Índice

Requerimientos funcionales 2

Requerimientos no funcionales 3

Especificaciones de Seguridad 4

Diagramas de caso de uso 5

Especificación casos de uso 8

Diagrama de clases 33

DER 33

Diagrama de arquitectura 33

Diagrama de despliegue 34

DETALLE DE INTERFAZ ENTRE SISTEMAS 36

# Requerimientos funcionales

WEB pública

* El sistema mostrará listado de las mascotas que fueron adoptadas.
* El sistema mostrará 3 mascotas en adopción de forma aleatoria.
* El sistema mostrará un listado de eventos vigentes .

WEB privada

* El sistema pedirá email y password para poder loguearse como administrador
* El sistema tendrá una pantalla para gestionar usuarios donde se puede dar de alta ,listar, buscar y modificar y borrar usuarios.
* El sistema tendrá una pantalla para gestionar los servicios donde se podrán dar de alta ,listar buscar editar y eliminar servicios.
* El sistema tendrá una pantalla para gestionar los eventos donde se podrá dar de alta,listar,buscar,editar y eliminar eventos, los eventos tendrán un periodo de vigencia, solo se mostraran al usuario los eventos vigentes
* El sistema tendrá una opción para desloguearse del sistema no permitiendo volver a entrar a las paginas de usuario, eventos, servicios y estadísticas sin volver a loquearse nuevamente
* El sistema tendrá una pantalla para mostrar gráficos estadísticos de las mascotas por tipo edad y sexo, adoptada y disponibles

Mobile

* El sistema permitirá al usuario crear su cuenta con su nombre, apellido, e-mail y contraseña con rol USUARIO.  
  El sistema permitirá ingresar usuarios indistinto de su rol.
* El sistema permitirá loguearse con facebook y en caso de no contar con una cuenta ya hecha creara una con rol de usuario.
* El sistema mostrará un listado de servicios vigentes.
* El sistema permitirá la contratación de los mismos a través de un link de Mercado Pago.
* El sistema mostrará un listado de eventos vigentes.  
  El sistema mostrará un listado de mis mascotas cargadas.
* El sistema permitirá al usuario crear una mascota para dar en adopción.
* El sistema permitirá al usuario dar de baja la mascota una vez que ya esté dada en adopción.
* El sistema mostrará un listado de mascotas disponibles en adopción con un botón de contacto.
* El sistema permitirá al usuario filtrar la búsqueda por sexo, tamaño, edad y distancia.
* El sistema permitirá al usuario a través del botón de contacto mandar un mail al rescatista de la mascota elegida para establecer un contacto.
* El sistema tendrá una opción para desloguearse del sistema y luego lo redirigirá a al Login.

# Requerimientos no funcionales

* Los administradores del sistema web deben poder aprender a gestionar usuarios, eventos y servicios de manera sencilla e intuitiva.
* La aplicación web debe ser compatible con los siguientes exploradores: Internet Explorer, Google Chrome, Safari.
* El sistema Mobile debe contar con textos que tengan un lenguaje amigable al usuario evitando el uso de términos técnicos.
* Las contraseñas se almacenaran en base de datos de forma encriptada.
* Únicamente los usuarios con rol de administrador podrán gestionar usuarios de la aplicación.
* El tiempo para iniciar o reiniciar el sistema no debe ser extenso.
* El proceso de desarrollo debe permitir subir cambios al servidor web y deployarlos en producción de forma sencilla para minimizar tiempo de espera de resolución de fallos y de indisponibilidad del servicio.
* Los sistemas tanto Mobile como Web deben contar con interfaces de usuarios definidas de forma homogénea.
* La aplicación permitirá el acceso de múltiples usuarios al mismo tiempo.

# Especificaciones de Seguridad

* Las contraseñas deben ser de 8 caracteres incluyendo una mayúscula , una minúscula y un número.
* Las contraseñas se almacenan en base de datos de forma encriptada usando la función de hashing BCRYPT provista por Spring Security.
* Los accesos a la aplicación son a través de roles ADMIN y USER .
* El usuario de bbdd tiene solo los permisos necesarios para poder operar la aplicación.
* Se Inyecta la URL de conexión de la base de datos mediante una variable de entorno , de esta manera evitamos guardar esa información dentro del código fuente y queda configurada en el servidor.
* Se Inyecta el APIKEY de Sendgrid mediante una variable de entorno de esta manera evitamos guardar esa información dentro del código fuente y queda configurada en el servidor.
* Las querys son manejadas con JPA lo que evita SQL Injection.
* No se muestran errores técnicos al usuario para evitar revelar información de la aplicación.
* Se utiliza mecanismo de oauth para loguearse con Facebook a la aplicación Mobile, esto es manejado automáticamente por la dependencia de Facebook.
* Se utiliza una APIKEY provista por google para utilizar las apis de google maps, esta apikey identifica nuestra aplicación ante google.
* Se utiliza APIKEY para el servicio de envió de mails provisto por SENDGRID
* Se utiliza APIKEY para el servicio de almacenamiento de imágenes provisto por cloudinary.

# Diagramas de caso de uso

Diagramas casos de uso Web Pública

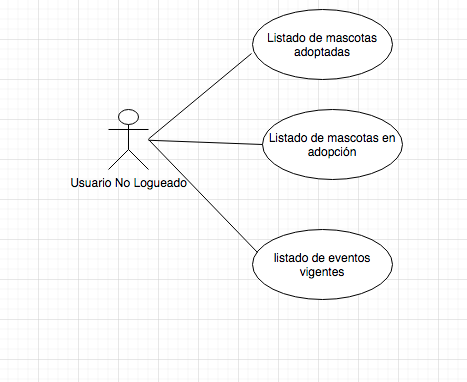
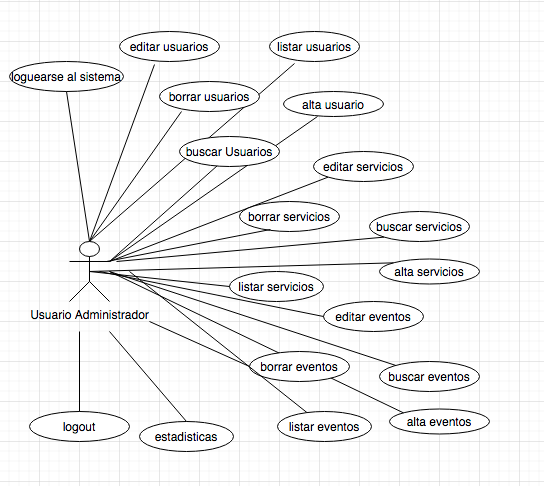
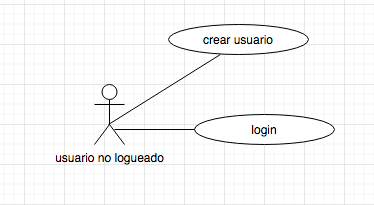
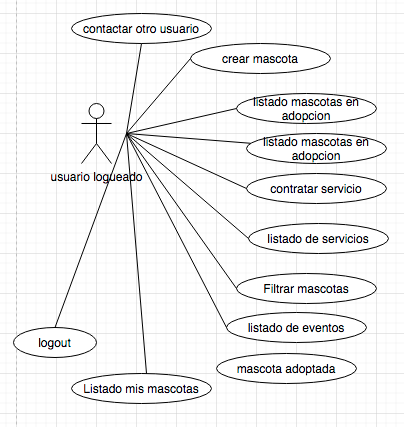


Diagrama casos de uso Web privada



Diagramas caso de uso Mobile





# Especificación casos de uso

*WEB pública*

|  |  |
| --- | --- |
| Casos de uso 1 | Listado de mascotas adoptadas |
| Actores | Usuario no logueado a la web |
| Descripción | El usuario puede ver un listado con las mascotas que han sido adoptadas |
| Precondición | Que haya mascotas adoptadas |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario ingresa a la página web publica 2. El Sistema muestra listado de mascotas adoptadas 3. El usuario puede visualizar foto y nombre de las mascotas adoptadas |  |
| Observaciones |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 2 | Listado de mascotas en adopción |
| Actores | Usuario no logueado a la web |
| Descripción | El usuario puede ver un listado con 3 mascotas disponibles para adopción  seleccionadas al azar |
| Precondición | Que haya mascotas disponibles para adopción |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario ingresa a la página web publica 2. El Sistema muestra listado de 3 mascotas para adopción 3. El usuario puede visualizar foto y nombre edad descripción,sexo y tamaño de las mascotas |  |
| Observaciones si se vuelve a recargar la pagina y hay mas de 3 mascotas disponibles para adopción se iran mostrando mascotas al azar pueden repetirse las mismas como no |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 3 | Listado de eventos vigentes |
| Actores | Usuario no logueado a la web |
| Descripción | El usuario puede ver un listado con los eventos vigentes |
| Precondición | Que haya eventos vigentes |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario ingresa a la página web publica 2. El Sistema muestra listado de los eventos vigentes que son aquellos que tengan fecha a partir del dia actual hasta un mes despues 3. El usuario puede visualizar el lugar,dirección,barrio,dia,hora y descripción del evento |  |
|  |  |
| PostCondición |  |

*Web privada*

|  |  |
| --- | --- |
| Casos de uso 4 | Loguearse al sistema |
| Actores | Usuario administrador |
| Descripción | El usuario se podrá loguear al sistema |
| Precondición | Contar con un usuario administrador |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario ingresa a la página web publica 2. El usuario hace click en el link admin 3. El usuario usuario ingresa a la pagina de login 4. El usuario ingresa email y password y hace click en el botón entrar 5. El usuario ingresa a la web privada situándose en la pagina de gestionar usuarios | 4 A El usuario ingresa Email y password erróneos o no existentes o ingresa con un usuario NO administrador  4 B El sistema muestra el mensaje “Verifique los datos ingresados” |
|  |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 5 | Alta usuario |
| Actores | Usuario administrador |
| Descripción | El usuario podrá dar de alta usuario |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de usuarios 2. El usuario completa los datos del usuario: nombre,apellido,email,password y rol USER o ADMIN 3. El usuario usuario hace click en crear 4. El sistema dara de alta un usuario nuevo 5. El usuario dado de alta se visualizara en el listado de usuarios | 4 1 El usuario no ingresa algún dato obligatorio o email con formato no valido  4 1.1 El sistema muestra el mensaje de error debajo del/los campo/s a corregir  4.2 El usuario ingresa un mail ya existente  4.2.1 El usuario hace click en crear  4.2.2 El sistema muestra un popup con mensaje de error indicando que ese mail ya fue registrado |
|  |  |
| PostCondición | El usuario quedara dado de alta en el sistema |

|  |  |
| --- | --- |
| Casos de uso 6 | Listar Usuarios |
| Actores | Usuario administrador |
| Descripción | El usuario podrá listar los usuario existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con usuarios dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de usuarios. 2. El usuario visualizará un listado de los usuarios existentes en la aplicación mostrando id,nombre,apellido ,mail rol y las acciones a realizar sobre ese usuario(borrar y editar). 3. Si hay mas de 10 usuarios el listado mostrara un paginado del mismo |  |
|  |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 7 | Buscar Usuarios |
| Actores | Usuario administrador |
| Descripción | El usuario podrá buscar entre los usuarios existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con usuarios dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de usuarios. 2. El usuario visualizará un listado de los usuarios existentes en la aplicación mostrando id,nombre,apellido ,mail rol y las acciones a realizar sobre ese usuario(borrar y editar). 3. El Usuario podrá ingresar en el campo buscar cualquier dato del usuario que se muestre en el listado, solo buscara por un campo a la vez 4. El usuario podrá ver el/los usuario/s que cumplan el criterio de búsqueda | * 1. El usuario ingresa un criterio de búsqueda inexistente   4.1 El usuario visualizará el listado vacío |
|  |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 8 | Borrar Usuarios |
| Actores | Usuario administrador |
| Descripción | El usuario podrá borrar usuarios existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con usuarios dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de usuarios. 2. El usuario visualizará un listado de los usuarios existentes en la aplicación mostrando id,nombre,apellido ,mail rol y las acciones a realizar sobre ese usuario(borrar y editar). 3. El Usuario hace click en el icono del tacho de basura ubicado en el listado de usuarios para borrar el usuario seleccionado 4. El sistema mostrara un mensaje indicando la acción exitosa |  |
|  |  |
| PostCondición | * El usuario borrado no aparecerá en el listado de usuarios |

|  |  |
| --- | --- |
| Casos de uso 9 | Editar Usuarios |
| Actores | Usuario administrador |
| Descripción | El usuario podrá editar usuarios existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con usuarios dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de usuarios. 2. El usuario visualizará un listado de los usuarios existentes en la aplicación mostrando id,nombre,apellido ,mail rol y las acciones a realizar sobre ese usuario(borrar y editar). 3. El Usuario hace click en el icono del lápiz ubicado en el listado de usuarios para editar el usuario seleccionado 4. El sistema abrirá un popup con los datos a modificar del usuario: nombre ,apellido, email y rol 5. El usuario modifica los datos y hace click en el botón grabar 6. El sistema muestra un mensaje indicando la operación exitosa | * 1. El usuario pone un email invalido o deja un campo obligatorio vacío   2. El sistema muestra error de validación |
|  |  |
| PostCondición | * El usuario aparecerá en el listado de usuarios con sus datos modificados |

|  |  |
| --- | --- |
| Casos de uso 10 | Alta Servicio |
| Actores | Usuario administrador |
| Descripción | El usuario podrá dar de alta servicios |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de servicios 2. El usuario completa los datos del usuario que presta el servicio 3. El usuario completa los datos del servicio 4. El sistema dara de alta un servicio nuevo 5. El servicio dado de alta se visualizara en el listado de servicios | 4 1 El usuario no ingresa algún dato obligatorio o email con formato no valido  4 1.1 El sistema muestra el mensaje de error debajo del/los campo/s a corregir  4.2 El usuario ingresa un mail de un prestador de servicio ya existente, al salir del campo email , se completaran automáticamente los campos del usuario que se hayan cargado anteriormente |
|  |  |
| PostCondición | El servicio quedara dado de alta en el sistema |

|  |  |
| --- | --- |
| Casos de uso 11 | Listar Servicios |
| Actores | Usuario administrador |
| Descripción | El usuario podrá listar los servicios existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con servicios dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de servicios. 2. El usuario visualizará un listado de los servicios existentes en la aplicación mostrando prestador ,email,tipo de servicio,titulo,descripción,costo,url de pago,fecha hasta y las acciones a realizar sobre ese servicio(borrar y editar). 3. Si hay mas de 10 servicios el listado mostrara un paginado del mismo |  |
|  |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 12 | Buscar Servicios |
| Actores | Usuario administrador |
| Descripción | El usuario podrá buscar entre los servicios existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con servicios dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de servicios 2. El usuario visualizará un listado de los servicios existentes en la aplicación mostrando prestador ,email, tipo de servicio,titulo,descripción,costo,url de pago, fecha hasta y las acciones a realizar sobre ese servicio(borrar y editar). 3. El Usuario podrá ingresar en el campo buscar cualquier dato del servicio que se muestre en el listado, solo buscara por un campo a la vez 4. El usuario podrá ver el/los servicios/s que cumplan el criterio de búsqueda | * 1. El usuario ingresa un criterio de búsqueda inexistente   4.1 El usuario visualizará el listado vacío |
|  |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 13 | Borrar Servicios |
| Actores | Usuario administrador |
| Descripción | El usuario podrá borrar servicios existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con servicios dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de servicios. 2. El usuario visualizará un listado de los servicios existentes en la aplicación mostrando prestador ,email, tipo de servicio,titulo,descripción,costo,url de pago, fecha hasta y las acciones a realizar sobre ese servicio(borrar y editar). 3. El Usuario hace click en el icono del tacho de basura ubicado en el listado de servicios para borrar el servicio seleccionado 4. El sistema mostrara un mensaje indicando la acción exitosa |  |
|  |  |
| PostCondición | * El servicio borrado no aparecerá en el listado de usuarios |

|  |  |
| --- | --- |
| Casos de uso 14 | Editar Servicios |
| Actores | Usuario administrador |
| Descripción | El usuario podrá editar servicios existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con servicios dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de servicios. 2. El usuario visualizará un listado de los servicios existentes en la aplicación mostrando prestador ,email, tipo de servicio,título,descripción,costo,url de pago, fecha hasta y las acciones a realizar sobre ese servicio(borrar y editar). 3. El Usuario hace click en el icono del lápiz ubicado en el listado de servicio para editar el servicio seleccionado 4. El sistema abrirá un popup con los datos a modificar del servicio: titulo ,costo, descripción ,tipo de servicio, fecha fin, url de pago 5. El usuario modifica los datos y hace click en el botón grabar 6. El sistema muestra un mensaje indicando la operación exitosa | * 1. El usuario pone un dato invalido o deja un campo obligatorio vacío   2. El sistema muestra error de validación |
|  |  |
| PostCondición | * El servicios aparecerá en el listado de servicios con sus datos modificados |

|  |  |
| --- | --- |
| Casos de uso 15 | Alta Evento |
| Actores | Usuario administrador |
| Descripción | El usuario podrá dar de alta eventos |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de eventos 2. El usuario completa los datos del evento 3. El sistema dará de alta un evento nuevo 4. El evento dado de alta se visualizara en el listado de eventos | 4 1 El usuario no ingresa algún dato obligatorio  4 1.1 El sistema muestra el mensaje de error debajo del/los campo/s a corregir |
|  |  |
| PostCondición | El evento quedara dado de alta en el sistema |

|  |  |
| --- | --- |
| Casos de uso 16 | Listar Eventos |
| Actores | Usuario administrador |
| Descripción | El usuario podrá listar los eventos existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con eventos dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de eventos. 2. El usuario visualizará un listado de los eventos existentes en la aplicación mostrando id evento,dia,horario,lugar,barrio y consultas y las acciones a realizar sobre ese evento(borrar y editar). 3. Si hay mas de 10 eventos el listado mostrara un paginado del mismo |  |
|  |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 17 | Buscar Eventos |
| Actores | Usuario administrador |
| Descripción | El usuario podrá buscar entre los eventos existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con eventos dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de eventos 2. El usuario visualizará un listado de los eventos existentes en la aplicación mostrando id evento,dia,horario,lugar,barrio y consultas y las acciones a realizar sobre ese evento(borrar y editar). 3. El Usuario podrá ingresar en el campo buscar cualquier dato del evento que se muestre en el listado, solo buscara por un campo a la vez 4. El usuario podrá ver el/los eventos/s que cumplan el criterio de búsqueda | * 1. El usuario ingresa un criterio de búsqueda inexistente   4.1 El usuario visualizará el listado vacío |
|  |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 18 | Borrar eventos |
| Actores | Usuario administrador |
| Descripción | El usuario podrá borrar eventos existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con eventos dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de eventos. 2. El usuario visualizará un listado de los eventos existentes en la aplicación mostrando id evento,día,horario,lugar,barrio y consultas y las acciones a realizar sobre ese evento(borrar y editar). 3. El Usuario hace click en el icono del tacho de basura ubicado en el listado de eventos para borrar el evento seleccionado 4. El sistema mostrara un mensaje indicando la acción exitosa |  |
|  |  |
| PostCondición | * El evento borrado no aparecerá en el listado de usuarios |

|  |  |
| --- | --- |
| Casos de uso 19 | Editar Eventos |
| Actores | Usuario administrador |
| Descripción | El usuario podrá editar eventos existentes |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, contar con eventos dados de alta en la aplicación |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario ingresa a la pantalla de gestión de eventos. 2. El usuario visualizará un listado de los eventos existentes en la aplicación mostrando id evento,día,horario,lugar,barrio y consultas y las acciones a realizar sobre ese evento(borrar y editar). 3. El Usuario hace click en el icono del lápiz ubicado en el listado de evento para editar el servicio seleccionado 4. El sistema abrirá un popup con los datos a modificar del evento: dia,horario,direccion lugar, barrio y consultas 5. El usuario modifica los datos y hace click en el botón grabar 6. El sistema muestra un mensaje indicando la operación exitosa | * 1. El usuario pone un dato invalido o deja un campo obligatorio vacío   2. El sistema muestra error de validación |
|  |  |
| PostCondición | * El evento aparecerá en el listado de eventos con sus datos modificados |

|  |  |
| --- | --- |
| Casos de uso 20 | logout |
| Actores | Usuario administrador |
| Descripción | El usuario podrá desloguearse de la aplicacion |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario selecciona la opción de logout del menú 2. El sistema desloguea al usuario y se redirige a la pantalla de la web publica |  |
|  |  |
| PostCondición | * El usuario no podrá acceder a ninguna funcionalidad de la web privada sin volver a loguearse nuevamente |

|  |  |
| --- | --- |
| Casos de uso 21 | estadísticas |
| Actores | Usuario administrador |
| Descripción | El usuario podrá ver estadísticas de mascotas |
| Precondición | Contar con un usuario administrador ,estar logueado en la web privada, |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. EL usuario selecciona la opción de estadísticas del menú 2. El sistema muestra gráficos de mascotas por estado (adoptadas-disponibles) , por tipo (perro-gato), por edad, y por sexo (macho –hembra) | * 1. No hay mascotas disponibles en el sistema   2. Los gráficos se mostraran vacíos |
| PostCondición |  |

*Mobile*

|  |  |
| --- | --- |
| Casos de uso 22 | Crear usuario |
| Actores | Usuario sin cuenta creada |
| Descripción | El usuario creará una cuenta |
| Precondición | Que no tenga cuenta |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario abre la aplicación 2. El Sistema muestra la pantalla de Login 3. El usuario presiona en Nueva Cuenta 4. El sistema muestra la pantalla de Nueva Cuenta 5. El usuario ingresa nombre, apellido, e-mail y la contraseña dos veces y presionara en el botón Crear 6. El sistema dará de alta un usuario nuevo | 6. 1. El usuario no cumple con las normas de seguridad de la contraseña  6. 1.1. El sistema muestra un mensaje “Las contraseñas deben tener 8 caracteres una mayúscula y un número”  6.2. El usuario no ingresa las contraseñas iguales  6.2.1. El sistema muestra un mensaje “Las contraseñas no coinciden”  6.3. El usuario ingresa un mail ya existente en la base  6.3.1 El sistema muestra un mensaje “Ya existe un usuario registrado con ese email”  6.4 El usuario no ingresa algún campo requerido  6.4.1 El sistema muestra un mensaje con “Todos los campos son requeridos” |
| Observaciones |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 23 | Login |
| Actores | Usuario con cuenta creada |
| Descripción | El usuario se podrá loguear al sistema |
| Precondición | Contar con un usuario |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario abre la aplicación 2. El Sistema muestra la pantalla de Login 3. El usuario ingresa su usuario y contraseña 4. El sistema muestra la pantalla de Cuenta | 4.1 El usuario ingresa email y password erróneos o inexistentes  4.1.1 El sistema muestra el mensaje “Usuario no encontrado” |
| Observaciones | Luego de ingresar a la cuenta se guardan los datos hasta el logout |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 24 | Login con facebook |
| Actores | Usuario no logueado |
| Descripción | El usuario se loguea a la app mobile con su cuenta con facebook |
| Precondición | Contar con un usuario de facebook |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario abre la aplicación 2. El Sistema muestra la pantalla de login 3. El usuario presiona el botón “Ingresar con facebook@ 4. Se abre la pagina de facebook pidiendo las credenciales y permisos necesarios 5. El usuario ingresa las credenciale y acepta los permisos 6. Si el usuario cuenta con una cuenta en la app y los mails coinciden se loguea a la app y se redirige al menu principal | 5.1 el usuario no acepta los permisos  5.2 El sistema no se loguea con facebook y se queda en la pantalla de login de la app  6.1 El usuario no tiene una cuenta en la app o el mail registrado es distinto al de facebook  6.2 el sistema le crea una cuenta usuario en la app para que pueda utilizar la aplicación mobile |
| Observaciones |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 25 | Listado de servicios |
| Actores | Usuario logueado en App |
| Descripción | El usuario visualizará una lista de servicios vigentes |
| Precondición | Que haya servicios vigentes |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario presiona la opción de servicios en el menú inferior de la pantalla 2. El Sistema muestra la pantalla de servicios 3. El usuario visualiza los servicios vigentes | 3.1. No hay servicios vigentes  3.1.1 El sistema muestra el siguiente texto en pantalla “No hay servicios disponibles” |
| Observaciones | Si se vuelve a hacer foco en servicios actualizará el listado si es que se ingresaron nuevos servicios |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 26 | Contratar servicio |
| Actores | Usuario logueado en App |
| Descripción | El usuario pagará un servicio |
| Precondición | Que haya servicios vigentes |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario presiona la opción de Servicios en el menú inferior de la pantalla 2. El Sistema muestra la pantalla de servicios 3. El usuario presiona en el botón Contratar 4. El sistema abre la URL de Mercado Pago en un navegador | 4.1. El sistema no puede acceder a la URL  4.1.1. El sistema muestra un mensaje “Ha ocurrido un error , intente mas tarde” |
| Observaciones |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 27 | Listado de eventos |
| Actores | Usuario logueado en App |
| Descripción | El usuario visualizará una lista de eventos vigentes |
| Precondición | Que haya eventos vigentes |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario presiona la opción de Eventos en el menú inferior de la pantalla 2. El Sistema muestra la pantalla de Eventos 3. El usuario visualiza los eventos vigentes | 3.1. No hay eventos vigentes  3.1.1 El sistema muestra el siguiente texto en pantalla “No hay eventos disponibles” |
| Observaciones | Si se vuelve a hacer foco en Eventos actualizará el listado si es que se ingresaron nuevos eventos |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 28 | Listado de mis mascotas |
| Actores | Usuario logueado en App |
| Descripción | El usuario visualizará una lista de mascotas ingresadas |
| Precondición | Que haya ingresado mascotas |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario presiona la opción de Mis Mascotas en el menú inferior de la pantalla 2. El Sistema muestra la pantalla de Mis Mascotas 3. El usuario visualiza las mascotas ingresadas | 3.1. Si no ingreso mascotas  3.1.1 El sistema muestra el siguiente texto en pantalla “Aún no cargaste mascotas” |
| Observaciones |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 29 | Crear mascota |
| Actores | Usuario logueado en App |
| Descripción | El usuario dará de alta una mascota |
| Precondición | Que tenga una cuenta creada |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario presiona la opción de Mis Mascotas en el menú inferior de la pantalla 2. El Sistema muestra la pantalla de Mis Mascotas 3. El presiona el ícono “+” ubicado en la parte inferior derecha de la pantalla 4. El Sistema muestra la pantalla de Crear Mascota 5. El usuario ingresa ubicación, una imagen, nombre, edad, sexo, tamano, descripcion, tipo de mascota y presiona Guardar 6. El sistema ingresa la mascota | 6.1 El usuario no ingresa algún dato obligatorio  61.1 El sistema muestra un mensaje “Todos los campos son requeridos” |
| Observaciones | Al guardar se redirigirá a la pantalla de Mis Mascotas en la que se agregara a la lista la mascota creada |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 30 | Mascota adoptada |
| Actores | Usuario logueado en App |
| Descripción | El usuario marca una mascota como adoptada |
| Precondición | Que tenga mascotas cargadas |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario presiona la opción de Mis Mascotas en el menú inferior de la pantalla 2. El Sistema muestra la pantalla de Mis Mascotas 3. El usuario presiona en el botón “Me Adoptaron!” 4. El sistema muestra un mensaje de felicitaciones |  |
| Observaciones | Luego de dar de baja una mascota el botón de “Me Adoptaron!” se convierte en el texto “Adoptada!” |
| PostCondición | La mascota adoptada desaparece del listado de mascotas disponibles |

|  |  |
| --- | --- |
| Casos de uso 31 | Listado de mascotas en adopción |
| Actores | Usuario logueado en App |
| Descripción | El usuario visualizará un listado de mascotas en adopción |
| Precondición | Que haya mascotas en adopción |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario presiona la opción de Buscar en el menú inferior de la pantalla 2. El Sistema muestra la pantalla Buscar 3. El usuario visualiza el listado de mascotas en adopción | 3.1. El sistema no tiene mascotas cargadas  3.1.1. La pantalla Buscar muestra el texto “No hay mascotas disponibles para los filtros aplicados” |
| Observaciones |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 32 | Filtrar mascotas |
| Actores | Usuario logueado en App |
| Descripción | El usuario visualizará un listado filtrado de mascotas en adopción |
| Precondición | Que haya mascotas en adopción |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario presiona la opción de Buscar en el menú inferior de la pantalla 2. El Sistema muestra la pantalla Buscar 3. El usuario presiona el botón de Filtros en la parte superior derecha de la pantalla 4. El Sistema muestra la pantalla Filtros 5. El usuario ingresa los valores deseados entre sexo, tamano, distancia y presiona el boton Aplicar 6. El Sistema muestra un listado de mascotas en la pantalla Buscar correspondiente a los filtros ingresados. | 6.1. El sistema no tiene mascotas entre esos valores  6.1.1. La pantalla Buscar muestra el texto “No hay mascotas disponibles para los filtros aplicados” |
| Observaciones |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 33 | Contactar a otro usuario |
| Actores | Usuario logueado en App |
| Descripción | El usuario se contactara con otro usuario |
| Precondición | Que haya mascotas en adopción |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario presiona la opción de Buscar en el menú inferior de la pantalla 2. El Sistema muestra la pantalla Buscar 3. El usuario presiona el icono de email 4. El sistema muestra un mensaje “Contacto Hecho! El rescatista se comunicara con vos en breve” |  |
| Observaciones |  |
| PostCondición |  |

|  |  |
| --- | --- |
| Casos de uso 34 | Logout |
| Actores | Usuario logueado en App |
| Descripción | El usuario se desloguea de la aplicación |
| Precondición | Que este logueado |
| Estado | Completo |
| Curso Normal | Curso Alternativo |
| 1. El Usuario presiona la opción de Cuenta en el menú inferior de la pantalla 2. El Sistema muestra la pantalla Cuenta 3. El usuario presiona el botón “Cerrar sesión” 4. El Sistema muestra la pantalla Login |  |
| Observaciones |  |
| PostCondición |  |

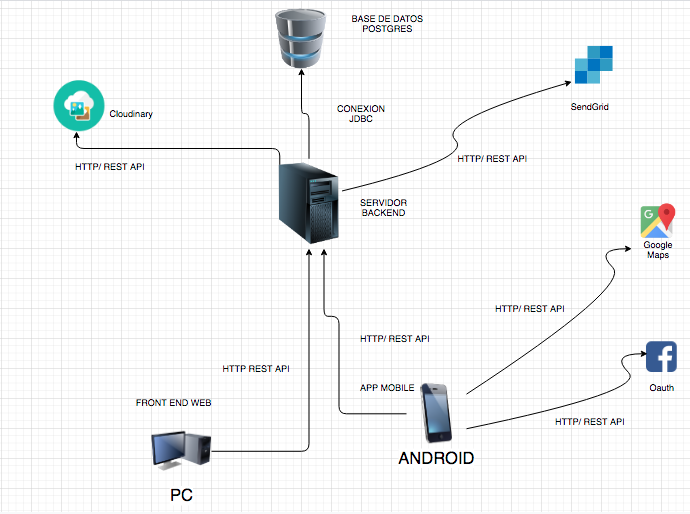
# Diagrama de clases

[*https://drive.google.com/file/d/1UY\_evvqxIUHE4Fibm77h9PTiRa6OMQLt/view?usp=sharing*](https://drive.google.com/file/d/1UY_evvqxIUHE4Fibm77h9PTiRa6OMQLt/view?usp=sharing)

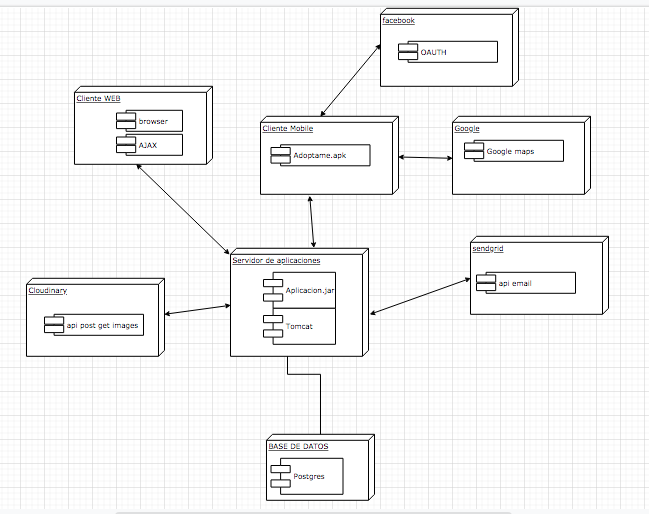
# DER

<https://drive.google.com/file/d/1UY_evvqxIUHE4Fibm77h9PTiRa6OMQLt/view?usp=sharing>

# Diagrama de arquitectura



# Diagrama de despliegue

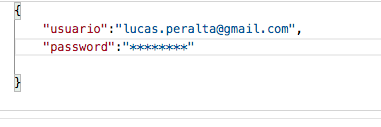


# Detalle de interfaz entre sistemas

Para comunicarse desde y hacia el servidor tanto WEB como MOBILE utilizamos servicios REST , cuya interfaz es en formato JSON y MULTIPART-FORM-DATA . En la pagina Web la navegación entre paginas es manejada por spring boot,

Por ejemplo el login desde mobile

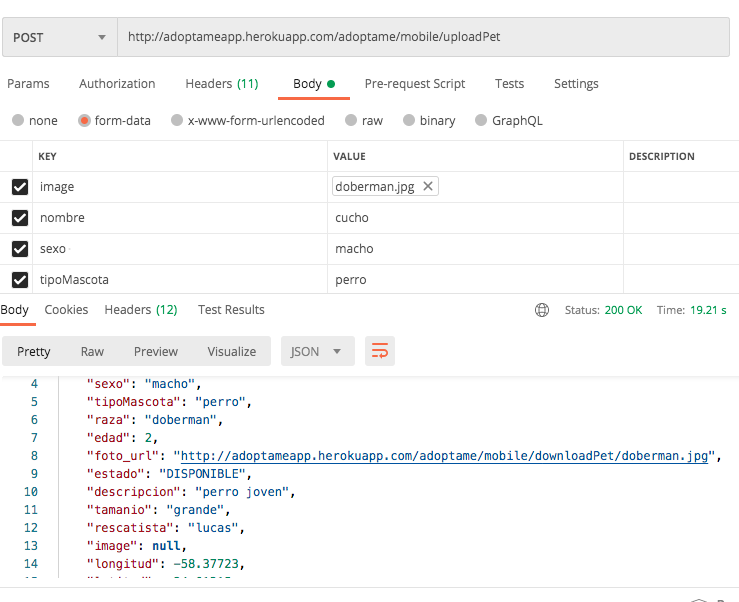
LOGIN REQUEST



RESPONSE



El único servicio que no utiliza JSON en el request es el servicio para crear mascota , ya que al tener que subir una imagen al servidor usamos **Multipart/form-data** para que sea mas eficiente que subir una imagen en base 64 mediante JSON, pero su respuesta es en formato JSON



para los servicios de envió de mail utilizamos el proveedor [SENDGRID](https://sendgrid.com/) el cual mediante una api java (que es solo un wrapper de servicios REST) permite realizar el envió de mail.

Ejemplo:

Email from = **new** Email(Constantes.***FROM\_ADDRESS***);  
  
Email to2 = **new** Email(to);  
Content content = **new** Content(**"text/plain"**, text);  
Mail mail = **new** Mail(from, subject, to2, content);  
System.***out***.println(**"api key sendgrid"**+ **sendGridApiKey**);  
SendGrid sg = **new** SendGrid(**sendGridApiKey**);  
Request request = **new** Request();  
**try** {  
 request.setMethod(Method.***POST***);  
 request.setEndpoint(**"mail/send"**);  
 request.setBody(mail.build());  
 Response response = sg.api(request);

Para el servicio de envio y almacenamiento de imágenes usamos un servicio de CDN (content delivery network) llamada cloudinary que nos permite almacenar imágenes y recuperarlas mediante servicios REST, para almacenar la imagen usamos una api java provista por cloudinary (es un wrapper de servicios rest) y para recuperar la imagen simplemente invocamos a la url que nos indica donde esta almacenada

Ejemplo upload:

Map config = **new** HashMap();  
config.put(**"cloud\_name"**, **cloudName**);  
config.put(**"api\_key"**, **apikey**);  
config.put(**"api\_secret"**, **apiSecret**);  
Cloudinary cloudinary = **new** Cloudinary(config);  
String fileName = **fileStorageService**.storeFile(params.getImage());  
String basePath=**fileStorageService**.getFileStorageLocation().toString();  
***logger***.info(**"path imagen en servidor "**+basePath);  
Map resultado;  
**try** {  
 **resultado= cloudinary.uploader().upload(basePath+"/"+fileName, ObjectUtils.*emptyMap*());**  
} **catch** (IOException e) {  
 e.printStackTrace();  
 **throw new** ImageUploadException();  
}

Ejemplo recuperar imagen

<https://res.cloudinary.com/ddetocnzj/image/upload/v1603594091/ifzyf71hdxkmn17gpb5x.jpg>

# Prototipo de interfaces de usuario

Prototipo Login

