si es funcional (F) o no funcional (NF) y el carácter, que puede ser obligatorio (OB), opcional (OP) y deseado (DE). Adicionalmente, en la columna contacto se registra el departamento administrativo o

estamento que puede brindar información sobre el requisito establecido. Por ejemplo, el contacto Andres Mauricio Acosta Pulido, puede asesorar con los requerimientos de acceso.

ID	Nombre Requerimiento	Descripción	Contacto	OB, OP, DE	F o NF	Prioridad
R1	Acceso a la plataforma	El sistema debe permitir al usuario acceder al sistema de forma remota.	Andres Acosta teléfono: 3134301879	OB	NF	4
R2	Actualización de evento	El sistema debe permitir al creador del evento, modificar la información de la actividad.	Andres Acosta teléfono: 3134301879	OB	F	5
R3	Control de registro	El sistema controlará el número de usuarios registrados en la plataforma.	Andres Acosta teléfono: 3134301879	OB	F	4
R4	Control de los eventos registrados	El sistema controlara el numero de eventos registrados en la plataforma.	Andres Acosta teléfono: 3134301879	OB	F	4
R5	Registro de eventos cancelados	El sistema guardará el número de eventos cancelados en la plataforma.	Andres Acosta teléfono: 3134301879	OP	F	3
R6	Inicio de sesión	El sistema debe permitir a todos los usuarios ingresar al sistema mediante la verificación de sus datos (usuario y contraseña).	Andres Acosta teléfono: 3134301879	OB	NF	5
R7	Actualización en tiempo real	El sistema debe actualizar las actividades en tiempo real.	Andres Acosta teléfono: 3134301879	DE	F	4
R8	Interfaz intuitiva	La interfaz del sistema debe ser intuitiva para que cualquier usuario	Andres Acosta teléfono: 3134301879	DE	NF	3



		pueda usarla fácilmente.				
R9	Notificación actividad nueva	El sistema notificará a los usuarios de la aplicación los nuevos eventos registrados	Andres Acosta teléfono: 3134301879	OP	NF	3
R10	Notificación actividad modificada	El sistema notificará a los usuarios de la aplicación las modificaciones de los eventos que se registraron.	Andres Acosta teléfono: 3134301879	DE	NF	4
R11	Confirmación de asistencia	El sistema guardará la asistencia de los usuarios que van a ir al evento.	Andres Acosta teléfono: 3134301879	DE	F	4
R12	Registro usuarios	El sistema permitirá el registro de usuarios, solicitándoles información como: Nombre completo, cédula, teléfono, email y contraseña.	Andres Acosta teléfono: 3134301879	OB	F	5
R13	Seguridad en los datos.	El sistema debe proteger la información personal de los usuarios registrados como la cédula y su contraseña.	Andres Acosta teléfono: 3134301879	OB	NF	5

**Tabla 1 Requisitos del sistema.**

## Actores

Del análisis de los requisitos del sistema, surgen los actores descritos a continuación con sus respectivas funciones:

ID	A-01	Nombre	Usuario
Descripción	El usuario es la persona que se beneficia del servicio que presta la aplicación, puede ser la persona que crea actividades de mascotas o la que solo vea los eventos que están registrados en ella.		
ID	A-02	Nombre	Administrador de Usuarios
Descripción	El administrador es la persona que administra la aplicación, teniendo actualizada y verificada la información que se mueve en la aplicación.		
ID	A-03	Nombre	Base de datos
Descripción	La base de datos es la entidad que almacena los datos de los usuarios que se registran en la aplicación, junto a las actividades creadas para mascotas y el registro de los usuarios que van asistir a la aplicación.		

Tabla 2 Actores del sistema

## Casos de Uso

Por el tipo de metodología implementado (Proceso Unificado), se guía el proceso por los casos de uso. Estos describen el sistema desde el punto de vista del usuario, pasando por alto los detalles técnicos. Se enfocan en describir las interacciones del usuario y las actividades o procesos del sistema, y son obtenidos a través de la actividades con las partes involucradas: quien usa el sistema y quien lo administra.

Inicialmente, se mostrará una descripción básica de todos los casos de uso encontrados. Después, usando las matrices de trazabilidad se establecerán los casos de uso base, que son aquellos que aportan en mayor manera a que el sistema cumpla su objetivo, o que son requeridos por otros casos de uso importantes.

## Descripción Básica Casos de Uso

<b>ID</b>	<b>CU-01</b>	<b>NOMBRE</b>	<b>Registro Usuario</b>
<b>Versión</b>	1.0		
<b>Autores</b>	Andrés Mauricio Acosta Pulido		
<b>Requerimientos asociados</b>	R1, R3, R6, R8, R12, R13		
<b>Actores asociados</b>	Usuario, Base de datos		
<b>Descripción</b>	El cliente realiza el registro en la aplicación para poder iniciar sesión en el sistema. El sistema permitirá el registro de usuarios, solicitándoles información como: Nombre completo, cédula, teléfono, email y contraseña.		

<b>ID</b>	<b>CU-02</b>	<b>NOMBRE</b>	<b>Crear Actividad</b>
<b>Versión</b>	1.0		
<b>Autores</b>	Andrés Mauricio Acosta Pulido		
<b>Requerimientos asociados</b>	R2, R4, R5, R7, R8		
<b>Actores asociados</b>	Usuario, Administrador		
<b>Descripción</b>	El cliente una vez registrado en el sistema puede crear actividades relacionadas a mascotas.		

<b>ID</b>	<b>CU-03</b>	<b>NOMBRE</b>	<b>Consultar eventos</b>
<b>Versión</b>	1.0		
<b>Autores</b>	Andrés Mauricio Acosta Pulido		
<b>Requerimientos asociados</b>	R7, R8, R10, R11		
<b>Actores asociados</b>	Usuario		
<b>Descripción</b>	El usuario puede consultar los eventos que estén registrados en la plataforma.		

ID	CU-04	NOMBRE	Registro de eventos cancelados
<b>Versión</b>		1.0	
<b>Autores</b>		Andrés Mauricio Acosta	
<b>Requerimientos asociados</b>		R5, R10	
<b>Actores asociados</b>		Administrador	
<b>Descripción</b>		El sistema guardará y mostrará el número de eventos cancelados en la plataforma.	

ID	CU-05	NOMBRE	Inicio de sesión
<b>Versión</b>		1.0	
<b>Autores</b>		Andrés Mauricio Acosta Pulido	
<b>Requerimientos asociados</b>		R1, R6, R13	
<b>Actores asociados</b>		Usuario, base de datos	
<b>Descripción</b>		El sistema debe permitir a todos los usuarios ingresar al sistema mediante la verificación de sus datos (usuario y contraseña)	

ID	CU-06	NOMBRE	Confirmación de asistencia
<b>Versión</b>		1.0	
<b>Autores</b>		Andrés Mauricio Acosta Pulido	
<b>Requerimientos asociados</b>		R7, R11, R13	
<b>Actores asociados</b>		Usuario, base de datos.	
<b>Descripción</b>		El sistema guardará la asistencia de los usuarios que van a ir al evento para que el usuario creador del evento pueda verificar la asistencia a el mismo.	

ID	CU-07	NOMBRE	Control de información
<b>Versión</b>		1.0	
<b>Autores</b>		Andrés Mauricio Acosta	
<b>Requerimientos asociados</b>		R3, R4, R5, R12, R13	
<b>Actores asociados</b>		administrador	
<b>Descripción</b>		El administrador del sistema controlara el numero de usuarios registrados en el junto a la información que se suministra en la aplicación para que no exceda el límite de espacio en disco que cuenta el servidor del sistema.	

ID	CU-08	NOMBRE	Notificación Usuarios
<b>Versión</b>		1.0	
<b>Autores</b>		Andrés Mauricio Acosta	
<b>Requerimientos asociados</b>		R2, R4, R5, R7, R9, R10, R11	
<b>Actores asociados</b>		Usuario	
<b>Descripción</b>		Se notificará a los usuarios cada vez que haya una actividad nueva o se modifique esta actividad, así se tendrá mayor interés y mejorará la falta de conocimiento por parte de los usuarios de los eventos que puedan cambiar en el tiempo.	

### Matrices de trazabilidad

Basados en los requisitos del sistema, se determinan los casos de uso iniciales que responden a las funcionalidades del sistema detectadas anteriormente. En esta primera versión solo se trabaja sobre aquellos casos de uso que se consideren fundamentales a partir de su relación con los requerimientos. Dicha relación se muestra en la tabla 3.

	ID Requisito	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13
ID R	Nombre Requisitos													
R1	Acceso a la plataforma		X	X	X	X	X					X	X	
R2	Actualización de evento				X			X		X	X			X
R3	Control de registro				X	X		X				X	X	
R4	Control de los eventos registrados													
R5	Registro de eventos cancelados					X								
R6	Inicio de sesión		X											X
R7	Actualización en tiempo real									X	X	X	X	X
R8	Interfaz intuitiva	X	X				X					X	X	
R9	Notificación actividad nueva													
R10	Notificación actividad modificada													
R11	Confirmación de asistencia	X	X	X	X			X						X
R12	Registro usuarios	X		X	X		X					X		
R13	Seguridad en los datos.													

Tabla 2 Relación Requisitos vs Requisitos

Podemos observar en esta matriz de trazabilidad que el requisito R1 acceso a la plataforma es el que más depende de todos los demás requisitos, ya que es vital para obtener una primera versión del a desarrollar.

	ID Requisito	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13
ID Caso uso	Nombre Caso uso													
CU1	Registro Usuario	X		X			X		X				X	X
CU2	Crear Actividad		x		x	x		x	x					
CU3	Consultar eventos							x	x		x	x		
CU4	Registro de eventos cancelados					x					x			
CU5	Inicio de sesión	x					x							x
CU6	Confirmación asistencia							x				x		x
CU7	Control de información			x	x	x							x	x
CU8	Notificación Usuarios		x		x	x		x		x	x	x		

Tabla 3 Relación Casos de uso vs Requisitos

En la **tabla 3** podemos ver que los requerimientos no funcionales (R1, R6, R8, R9, R10, R13) al no ser funcionalidades específicas dentro del sistema, están presentes en varios casos de uso, sin embargo no son necesarios para que el mismo cumpla su objetivo, así que se descartan como casos de uso base.

También se puede observar, que hay casos de uso que se relacionan con un buen número de requisitos como el CU1 (con 6 requisitos), el CU2 (con 5 requisitos), el CU3 (con 4 requisitos) y por último el CU8 (con 7 requisitos), Esto indica que aportan en gran medida a la solución del núcleo del problema, y que deben considerarse como casos de uso base.

En la **tabla 4** se muestra la relación de cada uno de los casos de uso con los actores. Esta tabla permite determinar qué casos de uso dependen de cada actor, y por lo tanto qué funcionalidades deben proporcionarle a cada actor en el sistema. Adicionalmente, esta matriz permite la creación del diagrama de casos de uso.

	ID caso de Uso	CU 1	CU 2	CU 3	CU 4	CU 5	CU 6	CU 7	CU 8
ID Actor	Actor								
A-01	Usuario	x	x	x	x	x	x		x
A-02	Administrador		x					x	
A-03	Base de datos	x				x	x		

Tabla 4 Dependencia de los Casos de Uso con los Actores

En la **tabla 5** se presenta la dependencia entre casos de uso. Esta tabla representa una ayuda para determinar el orden cronológico en el que se debe abordar la creación de los casos de uso: primero aquellos que tengan mayor número de casos de uso dependientes.

	ID caso de Uso	CU 1	CU 2	CU 3	CU 4	CU 5	CU 6	CU 7	CU 8
ID C.U	Nombre Caso uso								
CU1	Registro Usuario		x	x		x	x	x	
CU2	Crear Actividad				x		x	x	x
CU3	Consultar eventos		x		x		x		x
CU4	Registro de eventos cancelados			x			x	x	x
CU5	Inicio de sesión		x	x			x		
CU6	Confirmación asistencia				x			x	x
CU7	Control de información		x			x			
CU8	Notificación Usuarios								x

Tabla 5 Dependencia casos de uso vs casos de uso

En este caso, el requerimiento que llama más la atención por la cantidad de casos de uso que lo necesitan es CU1: Registro de Usuario, esto tiene mucho sentido ya que por la existencia de un usuario determina la cantidad de actividades de mascotas a registrar, junto



a su consulta y asistencia de los mismos. Por lo tanto se establece que Registro de Usuario es un caso de uso base. Los demás casos de uso, no tiene gran número de dependencias, por lo que no se consideran fundamentales para una primera fase de implementación.

En resumen, el análisis de las tablas anteriores, permite determinar que los casos de uso base desarrollados en la primera versión entregable del sistema, son los que aportan tanto a la solución del problema (CU2, CU3, CU5), como los que permiten que el sistema alcance su objetivo (CU1). En esta primera versión, los actores que tiene mayores interacciones son el Usuario y la base de datos, pues los casos de uso base dependen de estos actores, como se ve en la **tabla 4**.

De acuerdo con el análisis desarrollado, y conociendo cuales son los casos de uso base, se propone un orden cronológico en el que se pueden abordar los casos de uso para hacer un trabajo ordenado y eficaz, según las necesidades y requerimientos del sistema. En la **tabla 6** se listan los Casos de Uso priorizados, en donde 10 representa los casos de uso base y disminuye progresivamente según la relevancia del caso de uso, empezando con los primeros de la lista, hasta llegar al final de esta.

Puntuación	ID Caso de Uso	Nombre Caso uso
10	CU1	Registro Usuario
10	CU3	Consultar Actividad
10	CU2	Crear Actividad
5	CU4	Registro de eventos cancelados
5	CU6	Confirmación asistencia
5	CU5	Inicio de sesión
2	CU7	Control de información
1	CU8	Notificación Usuarios

### Diagrama de Casos de Uso Base

Una vez estudiado el entorno y actores del problema, se construyen los casos de uso del sistema, que determinan las actividades que el sistema debe tener para así lograr satisfacer las necesidades de los usuarios. en la **Ilustración 1** se muestran los casos de uso base detectados, junto con los usuarios y relaciones que influyen en su activación y comportamiento. Mostrando así el caso de uso más relevante para el actor de este diagrama.

## Diagramas de Casos de Uso

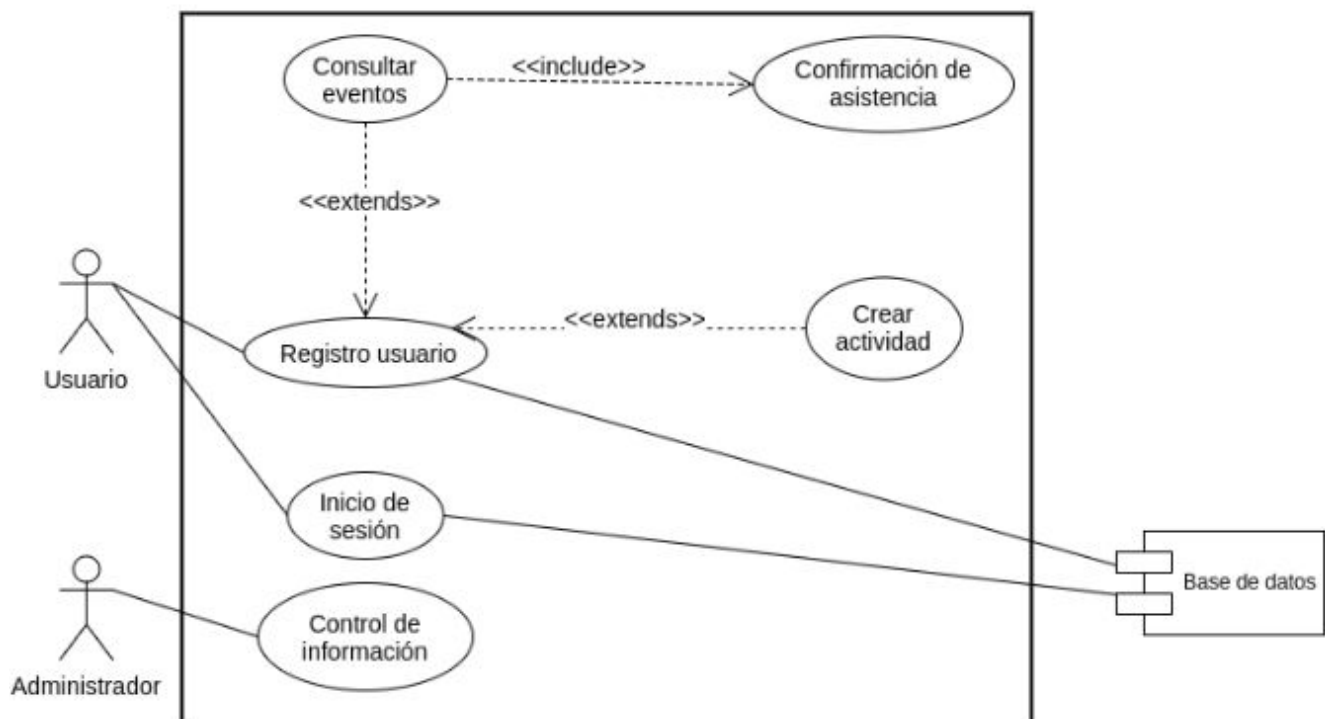


Imagen 1. Primera versión Diagrama de casos de uso

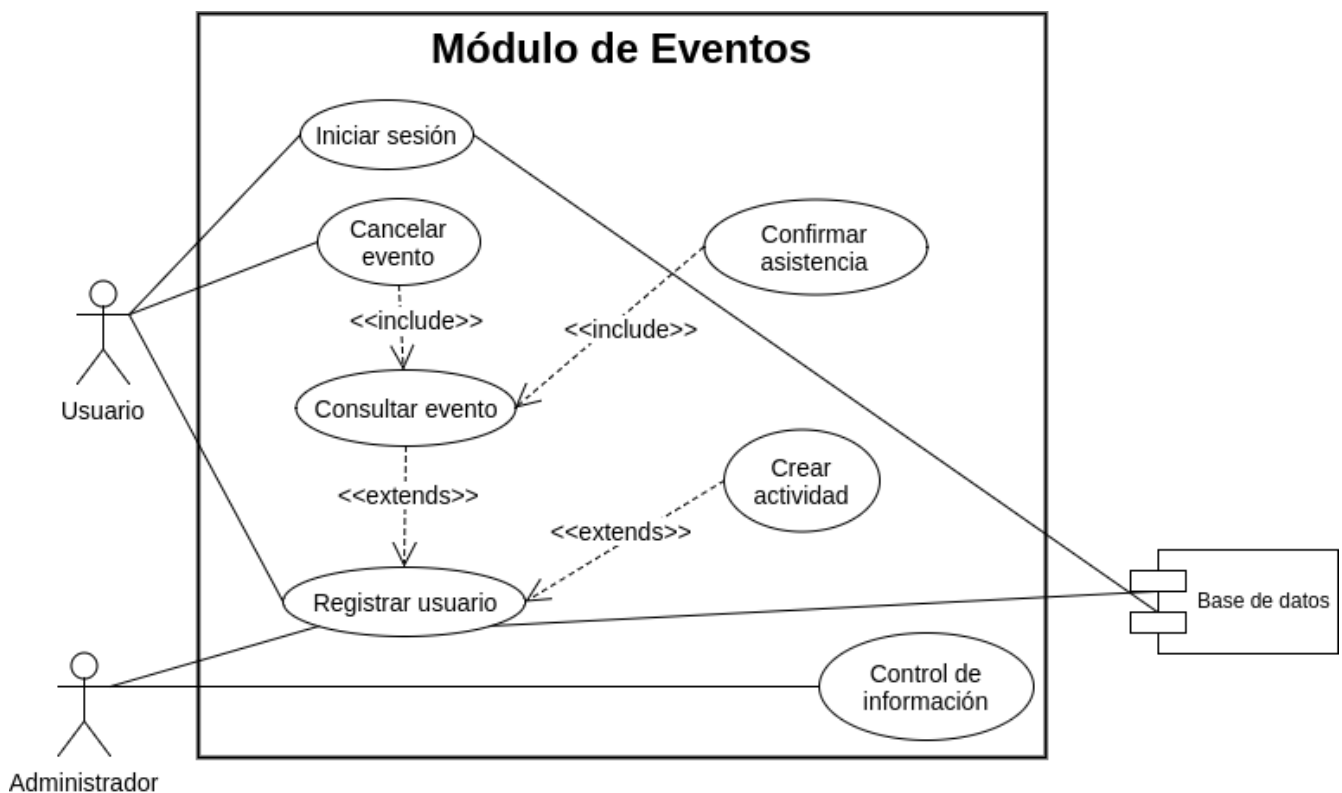


Imagen 1.1. Segunda versión Diagrama de casos de uso

### Descripción Detallada Casos de Uso Base

A continuación se presenta la descripción detallada de cada caso de uso base determinado en la sección anterior.

<b>ID</b>	<b>CU-01</b>	<b>NOMBRE</b>	<b>Registro Usuario</b>
<b>Versión</b>	1.0		
<b>Autores</b>	Andrés Mauricio Acosta Pulido		
<b>Requerimientos asociados</b>	R1, R3, R6, R8, R12, R13		
<b>Actores asociados</b>	Usuario, Base de datos		
<b>Descripción</b>	El cliente realiza el registro en la aplicación para poder iniciar sesión en el sistema. El sistema permitirá el registro de usuarios, solicitándoles información como: Nombre completo, cédula, teléfono, email y contraseña.		
<b>Precondiciones</b>	El usuario debe haber ingresado a la plataforma		
<b>Post condiciones</b>	Existe el Cliente en el sistema.		
<b>Secuencia normal</b>	<ol style="list-style-type: none"> <li>1. El cliente Ingresa a la plataforma</li> <li>2. El sistema pide un usuario y una contraseña.</li> <li>3. El usuario si no está registrado se registra</li> <li>4. Llena los siguientes datos <ol style="list-style-type: none"> <li>a. nombres</li> <li>b. apellidos</li> <li>c. usuario</li> <li>d. contraseña</li> <li>e. cedula</li> <li>f. correo</li> </ol> </li> </ol>		
<b>Flujo alternativo</b>	El sistema no admite más usuarios		
<b>Excepciones</b>	La base de datos no se encuentra disponible		
<b>Frecuencia</b>	depende de la disponibilidad		

<b>ID</b>	<b>CU-02</b>	<b>NOMBRE</b>	<b>Crear Actividad</b>
<b>Versión</b>	1.0		
<b>Autores</b>	Andrés Mauricio Acosta Pulido		
<b>Requerimientos asociados</b>	R2, R4, R5, R7, R8		
<b>Actores asociados</b>	Usuario, Administrador		
<b>Descripción</b>	El cliente una vez registrado en el sistema puede crear actividades relacionadas a mascotas.		
<b>Precondiciones</b>	El usuario debe estar registrado en la base de datos. El usuario debe haber iniciado sesión en la plataforma La actividad ya existe		
<b>Post condiciones</b>			
<b>Secuencia normal</b>	<ol style="list-style-type: none"> <li>1. El cliente da clic en el botón crear actividad.</li> <li>2. registra datos de la actividad</li> <li>3. actividad de perro o gato</li> <li>4. clic sobre el botón finalizar</li> </ol>		
<b>Flujo alternativo</b>	No se pueden crear más actividades en este momento.		
<b>Excepciones</b>	La base de datos no se encuentra disponible		
<b>Frecuencia</b>	Muy frecuente		

<b>ID</b>	<b>CU-03</b>	<b>NOMBRE</b>	<b>Consultar eventos</b>
<b>Versión</b>	1.0		
<b>Autores</b>	Andrés Mauricio Acosta Pulido		
<b>Requerimientos asociados</b>	R7, R8, R10, R11		
<b>Actores asociados</b>	Usuario		
<b>Descripción</b>	El usuario puede consultar los eventos que estén registrados en la plataforma.		
<b>Precondiciones</b>	El usuario debe estar registrado en la base de datos.		

	El usuario debe haber iniciado sesión en la plataforma
<b>Post condiciones</b>	El usuario puede dar clic en asistir al evento No pasa nada.
<b>Secuencia normal</b>	<ol style="list-style-type: none"> <li>1. El cliente da clic en el botón consultar eventos</li> <li>2. El cliente da clic sobre un evento</li> <li>3. El cliente puede dar clic en asistir al evento o sale del evento</li> <li>4. el cliente busca otro evento</li> </ol>
<b>Flujo alternativo</b>	
<b>Excepciones</b>	No se puede encontrar actividad. La actividad expiró.
<b>Frecuencia</b>	Muy frecuente

ID	CU-04	NOMBRE	Registro de eventos cancelados
<b>Versión</b>	1.0		
<b>Autores</b>	Andrés Mauricio Acosta		
<b>Requerimientos asociados</b>	R5, R10		
<b>Actores asociados</b>	Administrador		
<b>Descripción</b>	El sistema guardará y mostrará el número de eventos cancelados en la plataforma.		
<b>Precondiciones</b>	El Administrador iniciara sesión en la plataforma		
<b>Post condiciones</b>			
<b>Secuencia normal</b>	<ol style="list-style-type: none"> <li>1. El Administrador inicia sesión</li> <li>2. clic sobre el botón eventos cancelados</li> <li>3. descargar eventos</li> <li>4. Puede borrar o no los eventos</li> <li>5. salir</li> </ol>		
<b>Flujo alternativo</b>	No hay ningún evento cancelado		
<b>Excepciones</b>	Ninguna		
<b>Frecuencia</b>	Media		

<b>ID</b>	<b>CU-05</b>	<b>NOMBRE</b>	<b>Inicio de sesión</b>
<b>Versión</b>	1.0		
<b>Autores</b>	Andrés Mauricio Acosta Pulido		
<b>Requerimientos asociados</b>	R1, R6, R13		
<b>Actores asociados</b>	Usuario, base de datos		
<b>Descripción</b>	El sistema debe permitir a todos los usuarios ingresar al sistema mediante la verificación de sus datos (usuario y contraseña)		
<b>Precondiciones</b>	Estar registrado en el sistema		
<b>Post condiciones</b>	ventana de bienvenida		
<b>Secuencia normal</b>	1. ingrese nombre de usuario 2. ingrese contraseña 3. iniciar sesión		
<b>Flujo alternativo</b>	los datos no corresponden su cuenta está bloqueada recuperar contraseña		
<b>Excepciones</b>	Usuario no está registrado		
<b>Frecuencia</b>	Muy frecuente		

<b>ID</b>	<b>CU-06</b>	<b>NOMBRE</b>	<b>Confirmación de asistencia</b>
<b>Versión</b>	1.0		
<b>Autores</b>	Andrés Mauricio Acosta Pulido		
<b>Requerimientos asociados</b>	R7, R11, R13		
<b>Actores asociados</b>	Usuario, base de datos.		
<b>Descripción</b>	El sistema guardará la asistencia de los usuarios que van a ir al evento para que el usuario creador del evento pueda verificar la asistencia a el mismo.		
<b>Precondiciones</b>	Tiene que estar dentro de un evento, es decir consultando un evento		

<b>Post condiciones</b>	Registro de evento satisfactorio Notificación día del evento
<b>Secuencia normal</b>	1. clic en el botón asistir al evento o actividad
<b>Flujo alternativo</b>	la actividad tiene un número máximo de asistentes.
<b>Excepciones</b>	No se puede registrar en el evento, por favor intente más tarde
<b>Frecuencia</b>	Muy frecuente

<b>ID</b>	<b>CU-07</b>	<b>NOMBRE</b>	<b>Control de información</b>
<b>Versión</b>	1.0		
<b>Autores</b>	Andrés Mauricio Acosta		
<b>Requerimientos asociados</b>	R3, R4, R5, R12, R13		
<b>Actores asociados</b>	administrador		
<b>Descripción</b>	El administrador del sistema controlara el numero de usuarios registrados en el junto a la información que se suministra en la aplicación para que no exceda el límite de espacio en disco que cuenta el servidor del sistema.		
<b>Precondiciones</b>			
<b>Post condiciones</b>			
<b>Secuencia normal</b>	1. Ingresa administración del servidor		
<b>Flujo alternativo</b>			
<b>Excepciones</b>			
<b>Frecuencia</b>	Media		

ID	CU-08	NOMBRE	Notificación Usuarios
<b>Versión</b>	1.0		
<b>Autores</b>	Andrés Mauricio Acosta		
<b>Requerimientos asociados</b>	R2, R4, R5, R7, R9, R10, R11		
<b>Actores asociados</b>	Usuario		
<b>Descripción</b>	Se notificará a los usuarios cada vez que haya una actividad nueva o se modifique esta actividad, así se tendrá mayor interés y mejorará la falta de conocimiento por parte de los usuarios de los eventos que puedan cambiar en el tiempo.		
<b>Precondiciones</b>	Actividad creada, actividad modificada.		
<b>Post condiciones</b>			
<b>Secuencia normal</b>	1. Se envía un correo al cliente por el cambio de alguna actividad o creación.		
<b>Flujo alternativo</b>			
<b>Excepciones</b>	No pudo enviar la notificación		
<b>Frecuencia</b>	media		

## Diagramas

Los diagramas presentados a continuación son un resultado del análisis y desarrollo del proyecto. Cada tipo de diagrama modela el sistema desde una perspectiva diferente. Los diagramas de secuencia y estados hacen énfasis en el tratamiento de los casos de uso base, mientras el diagrama de clases, por ejemplo, abarca un poco más de profundidad, incluyendo otros casos de uso.

### Diagramas de clases

A continuación, se presenta una representación de las entidades del sistema, mediante la clasificación del modelo-vista-control, utilizando la implementación de DAO (Data Access Object) para facilitar la comunicación con la base de datos no relacional, implementando a su vez principios del diseño del software, tal como el principio de sustitución de Liskov.



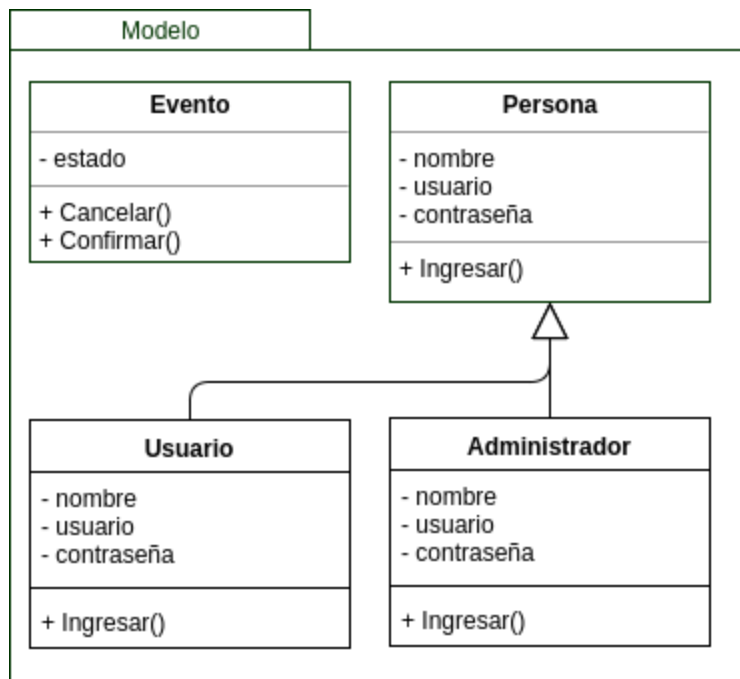


Imagen 2. Modelo en el Diagrama de clases

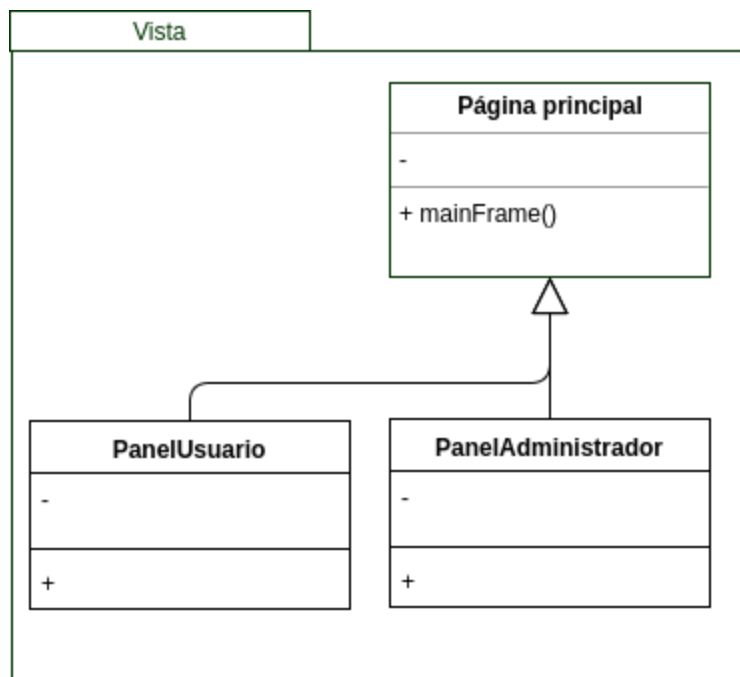


Imagen 3. Vista en el Diagrama de clases

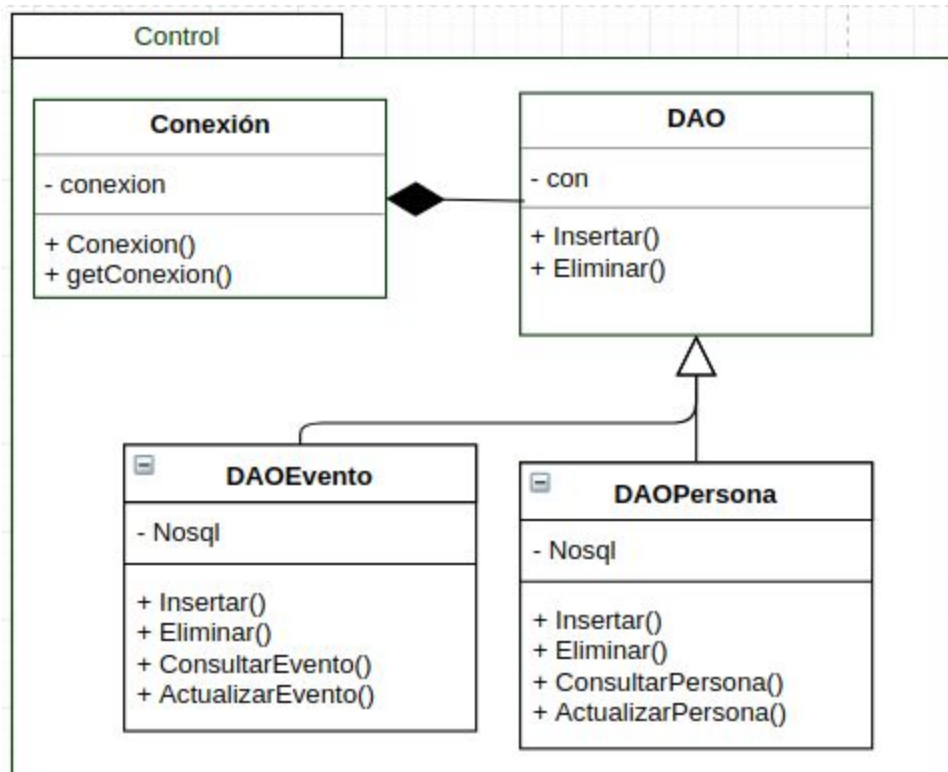


Imagen 4. Control en el Diagrama de clases

## Diagramas de estado

De acuerdo a la descripción detallada de los casos de uso, se determina que hay tres elementos del sistema que cambian durante tiempo de estado, estos son el usuario, el evento y el administrador (analizando el caso en el cual hay dos o más administradores con posibilidad de inactividad)

### Diagrama de estados Evento



### Diagrama de estados Administrador

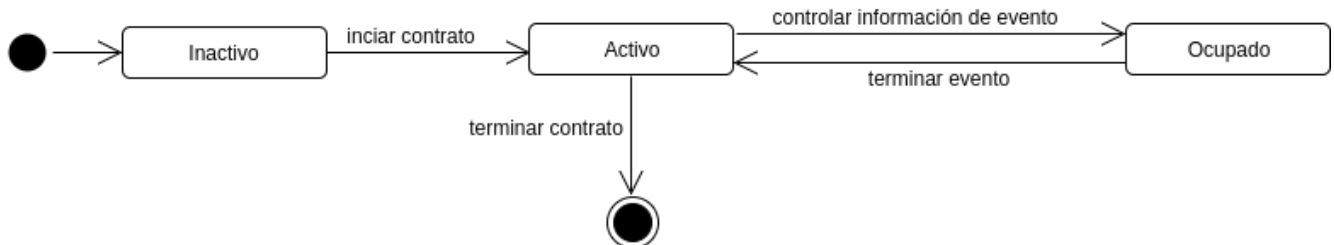


Diagrama de estados Usuario

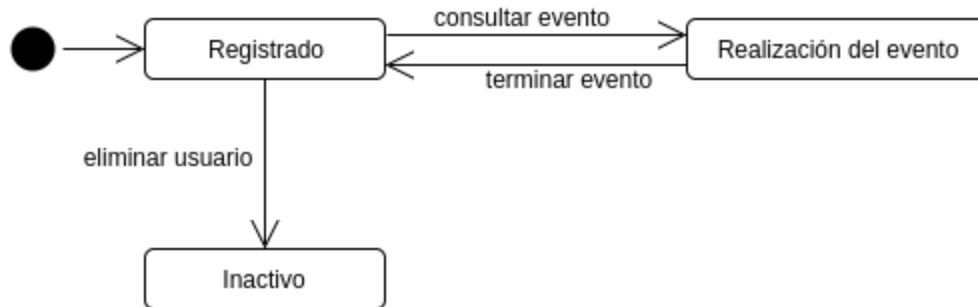


Imagen 5. Diagrama de estados Evento, Administrador y Usuario

### Diagramas de secuencia

A continuación se modela la interacción entre objetos en el sistema según el Lenguaje unificado de modelado (UML)

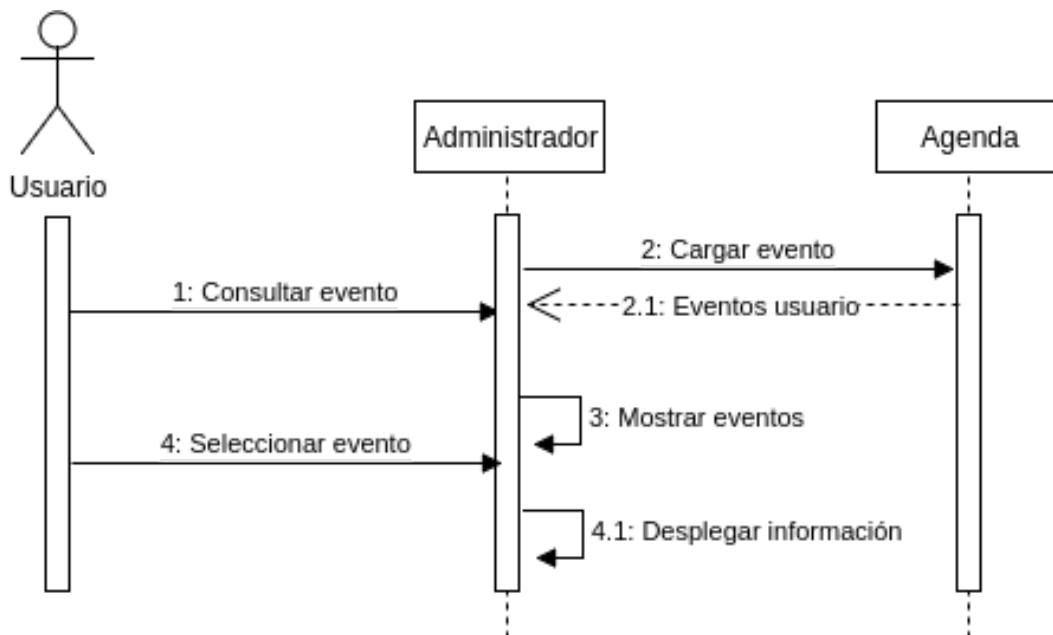
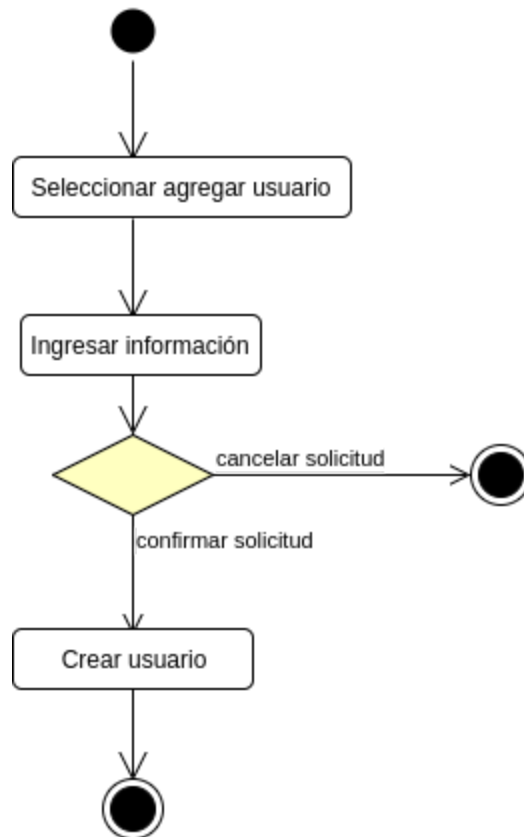


Imagen 6. Diagrama de secuencia consultar evento

## Diagramas de actividades

En esta sección se presentarán los casos de uso estudiados como proceso a través de una serie de actividades:

### *Diagrama de actividades agregar usuario*



**Diagrama de actividades consultar agenda**

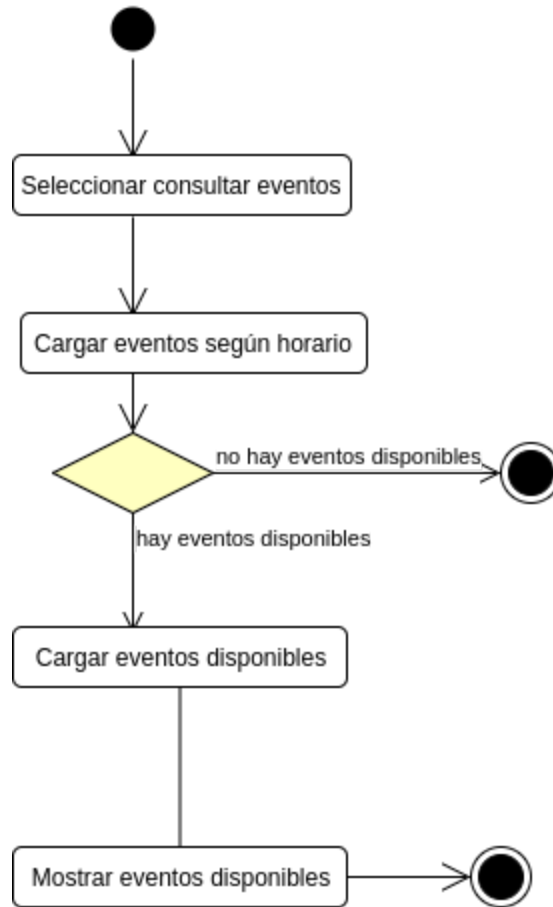


Imagen 7. Diagramas de actividades

## Diagramas de componentes

A continuación se presenta la representación del software dividido en componentes, mostrando las dependencias entre ellos

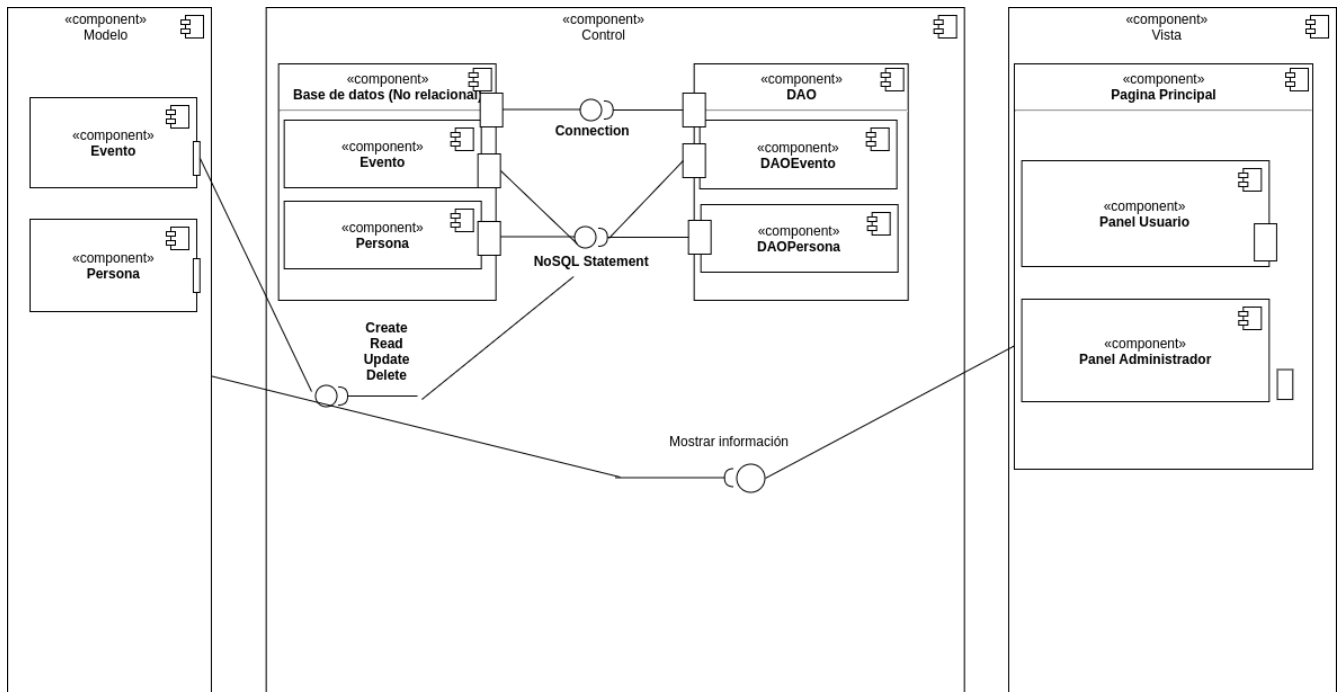


Imagen 8. Segunda versión del Diagrama de componentes

## Agenda de trabajo

A continuación, se presenta una proyección de agenda de trabajo por semanas

Semana	Etapa	Actividades
0 a 5	Análisis	Recolección, análisis de requerimientos Creación casos de uso Diagramas de casos de uso
5 a 10	Diseño	Definición tipo de arquitectura para el sistema
10 a 14	Implementación	Desarrollo del aplicativo con sus respectivas pruebas para hacer entrega al cliente

**tabla 12. Agenda**

## **Tecnología a utilizar.**

Luego de haber realizado un estudio detallado de la aplicación, entendiendo primeramente el contexto y la debida recolección de requisitos, junto al diseño de una primera parte de los diagramas para ayudarnos a ver más claramente el aplicativo a desarrollar pasamos al desarrollo del aplicativo, escogiendo como primera instancia las tecnologías a usar las cuales son:

### **MongoDB**

MongoDB es un sistema de base de datos documental no relacional de código abierto. Este sistema de base de datos, guarda los datos en documentos en formato JSON (en realidad los guarda en formato BSON), lo que hace que sea un formato muy manejable y entendible para trabajar con él, y las operaciones (o "consultas") sobre estos documentos se realizan en JavaScript. De MongoDB cabe destacar su flexibilidad y escalabilidad, lo que ayuda al desarrollo ágil de proyectos software. Son muchos los lenguajes que pueden trabajar con MongoDB (Java, PHP, Python, etc.) y por supuesto NodeJs es uno de ellos.

### **Express**

Express es un framework realizado en NodeJs muy ligero y flexible que proporciona un conjunto muy robusto de facilidades para crear fácilmente servidores web y recibir peticiones HTTP; por tanto, permite desarrollar API REST de forma muy rápida.

### **AngularJS**

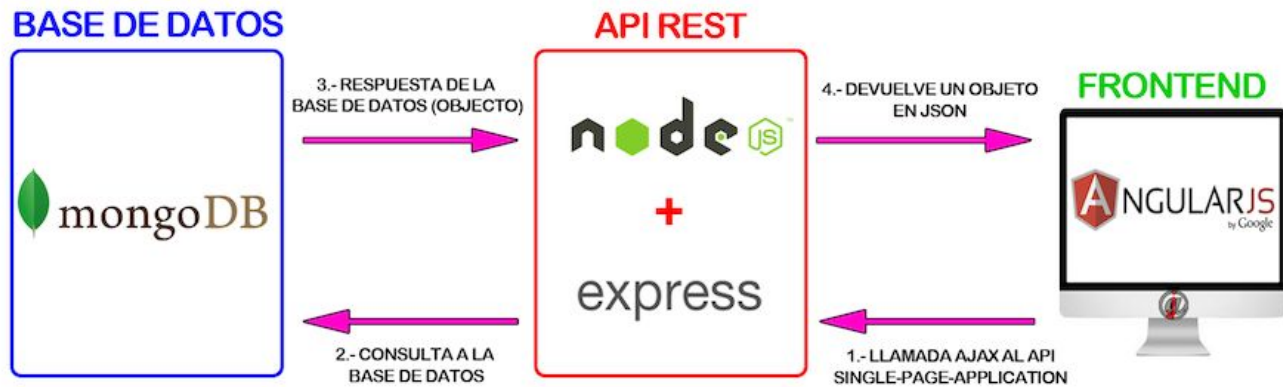
AngularJS es un framework para Frontend desarrollado por Google, que sigue el patrón MVC (Model-Vista-Controlador). AngularJS permite crear lo que se llama Single-Page-Applications, que son aplicaciones web que no necesitan recargar la página. Siendo AngularJS un framework muy potente y que facilita muchísimo la labor de los desarrolladores de Frontend.

### **NodeJS**

NodeJS (creado en el año 2009) es un entorno de programación (en Javascript) en la capa del servidor basado en el motor V8 de JavaScript, orientado a eventos no bloqueantes, lo que lo hace muy rápido y permite crear programas de red altamente escalables (servidores web) y emplear el "tiempo real".

## MEAN

Realizando en última instancia una aplicación tipo MEAN la cual recopila con el uso de estas 4 tecnologías, todo lo necesario (Frontend, Backend y Base de Datos) para desarrollar una aplicación, Explicando así el uso y la funcionalidad de cada tecnología en la siguiente imagen.



1

Imagen 9. Arquitectura del sistema.

Observando en la imagen que el Frontend lo desarrollamos con AngularJS, que es el encargado de hacer llamadas al API REST (Post, Put, Get y Delete), siguiendo del desarrollo con NodeJS que utiliza el framework de Express.

El API podrá hacer un CRUD (Create-Read-Update-Delete) a la base de datos MongoDB y cuando el API tenga los datos que se le han pedido en la llamada los devolverá a AngularJS (es decir al Frontend) en formato JSON y este los mostrará en pantalla sin necesidad de recargar la página ya que AngularJS mantiene el modelo de datos actualizado sin necesidad de recargar la página.

Como vemos esta es la arquitectura y el flujo de datos que utilizaremos para nuestro desarrollo de *"Sistema para el Control de actividades orientadas a mascotas."*

## PROTOTIPO SLIDE

Los siguientes prototipos de pantallas son un primer acercamiento al objetivo de la aplicación, la cual cuenta con dos módulos fundamentales, los cuales son el inicio de sesión y el prototipo de pantalla de inicio al ingresar a la plataforma, junto con el módulo del administrador el cual controla los datos del sistema y puede ver quienes están registrados en el.

<sup>1</sup> Todado de: [https://jarroba.com/wp-content/uploads/2014/07/MEAN\\_arquitectura\\_jarroba.png](https://jarroba.com/wp-content/uploads/2014/07/MEAN_arquitectura_jarroba.png)



INICIAR SESIÓN

Usuario

Contraseña

INICIAR

[Olvido su contraseña?](#)

Imagen 10. inicio de sesión.

Bienvenido al Control de animales Inicio Consultar Evento Crear Evento

---

### Evento Registrado

Nombre Evento	fecha	Correo Creador	Telefono
---------------	-------	----------------	----------

Imagen 11. Dentro de la plataforma.

Nº de Registros: 2

Nombre	Apellidos	correo	contraseña
Andres	Acosta	correo@correo.com	12345
Andres	Acosta	andres@andres.com	nomelase

Registrar Modificar Borrar

Imagen 12. Panel de control Administrador.

## **COSTOS.**

Luego de haber realizado un primer prototipo de nuestra aplicación, y saber a partir de la **tabla 12** cuánto tiempo gastaremos en el proyecto de “*Sistema para el control de actividades orientadas a mascotas.*” podemos estimar un costo de \$ 5.000.000 millones de pesos.

## **Conclusiones**

De la fase de inicio o el análisis del sistema permite una descripción inicial del sistema que determina los aspectos más relevantes y la funcionalidad general del sistema. En esta etapa se obtuvieron 13 requerimientos registrados en la tabla 1, tres actores registrados en la tabla 2, y 8 casos de uso descritos con brevedad, que representan la funcionalidad del sistema y su objetivo. Además luego de un riguroso análisis se establece una priorización de los casos de uso (tabla 6). Ente la fase de inicio de diseño, se obtiene un conocimiento amplio y consensuado de la lógica del negocio, es decir que ya está planteada la base comportamental del sistema, que incluye casos de uso no contemplados inicialmente. Al fijar una agenda (tabla 12) se determina que la primera versión debe abarcar las funciones descritas en los casos de uso base.

La primera versión del sistema comprende un conjunto de interfaces que cumplen con los requerimientos de agilidad y facilidad de navegación establecidos, en la cual el sistema solo cuenta con las funciones establecidas en los casos de uso. El desarrollo de este sistema ayudará a la población que tiene mascotas, y la aplicación de conocimientos de las bases de la ingeniería de software contribuye a la optimización de la calidad del sistema.

## **Bibliografía**

- [1]. PRESSMAN, Roger S. “Ingeniería del Software. Un Enfoque Práctico”. Séptima Edición. Mc. Graw Hill. 2010
- [2]. SOMMERVILLE, Ian. “Software Engineering”. Séptima Edición. Addison Wesley. 2004

## **Recursos Electrónicos.**

- [1]. <https://www.draw.io/>
- [2]. Software StarUML
- [3]. Arquitectura Tipo MEAN (Mongo, Express, AngularJS, NodeJS):  
[https://jarroba.com/wp-content/uploads/2014/07/MEAN\\_arquitectura\\_jarroba.png](https://jarroba.com/wp-content/uploads/2014/07/MEAN_arquitectura_jarroba.png)