



ERS

Proyecto: Control de expedientes de pacientes

Tabla de contenido

1- Introducción	5
1.1 Propósito	5
1.2 Descripción del problema	5
1.3 Lista de problemas detectados	5
1.4 Lista de fortalezas detectadas	5
1.5 Objetivos del sistema	5
1.5.1 Objetivo general	5
1.5.2 Objetivos específicos	6
1.5.2.1 Caso de uso 1	6
1.5.2.1 Caso de uso 2	6
1.5.2.1 Caso de uso 3	6
1.5.2.1 Caso de uso 4	6
1.5.2.1 Caso de uso 5	6
1.6 Perspectiva del producto por desarrollar (iteración 1).	7
1.7 Reglas del negocio	7
1.8 Suposiciones y dependencias	7
1.9 Alcances del sistema	7
1.10 Limitaciones o restricciones	8
1.11 SkateHolders y sus necesidades	8
1.12 Visión general de la estructura	8
2 Requerimientos funcionales	9
2.1 Contexto del sistema	9
2.1.1 Diagrama de contexto	9
2.2 Descripción detallada de cada caso de uso	9
2.2.1 Caso de uso 1	9
2.2.1.1 Texto del caso de uso	9
2.2.1.2 Pantalla y/o reporte del caso de uso	11
Registrar pacientes	11
Editar pacientes	11
Eliminar perfil de pacientes	12
Buscador de pacientes	12
2.2.1.3 Diagrama de actividades del caso de uso	13
2.2.1.4 Pruebas del caso de uso	13
2.2.1 Caso de uso 2	14
2.2.1.1 Texto del caso de uso	14
2.2.1.2 Pantalla y/o reporte del caso de uso	15
Registrar médico	15
Editar médico	15

Buscador:	16
2.2.1.3 Diagrama de actividades del caso de uso	17
2.2.1.4 Pruebas del caso de uso	17
2.2.1 Caso de uso 3	18
2.2.1.1 Texto del caso de uso	18
2.2.1.2 Pantalla y/o reporte del caso de uso	19
Registrar enfermedad	19
Editar enfermedad	19
Eliminar enfermedad	20
2.2.1.3 Diagrama de actividades del caso de uso	21
2.2.1.4 Pruebas del caso de uso	21
2.2.1 Caso de uso 4	22
2.2.1.1 Texto del caso de uso	22
2.2.1.2 Pantalla y/o reporte del caso de uso	23
Registrar cita	23
Editar cita	23
Eliminar cita	23
2.2.1.3 Diagrama de actividades del caso de uso	24
2.2.1.4 Pruebas del caso de uso	24
2.2.1 Caso de uso 5	25
2.2.1.1 Texto del caso de uso	25
2.2.1.2 Pantalla y/o reporte del caso de uso	26
Registrar observación	26
Editar observación	26
Eliminar observación	26
2.2.1.3 Diagrama de actividades del caso de uso	27
2.2.1.4 Pruebas del caso de uso	27
2.3 Patrones de requerimientos	27
2.3.1 Funciones de usuarios	27
3. Requerimientos no funcionales	29
3.1 Patrones de requerimientos del producto	29
3.1.1 Interfaz local del usuario	29
3.1.1.1 Dashboard	30
3.1.1.2 Interfaz de funciones	30
3.1.2 Rendimiento	31
3.1.3 Control de acceso	31
3.1.4 Comerciales	31
3.2 Organizacionales	31
3.2.1 Patrón documentación	31
3.2.2 Patrón tecnología	31
3.2.3 Patrón de diseño	31

Modelo (model)	32
Vista (view)	32
Controlado (Controller)	32
3.2.4 Entregas	32
3.2.5 Implementación	32
Apéndices	32

1- Introducción

1.1 Propósito

Se definirán cuales son los requerimientos que debe tener un programa que gestione correctamente un control de expedientes de usuarios.

El programa fue encargado por la ingeniera Adriana Alvarez, con el fin de ser usado en el hospital Calderón Guardia.

1.2 Descripción del problema

El problema de tener que manejar expedientes de forma física afecta a los(as) secretarios(as) y doctores(as) reduciendo el tiempo de búsqueda y manejo de éstos, además de poseer un alto costo en producción y mantenimiento. Una solución para este problema es crear un control digital de expedientes de pacientes, que pueda ser usado en computadoras y en dispositivos móviles, donde un doctor pueda ingresar al sistema con su nombre de usuario y contraseña y poder gestionar desde ahí pacientes, citas y enfermedades. Así como un usuario administrador que también pueda registrar doctores.

1.3 Lista de problemas detectados

- Costo de producción de expedientes.
- Costo de mantenimiento de expedientes.
- Espacio físico asignado al almacenamiento de expedientes.
- Alta durabilidad al crear y actualizar un nuevo expediente.
- No existen respaldos de expedientes.

1.4 Lista de fortalezas detectadas

- El aprendizaje del manejo de expedientes en físico generalmente resulta más sencillo. Esto por el miedo que aún hay en la tecnología.

1.5 Objetivos del sistema

1.5.1 Objetivo general

Desarrollar un producto de software capaz de automatizar los procesos ligados al control de los expedientes de pacientes del Hospital Calderón Guardia.

1.5.2 Objetivos específicos

1.5.2.1 Caso de uso 1

Objetivo específico: Mantenimiento de pacientes: el registro, visualización, búsqueda, edición y eliminación.

Criterio de éxito: Debe cumplir con los siguientes puntos.

- Entrada correcta y fácil de datos para el registro de pacientes.
- Entrada correcta y fácil de datos para la edición de pacientes.
- Búsqueda eficiente y con el mínimo tiempo de espera.
- Interfaz de usuario amigable.

1.5.2.1 Caso de uso 2

Objetivo específico: Mantenimiento de doctores: el registro, visualización, búsqueda, edición y eliminación.

Criterio de éxito: Debe cumplir con los siguientes puntos.

- Entrada correcta y fácil de datos para el registro de doctores.
- Entrada correcta y fácil de datos para la edición de doctores.
- Búsqueda eficiente y con el mínimo tiempo de espera.
- Interfaz de usuario amigable.

1.5.2.1 Caso de uso 3

Objetivo específico: Mantenimiento de enfermedades: el registro, visualización, búsqueda, edición y eliminación.

Criterio de éxito: Debe cumplir con los siguientes puntos.

- Entrada correcta y fácil de datos para el registro de enfermedades.
- Entrada correcta y fácil de datos para la edición de enfermedades.
- Búsqueda eficiente y con el mínimo tiempo de espera.
- Interfaz de usuario amigable.

1.5.2.1 Caso de uso 4

Objetivo específico: Control de citas: Control del expediente y citas del paciente, fecha de la cita, médico que lo atendió.

Criterio de éxito: Debe cumplir con los siguientes puntos.

- Entrada correcta y fácil de datos para la asignación de una cita a un paciente.
- Entrada correcta y fácil de datos para la edición de una cita a un paciente.

1.5.2.1 Caso de uso 5

Objetivo específico: Control de observaciones: Ingreso de las observaciones tomadas por médicos, tanto para citas como para pacientes.

Criterio de éxito: Debe cumplir con los siguientes puntos.

- Entrada correcta y fácil de datos para la creación de una observación a un paciente.
- Entrada correcta y fácil de datos para la edición de una observación a un paciente.
- Entrada correcta y fácil de datos para la creación de una observación a una cita.
- Entrada correcta y fácil de datos para la edición de una observación a una cita.

1.6 Perspectiva del producto por desarrollar (iteración 1).

Para médicos que necesitan tener un control más eficiente de expedientes, el Control de Expedientes es un sistema administrador en la web que provee acceso a un sistema donde se puede llevar el control de los pacientes, doctores, citas, enfermedades y observaciones. A diferencia del control actual de expedientes nuestro sistema trae mejoras en el tiempo de adquisición, de creación, de edición y eliminación de expedientes de pacientes, así como de sus citas, observaciones.

1.7 Reglas del negocio

Por completar, recomienda eliminar este punto

1.8 Suposiciones y dependencias

El producto de software a desarrollar tendrá una aplicación web, que desarrollada en el framework de Ruby on Rails. Alojada en un servidor local (por disposición del curso), y ligada a una base de datos en línea de MySQL.

La información de los pacientes será ingresada una vez entregado el producto y no durante el desarrollo del proyecto.

El control y uso de la aplicación será exclusivo de un usuario administrador.

1.9 Alcances del sistema

El sistema se debe encargar de llevar un control de expedientes de pacientes, permitir el control de pacientes, médicos y enfermedades. Además de llevar un control de las citas de un paciente, observaciones tomadas, médico que lo atendió, entre otros datos. Así mismo, debe permitir realizar búsquedas de pacientes, médicos y enfermedades. El sistema web solo tendrá los casos de uso especificados en este documento.

1.10 Limitaciones o restricciones

El producto de software solo cumplirá con los requerimientos especificados y los casos de uso respectivos.

Queda excluida cualquier otra funcionalidad no especificada en este documento.

La implementación y mantenimiento del sistema quedan excluidos por parte del equipo del desarrollo.

La inserción de los datos queda excluida por parte del equipo de desarrollo.

1.11 SkateHolders y sus necesidades

Administrador: Uso completo del sistema.

Médico: Uso completo del sistema, excepto el registro, edición y eliminación de médicos.

Nota: Los pacientes no usaran el sistema, sin embargo tienen un perfil dentro de éste el cual será administrado por el administrador de la aplicación o el médico con previo registro.

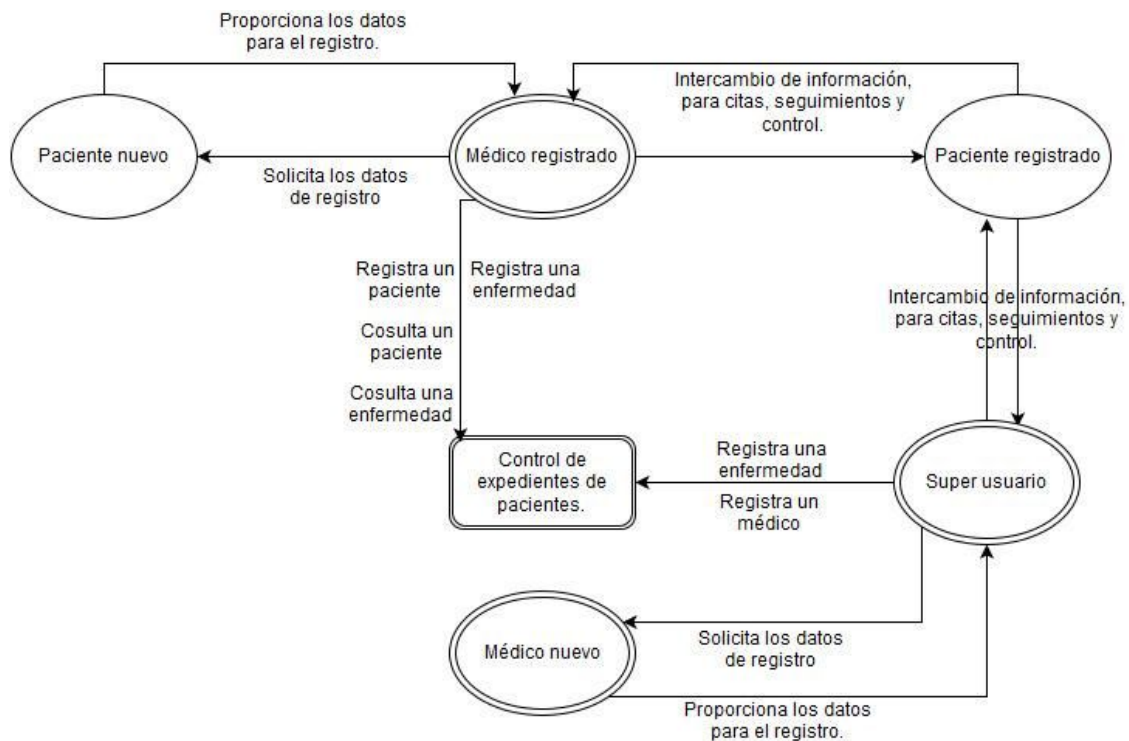
1.12 Visión general de la estructura

En esta primera iteración la se cubren los cinco casos de uso del sistema web. Con esto quedará finalizada la aplicación web, con todas las funcionalidades acá mencionadas.

2 Requerimientos funcionales

2.1 Contexto del sistema

2.1.1 Diagrama de contexto



2.2 Descripción detallada de cada caso de uso

2.2.1 Caso de uso 1

2.2.1.1 Texto del caso de uso


ID y nombre:	CU1 - Mantenimiento de pacientes
Creado por:	Edisson López y Kenneth Quirós
Fecha de creación:	04-01-2017
Primer actor:	Administrador y médicos
Segundos actores:	Pacientes
Descripción:	Realiza el registro de un expediente de un paciente. Toma los datos personales de él y crea un perfil dentro del sistema. Luego este puede ser visualizado, así como editado y eliminado. También está la función de búsqueda.

Disparador:	Para acceder a éste en el dashboard (barra lateral izquierda) de la aplicación, habrá un botón donde se podrá hacer uso éste.
Precondiciones:	PRE1: Identificación del usuario, como administrador o médico, con previo registro en el sistema. PRE2: Presionar el botón de Pacientes en la barra lateral izquierda.
Postcondiciones:	POST1: El cambio es realizado en el sistema. POST2: Se guarda/actualiza el cambio en la base de datos.
Flujo normal:	1. Se loguea en el sistema web. 2. Presiona el botón de pacientes en la barra lateral izquierda. 3. Selecciona alguna de las siguientes opciones: Registro, edición, eliminación, visualización y búsqueda. 4. Completa el registro, y presiona el botón indicado para finalizar.
Fluto alternativo:	1. Dentro de la visualización del paciente, se puede editar o eliminar a éste. 2. Dentro de la búsqueda del paciente, se puede entrar a la visualización de alguno de los pacientes dados en los resultados.
Excepciones:	
Prioridad:	Muy alta
Frecuencia de uso:	Muy alta
Suposiciones:	El médico debe haber ingresado al sistema.
Otra información:	

2.2.1.2 Pantalla y/o reporte del caso de uso

Registrar pacientes

Nuevo paciente



Nombre

Nombre

Primer apellido

Segundo apellido

Cédula

Cédula 0-0000-0000

Género

Género

Imagen de paciente

Seleccionar archivo Ningún archivo seleccionado

Nacionalidad

Nacionalidad

Fecha de nacimiento

Día

Mes

Año

Fecha de fallecimiento


Día

Mes

Año

Editar pacientes

Información personal



Nombre:

k

Primer apellido:

qqq

Segundo apellido:


nnn

Citas

Editar

Desde el perfil de del paciente, se presiona el botón de *Editar*. Abriendo la ventana de edición del perfil, la que se muestra a continuación:

Editar paciente



Nombre

Cédula

Género

Eliminar perfil de pacientes

En la parte de abajo del perfil del paciente, está el botón de eliminar, el siguiente:

Eliminar paciente

✕ Eliminar

Al presionarlo nos devuelve un mensaje de si "Desea continuar", y presionamos aceptar.

controldeexpedientesvirtual-kenquiros64.c9users.io dice:

Desea continuar?

Aceptar
Cancelar

Buscador de pacientes

Buscar un paciente

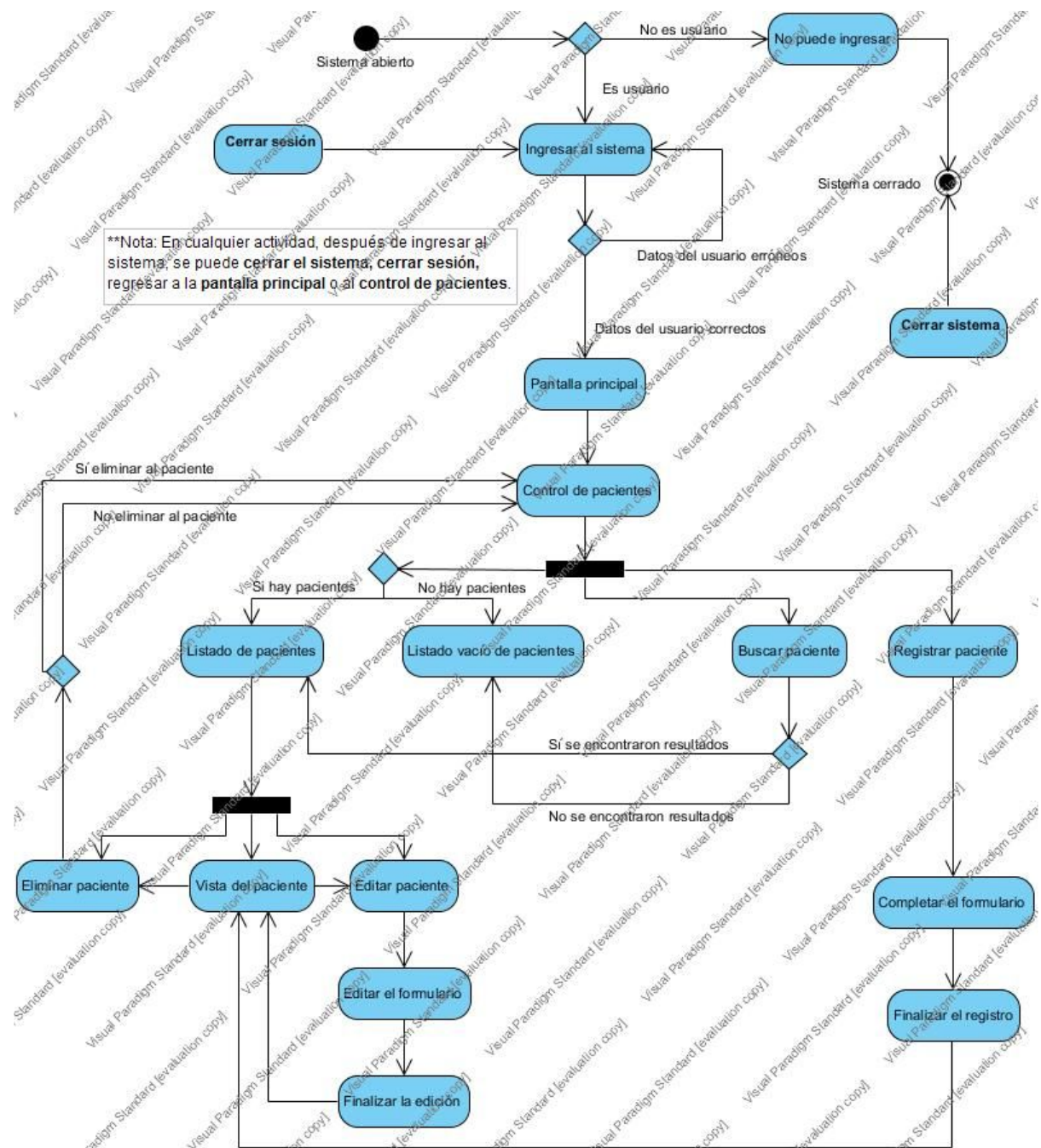
Lista de pacientes

Buscar

Cédula	Nombre			
7-1234-1234	k qqq nnn	Mostrar	Editar	✕ Eliminar
7-0224-0442	kk kk kk	Mostrar	Editar	✕ Eliminar
7-1235-1235	kkasd kk kknknk	Mostrar	Editar	✕ Eliminar
7-2234-1213	kk kk kk	Mostrar	Editar	✕ Eliminar

Se introduce el nombre o el número de cédula del paciente, se presiona el botón de buscar. Abajo aparecen los resultados, desplegando 5 columnas: número de cédula, nombre, opción de mostrar, opción de editar y opción de eliminar.

2.2.1.3 Diagrama de actividades del caso de uso



2.2.1.4 Pruebas del caso de uso

Para realizar las pruebas se utilizó la tecnología de Ruby on Rails, usando dos gemas para escribir pruebas automatizadas, imitando el comportamiento de un usuario real,

interactuando con el código HTML de nuestro sistema web. Las dos gemas fueron Selenium-webdriver y rspec. Estas se adjuntan en el anexo y dentro del sistema web.

2.2.1 Caso de uso 2


2.2.1.1 Texto del caso de uso

ID y nombre:	CU2 - Mantenimiento de médicos
Creado por:	Edisson López y Kenneth Quirós
Fecha de creación:	04-01-2017
Primer actor:	Administrador, médicos.
Segundos actores:	Médicos
Descripción:	Realiza el registro de un médico. Toma los datos personales de él y crea un perfil dentro del sistema. Luego este puede ser visualizado, así como editado y eliminado. También está la función de búsqueda.
Disparador:	Para acceder a éste en el dashboard (barra lateral izquierda) de la aplicación, habrá un botón donde se podrá hacer uso éste.
Precondiciones:	PRE1: Identificación del usuario administrador o médico. PRE2: Presionar el botón de Médicos en la barra lateral izquierda.
Postcondiciones:	POST1: El cambio es realizado en el sistema. POST2: Se guarda/actualiza el cambio en la base de datos.
Flujo normal:	1. El administrador o el médico se loguea en el sistema web. 2. Presiona el botón de médicos en la barra lateral izquierda. 3. Selecciona alguna de las siguientes opciones: Registro, edición, eliminación, visualización o búsqueda. Las primeras 3 funciones solo la podrán ejecutar los administradores del sistema. 4. Completa el registro, y presiona el botón indicado para finalizar.
Fluto alternativo:	1. Dentro de la visualización del médico, se puede editar o eliminar a éste. Este solo puede ser ejecutado por el administrador. 2. Dentro de la búsqueda del médico, se puede entrar a la visualización de alguno de los médicos dados en los resultados.
Excepciones:	El registro, edición y eliminación de médicos solo podrá ser realizada por el administrador.
Prioridad:	Alta
Frecuencia de uso:	Media
Suposiciones:	El administrador o el médico debe haber ingresado al sistema.
Otra información:	Solo el o los administradores pueden hacer registros, cambios o eliminación de los médicos.

2.2.1.2 Pantalla y/o reporte del caso de uso

Registrar médico

Nuevo médico



Nombre

Cédula

Cédula

Nacionalidad

☐ Inactivo

Editar médico

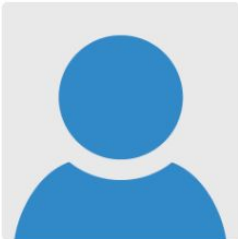
En la lista de médicos está el botón de *Editar*:

Lista de médicos

Código	Cédula	Nombre	
		Administrador	<input type="button" value="Mostrar"/>
ME-1	7-1234-1234	k qn n	<input type="button" value="Mostrar"/>

Pantalla de edición:

Editar Medico



Nombre

Edisson

López

Díaz

Buscador:

Buscar un médico

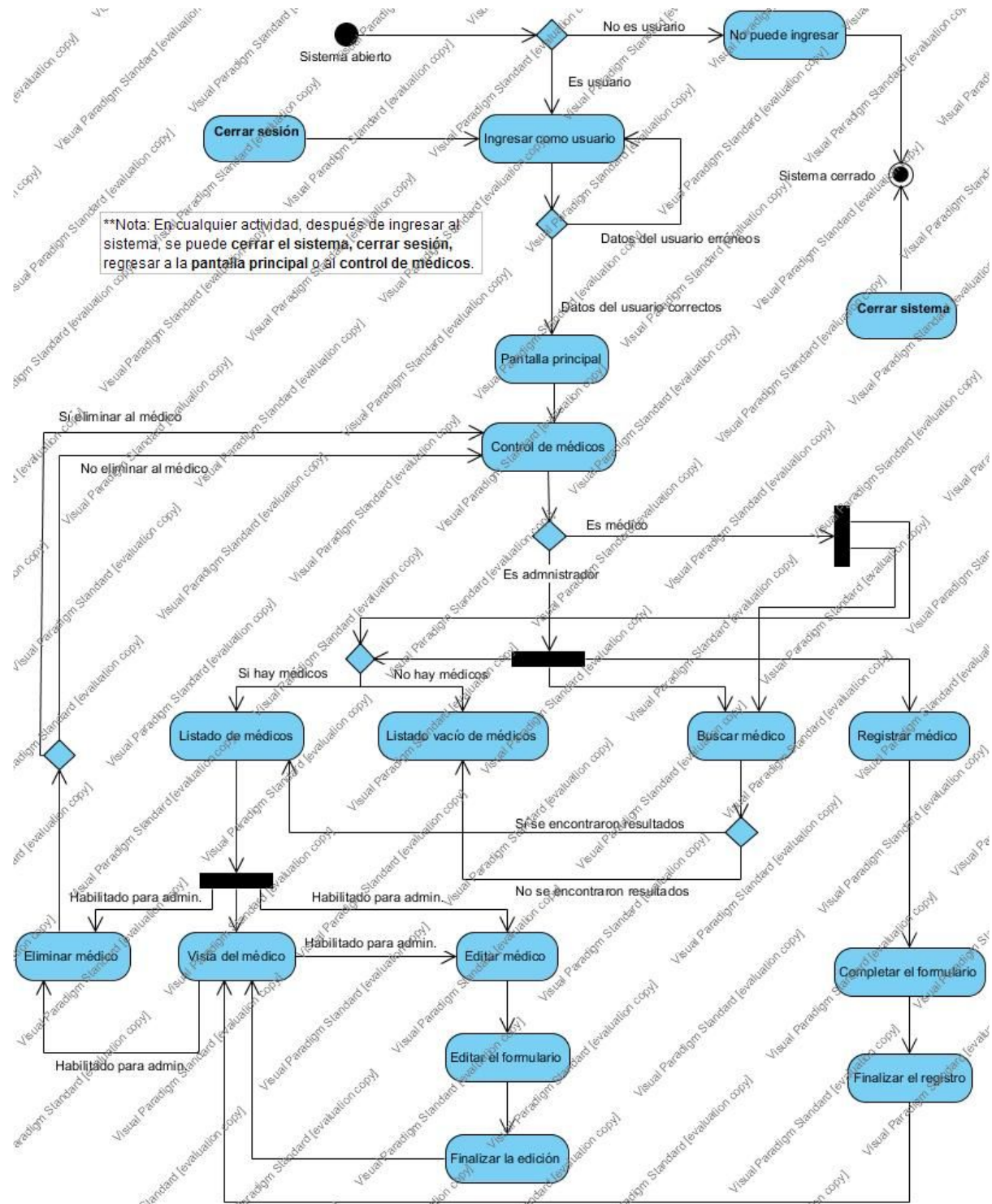
Lista de médicos

Buscar

Código	Cédula	Nombre	
ME-1	7-1234-1234	k qn n	<div>Mostrar</div>

Los resultados despliegan cuatro columnas, código, cédula, nombre y el botón de mostrar.

2.2.1.3 Diagrama de actividades del caso de uso



2.2.1.4 Pruebas del caso de uso

Para realizar las pruebas se utilizó la tecnología de Ruby on Rails, usando dos gemas para escribir pruebas automatizadas, imitando el comportamiento de un usuario real,

interactuando con el código HTML de nuestro sistema web. Las dos gemas fueron Selenium-webdriver y rspec. Estas se adjuntan en el anexo y dentro del sistema web.

2.2.1 Caso de uso 3

2.2.1.1 Texto del caso de uso

ID y nombre:	CU3 - Mantenimiento de enfermedades
Creado por:	Edisson López y Kenneth Quirós
Fecha de creación:	04-01-2017
Primer actor:	Administrador, médicos.
Segundos actores:	Pacientes.
Descripción:	Realiza el registro de una enfermedad. Crea un registro con el nombre, la nomenclatura y la descripción de ésta dentro del sistema. Luego este puede ser visualizado, así como editado y eliminado. También está la función de búsqueda.
Disparador:	Para acceder a éste en el dashboard (barra lateral izquierda) de la aplicación, habrá un botón donde se podrá hacer uso éste.
Precondiciones:	PRE1: Identificación del usuario administrador o médico. PRE2: Presionar el botón de Enfermedades en la barra lateral izquierda.
Postcondiciones:	POST1: El cambio es realizado en el sistema. POST2: Se guarda/actualiza el cambio en la base de datos.
Flujo normal:	1. El administrador o el médico ingresa en el sistema web. 2. Presiona el botón de enfermedades en la barra lateral izquierda. 3. Selecciona alguna de las siguientes opciones: Registro, edición, eliminación, visualización o búsqueda. Las primeras 3 funciones solo la podrán ejecutar los administradores del sistema. 4. Completa el registro, y presiona el botón indicado para finalizar.
Fluto alternativo:	1. Dentro de la visualización de la enfermedad, se puede editar o eliminar a ésta. Este solo puede ser ejecutado por el administrador. 2. Dentro de la búsqueda de la enfermedad, se puede entrar a la visualización de alguno de las enfermedades dadas en los resultados.
Excepciones:	El registro, edición y eliminación de enfermedades solo podrá ser realizada por el administrador.
Prioridad:	Media
Frecuencia de uso:	Media
Suposiciones:	El administrador o el médico debe haber ingresado al sistema.
Otra información:	Solo el o los administradores pueden hacer registros, cambios o eliminación de las enfermedades.

2.2.1.2 Pantalla y/o reporte del caso de uso

Registrar enfermedad

Nueva enfermedad

Código

Descripción

Finalizar

Editar enfermedad

Lista de enfermedades

[Nueva enfermedad](#)

Código	Descripción			
CA-01	Cáncer de Prostata	Mostrar	Editar	Eliminar
CA-02	Cáncer de mama	Mostrar	Editar	Eliminar

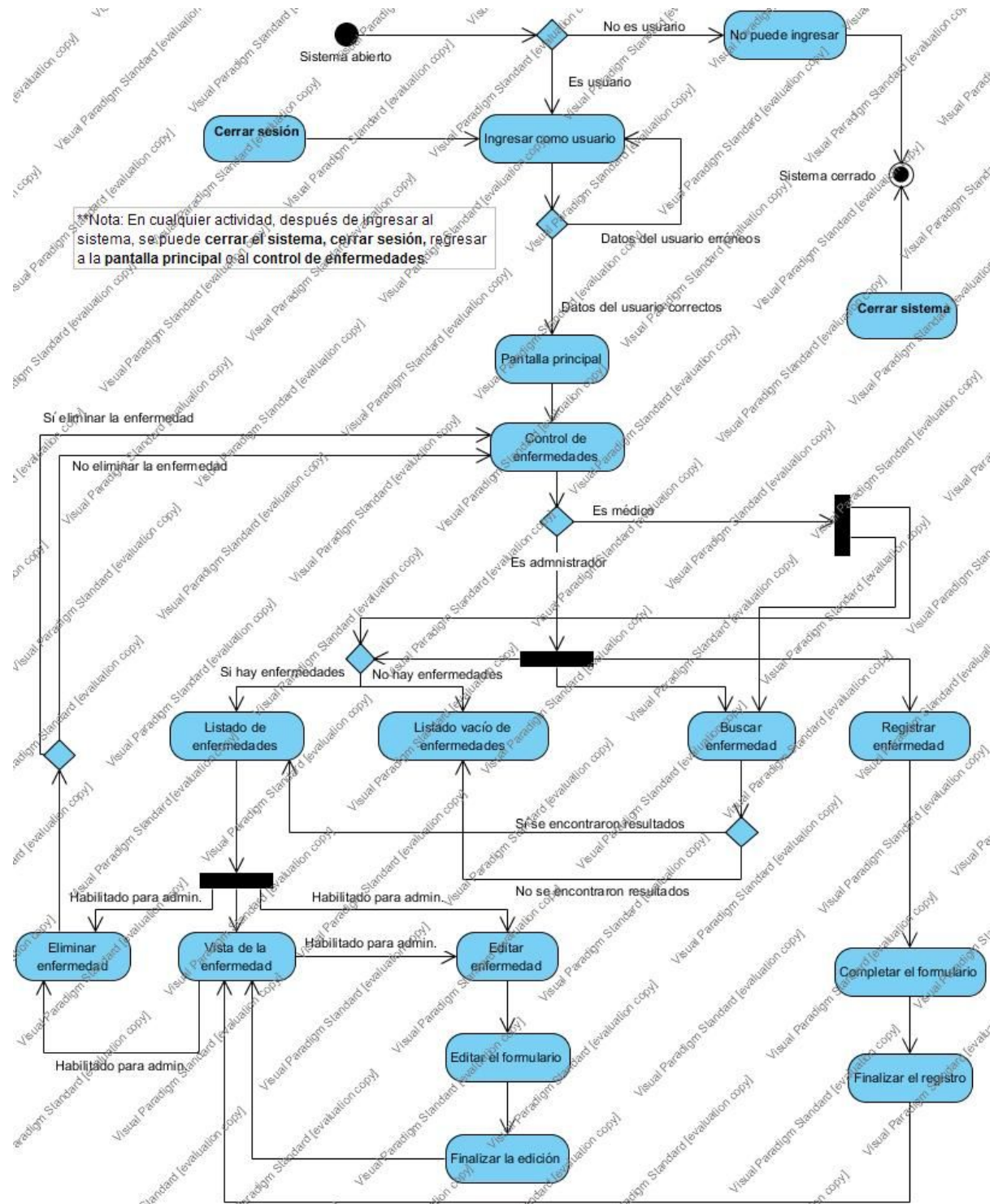
Eliminar enfermedad

Lista de enfermedades

[Nueva enfermedad](#)

Codigo	Descripcion			
CA-01	Cáncer de Prostata	Mostrar	Editar	✕ Eliminar
CA-02	Cáncer de mama	Mostrar	Editar	✕ Eliminar

2.2.1.3 Diagrama de actividades del caso de uso



2.2.1.4 Pruebas del caso de uso

Para realizar las pruebas se utilizó la tecnología de Ruby on Rails, usando dos gemas para escribir pruebas automatizadas, imitando el comportamiento de un usuario real,

interactuando con el código HTML de nuestro sistema web. Las dos gemas fueron Selenium-webdriver y rspec. Estas se adjuntan en el anexo y dentro del sistema web.

2.2.1 Caso de uso 4

2.2.1.1 Texto del caso de uso

ID y nombre:	CU4 - Gestión de citas
Creado por:	Edisson López y Kenneth Quirós
Fecha de creación:	04-01-2017
Primer actor:	Administrador, médicos.
Segundos actores:	Pacientes.
Descripción:	Realiza el registro de una cita a paciente. Luego este puede ser visualizada, así como editada y eliminada.
Disparador:	Para acceder a éste en el dashboard (barra lateral izquierda) de la aplicación, habrá un botón donde se podrá hacer uso éste. Además de tener un botón dentro del perfil de pacientes con el nombre de citas.
Precondiciones:	PRE1: Identificación del usuario administrador o médico. PRE2: Presionar el botón de Citas en la barra lateral izquierda.
Postcondiciones:	POST1: El cambio es realizado en el sistema. POST2: Se guarda/actualiza el cambio en la base de datos.
Flujo normal:	1. El administrador o el médico ingresa en el sistema web. 2. Presiona el botón de citas en la barra lateral izquierda. 3. Selecciona alguna de las siguientes opciones: Nueva cita, edición de citas, visualización de citas, eliminación de citas. 4. Completa el registro, y presiona el botón indicado para finalizar.
Fluto alternativo:	1.1 Dentro de la visualización del paciente, se pueden ver las citas que tiene ese paciente, presionando el botón de citas. 1.2 Se muestra un listado de las citas, además de un botón de nueva cita. En este se puede crear una nueva cita para el paciente.
Excepciones:	No se puede hacer una búsqueda global de citas. Solo se pueden visualizar todas las citas de un paciente en específico.
Prioridad:	Alta
Frecuencia de uso:	Alta
Suposiciones:	El administrador o el médico debe haber ingresado al sistema.
Otra información:	

2.2.1.2 Pantalla y/o reporte del caso de uso

Registrar cita

Nueva cita

Paciente:
k qq q nnn

Médico encargado:
Administrador

Fecha:
Día
Mes
Año

Hora:
Hora
Minutos

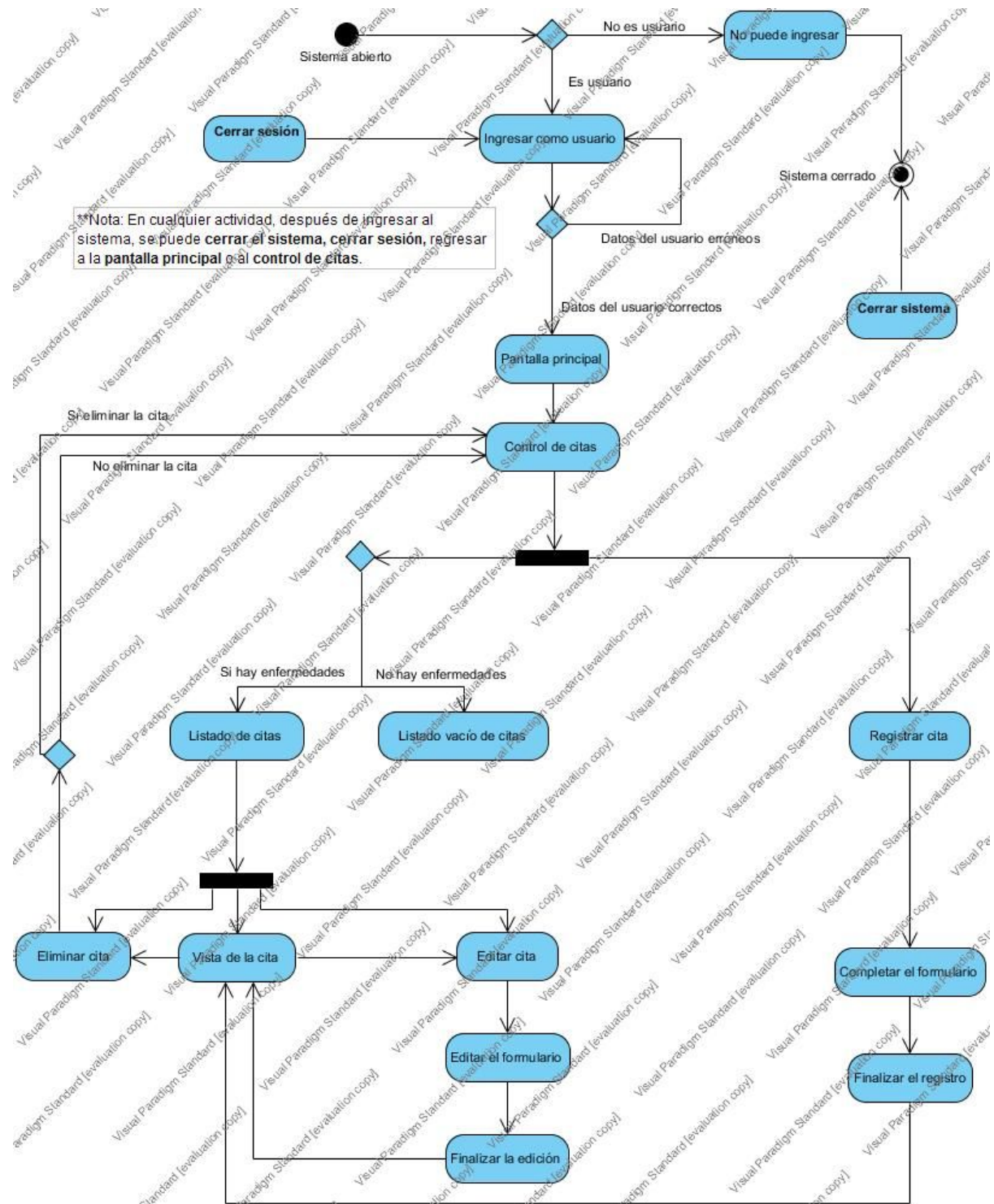
Editar cita

Lista de citas					
Fecha	Hora	Médico			
2105-10-15	13:11	Administrador	Mostrar	Editar	Eliminar
2115-03-02	04:02	Administrador	Mostrar	Editar	Eliminar

Eliminar cita

Lista de citas					
Fecha	Hora	Médico			
2105-10-15	13:11	Administrador	Mostrar	Editar	Eliminar
2115-03-02	04:02	Administrador	Mostrar	Editar	Eliminar

2.2.1.3 Diagrama de actividades del caso de uso



2.2.1.4 Pruebas del caso de uso

Para realizar las pruebas se utilizó la tecnología de Ruby on Rails, usando dos gemas para escribir pruebas automatizadas, imitando el comportamiento de un usuario real,

interactuando con el código HTML de nuestro sistema web. Las dos gemas fueron Selenium-webdriver y rspec. Estas se adjuntan en el anexo y dentro del sistema web.

2.2.1 Caso de uso 5

2.2.1.1 Texto del caso de uso

ID y nombre:	CU5 - Observaciones
Creado por:	Edisson López y Kenneth Quirós
Fecha de creación:	04-01-2017
Primer actor:	Administrador, médicos.
Segundos actores:	Pacientes.
Descripción:	Realiza el registro de una observación, tanto para pacientes como para citas. Luego esta puede ser visualizada, así como editada y eliminada.
Disparador:	Para acceder se puede hacer dentro de pacientes y citas. Dentro de estas dos pantallas hay un botón para la visualización de todas las observaciones registradas.
Precondiciones:	PRE1: Identificación del usuario administrador o médico. PRE2: Presionar el botón de observaciones. Este botón está ubicado en la pantalla de pacientes y de citas.
Postcondiciones:	POST1: El cambio es realizado en el sistema. POST2: Se guarda/actualiza el cambio en la base de datos.
Flujo normal:	1. El administrador o el médico ingresa en el sistema web. 2. Presiona el botón de observaciones en la pantalla de un paciente o en la pantalla de una cita. 3. En el listado de observaciones, se puede: 3.1. Seleccionar un observación: Aquí se puede visualizar, también está la opción de editar y eliminar. 3.2. Presionar el botón de crear nueva observación: Se completa el formulario y se presiona el botón de finalizar para completar la nueva observación.
Fluto alternativo:	1. Desde el perfil de un paciente, se presiona el botón de observaciones. 2. Desde la visualización de una cita, se presiona el botón de observaciones.
Excepciones:	No existen búsquedas de observaciones. Sin embargo se pueden ver todas las observaciones realizadas a un paciente, o a una cita.
Prioridad:	Media
Frecuencia de uso:	Media
Suposiciones:	El administrador o el médico debe haber ingresado al sistema.
Otra información:	

2.2.1.2 Pantalla y/o reporte del caso de uso

Registrar observación

Las observaciones están en la información del paciente y en la información de las citas. Se podrán agregar desde el siguiente botón:



Luego solo es necesario escribir la observación y presionar, otra vez, agregar:



Editar observación

Solo es necesario reescribir la observación.

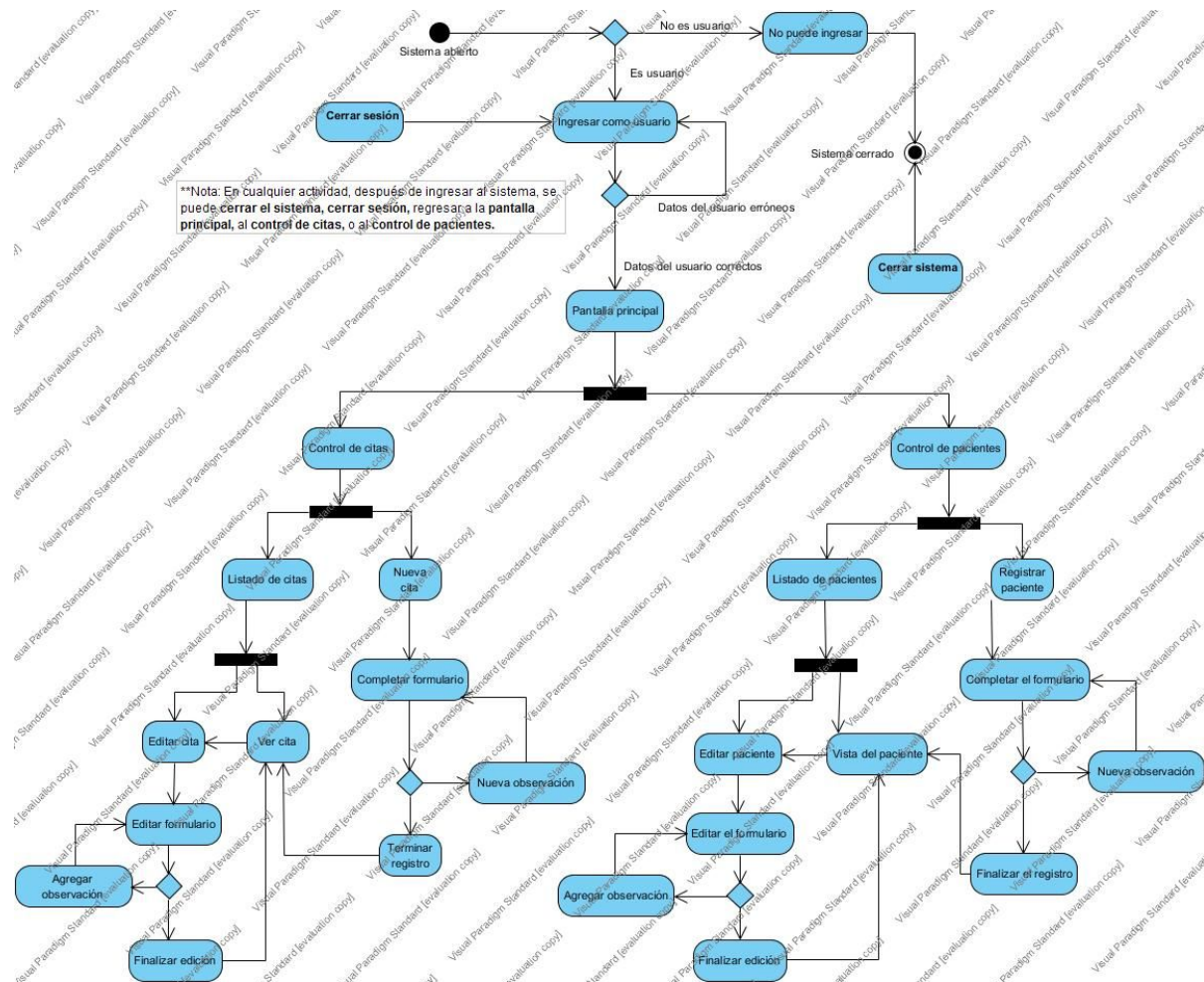


Eliminar observación



Solo se presiona el botón *Eliminar* y esta se borrara, sin preguntar nada.

2.2.1.3 Diagrama de actividades del caso de uso



2.2.1.4 Pruebas del caso de uso

Para realizar las pruebas se utilizó la tecnología de Ruby on Rails, usando dos gemas para escribir pruebas automatizadas, imitando el comportamiento de un usuario real, interactuando con el código HTML de nuestro sistema web. Las dos gemas fueron Selenium-webdriver y rspec. Estas se adjuntan en el anexo y dentro del sistema web.

2.3 Patrones de requerimientos

2.3.1 Funciones de usuarios

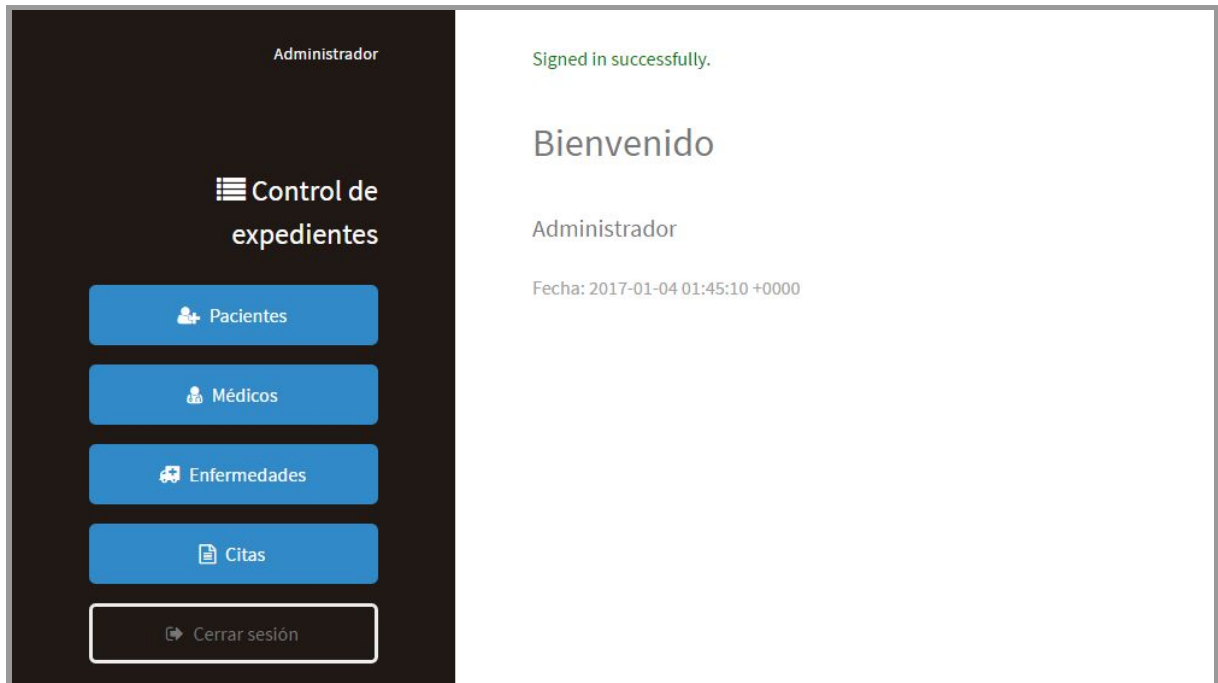
Funcionalidad	Administrador	Médico
CU1: Registrar pacientes.	✓	✓
CU1: Editar perfil de pacientes.	✓	✓

CU1: Eliminar perfil de pacientes.	✓	✓
CU1: Usar el buscador de pacientes.	✓	✓
CU2: Registrar médicos.	✓	✗
CU2: Editar perfil de médicos.	✓	✗
CU2: Eliminar el perfil de un médico del sistema.	✓	✗
CU2: Usar el buscador de médicos.	✓	✓
CU3: Registrar una enfermedad.	✓	✗
CU3: Editar la información de una enfermedad.	✓	✗
CU3: Eliminar una enfermedad del sistema.	✓	✗
CU3: Usar el buscador de enfermedades.	✓	✓
CU4: Crear citas a pacientes	✓	✓
CU4: Editar las citas de pacientes	✓	✓
CU4: Eliminar las citas de pacientes	✓	✓
CU5: Crear una observación de una cita.	✓	✓
CU5: Editar una observación de una cita.	✓	✓
CU5: Eliminar una observación de una cita.	✓	✓
CU5: Crear una observación de un paciente.	✓	✓
CU5: Editar una observación de un paciente.	✓	✓
CU5: Eliminar una observación de un paciente.	✓	✓

3. Requerimientos no funcionales

3.1 Patrones de requerimientos del producto

3.1.1 Interfaz local del usuario



La interfaz está dividida horizontalmente en dos partes. Una sección de botones (dashboard) y la sección de uso de funciones.

3.1.1.1 Dashboard



Título: Muestra el nombre de la aplicación: “Control de expedientes”.

Botón “Pacientes”: Redirige a la pantalla para poder hacer un control de los sistema de pacientes.

Botón “Médicos”: Redirige a la pantalla para poder hacer un control de los sistema de médicos.

Botón “Enfermedades”: Redirige a la pantalla para poder hacer un control de los sistema de enfermedades.

Botón “Citas”: Redirige a la pantalla para poder hacer un control de los sistema de citas.

Botón “Cerrar sesión”: Cerramos la sesión de usuario, redirige a la pantalla de inicio de sesión.

3.1.1.2 Interfaz de funciones



Mensaje de éxito: Mensaje de que se ha ingresado correctamente al sistema web.

Mensaje de bienvenida: Incluye un “Bienvenido”, más el nombre del médico, en este caso, del administrador.

Fecha y hora: Muestra la fecha y hora actual.

3.1.2 Rendimiento

Principalmente dependerá del equipo y el tipo de conexión que se vaya a implementar al sistema cuando se haga la entrega.

3.1.3 Control de acceso

Cualquier persona puede entrar a la sistema web, sin embargo para poder hacer uso de sus funciones solo podrán los usuarios registrados.

3.1.4 Comerciales

El sistema web no posee comerciales.

3.2 Organizacionales

3.2.1 Patrón documentación

Se sigue un patrón de *Especificación de Requerimientos de Sistema* definido por el profesor del Tecnológico de Costa Rica, Dr. Jaime Solano Soto.

3.2.2 Patrón tecnología

Utilizamos la última versión de Ruby 2.2, y la versión de Rails 4.2.4. Además se trabajó con distintas gemas las cuales le brindan flexibilidad al programador, así como la mejora de rendimiento para la mejor respuesta y experiencia para cada usuario, y el aseguramiento la funcionalidad y completitud de la página

3.2.3 Patrón de diseño

Rails utiliza el patrón de diseño MVC, el cual está dividido en las siguientes 3 secciones:

Modelo (model)

Contiene los datos, el estado de la aplicación. No puede ser accesado por el usuario de aplicación.

Vista (view)

Esta sección es la que ve el usuario, aquí va toda la información que se desee mostrar públicamente. El modelo va enlazado a esta sección, y un modelo puede tener muchas vistas, para distintas funciones mostrar, editar, crear, eliminar o navegar.

Controlado (Controller)

Esta parte es la que recibe los eventos que genera el usuario desde las vistas, por él pasa los datos que la vista va a utilizar. Además de estar conectado con las vistas, también hace conexión con los modelos, generando así el puente para mantener el enlace del MVC.

3.2.4 Entregas

La entrega del software se hará de dos formas, una en línea y otra en físico(tangible).

En línea: Se creará una carpeta en línea, con acceso solamente al cliente.

CD: Se creará un cd con los mismos archivos que están en digital. El cd irá con una casetera con portada. Esto por seguir recomendaciones de neuromarketing.

3.2.5 Implementación

La implementación del sistema queda a cargo del usuario.

Apéndices