

CAPITULO I

**GESTIONAR EL CONTROL Y
MONITOREO DE UNA VIVIENDA,
USANDO LA TECNOLOGÍA WEB, Y EL
HARDWARE DE CÓDIGO ABIERTO
ARDUINO**

EL PROYECTO

I.1. Presentación del Proyecto

I.1.1. Título

GESTIONAR EL CONTROL Y MONITOREO DE UNA VIVIENDA, USANDO LA TECNOLOGÍA WEB, Y EL HARDWARE DE CÓDIGO ABIERTO ARDUINO.

I.1.2. Carrera

Ingeniería Informática

I.1.3. Duración del proyecto

Doce Meses

I.1.4. Área/línea de investigación priorizada

- Domótica.
- Hardware Libre Arduino.

I.2. Personal vinculado al proyecto

I.2.1. Director del proyecto

Rodas Apellido Paterno	Flores Apellido Materno	Jahzeel Issac Nombre	5811011 C.I.
2 Grupo de Taller III	Ing. Informática Carrera	Ciencias y Tecnología Facultad:	
66-31035 Teléfono	70226107 Celular	jahzeelrodas@gmail.com Correo electrónico	Firma

Tabla 1 Información personal del Director del Proyecto

I.2.2. Equipo de trabajo

Categoría	Nombres y Apellidos	Profesión	C.I.	Firma
Director	Jahzeel Issac Rodas Flores	Estudiante Ing. Informática	5811011 Tja.	
Investigador	Jahzeel Issac Rodas Flores	Estudiante Ing. Informática	5811011 Tja.	
Analista de Sistemas	Jahzeel Issac Rodas Flores	Estudiante Ing. Informática	5811011 Tja.	
Diseñador Grafico	Jahzeel Issac Rodas Flores	Estudiante Ing. Informática	5811011 Tja.	
Programador	Jahzeel Issac Rodas Flores	Estudiante Ing. Informática	5811011 Tja.	
Tutor	Marcelo Cespedes	Ingeniero Electronico		

Tabla 2 Información del equipo de trabajo

I.3.1. Resumen ejecutivo del proyecto

La domótica es un conjunto de servicios integrados en la vivienda, garantizados por una serie de sistemas que aseguran al usuario mayores ventajas en aspectos relacionados con el confort, la seguridad, las comunicaciones y la información. Se entiende por domótica el conjunto de sistemas capaces de automatizar una vivienda, y que pueden estar integrados por medio de redes interiores y exteriores de comunicación, cableadas o inalámbricas, y cuyo control goza de cierta ubicuidad, desde dentro y fuera del hogar. Se podría definir como la integración de la tecnología en el diseño inteligente de un recinto cerrado. El término domótica viene de la unión de las palabras domus (que significa casa en latín) y tica (de automática, palabra en griego, que significa “que funciona por sí sola”). La amplitud de una solución de domótica puede variar desde un único dispositivo, que realiza una sola acción, hasta amplios sistemas que controlan prácticamente todas las instalaciones dentro de la vivienda. Los distintos dispositivos de los sistemas de domótica se pueden clasificar en los siguientes grupos:

- Controlador – Los controladores son los dispositivos que gestionan el sistema según la programación y la información que reciben.
- Actuador – El actuador es un dispositivo capaz de ejecutar y/o recibir una orden del controlador y realizar una acción sobre un aparato o sistema (encendido/apagado, subida/bajada, apertura/cierre, etc.).
- Sensor – El sensor es el dispositivo que monitoriza el entorno captando información que transmite al sistema (sensores de agua, gas, humo, temperatura, viento, humedad, lluvia, iluminación, etc.).
- Bus – Es bus es el medio de transmisión que transporta la información entre los distintos dispositivos por un cableado propio, por la redes de otros sistemas (red eléctrica, red telefónica, red de datos) o de forma inalámbrica.
- Interface – Los interfaces refiere a los dispositivos (pantallas, móvil, Internet, conectores) y los formatos (binario, audio) en que se muestra la información del sistema para los usuarios (u otros sistemas) y donde los mismos pueden interactuar con el sistema.

El elevado índice de vulnerabilidad de las viviendas, tanto en el aspecto de los antisociales como de los desastres domésticos, consecuencia de la no aplicación de una vivienda con implementación Domotica, es la principal causa para el desarrollo del proyecto.

El proyecto “Gestionar el Control y Monitoreo de una Vivienda, Usando La Tecnología Web, y el hardware de código abierto ARDUINO.” se basa en realizar un sistema web que permita el control y monitoreo de viviendas usando la tecnología de web, el sistema web que se desarrolla es la interfaz entre el usuario y la vivienda domótica que se encarga de ejecutar las órdenes del usuario a los diferentes elementos de la vivienda tales como luces, puertas, persianas y otros que serán controlados automáticamente. Además el usuario podrá ver y revisar el estado de los componentes instalados en la vivienda, usando para tal motivo el principio de la DOMOTICA.

Para demostrar la funcionalidad del sistema se realizara el armado de una maqueta en la que se pueda ver el uso de los componentes que se podrá controlar y monitorear en tiempo real desde el sistema, para la realización de este componente se hará uso de hardware libre usando placas arduino, algunos sensores y otros.

Se llama hardware libre, hardware de código abierto, electrónica libre o máquinas libres a aquellos dispositivos de hardware cuyas especificaciones y diagramas esquemáticos son de acceso público, ya sea bajo algún tipo de pago, o de forma gratuita. La filosofía del software libre es aplicable a la del hardware libre, y por eso forma parte de la cultura libre. Un ejemplo de hardware libre es la arquitectura UltraSparc y Arduino cuyas especificaciones estén disponibles bajo una licencia libre.

Descripción, Fundamentación y Justificación del Proyecto

Hoy en día una vivienda es algo vivo: sistemas de climatización, juegos de luces, persianas, gas, red telefónica, seguridad e incluso sistemas informáticos alejan a las viviendas actuales del viejo concepto de lugar de refugio, hoy le pedimos más. Si el control integrado de estos sistemas en una vivienda es algo necesario, los grandes edificios con sus sistemas de cámaras, megafonía, redes informáticas etc., dan vida a las viviendas y nos obligan a desterrar el concepto estático; las tecnologías actuales, aplicadas a las viviendas, requieren un esmerado control. Hasta el momento, en un edificio todos los sistemas eran independientes, hoy la

domótica ofrece soluciones que integran y relacionan entre sí dichos elementos suponiendo una clara ventaja para el usuario.

El ritmo de vida actual ha provocado un fenómeno cultural sin precedentes, nos encontramos inmersos en la Sociedad de la Comunicación de Información, donde la domótica se convierte en una necesidad actual y vital. La rápida evolución tecnológica de la electrónica e informática, ha inundado nuestro entorno con televisores, telefotos, equipos de fax y módem, redes y sistemas informáticos tanto en oficinas como en viviendas particulares. Hasta los electrodomésticos están experimentando una vertiginosa evolución, hoy en día los fabricantes ofrecen frigoríficos inteligentes capaces de hacer telefónicamente pedidos o indicarnos que al regresar de la oficina hagamos determinadas compras. El número de documentales de televisión y prensa sobre domótica aumenta día a día.

Por esta razón es que nace la iniciativa de incorporar un sistema WEB, que permita utilizar la domótica para mejorar la calidad de vida al cliente y brindar comodidad en el control y monitoreo de vivienda. Para lo cual se planteó un propósito claro, conciso, realizable y medible el cual mencionamos a continuación: Control y Monitoreo de Viviendas online a través de un sistema web.

Se pretende desarrollar el proyecto realizando un sistema WEB, que permita al usuario final el control de componentes adaptados para el uso de la domótica y el monitoreo de una vivienda a través de la verificación del estado de los dispositivos que se hayan implementado para el uso de la domótica.

I.3.2. Objetivos

I.3.2.1. Objetivo General

Gestionar el Control y Monitoreo de una Vivienda, Usando La Tecnología Web, y el hardware de código abierto ARDUINO.

Objetivos específicos

- Desarrollo de un Sistema Web orientado a gestionar el Control y Monitoreo de una Vivienda, Usando La Tecnología Web, y el hardware de código abierto ARDUINO.

- Desarrollo de una maqueta o tablero utilizando hardware libre para demostrar la funcionalidad del sistema.

I.3.3. Metodología

Para el cumplimiento de los componentes se utilizaran las siguientes metodologías:

1. Sistema Web destinado a la Control y Monitoreo de una Vivienda.

Se empleará la metodología RUP, ya que es el más utilizado en diversos proyectos con gran aceptación. La metodología RUP cumplirá con las tres primeras fases:

- Inicio: En esta fase determinamos los requerimientos, donde se identifica los requisitos del producto según las necesidades del usuario. Se realiza la elaboración de los casos de uso del negocio, como también el calendario de actividades y un cronograma de control de fechas de inicio y entrega de cada documento.
- Elaboración: En ésta etapa se realizará el análisis y diseño del sistema WEB, se obtiene una versión preliminar del Modelo de Análisis / Diseño, también permitirá hacer una revisión general para asegurar el cumplimiento de los objetivos.
Se realizarán los diagramas UML (casos de uso, de secuencia, de estado, de clases) para modelar, especificar y visualizar la interacción entre los actores y los elementos que conforman el sistema al igual que el funcionamiento de cada uno de ellos.
- Construcción: En ésta fase se da inicio a la “Programación del Sistema” y “Pruebas del Sistema”, donde el producto es construido en base a 2 iteraciones, cada una abarcará el 50% de la programación del producto.
- En el aspecto de seguridad del sistema solo se permitirá el acceso al mismo si y solo si el usuario se encuentra registrado y el nombre de usuario y la contraseña son válidas una vez realizada la validación del usuario desde el lado del servidor como desde el lado del cliente se maneja la seguridad haciendo uso de sesiones de usuario.

Hay ocasiones en que el contenido en las bases de datos o que genera el sistema deben tener seguridad ya que la información es sensible (peligrosa en las manos equivocadas).

Un importante factor de la necesidad de las sesiones es identificar a una persona de manera virtual.

Una razón secundaria pero aun importante es la de permitir la utilización personalizada de recursos y privilegios de acceso.

2. Desarrollo de una maqueta o tablero utilizando hardware libre para demostrar la funcionalidad del sistema.

Se implementara una maqueta o tablero electrónico para demostrar el correcto funcionamiento del sistema. Para tal motivo se utilizaran placas ó tableros de prueba “ARDUINO”, básicamente lo que se pretende es utilizar una PC como servidor el cual se comunicara con las placas arduino mediante USB, la parte de programación de las placas se realizara en el IDE propio de arduino. De esta manera se cumplirá el objetivo de control y monitoreo de una vivienda, usando la tecnología web.

En el prototipo de vivienda domótica a escala 1:25 se podrá observar el funcionamiento de sensores, motores paso a paso, relés y otros. Con los que se podrá controlar puertas, ventanas, persianas y otros.

I.3.4. Resultados esperados

Sistema WEB que “Optimiza el Control y Monitoreo de Una Vivienda”. Al término del proyecto se ha desarrollado un sistema WEB totalmente funcional. Capaz de manejar sesiones de usuarios, Abm’s de usuario, dispositivos, perfiles, etc. El sistema desarrollado guardara un registro de los cambios de estado de los componentes electrónicos y generara un reporte a petición del usuario.

2. Armado de una maqueta o tablero utilizando hardware libre para demostrar la funcionalidad del sistema. Al término del proyecto se ha desarrollado un prototipo a escala 1:25 de una vivienda con implementación de DOMOTICA. Utilizando una placa Arduino Mega 2560 REV3, Una Shield Ethernet, sensores como el DTH11 que mide temperatura y

humedad, sensor de luminosidad, sensores infrarrojos en puertas y ventanas, sensor de llamas, Etc.

I.3.5. Transferencia de resultados

I.3.5.1. Grupo de beneficiarios de los resultados

Los habitantes de la ciudad de Tarija provincia-cercado se beneficiara con la implementación de las casas inteligentes, tanto en la parte de seguridad reduciendo el perpetrado de las viviendas. Como así también en la parte de monitoreo de instalaciones de sistema cerrado, esto es de gran ayuda en el caso de tener personas que necesiten ser monitoreadas constantemente y a cualquier hora del día, por diferentes motivos.

I.3.6. Cronograma de Actividades

Tarea	Comienzo	Duración	Fin
Determinación de Requerimientos	01-abr	20	21-abr
Análisis y Diseño del Sistema	21-abr	60	20-jun
Programación Del Sistema	20-jun	90	18-sep
Pruebas Del Sistema	18-sep	20	08-oct
Determinación de Requerimientos de Hardware	08-oct	10	18-oct
Análisis y Diseño de los Elementos Hardware que Serán Instalados Para el Tablero	18-oct	30	17-nov
Programación De Las Placas Arruino	17-nov	15	02-dic
Pruebas de Hardware	02-dic	10	12-dic

Tabla 4 Cronograma de actividades del proyecto

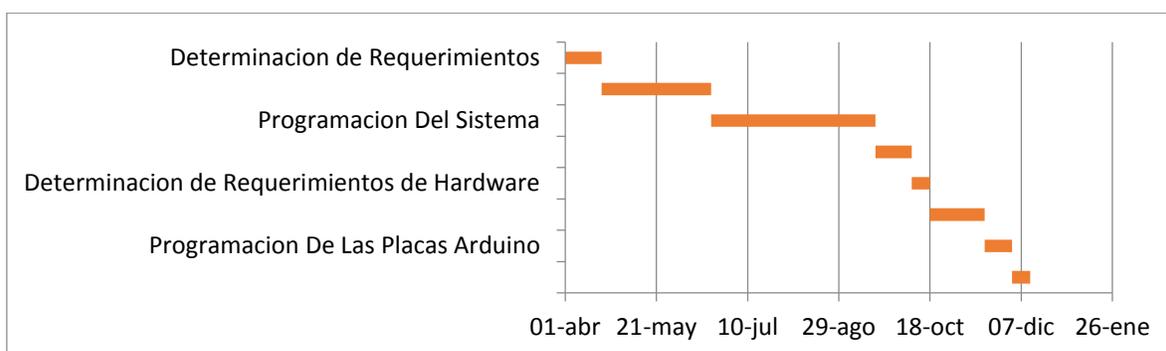


Figura 1 Cronograma de actividades del proyecto

I.3.7. Árbol de problemas

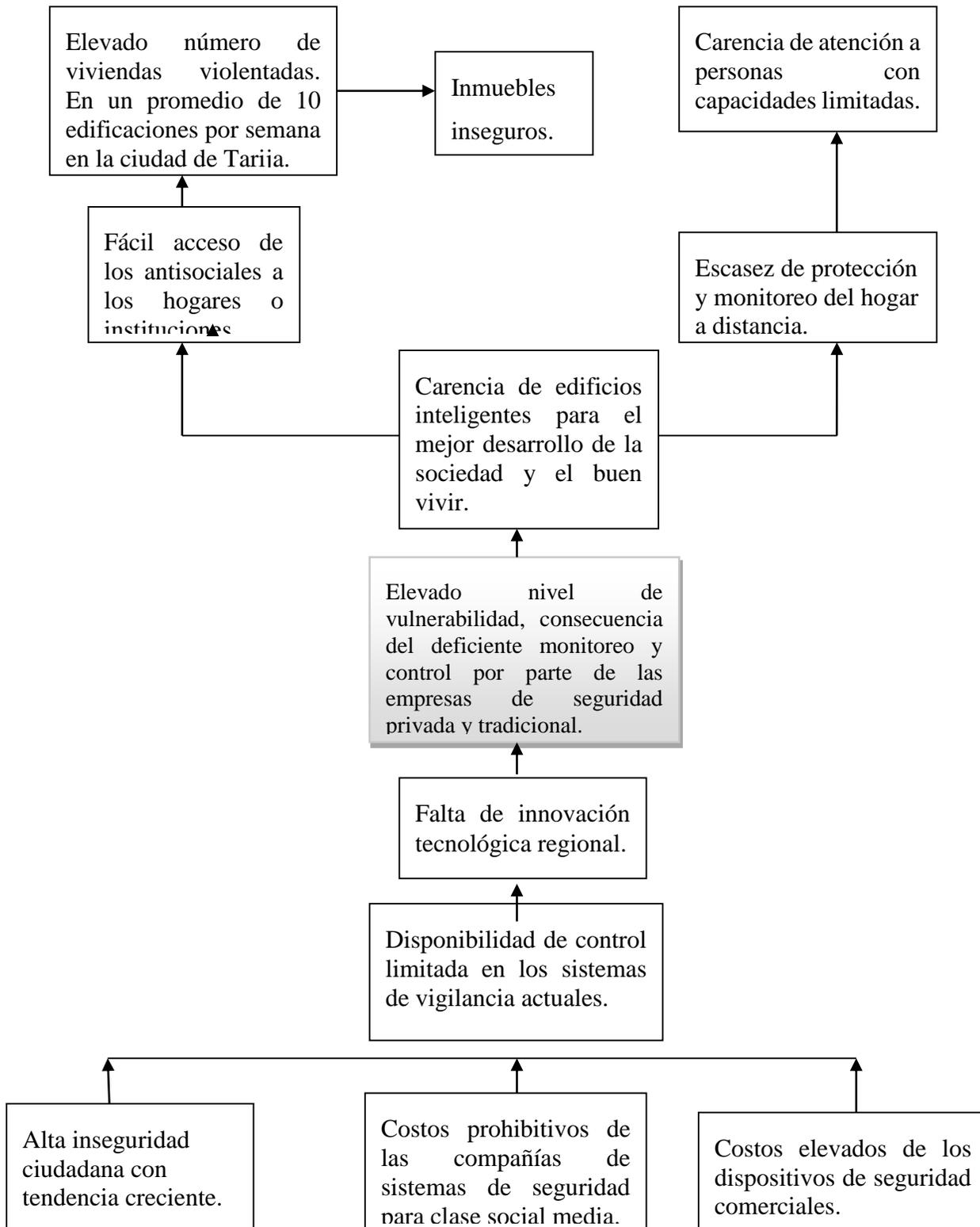


Figura 2 Árbol de problemas

I.3.8. Árbol de Objetivos

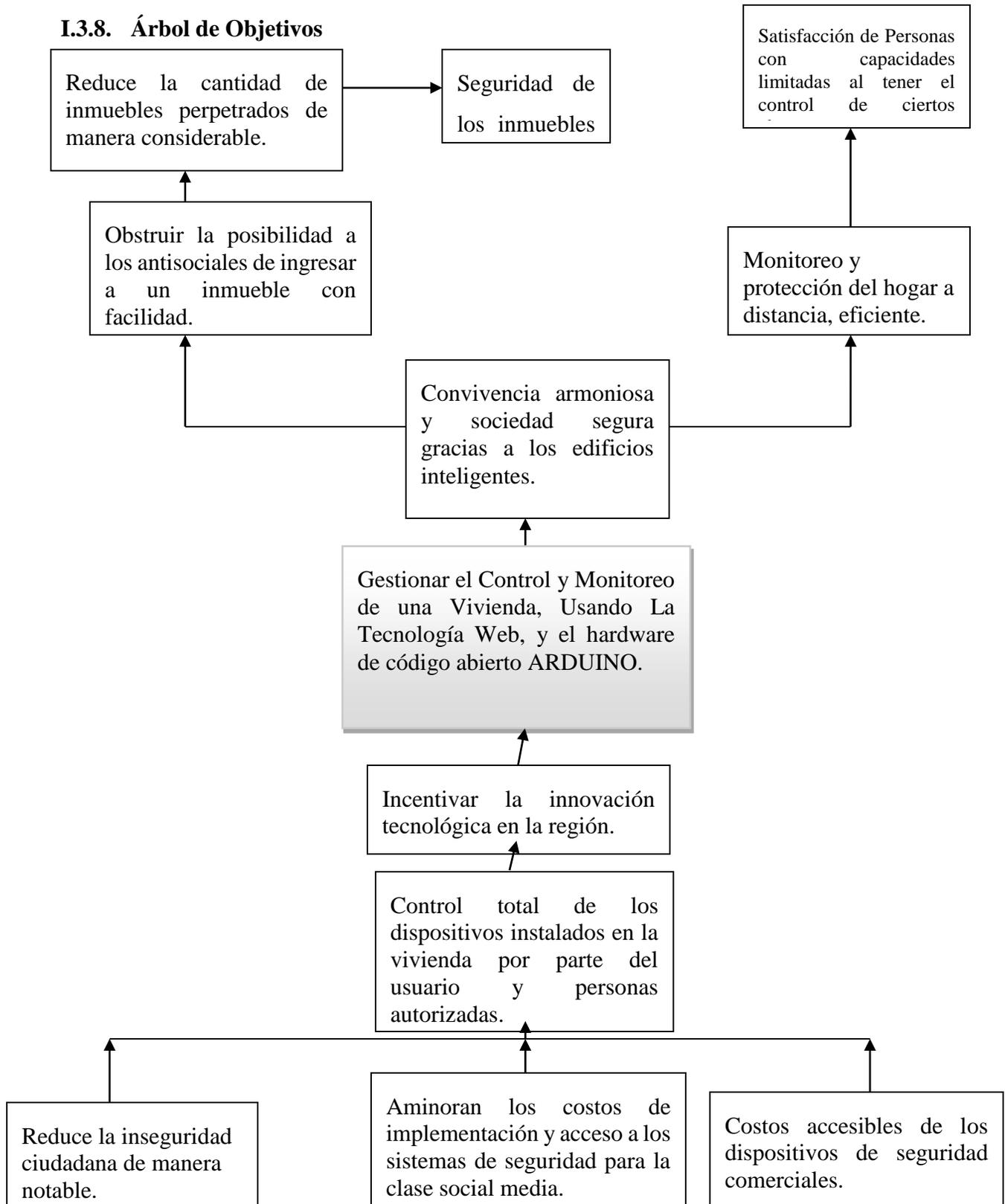


Figura 3 Árbol de objetivos

I.3.9. Matriz de Marco Lógico

Resumen Narrativo del Proyecto	Indicadores	Medios de Verificación	Supuestos
<p>Fin</p> <p>Contribuir y ayudar a la sociedad de la ciudad de Tarija provincia cercado en la protección de sus hogares, tanto en el control de los artículos electrónicos, como en el monitoreo de su vivienda esto último con el fin de ver qué sucede dentro de la vivienda en cualquier momento. Brindándoles la oportunidad de conocer los beneficios que tiene la aplicación de las casas inteligentes (Domotica).</p>	<ul style="list-style-type: none"> • A partir de la gestión 2015 la sociedad de Tarija y la provincia cercado en general se beneficiara en el área de seguridad social, puesto que la implementación de una casa inteligente será accesible por el 60% de personas que conviven en sociedad. • Los usuarios de las casas inteligentes se incrementaran a partir de la gestión 2016 en no menos del 10% en la provincia cercado de la ciudad de tarija. 	<ul style="list-style-type: none"> • Informe que refleje el aumento de la demanda, por parte de la empresa que brindara y se encargara del desarrollo e implementación de las casas inteligentes. • Datos estadísticos de encuestas realizadas a los usuarios. 	<ul style="list-style-type: none"> • Se cuenta con el hardware necesario para el desarrollo e implementación de una casa inteligente. • Las autoridades de la UAJMS aprueban el proyecto.

<p>Objetivo General (Propósito)</p> <p>Gestionar el Control y Monitoreo de una Vivienda, Usando La Tecnología Web, y el hardware de código abierto ARDUINO.</p>	<ul style="list-style-type: none"> • Al finalizar el proyecto el 70% de los acceso a la vivienda puertas ventanas y otros podrán ser controlados y monitoreados a través del sistema web. • Al finalizar el proyecto en diciembre de 2014, el 60% de los potenciales usuarios a los que está destinado el proyecto, demuestran interés por el mismo. 	<ul style="list-style-type: none"> • En la maqueta existen 7 accesos a la vivienda de los cuales solo 4 se están controlando desde el sistema web. $NAC = \frac{NACA}{NTAI}$ $NAC = \frac{4}{7} = 0,7142$ $0,7142 * 100 = 71,42\%$ <p>NAC= Numero de accesos controlados. NACA= Numero de accesos controlados con arduino. NTAI= Número total de accesos en el inmueble.</p>	<ul style="list-style-type: none"> • El personal involucrado apoya de manera constante y voluntaria el desarrollo del proyecto.
-----------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------

Objetivos Específicos (Componentes)			
<p>1. Sistema Web destinado a gestionar el control y monitoreo de una vivienda usando la tecnología web y el hardware de código abierto arduino.</p> <p>2. Desarrollo de una maqueta utilizando hardware libre para demostrar la funcionalidad del sistema.</p>	<ul style="list-style-type: none"> • Al finalizar el proyecto en diciembre de 2014 se concluye el desarrollo del “Sistema Web Destinado al Control y Monitoreo de una vivienda”. • Al finalizar el proyecto en diciembre de 2014 se concluye con el desarrollo de una maqueta o tablero utilizando hardware libre para demostrar la funcionalidad del sistema. Lo cual se hace evidente. 	<ul style="list-style-type: none"> • Certificado del cumplimiento del sistema firmado por el tutor del proyecto. • Certificado de culminación y funcionamiento de una maqueta a escala 1:25 con sensores, controlador y actuadores firmado por el tutor del proyecto. 	<ul style="list-style-type: none"> • La situación estable de la UAJMS permite el normal desarrollo del proyecto. • El equipo cuenta con las herramientas necesarias para la construcción del sistema y del componente hardware.

<p>Actividades</p> <p>1. Sistema web destinado a gestionar el control y monitoreo de una vivienda usando la tecnología web y el hardware de código abierto arduino.</p> <p>1.1. Determinación de requerimientos.</p> <p>1.2. Análisis y Diseño del Sistema.</p> <p>1.3. Programación del sistema.</p> <p>1.4. Pruebas del sistema.</p> <p>2. Desarrollo de una maqueta o tablero utilizando hardware libre para demostrar la funcionalidad del sistema.</p> <p>2.1. Determinación de requerimientos de hardware.</p> <p>2.2. Análisis y diseños de los elementos hardware que serán instalados para el desarrollo de la maqueta o tablero.</p>	<table border="1"> <thead> <tr> <th>ITEM</th> <th>RUBROS</th> <th>Aporte Universidad</th> <th>Otro Aporte</th> <th>TOTAL (Bs.)</th> </tr> </thead> <tbody> <tr> <td>20000</td> <td>SERVICIOS NO PERSONALES</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td>21000. Servicios Básicos</td> <td></td> <td></td> <td>420</td> </tr> <tr> <td></td> <td>22000. Servicios de transporte</td> <td></td> <td></td> <td>100</td> </tr> <tr> <td></td> <td>24000. Mantenimiento y reparación</td> <td></td> <td></td> <td>1000</td> </tr> <tr> <td></td> <td>25000. Servicios Profesionales y Comerciales</td> <td></td> <td></td> <td>12600</td> </tr> <tr> <td></td> <td>Sub total rubro</td> <td></td> <td></td> <td>14120</td> </tr> <tr> <td>30000</td> <td>MATERIALES Y SUMINISTROS</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td>31000. Alimentos y Productos Forestales</td> <td></td> <td></td> <td>80</td> </tr> <tr> <td></td> <td>32000. Productos de Papel, Cartón e Impresos</td> <td></td> <td></td> <td>180</td> </tr> <tr> <td></td> <td>39000. Productos Varios.</td> <td></td> <td></td> <td>50</td> </tr> <tr> <td></td> <td>Sub total rubro</td> <td></td> <td></td> <td>310</td> </tr> <tr> <td></td> <td>TOTAL</td> <td></td> <td></td> <td>13430</td> </tr> </tbody> </table>	ITEM	RUBROS	Aporte Universidad	Otro Aporte	TOTAL (Bs.)	20000	SERVICIOS NO PERSONALES					21000. Servicios Básicos			420		22000. Servicios de transporte			100		24000. Mantenimiento y reparación			1000		25000. Servicios Profesionales y Comerciales			12600		Sub total rubro			14120	30000	MATERIALES Y SUMINISTROS					31000. Alimentos y Productos Forestales			80		32000. Productos de Papel, Cartón e Impresos			180		39000. Productos Varios.			50		Sub total rubro			310		TOTAL			13430	<p>Informe del desembolso por parte del director del proyecto</p> <p>Documentación del análisis y diseño del sistema</p> <p>Utilización de las normas de modelado a los diagramas de análisis diseño del sistema, mediante la metodología de desarrollo de sistemas RUP</p>	<p>Los desembolsos se realizan de acuerdo al cronograma.</p> <p>Acceso a internet de manera irrestricta</p>
ITEM	RUBROS	Aporte Universidad	Otro Aporte	TOTAL (Bs.)																																																																
20000	SERVICIOS NO PERSONALES																																																																			
	21000. Servicios Básicos			420																																																																
	22000. Servicios de transporte			100																																																																
	24000. Mantenimiento y reparación			1000																																																																
	25000. Servicios Profesionales y Comerciales			12600																																																																
	Sub total rubro			14120																																																																
30000	MATERIALES Y SUMINISTROS																																																																			
	31000. Alimentos y Productos Forestales			80																																																																
	32000. Productos de Papel, Cartón e Impresos			180																																																																
	39000. Productos Varios.			50																																																																
	Sub total rubro			310																																																																
	TOTAL			13430																																																																

<p>2.3. Programación de las placas arduino.</p> <p>2.4. Armado y configurado del servidor.</p> <p>2.5. Pruebas de hardware.</p>	<p>Adquisición de las placas arduino y componentes con un costo.</p> <p>TOTAL: 2000</p>	<p>Sketch de arduino que se cargara en la placa.</p> <p>Configuración de router D-LINK DIR 400</p>	
---------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------	--

Tabla 5 Matriz de marco lógico del proyecto

I.4. Presupuesto / Justificación

ITEM	RUBROS	Aporte Universidad	Otro Aporte	TOTAL (Bs.)
20000	SERVICIOS NO PERSONALES			
	21000. Servicios Básicos			420
	22000. Servicios de transporte			100
	24000. Mantenimiento y reparación			1000
	25000. Servicios Profesionales y Comerciales			12600
	Sub total rubro			14120
30000	MATERIALES Y SUMINISTROS			
	31000. Alimentos y Productos Forestales			80
	32000. Productos de Papel, Cartón e Impresos			180
	39000. Productos Varios.			50
	Sub total rubro			310
	TOTAL			13430

Tabla 6 Resumen del presupuesto del proyecto

1) GRUPO 10000. SERVICIOS PERSONALES

a) SUB GRUPO 12000. Empleados no Permanentes

Partida	Personal	Remuneración	Tiempo/meses	Total
12100	Personal Eventual			
Total				

Tabla 7 Empleados no permanentes

2) GRUPO 20000. SERVICIOS NO PERSONALES

b) SUB GRUPO 21000. Descripción de los gastos de servicios básicos

Partida	Tipo de servicio básico *	Costo	Tiempo mes	Costo Total
21100	Comunicación	20	3	60
21200	Energía Eléctrica	30	8	240
21300	Agua	30	8	240
21400	Servicios Telefónicos	30	8	240
Total				780

Tabla 8 Servicios no personales

c) SUB GRUPO 22000. Descripción de los gastos de viajes y transporte de personal

Partida	Personal	Lugar	N° de viajes	Costo unitario*	Costo total
Total					0

Tabla 9 Descripción de viajes y transporte personal

d) SUB GRUPO 23000. Descripción de los gastos por concepto de alquileres de equipos y maquinarias

Partida	Alquiler de equipo y maquinaria	Costo unitario	Tiempo mes	Costo total
23100	Alquiler de Edificios	4000	12	48000
Total				48000

Tabla 10 Gastos de alquileres de equipo y maquinaria

e) SUB GRUPO 24000. Descripción mantenimiento y reparación

Partida	Mantenimiento y reparación de equipo y maquinaria	Costo unitario	Tiempo mes	Costo total
24300	Otros Gastos por Mantenimiento y Reparación	400	6	1200
Total				1200

Tabla 11 Gastos de mantenimiento y reparación

f) SUB GRUPO 25000. Descripción de los gastos en servicios profesionales y comerciales

Partida	Tipo de servicio profesional y comercial *	Cantidad	Costo unitario	Tiempo mes	Costo total
25200	Estudios e Investigaciones	5	1000	6	30000
25500	Publicidad	10	1000	12	120000
Total					

Tabla 12 Gastos de servicios profesionales y comerciales

3) GRUPO 30000. MATERIALES Y SUMINISTROS

g) SUB GRUPO 31000. Descripción de los gastos Alimentos y Productos Agroforestales

Partida	Tipo de material *	Cantidad	Costo/Unitario	Total
31110	Refrigerios y Gastos Administrativos	10	8	80
Total				80

Tabla 13 Gastos en alimentos y productos agroforestales

h) SUB GRUPO 32000. Descripción del gasto de Productos de Papel, Cartón e Impresos

Partida	Tipo de material *	Cantidad	Costo/Unitario	Total
32100	Papel de Escritorio	20	30	600
Total				600

Tabla 14 Gastos en papel, cartón e impresos

i) SUB GRUPO 33000. Descripción del gasto en textiles y vestuario

Partida	Productos textiles y vestuarios	Cantidad	Costo/Unitario	Total
Total				0

Tabla 15 Gastos en textiles y vestuario

j) SUB GRUPO 34000. Combustibles, Productos Químicos, Farmacéuticos y Otros

Partida	Combustibles, Productos Químicos, Farmacéuticos y Otros	Cantidad	Costo/Unitario	Total
Total				0

Tabla 16 Gastos de productos quimicos

k) SUB GRUPO 39000. Descripción del gasto en productos varios

Partida	Productos de cuero y caucho	Cantidad	Costo/Unitario	Total
39500	Útiles de Escritorio y de Oficina			50
Total				50

Tabla 17 Gastos de productos varios

4) GRUPO 40000. ACTIVOS REALES

l) SUB GRUPO 43000. Descripción del gasto de Maquinaria y Equipo

Partida	Tipos de productos	Cantidad	Costo/Unitario	Total
				0
Total				0

Tabla 18 Gastos de maquinaria y equipo

m) SUB GRUPO 46000. Descripción de estudios y proyectos para inversión

Partida	Productos textiles y vestuarios	Cantidad	Costo/Unitario	Total
				0
Total				0

Tabla 19 Gastos de estudios para la investigación

n) SUB GRUPO 49000. Descripción del gasto de Otros Activos

Partida	Tipos de productos *	Cantidad	Costo/Unitario	Total
Total				

Tabla 20 Gastos de otros servicios

I.5. Curriculum Vitae

I.5.1. Antecedentes personales

RODAS	FLORES	JAHZEEL ISSAC	5811011
Apellido Paterno	Apellido Materno	Nombre	C.I.
11/04/1987	MASCULINO	BARRIO LAS PANOSAS/ CALLE MENDES 0215	
Fecha de nacimiento	Sexo	Dirección	
		ESQUINA AVAROA	

TARIJA	6631035	70226107	jahzeelrodas@hotmail.com
Ciudad	Teléfono Domicilio	Celular	Correo electrónico

Tabla 21 Información personal

I.5.2. Antecedentes académicos

Título obtenido	Universidad	País	Año

Tabla 22 Antecedentes académicos

I.5.3. Participación en proyectos de investigación

Título proyecto	Institución	Cargo	Año

Tabla 23 Proyectos en los que se participo

I.5.4. Publicaciones realizadas (libros, revistas, compendios y otros)

Autor	Tipo de publicación, Año, título, volumen, páginas, editorial

Tabla 24 Publicaciones realizadas

CAPITULO II

COMPONENTES

COMPONENTE 1:

Desarrollo de un Sistema Web orientado a gestionar el Control y Monitoreo de una Vivienda, Usando La Tecnología Web, y el hardware de código abierto ARDUINO.

PLAN DE DESARROLLO DE SOFTWARE

II. CAPITULO II: COMPONENTES

II.1. GESTIONAR EL CONTROL Y MONITOREO DE UNA VIVIENDA, USANDO LA TECNOLOGÍA WEB, Y EL HARDWARE DE CÓDIGO ABIERTO ARDUINO.

II.1.1. Plan de Desarrollo de Software

II.1.1.1. Introducción

El presente documento es un plan de desarrollo del software que será incluida en la propuesta elaborada como respuesta al proyecto de la asignatura de Taller III de la Carrera de Ingeniería Informática de la Facultad de Ciencias y Tecnología de la Universidad Autónoma Juan Misael Saracho. Este documento sentará las bases para el desarrollo del proyecto y ofreciendo una visión global del enfoque de desarrollo propuesto.

El proyecto será desarrollado por el universitario Jahzeel Issac Rodas Flores, basado en la metodología RUP (Rational Unified Process), cumpliendo las 4 fases que contiene esta y haciendo uso de la terminología en este documento.

El enfoque de desarrollo propuesto constituye una configuración del proceso RUP de acuerdo a las características del proyecto, seleccionando los roles de los participantes, las actividades a realizar y los artefactos o entregables, que serán generados. Este documento es a su vez uno de los artefactos RUP.

II.1.1.1.1. Propósito

El propósito del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto, En él se describe el enfoque de desarrollo del software.

Los usuarios del plan de Desarrollo del software son:

- El jefe del proyecto, que lo utiliza para organizar la agenda y necesidades de recursos, y realizar su seguimiento.
- Los Docentes de la materia de Taller III, para evaluar el cumplimiento del proyecto.

II.1.1.1.2. Alcance

Con el plan de Desarrollo del Software se pretende “**GESTIONAR EL CONTROL Y MONITOREO DE UNA VIVIENDA, USANDO LA TECNOLOGÍA WEB, Y EL**

HARDWARE DE CÓDIGO ABIERTO ARDUINO”, abarcando las cuatro fases de la metodología RUP.

Para ello se elaborará un cronograma con cada una de las actividades a realizar y se especificará los detalles de construcción del proyecto para los distintos roles que cumplen los desarrolladores.

II.1.1.1.3. Resumen

Después de esta introducción, el resto del documento está organizado en las siguientes secciones:

Vista General del Proyecto.- Proporciona una descripción del propósito, alcance y objetivos del proyecto, estableciendo los artefactos que serán producidos y utilizados durante el proyecto.

Organización del Proyecto.- Describe la estructura organizacional del equipo de desarrollo.

Gestión del Proceso.- Explica los costos y planificación estimada, define las fases e hitos del proyecto y describe como se realizará su seguimiento.

Planes y Guías de aplicación.- Proporciona una lista global del proceso de desarrollo de software, incluyendo métodos, herramientas y técnicas que serán utilizadas.

II.1.1.2. Vista General del Proyecto

II.1.1.2.1. Propósito, Alcance y Objetivos

Este documento ha sido redactado con el fin de marcar las pautas generales y las especificaciones que deberá seguir el sistema a desarrollar (DOMUINO).

El presente servirá de canal de comunicación entre las distintas partes implicadas en el desarrollo del sistema y deberán tomar parte en su confección los miembros de cada una de estas partes.

Se ha constatado la necesidad de un sistema informático que automatice las tareas de control y monitoreo de una vivienda haciendo de uso de la web y el hardware de código abierto ARDUINO.

El producto que vamos a describir es un sistema que será capaz de monitorear y controlar un inmueble a distancia usando la tecnología web y el principio de domótica.

Este sistema de control y monitoreo de un inmueble haciendo uso de la tecnología WEB “DOMUINO” de ahora en adelante, debe ser capaz de adaptarse a cualquier vivienda, o sea, según el número de pisos, puertas, ventanas y otros, el sistema deberá funcionar de igual forma y con todas sus prestaciones.

DOMUINO deberá permitir al usuario programar eventos en fecha y hora determinadas, además deberá ser capaz de informarnos del estado global de la vivienda o de algún dispositivo específico en cualquier instante de tiempo, advirtiéndonos de cualquier evento que haya ocurrido en el día, generara reportes por hora, día, mes o cuando el usuario lo requiera de cada uno de los componentes o un reporte general del inmueble, por otra parte DOMUINO maneja sesiones para controlar los privilegios de cada uno de los usuarios que vayan a utilizar el sistema, DOMUINO será completamente configurable por el usuario con privilegios superiores una vez que se le haya asignado a este un usuario y contraseña.

El mismo podrá tener acceso a:

- Registro de usuarios.
- Registro de perfiles personalizados por el usuario.
- Registro de los puntos de control de la vivienda.
- Registro de los datos del día.
- Consulta de la situación de la vivienda en tiempo real.
- Consulta del estado de los dispositivos instalados en tiempo real.
- Manejo de los diferentes dispositivos instalados en la vivienda.
- Gestión de la temperatura ambiente.

II.1.1.2.2. Suposiciones y Restricciones

Acontecimientos que deben ocurrir para que el proyecto se ejecute con éxito, pero que están fuera del ámbito de control por parte del proyectista.

Suposiciones.- DOMUINO deberá funcionar perfectamente en cualquier plataforma y navegador. En el documento se expresan los requisitos en términos de lo que el sistema debe

proporcionar a los usuarios que acceden a él. Se asume que los requisitos descritos en este documento son estables una vez que sea aprobado por la UAJMS Facultad de Ciencias y Tecnología carrera de Ingeniería Informática.

Restricciones.- DOMUINO se desarrollara usando un lenguaje de programación orientado a objetos: JAVA, y un motor de base de datos denominado PostgreSQL para la base de datos, en el inmueble se instalaran sensores y componentes electrónicos que interactuaran con DOMUINO a través de placas ARDUINO específicamente el modelo MEGA 2560 Rev3 el cual se conectara a internet mediante una tarjeta de red llamada “Arduino Ethernet Shield”, siempre será preferible utilizar software y hardware libre. DOMUINO deberá proporcionar exclusivamente una interfaz WEB, lo cual facilitara su utilización en múltiples plataforma a menor coste. Se considerara adicionalmente que DOMUINO utiliza el protocolo TCP/IP. La disponibilidad de DOMUINO deberá ser absoluta, excepto en el caso de cortes en la red de comunicaciones o el tendido eléctrico donde se encuentre el servidor.

Entregable del Proyecto

A continuación se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituye la configuración de RUP desde la perspectiva de artefactos, y que proponemos para este proyecto.

Es preciso destacar que de acuerdo a la filosofía de RUP (y todo proceso iterativo e incremental), y todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, solo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos, Sin embargo, el resultado de cada iteración y los hilos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de los artefactos. Esto será indicado más adelante cuando se presenten los objetivos de cada iteración.

1) Plan de Desarrollo de Software

Es el presente documento.

2) **Modelo de Casos de Uso del Negocio**

Es un modelo de las funciones de negocio vistas desde la perspectiva de los actores externos (Agentes de registro, solicitantes finales, otros sistemas, etc.) que permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito. Este modelo se presenta con un Diagrama de Casos de Uso empleando estereotipos específicos para este modelo.

3) **Modelo de Objetos del Negocio**

Es un modelo que describe la realización de cada caso de uso del negocio, estableciendo los actores internos, la información que en términos generales manipulan y los flujos de trabajo (workflows) asociados al caso de uso del negocio. Para la representación de este modelo se utilizan Diagramas de Colaboración para mostrar actores externos, internos y las entidades que manipulan, un Diagrama de Clases para mostrar gráficamente las entidades del sistema y sus relaciones, y Diagramas de Actividad para mostrar los flujos de trabajo.

4) **Glosario**

Es un documento que define los principales términos usados en el proyecto. Permite establecer una terminología consensuada.

5) **Modelo de Casos de Uso**

El modelo de Casos de Uso presenta las funciones del sistema y los actores que lo utilizan. Se representan mediante Diagramas de Casos de Uso.

6) **Especificaciones de Casos de Uso**

Para los casos de uso que lo requieran (cuya funcionalidad no se evidente o que no baste con una simple descripción narrativa), se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen; precondiciones, pos condiciones, flujo de eventos, requisitos no funcionales asociados. También, para casos de uso cuyo flujo de eventos sea complejo podrá adjuntarse una representación gráfica mediante un Diagrama de Actividad.

7) Visión

Este documento define la visión del producto desde la perspectiva del Cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema.

8) Especificaciones Adicionales

Este documento capturará todos los requisitos que no han sido incluidos como parte de los casos de uso y se refieren requisitos no-funcionales globales. Dichos requisitos incluyen: requisitos legales o normas, aplicación de estándares, requisitos de calidad del producto, tales como: confiabilidad, desempeño, etc., u otros requisitos de ambiente, tales como: sistema operativo, requisitos de compatibilidad, etc.

9) Diseño de Interfaces de Usuario

Se trata de prototipos que permiten al usuario hacerse una idea más o menos precisa de las interfaces que proveerá el sistema y así, conseguir retroalimentación de su parte respecto a los requisitos del sistema. Estos prototipos se realizarán como: dibujos a mano en papel, dibujos con alguna herramienta gráfica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo al avance del proyecto. Solo los de este último tipo serán entregados al final de la fase de elaboración, los otros serán desechados. Asimismo, este artefacto, será desechado en la fase de construcción en la medida que el resultados de las iteración vayan desarrollando el producto final.

10) Modelo de Análisis y Diseño

Este modelo establece la realización de los casos de uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación) hacia una de diseño (incluyendo una orientación hacia el entorno de implementación), de acuerdo al avance del proyecto.

Modelo de Datos

Previendo que la persistencia de la información del sistema será soportada por una base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este modelo se utiliza un Diagrama de Clases (donde se utiliza un profile UML para Modelado de Datos, para conseguir la representación de tablas, claves, etc.).

11) Modelo de Implementación

Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema. (Este modelo es sólo una versión preliminar al final de la fase de Elaboración, posteriormente tiene bastante refinamiento).

12) Modelo de Despliegue

Este modelo muestra el despliegue la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes.

13) Casos de Prueba

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados. Estos casos de prueba son aplicados como pruebas de regresión en cada iteración. Cada caso de prueba llevará asociado un procedimiento de prueba con las instrucciones para realizar la prueba, y dependiendo del tipo de prueba dicho procedimiento podrá ser automatizable mediante un script de prueba.

14) Lista de Riesgos

Este documento incluye una lista de los riesgos conocidos y vigentes en el proyecto, ordenados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación.

15) Material de Apoyo al Usuario Final

Corresponde a un conjunto de documentos y facilidades de uso del sistema, incluyendo; Guías de usuario y otros necesarios.

16) Producto

Los ficheros del producto serán empaquetados y almacenados en un CD con las librerías necesarias para el uso, diseño y modificación por parte de diseñadores y programadores para futuras mejoras del sistema. El producto, a partir de la primera iteración de la fase de construcción, es desarrollado incremental e iterativamente, obteniéndose una nueva release al final de cada iteración.

II.1.1.2.3. Evolución del Plan de Desarrollo del Software

El Plan de Desarrollo de Software se revisará semanalmente y se refinará antes del comienzo de cada iteración.

II.1.1.3. Organización del Proyecto.

II.1.1.3.1. Participante en el Proyecto

Director del Proyecto. Univ. Jahzeel Issac Rodas Flores

II.1.1.3.2. Roles y Responsabilidades

A continuación se describen las principales responsabilidades durante las fases de inicio y elaboración, de acuerdo con los roles que desempeñan en RUP.

Responsable	Actividades
Director: Jahzeel Issac Rodas Flores	Planificar y controlar el cronograma del proyecto. Asignar y gestionar recursos y prioridades a los distintos componentes y actividades del proyecto. Coordinar las iteraciones con los usuarios del proyecto. Establecer un conjunto de prácticas que aseguren la calidad e integridad del proyecto. Supervisar el desarrollo del proyecto.
Analista y Diseñador	Capturar la especificación y validación de requisitos interactuando con los usuarios. Elaborar el análisis y diseño del sistema. Elaborar las pruebas funcionales del sistema.
Programadores	Realizar la programación del sistema. Elaborar las pruebas funcionales del sistema.

Tabla 25 Detalle de responsabilidades del personal del proyecto

II.1.1.4. Gestión del Proceso

II.1.1.4.1. Estimaciones del Proyecto

El presupuesto del proyecto y los recursos involucrados se adjuntan en un documento por separado.

II.1.1.4.2. Plan del Proyecto

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

II.1.1.4.2.1. Plan de las Fases

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una distribución de tiempo y el número de iteraciones de cada fase de forma preliminar.

Fase	Nro. Iteraciones	Duración (días)
Fase de Inicio	1	40
Fase de Elaboración	2	72
Fase de Construcción	3	115
Fase de Transición	1	39

Tabla 26 Cuadro de iteraciones de las fases RUP del proyecto

Los hitos que marcan el final de cada fase se describen en la siguiente tabla.

Descripción	Hito
Fase de Inicio	En esta fase desarrollarán los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto visión. Los principales casos de uso serán identificados y se harán un refinamiento del Plan de Desarrollo del Proyecto del proyecto. La aceptación del Cliente/Usuario del artefacto visión y el Plan de Desarrollo marcan el final de esta fase.

Fase de Elaboración	<p>En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y/o críticas del sistema). Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera etapa de la fase de construcción deben estar analizados y diseñados (en el modelo de análisis/diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase. En nuestro caso particular, por no incluirse las fases siguientes, la revisión y entrega de todos los artefactos hasta este punto de desarrollo también como hito. La primera y única iteración tendrá como objetivo la identificación y especificación de los principales casos de uso, así como su realización preliminar en el Modelo de Análisis/ Diseño, también permitirá hacer una revisión general del estado de los artefactos hasta este punto y ajustar si es necesario la planificación para asegurar el cumplimiento de los objetivos.</p>
Fase de Construcción	<p>Durante la fase de construcción se termina de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis/Diseño. El producto se construye en base a 1 iteración, en la cual se produce una etapa a la cual se le aplican las pruebas y se valida con el cliente/usuario. Se comienza la elaboración de material de apoyo al usuario.</p>
Fase de Transición	<p>En esta fase se prepararán para distribución, asegurando una implantación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito</p>

	que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto.
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla 27 Cuadro de hitos del fin de fases del proyecto

II.1.1.4.2.2. Calendario del Proyecto

El calendario del proyecto es supervisado por el director del proyecto donde el mismo controla el avance según las fechas de presentación establecidas.

II.1.1.4.3. Seguimiento y Control del Proyecto

II.1.1.4.3.1. Gestión de Requisitos

De acuerdo al tiempo de vida de la realización del proyecto se irá analizando los siguientes requisitos.

Requisitos Organizacionales.-

El personal debe seguir como estándar la metodología RUP bajo un modelado con el lenguaje UML.

Se debe contar con personal que conozca sobre la plataforma java, programación, manejo del gestor de datos Postgres, generación de reportes, manejo de componentes java, conocimiento medio-avanzado de electrónica, manejo y programación de ARDUINO.

Requisitos de Personal y Usuarios.-

El usuario debe tener interés de participación en el desarrollo del sistema.

Los usuarios deben contar con un nombre de usuario y clave para poder realizar aportes de información.

Requisitos Físicos y de Funcionamiento.-

Para el funcionamiento se debe contar con un equipo que soporte la base de datos y que tenga instalado la máquina virtual java. El usuario debe contar con un equipo de computación capaz de poder manipular información vía web con conexión a internet.

II.1.1.4.3.2. Control de Plazos

Fase	Nro. de Iteraciones	Inicio
Inicio	Primera	
Elaboración	Primera	
Construcción	Primera	
Transición	Primera	

Tabla 28 Control de plazos

II.1.1.4.3.3. Control de Calidad

Nuestro sistema de control de calidad se basa en los puntos siguientes:

- Detectar problemas.
- Delimitar el área problemática.
- Estimar factores que probablemente provoquen el problema.
- Determinar si el efecto tomado como problema es verdadero o no.
- Confirmar los efectos de mejora.
- Detectar desfases.
- Realizar pruebas en cada versión.

Objetivos del control de calidad

- Aumentar la satisfacción del cliente.
- Equilibrar el esfuerzo en múltiples demandas.
- Obtener el mejor producto.
- Disfrutar de una ventaja competitiva.
- Disponer de métricas objetivas de valoración.
- Ahorrar tiempo y dinero.

II.1.1.4.3.4. Gestión de Riesgos

II.1.1.4.3.4.1. Introducción

Este documento incluye una lista de los riesgos conocidos y vigentes en el proyecto, ordenados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación.

II.1.1.4.3.4.2. Alcance

Este plan es emprendido ante la fase de elaboración para asegurar que ninguno de los riesgos identificados sea direccionado durante la misma fase de elaboración. Apenas el plan de gestión de riesgos haya sido documentado, el proceso de prevención de riesgos estará ocupado para monitorear y controlar la probabilidad y el impacto de los riesgos sobre el proyecto.

II.1.1.4.3.4.3. Proceso de Manipulación de Riesgos

En esta sección se establecen todos los aspectos que son necesarios para que cada riesgo que se pueda identificar dentro del proyecto en desarrollo sea debidamente documentado, evaluado y mitigado tanto como sea necesario. Para ello se establecen los procedimientos que en este proyecto deben ser ejecutados al momento de identificar un determinado riesgo, el mismo está acorde a las políticas de la organización.

II.1.1.4.3.4.4. Roles Encargados de Gestionar los Riesgos

Se establecen los roles participantes para el manejo de riesgos en su proyecto, los cuales van a ser los encargados de gestionar los riesgos y conforme a cada uno se señalan cuáles son sus funciones específicas.

Roles para el Manejo de Riesgos	Funciones
Jefe del Proyecto	Debe establecer un plan donde se debe contemplar los riesgos que sean identificados para el proyecto, adicionalmente dicho plan puede contener las descripciones, análisis, prioridades y estrategias que

	<p>sirvan para minimizar el impacto que los riesgos puedan llegar a tener.</p> <p>Asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios.</p> <p>Mantiene al equipo del proyecto enfocado en los objetivos.</p> <p>Gestión de riesgos. Planificación y control del proyecto.</p> <p>Identificar y valorar preliminarmente los riesgos encontrados.</p>
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla 29 Roles encargados de gestionar riesgos

II.1.1.4.3.4.5. Escala de Ponderación de los Riesgos

Se establecen un sistema de puntuación que sirva para establecer la calificación a los diversos riesgos, como propuesta tenemos:

Puntaje	Calificación	Descripción General
0% - 20%	Muy Bajo	Riesgo muy improbable, la probabilidad de ocurrencia es bajísima y es muy posible que no ocurra.
20% - 40%	Bajo	Riesgo de baja probabilidad de ocurrencia, es posible que la no ocurrencia de este.

40% - 60%	Moderado	Riesgo medianamente probable, puede ocurrir como puede no ocurrir.
60% - 80%	Alto	Riesgo muy probable, podría ocurrir, es muy probable la ocurrencia en el proceso de desarrollo.
80% - 100%	Muy alto	Riesgo altamente probable para ocurrir dado que las circunstancias que generarían este evento son muy probables.

Tabla 30 Escala de ponderación de riesgos

II.1.1.4.3.4.6. Lista de Riegos

Riesgo	Ponderación	Efecto	Estrategia		Plan de Contingencia	Políticas de Supervisión
			Anulación	Disminución o Preventiva		
La ausencia del Director del proyecto o que se desvíe del objetivo de terminar el proyecto.	20%	No poder llevar a cabo el proyecto y reprobar la materia.	-	*Ser consciente por sí mismo y trazarse el objetivo para poder llevar a su término el proyecto.		Realizar un control y seguimiento a los objetivos del proyecto.
Que el personal cambie en la institución	20%	Que no se brinde la misma ayuda al equipo de desarrollo.	-	*Conversaciones con el nuevo personal.		Realizar un control y seguimiento a posibles cambios del personal de la institución.

Los usuarios rehúsan utilizar el sistema.	20%	Los usuarios no quieren que su sistema sea automatizado.	-	*En conversación con los usuarios finales, indicar las ventajas que presentara el software que se desarrollara.		Realizar un control y seguimiento a los usuarios.
Los usuarios no tengan conocimiento en el manejo del sistema.	50%	El sistema sea mal utilizado o solo utilizado en parte.	-	*Aconseja al usuario interesado a cerca de capacitación para realizarse en el presente o futuro.		Realizar un control y seguimiento a los usuarios en el manejo del sistema.

Realizar una mala estimación del tamaño del software.	50%	El producto no será entregado en la fecha acordada.	-	*Compara con otros proyectos al nuestro para tener una vista general de una estimación del tamaño del proyecto.	Realizar un control y seguimiento a las estimaciones del proyecto.
Realizar una mala estimación de los riesgos.	25%	El software no llegue a satisfacer las necesidades del usuario.		*Obtener un mayor conocimiento de las acciones se realizan en la empresa cliente, las acciones que realizara nuestro	Realizar un control y seguimiento a los a los posibles riesgos.

No tener buen conocimiento en el manejo de las herramientas de trabajo.	30%	El software pueda tener fallas y no pueda ser entregado a tiempo.		*Asegurarse que las personas miembros del equipo de desarrollo cuentan con el conocimiento necesario y		Realizar un control y seguimiento a las herramientas con las que se va a trabajar.
Las Interfaces y el diseño de las pantallas no sean correctos.	30%	Las interfaces para el usuario lleguen a ser desagradables.		*Establecer con anticipación las interfaces y el entorno grafico a utilizar.		Realizar un control y seguimiento a las interfaces o pantallas que vayan de
No se cuenta con el Hardware y Software	50%	El sistema no pueda ser implementado.		*Aconsejar y plantear equipo de hardware y recursos de software		Realizar un control y seguimiento al software y hardware.

El personal está enfermo y no disponible en momentos críticos en el	50%	La empresa quede perjudicada y no se presente el proyecto en				
---------------------------------------------------------------------	-----	--------------------------------------------------------------	--	--	--	--

Tabla 31 Lista de riesgos

II.1.1.5. Referencias

IBERDROLA “Aplicaciones de la domótica”
<https://www.iberdrola.es/webibd/corporativa/iberdrola?IDPAG=ESWEBCLIHOGAHOASEESTAPL>

[consulta 4/13]

© Broken Mind 2010 - Reinventamos el concepto de hogar, Líderes en Casas Inteligentes y soluciones de Domótica: Implementar una solución de domótica o “Casa Inteligente” representa una inimaginable cantidad de beneficios que le facilitarán su vida diaria”.

<http://www.broken-mind.com/beneficios.php>

[consulta 4/13]

domotica-maiher: “Avances y ventajas de una instalación domótica”

<http://www.domotica-maiher.com/ventajas/>

[consulta 4/13]

Diario La Nación: Domótica

<http://es.wikipedia.org/wiki/Dom%C3%B3tica>

[consulta 4/13]

© Asociación Española de Domótica (CEDOM): Que es DOMOTICA

<http://www.cedom.es/que-es-domotica.php>

[consulta 4/13]

Share-Alike Non-Commercial: Domótica con Arduino

<http://es.opendomo.org/arduino>

[consulta 4/13]

Domino (DOMótica + arduino)

<http://es.opendomo.org/domino>

[consulta 4/13]

Oxer, Jonathan; Blemings, Hugh (28 de diciembre de 2009): Arduino

<http://es.wikipedia.org/wiki/Arduino>

[consulta 4/13]

Acceder al puerto serie programando en Java, 2ª parte

<http://www.hell-desk.com/programacion/acceder-al-puerto-serie-programando-en-java-2a-parte/>

[consulta 4/13]

Adriana Torrez 2010: Matriz de Marco Logico

<http://www.slideshare.net/PREVAL/matriz-de-marco-logico-presentation>

[consulta 4/13]

Wikipedia 2009: El enfoque de marco lógico (EML)

http://es.wikipedia.org/wiki/Marco_l%C3%B3gico

[consulta 4/13]

Instituto Latinoamericano y del Caribe de Planificación Económica y Social (ILPES): Metodología del marco lógico para la planificación, el seguimiento y la evaluación de proyectos y programas

<http://www.eclac.org/cgi-bin/getProd.asp?xml=/publicaciones/xml/9/22239/P22239.xml&xsl=/ilpes/tpl/p9f.xsl&base=/ilpes/tpl/top-bottom.xsl>

[consulta 4/13]

Gualberto J.M. Milocco: Sistema de Marco Lógico (SML)

<http://www.monografias.com/trabajos27/marco-logico/marco-logico.shtml>

[consulta 4/13]

Todo Sobre Proyectos - De Mónica Thompson

<http://todosobreproyectos.blogspot.com/2009/02/el-marco-logico.html>

[consulta 4/13]

hefrogan, sept. 2010: Pasos Para La Elaboraciòn Del Marco Logico

<http://www.buenastareas.com/ensayos/Pasos-Para-La-Elaboraci%C3%B2n-Del-Marco/685258.html>

[consulta 4/13]

Protocolo http:

http://es.wikipedia.org/wiki/Hypertext_Transfer_Protocol.

Que es html:

<http://es.wikipedia.org/wiki/HTML>

Recomendaciones para IEEE 830:

IEEE Recommended Practice for Software Requirements Specification. ANSI/IEEE std. 830, 1998.

Definición de Arduino:

<http://es.wikipedia.org/wiki/Arduino>

Principio de Domótica:

<http://e-archivo.uc3m.es/handle/10016/8248>

II.1.2. Modelo de Casos de Uso del Negocio

II.1.2.1. Introducción

El modelo de casos de uso del negocio es un artefacto de la disciplina requisitos en la metodología RUP la cual estamos implementando. Define los límites del proceso de modelado posterior.

II.1.2.1.1. Propósito

- Comprender la estructura y la dinámica de la organización.
- Comprender problemas actuales e identificar posibles mejoras

II.1.2.1.2. Alcance

- Describe los procesos de negocio y los clientes.
- Identifica y describe los procesos de negocio según los objetivos de la organización.
- Definir un caso de uso del negocio para cada proceso de negocio.

II.1.2.2. Descripción Textual de cada uno de los Procesos del Negocio.

Registro de usuarios

El registro de usuario se realizara en primera instancia con un administrador del sistema y una vez que se haya otorgado los privilegios necesarios este usuario será capaz de registrar nuevos usuarios los cuales tendrán privilegios limitados por el usuario con mayor jerarquía se entiende por el registro de usuarios con privilegios limitados a los miembros de la familia o persona que interactuaran con el sistema limitándose solamente al control y monitoreo del inmueble, los usuarios con privilegios limitados no podrán generar reportes por ejemplo, ya que cada acción o proceso que realicen el sistema lo guardara con fecha y hora haciendo referencia al usuario que realizo dicho proceso, un usuario como administrador de su propio inmueble podrá configurar el sistema para la gestión de luces, puertas, ventanas, cámaras de vigilancia, la temperatura, usuarios, etc.

Registro de perfiles personalizados por el usuario

El usuario podrá adicionar y personalizar perfiles de acción como por ejemplo podrá crear un perfil llamado “me voy a trabajar” en el cual se establece que se apaguen todas las luces

de la casa, se cierren puertas y ventanas y se activen las alarmas con solo seleccionar el perfil de esta manera se evita que el usuario este apagando las luces, cerrando las puertas y activando la alarma una por una. Otro ejemplo sería un perfil llamado “nos vamos de viaje” en el que se establezca que se enciendan las luces a cierta hora del día, se apaguen en otro momento del día, de manera totalmente automática, una vez más el usuario podrá monitorear el funcionamiento de DOMUINO desde cualquier dispositivo con acceso a internet, otro ejemplo del uso de perfiles es la gestión de la temperatura ambiente, en donde el usuario podrá establecer una temperatura en la cual quiere que permanezca su inmueble, en determinados días.

Registro de los puntos de control del inmueble

En la base de datos se almacenara cada uno de los puntos de control, entiéndase como punto de control a todos los dispositivos eléctricos y electrónicos que se instalaran y controlaran en el inmueble. Como ser sensores de luz, de presencia, de llamas, motores que gestionaran el cierre y apertura de puertas y ventanas, lámparas, cámaras de seguridad, y otros. El registro en la base de datos se realizara con un código de dispositivo, nombre y ubicación.

Registro de los datos del día

DOMUINO será capaz de guardar en la base de datos todas y cada una de las acciones que realicen los usuarios durante todo el día y se podrá generar un reporte con los datos almacenados en el momento que el usuario así lo requiera, los datos guardados nos permitirán saber con exactitud qué acciones y procesos se realizaron y que usuario se encargó de tales movimientos, esto será posible ya que se guardara el nombre de usuario y el tipo de acción o proceso que realizo con fecha y hora.

Consulta de la situación del inmueble en tiempo real

DOMUINO tendrá la capacidad de mostrar el estado de los dispositivos instalados en la vivienda de manera global y en cualquier momento del día y desde cualquier lugar desde donde el usuario tenga acceso a internet.

Consulta del estado de los dispositivos instalados

DOMUINO mostrara si cada uno de los dispositivos instalados en la vivienda se encuentra activo o inactivo o si es que cambio de estado de manera inesperada, es decir en el caso de que haya un intruso o de que se hayan activado o desactivado algunos dispositivos de forma manual o repentina.

Manejo de los diferentes dispositivos instalados en el inmueble

A diferencia del manejo de perfiles de usuario, en este punto se podrá realizar los cambios de estado (activo o inactivo) de los dispositivos de manera independiente es decir uno por uno y haciendo uso de DOMUINO desde un navegador WEB.

Gestión de la temperatura ambiente

DOMUINO permitirá al usuario ver cuál es la temperatura en su vivienda en cualquier momento del día en tiempo real, además podrá determinar una temperatura para que DOMUINO active la calefacción o el aire acondicionado de manera automática, esto gracias al sensor de humedad y temperatura DTH11 que se instalará en el inmueble.

II.1.2.3. Descripción de Actores del negocio

Al mismo tiempo que se determinan los procesos del negocio, es posible identificar los agentes implicados en el mismo.

Se identificó los siguientes actores:

- **Administrador Del Sistema**
Persona encargada de la configuración del sistema (software - hardware) y usuario con todos los privilegios tanto en el sistema como en el manejo de las placas ARDUINO.
- **Cliente**
Personas encargadas de altas, bajas y modificaciones del sistema con privilegios reservados y limitados por el administrador del sistema.
- **Usuarios**

Personas o empresas tienen limitados privilegios en cuanto al control y monitoreo de inmuebles.

II.1.3. Modelo de Objetos del Negocio

II.1.3.1. Introducción

El modelo de objetos del negocio es un artefacto de la disciplina requisitos en la metodología RUP la cual está implementando.

II.1.3.1.1. Propósito

- Comprender la Estructura y la Dinámica de los sistemas en tiempo real aplicados a la domótica.
- Comprender los procesos del negocio de la organización.

II.1.3.1.2. Alcance

- Describe el comportamiento de los procesos de negocio

Identificar y definir los objetos del negocio.

II.1.4. Glosario

II.1.4.1. Introducción

El presente documento recoge los términos manejados durante la elaboración del proyecto de desarrollo del sistema. Se trata de un diccionario informal de datos y de definiciones de la nomenclatura que se maneja en la construcción del sistema.

II.1.4.1.1. Propósito

El propósito del presente documento es definir la terminología manejada en el proyecto a desarrollar, también sirve como guía de consulta para la clarificación de los puntos conflictivos o poco esclarecidos del proyecto.

II.1.4.1.2. Alcance

El alcance del presente documento se extiende a todo el proyecto en desarrollo.

II.1.4.2. Organización del Glosario

El presente documento está organizado por definiciones de términos ordenados de forma ascendente según ordenación alfabética tradicional.

II.1.4.3. Definiciones

II.1.4.3.1. Administrador

Un administrador de sistema es aquella persona que se dedica a mantener y operar un sistema cómputo.

II.1.4.3.2. Actividad

Conjunto de operaciones o tareas propias de un componente de un proyecto.

II.1.4.3.3. Beneficiarios

Los usuarios en general que deseen implementar los beneficios de la domótica y los sistemas en tiempo real de la ciudad de Tarija provincia cercado.

II.1.4.3.4. Inmueble

Instalaciones donde se realiza la instalación de las placas arduino y los componentes electrónicos que interactuaran con el sistema en los lugares que los usuarios soliciten.

II.1.4.3.5. Componente

Módulo de un proyecto que tiene una función específica. Está compuesta por actividades que buscan cumplir con los objetivos previstos.

II.1.4.3.6. UML

UML son las siglas para Unified Modeling Language, que en castellano quiere decir: Lenguaje de Modelado Unificado. Para comprender que es el UML, basta con analizar cada una de las palabras que lo componen, por separado [3].

- **Lenguaje:** El UML es, precisamente, un lenguaje. Lo que implican que éste cuenta con una sintaxis y una semántica. Por lo tanto, al modelar un concepto en UML, existen reglas sobre cómo deben agruparse los elementos del lenguaje y el significado de esta agrupación.
- **Modelado:** El UML es visual. Mediante su sintaxis se modelan distintos aspectos del mundo real, que permiten una mejor interpretación y entendimiento de este.
- **Unificado:** Unifica varias técnicas de modelado en un sola.

Ya que el UML proviene de técnicas orientadas a objetos, se crea con la fuerte intención de que este permita un correcto modelado orientado a objetos.

Análisis: Proceso que permite pasar del sistema real a un modelo conceptual.

II.1.5. Modelo de Casos de Uso

II.1.5.1. Introducción

El presente documento es un artefacto de la disciplina de requisitos en la metodología RUP la cual estamos implementando.

II.1.5.1.1. Propósito

- Comprender la estructura y la dinámica del sistema desarrollado.
- Identificar el nivel de complejidad del sistema.
- Identificar posibles mejoras.
- Modelar el contexto del sistema
- Involucrar a los usuarios en las etapas iniciales del análisis y diseño del sistema

II.1.5.1.2. Alcance

- Identificar y definir procesos del sistema según los objetivos de la organización
- Definir un caso de uso para cada proceso del sistema (el diagrama de caso de uso nos detalla el contexto y los límites de la organización).
- Descubrir los procesos del sistema y del cliente.

II.1.5.2. Diagrama de Casos de Uso.

II.1.5.2.1. Caso de Uso General.

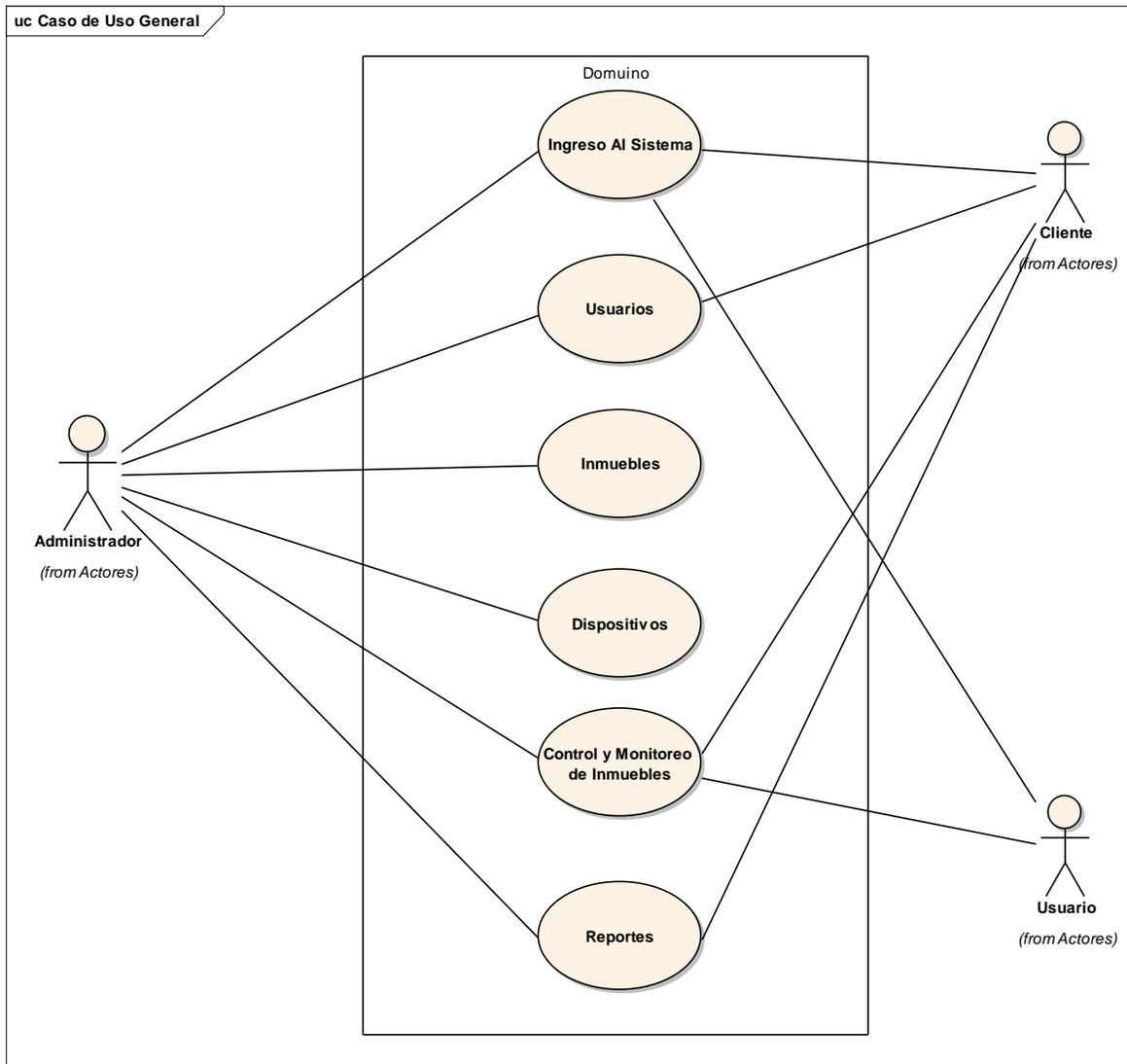


Figura 3 Caso de Uso General

II.1.5.2.2. Ingreso al Sistema

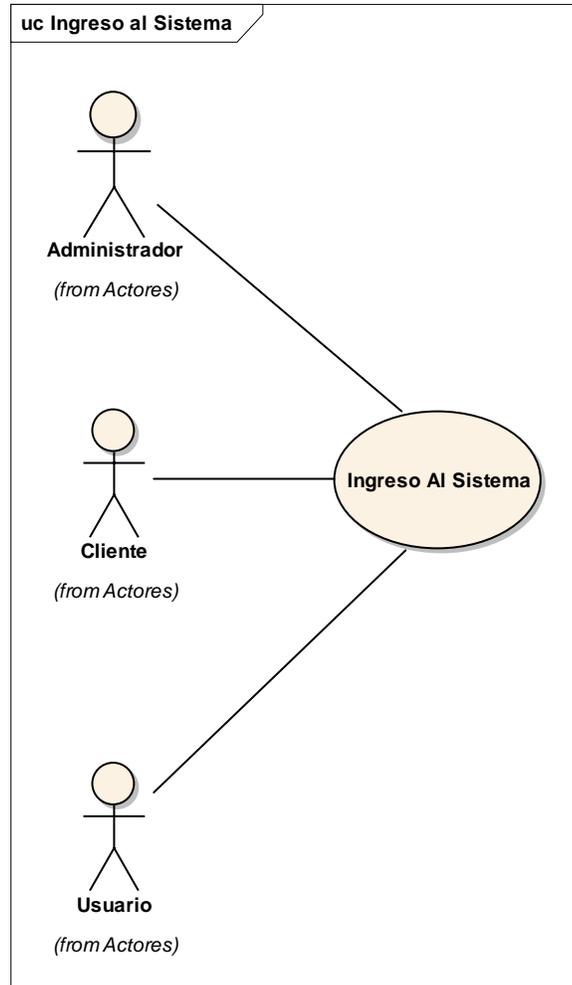


Figura 4 Caso de uso ingreso al sistema

II.1.5.2.3. Gestionar Usuarios

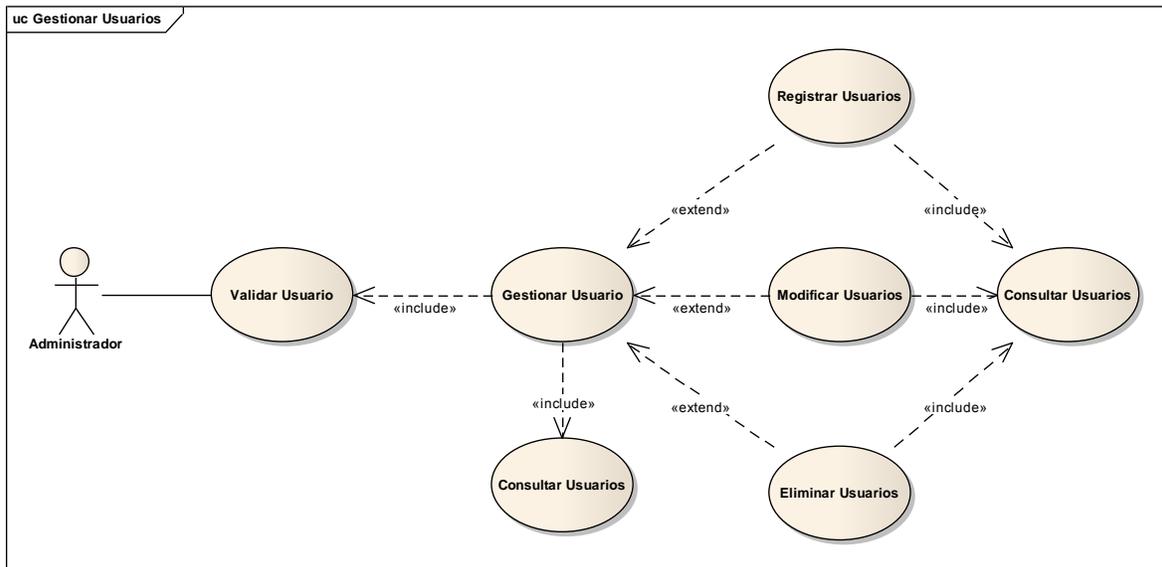


Figura 5 Diagrama de caso de uso gestionar usuarios

II.1.5.2.4. Gestionar Dispositivos

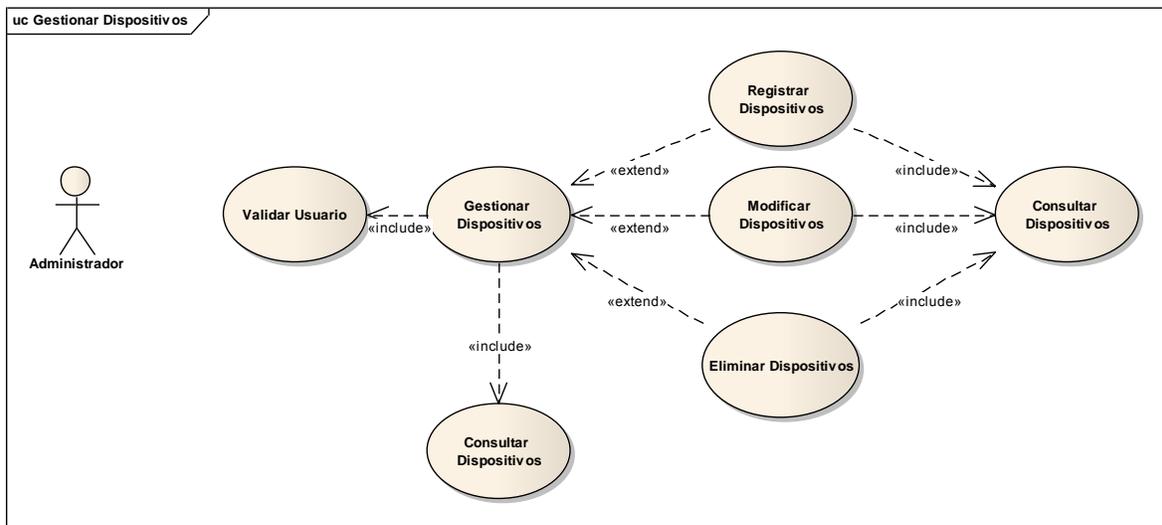


Figura 6 Diagrama de caso de uso gestionar dispositivos

II.1.5.2.5. Gestionar Inmueble

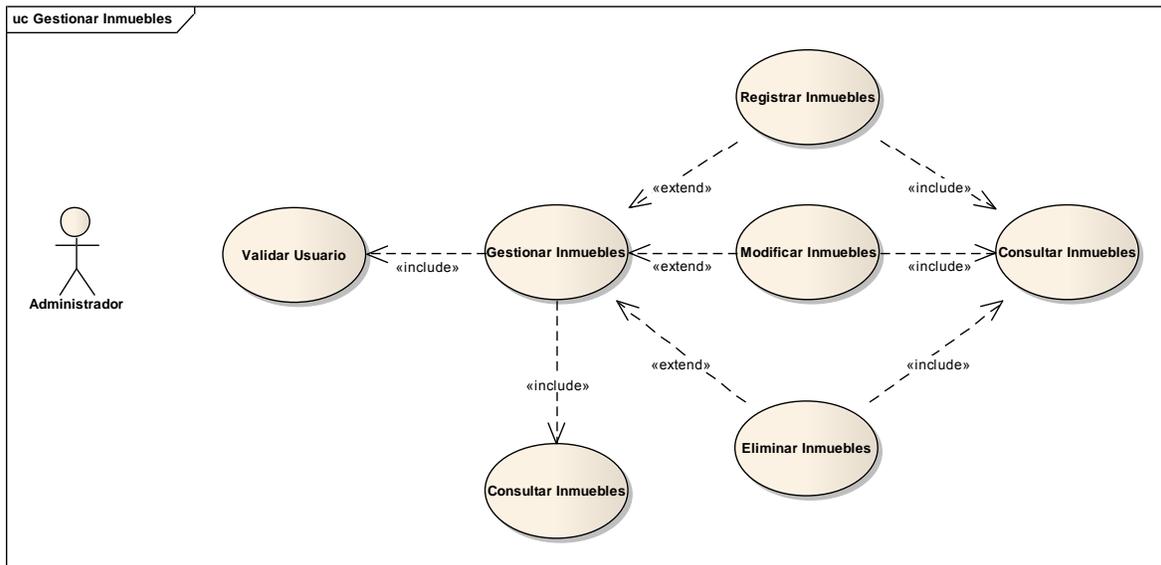


Figura 7 Diagrama de caso de uso gestionar inmueble

II.1.5.2.6. Gestionar Ambientes

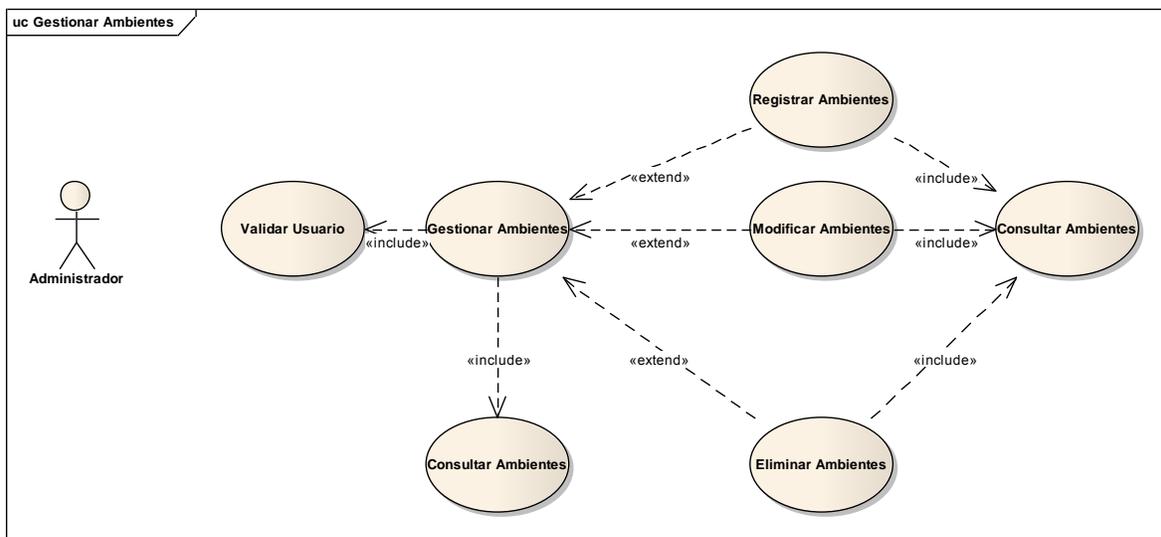


Figura 8 Diagrama de caso de uso gestionar ambientes

II.1.5.2.7. Gestionar Controlador

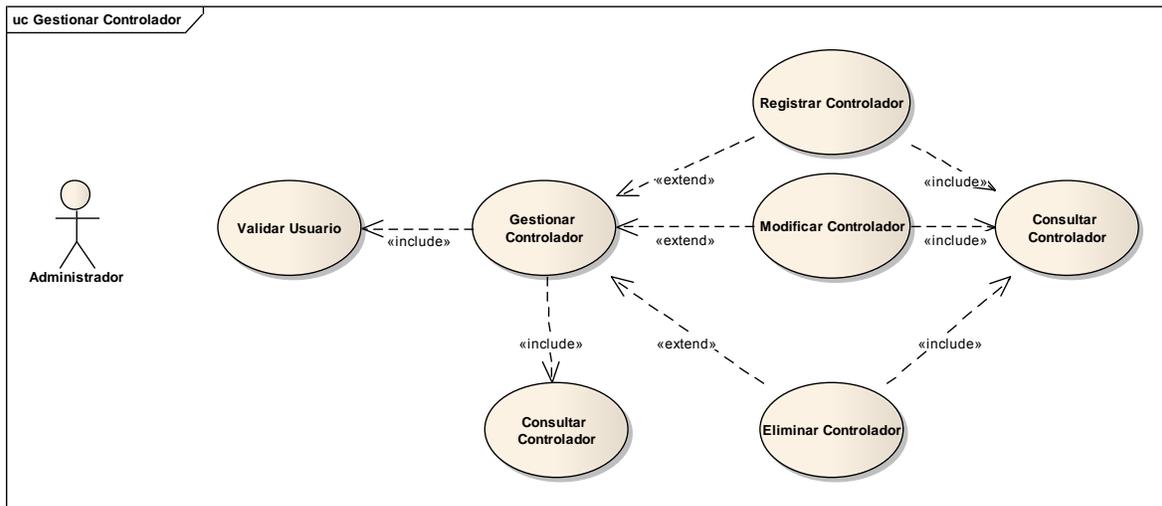


Figura 9 Diagrama de caso de uso gestionar controlador

II.1.5.2.8. Gestionar Dispositivo – Ambiente

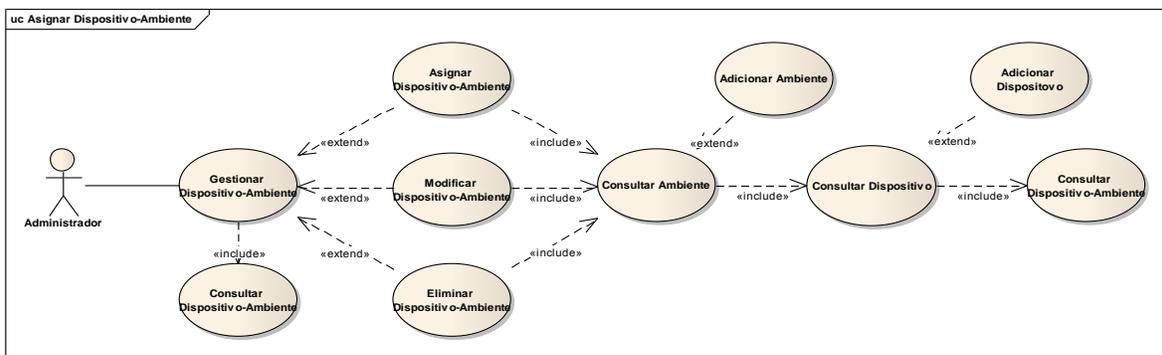


Figura 10 Diagrama de caso de uso gestionar dispositivo - ambiente

II.1.5.2.9. Gestionar Ambiente – Piso

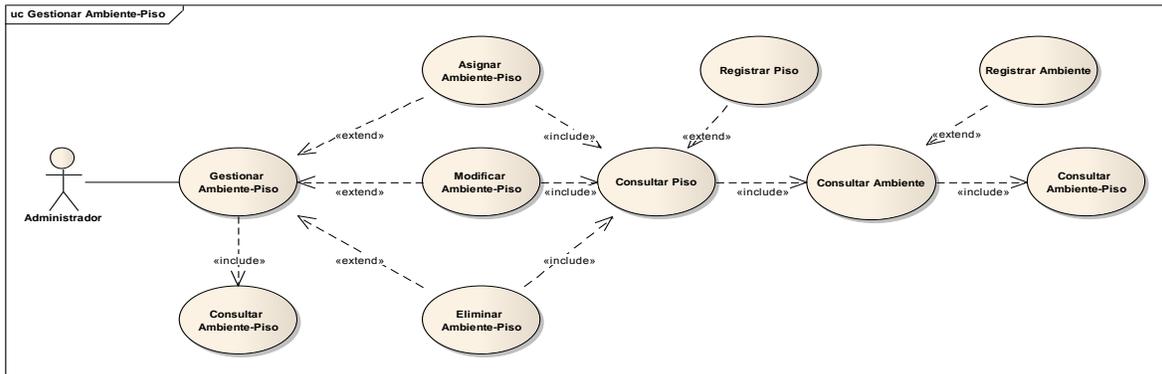


Figura 11 Diagrama de caso de uso gestionar ambiente - piso

II.1.5.2.10. Gestionar Piso – Inmueble

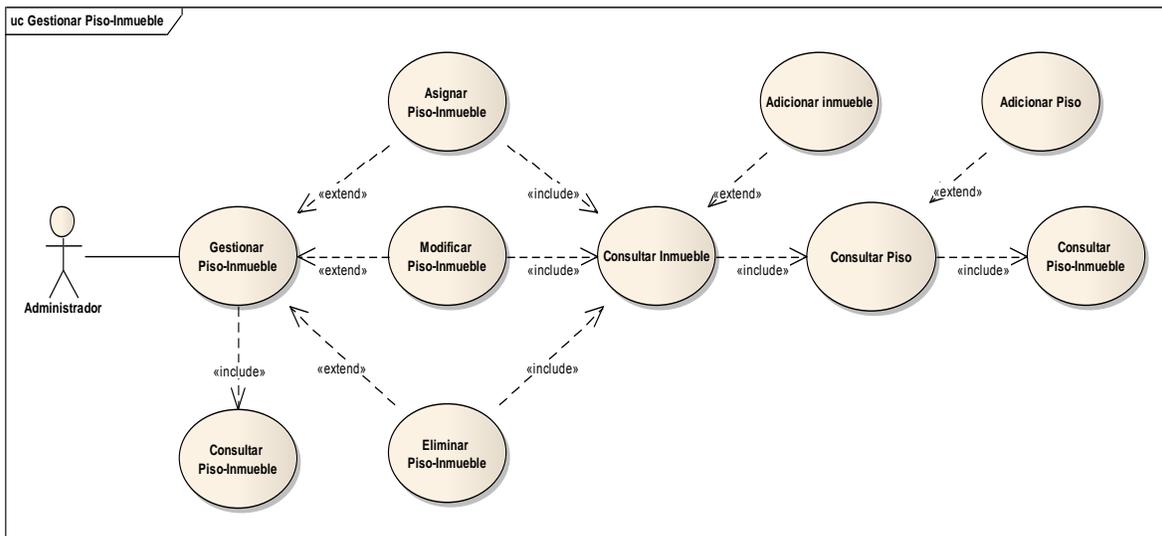


Figura 12 Diagrama de caso de uso gestionar piso - inmueble

II.1.5.2.11. Gestionar Inmueble – Usuario

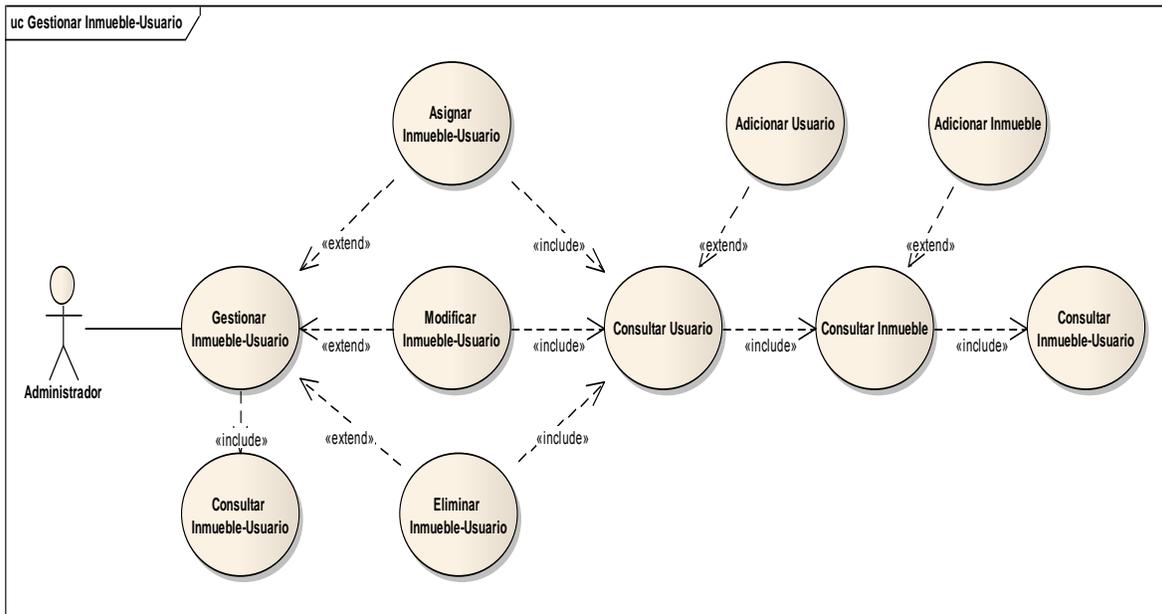


Figura 13 Diagrama de caso de uso gestionar inmueble - usuario

II.1.5.2.12. Gestionar Roles

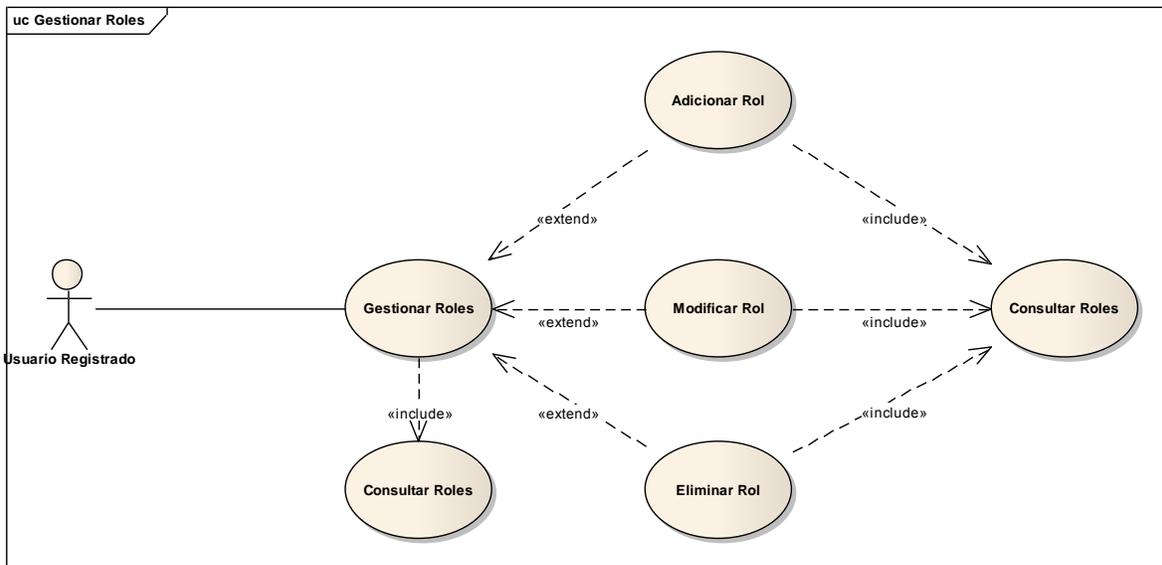


Figura 14 Diagrama de caso de uso gestionar roles

II.1.5.2.13. Gestionar Manejo de Dispositivos

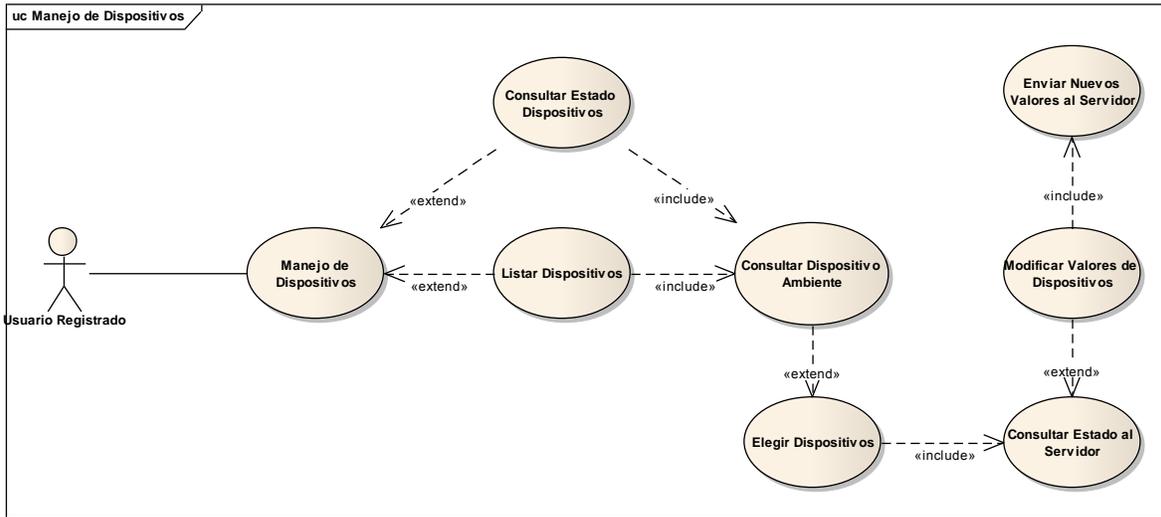


Figura 15 Diagrama de caso de uso gestionar manejo de dispositivos

II.1.5.2.14. Gestionar Reportes

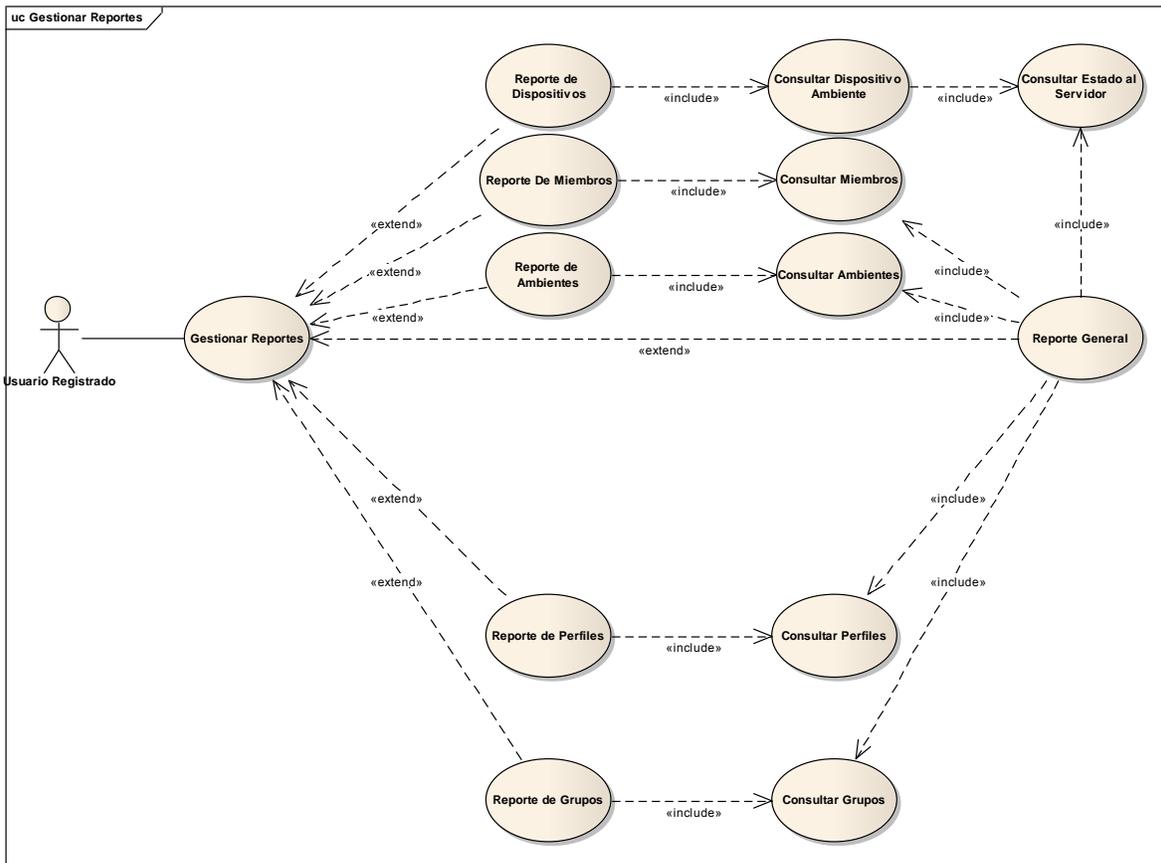


Figura 16 Diagrama de caso de uso gestionar reportes

II.1.6. Especificaciones

II.1.6.1. Introducción

Las Especificaciones de los casos de uso es una descripción detallada de los casos de uso del sistema.

II.1.6.1.1. Propósito

Interpretar y describir los casos de uso.

II.1.6.1.2. Alcance

Describe los procesos internos de los casos de uso.

II.1.6.2. Especificaciones de Casos de Uso

II.1.6.2.1. Ingreso del usuario al sistema

II.1.6.2.1.1. Caso de Uso: Ingreso al Sistema

Actor: Administrador, Clientes, Usuarios.

Explicación: En el caso de uso Validar usuario se verifica la identidad del usuario para ingresar en el sistema.

Descripción:

Curso Normal	Alternativa
1. Se muestra la interface Ingresar al sistema	
2. Se introduce Usuario	
3. Se introduce Contraseña	
4. Por aceptar el sistema valida usuario y clave ingresados	
5. Si los datos son correctos, se verifica la existencia del usuario y clave en la base datos.	5. Si el usuario es incorrecto se muestra mensaje de error “Verifique Usuario y Contraseña” en la pantalla de ingreso al sistema.
6. Si el usuario existe y este es el administrador de DOMUINO se muestra la pantalla principal del modo administrador en donde se puede ver, agregar, eliminar y modificar, usuarios, dispositivos, inmuebles.	

<p>Si el usuario es cliente, se muestra la pantalla principal del modo cliente en donde se muestra los inmuebles que le pertenecen y se le permite gestionar usuarios de nivel 2 y modificar nombres</p> <p>Si el usuario es de tipo 3 se le muestra la pantalla de control y monitoreo únicamente del ambiente que se le fue designado</p>	
<p>Precondiciones:</p> <p>El actor no tiene privilegios de poder ver las prestaciones del sistema.</p>	
<p>Pos condiciones: El usuario ingresar al sistema con los privilegios asignados de acuerdo a su rol, para realizar acciones en el sistema.</p>	

Tabla 32 Caso de uso: Ingresar al sistema

II.1.6.2.2. Gestión Usuarios

II.1.6.2.2.1. Caso de Uso: Gestionar Usuarios

Actor: Administrador

Explicación: En el caso de uso de Gestionar Usuarios se explica el proceso de cómo se carga la “Gestión de Usuarios” desde el Módulo del Administrador.

Descripción:

Curso Normal	Alternativas
1. El administrador selecciona “Gestionar Usuarios”	
2. Se Carga todos los usuarios en sus diferentes niveles y con sus respectivas opciones.	

Tabla 33 Caso de uso: Gestionar usuarios

II.1.6.2.2.2. Caso de Uso: Agregar Usuario

Actor: Administrador

Explicación: En el caso de uso Usuario se explica el proceso de cómo se puede agregar usuarios desde el Módulo Administrador.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona la opción “Nuevo Usuario”	
2. Se despliega la pantalla “Formulario de Registro”	2. El Administrador Presiona “Regresar” y vuelve a la pantalla Gestionar Usuarios.
3. El Administrador llena todos los datos requeridos y presionar “Crear Cuenta”	3. El sistema determina que no llenaron correctamente los campos de texto y muestra un mensaje.
4. Se guardan los datos correctamente desplegando un mensaje de registro de usuario satisfactorio en la pantalla Gestión de Usuarios.	4. Se produce un error en el ingreso de datos del usuario y se despliega un mensaje de error.

Tabla 34 Caso de uso: Agregar usuario

II.1.6.2.2.3. Caso de Uso: Modificar Usuario

Actor: Administrador

Explicación: En el caso de uso “Modificar Usuario” se explica el proceso de cómo se modifica un usuario de la base de datos, para ello se debe tener registrado un usuario y seleccionarlo de la tabla.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un usuario y presiona el botón modificar	
2. Se carga la pantalla “Modificar Usuario”	2. El Administrador al presionar “Regresar” se vuelve a la pantalla Gestionar Usuarios.
3. El Administrador modifica los datos que se ve convenientes hacerlo y presiona “Modificar”	3. El sistema determina que no se llenaron correctamente los campos de texto y muestra un mensaje.

4. Se guardan los datos modificados correctamente desplegando un mensaje de confirmación y se vuelve a la pantalla “Gestionar Usuarios”	4. Se produce un error en el ingreso de los datos del usuario y se despliega un mensaje de error.
-----------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------

Tabla 35 Caso de uso: Modificar usuario

II.1.6.2.2.4. Caso de Uso: Ver Usuario

Actor: Administrador

Explicación: En el caso de uso “Ver Usuario” se explica el proceso de cómo se puede ver los datos de usuario, para ello se debe tener registrado un usuario y seleccionarlo de la tabla.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un usuario y presiona “detalles”	
2. Se carga la pantalla “Detalle Usuario” con los datos del usuario	3. El Administrador presiona “Aceptar” y se vuelve a la pantalla de “Gestionar Usuarios”. 4. El administrador presiona “Editar” y se muestra la pantalla “Modificar Usuario”
5. El Administrador puede ver todos los datos del usuario.	

Tabla 36 Caso de uso: Ver usuario

II.1.6.2.3. Gestionar Dispositivos

II.1.6.2.3.1. Caso de Uso: Gestionar Dispositivos

Actor: Administrador

Explicación: En el caso de uso de Gestionar Dispositivos se explica el proceso de cómo se carga la “Gestión de Dispositivo” desde el Módulo del Administrador.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona “Dispositivos”	

2. Se carga todos los dispositivos de los diferentes inmuebles y sus respectivas opciones.	
--------------------------------------------------------------------------------------------	--

Tabla 37 Gestionar dispositivos

II.1.6.2.3.2. Caso de Uso: Agregar Dispositivo

Actor: Administrador

Explicación: En el caso de uso de Agregar Dispositivo se explica el proceso de cómo se puede agregar desde el Módulo del Administrador.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona “Registrar Dispositivo”	
2. Se despliega la pantalla “Formulario de Registro”	2. El Administrador Presiona “Regresar” y se vuelve la pantalla “Gestionar Dispositivo”.
3. El Administrador llena todos los datos requeridos y presionar guardar	3. El sistema determina que no llenaron correctamente los campos de texto y muestra un mensaje.
4. Se guardan los datos correctamente desplegando un mensaje de registro de dispositivo satisfactorio en la pantalla Gestión de Dispositivos	4. Se produce un error en el ingreso de datos del usuario y se despliega un mensaje de error.

Tabla 38 Agregar Dispositivo

II.1.6.2.3.3. Caso de Uso: Modificar Dispositivo

Actor: Administrador

Explicación: En el caso de uso “Modificar Dispositivo” se explica el proceso de cómo se modifica un usuario de la base de datos, para ello se debe tener registrado un dispositivo y seleccionarlo de la tabla.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un dispositivo y presiona el botón modificar	
2. Se carga la pantalla “Modificar Dispositivo”.	2. El Administrador presiona “Regresar” y se vuelve a la pantalla “Gestionar Dispositivos”
3. El Administrador modifica los datos que se ve convenientes hacerlo y presiona “Modificar”	3. El sistema determina que no se llenaron correctamente los campos de texto y muestra un mensaje.
4. Se guardan los datos modificados correctamente desplegando un mensaje de confirmación y se vuelve a la pantalla de “Gestionar Dispositivos”.	4. Se produce un error en el ingreso de los datos del usuario y se despliega un mensaje de error.

Tabla 39 Caso de uso: Modificar dispositivo

II.1.6.2.3.4. Caso de Uso: Ver Dispositivo

Actor: Administrador

Explicación: En el caso de uso “Ver Dispositivo” se explica el proceso de cómo se puede ver los datos del dispositivo, para ello se debe tener registrado y seleccionarlo de la tabla.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un Dispositivo y presiona “Detalles”	

2. Se carga la pantalla “detalle dispositivo” con los datos del dispositivo	2. El Administrador presiona “Regresar” y se vuelve a la pantalla de “Gestionar Dispositivos”.
3. El Administrador puede ver todos los datos del dispositivo y puede imprimir sus datos	

Tabla 40 Caso de uso: Ver dispositivos

II.1.6.2.4. Gestionar Inmuebles

II.1.6.2.4.1. Caso de Uso: Gestionar Inmueble

Actor: Administrador

Explicación: En el caso de uso de Gestionar Inmueble se explica el proceso de cómo se carga la “Gestión Inmueble” desde el Módulo del Administrador.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona “Inmuebles”	
2. Se carga todos los inmuebles de los diferentes usuarios y sus respectivas opciones.	

Tabla 41 Caso de uso: Gestionar inmueble

II.1.6.2.4.2. Caso de Uso: Agregar Inmueble

Actor: Administrador

Explicación: En el caso de uso de Agregar Inmueble se explica el proceso de cómo se puede agregar inmueble desde el Módulo del Administrador.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona la opción “Agregar Inmueble”	
2. Se despliega la pantalla “Agregar Inmueble”	2. El Administrador Presiona “Cancelar” y se vuelve a la pantalla “Gestionar Inmueble”.

3. El Administrador llena todos los datos requeridos y presionar guardar	3. El sistema determina que no llenaron correctamente los campos de texto y muestra un mensaje.
4. Se guardan los datos correctamente desplegando un mensaje de confirmación en la pantalla “Gestionar Inmueble”	4. Se produce un error en el ingreso de datos del usuario y se despliega un mensaje de error.

Tabla 42 Caso de uso: Agregar inmueble

II.1.6.2.4.3. Caso de Uso: Modificar Inmueble

Actor: Administrador

Explicación: En el caso de uso “Modificar Inmueble” se explica el proceso de cómo se modifica un inmueble de la base de datos, para ello se debe tener registrado un usuario y seleccionarlo de la tabla.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un inmueble y presiona el botón modificar	
2. Se carga la pantalla “Modificar Inmueble”	2. El Administrador presionar “Cancelar” y se vuelve a la pantalla “Gestionar Inmuebles”
3. El Administrador modifica los datos que se ve convenientes hacerlo y presiona “Guardar”	3. El sistema determina que no se llenaron correctamente los campos de texto y muestra un mensaje.
4. Se guardan los datos modificados correctamente desplegando un mensaje de confirmación en la pantalla “Gestionar Inmueble”	4. Se produce un error en el ingreso de los datos del usuario y se despliega un mensaje de error.

Tabla 43 Caso de uso: Modificar inmueble

II.1.6.2.4.4. Caso de Uso: Ver Inmueble

Actor: Administrador

Explicación: En el caso de uso “Ver Inmueble” se explica el proceso de cómo se puede ver los datos del Inmueble, para ello se debe tener registrado y seleccionarlo de la tabla.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un Enfermera y presiona “Ver”	
2. Se carga la pantalla “Ver Inmueble” con los datos del inmueble	2. El Administrador presiona “Cancelar” y vuelve a la pantalla “Gestionar Inmuebles”
3. El administrador puede ver todos los datos del inmueble y puede imprimir sus datos	

Tabla 44 Caso de uso: Ver inmueble

II.1.6.2.5. Gestionar Ambientes

II.1.6.2.5.1. Caso de Uso: Gestionar Ambientes

Actor: Administrador

Explicación: En el caso de uso de Gestionar ambientes se explica el proceso de cómo se carga la “Gestión Ambientes” desde el Módulo del administrador.

Curso Normal	Alternativas
1. El Administrador selecciona “Gestionar Ambientes”	
2. Se carga todos los Ambientes de los diferentes inmuebles y sus respectivas opciones.	

Tabla 45 Caso de uso: Gestionar ambientes

II.1.6.2.5.2. Caso de Uso: Agregar Ambiente

Actor: Administrador

Explicación: En el caso de uso de Agregar Ambiente se explica el proceso de cómo se puede agregar desde el Módulo del Administrador.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona la opción “Agregar Ambiente”	
2. Se despliega la pantalla “Agregar Ambiente”	2. El Administrador Presiona “Cancelar” y se vuelve a la pantalla “Gestionar Ambiente”
3. El Administrador llena todos los datos requeridos y presiona guardar	3. El sistema determina que no llenaron correctamente los campos de texto y muestra un mensaje.
4. Se guardan los datos correctamente desplegando un mensaje de confirmación en la pantalla “Gestionar Ambiente”	4. Se produce un error en el ingreso de datos del usuario y se despliega un mensaje de error.

Tabla 46 Caso de uso: Agregar ambiente

II.1.6.2.5.3. Caso de Uso: Modificar Ambiente

Actor: Administrador

Explicación: En el caso de uso “Modificar Ambiente” se explica el proceso de cómo se modifica un usuario de la base de datos, para ello se debe tener registrado un ambiente y seleccionarlo de la tabla.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un Ambiente y presiona el botón modificar	
2. Se carga la pantalla “Modificar Ambiente”	2. El Administrador presiona “Cancelar” se vuelve a la pantalla “Gestionar Ambientes”

3. El Administrador modifica los datos que se ve convenientes hacerlo y presiona “Guardar”	3. El sistema determina que no se llenaron correctamente los campos de texto y muestra un mensaje.
4. Se guardan los datos modificados correctamente desplegando un mensaje de confirmación en la pantalla “Gestionar Ambiente”.	4. Se produce un error en el ingreso de los datos del usuario y se despliega un mensaje de error.

Tabla 47 Caso de uso: Modificar inmueble

II.1.6.2.5.4. Caso de Uso: Ver Ambiente

Actor: Administrador

Explicación: En el caso de uso “Ver Ambiente” se explica el proceso de cómo se puede ver los datos del Inmueble, para ello se debe tener registrado un Ambiente y seleccionarlo de la tabla.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un Ambiente y presiona “Ver”	
2. Se carga la pantalla “Ver Ambiente” con los datos del ambiente	2. El Administrador presiona “Cancelar” y se vuelve a la pantalla “Gestionar Ambiente”
3. El Administrador puede ver todos los datos del Ambiente puede imprimir sus datos	

Tabla 48 Caso de uso: Ver ambiente

II.1.6.2.6. Gestionar Dispositivo - Ambiente

II.1.6.2.6.1. Caso de Uso: Dispositivo - Ambiente

Actor: Administrador

Explicación: En el caso de uso de Gestionar Dispositivo - Ambiente se explica el proceso de cómo se carga la “Gestión Dispositivo - Ambiente” desde el Módulo del Administrador.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona “Gestionar Dispositivo - Ambiente”	
2. Se carga todos dispositivo y ambiente que existe en los diferentes inmuebles.	

Tabla 49 Caso de uso: Gestionar dispositivo - ambiente

II.1.6.2.6.2. Caso de Uso: Modificar Dispositivo - Ambiente

Actor: Administrador

Explicación: En el caso de uso “Modificar Dispositivo - Ambiente” se explica el proceso de cómo se modifica un Dispositivo - Ambiente de la base de datos, para ello se debe tener registrado el Dispositivo - Ambiente y seleccionarlo de la tabla.

Curso Normal	Alternativas
1. El Administrador selecciona un dispositivo y presiona el botón modificar	
2. Se carga la pantalla “Modificar Dispositivo - Ambiente”	2. El administrador presiona “Cancelar” y se vuelve a la pantalla “Gestionar Dispositivo - Ambiente”
3. El administrador modifica los datos que se ve convenientes hacerlo y presiona “Guardar”	3. El sistema determina que no se llenaron correctamente los campos de texto y muestra un mensaje.
4. Se guardan los datos modificados correctamente desplegando un mensaje de confirmación en la pantalla “Gestionar Dispositivo - Ambiente”.	4. Se produce un error en el ingreso de los datos del usuario y se despliega un mensaje de error.

Tabla 50 Caso de uso modificar: Dispositivo - Ambiente

II.1.6.2.6.3. Caso de Uso: Ver Dispositivo - Ambiente

Actor: Administrador

Explicación: En el caso de uso “Ver Dispositivo - Ambiente” se explica el proceso de cómo se puede ver los datos de los Dispositivo - Ambiente, para ello se debe tener registrado y seleccionarlo de la tabla.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un Rol y presiona “Ver”.	
2. Se carga la pantalla “Ver Dispositivo - Ambiente” con los datos del Dispositivo - Ambiente.	2. El Administrador presiona “Cancelar” y se vuelve a la pantalla “Gestionar Dispositivo - Ambiente”
3. El Administrador puede ver todos los datos del Dispositivo - Ambiente y puede imprimir.	

Tabla 51 Caso de uso: Ver dispositivo - ambiente

II.1.6.2.7. Gestionar Inmueble - Usuario

II.1.6.2.7.1. Case de Uso: Gestionar Inmueble - Usuario

Actor: Administrador

Explicación: En el caso de uso de Gestionar Inmueble - Usuario se explica el proceso de cómo se carga la “Gestionar Inmueble - Usuario” desde el Módulo del Administrador.

Descripción:

Curso Normal	Alternativas
1. El administrador selecciona “Registrar Inmueble”	
2. Se carga la pantalla formulario de registro.	
3. El usuario llena el formulario en el cual se encuentra un campo con selección de usuario.	

Tabla 52 Caso de Uso Gestionar Inmueble - Usuario.

II.1.6.2.7.2. Caso de Uso: Modificar Inmueble - Usuario

Actor: Administrador

Explicación: En el caso de uso “Modificar Inmueble - Usuario” se explica el proceso de cómo se modifica un Inmueble - Usuario de la base de datos, para ello se debe tener registrado un Inmueble - Usuario y seleccionarlo de la tabla.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un Inmueble - Usuario y presiona el botón modificar	
2. Se carga la pantalla “Modificar Inmueble - Usuario”	2. El Administrador presionar “Cancelar” y se vuelve a la pantalla “Gestionar Inmueble - Usuario”
3. El Administrador modifica los datos que se ve convenientes hacerlo y presiona “Guardar”	3. El sistema determina que no se llenaron correctamente los campos de texto y muestra un mensaje.
4. Se guardan los datos modificados correctamente desplegando un mensaje de confirmación en la pantalla “Gestionar Inmueble - Usuario”	4. Se produce un error en el ingreso de los datos del usuario y se despliega un mensaje de error.

Tabla 53 Caso de uso: Modificar inmueble – usuario

II.1.6.2.7.3. Caso de Uso: Ver Inmueble - Usuario

Actor: Administrador

Explicación: En el caso de uso “Ver Inmueble - Usuario” se explica el proceso de cómo se puede ver los Inmueble - Usuario, para ello se debe tener registrado y seleccionarlo de la tabla.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un reporte inmuebles “Reportes Inmueble”	

2. Se carga la pantalla “reportes”.	2. El administrador presiona “Regresar” y se vuelve a la pantalla “Gestionar Inmueble - Usuario”
3. El Administrador puede ver todos los datos.	
4. El administrador Selecciona reporte de inmuebles por usuario	

Tabla 54 Caso de uso: Ver inmueble usuario

II.1.6.2.8. Caso de Uso Generar Reporte

II.1.6.2.8.1. Caso de Uso: Generar Reporte

Actor: Administrador

Explicación: En el caso de uso “Generar Reportes” se explica el proceso de cómo se puede ver datos generales del inmueble y de sus componentes.

Descripción:

Curso Normal	Alternativas
1. El Administrador selecciona un “Reporte”	
2. Se carga la pantalla “Reportes” donde puede seleccionar los diferentes módulos del cual quiere sacar un reporte.	
3. El Administrador puede ver las diferentes opciones para generar un reporte.	3 . Puede Volver a la pantalla de reportes, presionando el botón volver en la pantalla
4. Se despliega una pantalla con el reporte deseado.	4 . Puede guardar el Reporte en formato PDF .

Tabla 55 Caso de uso: Generar reporte

II.1.7. Modelado de Diagrama de Actividades

II.1.7.1. Introducción

- El diagrama de actividades es un artefacto de la disciplina análisis de sistemas en la metodología RUP la cual estamos implementando.
- Los diagramas de actividad se utilizan para modelar los aspectos dinámicos de un sistema, esto implica modelar los pasos secuenciales de un proceso.

II.1.7.1.1. Propósito

- Comprender la estructura y la dinámica del sistema deseado.
- Identificar posibles mejoras en el sistema.

II.1.7.1.2. Alcances

- Describir los procesos del sistema y los clientes
- Identificar y definir los procesos de los casos de uso según los objetivos de la organización.
- Definir un diagrama de actividad para cada caso de uso del sistema.

II.1.7.1.3. Diagrama de Actividades

II.1.7.1.3.1. Diagrama de Actividad: Caso de Uso Ingreso al Sistema

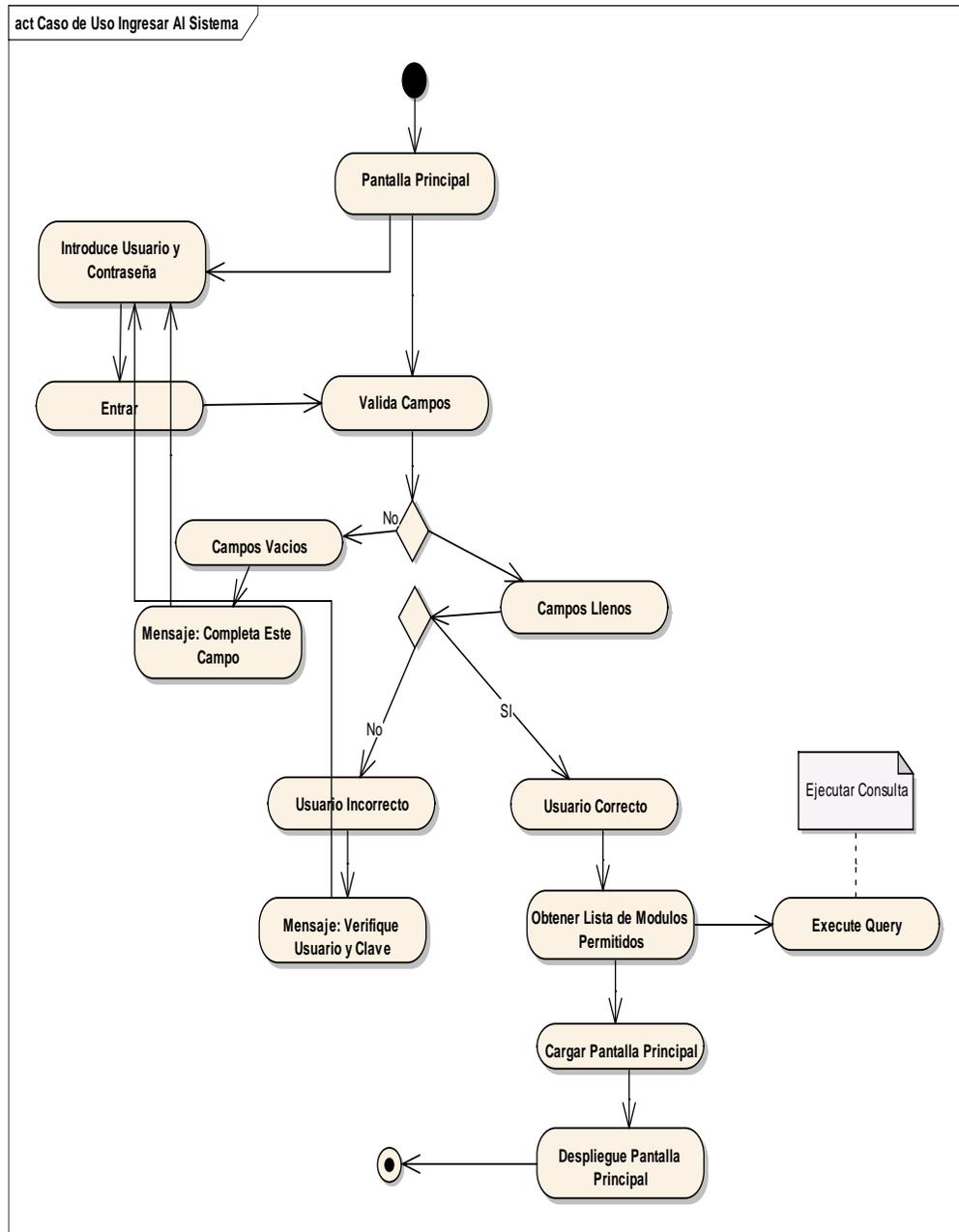


Figura 17 Diagrama de actividad: Caso de uso ingreso al sistema

II.1.7.1.3.2. Diagrama de Actividad: Caso de Uso Gestionar Usuario

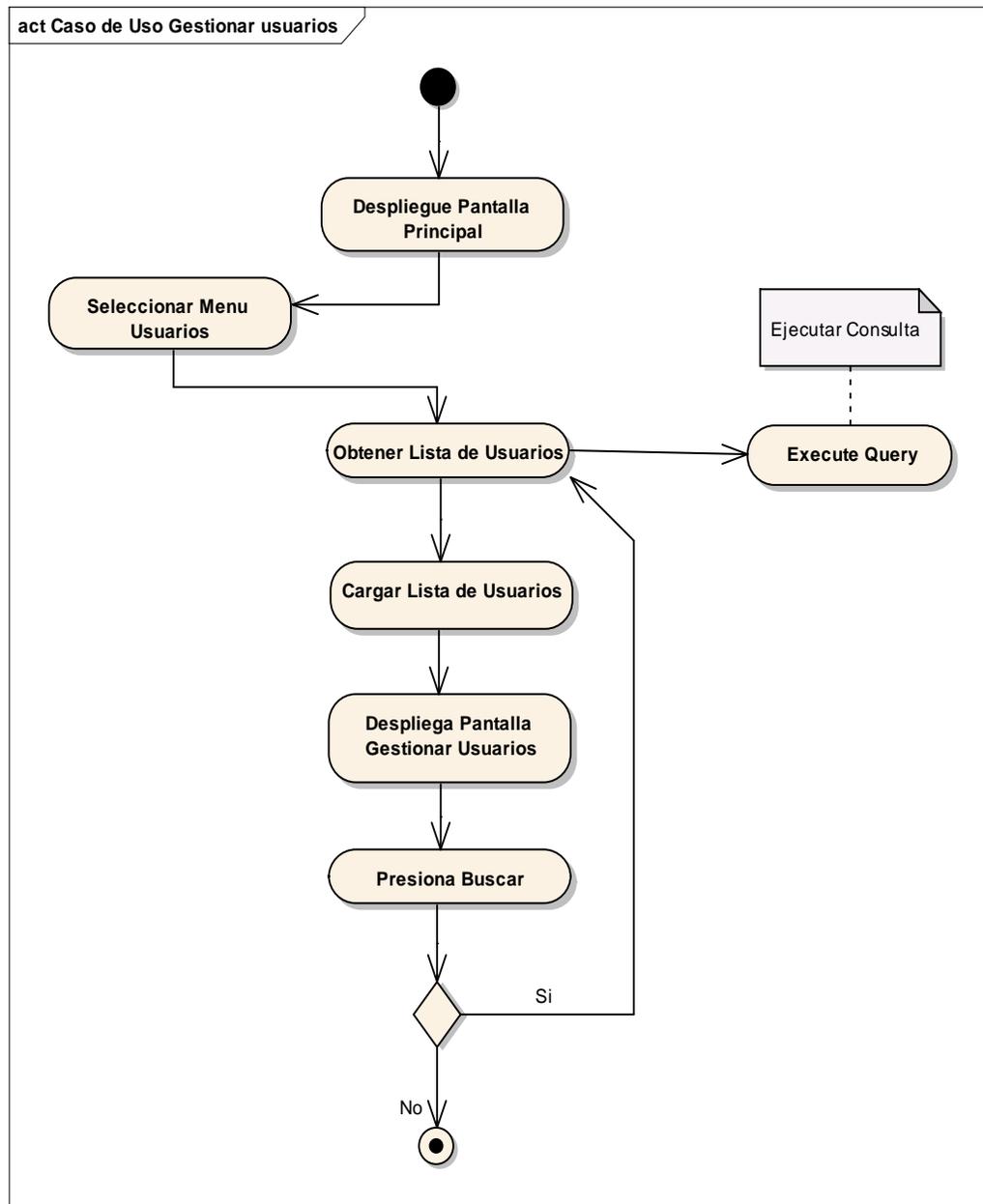


Figura 18 Diagrama de actividad: Caso de uso gestionar usuario

II.1.7.1.3.3. Diagrama de Actividad: Caso de Uso Registrar Usuarios

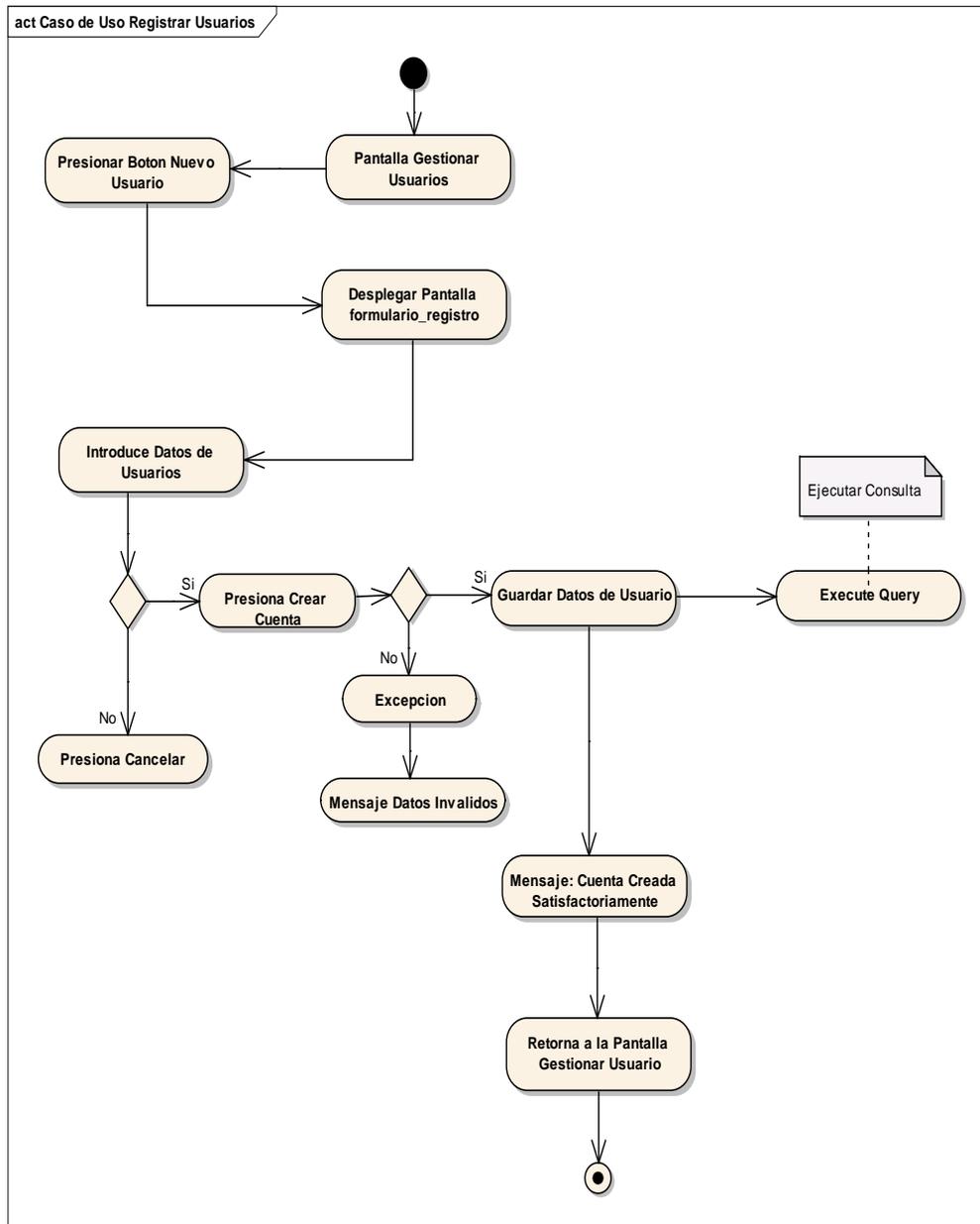


Figura 19 Diagrama de actividad: Caso de uso registrar usuarios

II.1.7.1.3.4. Diagrama de Actividad: Caso de Uso Modificar Usuarios

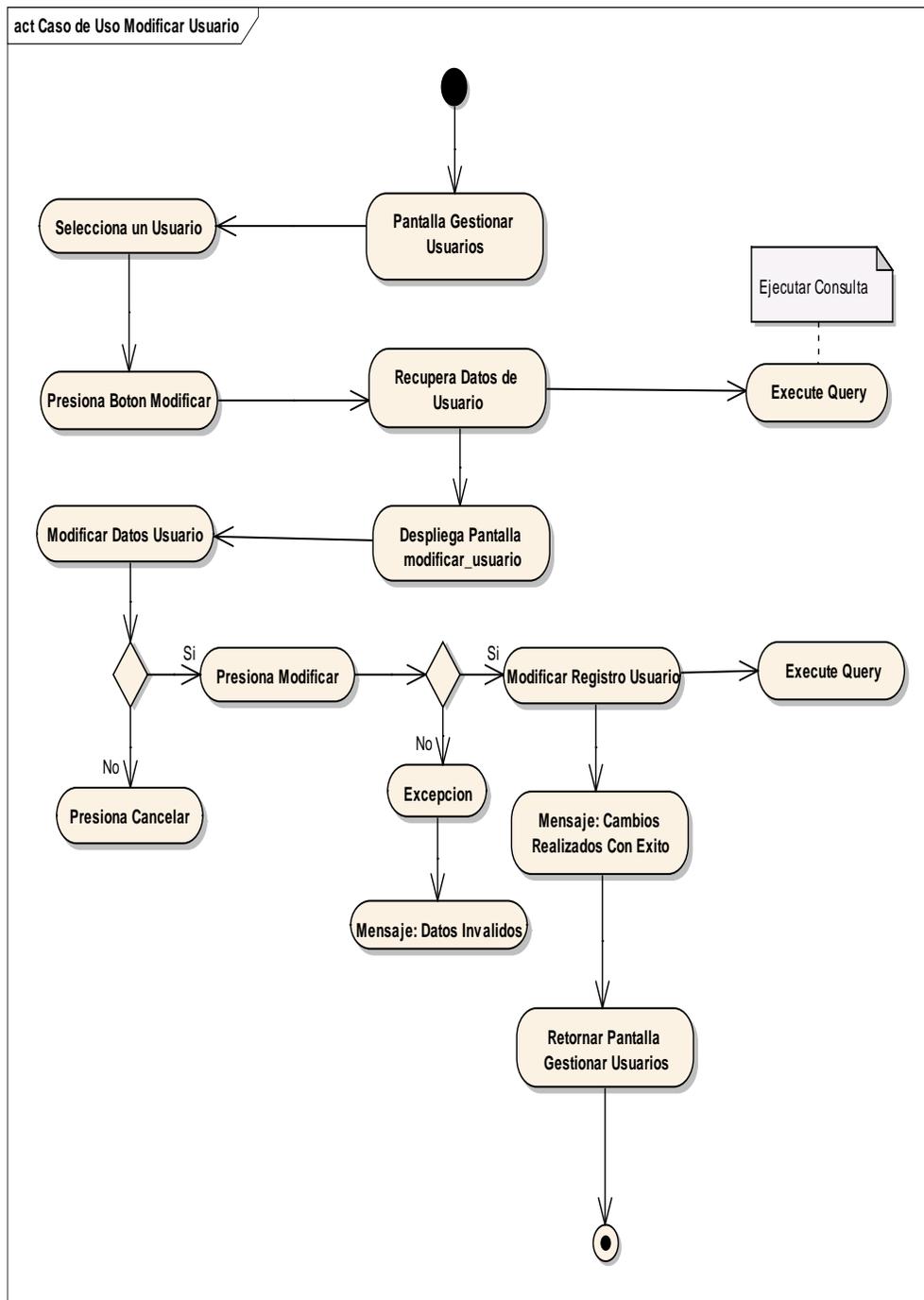


Figura 20 Diagrama de actividad: Caso de uso modificar usuario

II.1.7.1.3.5. Diagrama de Actividad: Caso de Uso Eliminar Usuarios

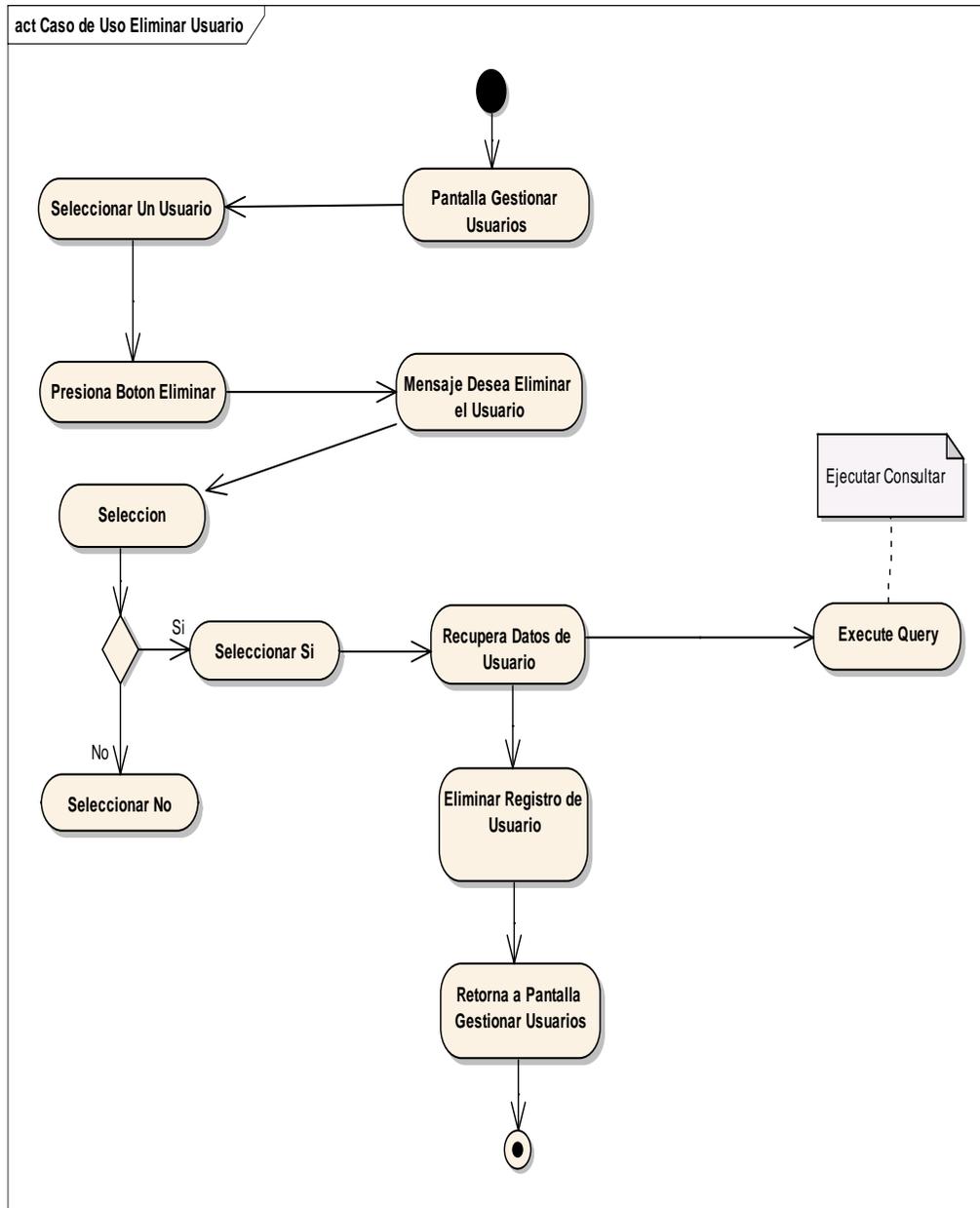


Figura 21 Diagrama de actividad: Caso de uso eliminar usuarios

II.1.7.1.3.6. Diagrama de Actividad: Caso de Uso Gestionar Dispositivos

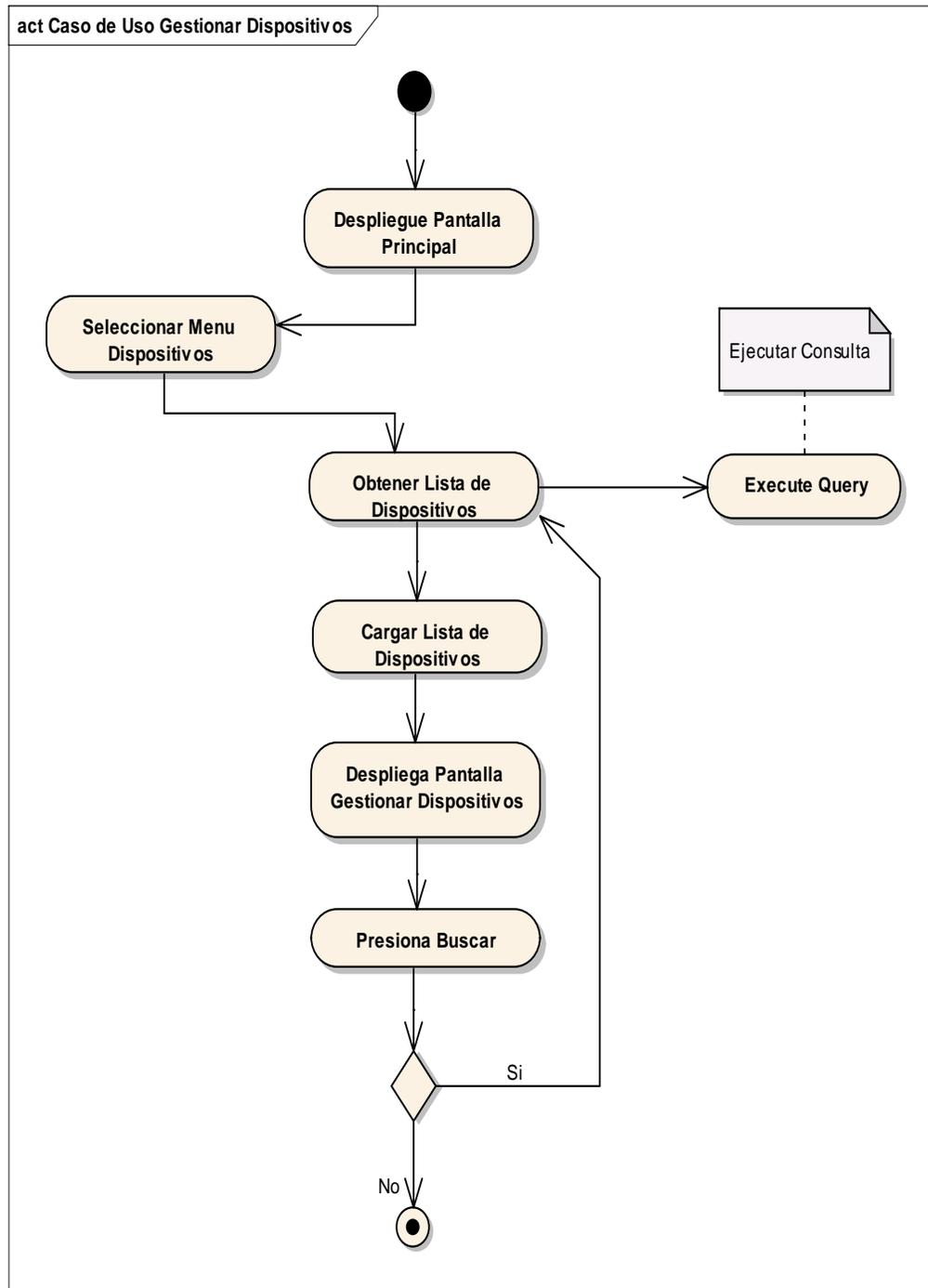


Figura 22 Diagrama de actividad: Caso de uso gestionar dispositivos

II.1.7.1.3.7. Diagrama de Actividad: Caso de Uso Registrar Dispositivos

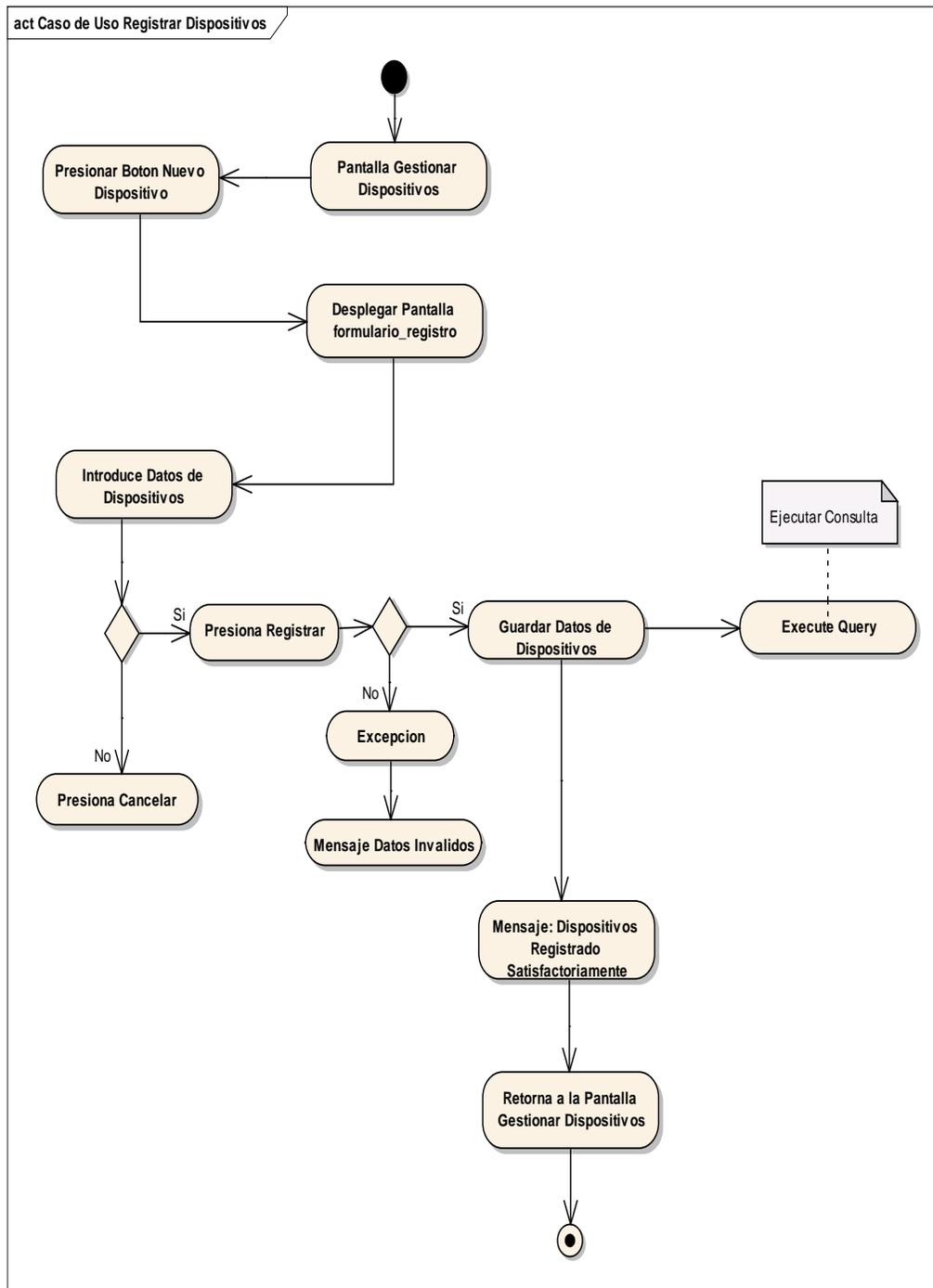


Figura 23 Diagrama de actividad: Caso de uso registrar dispositivos

II.1.7.1.3.8. Diagrama de Actividad: Caso de Uso Modificar Dispositivos

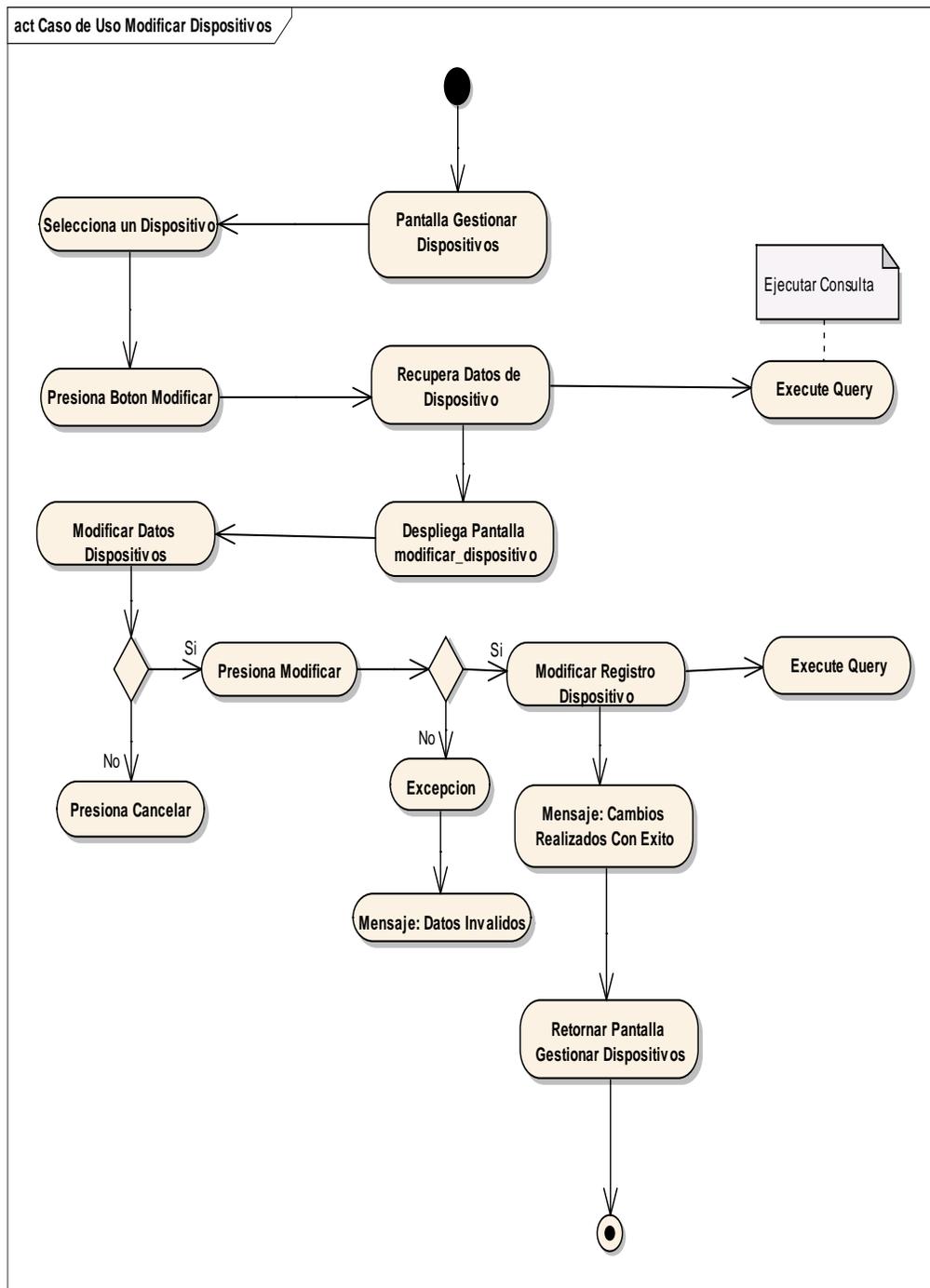


Figura 24 Diagrama de actividad: Caso de uso modificar dispositivos

II.1.7.1.3.9. Diagrama de Actividad: Caso de Uso Eliminar Dispositivos

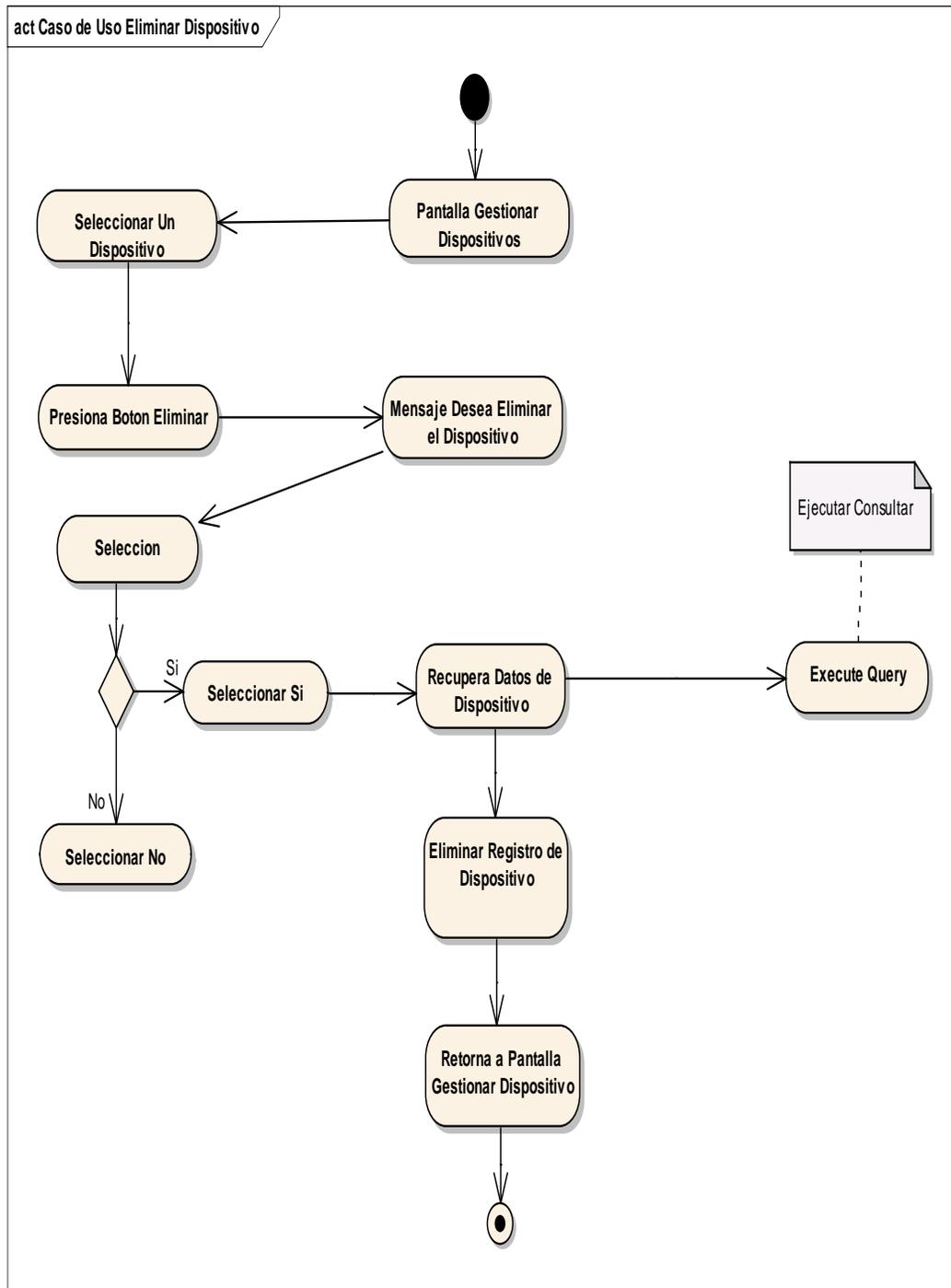
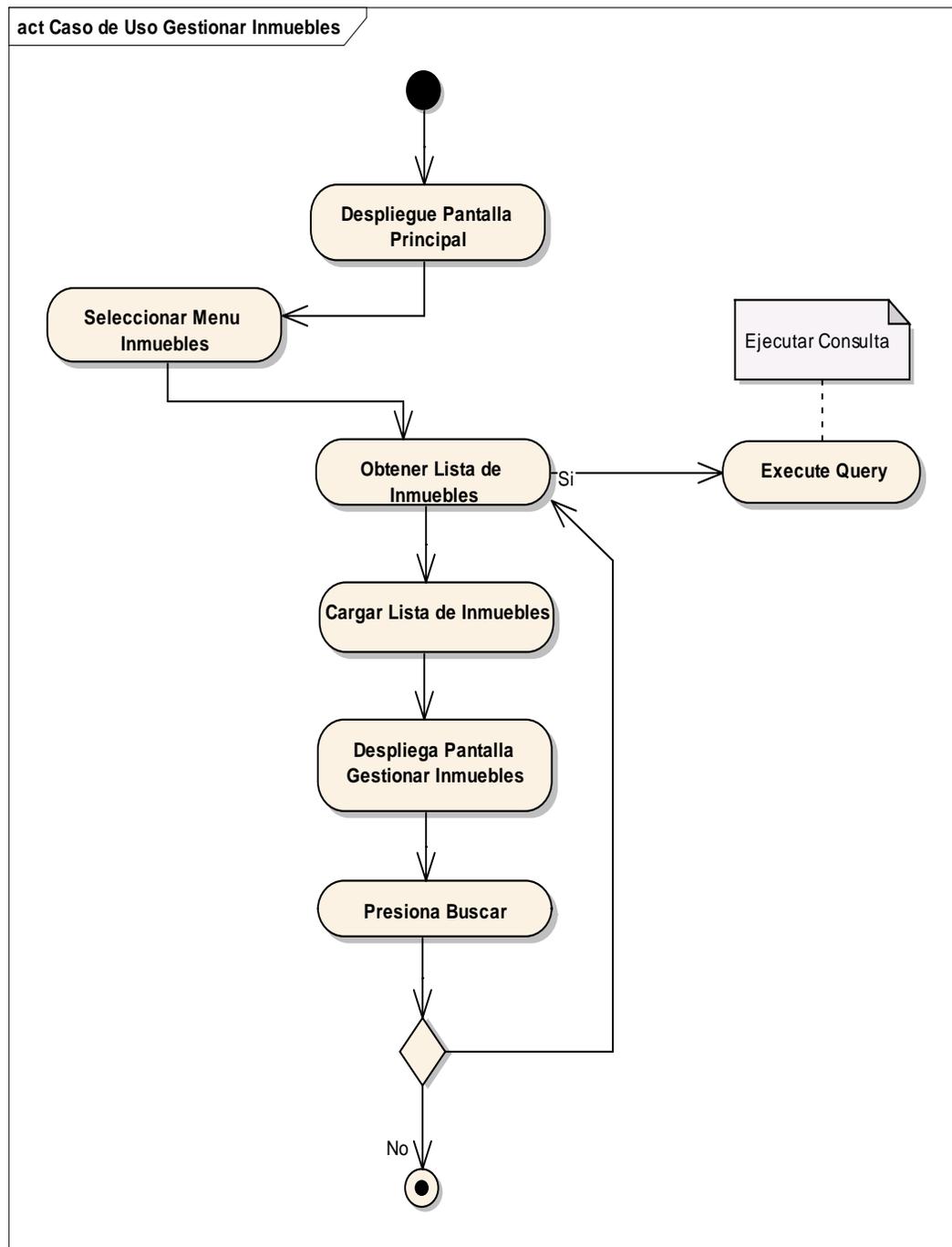


Figura 25 Diagrama de actividad: Caso de uso eliminar dispositivos

II.1.7.1.3.10. Diagrama de Actividad: Caso de Uso Gestionar Inmuebles**Figura 26 Diagrama de actividad: Caso de uso gestionar inmuebles**

II.1.7.1.3.11. Diagrama de Actividad: Caso de Uso Registrar Inmuebles

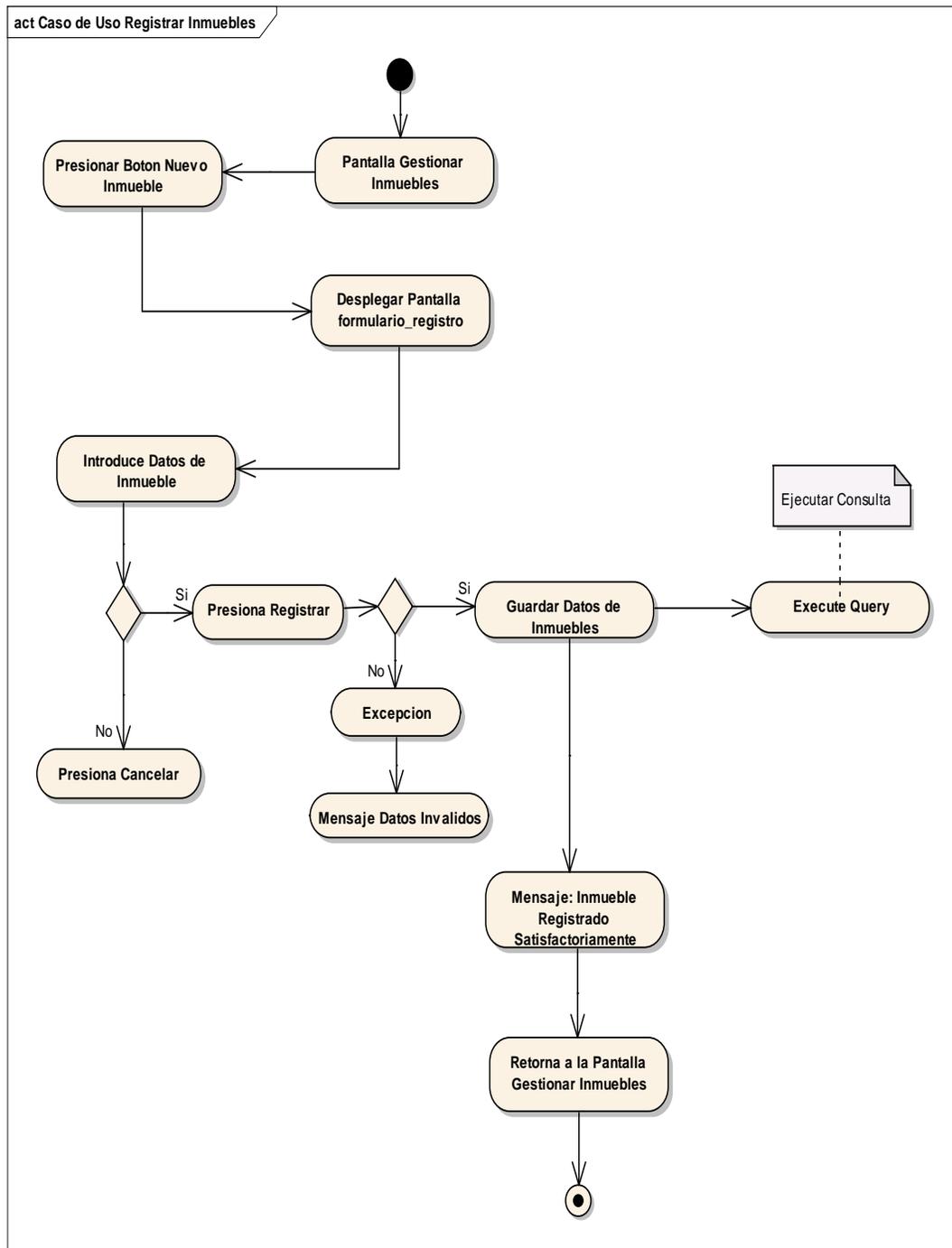


Figura 27 Diagrama de actividad: Caso de uso registrar inmuebles

II.1.7.1.3.12. Diagrama de Actividad: Caso de Uso Modificar Inmuebles

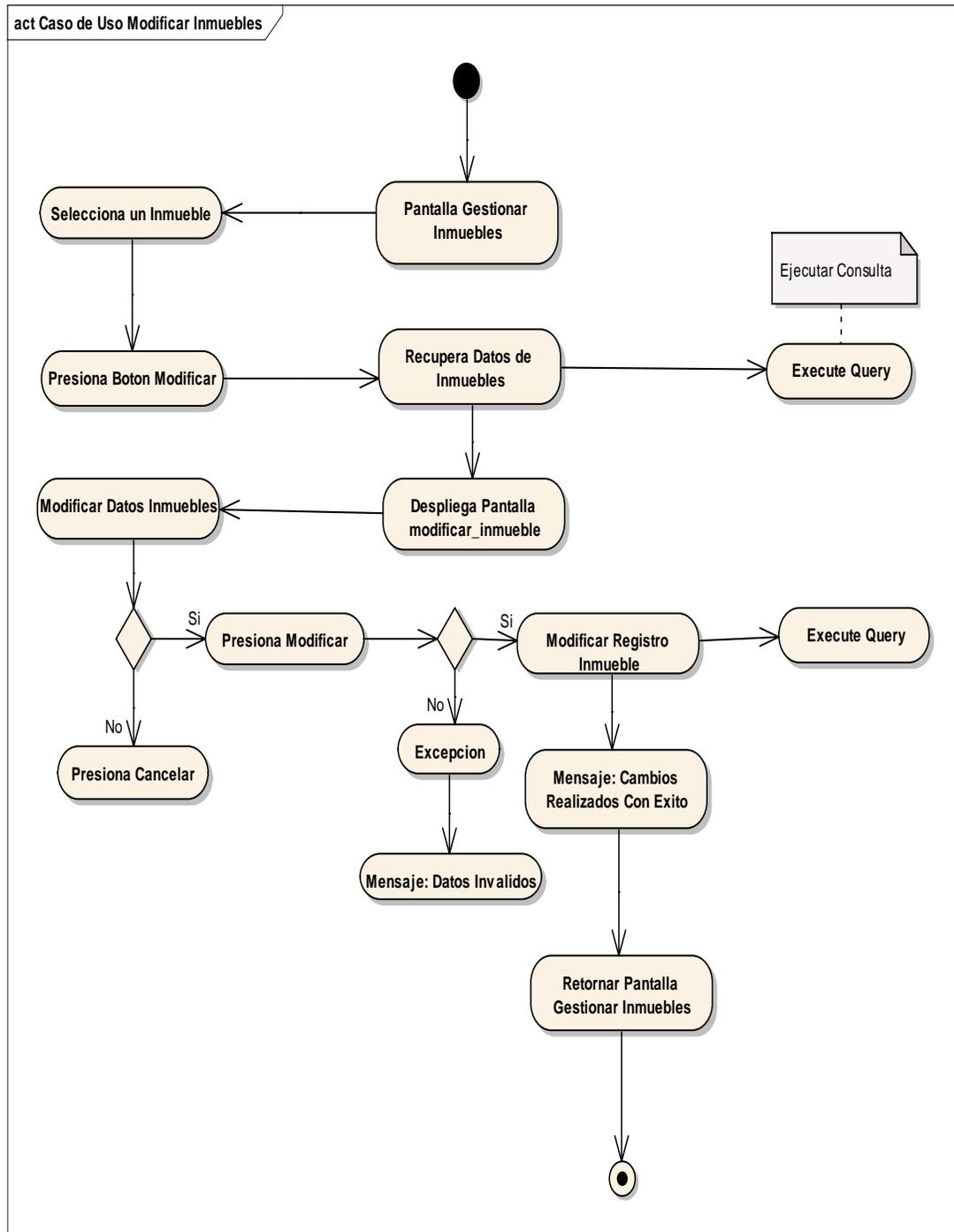


Figura 28 Diagrama de actividad: Caso de uso modificar inmuebles

II.1.7.1.3.13. Diagrama de Actividad: Caso de Uso Eliminar Inmuebles

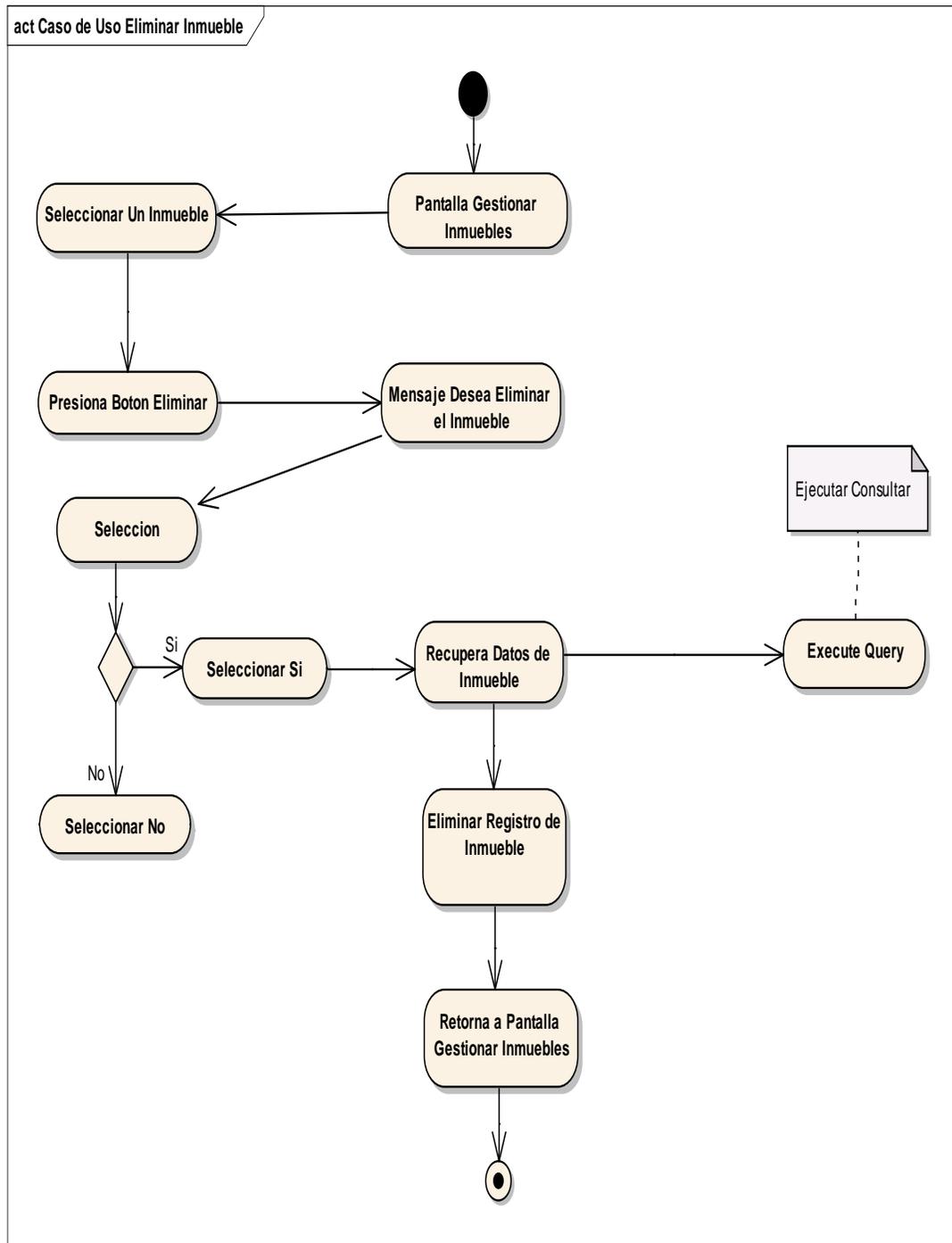


Figura 29 Diagrama de actividad: Caso de uso eliminar inmuebles

II.1.7.1.3.14. Diagrama de Actividad: Caso de Uso Gestionar Ambientes

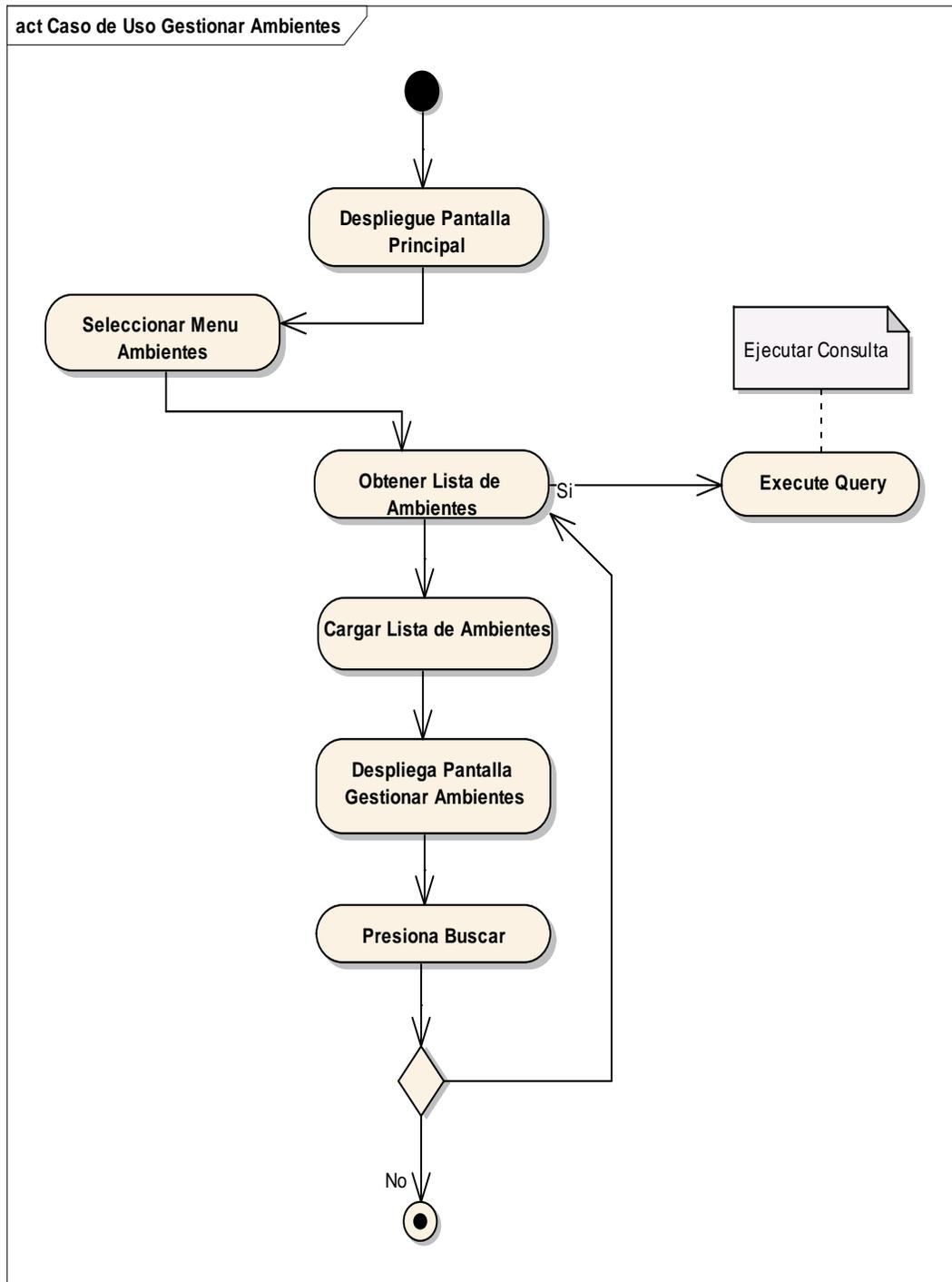


Figura 30 Diagrama de actividad: Caso de uso gestionar ambientes

II.1.7.1.3.15. Diagrama de Actividad: Caso de Uso Registrar Ambientes

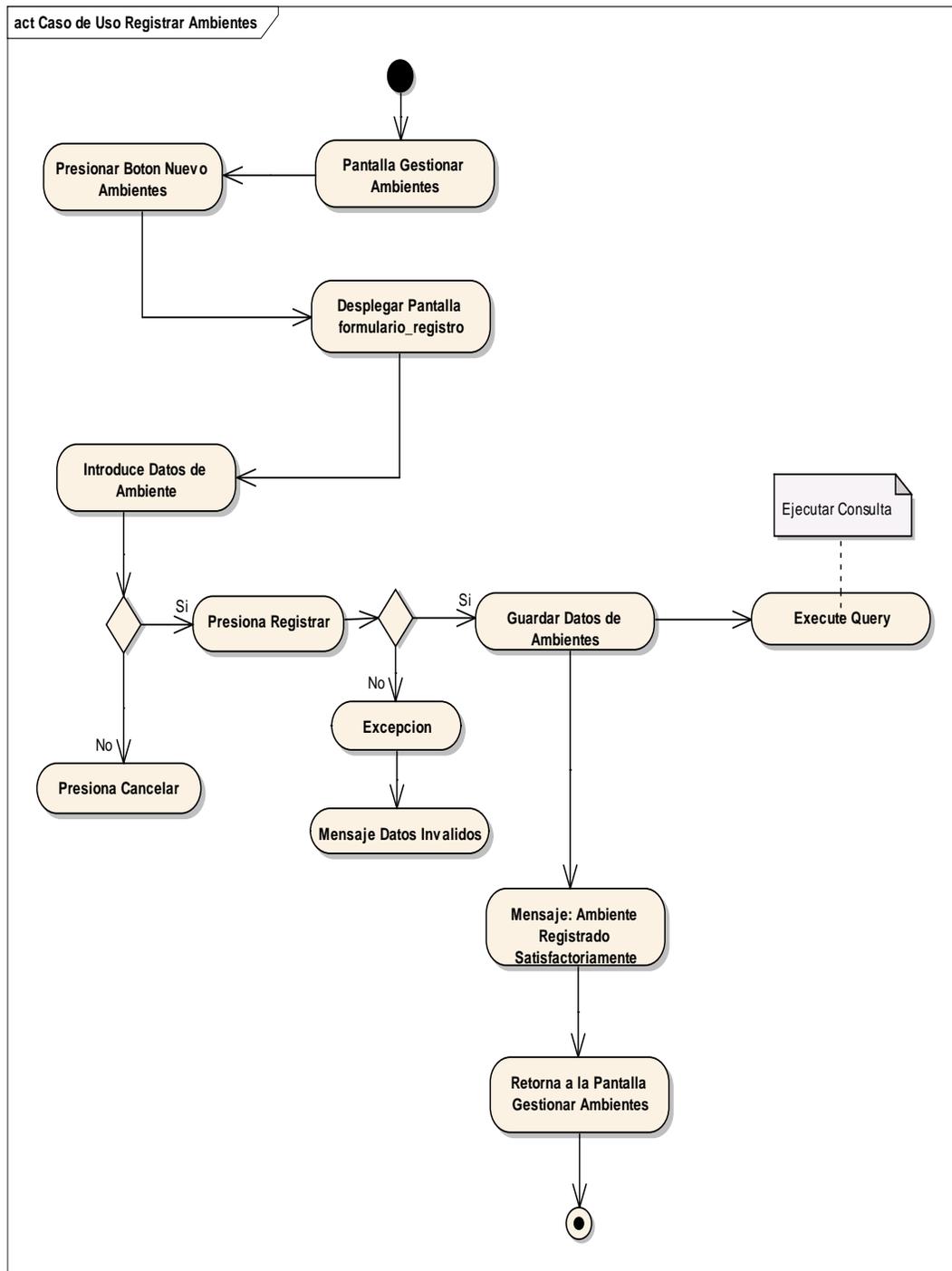


Figura 31 Diagrama de actividad: Caso de uso registrar ambientes

II.1.7.1.3.16. Diagrama de Actividad: Caso de Uso Modificar Ambientes

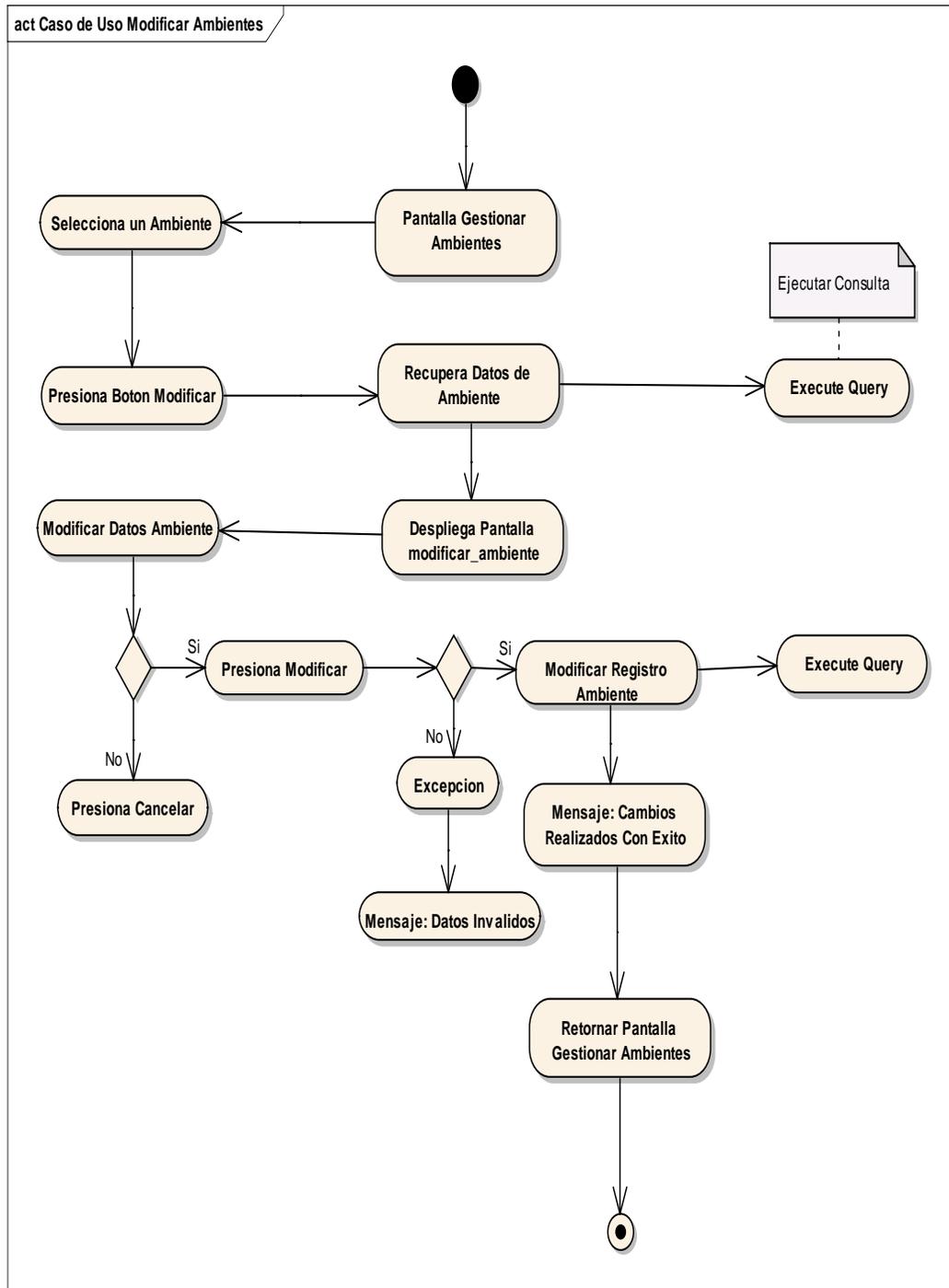


Figura 32 Diagrama de actividad: Caso de uso modificar ambientes

II.1.7.1.3.17. Diagrama de Actividad: Caso de Uso Eliminar Ambientes

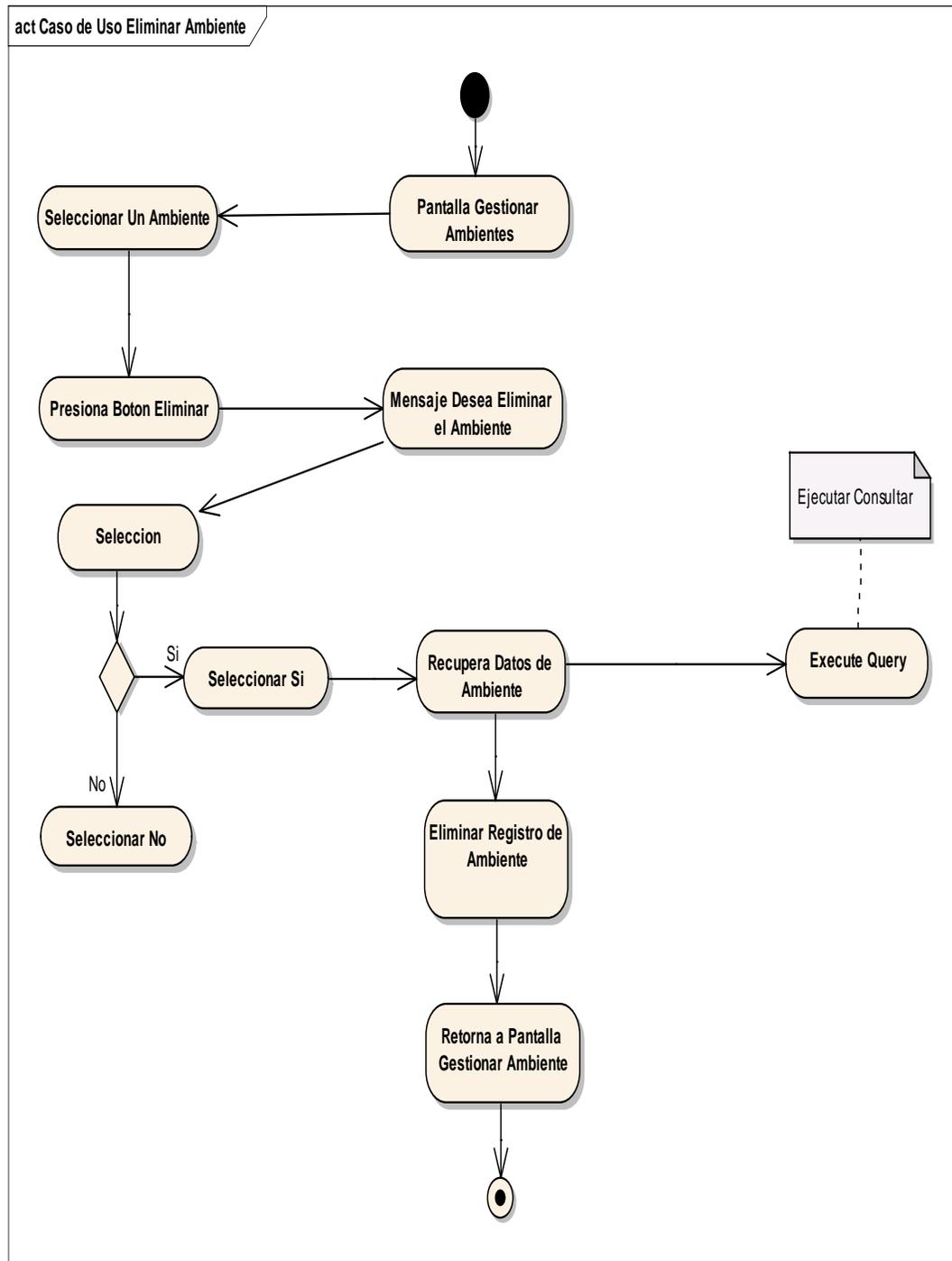


Figura 33 Diagrama de actividad: Caso de uso eliminar ambientes

II.1.7.1.3.18. Diagrama de Actividad: Caso de Uso Gestionar Pisos

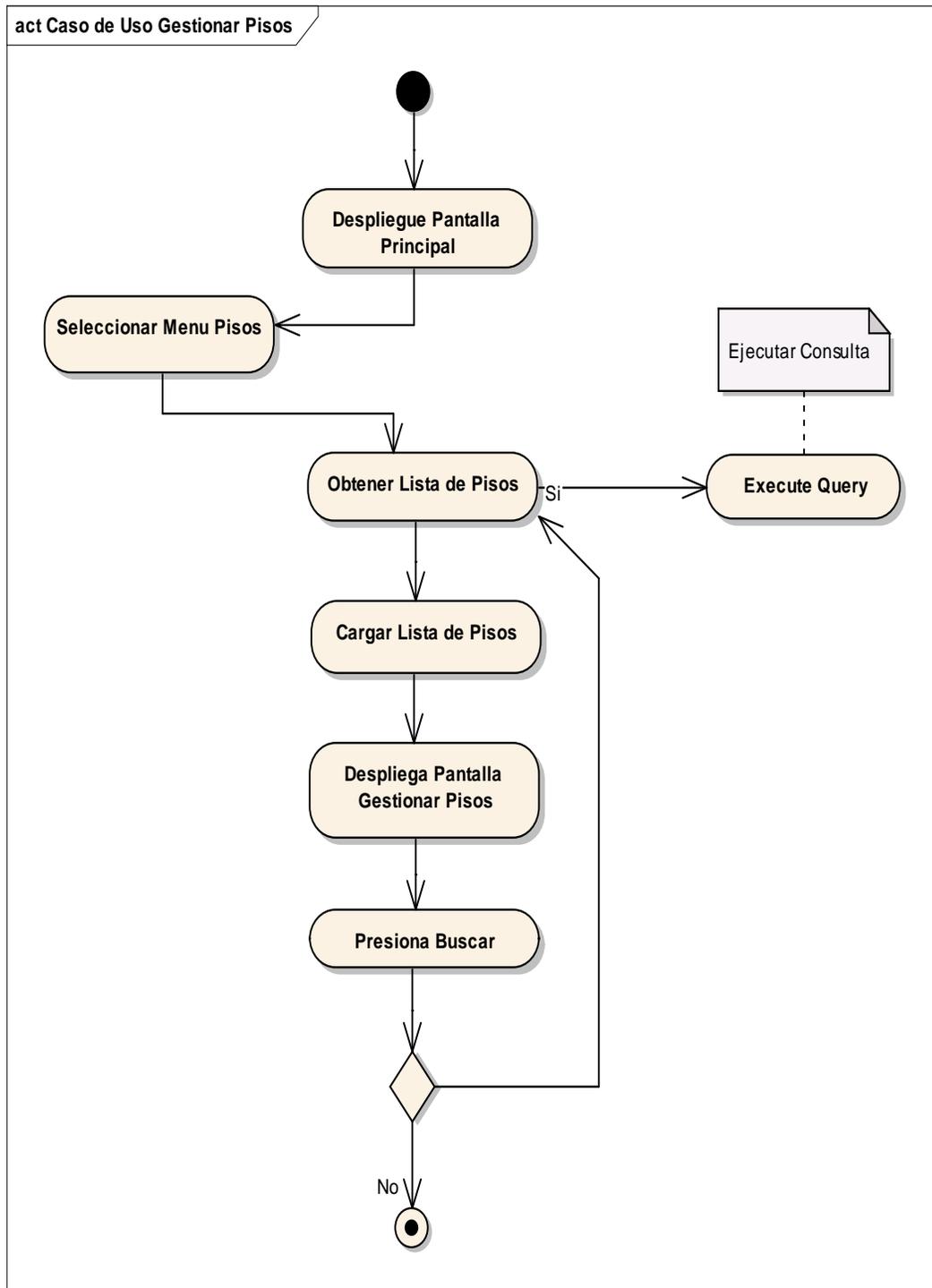


Figura 34 Diagrama de actividad: Caso de uso gestionar pisos

II.1.7.1.3.19. Diagrama de Actividad: Caso de Uso Registrar Pisos

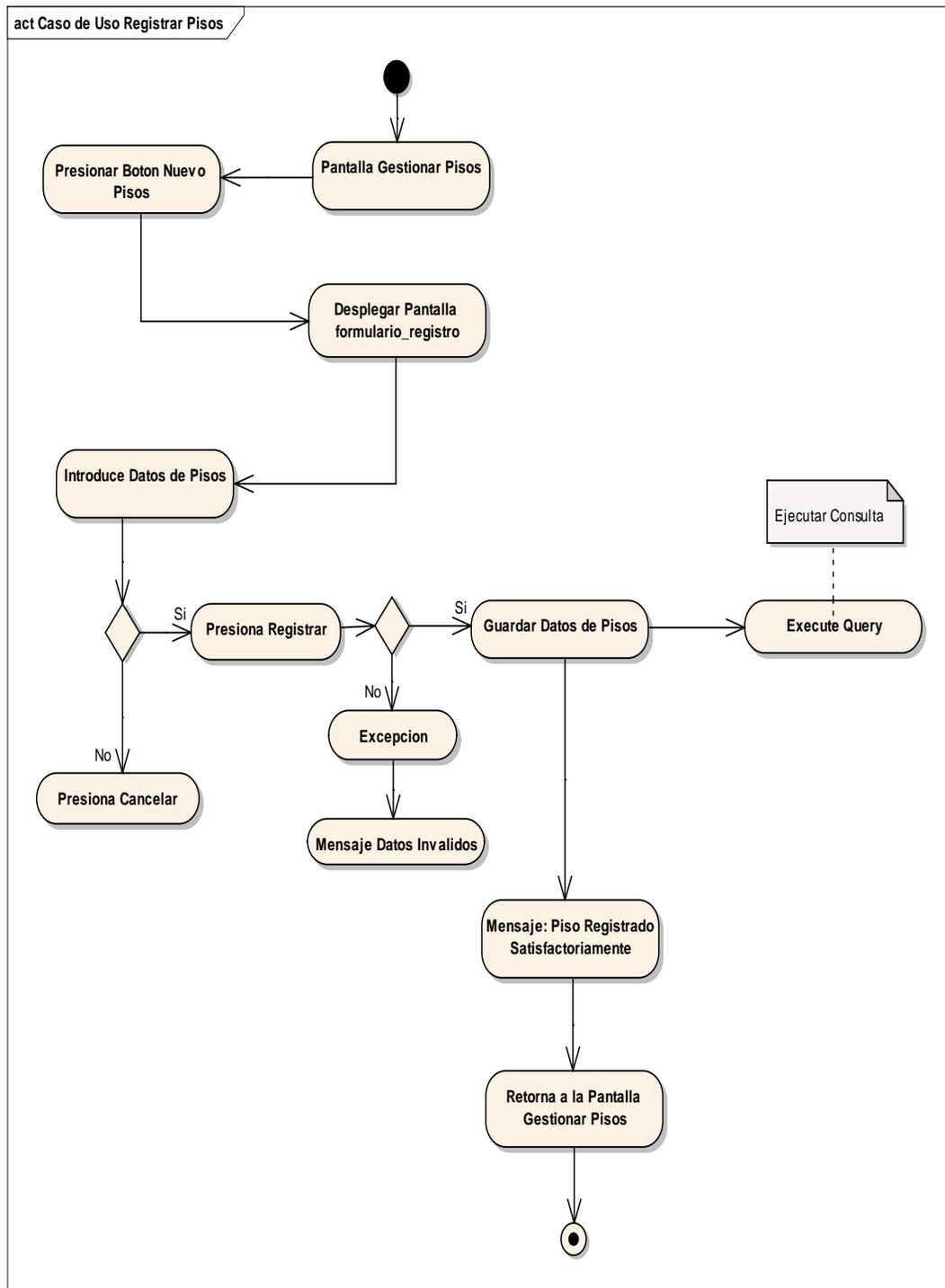


Figura 35 Diagrama de actividad: Caso de uso registrar pisos

II.1.7.1.3.20. Diagrama de Actividad: Caso de Uso Modificar Pisos

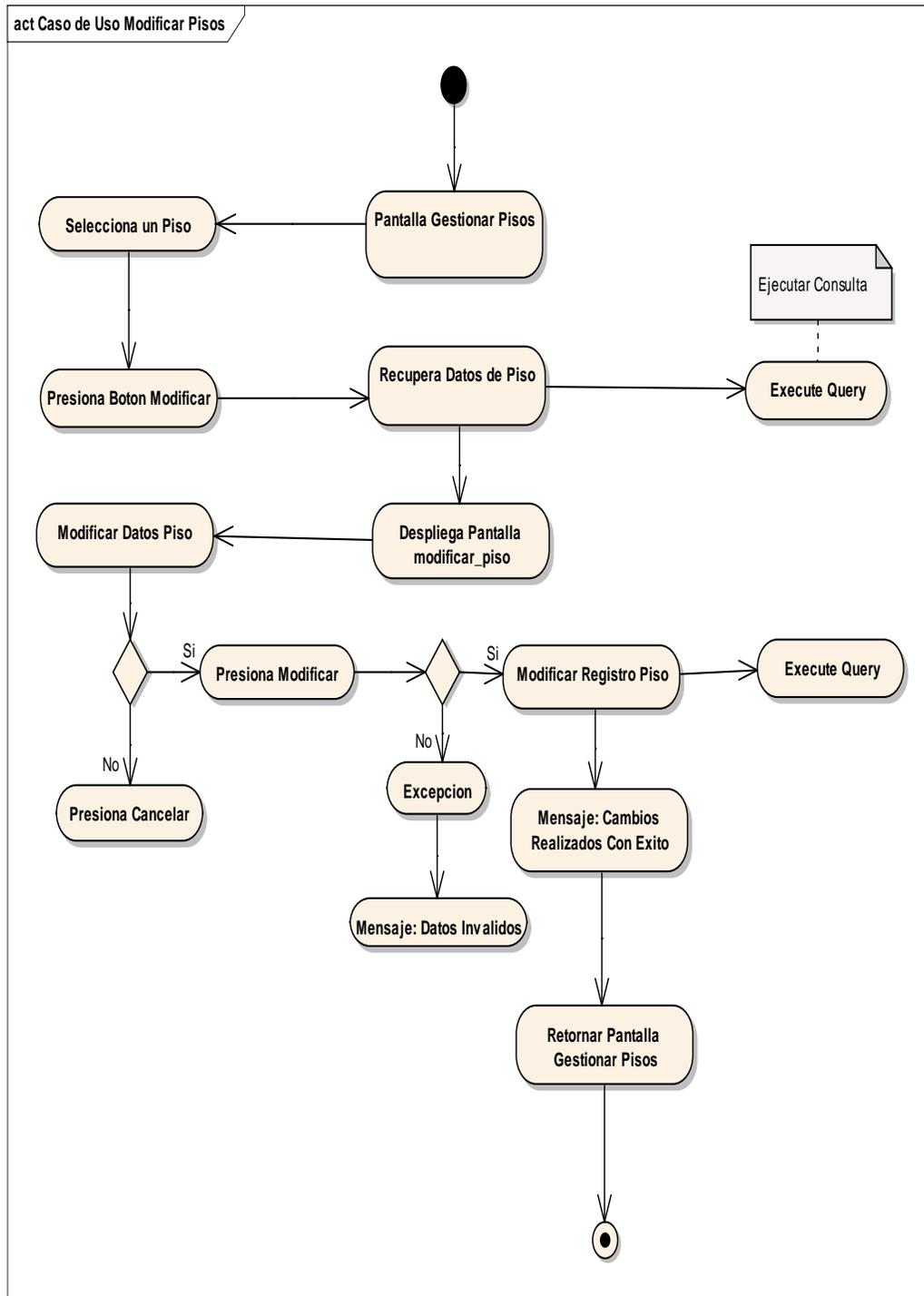


Figura 36 Diagrama de actividad: Caso de uso modificar piso

II.1.7.1.3.21. Diagrama de Actividad: Caso de Uso Eliminar Pisos

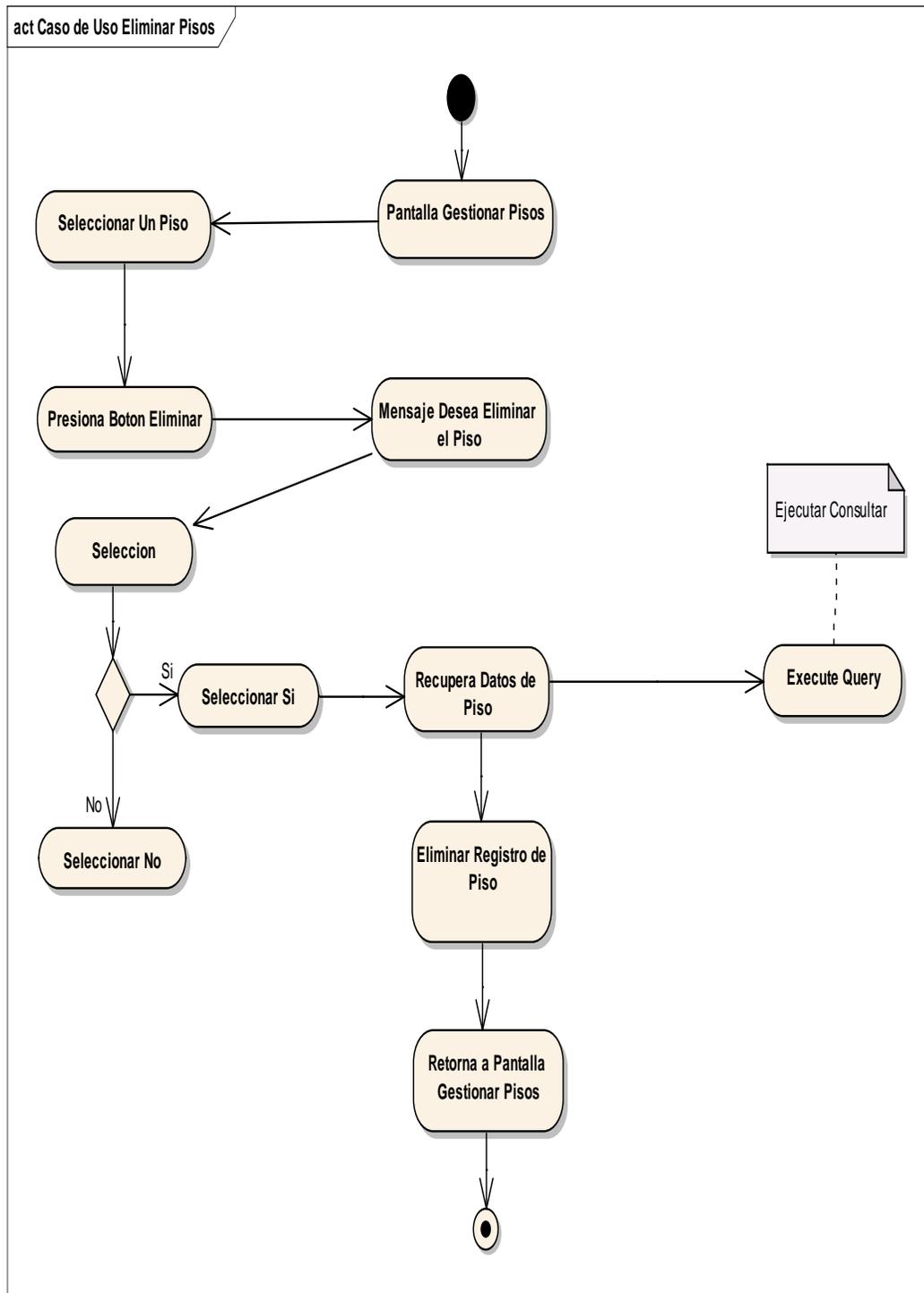


Figura 37 Diagrama de actividad: Caso de uso eliminar pisos

II.1.7.1.3.22. Diagrama de Actividad: Caso de Uso Gestionar Controlador

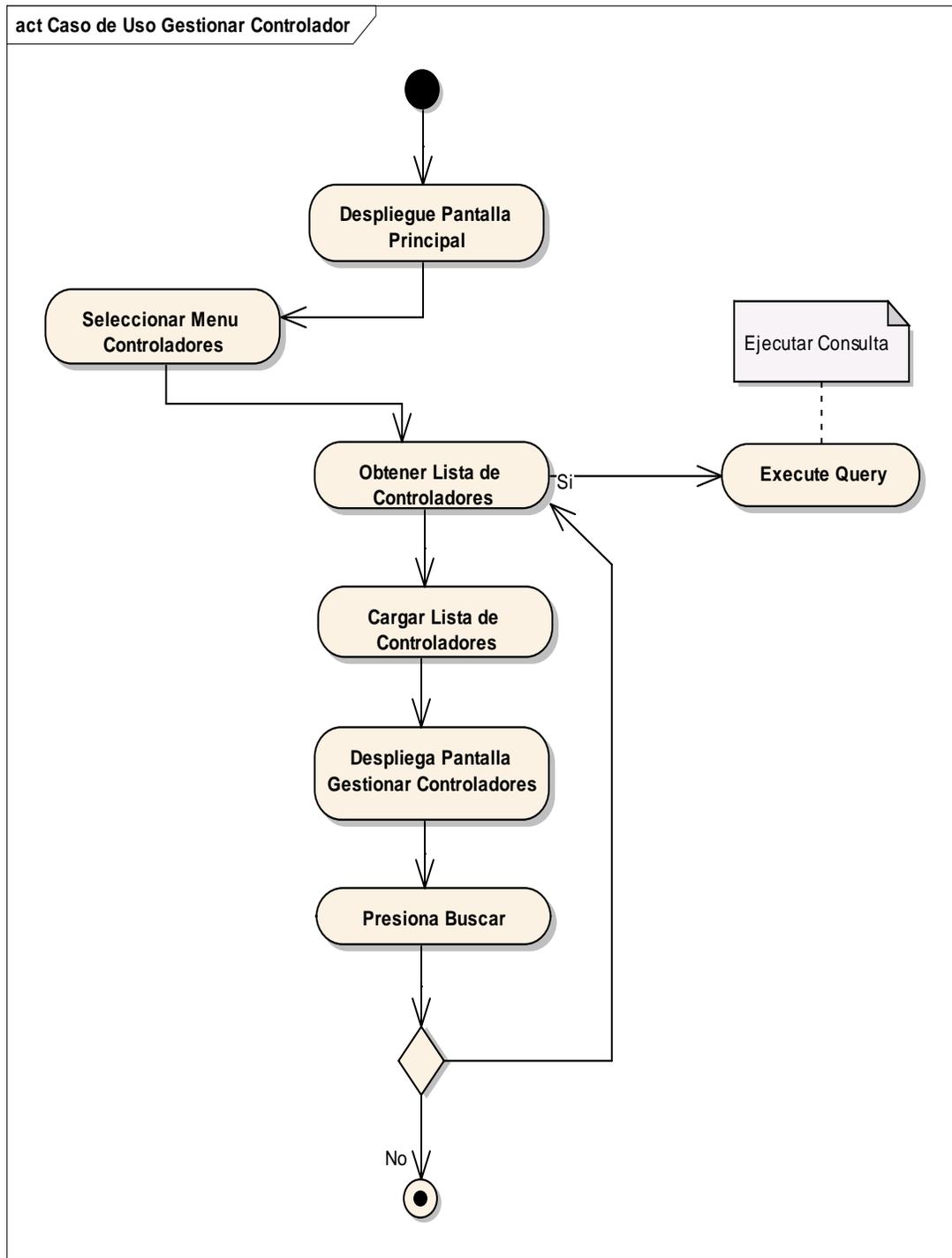


Figura 38 Diagrama de actividad: Caso de uso gestionar controlador

II.1.7.1.3.23. Diagrama de Actividad: Caso de Uso Registrar Controlador

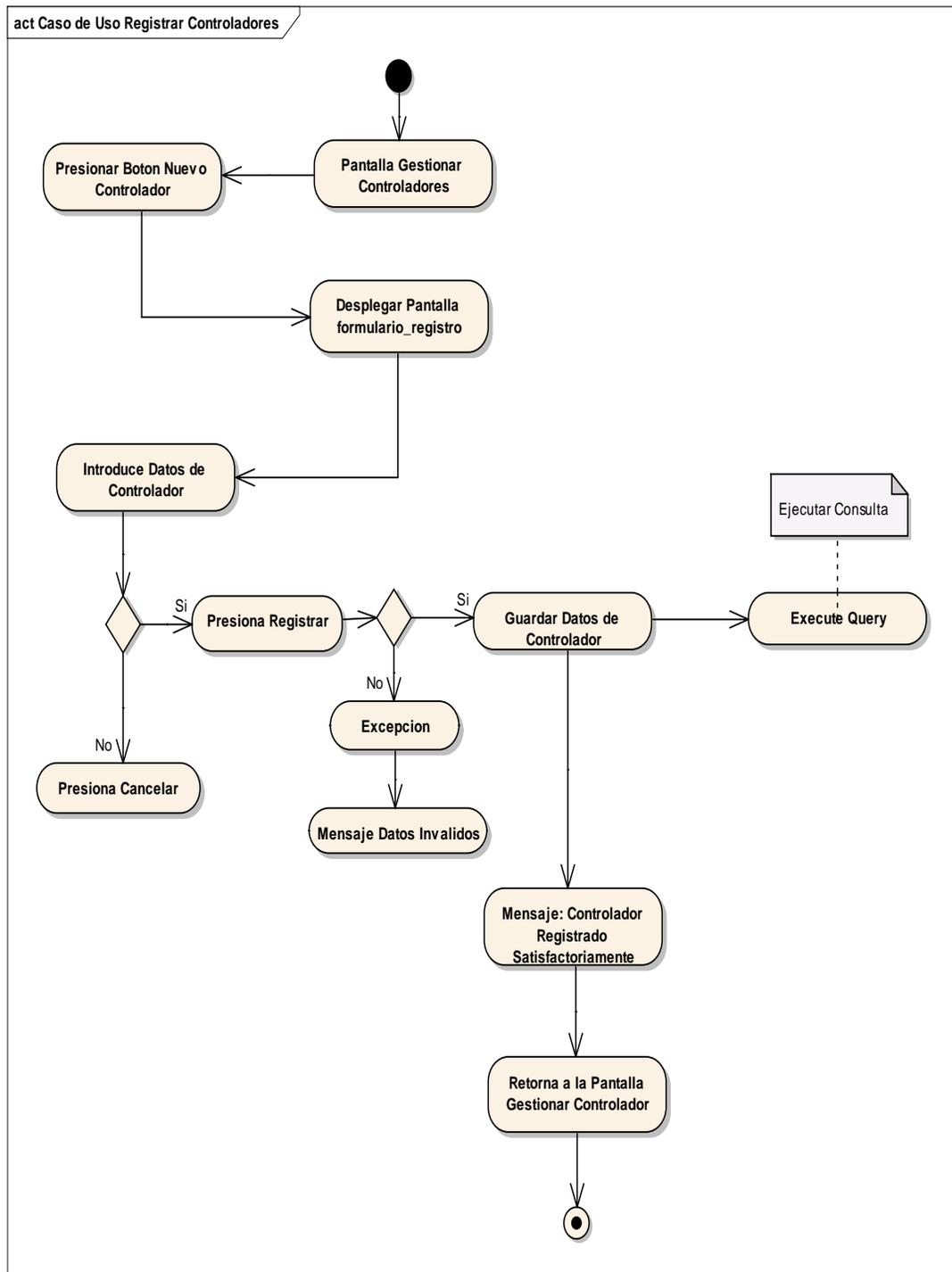


Figura 39 Diagrama de actividad: Caso de uso registrar controlador

II.1.7.1.3.24. Diagrama de Actividad: Caso de Uso Modificar Controlador

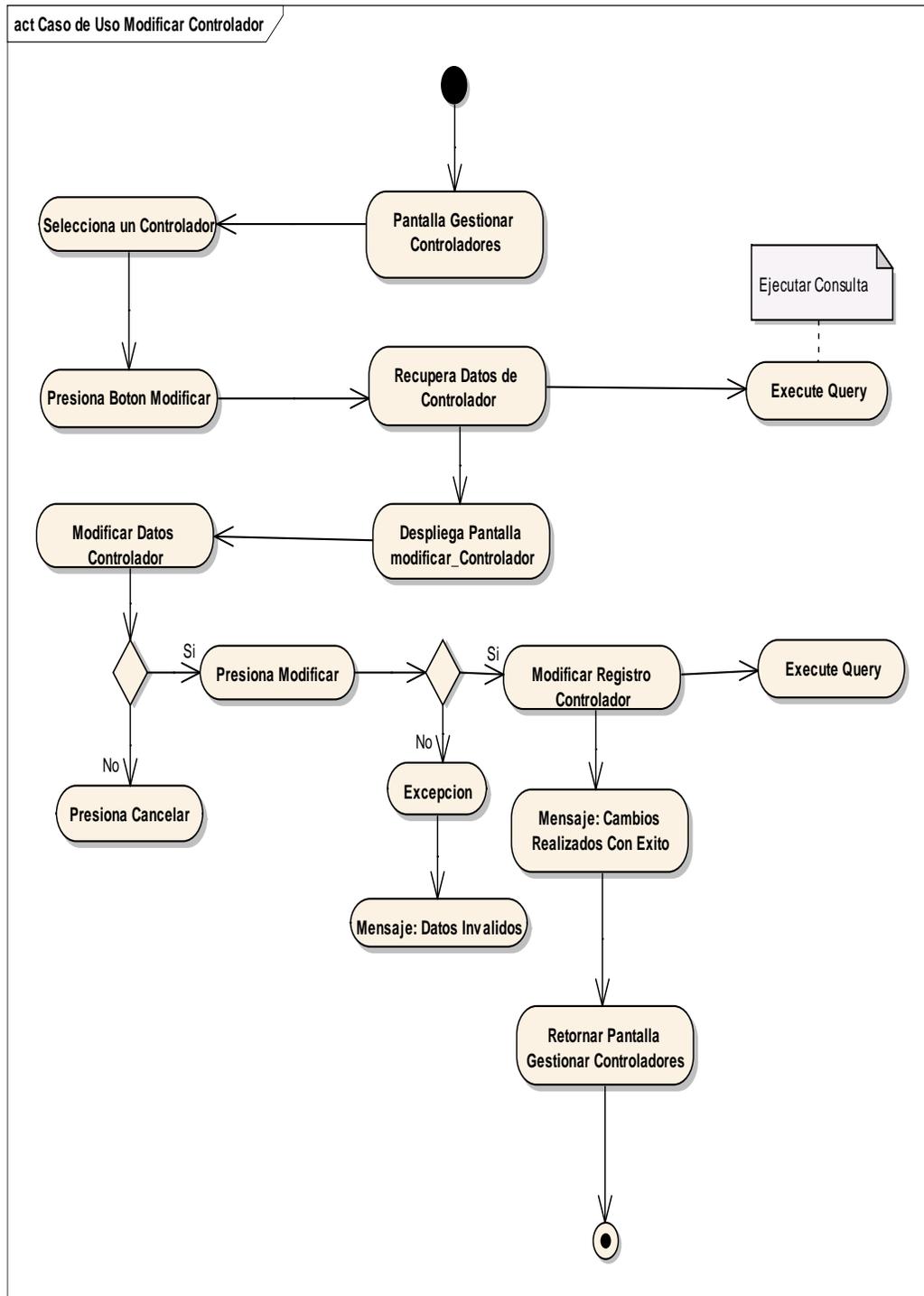


Figura 40 Diagrama de actividad: Caso de uso modificar controlador

II.1.7.1.3.25. Diagrama de Actividad: Caso de Uso Eliminar Controlador

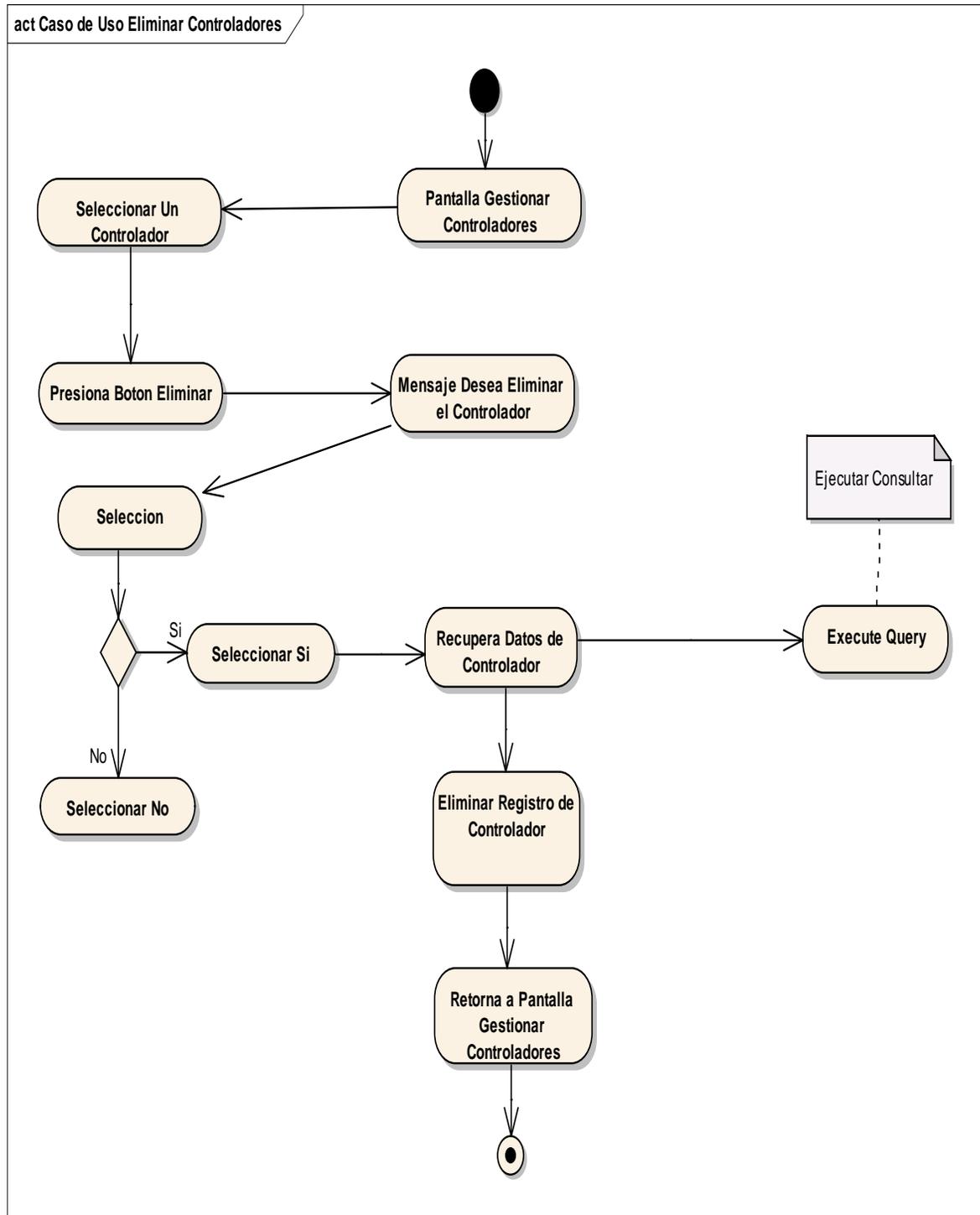


Figura 41 Diagrama de actividad: Caso de uso eliminar controlador

II.1.7.1.3.26. Diagrama de Actividad: Caso de Uso Dispositivo-Ambiente

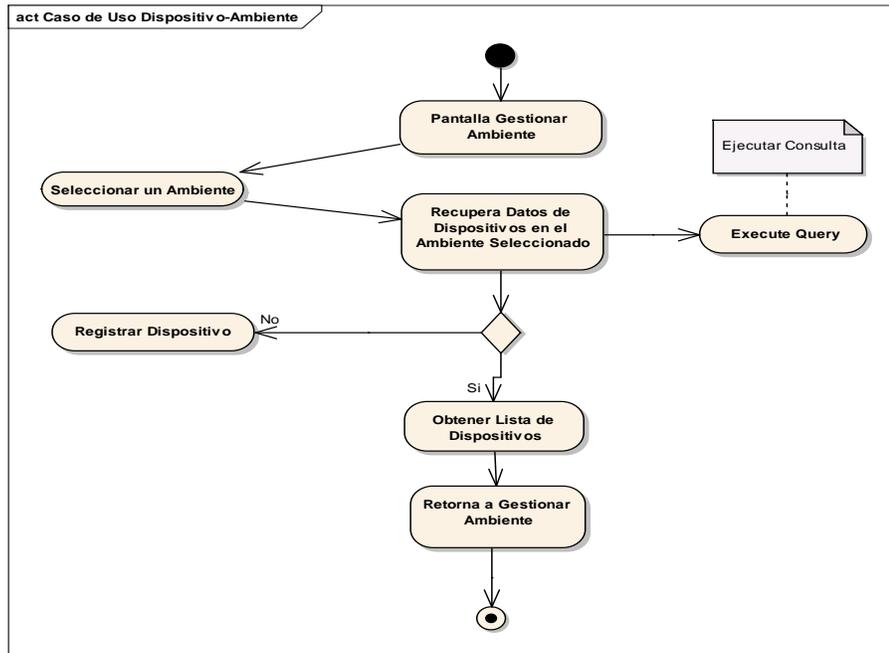


Figura 42 Diagrama de actividad: Caso de uso dispositivo ambiente

II.1.7.1.3.27. Diagrama de Actividad: Caso de Uso Ambiente-Piso

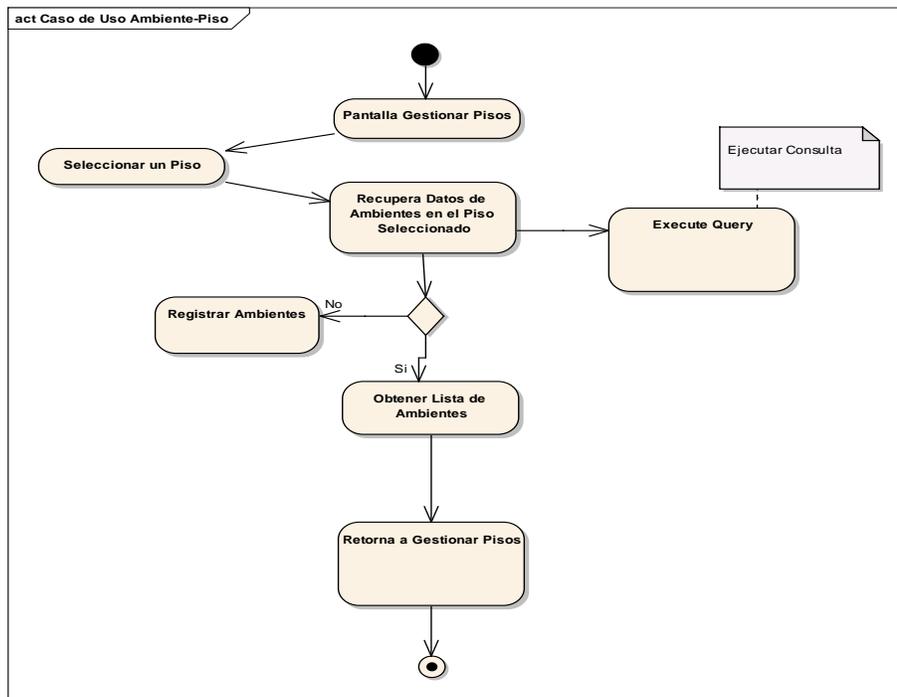


Figura 43 Diagrama de actividad: Caso de uso ambiente piso

II.1.7.1.3.28. Diagrama de Actividad: Caso de Uso Piso-Inmueble

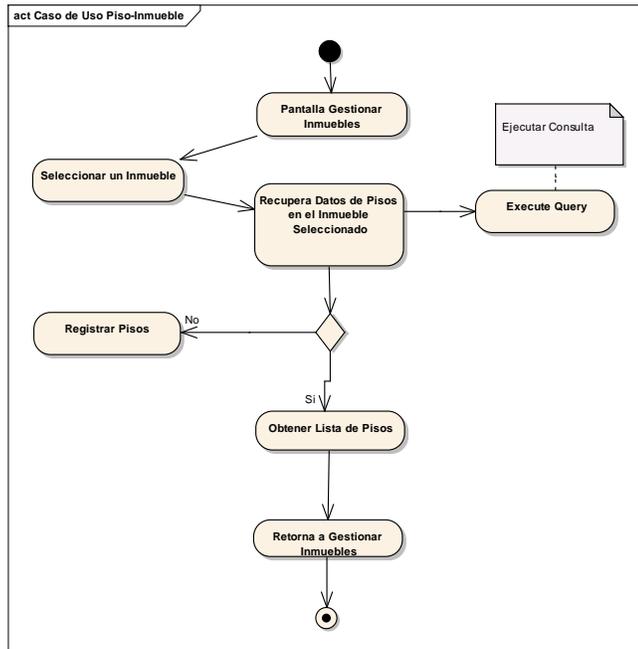


Figura 44 Diagrama de actividad: Caso de uso piso inmueble

II.1.7.1.3.29. Diagrama de Actividad: Caso de Uso Inmueble-Usuario

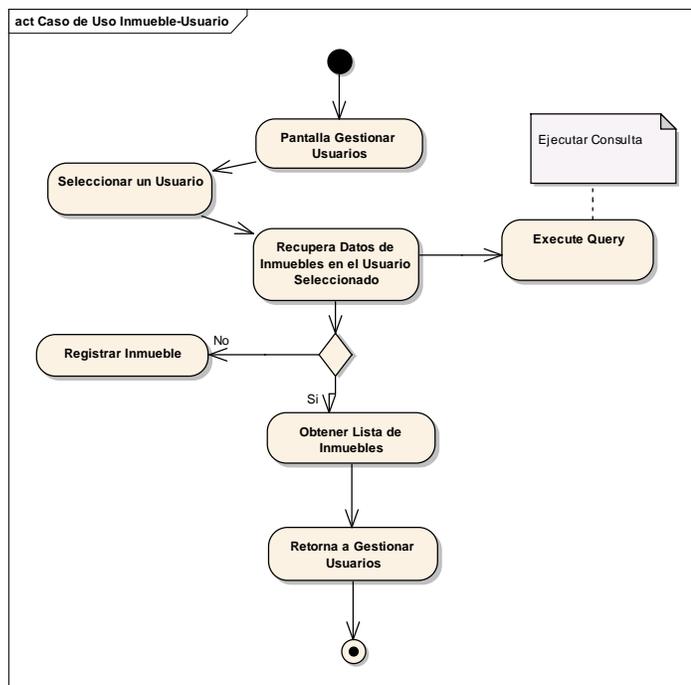


Figura 45 Diagrama de actividad: Caso de uso inmueble usuario

II.1.7.1.3.30. Diagrama de Actividad: Caso de Uso Gestionar Miembros

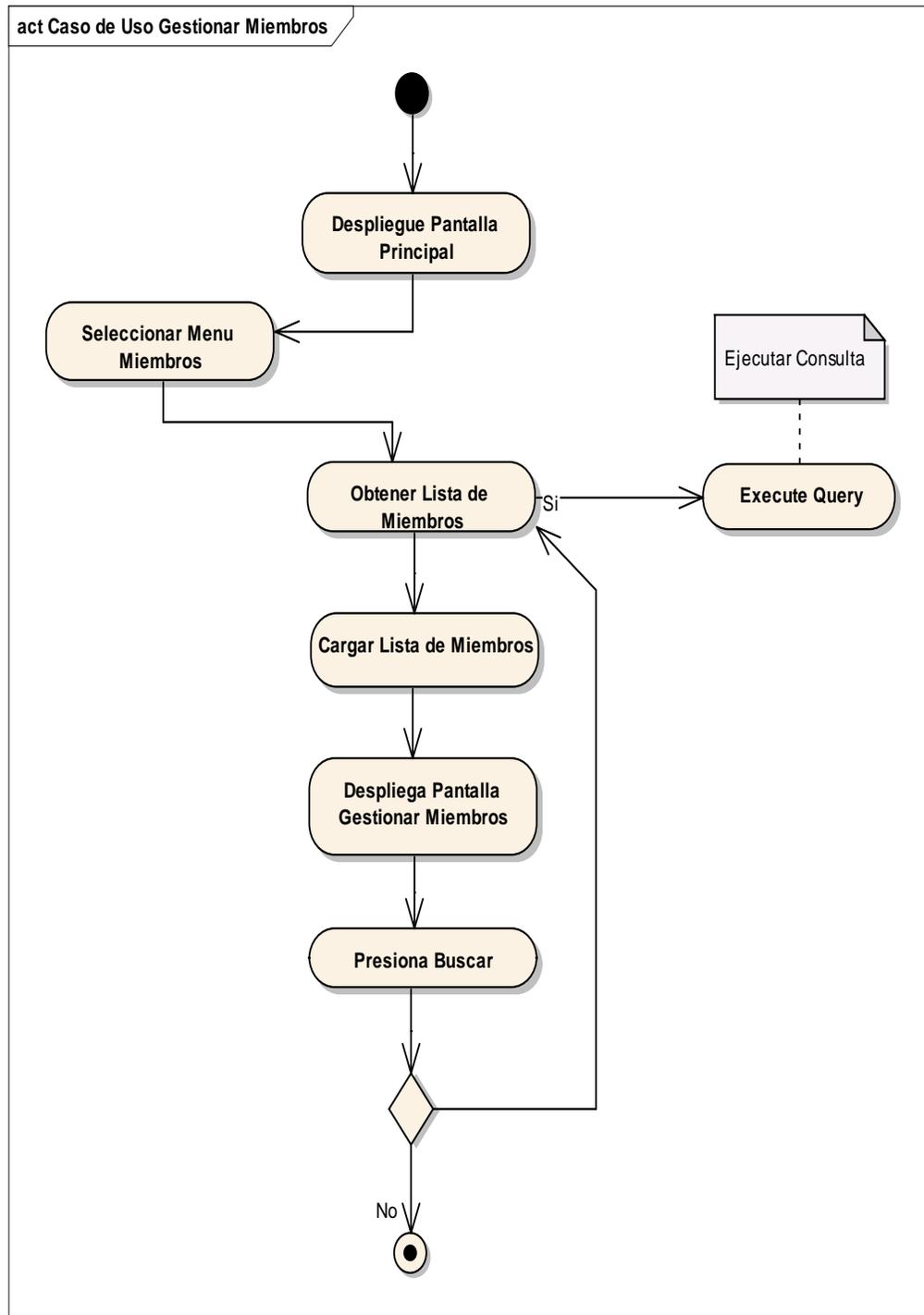


Figura 46 Diagrama de actividad: Caso de uso gestionar miembros

II.1.7.1.3.31. Diagrama de Actividad: Caso de Uso Registrar Miembros

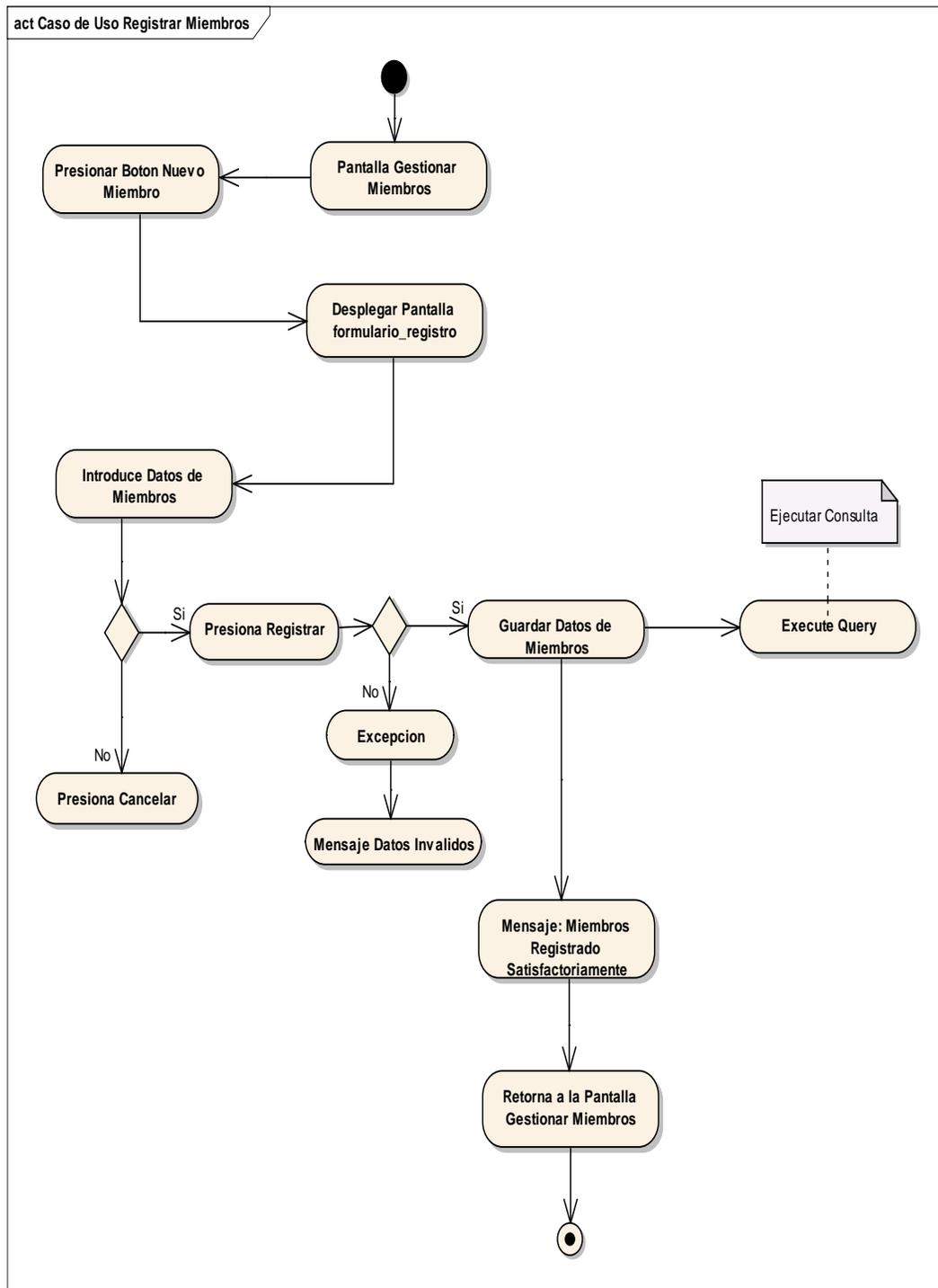


Figura 47 Diagrama de actividad: Caso de uso registrar miembros

II.1.7.1.3.32. Diagrama de Actividad: Caso de Uso Modificar Miembros

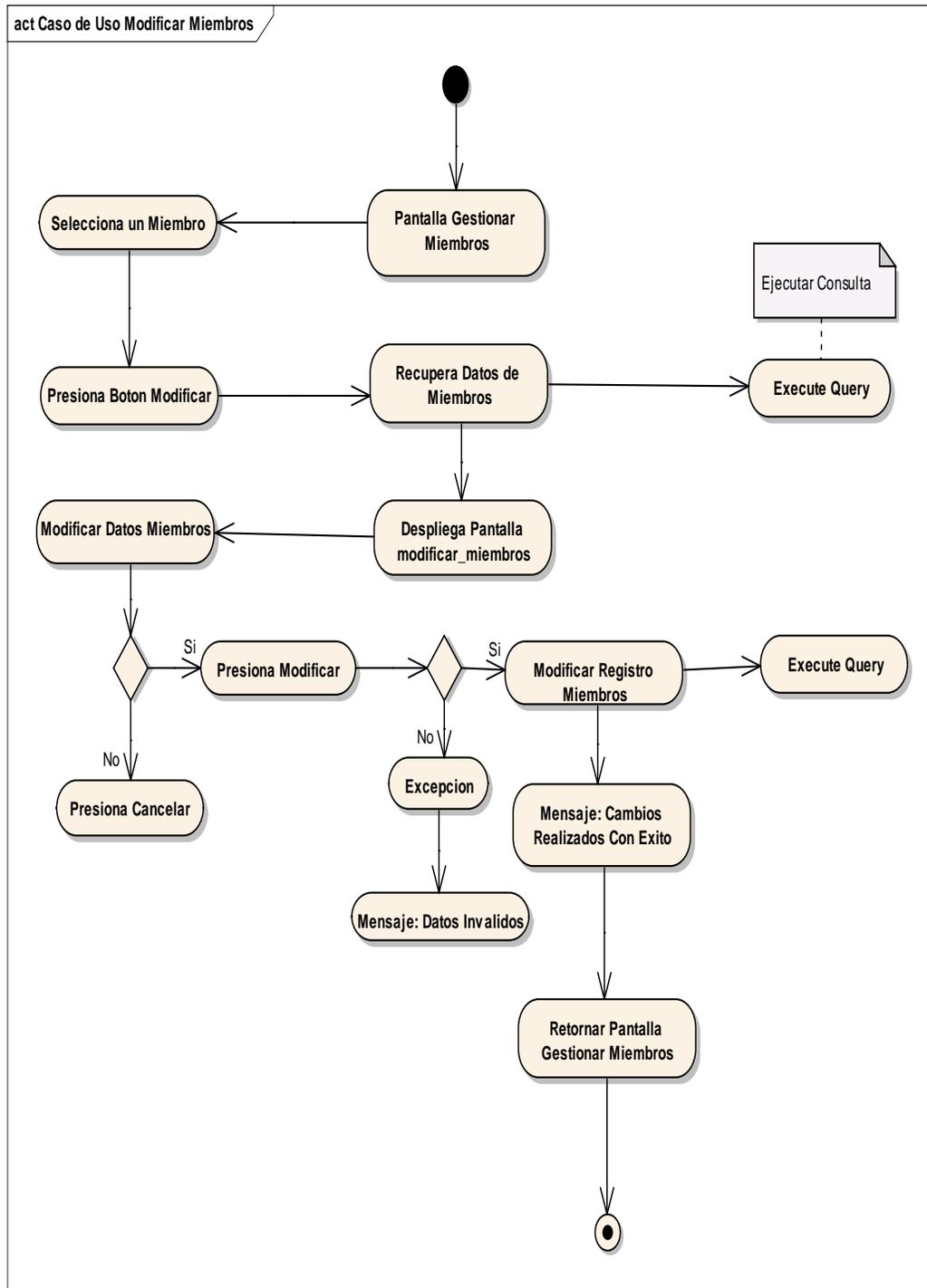


Figura 48 Diagrama de actividad: Caso de uso modificar miembros

II.1.7.1.3.33. Diagrama de Actividad: Caso de Uso Eliminar Miembros

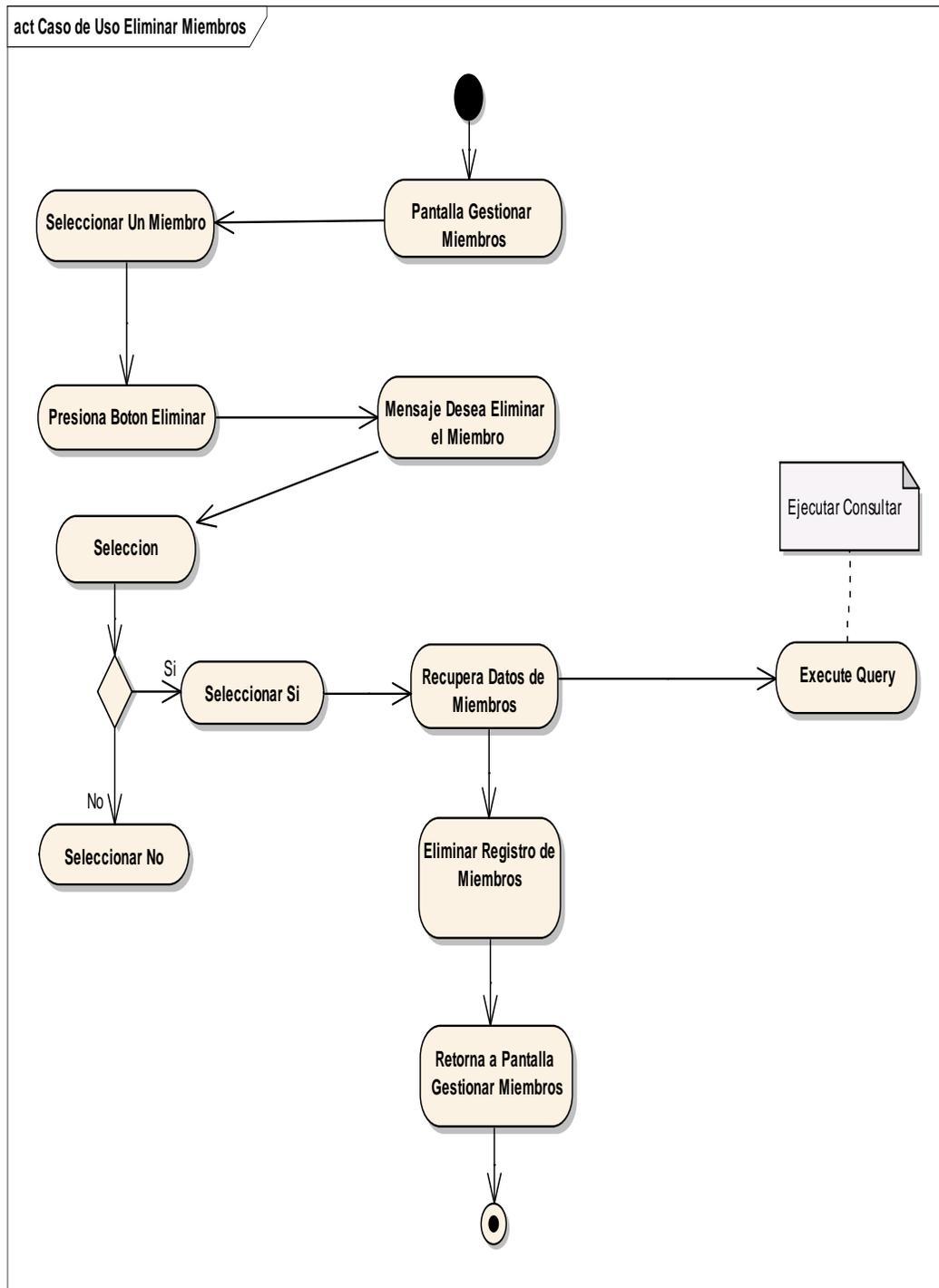
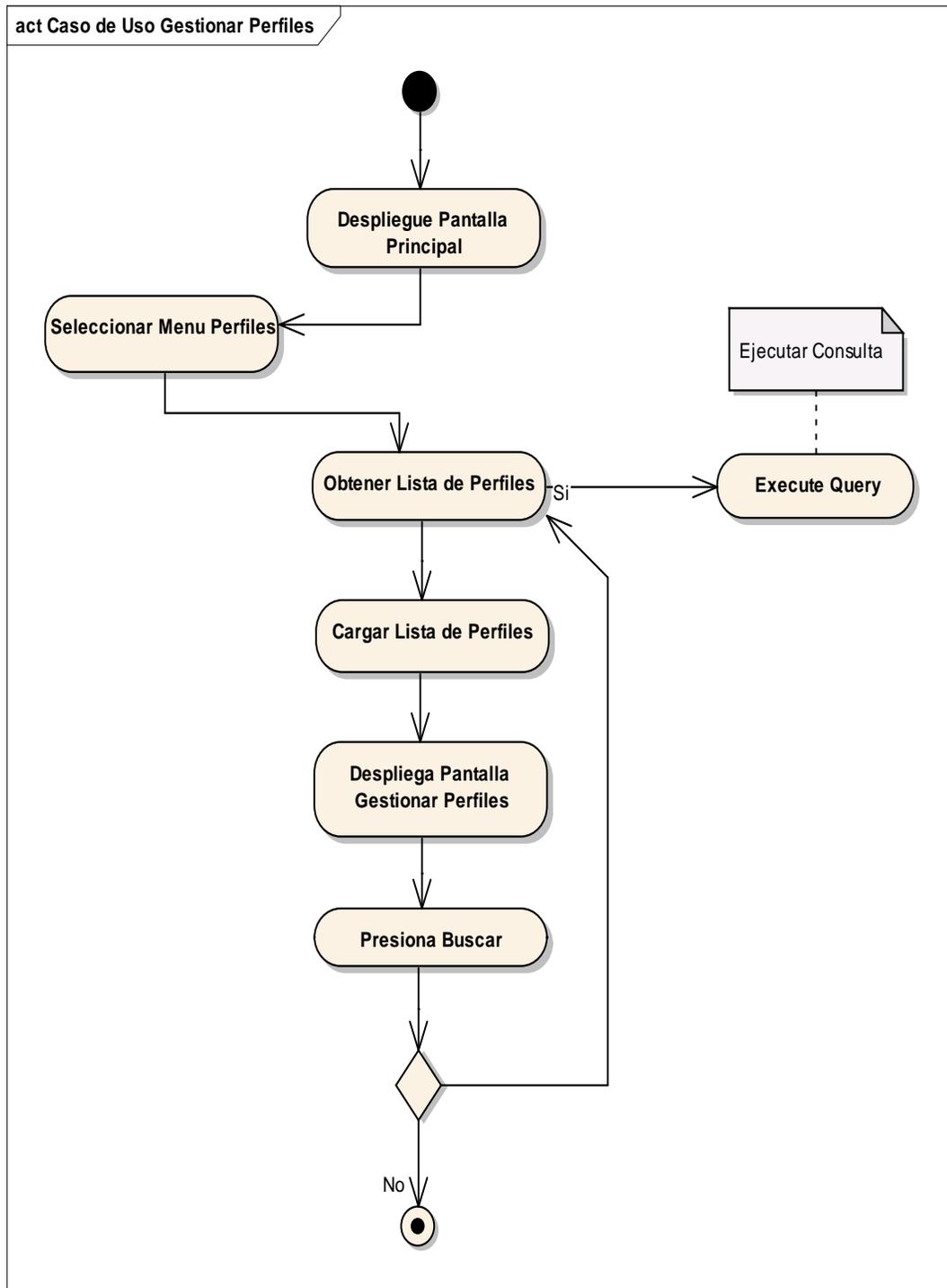


Figura 49 Diagrama de actividad: Caso de uso eliminar miembros

II.1.7.1.3.34. Diagrama de Actividad: Caso de Uso Gestionar Perfiles**Figura 50 Diagrama de actividad: Caso de uso gestionar perfiles**

II.1.7.1.3.35. Diagrama de Actividad: Caso de Uso Registrar Perfiles

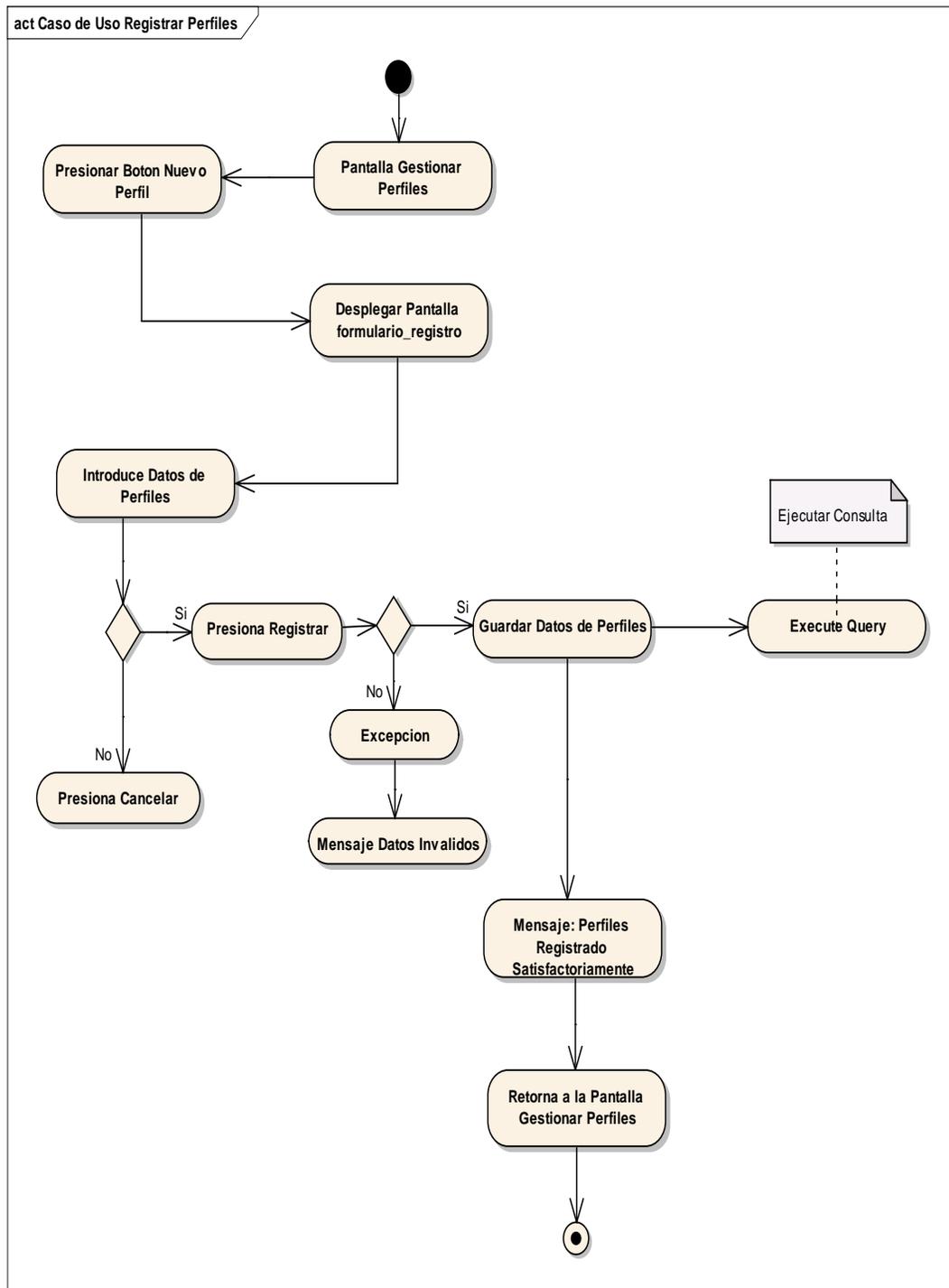


Figura 51 Diagrama de actividad: Caso de uso registrar perfiles

II.1.7.1.3.36. Diagrama de Actividad: Caso de Uso Modificar Perfiles

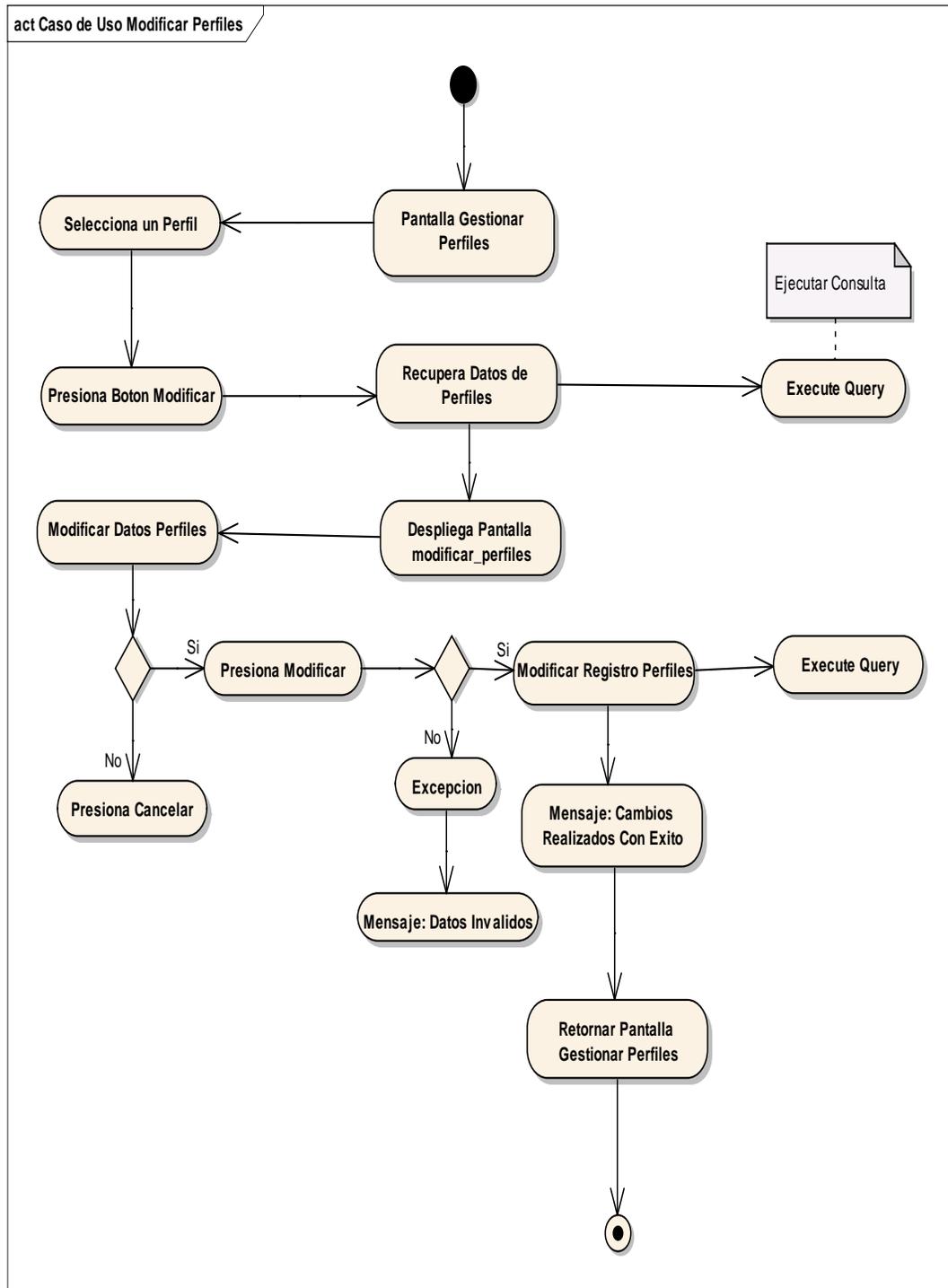


Figura 52 Diagrama de actividad: Caso de uso modificar perfiles

II.1.7.1.3.37. Diagrama de Actividad: Caso de Uso Eliminar Perfiles

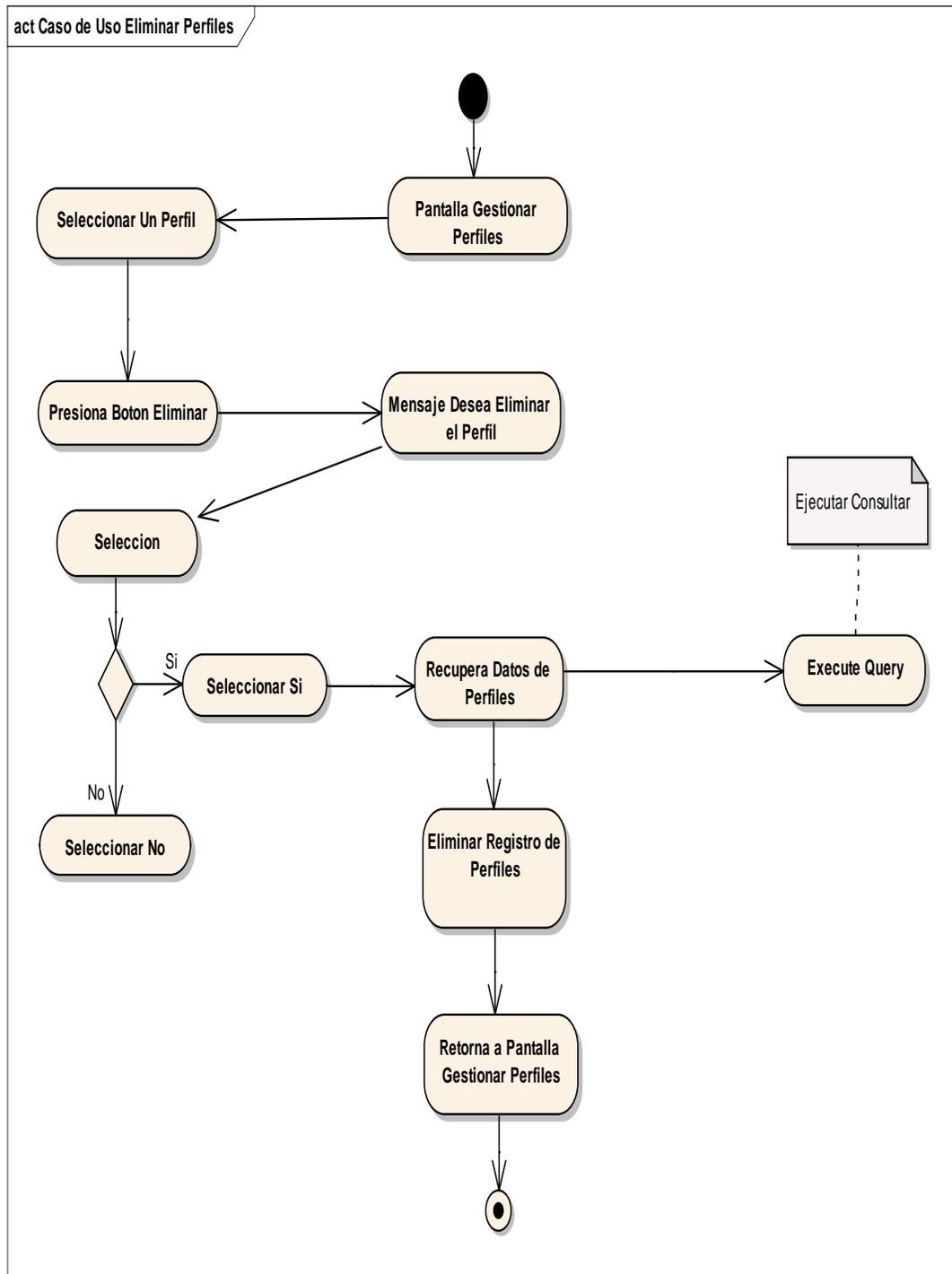


Figura 53 Diagrama de actividad: Caso de uso eliminar perfiles

II.1.7.1.3.38. Diagrama de Actividad: Caso de Uso Gestionar Manejo de Dispositivos

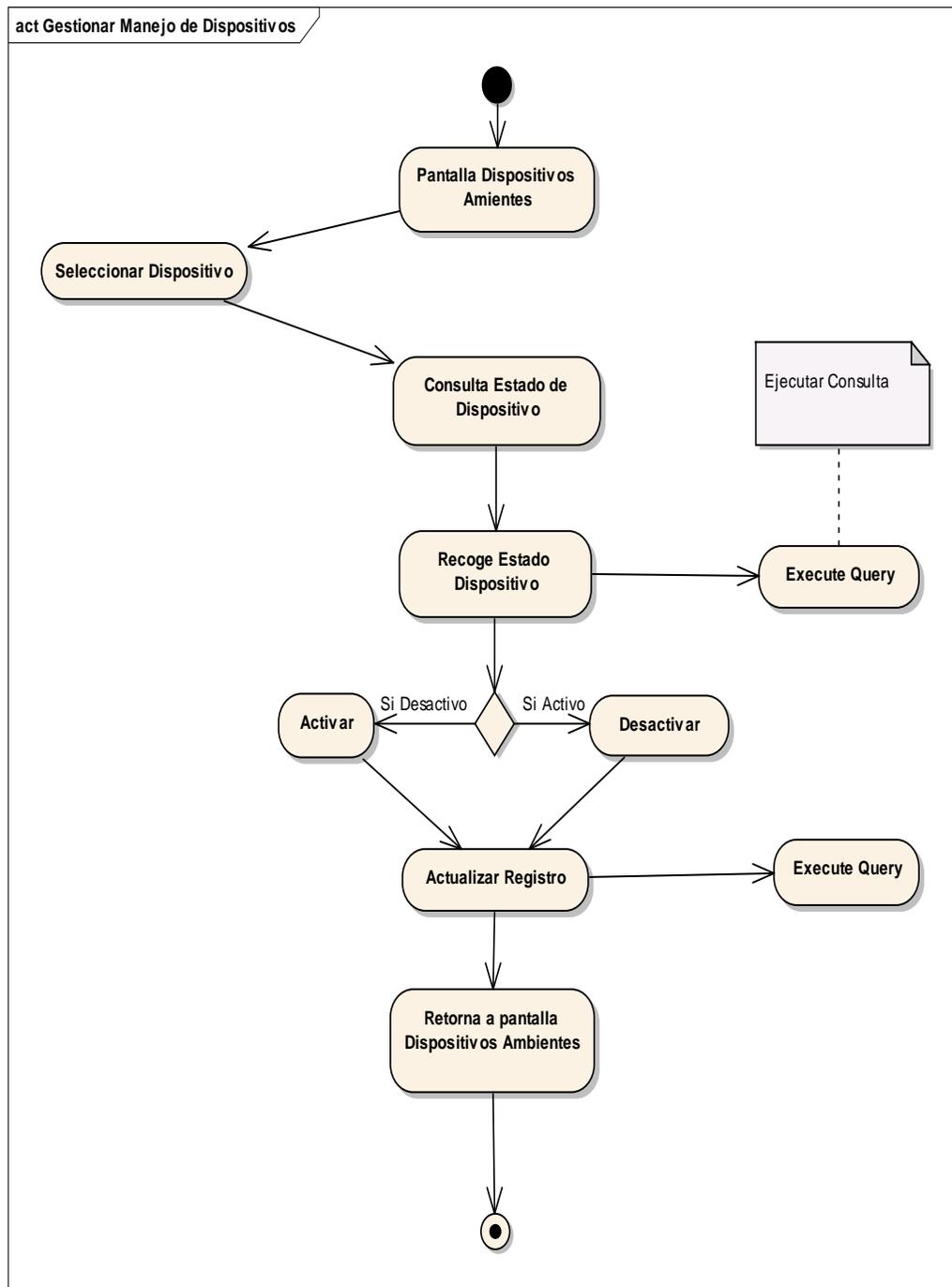


Figura 54 Diagrama de actividad: Caso de uso manejo de dispositivos

II.1.7.1.3.39. Diagrama de Actividad: Caso de Uso Gestionar Reportes

II.1.8. Modelado de Diagramas de Secuencia

II.1.8.1. Introducción

En un diagrama de secuencia se indicarán los módulos o clases que forman parte del programa y las llamadas que se hacen en cada uno de ellos para realizar una tarea determinada.

Se realizan diagramas de secuencia para definir acciones que se pueden realizar en la aplicación en cuestión.

El detalle que se muestre en el diagrama de secuencia debe estar en consonancia con lo que se intenta mostrar o bien con la fase de desarrollo en la que esté el proyecto, no es lo mismo un diagrama de secuencia que muestre la acción diferente a otra que es realmente.

II.1.8.1.1. Propósito

- Comprender la dinámica, sistemas deseados para la organización.
- Identificar clases de análisis y diseño.
- Poder demostrar la secuencia en que se procesan los métodos.

II.1.8.1.2. Alcance

- Describir la dinámica del sistema en el tiempo de vida de las clases u objetos.
- Definir un diagrama de secuencia para cada caso de uso.
- Definir la función de cada método.

II.1.8.1.3. Diagrama de Secuencia

II.1.8.1.3.1. Diagrama de Secuencia: Caso de Uso Ingresar al Sistema

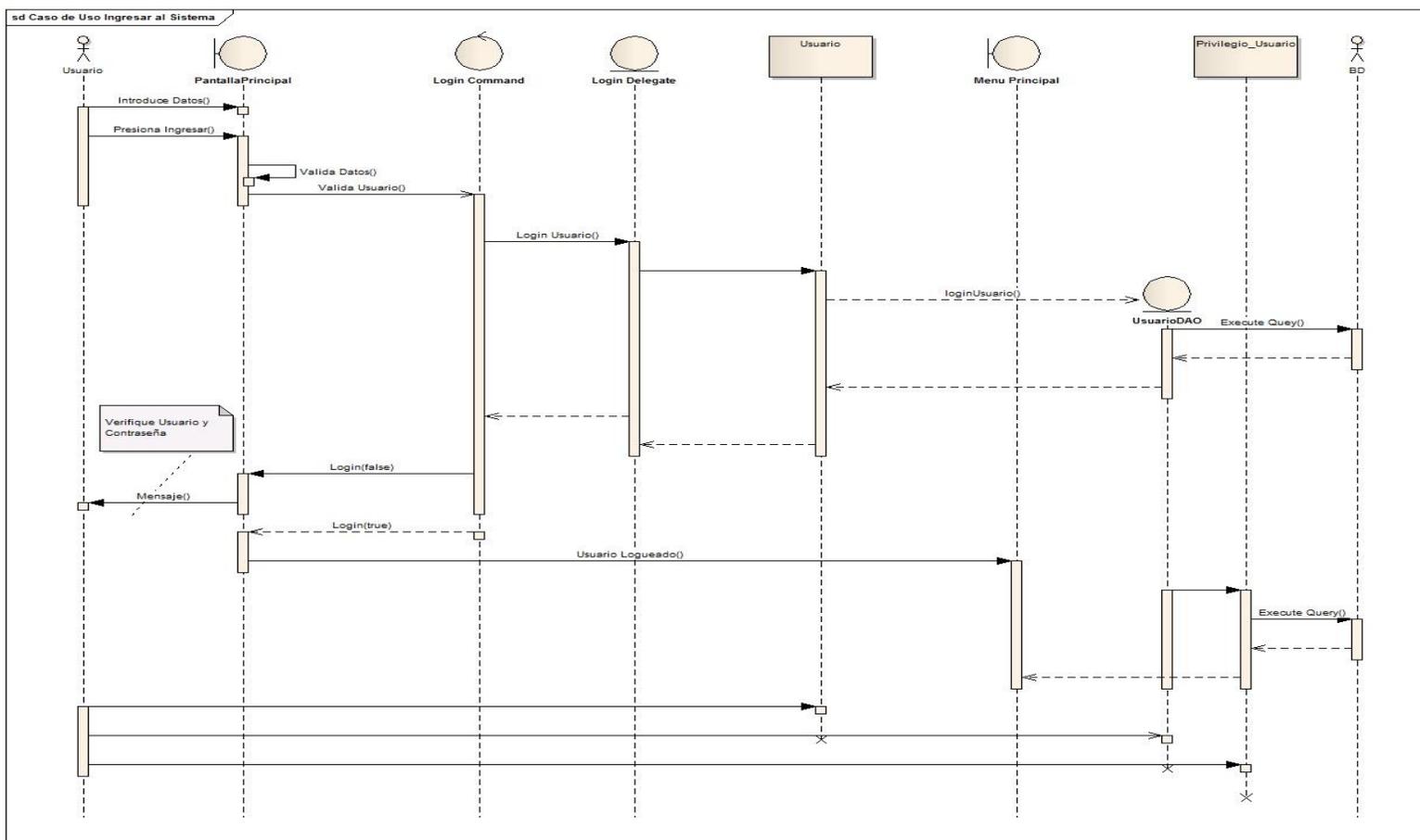


Figura 55 Diagrama de secuencia: caso de uso ingresar al sistema

II.1.8.1.3.2. Diagrama de Secuencia: Caso de Uso Gestionar Usuarios

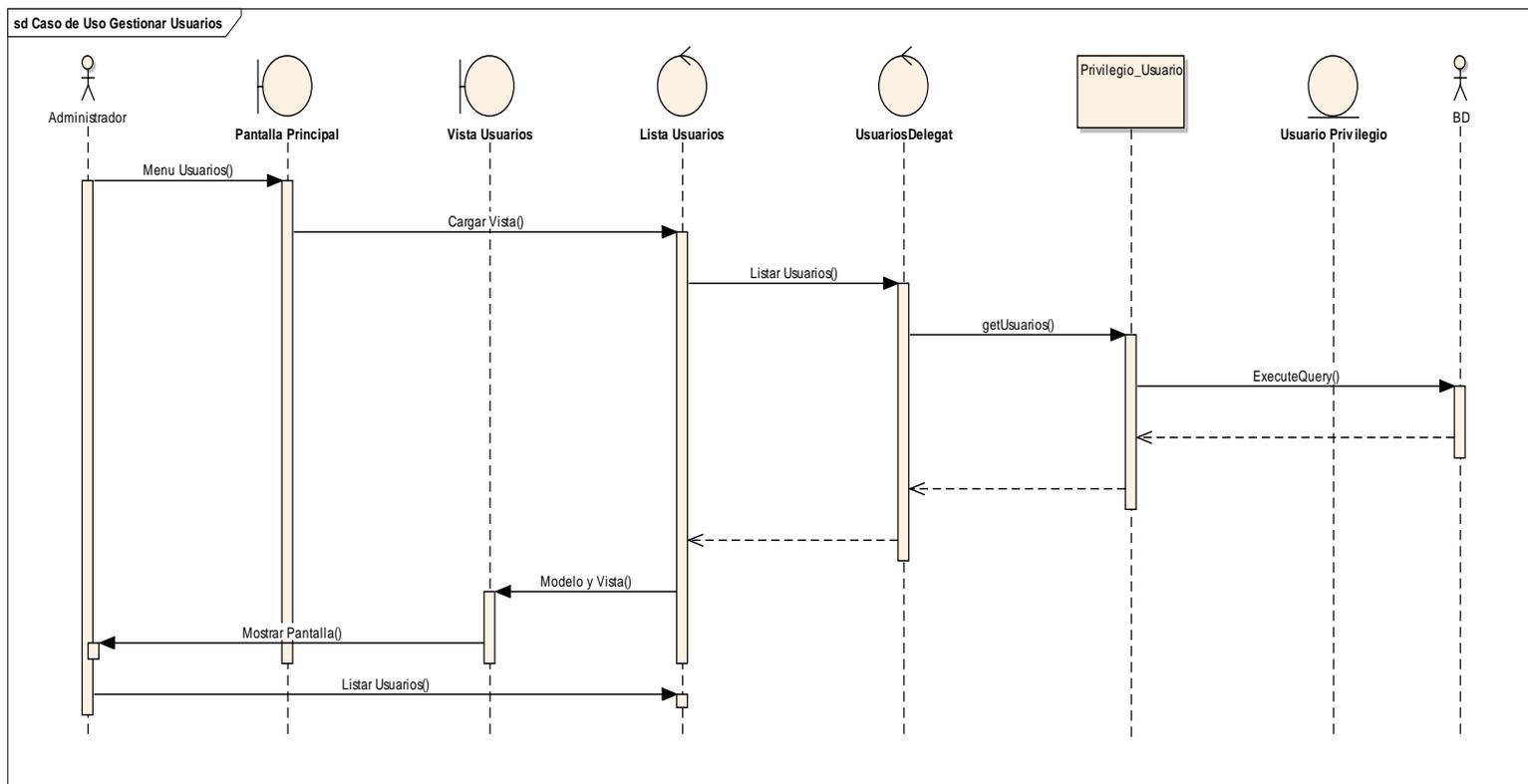


Figura 56 Diagrama de secuencia: caso de uso gestionar usuarios

II.1.8.1.3.3. Diagrama de Secuencia: Caso de Uso Registrar Usuario

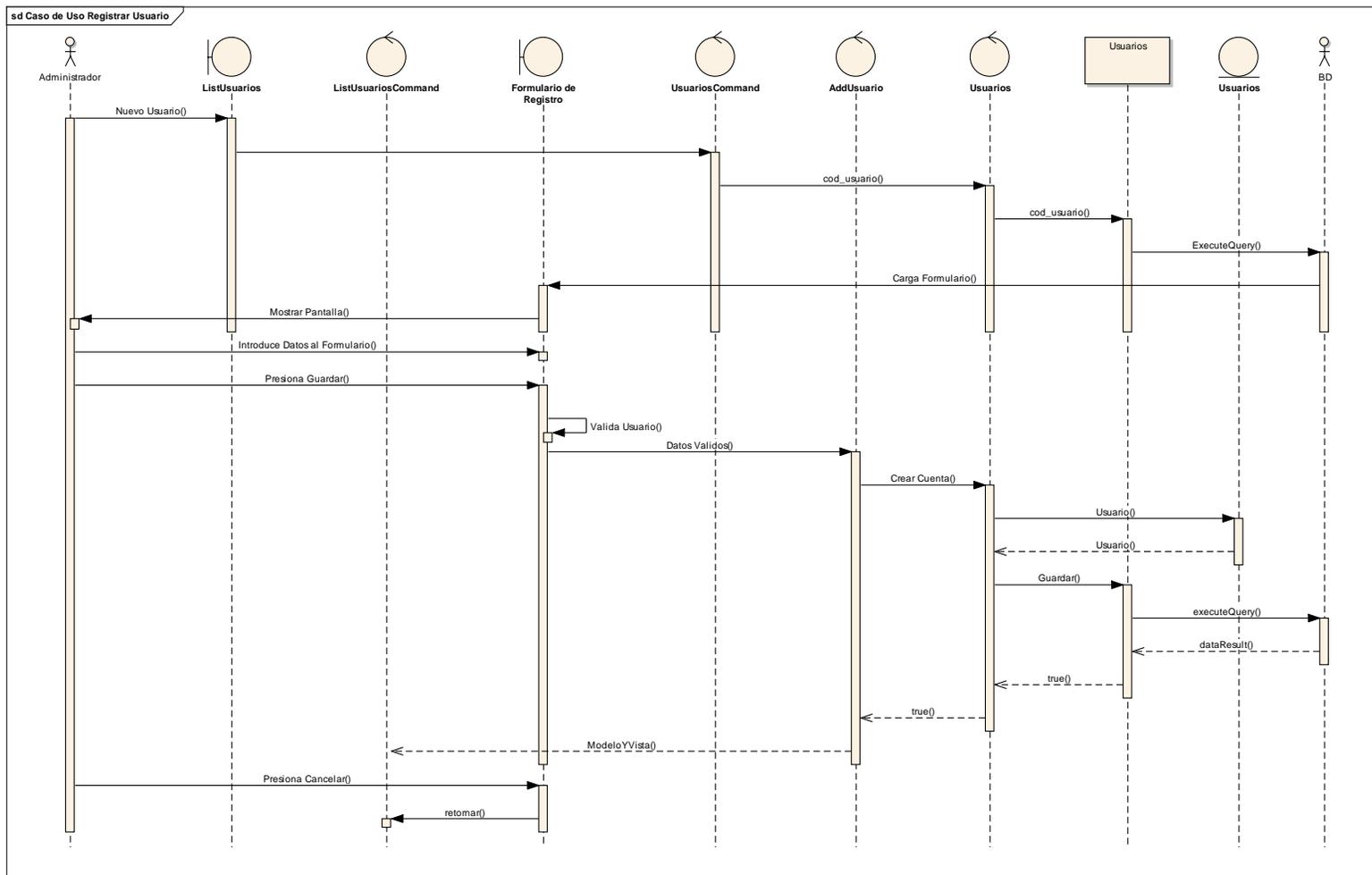


Figura 57 Diagrama de secuencia: caso de uso registrar usuario

II.1.8.1.3.4. Diagrama de Secuencia: Caso de Uso Modificar Usuarios

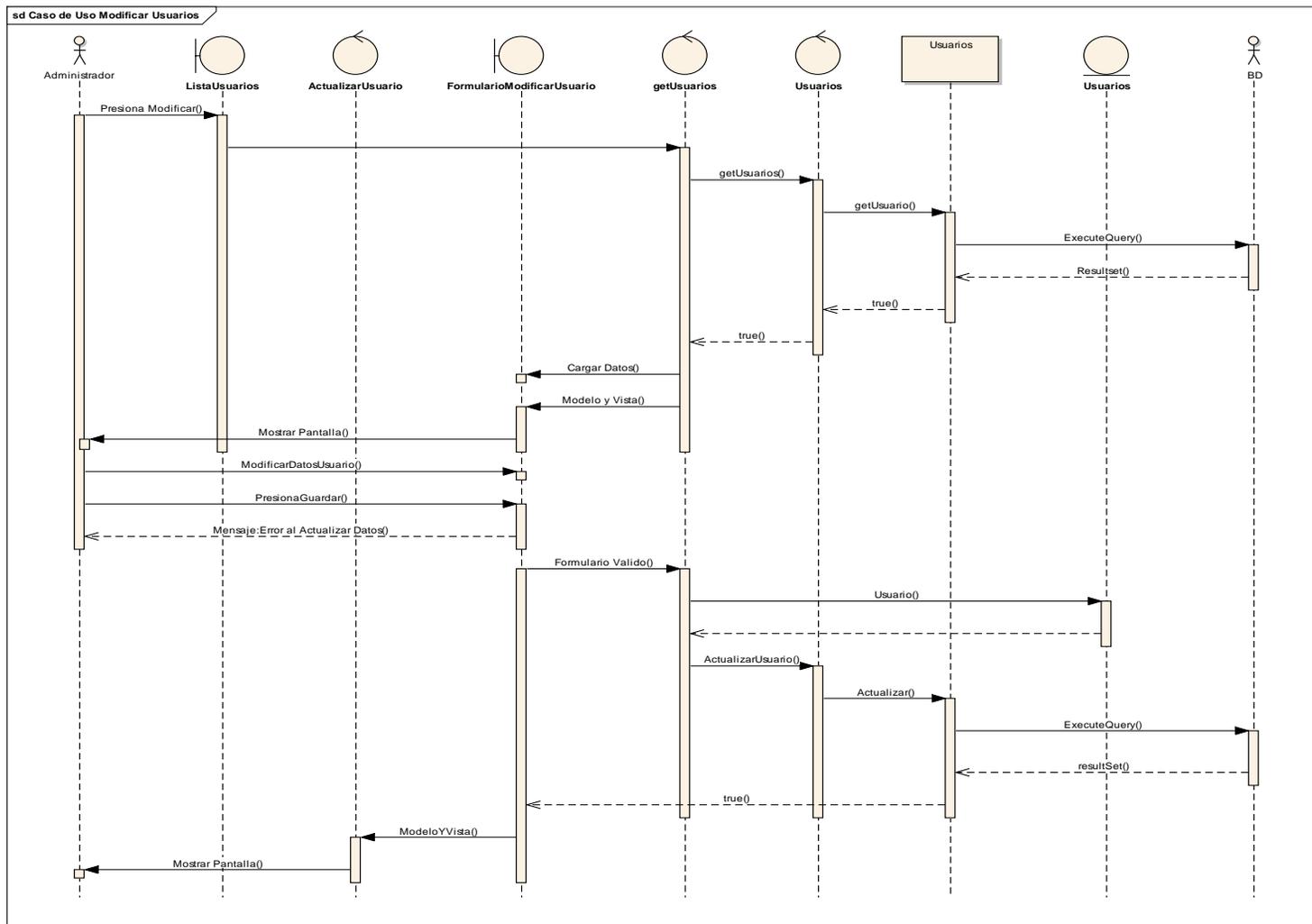


Figura 58 Diagrama de secuencia: caso de uso modificar usuarios

II.1.8.1.3.5. Diagrama de Secuencia: Caso de Uso Eliminar Usuarios

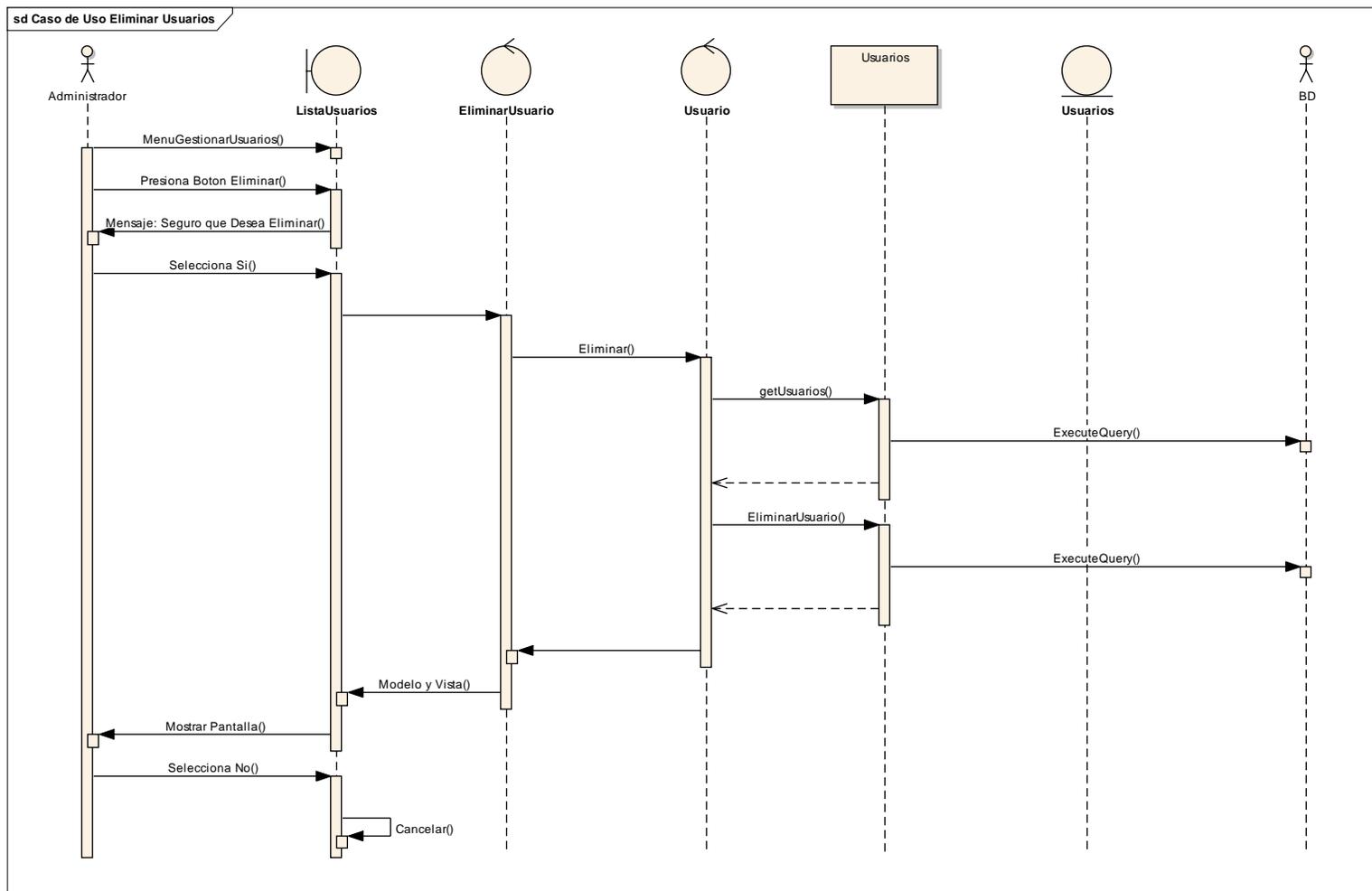


Figura 59 Diagrama de secuencia: caso de uso eliminar usuarios

II.1.8.1.3.6. Diagrama de Secuencia: Caso de Uso Gestionar Dispositivos

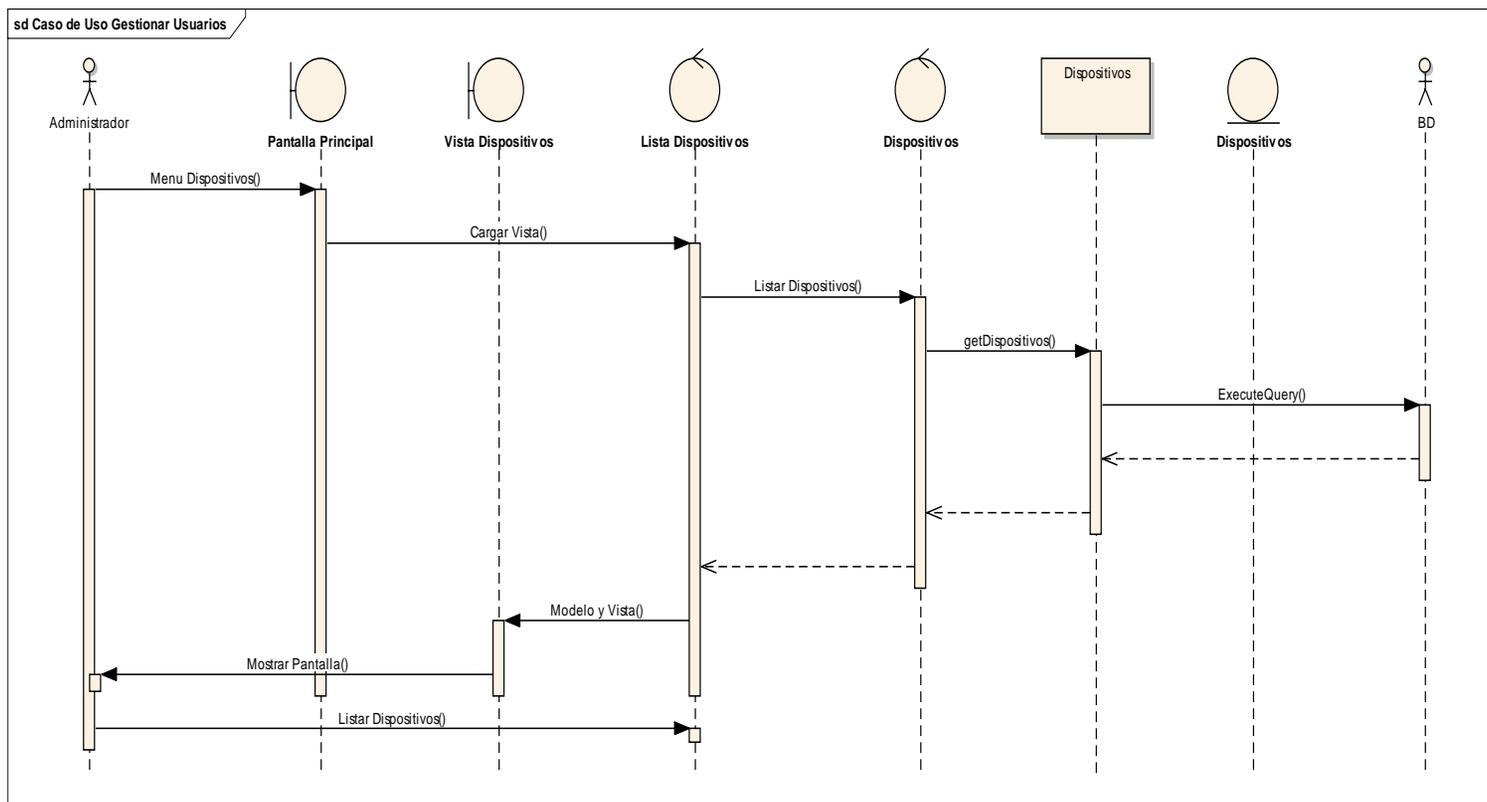


Figura 60 Diagrama de secuencia: caso de uso gestionar dispositivos

II.1.8.1.3.7. Diagrama de Secuencia: Caso de Uso Registrar Dispositivo

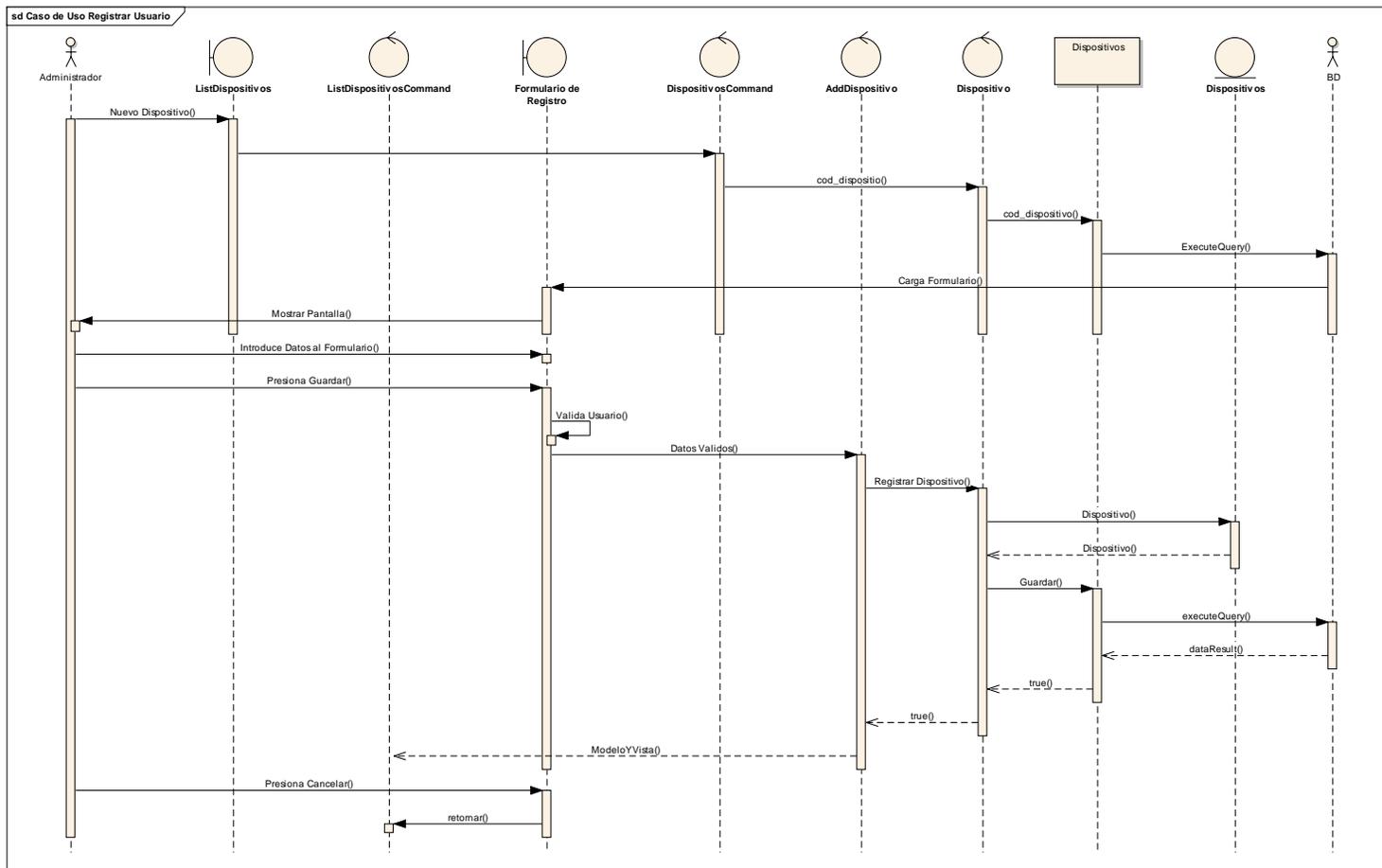


Figura 61 Diagrama de secuencia: caso de uso registrar dispositivo

II.1.8.1.3.8. Diagrama de Secuencia: Caso de Uso Modificar Dispositivo

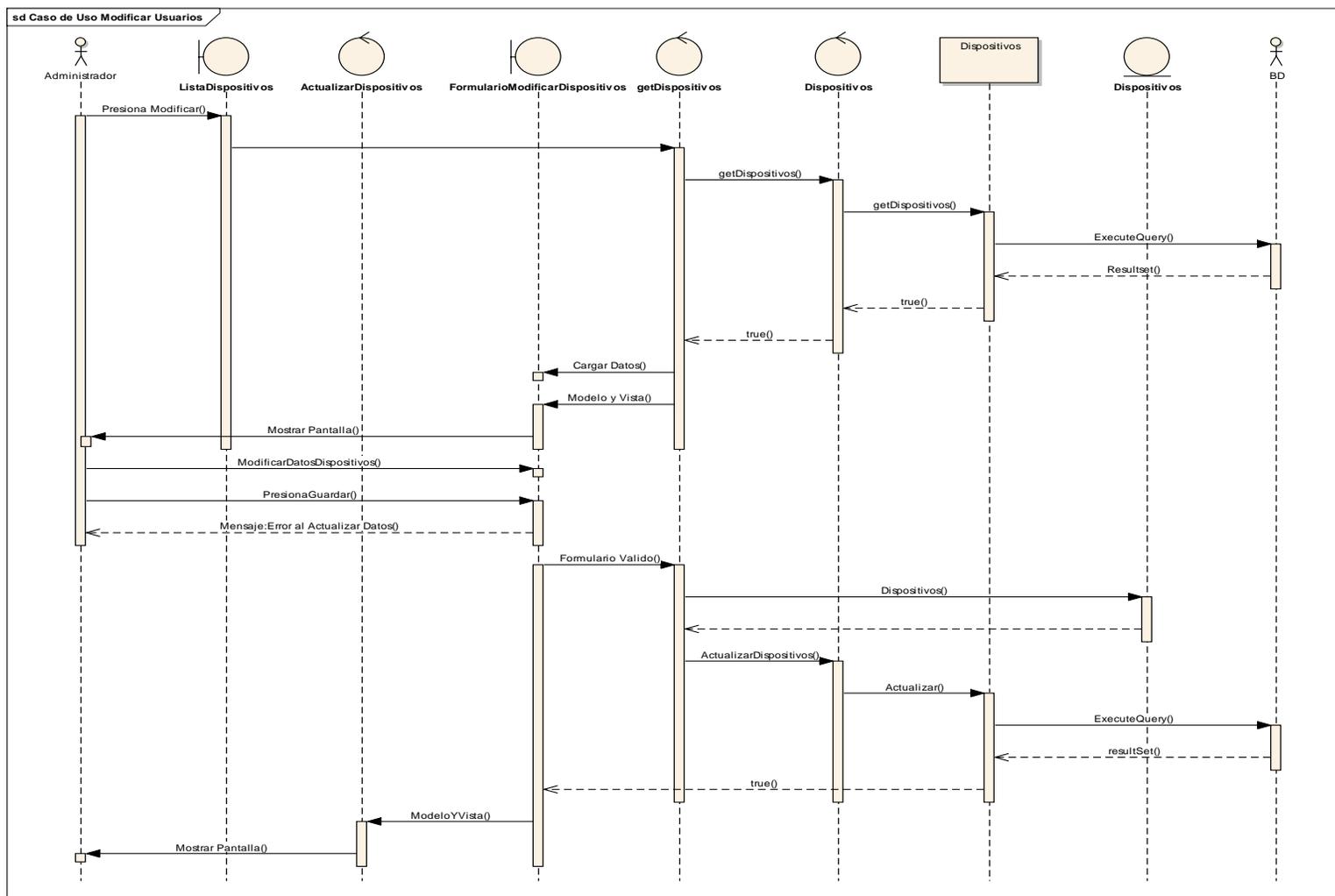


Figura 62 Diagrama de secuencia: caso de uso modificar dispositivo

II.1.8.1.3.9. Diagrama de Secuencia: Caso de Uso Eliminar Dispositivo

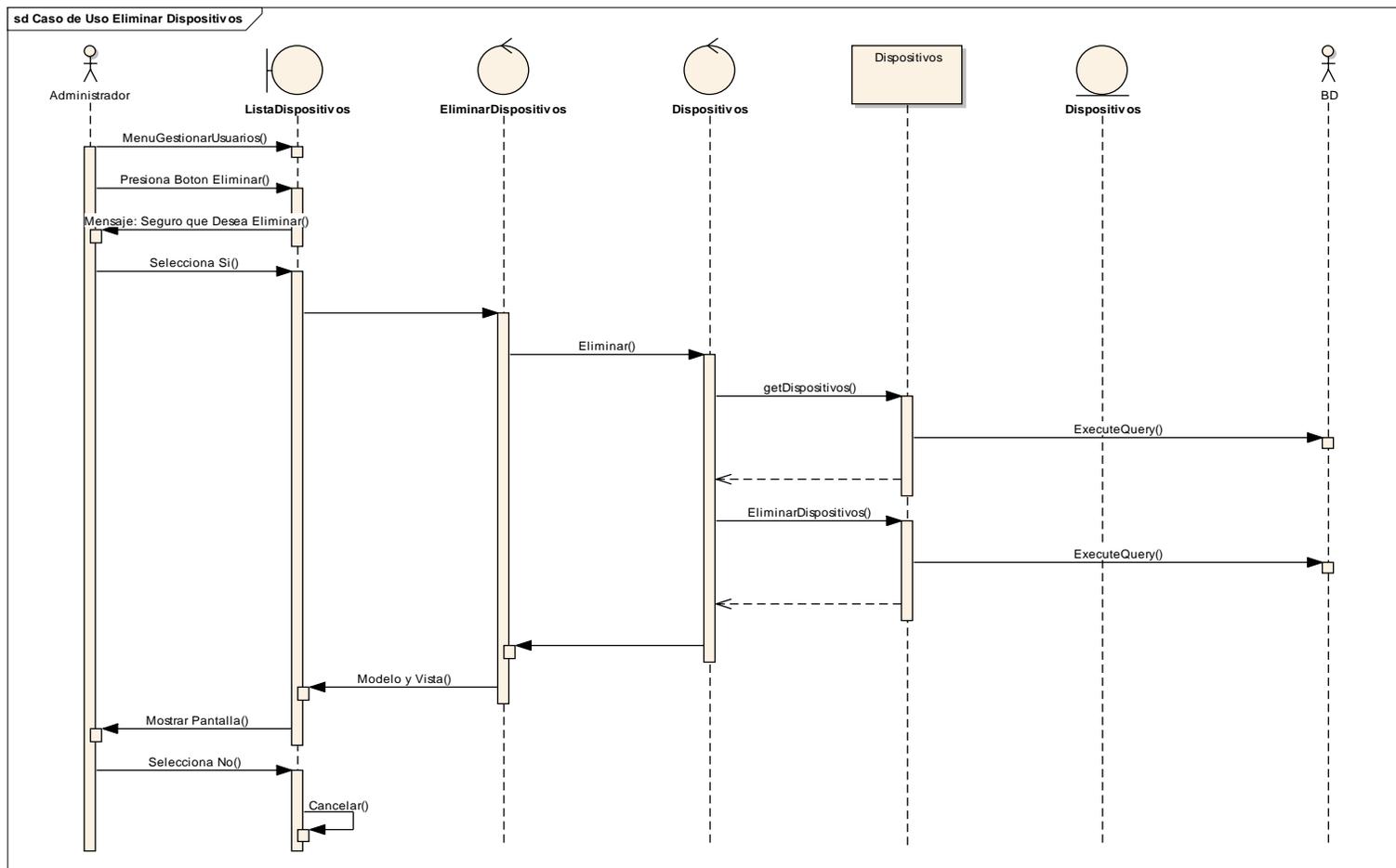


Figura 63 Diagrama de secuencia: caso de uso eliminar dispositivo

II.1.8.1.3.10. Diagrama de Secuencia: Caso de Uso Gestionar Inmuebles

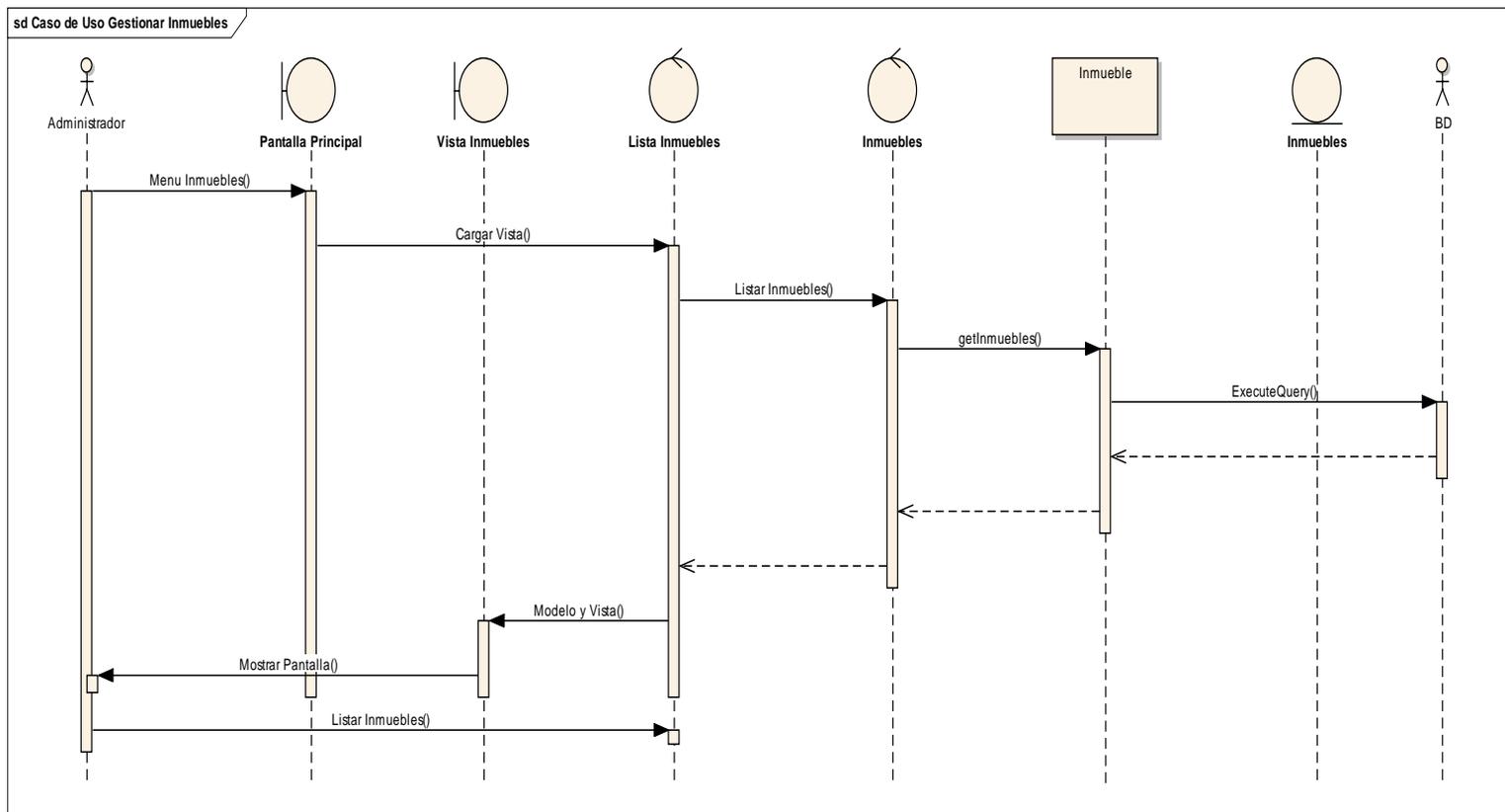


Figura 64 Diagrama de secuencia: caso de uso gestionar inmuebles

II.1.8.1.3.11. Diagrama de Secuencia: Caso de Uso Registrar Inmuebles

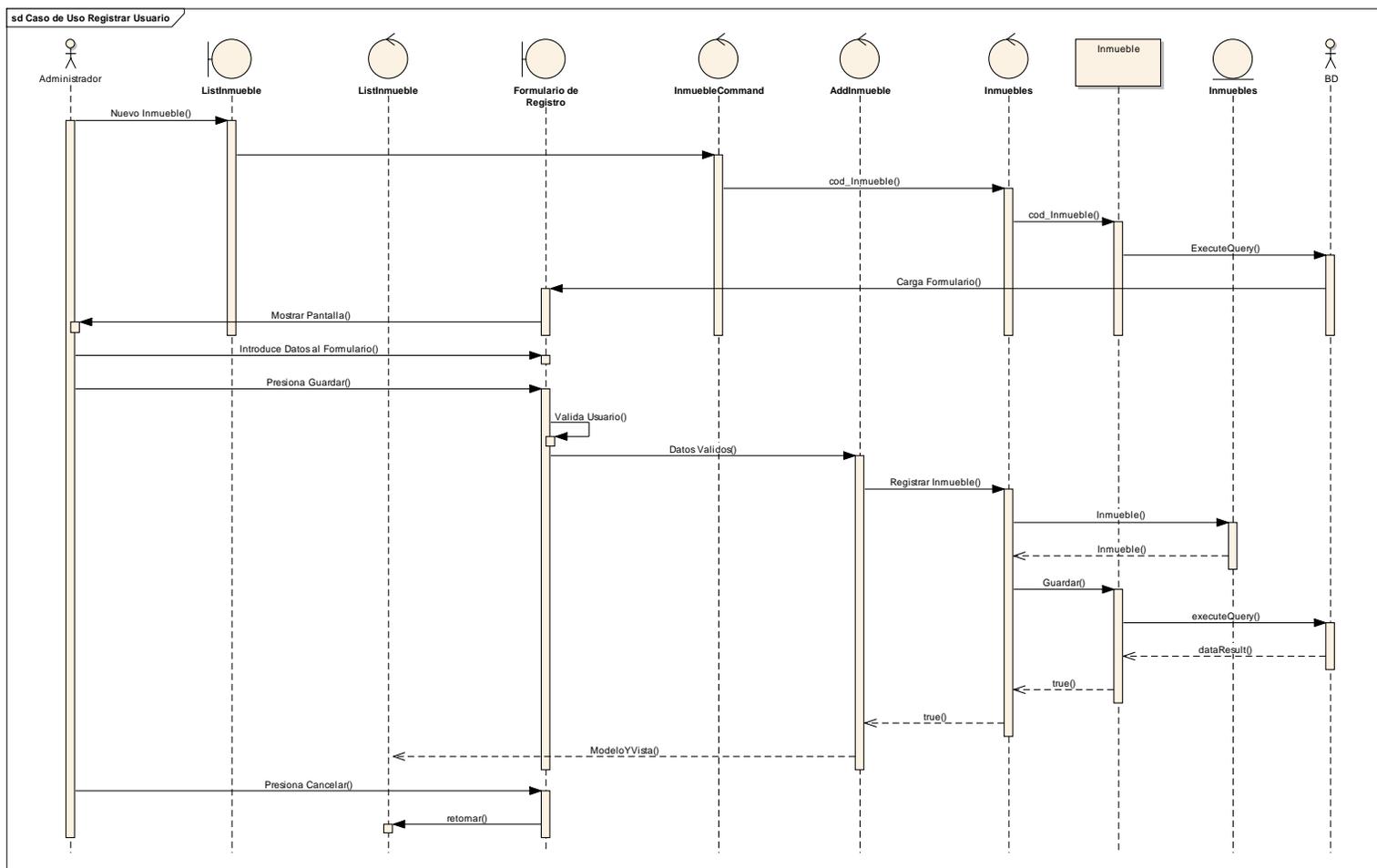


Figura 65 Diagrama de secuencia: caso de uso registrar inmuebles

II.1.8.1.3.12. Diagrama de Secuencia: Caso de Uso Modificar Inmuebles

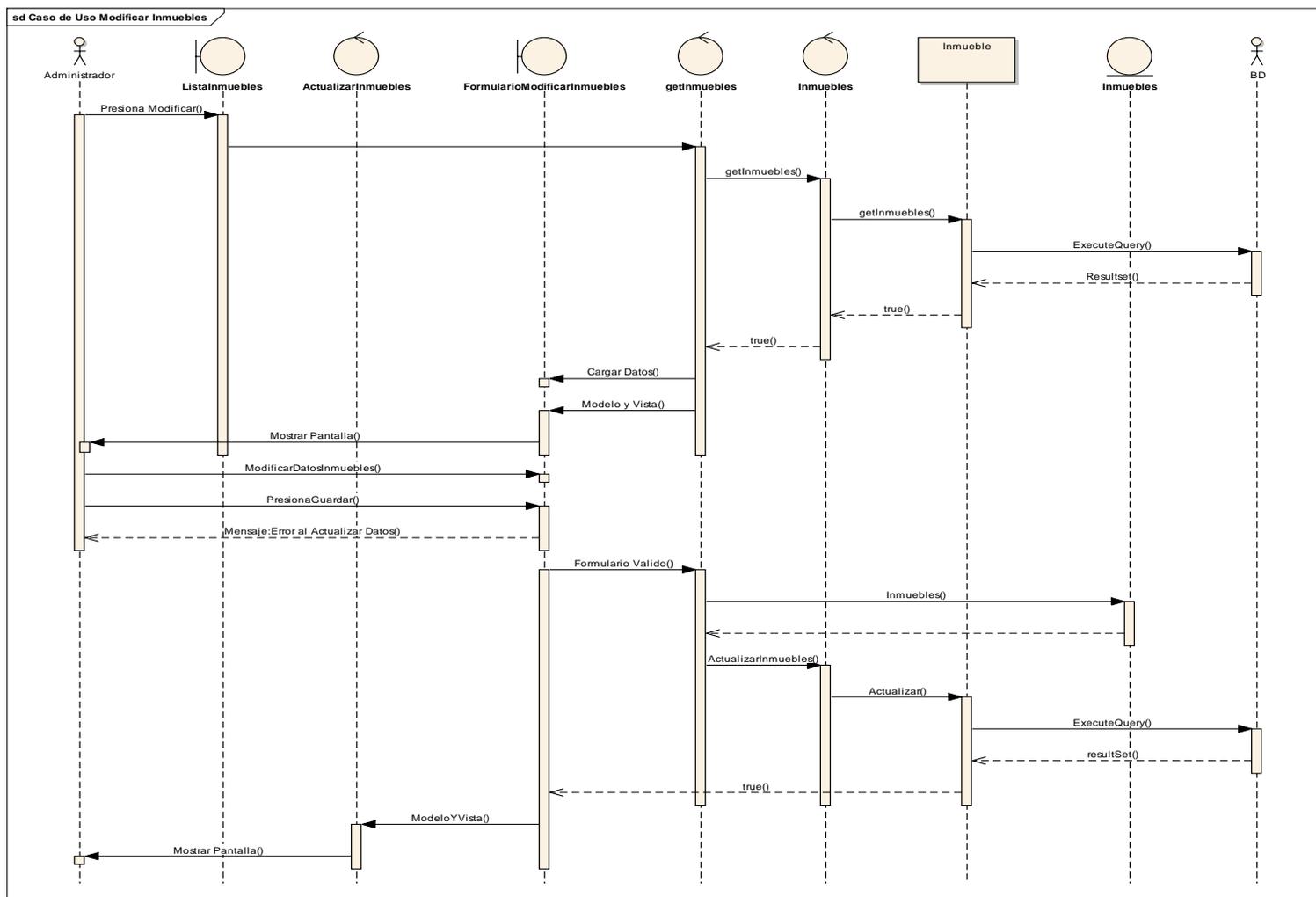


Figura 66 Diagrama de secuencia: caso de uso modificar inmuebles

II.1.8.1.3.13. Diagrama de Secuencia: Caso de Uso Eliminar Inmuebles

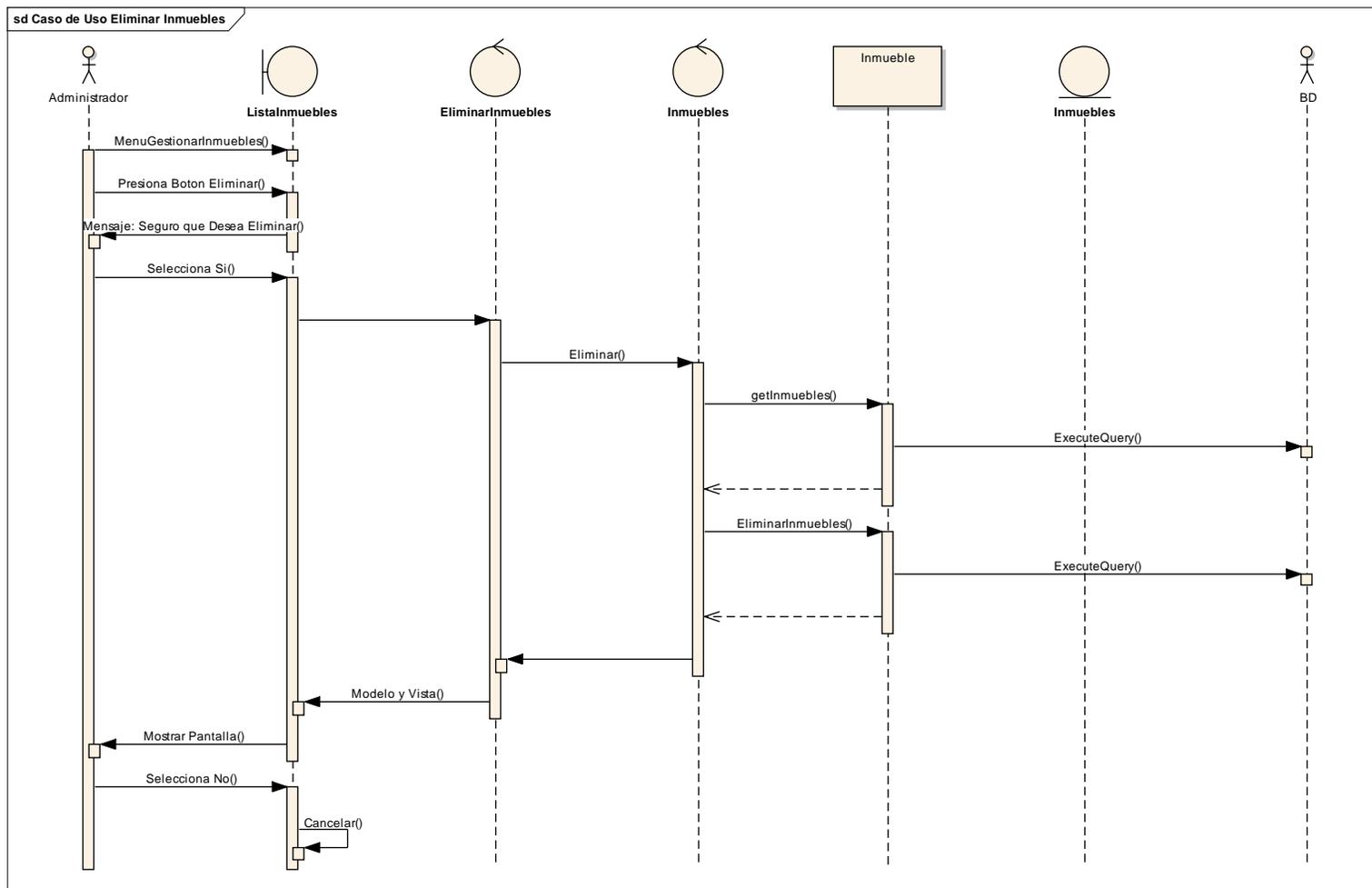


Figura 67 Diagrama de secuencia: caso de uso eliminar inmuebles

II.1.8.1.3.14. Diagrama de Secuencia: Caso de Uso Gestionar Ambientes

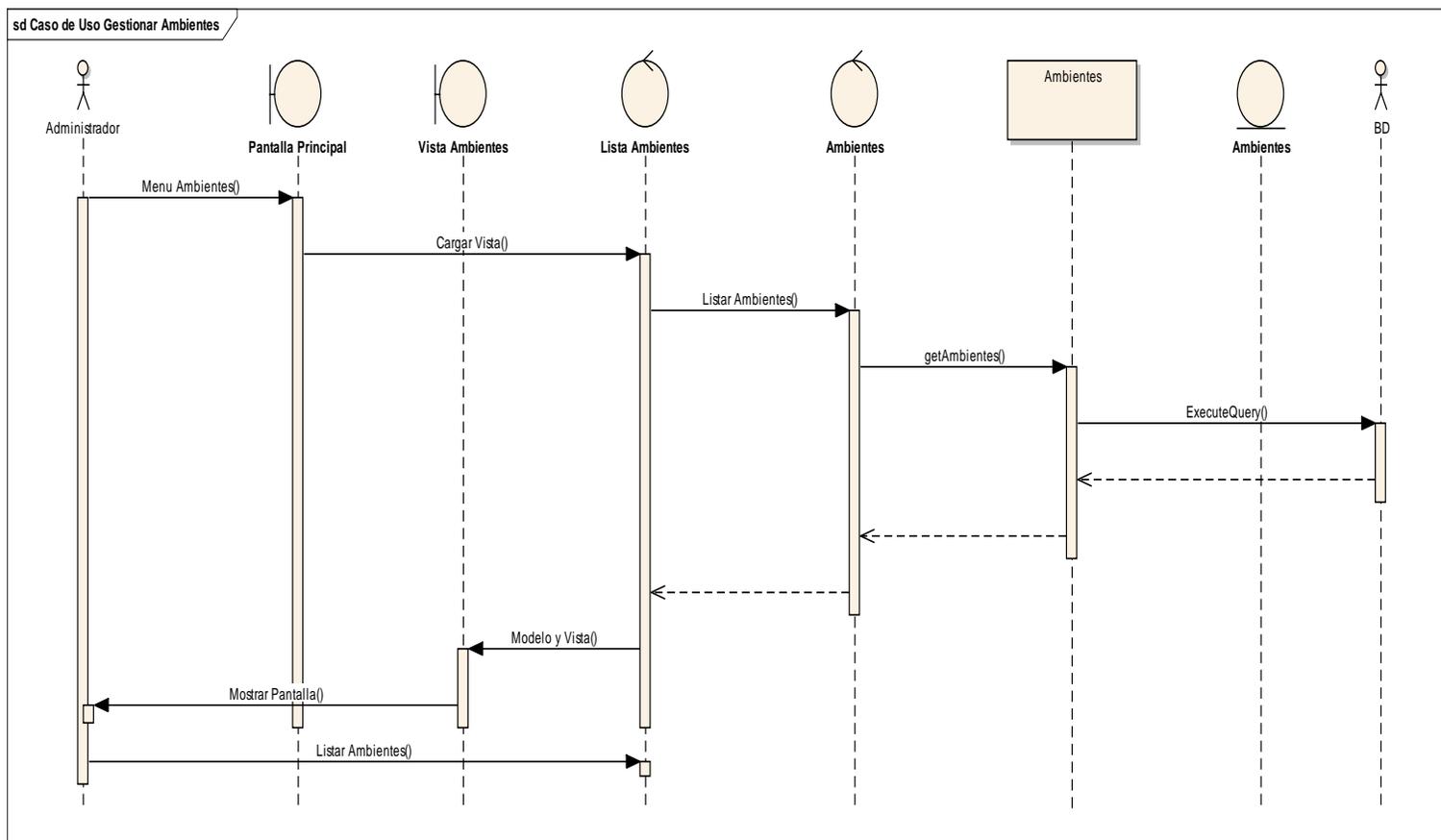


Figura 68 Diagrama de secuencia: caso de uso gestionar ambientes

II.1.8.1.3.15. Diagrama de Secuencia: Caso de Uso Registrar Ambientes

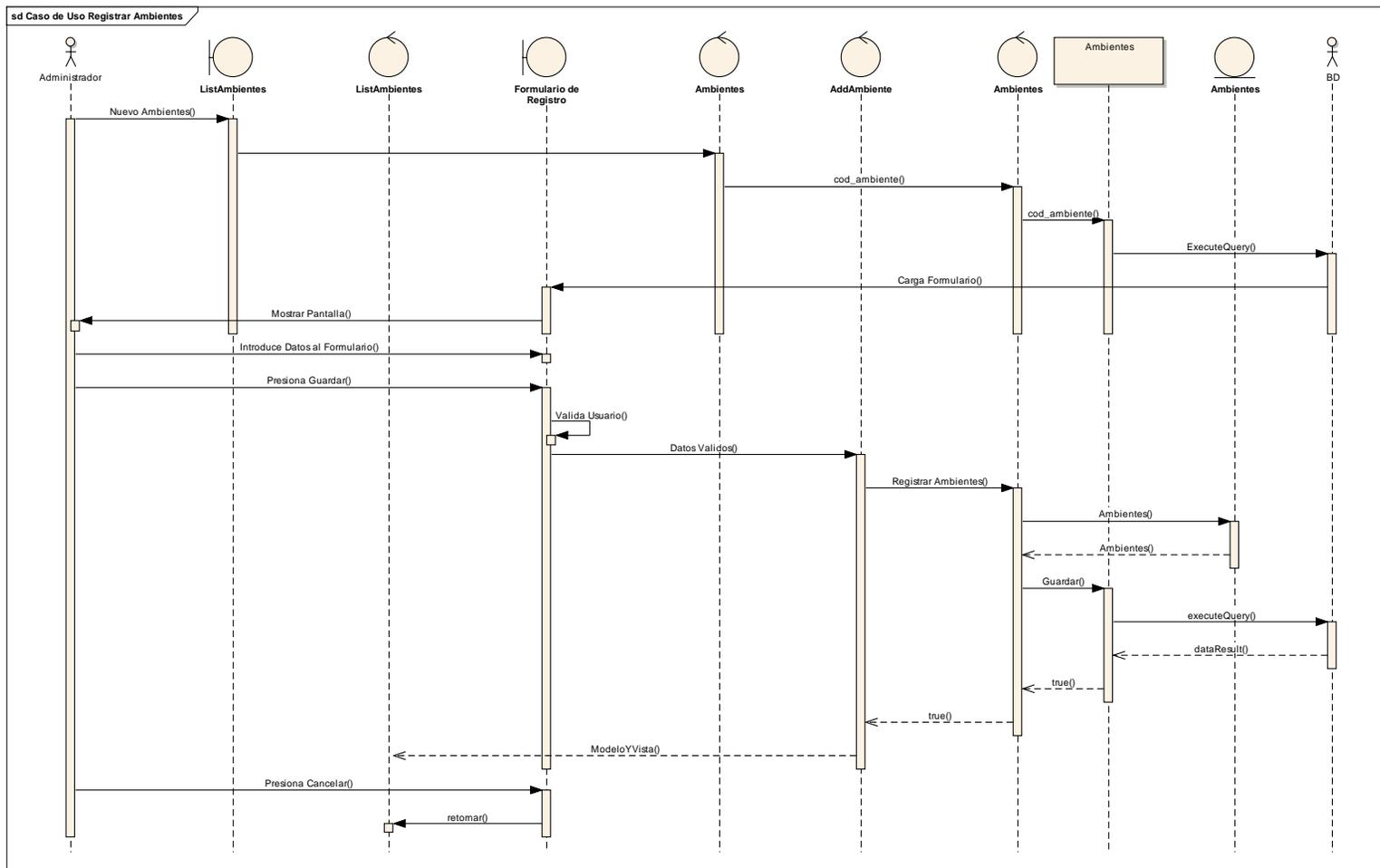


Figura 69 Diagrama de secuencia: caso de uso registrar ambientes

II.1.8.1.3.16. Diagrama de Secuencia: Caso de Uso Modificar Ambientes

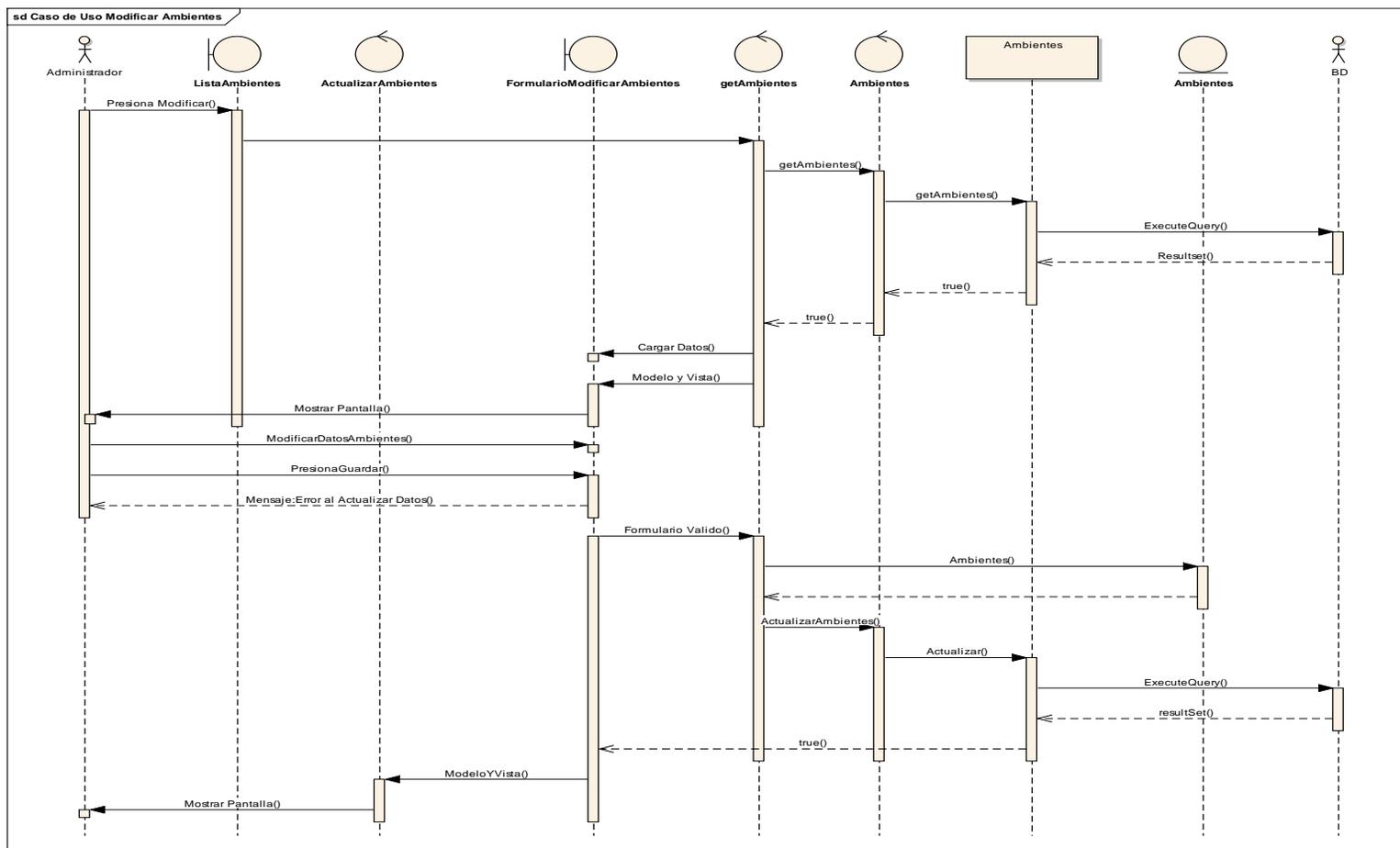


Figura 70 Diagrama de secuencia: caso de uso modificar ambientes

II.1.8.1.3.17. Diagrama de Secuencia: Caso de Uso Eliminar Ambientes

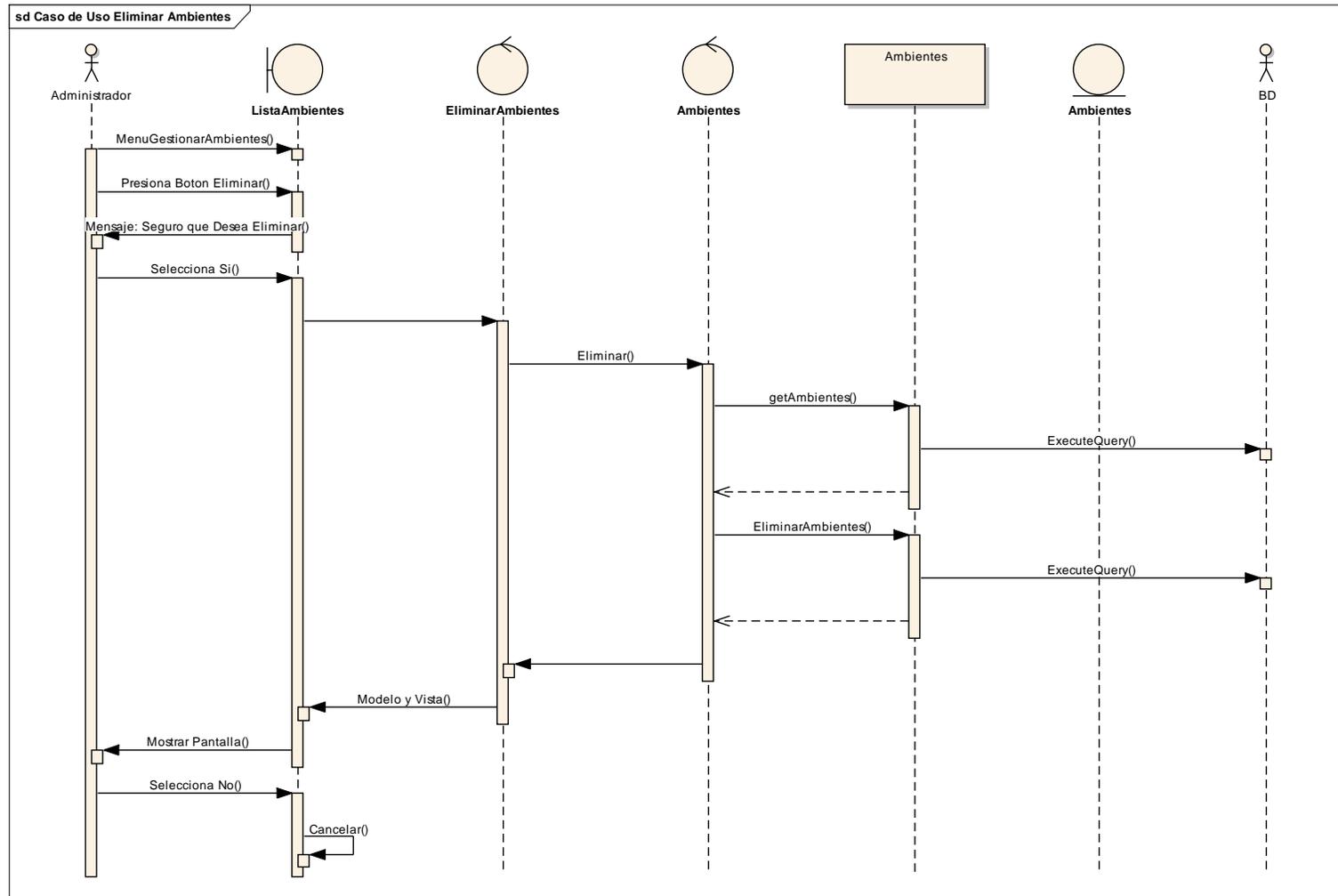


Figura 71 Diagrama de secuencia: caso de uso eliminar ambientes

II.1.9. Modelado de Datos

Previendo que la persistencia de la información del sistema será soportada por una base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este modelo se utiliza un Diagrama de Clases (donde se utiliza un profile UML para Modelado de Datos, para conseguir la representación de tablas, claves, etc.).

II.1.9.1. Modelado de Diagrama de Clases

II.1.9.1.1. Introducción

El diagrama de clases es el diagrama principal para el análisis y diseño. Un diagrama de clases representa las clases del sistema con sus relaciones estructurales y de herencia. La definición de clase incluye definiciones para atributos y operaciones. El modelo de casos de uso aporta información para establecer las clases, objetos, atributos y operaciones.

II.1.9.1.2. Mecanismos de Abstracción

- Clasificación / Instanciación
- Composición / Descomposición
- Agrupación / Individualización
- Especialización / Generalización

La clasificación es una de los mecanismos de abstracción más utilizados. La clase define el ámbito de definición de un conjunto de objetos, y cada objeto pertenece a una clase. Los objetos se crean por instanciación de las clases.

Cada clase se representa en un rectángulo con tres compartimientos.

- Nombre de la clase
- Atributos de la clase
- Operaciones de la clase

Los atributos de una clase no deberían ser manipulables directamente por el resto de objetos. Por esta razón se crearon niveles de visibilidad para los elementos que son:

- (-) Privado: es el más fuerte. Esta parte es totalmente invisible.
- (#) Los atributos u operaciones protegidas están visibles para las clases de herencia.
- (+) Los atributos u operaciones públicas son visibles desde otras clases y también por clases de herencia.

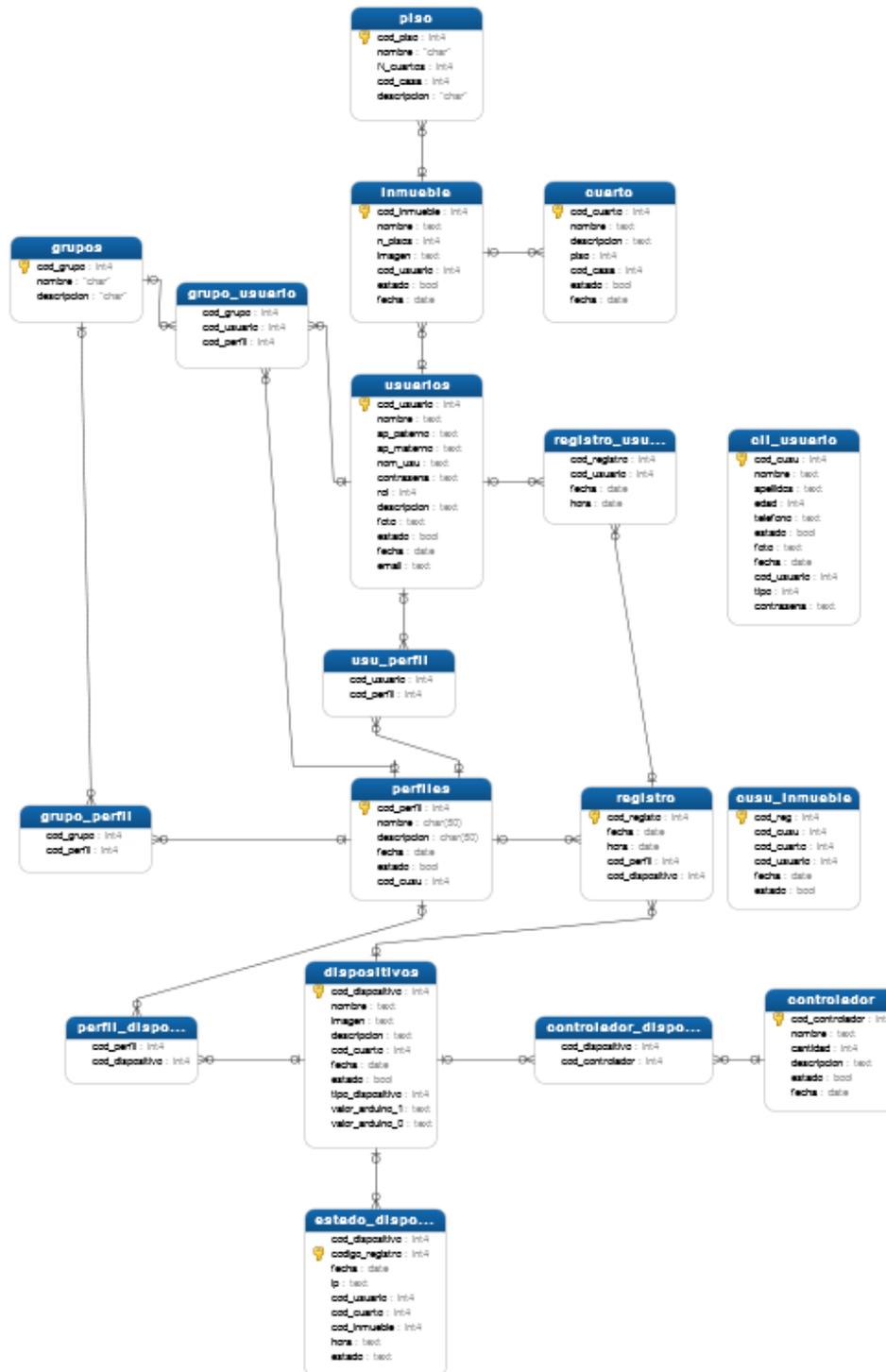
II.1.9.2. Modelo Entidad Relación

II.1.9.2.1. Introducción

Cuando se utiliza una base de datos para gestionar información, se está plasmando una parte del mundo real en una serie de tablas, registros y campos ubicados en un ordenador; creándose un modelo parcial de la realidad. Antes de crear físicamente estas tablas en el ordenador se debe realizar un modelo de datos.

Se suele cometer el error de ir creando nuevas tablas a medida que se van necesitando, haciendo así el modelo de datos y la construcción física de las tablas simultáneamente. El resultado de esto acaba siendo un sistema de información parcheado, con datos dispersos que terminan por no cumplir adecuadamente los requisitos necesarios.

II.1.9.2.2. Modelo Entidad Relación



II.1.9.2.3. Creación de la Base de Datos

```

-- Database: "Domuino"

-- DROP DATABASE "Domuino";

CREATE DATABASE "Domuino"
  WITH OWNER = postgres
       ENCODING = 'UTF8'
       TABLESPACE = pg_default
       LC_COLLATE = 'Spanish_Bolivia.1252'
       LC_CTYPE = 'Spanish_Bolivia.1252'
       CONNECTION LIMIT = -1;
-- Table: cli_usuario

-- DROP TABLE cli_usuario;

CREATE TABLE cli_usuario
(
  cod_cusu integer NOT NULL DEFAULT nextval('c_usuario'::regclass),
  nombre text,
  apellidos text,
  edad integer,
  telefono text,
  estado boolean DEFAULT true,
  foto text,
  fecha date,
  cod_usuario integer,
  tipo integer,
  contrasena text,
  CONSTRAINT cli_usuario_pkey PRIMARY KEY (cod_cusu)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE cli_usuario
  OWNER TO postgres;

-- Table: controlador

-- DROP TABLE controlador;

```

```

CREATE TABLE controlador
(
  cod_controlador integer NOT NULL DEFAULT nextval('incremento'::regclass),
  nombre text,
  cantidad integer,
  descripcion text,
  estado boolean DEFAULT true,
  fecha date,
  CONSTRAINT controlador_pkey PRIMARY KEY (cod_controlador)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE controlador
  OWNER TO postgres;
-- Table: controlador_dispositivo

-- DROP TABLE controlador_dispositivo;

CREATE TABLE controlador_dispositivo
(
  cod_dispositivo integer,
  cod_controlador integer,
  CONSTRAINT controlador_dispositivo_cod_controlador_fkey FOREIGN KEY
(cod_controlador)
  REFERENCES controlador (cod_controlador) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT controlador_dispositivo_cod_dispositivo_fkey FOREIGN KEY
(cod_dispositivo)
  REFERENCES dispositivos (cod_dispositivo) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE controlador_dispositivo
  OWNER TO postgres;

-- Table: cuarto
-- DROP TABLE cuarto;
CREATE TABLE cuarto
(
  cod_cuarto integer NOT NULL DEFAULT nextval('cuartos'::regclass),

```

```

nombre text,
descripcion text,
piso integer,
cod_casa integer,
estado boolean DEFAULT true,
fecha date,
CONSTRAINT cuarto_pkey PRIMARY KEY (cod_cuarto),
CONSTRAINT cuarto_cod_casa_fkey FOREIGN KEY (cod_casa)
  REFERENCES inmueble (cod_inmueble) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE cuarto
  OWNER TO postgres;

-- Table: cusu_inmueble

-- DROP TABLE cusu_inmueble;

CREATE TABLE cusu_inmueble
(
  cod_reg integer NOT NULL DEFAULT nextval('registrocim'::regclass),
  cod_cusu integer,
  cod_cuarto integer,
  cod_usuario integer,
  fecha date,
  estado boolean NOT NULL DEFAULT true,
  CONSTRAINT cusu_inmueble_pkey PRIMARY KEY (cod_reg)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE cusu_inmueble
  OWNER TO postgres;

-- Table: dispositivos

-- DROP TABLE dispositivos;

CREATE TABLE dispositivos
(

```

```

cod_dispositivo integer NOT NULL DEFAULT nextval('coddispositivo'::regclass),
nombre text,
imagen text,
descripcion text,
cod_cuarto integer,
fecha date,
estado boolean DEFAULT true,
tipo_dispositivo integer,
valor_arduino_1 text,
valor_arduino_0 text,
CONSTRAINT dispositivos_pkey PRIMARY KEY (cod_dispositivo)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE dispositivos
  OWNER TO postgres;

-- Table: estado_dispositivo
-- DROP TABLE estado_dispositivo;

CREATE TABLE estado_dispositivo
(
  cod_dispositivo integer,
  codigo_registro integer NOT NULL DEFAULT nextval('regdispositivo'::regclass),
  fecha date,
  ip text,
  cod_usuario integer,
  cod_cuarto integer,
  cod_inmueble integer,
  hora text,
  estado text,
  CONSTRAINT estado_dispositivo_pkey PRIMARY KEY (codigo_registro),
  CONSTRAINT estado_dispositivo_cod_dispositivo_fkey FOREIGN KEY
(cod_dispositivo)
  REFERENCES dispositivos (cod_dispositivo) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE estado_dispositivo

```

```
OWNER TO postgres;

-- Table: grupo_perfil
-- DROP TABLE grupo_perfil;
CREATE TABLE grupo_perfil
(
  cod_grupo integer,
  cod_perfil integer,
  CONSTRAINT grupo_perfil_cod_grupo_fkey FOREIGN KEY (cod_grupo)
    REFERENCES grupos (cod_grupo) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT grupo_perfil_cod_perfil_fkey FOREIGN KEY (cod_perfil)
    REFERENCES perfiles (cod_perfil) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE grupo_perfil
  OWNER TO postgres;

-- Table: grupo_usuario
-- DROP TABLE grupo_usuario;

CREATE TABLE grupo_usuario
(
  cod_grupo integer,
  cod_usuario integer,
  cod_perfil integer,
  CONSTRAINT grupo_usuario_cod_grupo_fkey FOREIGN KEY (cod_grupo)
    REFERENCES grupos (cod_grupo) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT grupo_usuario_cod_perfil_fkey FOREIGN KEY (cod_perfil)
    REFERENCES perfiles (cod_perfil) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT grupo_usuario_cod_usuario_fkey FOREIGN KEY (cod_usuario)
    REFERENCES usuarios (cod_usuario) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
```

```
ALTER TABLE grupo_usuario
  OWNER TO postgres;

-- Table: grupos

-- DROP TABLE grupos;

CREATE TABLE grupos
(
  cod_grupo integer NOT NULL,
  nombre "char",
  descripcion "char",
  CONSTRAINT grupos_pkey PRIMARY KEY (cod_grupo)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE grupos
  OWNER TO postgres;

-- Table: inmueble

-- DROP TABLE inmueble;

CREATE TABLE inmueble
(
  cod_inmueble integer NOT NULL DEFAULT nextval('codinmueble'::regclass),
  nombre text,
  n_pisos integer,
  imagen text,
  cod_usuario integer,
  estado boolean DEFAULT true,
  fecha date,
  CONSTRAINT casa_pkey PRIMARY KEY (cod_inmueble),
  CONSTRAINT casa_cod_usuario_fkey FOREIGN KEY (cod_usuario)
    REFERENCES usuarios (cod_usuario) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
```

```

);
ALTER TABLE inmueble
  OWNER TO postgres;

-- Table: perfil_dispositivo

-- DROP TABLE perfil_dispositivo;

CREATE TABLE perfil_dispositivo
(
  cod_perfil integer,
  cod_dispositivo integer,
  CONSTRAINT perfil_dispositivo_cod_dispositivo_fkey FOREIGN KEY
(cod_dispositivo)
  REFERENCES dispositivos (cod_dispositivo) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT perfil_dispositivo_cod_perfil_fkey FOREIGN KEY (cod_perfil)
  REFERENCES perfiles (cod_perfil) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE perfil_dispositivo
  OWNER TO postgres;
-- Table: perfiles

-- DROP TABLE perfiles;

CREATE TABLE perfiles
(
  cod_perfil integer NOT NULL DEFAULT nextval('perfil'::regclass),
  nombre character(50),
  descripcion character(50),
  fecha date,
  estado boolean DEFAULT true,
  cod_cusu integer,
  CONSTRAINT perfiles_pkey PRIMARY KEY (cod_perfil)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE perfiles

```

```
OWNER TO postgres;

-- Table: piso

-- DROP TABLE piso;

CREATE TABLE piso
(
  cod_piso integer NOT NULL,
  nombre "char",
  "N_cuartos" integer,
  cod_casa integer,
  descripcion "char",
  CONSTRAINT piso_pkey PRIMARY KEY (cod_piso),
  CONSTRAINT piso_cod_casa_fkey FOREIGN KEY (cod_casa)
    REFERENCES inmueble (cod_inmueble) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE piso
  OWNER TO postgres;

-- Table: registro

-- DROP TABLE registro;

CREATE TABLE registro
(
  cod_registro integer NOT NULL,
  fecha date,
  hora date,
  cod_perfil integer,
  cod_dispositivo integer,
  CONSTRAINT registro_pkey PRIMARY KEY (cod_registro),
  CONSTRAINT registro_cod_dispositivo_fkey FOREIGN KEY (cod_dispositivo)
    REFERENCES dispositivos (cod_dispositivo) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT registro_cod_perfil_fkey FOREIGN KEY (cod_perfil)
```

```

REFERENCES perfiles (cod_perfil) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE registro
  OWNER TO postgres;

-- Table: registro_usuario
-- DROP TABLE registro_usuario;
CREATE TABLE registro_usuario
(
  cod_registro integer,
  cod_usuario integer,
  fecha date,
  hora date,
  CONSTRAINT registro_usuario_cod_registro_fkey FOREIGN KEY (cod_registro)
    REFERENCES registro (cod_registro) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT registro_usuario_cod_usuario_fkey FOREIGN KEY (cod_usuario)
    REFERENCES usuarios (cod_usuario) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE registro_usuario
  OWNER TO postgres;
-- Table: usu_perfil

-- DROP TABLE usu_perfil;

CREATE TABLE usu_perfil
(
  cod_usuario integer,
  cod_perfil integer,
  CONSTRAINT usu_perfil_cod_perfil_fkey FOREIGN KEY (cod_perfil)
    REFERENCES perfiles (cod_perfil) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT usu_perfil_cod_usuario_fkey FOREIGN KEY (cod_usuario)
    REFERENCES usuarios (cod_usuario) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION

```

```
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE usu_perfil  
  OWNER TO postgres;  
  
-- Table: usuarios  
-- DROP TABLE usuarios;  
  
CREATE TABLE usuarios  
(  
  cod_usuario integer NOT NULL DEFAULT nextval('incremento'::regclass),  
  nombre text,  
  ap_paterno text,  
  ap_materno text,  
  nom_usu text,  
  contrasena text,  
  rol integer,  
  descripcion text,  
  foto text,  
  estado boolean DEFAULT true,  
  fecha date,  
  email text,  
  CONSTRAINT usuarios_pkey PRIMARY KEY (cod_usuario)  
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE usuarios  
  OWNER TO postgres;
```

II.1.9.3. Diccionario de Datos**II.1.9.3.1. Cli_Usuario**

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_cusu	integer	Yes	Yes	nextval('c_usuario'::regclass)	
nombre	text	No	No		
apellidos	text	No	No		
edad	integer	No	No		
telefono	text	No	No		
estado	boolean	No	No	true	
foto	text	No	No		
fecha	date	No	No		
cod_usuario	integer	No	No		
tipo	integer	No	No		
contrasena	text	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
cli_usuario_pkey	Primary key	(cod_cusu)	

Tabla 56 Diccionario de datos tabla cli_usuario

II.1.9.3.2. Controlador

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_controlador	integer	Yes	Yes	nextval('incremento'::regclass)	
nombre	text	No	No		
cantidad	integer	No	No		
descripcion	text	No	No		
estado	boolean	No	No	true	
fecha	date	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
controlador_pkey	Primary key	(cod_controlador)	

Tabla 57 Diccionario de datos tabla controlador

II.1.9.3.3. Controlador_dispositivo

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_dispositivo	integer	No	No		
cod_controlador	integer	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
controlador_dispositivo_cod_controlador_fkey	Foreign key	(cod_controlador) REFERENCES controlador (cod_controlador) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
controlador_dispositivo_cod_dispositivo_fkey	Foreign key	(cod_dispositivo) REFERENCES dispositivos (cod_dispositivo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Tabla 58 Diccionario de datos tabla controlador_dispositivo

II.1.9.3.4. Cuarto

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_cuarto	integer	Yes	Yes	nextval('cuartos::regla ss')	
nombre	text	No	No		
descripcion	text	No	No		
piso	integer	No	No		
cod_casa	integer	No	No		
estado	boolean	No	No	true	
fecha	date	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
cuarto_pk ey	Primary key	(cod_cuarto)	
cuarto_cod_casa_fk ey	Foreign key	(cod_casa) REFERENCES inmueble (cod_inmueble) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Tabla 59 Diccionario de datos tabla cuarto

II.1.9.3.5. Cusu_inmueble

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_reg	integer	Yes	Yes	nextval('registrocim'::regclass)	
cod_cusu	integer	No	No		
cod_cuarto	integer	No	No		
cod_usuario	integer	No	No		
fecha	date	No	No		
estado	boolean	Yes	No	true	

Restricción

Nombre	Tipo	Definición	Comentario
cusu_inmueble_pkey	Primary key	(cod_reg)	

Tabla 60 Diccionario de datos tabla cusu_inmueble

II.1.9.3.6. Dispositivos

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria	Por Defecto	Comentario
cod_dispositivo	integer	Yes	Yes	nextval('coddispositivo'::regclass)	
nombre	text	No	No		
imagen	text	No	No		
descripcion	text	No	No		
cod_cuarto	integer	No	No		
fecha	date	No	No		
estado	boolean	No	No	true	
tipo_dispositivo	integer	No	No		
valor_arduino_1	text	No	No		
valor_arduino_0	text	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
dispositivos_pkey	Primary key	(cod_dispositivo)	

Tabla 61 Diccionario de datos tabla dispositivos

II.1.9.3.7. Estado_dispositivos

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_dispositivo	integer	No	No		
codigo_registro	integer	Yes	Yes	nextval('regdispositivo'::regclass)	
fecha	date	No	No		
ip	text	No	No		
cod_usuario	integer	No	No		
cod_cuarto	integer	No	No		
cod_inmueble	integer	No	No		
hora	text	No	No		
estado	text	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
estado_dispositivo_pkey	Primary key	(codigo_registro)	
estado_dispositivo_cod_dispositivo_fkey	Foreign key	(cod_dispositivo) REFERENCES dispositivos (cod_dispositivo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Tabla 62 Diccionario de datos tabla estado_dispositivo

II.1.9.3.8. Grupo_perfil

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_grupo	integer	No	No		
cod_perfil	integer	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
grupo_perfil_cod_grupo_fk	Foreign key	(cod_grupo) REFERENCES grupos (cod_grupo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
grupo_perfil_cod_perfil_fk	Foreign key	(cod_perfil) REFERENCES perfiles (cod_perfil) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Tabla 63 Diccionario de datos tabla grupo_perfil

II.1.9.3.9. Grupo_usuario

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_grupo	integer	No	No		
cod_usuario	integer	No	No		
cod_perfil	integer	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
grupo_usuario_cod_grupo_fk	Foreign key	(cod_grupo) REFERENCES grupos (cod_grupo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
grupo_usuario_cod_perfil_fk	Foreign key	(cod_perfil) REFERENCES perfiles (cod_perfil) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
grupo_usuario_cod_usuario_fk	Foreign key	(cod_usuario) REFERENCES usuarios (cod_usuario) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Tabla 64 Diccionario de datos tabla grupo_usuario

II.1.9.3.10. Grupos

Columnas

Nombre	Tipo de Dato?	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_grupo	integer	Yes	Yes		
nombre	"char"	No	No		
descripcion	"char"	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
grupos_pkey	Primary key	(cod_grupo)	

Tabla 65 Diccionario de datos tabla grupos

II.1.9.3.11. Inmueble

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_inmueble	integer	Yes	Yes	nextval('codinmueble'::regclass)	
nombre	text	No	No		
n_pisos	integer	No	No		
imagen	text	No	No		
cod_usuario	integer	No	No		
estado	boolean	No	No	true	
fecha	date	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
casa_pkey	Primary key	(cod_inmueble)	
casa_cod_usuario_fk	Foreign key	(cod_usuario) REFERENCES usuarios (cod_usuario) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Tabla 66 Diccionario de datos tabla inmueble

II.1.9.3.12. Perfil_dispositivo

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_perfil	integer	No	No		
cod_dispositivo	integer	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
perfil_dispositivo_cod_dispositivo_fkey	Foreign key	(cod_dispositivo) REFERENCES dispositivos (cod_dispositivo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
perfil_dispositivo_cod_perfil_fk	Foreign key	(cod_perfil) REFERENCES perfiles (cod_perfil) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Tabla 67 Diccionario de datos tabla perfil_dispositivo

II.1.9.3.13. Perfiles

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_perfil	integer	Yes	Yes	nextval('perfil::regclas')	
nombre	character(50)	No	No		
descripcion	character(50)	No	No		
fecha	date	No	No		
estado	boolean	No	No	true	

cod_cusu	integer	No	No		
----------	---------	----	----	--	--

Restricción

Nombre	Tipo	Definición	Comentario
perfiles_pkey	Primary key	(cod_perfil)	

Tabla 68 Diccionario de datos tabla perfiles

II.1.9.3.14. Piso

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_piso	integer	Yes	Yes		
nombre	"char"	No	No		
N_cuartos	integer	No	No		
cod_casa	integer	No	No		
descripcion	"char"	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
piso_pkey	Primary key	(cod_piso)	
piso_cod_casa_fk	Foreign key	(cod_casa) REFERENCES inmueble (cod_inmueble) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Tabla 69 Diccionario de datos tabla piso

II.1.9.3.15. Registro

Columnas

Nombre	Tipo de Dato?	No Nulo?	Clave Primaria	Por Defecto	Comentario
cod_registro	integer	Yes	Yes		
fecha	date	No	No		
hora	date	No	No		
cod_perfil	integer	No	No		
cod_dispositivo	integer	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
registro_pkey	Primary key	(cod_registro)	
registro_cod_dispositivo_fkey	Foreign key	(cod_dispositivo) REFERENCES dispositivos (cod_dispositivo) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
registro_cod_perfil_fkey	Foreign key	(cod_perfil) REFERENCES perfiles (cod_perfil) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Tabla 70 Diccionario de datos tabla registro

II.1.9.3.16. Registro_usuario**Columnas**

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_registro	integer	No	No		
cod_usuario	integer	No	No		
fecha	date	No	No		
hora	date	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
registro_usuario_cod_registro_fk	Foreign key	(cod_registro) REFERENCES registro (cod_registro) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
registro_usuario_cod_usuario_fk	Foreign key	(cod_usuario) REFERENCES usuarios (cod_usuario) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Tabla 71 Diccionario de datos tabla registro_usuario

II.1.9.3.17. Usu_perfil

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_usuario	integer	No	No		
cod_perfil	integer	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
usu_perfil_cod_perfil_fkey	Foreign key	(cod_perfil) REFERENCES perfiles (cod_perfil) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	
usu_perfil_cod_usuario_fkey	Foreign key	(cod_usuario) REFERENCES usuarios (cod_usuario) MATCH SIMPLE ON UPDATE NO ACTION ON DELETE NO ACTION	

Tabla 72 Diccionario de datos tabla usu_perfil

II.1.9.3.18. Usuarios

Columnas

Nombre	Tipo de Dato	No Nulo?	Clave Primaria?	Por Defecto	Comentario
cod_usuario	integer	Yes	Yes	nextval('incremento'::regclass)	
nombre	text	No	No		
ap_paterno	text	No	No		
ap_materno	text	No	No		
nom_usu	text	No	No		

contrasena	text	No	No		
rol	integer	No	No		
descripcion	text	No	No		
foto	text	No	No		
estado	boolean	No	No	true	
fecha	date	No	No		
email	text	No	No		

Restricción

Nombre	Tipo	Definición	Comentario
usuarios_pkey	Primary key	(cod_usuario)	

Tabla 73 Diccionario de datos tabla usuarios

II.1.9.4. Plan de Pruebas

II.1.9.4.1. Descripción de Aspectos Generales

Esta sección establece el alcance y el objetivo del Plan de Pruebas. Es aquí donde se describen los aspectos fundamentales del esfuerzo que se hará para probar cada uno de los módulos que conforman el sistema descrito en este Plan de Desarrollo de Software, independiente las características y tamaño que ésta pueda tener.

II.1.9.4.1.1. Objetivo

Este Plan de Pruebas fue desarrollado con el fin de guiar el proceso de pruebas al proyecto Domuino para el Mejoramiento del control y monitoreo de inmuebles utilizando la tecnología web y el hardware de código abierto ARDUINO desarrollado por nuestro grupo de trabajo, esto con el fin de asegurar una excelente calidad del software desarrollado, encontrando para ello errores que puedan perjudicar en el funcionamiento de este, los cuales una vez corregidos, se podrá elaborar un documento que presente las evaluaciones correspondientes en el plazo más corto posible.

II.1.9.4.1.2. Entorno o Marco

La necesidad de aplicar este plan de pruebas radica en la necesidad de obtener un software de calidad, que cumpla con los requerimientos del sistema DOMUINO, además que sea aprobado por la comunidad de usuarios finales des sistema.

II.1.9.4.1.3. Arquitectura Técnica

Para la realización de las pruebas se tomará en cuenta los módulos desarrollados en la fase de construcción y los diagramas realizados en la fase de elaboración.

II.1.9.4.1.4. Alcance

Este plan describe las pruebas del sistema, que serán aplicadas a los componentes del Sistema Informático para el Mejoramiento del control y monitoreo de inmuebles utilizando la tecnología web y el hardware de código abierto ARDUINO.

Se asume que antes de probar cada módulo habrá una revisión informal y solo el código que ha sido revisado como exitoso será probado.

Las unidades de prueba serán realizadas a través del programa manejador de pruebas que ejecuta el chequeo de los límites y las pruebas básicas de caja negra.

Las siguientes medidas de desempeño serán probadas:

- Validación correcta de las Entradas de Datos.
- Tiempo de respuesta de carga del Sitio Web.
- Tiempo de respuesta y Pool de conexiones a la Base de Datos.

II.1.9.4.1.5. Referencias

- Glosario del Proyecto.
- Plan de Desarrollo de Software.

II.1.9.4.2. Descripción de las pruebas planteadas

II.1.9.4.2.1. Descripción de las pruebas incluidas

Las siguientes pruebas serán realizadas a todo el Sistema de Informático:

- Pruebas de Pool de Conexiones.
- Pruebas funcionales.
- Pruebas de Validación de Datos.
- Pruebas de Perfiles de desempeño.

II.1.9.4.3. Planeación de la Ejecución de las Pruebas

La siguiente es una descripción de los tipos y técnicas de pruebas que se harán sobre el Sistema Informático, junto con la manera en la que éstas se ejecutarán.

II.1.9.4.4. Tipos de Técnicas y Pruebas

II.1.9.4.4.1. Pruebas de Pool de Conexiones

Las pruebas de pooling de conexiones serán realizadas para asegurar que los datos no son corruptos, dentro de las estructuras de datos internas. Estas pruebas son realizadas de forma independiente de la Interfaz de Usuario, en forma de pruebas de caja negra.

Objetivo de la Técnica:	Verificar la las conexiones realizadas a la base de datos y determinar el rendimiento del Sistema y del Motor de Base de Datos (PostgreSQL).
Técnica:	Analizar la carga a la conexión mediante el JDBC, simulando usuarios conectados.
Hitos:	Rastreo de cuello de botella en la conexión a la base de datos, realizada por el sistema.
Herramientas Requeridas:	Apache JMeter.
Criterios de Éxito:	Las conexiones se realizan con fluidez y una por usuario.

II.1.9.4.4.2. Pruebas Funcionales

Las pruebas funcionales serán realizadas para verificar que todos los requerimientos funcionales se cumplen satisfactoriamente. Estas serán cumplidas a través de pruebas de caja negra.

Objetivo de la Técnica:	Verificar requerimientos funcionales del sistema.
Técnicas:	Verificar que los requerimientos funcionales expuestos en los casos de uso se cumplen.
Hitos:	Casos de Uso del Sistema Informático.

Herramientas Requeridas:	Herramientas de backup y recuperación. Herramientas de Instalación y Monitoreo (registro, disco duro, CPU, memoria, etc.) Herramientas de Generación de Datos
Criterio de éxito:	Los siguientes elementos son probados exitosamente: <ul style="list-style-type: none"> - Todos los casos de uso del Sistema Informático. - La prueba funcional fue realizada juntamente con la persona encargada de su Administración.

II.1.9.4.4.3. Pruebas de validación de Datos

Las pruebas de validación de datos verifican la interacción del usuario con el software y la información que introduce. El objetivo de las pruebas de la interfaz de usuario es asegurar que dicha interfaz proporciona al usuario el acceso y validación a todos los datos introducidos por el Usuario con motivo de dar Integridad a la información.

Objetivo de la Técnica:	Verificar y probar la Interfaz de entrada Usuario - Sistema Verificar todas las posibles combinaciones de caracteres para entradas del sistema.
Técnica:	Crear o modificar pruebas para cada ventana para verificar todas las entradas de datos.
Hitos:	El testear verificara todas las entradas del Sistema.
Herramientas Requeridas:	-

Criterio de Éxito:	Todas las entradas han sido verificadas mediante combinaciones de caracteres para dar con los posibles errores del Sistema.
---------------------------	-----------------------------------------------------------------------------------------------------------------------------

II.1.9.4.4.4. Pruebas de Perfiles de Desempeño

Los perfiles de desempeño prueban tiempos de respuesta, tasas de transacción y otros requerimientos sensibles al tiempo son medidos y evaluados. El objetivo de los perfiles de desempeño es verificar el desempeño con el que los requerimientos se han cumplido. Un perfil de desempeño es implementado y ejecutado para refinar el desempeño y comportamiento de los elementos objetivo de las pruebas.

Objetivo de la Técnica:	Determinar el comportamiento del objeto de la prueba bajo: <ul style="list-style-type: none"> - Carga de trabajo anticipada. - Peor caso de la carga de trabajo.
Técnica:	Concurrencia de visita al Sistema Informático. Prueba de Stress al servidor Web.
Hitos:	Cálculo de tiempos de software.
Herramientas requeridas:	JMeter
Criterio de éxito:	Los tiempos de respuesta son los esperados para el número de usuarios que acceden al Sistema Informático.

II.1.9.4.4.5. Ejecución de Pruebas

II.1.9.4.4.5.1. Demostración y Resultado de la Prueba de la Validación de Datos

II.1.9.4.4.5.2. Partición Equivalente: Casos de Prueba Caja Negra

Entrada de Datos Ingresar al Sistema

Usuario: Alfanumérico 30 caracteres

Contraseña: Alfanumérico 30 caracteres

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas
Tipo Usuario Tamaño Usuario	SI	1.- Alfanumérico 2.- $1 \leq \text{caracteres} \leq 30$	3.- Vacío 4.- $\text{caracteres} > 30$
Tipo Contraseña Tamaño Contraseña	SI	5.- Alfanumérico 6.- $1 \leq \text{caracteres} \leq 30$	7.- Vacío 8.- $\text{caracteres} > 30$

Casos de Prueba Validas

CP1	
Usuario	Contraseña
administrador	administrador

Cubre las clases de equivalencias válidas: 1-2-5-6

Casos de Prueba Inválidas

CP2	
Usuario	Contraseña
	administrador

Cubre las clases de equivalencias inválidas: 3

CP3	
Usuario	Contraseña
administrador	

Cubre las clases de equivalencias inválidas: 7

Entrada de Datos Formulario ABM Gestionar Usuario

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas
Tipo Nombre Tamaño Nombre	SI	1.- Alfabético 2.- 1<=caracteres=>30	3.- Vacío 4.- caracteres>30
Tipo Apellido Paterno Tamaño Apellido Paterno	SI	5.- Alfabético 6.- 1<=caracteres=>30	7.- Vacío 8.- caracteres>30
Tipo Apellido Materno Tamaño Apellido Materno	No	9.- Alfabético 10.- 1<=caracteres=>30	11.- Vacío 12.- caracteres>30
Tipo Nombre de Usuario Tamaño Nombre de Usuario	SI	13.- Alfanumérico 14.- 1<=caracteres=>30	15.- Vacío 16.- caracteres>30
Tipo Contraseña Tamaño Contraseña	SI	17.- Alfanumérico 18.- 1<=caracteres=>30	19.- Vacío 20.- caracteres>30
Tipo Rol Tamaño Rol	SI	21.- Administrador, Cliente	
Tipo Dirección Tamaño Dirección	SI	22.- Alfanumérico 23.- 1<=caracteres=>50	24.- Vacío 25.- caracteres>50
Tipo Foto Tamaño Foto	SI	26.- Alfanumérico 27.- 1<=caracteres=>50	28.- Vacío 29.- caracteres>50
Tipo Fecha Tamaño Fecha	SI	30.- Fechas 31.- DD /MM/ AAAA	32.- Vacío 33.- Cualquier fecha fuera del formato
Tipo Email Tamaño Email	SI	34.- Alfanumérico 35.- 1<=caracteres=>30	36.- no existe 37.- caracteres>30

Caso de Prueba Validas

CP1								
Nombre	Apellido Paterno	Apellido Materno	Usuario	Contraseña	Rol	Foto	Fecha	Email
Jahzeel	Rodas	Flores	Jahzeel	Rodas	Administrador	C:\Users\Public\Pictures\Sample Pictures\A	11/02/2015	jahzeel@gmail.com

Cubre las clases de equivalencias validas: 1-2-5-6-9-10-13-14-17-18-21-22-23-26-27-30-31-34-35

Caso de Prueba Validas

CP2								
Nombre	Apellido Paterno	Apellido Materno	Usuario	Contraseña	Rol	Foto	Fecha	Email
Jahzeel	Rodas		Jahzeel	Rodas	Administrador	C:\Users\Public\Pictures\Sample Pictures\A	11/02/2015	jahzeel@gmail.com

Cubre las clases de equivalencias validas: 1-2-5-6-13-14-17-18-21-22-23-26-27-30-31-34-

35

Caso de Prueba Inválidas

CP2								
Nombre	Apellido Paterno	Apellido Materno	Usuario	Contraseña	Rol	Foto	Fecha	Email
	Rodas		Jahzeel		Administrador	C:\Users\Public\Pictures\Sample Pictures\A		jahzeel@gmail.com

Cubre las clases de equivalencias inválidas: 3-19-32

Entrada de Datos Formulario ABM Gestionar Inmuebles

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas
Tipo Nombre Tamaño Nombre	SI	1.- Alfanumérico 2.- 1<=caracteres=>30	3.- Vacío 4.- caracteres>30
Tipo Número de Pisos Tamaño Número de Pisos	SI	5.- Numérico 6.- 1<=caracteres=>2	7.- Letras 8.- caracteres>2
Tipo Imagen Tamaño Imagen	SI	9.- Alfanumérico 10.- 1<=caracteres=>30	11.- Vacío 12.- caracteres>30
Tipo Cod_Usuario Tamaño Cod_Usuario	SI	13.- Numérico 14.- 1<=caracteres=>2	15.- Vacío 16.- caracteres>2
Tipo Fecha Tamaño Fecha	SI	17.- Fechas 18.- DD/MM/AAAA	19.- Cualquier fecha fuera del formato.

Caso de Prueba Validas

CP1				
Nombre	Número de Pisos	Imagen	Cod_usuario	Fecha
Casa Rodas	1	C:\Users\Public\Pictures\Sample Pictures\	10	11/02/2015

Cubre las clases de equivalencias validas: 1-2-5-6-9-10-13-14-17-18

Caso de Prueba Inválidas

CP2				
Nombre	Número de Pisos	Imagen	Cod_usuario	Fecha
	cl	C:\Users\Public\Pictures\Sample Pictures\	10	11/02/2015

Cubre las clases de equivalencias inválidas: 3-7

Entrada de Datos Formulario ABM Gestionar Dispositivos

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas
Tipo Nombre Tamaño Nombre	SI	1.- Alfanumérico 2.- 1<=caracteres=>20	3.- Vacío 4.- caracteres>20
Tipo Imagen Tamaño Imagen	SI	5.- Alfanumérico 6.- 1<=caracteres=>50	7.- Vacío 8.- caracteres>50
Tipo Descripción Tamaño Descripción	SI	9.- Alfanumérico 10.- 1<=caracteres=>30	11.- Vacío 12.- caracteres>30
Tipo Cod_Cuarto Tamaño Cod_cuarto	SI	13.- Numérico 14.- 1<=caracteres=>3	15.- Letras 16.- caracteres>3
Tipo Fecha Tamaño Fecha	SI	17.- Fechas 18.- DD /MM/ AAAA	19.- Cualquier fecha fuera del formato.
Tipo Tipo_Dispositivo Tamaño Tipo_Dispositivo	SI	20.- Numérico 21.- caracteres=1	22.- Letras 23.- caracteres>1
Tipo Valor_Arduino_1 Tamaño Valor_Arduino_1	SI	24.- Alfanumérico 25.- 1	26.- Vacío 27.- caracteres>1
Tipo Valor_Arduino_0 Tipo Valor_Arduino_0	SI	28.- Alfanumérico 29.- 1	30.- No existe 31.- caracteres>1

Caso de Prueba Validas

CPI							
Nombre	Imagen	Descripción	Cod_cuarto	Fecha	Tipo Dispositivo	Valor_Arduino_1	Valor_Arduino_0
Switch	C:\Users\Public\Pictures\Sample	Nos permite prender y apagar	15	15/02/2015		1	0

	Pictures\ a	dispositivos			Encendido - Apagado		
--	----------------	--------------	--	--	------------------------	--	--

Cubre las clases de equivalencias validas: 1-2-5-6-9-10-13-14-17-18-20-21-24-25-28-29

Caso de Prueba invalidas

CP2							
Nombre	Imagen	Descripcion	Cod_cuarto	Fecha	Tipo Dispositivo	Valor_Arduino_1	Valor_Arduino_0
	C:\Users\ Public\Pictures\ Sample Pictures\ a	Nos permite prender y apagar dispositivos	15	15/02/2015	asd	1	0

Cubre las clases de equivalencias inválidas: 3-22

Entrada de Datos Listar Usuarios por Tipo de Usuario

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas
Tipo Rol Tamaño Rol	SI	1.- Todos los usuarios, Administradores, Clientes	2.- Vacío

Entrada de Datos Listar Usuarios por Rango de Fecha de Registro

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas
Tipo Rol	SI	1.- Todos los usuarios, Administradores, Clientes	2.- No existe
Tipo Fecha_Inicio	SI	3.- Fechas 4.- DD/MM/AAAA	5.- Cualquier fecha fuera del formato.
Tipo Fecha_Final	SI	6.- Fechas 7.- DD/MM/AAAA	8.- Cualquier fecha fuera del formato.

Entrada de Datos Listar Inmuebles por Rango de Fecha de Registro

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas
Tipo Fecha_Inicio	SI	1.- Fechas 2.- DD/MM/AAAA	3.- Cualquier fecha fuera del formato.
Tipo Fecha_Final	SI	4.- Fechas 5.- DD/MM/AAAA	6.- Cualquier fecha fuera del formato.

Entrada de Datos Listar Inmuebles por Usuario

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas
Tipo Usuario	SI	1.- Alfanumérico 2.- Valor>1	3.- No existe 4.- Valor<=0

Entrada de Datos Listar Inmuebles

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas
Tipo Inmueble	SI	1.- Alfanumérico 2.- Valor>1	3.- No existe 4.- Valor<=0

Entrada de Datos Listar Dispositivos Inmueble

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas

Tipo Inmueble	SI	1.- Alfanumérico 2.- Valor>1	3.- No existe 4.- Valor<=0
---------------	----	---------------------------------	-------------------------------

Entrada de Datos Listar Dispositivos Usuario

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas
Tipo Usuario	SI	1.- Alfanumérico 2.- Valor>1	3.- No existe 4.- Valor<=0

Entrada de Datos Listar Dispositivos Fecha Inmueble

Condiciones externas	Obligatorio	Clase de Equivalencia válidas	Clase de Equivalencia inválidas
Tipo Inmueble	SI	1.- Alfanumérico 2.- Valor>1	3.- No existe 4.- Valor<=0
Tipo Fecha	SI	3.- Fechas 4.- DD/MM/AAAA	5.- Cualquier fecha fuera del formato.

II.10.3.4.5.3. Prueba de Perfil de Desempeño

El componente principal de JMeter es denominado *Plan de Prueba* o *Test Plan*, en él se definen todos los aspectos relacionados con una prueba de carga, como : parámetros empleados por requisición, tipo de reportes a generarse con los resultados obtenidos, la posible reutilización de requisiciones compuestas por usuarios, entre otros aspectos.

A continuación se ilustra paso a paso un *Plan de Prueba* utilizado para simular una carga de 50 usuarios solicitando la página principal en determinado sitio.

Estando en la interface principal de JMeter (ilustrada anteriormente), en la columna izquierda debe observar un icono llamado Test Plan -- selecciónelo -- al llevar a cabo este paso, en la ventana derecha aparecerán varias opciones, aquella encontrada en la parte superior le permite asignar un nombre a su plan de prueba, defínalo a su criterio. El resto de las opciones representan funcionalidades avanzadas que no serán descritas para esta simulación.

Definido el nombre, colóquese nuevamente en el icono Test Plan y oprima el botón derecho de su "mouse", del menú generado seleccione la opción Añadir -- Grupo de Hilos; para

efectos prácticos un "Grupo de Hilos" es considerado el grupo de usuarios que desea simular para su aplicación.

Ahora seleccione el icono Grupo de Hilos recientemente creado. Al llevar a cabo este paso la ventana derecha mostrará la siguiente serie de opciones:

Nombre: Utilizado para definir un nombre más descriptivo sobre el grupo de usuarios, como: "Usuarios Esperados".

Número de hilos: Equivale al número de usuarios que se desean simular, en este caso utilizaremos 50.

Periodo de subida: Es el lapso de tiempo en segundos que se desea tener entre cada grupo de usuarios ("Grupo de Hilos"), utilizaremos 15.

Contador de Bucles: Utilizado para indicar si la simulación para grupos de usuarios ("Grupo de Hilos") será llevada a cabo infinitamente, esto es, si selecciona esta opción indica que desea simular 50 usuarios, esperar 15 segundos ("Periodo de subida"), simular otros 50 usuarios y así sucesivamente. Para esta prueba es recomendable defina 15 ciclos para simular un total de 750 usuarios en cuestión de 150 segundos.

Planificador: Finalmente, esta opción permite definir tiempos de arranque para determinados grupos de usuarios ("Grupo de Hilo "), para efectos prácticos de esta simulación no será seleccionada esta opción y se iniciará la prueba a nuestra discreción.

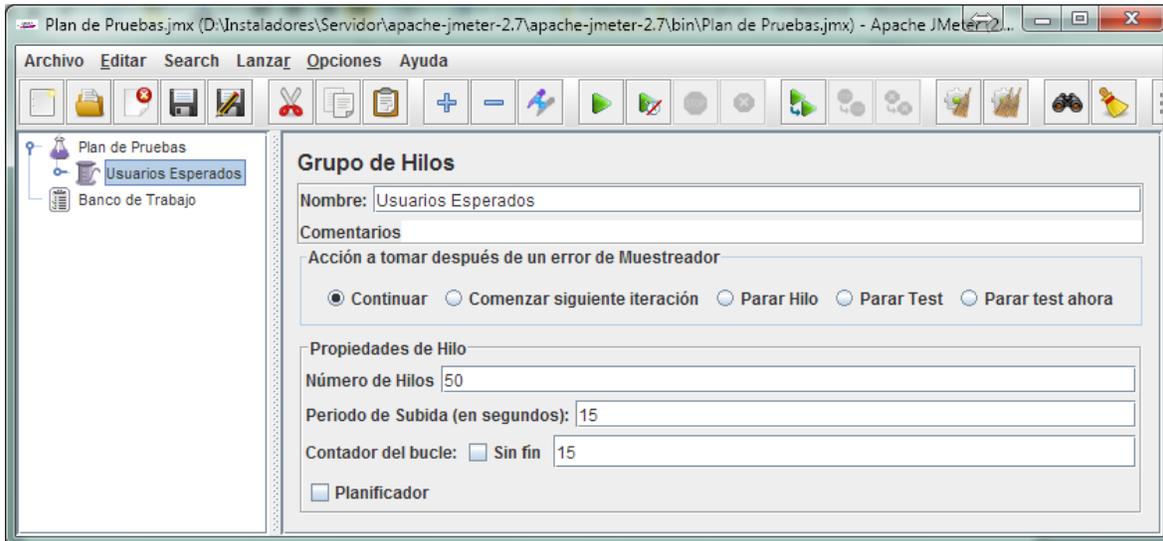


Figura 72 Plan de Pruebas usuarios esperados

Una vez definidas las características del grupo de usuarios ("Grupo de Hilos"), colóquese nuevamente en este icono de la columna izquierda y ahora seleccione la opción Añadir -- Muestreador – Petición HTTP. Lo anterior genera un icono denominado Petición HTTP utilizado para definir las requisiciones de simulación. Si selecciona este último icono aparecerán las siguientes opciones en la ventana derecha:

Nombre de Servidor o IP: Empleado para definir la dirección I.P o nombre del servidor donde será llevada a cabo la prueba de carga, se utilizará 127.0.0.1 para indicar un servidor local.

Número de Puerto: Define el puerto TCP de operación del servidor, será empleado 8080.

Ruta: Utilizado para definir la ruta de acceso para llevar a cabo la prueba, será definido /Domotica/, que es la página principal de nuestro sistema web.

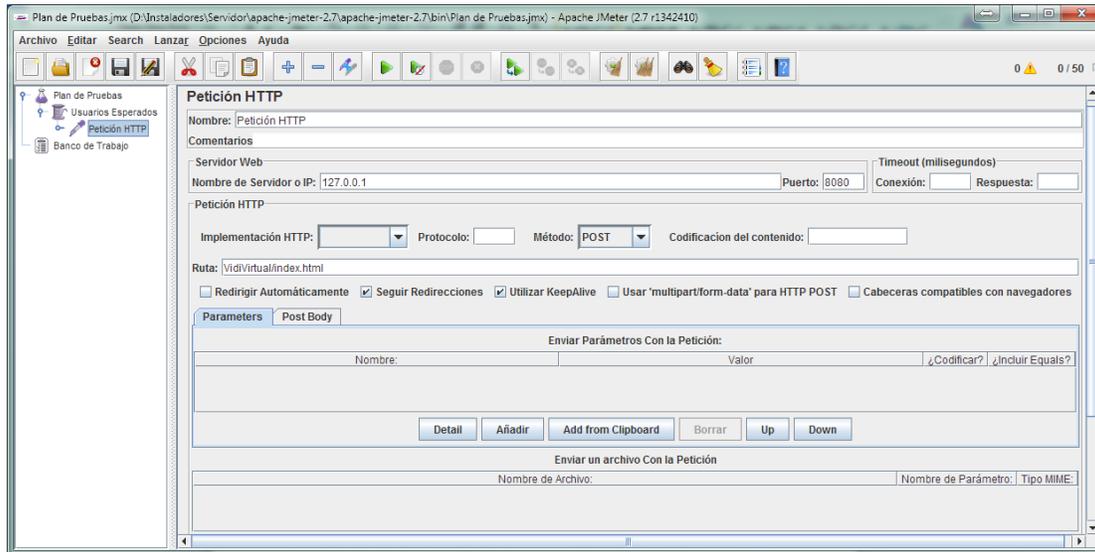


Figura 73 Petición HTTP

- Ahora debe colocarse en el icono de HTTP Request y oprimir el botón derecho de su "mouse", del menú generado seleccione la opción Añadir -- Receptor – Gráfico de Resultados, así como Añadir -- Receptor – Vista de Resultados en Árbol
- Finalmente guarde su plan de prueba ("Plan Pruebas") y ejecútelo seleccionando la opción Arrancar del menú superior. Si se coloca en el icono Gráfica de Resultados mientras se está llevando a cabo la simulación, puede observar como es construida interactivamente la gráfica de resultados.

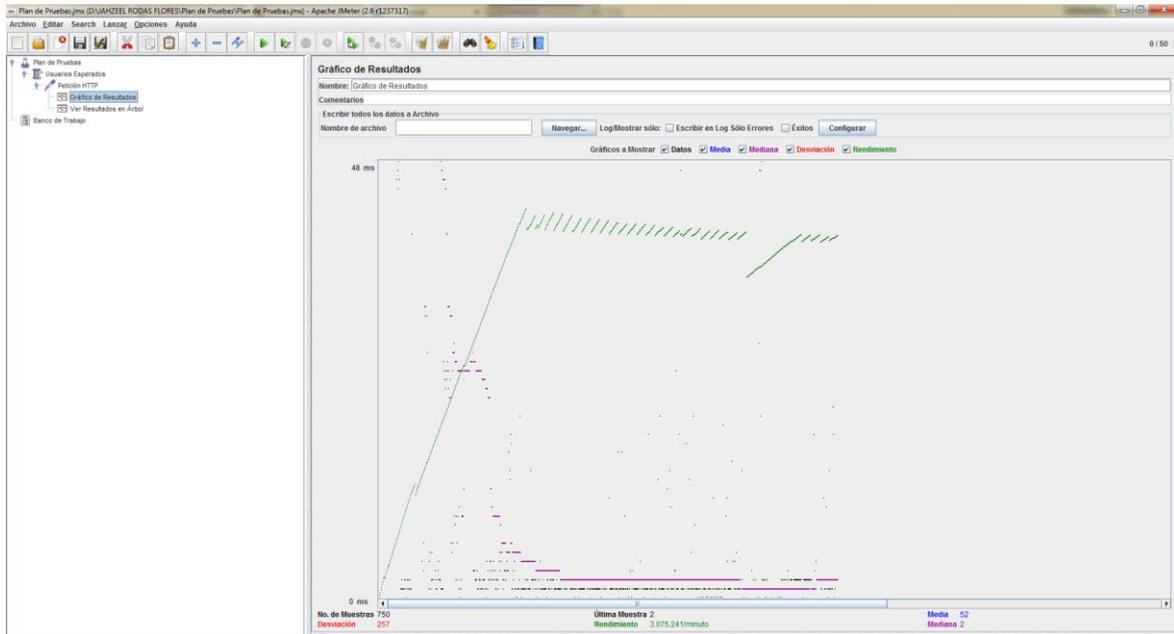


Figura 74 Grafico de resultados DOMUINO

El Resultado Indica que la Página Web DOMUINO soporta 3,075 Usuarios en un minuto.

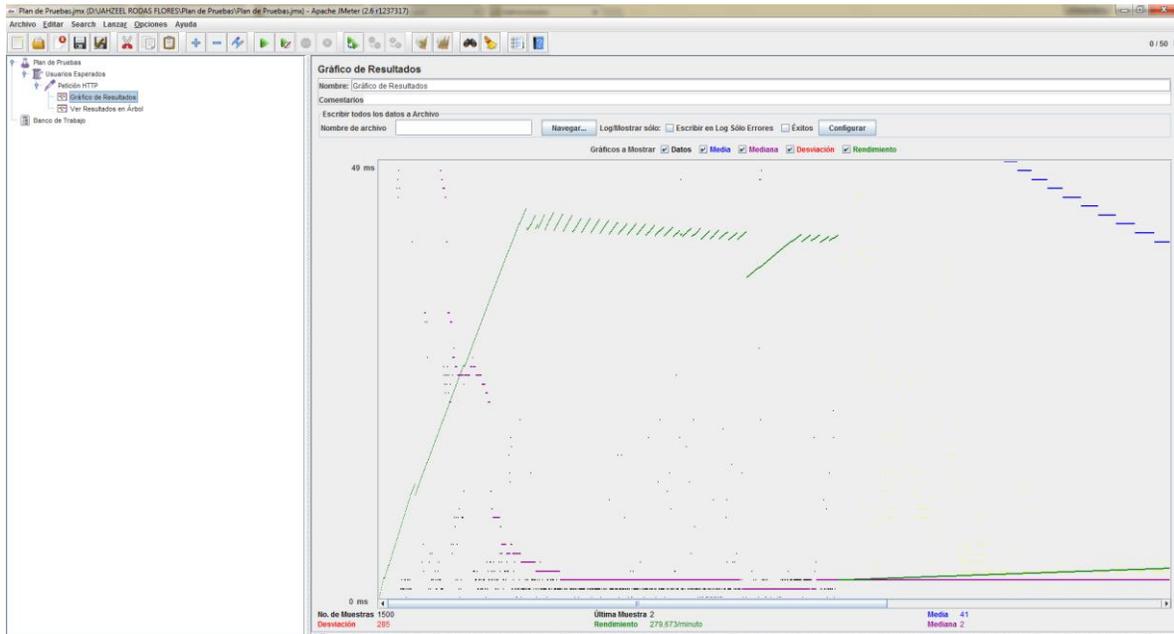


Figura 75 Grafico de resultados: administrador

El Resultado Indica que la Página Web DOMUINO/administrador soporta 279,673 Usuarios en un minuto.

II.10.3.4.5.4. Prueba de Pool de Conexiones

La prueba también se realiza con la herramienta JMeter, y su configuración es la siguiente:

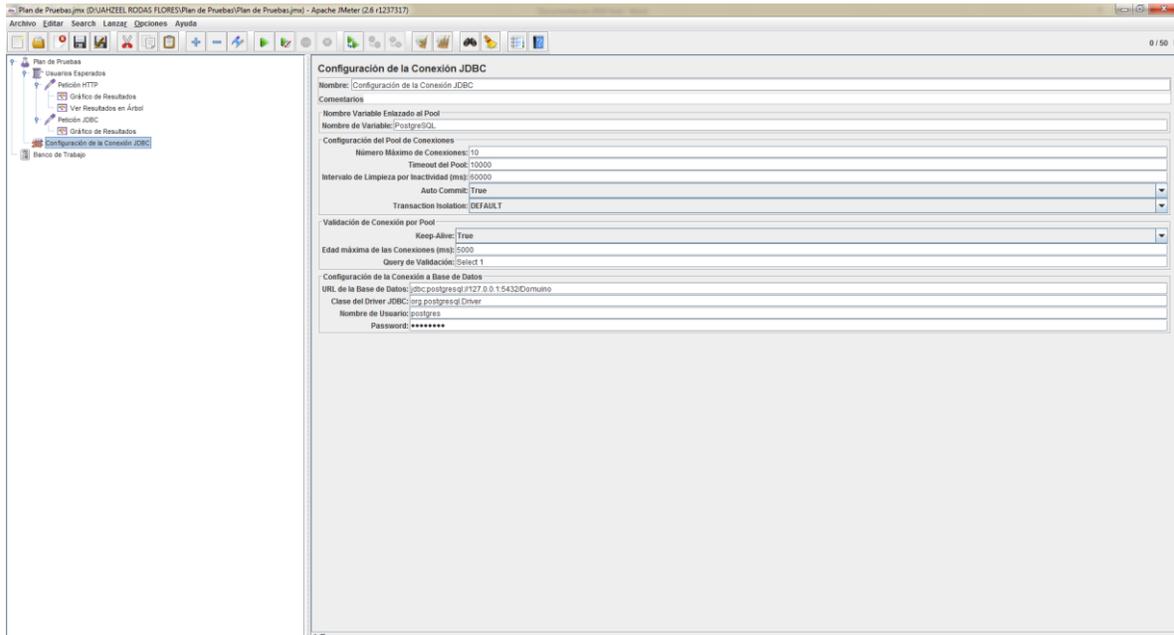


Figura 76 Configuración de la conexión JDBC

En la cual se probara la siguiente consulta:

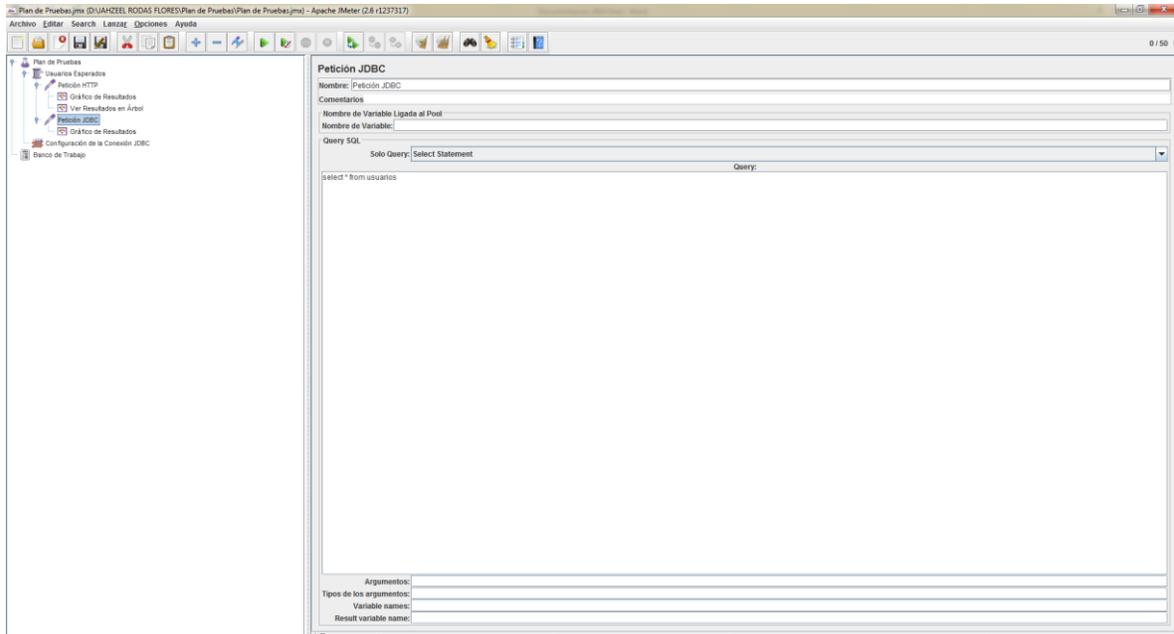


Figura 77 Petición JDBC

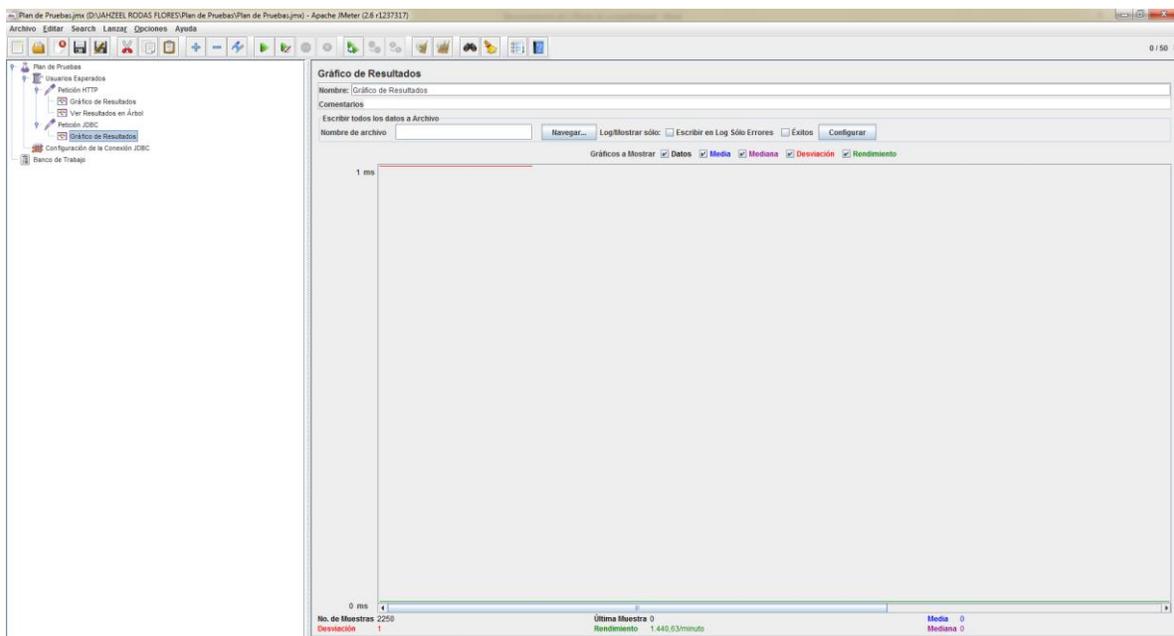


Figura 78 Grafico de resultados JDBC

Por conexión se puede realizar la consulta a una velocidad de 1.440,63 consultas por minuto.

II.1.10. Diseño de Interfaces de Usuario

II.1.10.1. Introducción

Se trata de prototipos que permiten al usuario tener una idea sobre las interfaces que proveerá el sistema.

II.1.10.1.1. Propósito

Presentar los prototipos de pantallas para que el usuario tenga una idea de la interfaz que se presentaran en el Sistema.

II.1.10.1.2. Alcance

Mostrar los Prototipos de pantallas, sujeto a modificaciones a los largo del desarrollo del sistema.

II.1.10.2. Prototipo de Pantallas

II.1.10.2.1. Ingreso de Usuario al Sistema

II.1.10.2.1.1. Pantalla Validar Usuario

La siguiente pantalla permite el acceso de los usuarios al sistema, caso contrario no podrán acceder al sistema.



Figura 79 Pantalla: Validar usuario

II.1.10.2.1.2. Pantalla Selección de Menú

Pantalla que le permite al usuario elegir procesos según el rol que le es asignado.



Figura 80 Pantalla: Selección de menu

II.1.10.2.1.3. Pantalla Gestionar Usuarios

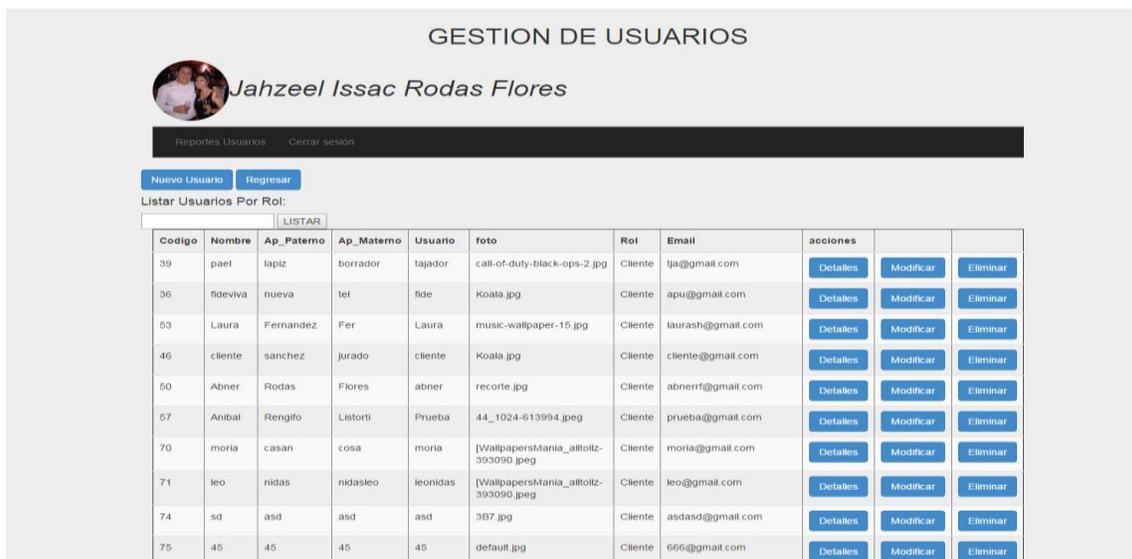


Figura 81 Pantalla: Gestionar usuarios

II.1.10.2.1.4. Pantalla Seleccionar Usuarios por Tipo



Figura 82 Pantalla: Seleccionar usuarios por tipo

II.1.10.2.1.5. Pantalla Registrar Usuario

The screenshot shows the 'Formulario de Registro' (Registration Form) with the following fields and labels:

- Nombre de Usuario:** Input field containing 'Usuario'.
- Direccion de Correo Electronico:** Input field containing 'correo@ejemplo.com'.
- Nombre:** Input field containing 'Nombre (s)'.
- Apellido Paterno:** Input field containing 'Apellido Paterno'.
- Apellido Materno:** Input field containing 'Apellido Materno'.
- Contraseña:** Input field containing 'Clave de acceso'.
- Direccion:** Input field containing 'Referencias de Localizacion'.
- Fecha:** Input field containing 'dd/mm/aaaa'.
- Seleccionar Rol:** Input field.
- Fotografia:** Input field with a 'Choose file' button next to it.

A blue 'Crear cuenta' (Create account) button is located at the bottom of the form.

Figura 83 Pantalla: Registrar Usuarios

II.1.10.2.1.6. Pantalla Modificar Usuario

Codigo de Usuario:
85

Nombre de Usuario:
abner

Email:
abner.rodas41@gmail.com

Nombre:
abner

Apellido Paterno:
rodas

Apellido Materno:
flores

Apellido Contraseña:
31035

Descripcion:
Referencias de Localizacion

Activo Desde :miércoles 16 de septiembre de 2015
dd/mm/aaaa

Foto:

Choose file

Modificar Regresar

Figura 84 Pantalla: Modificar usuario

II.1.10.2.1.7. Pantalla Reportes Usuarios

GESTION DE USUARIOS

 Jahzeel Issac Rodas Flores

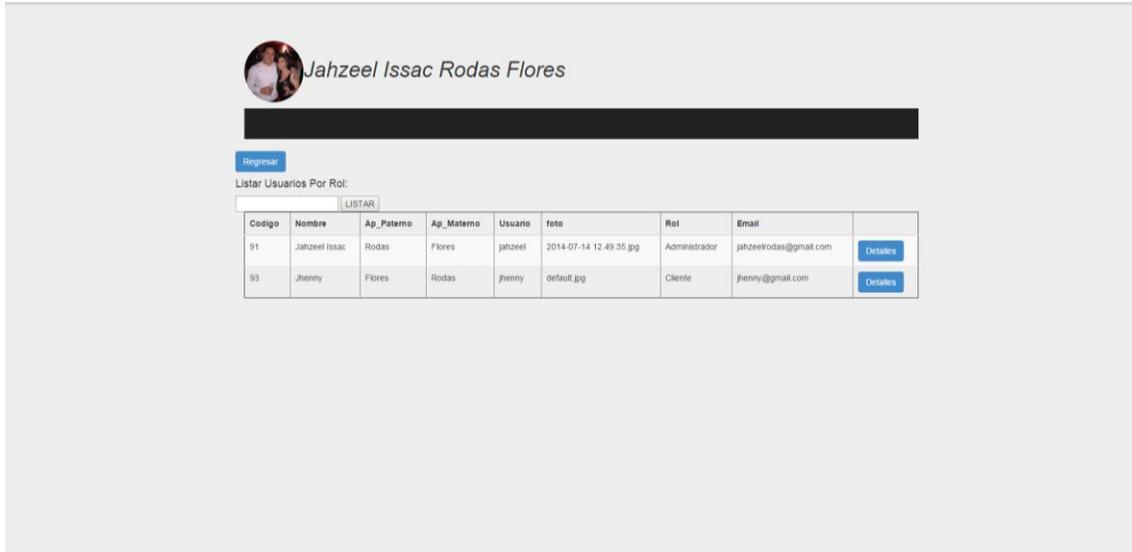
Reporte Gral. De Usuarios
Mostrar

Usuarios por fecha de registro
Mostrar

REGRESAR

Figura 85 Pantalla: Reportes usuarios

II.1.10.2.1.8. Pantalla Reporte General de Usuarios



Regresar

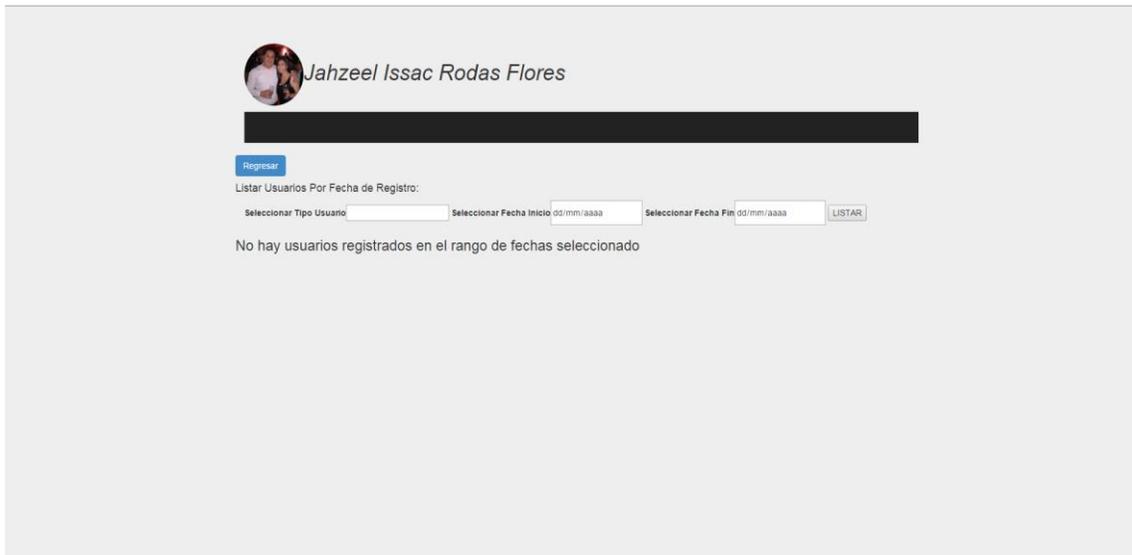
Listar Usuarios Por Rol:

LISTAR

Codigo	Nombre	Ap_Paterno	Ap_Materno	Usuario	foto	Rol	Email	
91	Jahzeel Issac	Rodas	Flores	jahzeel	2014-07-14 12:49:35.jpg	Administrador	jahzeelrodas@gmail.com	Detalles
93	Jhenny	Flores	Rodas	jhenny	default.jpg	Cliente	jhenny@gmail.com	Detalles

Figura 86 Pantalla: Reporte general de usuarios

II.1.10.2.1.9. Pantalla Reporte de Usuarios por Fecha de Registro



Regresar

Listar Usuarios Por Fecha de Registro:

Seleccionar Tipo Usuario: Seleccionar Fecha Inicio: Seleccionar Fecha Fin: LISTAR

No hay usuarios registrados en el rango de fechas seleccionado

Figura 87 Pantalla: Reporte de usuarios por fecha de registro

II.1.10.2.1.10. Pantalla Gestión Inmuebles

GESTION INMUEBLES



Jahzeel Issac Rodas Flores

[Reportes Inmuebles](#) [Cerrar sesión](#)

[Registrar Inmueble](#) [Regresar](#)

Codigo	Nombre	Numero de Pisos	Imagen	Cod Usuario	Estado	Fecha	acciones
5	Casa de Abner	1	IMG-20140919-WA0003.jpg	50	true	2014-11-14	Seleccionar Modificar Eliminar
6	Casa	1	la bohemia.jpg	50	true	2014-11-20	Seleccionar Modificar Eliminar
7	Table	1	[WallpapersMania_allitoliz-393090.jpeg	50	true	2015-08-27	Seleccionar Modificar Eliminar
8	Maqueta	1	387.jpg	53	true	2015-09-02	Seleccionar Modificar Eliminar
4	Amanecida	2		39	true	2014-09-20	Seleccionar Modificar Eliminar
9	Huellitas	4	310327.jpg	53	true	2015-09-12	Seleccionar Modificar Eliminar
11	Prueba	1	Como-vender-una-casa-rapidamente-5.jpg	83	true	2015-09-18	Seleccionar Modificar Eliminar

Figura 88 Pantalla: Gestión inmuebles

II.1.10.2.1.11. Pantalla Registrar Inmueble

Registro de Inmuebles

Nombre de Inmueble:

Numero de Pisos o Plantas:

Imagen:

Nombre de Usuario:

Fecha de Registro:

Figura 89 Pantalla: Registrar inmueble

II.1.10.2.1.12. Pantalla Modificar Inmueble

Usuario : *Jahzeel Issac Rodas Flores*

Codigo de Inmueble:

Nombre de Inmueble:

Numero de Pisos:

Foto:


Usuario Asignado:

Fecha de Registro: **Viernes 18 de septiembre de 2015**

Figura 90 Pantalla: Modificar inmueble

II.1.10.2.1.13. Pantalla Seleccionar Inmueble

 *Jahzeel Issac Rodas Flores*

Bienvenido a:
Prueba Propiedad de jazel rodas velasquez

Codigo	Nombre	Descripcion	Piso	Casa	acciones
17	Cocina Prueba	Breve Descripcion	1	11	<input type="button" value="Seleccionar"/> <input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>

Figura 91 Pantalla Seleccionar inmueble

II.1.10.2.1.14. Pantalla Registrar Ambiente – Cuarto



Registro de Cuartos o Ambientes

Nombre de Cuarto:

Descripcion:

Piso o Planta:

Casa o Inmueble: Prueba

Fecha de Registro:

Figura 92 Pantalla: Registrar ambiente cuarto

II.1.10.2.1.15. Pantalla Reportes Inmuebles



GESTION DE INMUEBLES

 *Jahzeel Issac Rodas Flores*

 Reporte Gral. De Inmuebles

 Inmueble por Fecha

 Inmueble Usuario

Figura 93 Pantalla: Reportes inmuebles

II.1.10.2.1.16. Pantalla Reporte General de Inmuebles

GESTION DE INMUEBLES


Jahzeel Issac Rodas Flores

[Redacted]

[Regresar]

Codigo	Nombre	Usuario	Imagen	Estado	fecha	Pisos	acciones
15	casa de las panosas	.jhenny	Como-vender-una-casa-rapidamente-5.jpg	Activo	2016-01-23	1	[Detalles]

Figura 94 Pantalla: Reporte general de inmuebles

II.1.10.2.1.17. Pantalla Reporte de Inmuebles Por Fecha de Registro


Jahzeel Issac Rodas Flores

[Redacted]

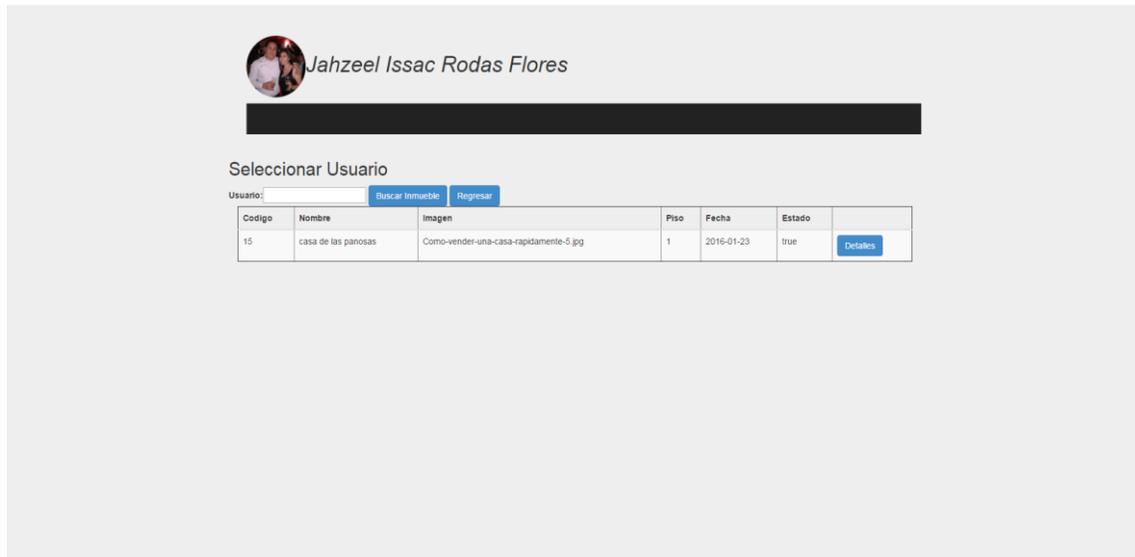
[Regresar]

Listar Inmuebles Por Fecha de Registro:

Codigo	Nombre	Ap_Paterno	Ap_Materno	Usuario	foto	Rol	Email
15	casa de las panosas	.jhenny	Activo	2016-01-23	1	true	[Detalles]

Figura 95 Pantalla: Reporte de inmuebles por fecha de registro

II.1.10.2.1.18. Pantalla Reporte de Inmuebles por Usuario



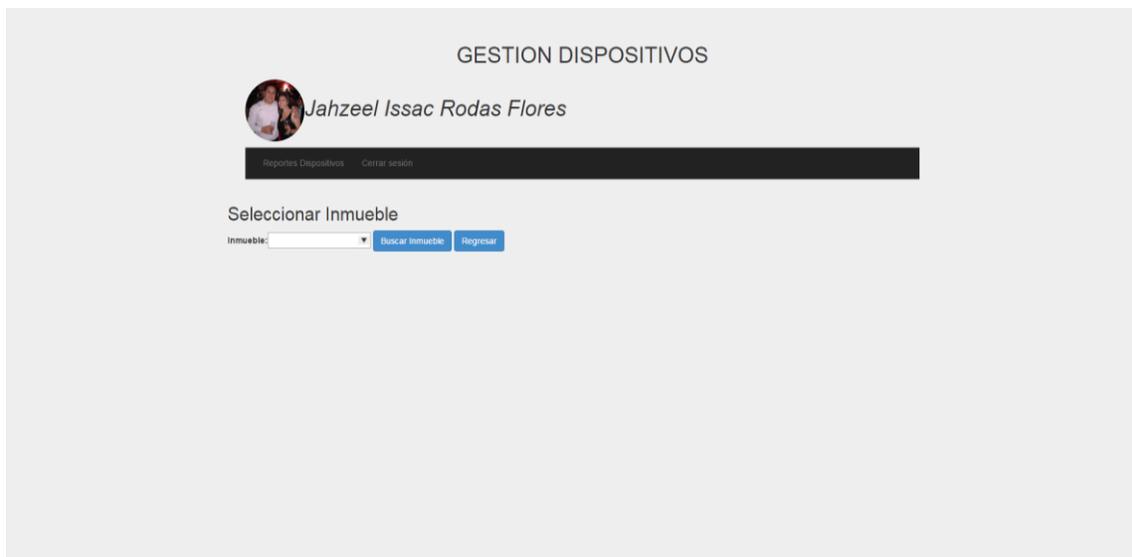
Seleccionar Usuario

Usuario: [Buscar Inmueble](#) [Regresar](#)

Codigo	Nombre	Imagen	Piso	Fecha	Estado	
15	casa de las panosas	Como-vender-una-casa-rapidamente-5.jpg	1	2016-01-23	true	Detalles

Figura 96 Pantalla: Reporte inmuebles por usuario

II.1.10.2.1.19. Pantalla Gestión Dispositivos



GESTION DISPOSITIVOS

Jahzeel Issac Rodas Flores

[Reportes Dispositivos](#) [Cerrar sesión](#)

Seleccionar Inmueble

Inmueble: [Buscar Inmueble](#) [Regresar](#)

Figura 97 Pantalla: Gestión dispositivos

II.1.10.2.1.20. Pantalla Reportes Dispositivos

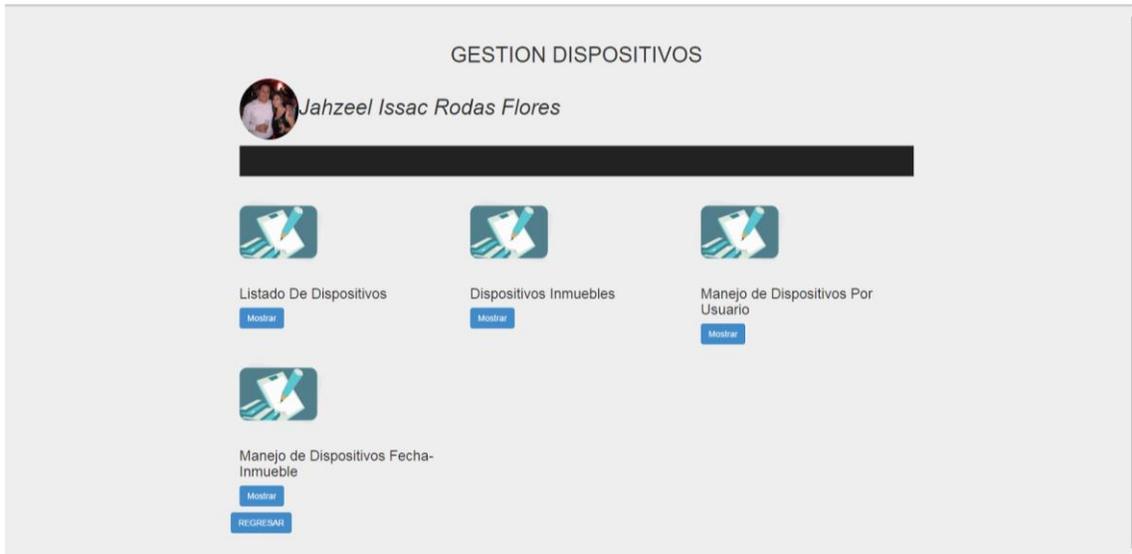


Figura 98 Pantalla: Reportes dispositivos

II.1.10.2.1.21. Pantalla Reporte Listado de Dispositivos

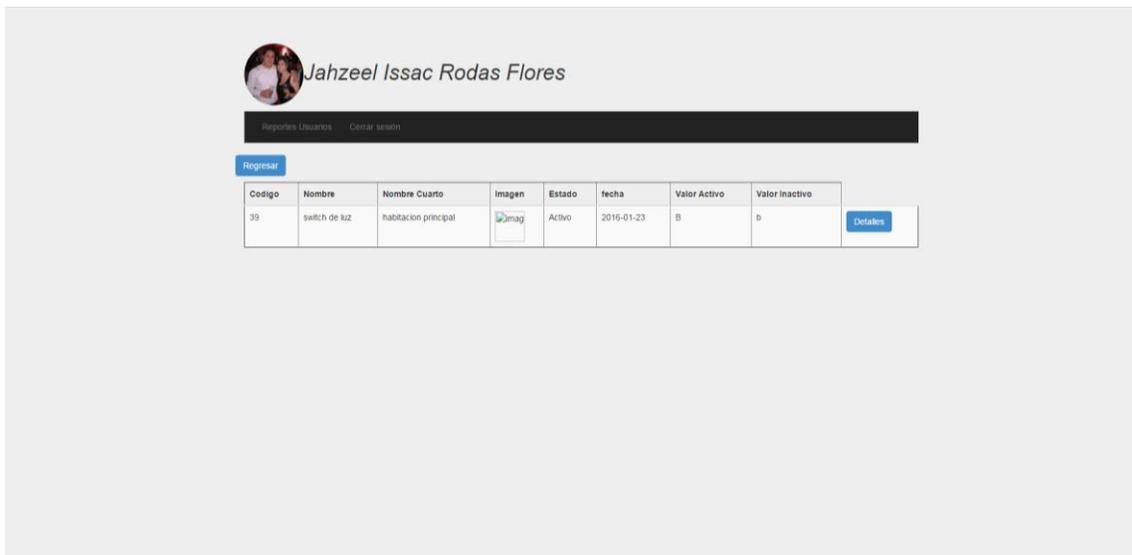


Figura 99 Pantalla: Reporte listado de dispositivos

II.1.10.2.1.22. Pantalla Reporte Dispositivos Inmueble

GESTION DISPOSITIVOS


Jahzeel Issac Rodas Flores

Cerrar sesion

Cuarto o Ambiente: "habitacion principal"

[Registrar Dispositivo](#)
[Regresar](#)

Codigo	Nombre	Imagen	Descripcion	Cod_Cuarto	Arduino Activo	Arduino Inactivo	acciones
39	switch de luz		Breve Descripcion	28	B	D	Seleccionar Modificar Eliminar

Figura 100 Pantalla: Reporte dispositivos inmueble

II.1.10.2.1.23. Pantalla Reporte Manejo de Dispositivos por Usuario

GESTION DISPOSITIVOS


Jahzeel Issac Rodas Flores

Seleccionar Usuario

Usuario:
[Buscar](#)
[Regresar](#)

No existe registro del manejo de dispositivos para el usuario seleccionado

Figura 101 Pantalla: Reporte manejo de dispositivos por usuario

II.1.10.2.1.24. Pantalla Manejo Dispositivos Fecha Inmueble

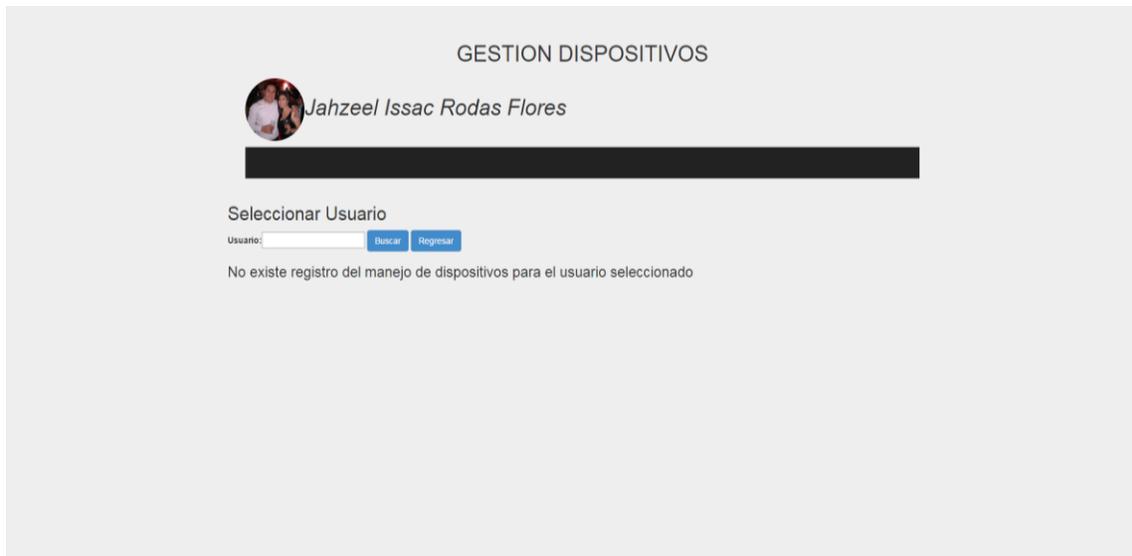


Figura 102 Pantalla: Manejo de dispositivos fecha inmueble

II.1.10.2.1.25. Pantalla Monitoreo de Inmuebles



Figura 103 Pantalla: Monitoreo de inmuebles

II.1.10.2.1.26. Pantalla Cerrar Sesión

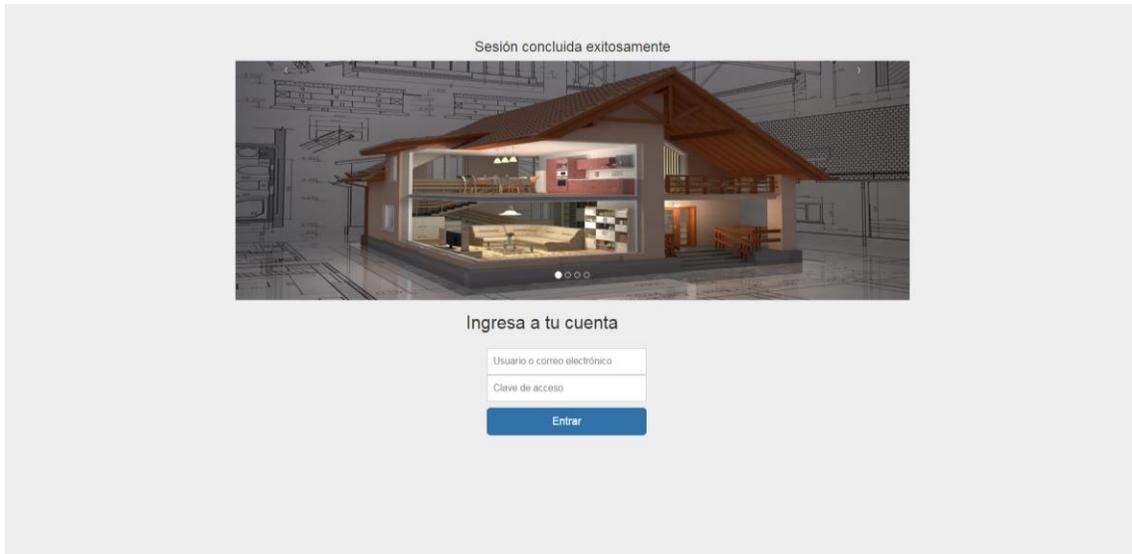


Figura 104 Pantalla: Cerrar sesión

II.1.10.2.1.27. Pantalla Cliente

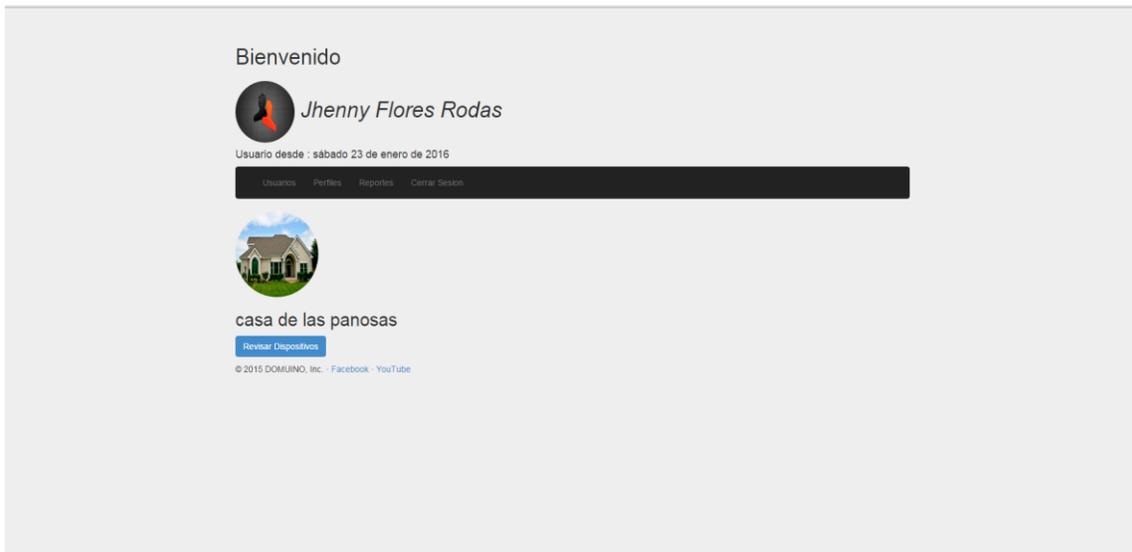


Figura 105 Pantalla: Cliente

II.1.10.2.1.28. Pantalla Usuarios – Clientes

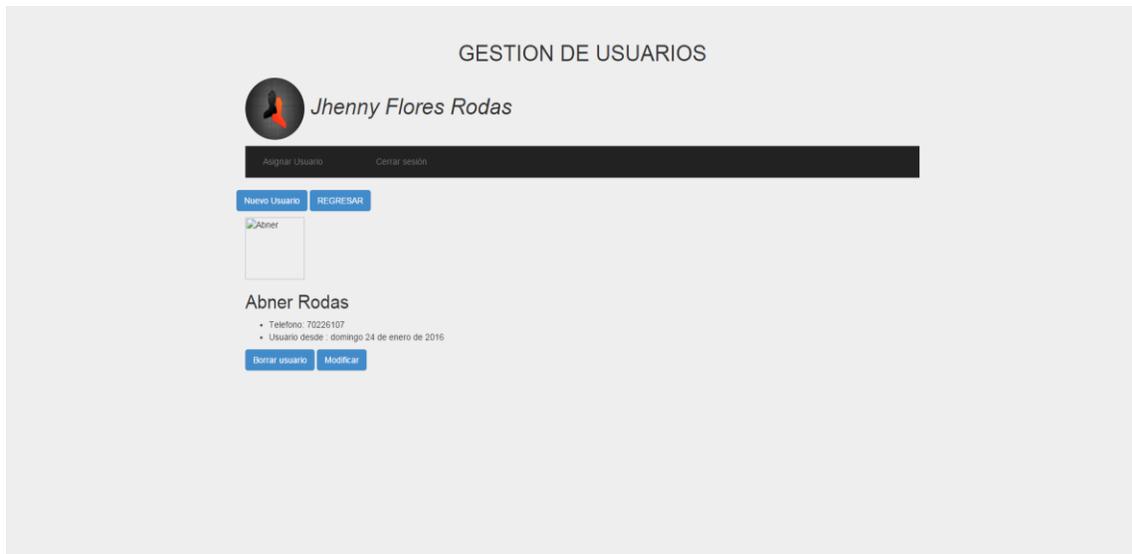


Figura 106 Pantalla: Usuarios clientes

II.1.10.2.1.29. Pantalla Crear Cuenta Usuario cliente

The screenshot shows a "Registro de Usuario" form. At the top, it displays "Usuario : Jhenny Flores Rodas". The form fields are as follows:

- Nombre:** Input field with "Nombre" label.
- Apellidos:** Input field with "Apellidos" label.
- Contraseña:** Input field with "Contraseña" label.
- Edad:** Input field with "Edad" label.
- Telefono:** Input field with "Telefono" label.
- Fecha de Registro:** Input field with "dd/mm/aaaa" label.
- Fotografia:** Input field with "Choose file" button.

At the bottom of the form is a blue button labeled "Crear cuenta".

Figura 107 Pantalla: Crear cuenta usuario cliente

II.1.10.2.1.30. Pantalla Modificar Usuario Cliente

Jhenny Flores Rodas

Código de Usuario:

Nombre de Usuario:

Apellidos:

Edad:

Teléfono:

Activo Desde :domingo 24 de enero de 2016

Foto:

Figura 108 Pantalla: Modificar usuario cliente

II.1.10.2.1.31. Pantalla Asignar Usuario Ambiente o Cuarto

Jhenny Flores Rodas

Cambiar Sesión

Seleccionar Usuario y Ambiente:

Usuario Ambiente

Asignado Por	Al Usuario	El Cuarto o Ambiente	En Fecha	
Jhenny Flores	Abner Rodas	habitacion principal	2016-01-24	<input type="button" value="Eliminar"/>

Figura 109 Pantalla: Asignar usuario ambiente o cuarto

II.1.10.2.1.32. Pantalla Reportes Cliente

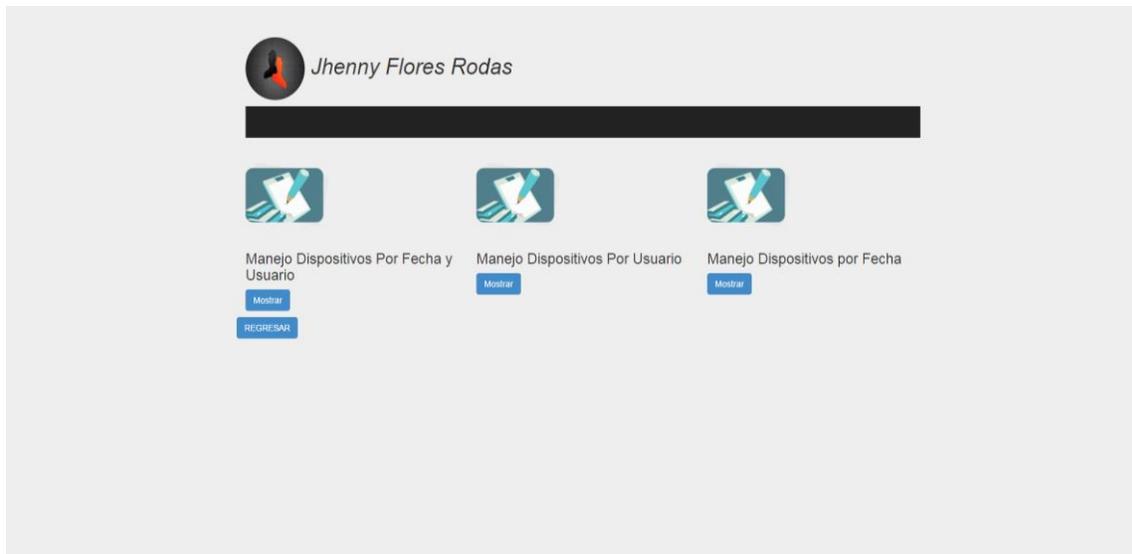


Figura 110 Pantalla: Reportes cliente

II.1.10.2.1.33. Pantalla Reporte Manejo de Dispositivos por Fecha y Usuario



Figura 111 Pantalla: Reporte manejo de dispositivos por fecha y usuario

II.1.10.2.1.34. Pantalla Reporte Manejo Dispositivo por Usuarios

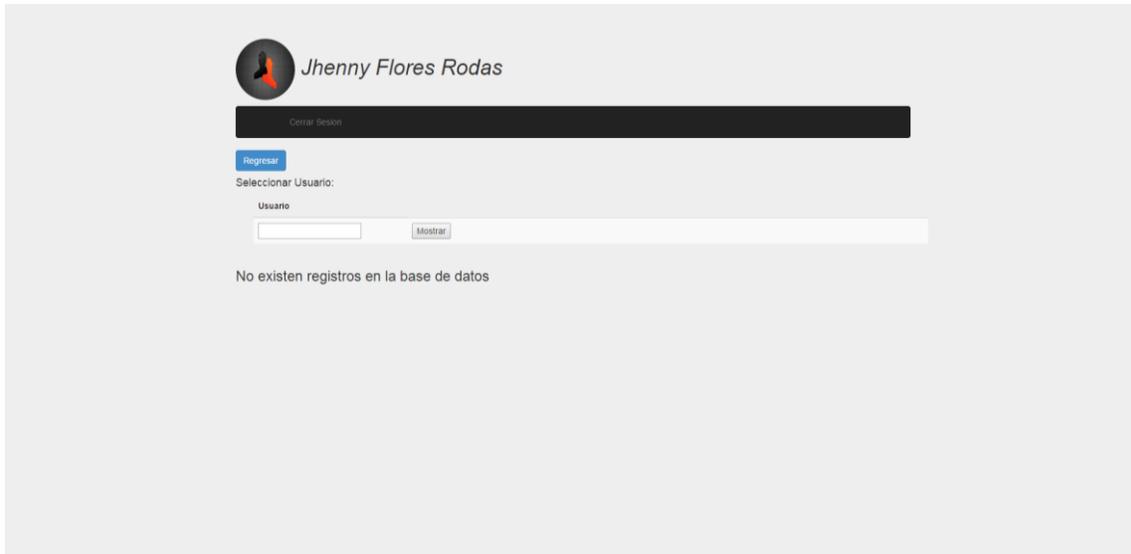


Figura 112 Pantalla: Reporte manejo dispositivos por usuario

II.1.10.2.1.35. Pantalla Manejo Dispositivos por Fecha



Figura 113 Pantalla: Manejo de dispositivos por fecha

II.1.10.2.1.36. Pantalla Cliente Usuario

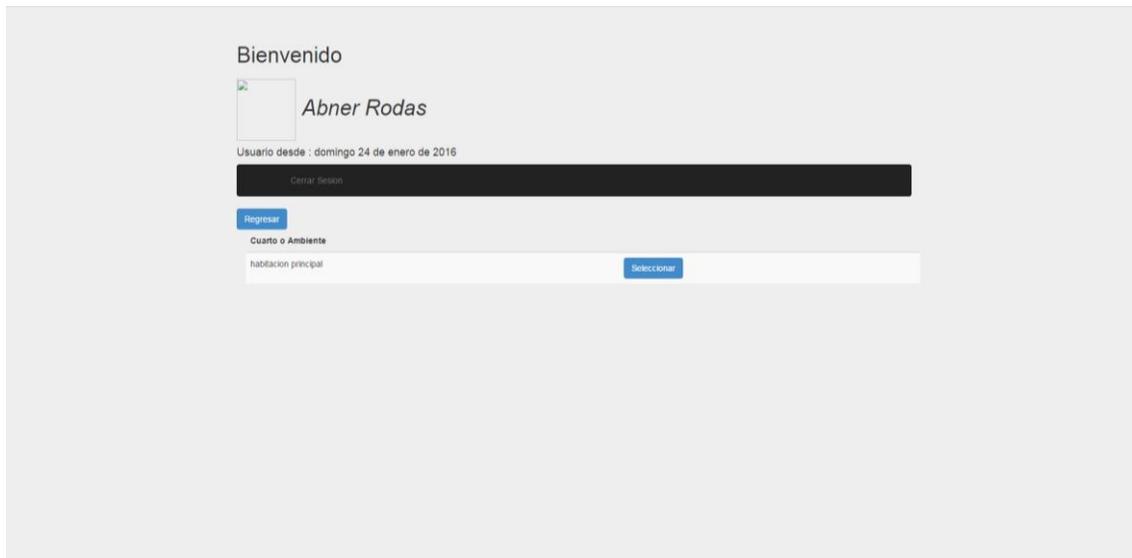


Figura 114 Pantalla: Cliente usuario

COMPONENTE 2:

**DESARROLLO DE UNA MAQUETA
O TABLERO UTILIZANDO EL
HARDWARE LIBRE ARDUINO PARA
DEMOSTRAR LA FUNCIONALIDAD
DEL SISTEMA**

II.2.COMPONENTE 2: DESARROLLO DE UNA MAQUETA O TABLERO UTILIZANDO HARDWARE LIBRE ARDUINO PARA DEMOSTRAR LA FUNCIONALIDAD DEL SISTEMA

II.2.1. Introducción

El objetivo de este componente es la fabricación, armado, instalación y configuración de los diferentes componentes electrónicos de una maqueta a escala 1:25 para dar a conocer los beneficios y demostrar como un sistema en domótica puede mejorar nuestro diario vivir.

Aplicando la arquitectura centralizada de domótica y utilizando como controlador una placa arduino 2560 rev3. Como actuadores para las puertas y el portón los motores reciclados de las bandejas de lectores de cd y dvd en desuso.

Instalamos algunos sensores como el DHT 11 que es un sensor de humedad y temperatura, un sensor de llamas, sensores ldr para el sistema de alarma externo.

En la conexión se utiliza hilos de cobre, resistencias de diferentes valores, placas de prueba (protoboard), pulsadores, puentes h L298N, puentes h L293D, buzzer o zumbador para emitir pitidos de alarma, leds de luz blanca,etc.

II.2.2. Arquitectura Centralizada

La arquitectura de los sistemas hace referencia a la manera en que están organizados sus componentes.

En un sistema de domótica de arquitectura centralizada, un controlador centralizado, envía la información a los actuadores e interfaces según el programa, la configuración y la información que recibe de los sensores, sistemas interconectados y usuarios.



Figura 115 Diagrama de conexión sensor-controlador-actuador

II.2.3. Arduino

Arduino es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

El hardware consiste en una placa con un microcontrolador Atmel AVR y puertos de entrada/salida. Los microcontroladores más usados son el Atmega168, Atmega328, Atmega1280, y Atmega8 por su sencillez y bajo coste que permiten el desarrollo de múltiples diseños. Por otro lado el software consiste en un entorno de desarrollo que implementa el lenguaje de programación Processing/Wiring y el cargador de arranque que es ejecutado en la placa. Se programa en el ordenador para que la placa controle los componentes electrónicos.

Desde octubre de 2012, Arduino se utiliza también con microcontroladoras CortexM3 de ARM de 32 bits, que coexistirán con las más limitadas, pero también económicas AVR de 8 bits. ARM y AVR no son plataformas compatibles a nivel binario, pero se pueden programar con el mismo IDE de Arduino y hacerse programas que compilen sin cambios en las dos plataformas. Eso sí, las microcontroladoras CortexM3 usan 3,3V, a diferencia de la mayoría de las placas con AVR, que generalmente usan 5V. Sin embargo, ya anteriormente se lanzaron placas Arduino con Atmel AVR a 3,3V como la Arduino Fio y existen compatibles de Arduino Nano y Pro como Meduino en que se puede conmutar el voltaje.

Arduino se puede utilizar para desarrollar objetos interactivos autónomos o puede ser conectado a software tal como Adobe Flash, Processing, Max/MSP, Pure Data. Las placas se pueden montar a mano o adquirirse. El entorno de desarrollo integrado libre se puede descargar gratuitamente.

Arduino puede tomar información del entorno a través de sus entradas analógicas y digitales, puede controlar luces, motores y otros actuadores. El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing). Los proyectos hechos con Arduino pueden ejecutarse sin necesidad de conectar a un ordenador.

También cuenta con su propio software que se puede descargar de su página oficial que ya incluye los drivers de todas las tarjetas disponibles lo que hace más fácil la carga de códigos desde el computador.

El proyecto Arduino recibió una mención honorífica en la categoría de Comunidades Digital en el Prix Ars Electrónica de 2006.

Arduino se inició en el año 2005 como un proyecto para estudiantes en el Instituto IVREA, en Ivrea (Italia). En ese tiempo, los estudiantes usaban el microcontrolador BASIC Stamp, cuyo coste era de 100 dólares estadounidenses, lo que se consideraba demasiado costoso para ellos. Por aquella época, uno de los fundadores de Arduino, Massimo Banzi, daba clases en Ivrea.

El nombre del proyecto viene del nombre del Bar di Re Arduino (Bar del Rey Arduino) donde Massimo Banzi pasaba algunas horas. En su creación, contribuyó el estudiante colombiano Hernando Barragán, quien desarrolló la tarjeta electrónica Wiring, el lenguaje de programación y la plataforma de desarrollo. Una vez concluida dicha plataforma, los investigadores trabajaron para hacerlo más ligero, más económico y disponible para la comunidad de código abierto (hardware y código abierto). El instituto finalmente cerró sus puertas, así que los investigadores, entre ellos el español David Cuartielles, promovieron la idea. Banzi afirmaría años más tarde, que el proyecto nunca surgió como una idea de negocio,

sino como una necesidad de subsistir ante el inminente cierre del Instituto de diseño Interactivo IVREA. Es decir, que al crear un producto de hardware abierto, éste no podría ser embargado.

Posteriormente, Google colaboró en el desarrollo del Kit Android ADK (Accessory Development Kit), una placa Arduino capaz de comunicarse directamente con teléfonos móviles inteligentes bajo el sistema operativo Android para que el teléfono controle luces, motores y sensores conectados de Arduino.

Para la producción en serie de la primera versión se tomó en cuenta que el coste no fuera mayor de 30 euros, que fuera ensamblado en una placa de color azul, debía ser Plug and Play y que trabajara con todas las plataformas informáticas tales como MacOSX, Windows y GNU/Linux. Las primeras 300 unidades se las dieron a los alumnos del Instituto IVREA, con el fin de que las probaran y empezaran a diseñar sus primeros prototipos.

En el año 2005, se incorporó al equipo el profesor Tom Igoe, que había trabajado en computación física, después de que se enterara del mismo a través de Internet. Él ofreció su apoyo para desarrollar el proyecto a gran escala y hacer los contactos para distribuir las tarjetas en territorio estadounidense. En la feria Maker Fair de 2011 se presentó la primera placa Arduino 32 bit para trabajar tareas más pesadas.

El Arduino Mega es probablemente el microcontrolador más capaz de la familia Arduino. Posee 54 pines digitales que funcionan como entrada/salida; 16 entradas análogas, un cristal oscilador de 16 MHz, una conexión USB, un botón de reset y una entrada para la alimentación de la placa.

Arduino Mega posee las siguientes especificaciones:

Microcontrolador: ATmega2560

Voltaje Operativo: 5V

Voltaje de Entrada: 7-12V

Voltaje de Entrada(límites): 6-20V

Pines digitales de Entrada/Salida: 54 (de los cuales 15 proveen salida PWM)

Pines análogos de entrada: 16

Corriente DC por cada Pin Entrada/Salida: 40 mA

Corriente DC entregada en el Pin 3.3V: 50 mA

Memoria Flash: 256 KB (8KB usados por el bootloader)

SRAM: 8KB

EEPROM: 4KB

Clock Speed: 16 MHz

Alimentación

Arduino Mega puede ser alimentado mediante el puerto USB o con una fuente externa de poder. La alimentación es seleccionada de manera automática.

Cuando se trabaja con una fuente externa de poder se debe utilizar un convertidor AC/DC y regular dicho voltaje en el rango operativo de la placa. De igual manera se puede alimentar el micro mediante el uso de baterías. Preferiblemente el voltaje debe estar en el rango de los 7V hasta los 12V.

Arduino Mega posee algunos pines para la alimentación del circuito aparte del adaptador para la alimentación:

VIN: A través de este pin es posible proporcionar alimentación a la placa.

5V: Podemos obtener un voltaje de 5V y una corriente de 40mA desde este pin.

3.3V: Podemos obtener un voltaje de 3.3V y una corriente de 50mA desde este pin.

GND: El ground (0V) de la placa.

Arduino puede ser programado de una manera muy fácil utilizando el lenguaje propio de Arduino junto con la interfaz Arduino IDE.

II.2.4. Sensor DHT11

El DHT11 es un sensor que proporciona una salida de datos digital. Entre sus ventajas podemos mencionar el bajo coste y el despliegue de datos digitales. Esto supone una gran ventaja frente a los sensores del tipo análogo, como el LM335 por ejemplo, en los cuales las fluctuaciones en el voltaje alteran la lectura de datos.

Entre las desventajas pues, el DHT11 solo lee enteros, no podemos leer temperaturas con decimales por lo que tenemos que pensarlo muy bien a la hora de utilizar este sensor para trabajos en los que se requieran lecturas precisas de temperatura y/o humedad.

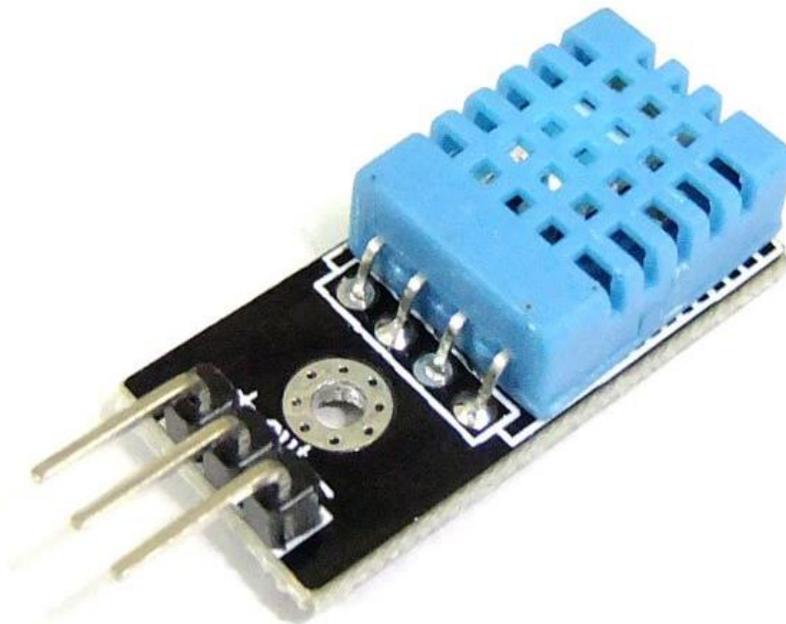


Figura 118 Sensor de temperatura y humedad DHT11

II.2.5. Sensor de Llamas

No resulta extraño que los sensores de llamas se encuentren entre los más vendidos. La industria los suministra de todos los tipos, tamaños y precios, porque la idea de que se te quemara la casa o la empresa es algo que ayuda mucho a sacar la cartera y buscar un detector que sea capaz de avisarte con tiempo de que hay llamas cerca de tus cosas.

En realidad, es muy probable que tengas varios de estos detectores en tu casa. Imagínate, por ejemplo, los calentadores de gas, los hornos domésticos de gas, o un sencillo calentador de agua y calefacción.

Todos ellos requieren una llama quemando gas. Si por cualquier motivo, la llama se extingue y no lo detectases, el gas seguiría fluyendo y causando un embalsamiento, listo para causar un disgusto explosivo.

Por eso, todos estos electrodomésticos, y sus equivalentes industriales, incorporan por ley, un detector de llama, que pueda cortar el gas en cuanto detecte falta de llama.

Los sensores más sencillos (y baratos), suelen ser del tipo de detectores de ultravioletas, o sensores de infrarrojos.

El mercado ofrece, también, un amplio espectro de sensores del tipo de detectores de humo y de gases nocivos como monóxido de Carbono (CO) típico de la combustión, Dioxido de Carbono (CO₂), detectores de humo, o de gases peligrosos libres, como hidrocarburos (Propano, Butano y demás).

En un entorno normal, la ignición de una llama provoca una ionización potente de gases comunes (En realidad un plasma ionizado, similar al que ilumina una lámpara halógena), que generan un patrón típico de frecuencia en el rango del ultravioleta. El detector es un semiconductor sensible a este rango de frecuencias.

Cuando la llama se estabiliza, es relativamente sencillo detectar su espectro infrarrojo, con una fotorresistencia LDR, pero diseñada para ser específicamente sensible a un rango característico.

Usaremos un sensor muy sencillo y fácil de manejar que es un sensor infrarrojo de llamas. Funcionan detectando una longitud de onda específica (de unos 760 nm) que son características de las llamas, aunque son relativamente fáciles de engañar y pueden dar falsos positivos con ciertas luces.

Este sensor se encuentra muy frecuentemente encapsulado con un soporte y un potenciómetro para ajustar la sensibilidad, por poco dinero. Incluso se vende en formato múltiple, con varias cabezas apuntando en distintas direcciones.

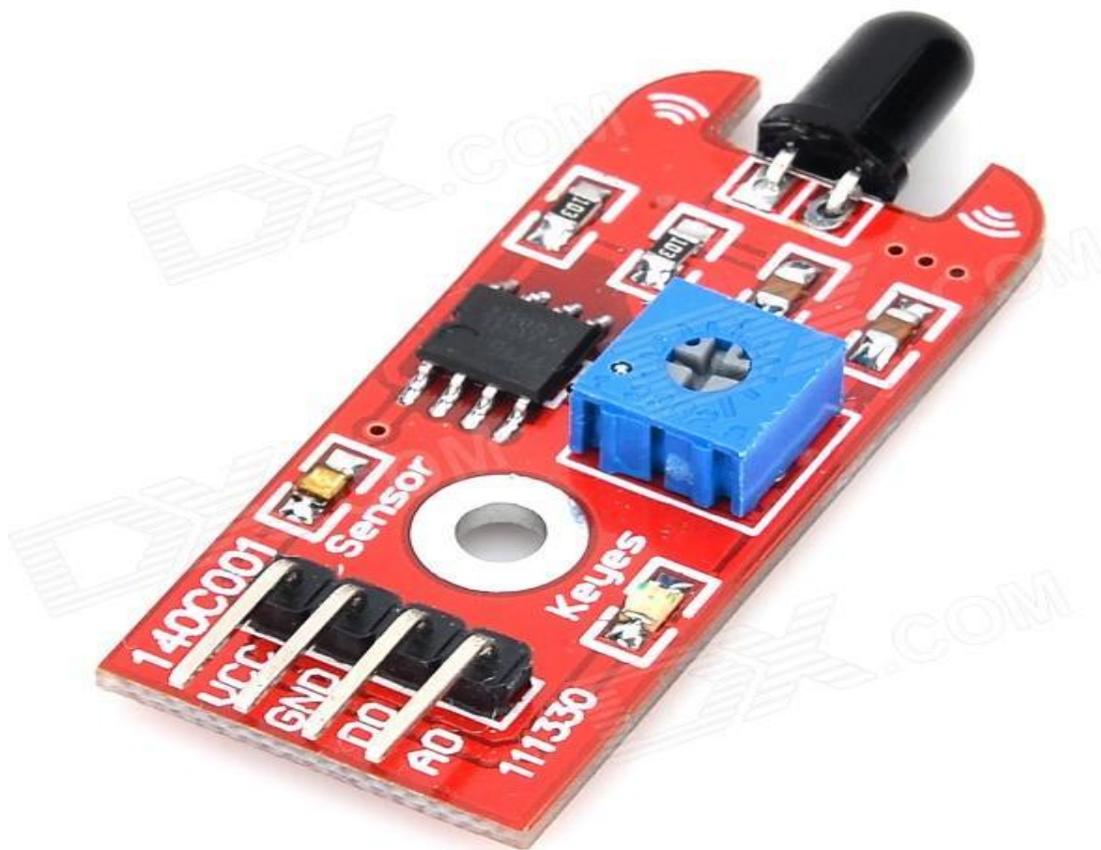


Figura 119 Sensor de flama

II.2.6. Sensor LDR

Un sensor fotoeléctrico o fotocélula es un dispositivo electrónico que responde al cambio en la intensidad de la luz. Estos sensores requieren de un componente emisor que genera la luz, y un componente receptor que percibe la luz generada por el emisor.

Todos los diferentes modos de sensado se basan en este principio de funcionamiento. Están diseñados especialmente para la detección, clasificación y posicionado de objetos; la detección de formas, colores y diferencias de superficie, incluso bajo condiciones ambientales extremas.

Los sensores de luz se usan para detectar el nivel de luz y producir una señal de salida representativa respecto a la cantidad de luz detectada. Un sensor de luz incluye un transductor fotoeléctrico para convertir la luz a una señal eléctrica y puede incluir electrónica para condicionamiento de la señal, compensación y formateo de la señal de salida.

El sensor de luz más común es el LDR -Light Dependant Resistor o Resistor dependiente de la luz-.Un LDR es básicamente un resistor que cambia su resistencia cuando cambia la intensidad de la luz.



Figura 120 Resistencia variable LDR

II.2.7. Protoboard

El protoboard o breadbord: Es una especie de tablero con orificios, en la cual se pueden insertar componentes electrónicos y cables para armar circuitos. Como su nombre lo indica, esta tableta sirve para experimentar con circuitos electrónicos, con lo que se asegura el buen funcionamiento del mismo.

Estructura del protoboard: Básicamente un protoboard se divide en tres regiones:

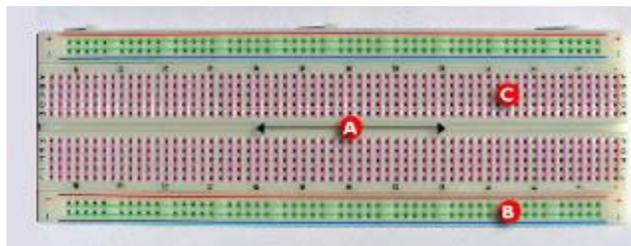


Figura 121 Protoboard

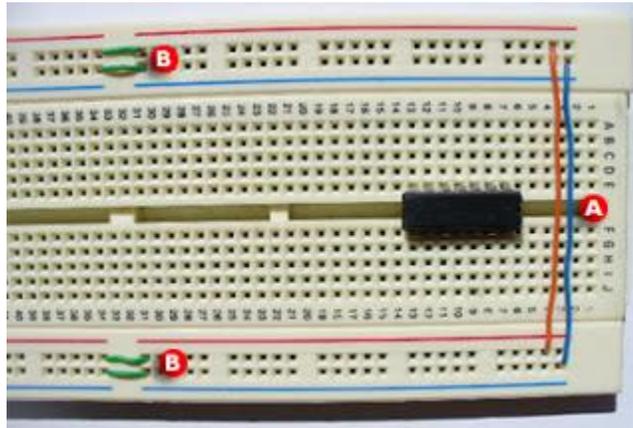
A) Canal central: Es la región localizada en el medio del protoboard, se utiliza para colocar los circuitos integrados.

B) Buses: Los buses se localizan en ambos extremos del protoboard, se representan por las líneas rojas (buses positivos o de voltaje) y azules (buses negativos o de tierra) y conducen de acuerdo a estas, no existe conexión física entre ellas. La fuente de poder generalmente se conecta aquí.

C) Pistas: Las pistas se localizan en la parte central del protoboard, se representan y conducen según las líneas rosas.

Recomendaciones al utilizar el protoboard: A continuación veremos una serie de consejos útiles pero no esenciales.

1.- Hacer las siguientes conexiones:

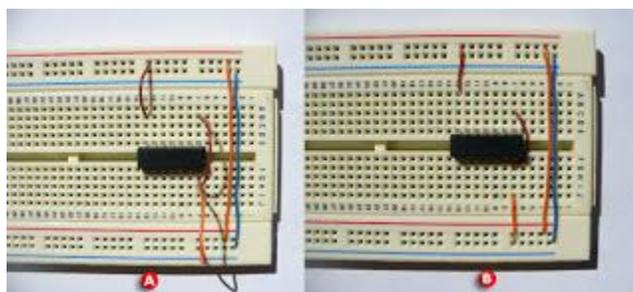


A) Esta conexión nos sirve para que ambos pares de buses conduzcan corriente al agregarles una fuente de poder, así es más fácil manipular los circuitos integrados.

B) Algunos protoboards tienen separada la parte media de los buses, es por eso que se realiza esta conexión para darle continuidad a la corriente.

2.- Coloca los circuitos integrados en una sola dirección, de derecha a izquierda o viceversa.

3.- Evita el cableado aéreo (A), resulta confuso en circuitos complejos. Un cableado ordenado (B) mejora la comprensión y portabilidad.



II.2.8. Pulsadores

Un botón o pulsador es un dispositivo utilizado para realizar cierta función. Los botones son de diversas formas y tamaño y se encuentran en todo tipo de dispositivos, aunque principalmente en aparatos eléctricos y electrónicos.

Los botones son por lo general activados, al ser pulsados con un dedo. Permiten el flujo de corriente mientras son accionados. Cuando ya no se presiona sobre él vuelve a su posición de reposo.

Puede ser un contacto normalmente abierto en reposo NA o NO (Normally Open en Inglés), o con un contacto normalmente cerrado en reposo NC.

Consta del botón pulsador; una lámina conductora que establece contacto con los dos terminales al oprimir el botón, y un muelle que hace recobrar a la lámina su posición primitiva al cesar la presión sobre el botón pulsador.

El botón de un dispositivo electrónico funciona por lo general como un interruptor eléctrico, es decir en su interior tiene dos contactos, al ser pulsado uno, se activará la función inversa de la que en ese momento este realizando, si es un dispositivo NA (normalmente abierto) será cerrado, si es un dispositivo NC (normalmente cerrado) será abierto.



Figura 122 Pulsadores

II.2.9. Puente H L298N

El módulo controlador L298N, usa un chip L298N que puede controlar directamente 2 motores de 3-30 VDC, tiene una salida de interfaz de 5V, soporta control desde sistemas de microcontroladores de 3.3V y 5V, con este módulo puedes controlar fácilmente la velocidad y dirección de tus motores de DC, también puedes controlar motores paso a paso de 2 fases.

Este módulo puede ser fácilmente integrado a plataformas móviles, mecanismos lineales, etc,

Especificaciones:

Chip controlador de motor: L298N puente H doble

Voltaje de alimentación del chip: +5.5V a +24V

Pico de corriente soportado: 2A

Voltaje de alimentación de las terminales de control: 3 - 5.5V

Voltage de control: Niveles Alto = 5.5V, Bajo = 0V

Potencia máxima de consumo: 20W

Tamaño PCB: 53.5mm * 45mm * 30mm

Distancia entre agujeros de sujeción: 47.5mm y 39mm

Peso: 26 gramos

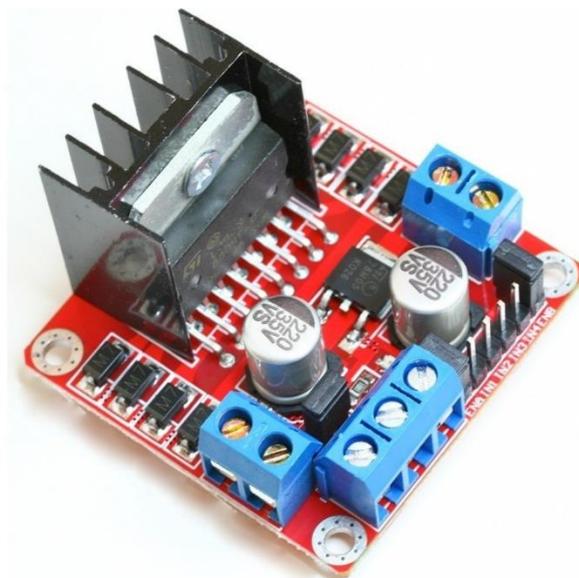


Figura 123 Puente H L298N

II.2.10. Puente H L293D

El L293 y L293D incluye en su interior 4 drivers o medio puente H. La corriente máxima que el L293 puede manejar es de 1A a voltajes desde 4.5 volts a 36 volts, mientras que la corriente constante es de 600 mA. Todas las entradas del circuito son compatibles con TTL, por lo que pueden manejarse con la mayoría de los microcontroladores y circuitos lógicos del mercado. La salida de los drivers puede activarse y desactivarse en pares, mediante señales de habilitación. Cuando se presenta un 1 lógico en los pines de habilitación, las señales correspondientes de salida estarán activas y en fase con la señal de entrada. Cuando las entradas de habilitación presentan un estado lógico 0, las salidas se encuentran en estado de alta impedancia.



Figura 124 Puente H L293D

II.2.11. Buzzer o Zumbador

Zumbador, buzzer en inglés, es un transductor electroacústico que produce un sonido o zumbido continuo o intermitente de un mismo tono (generalmente agudo). Sirve como mecanismo de señalización o aviso y se utiliza en múltiples sistemas, como en automóviles o en electrodomésticos, incluidos los despertadores.

Inicialmente este dispositivo estaba basado en un sistema electromecánico que era similar a una campana eléctrica pero sin el badajo metálico, el cual imitaba el sonido de una campana.



Figura 125 Buzzer o zumbador

II.2.12. Led

Los ledes se usan como indicadores en muchos dispositivos y en iluminación. Los primeros ledes emitían luz roja de baja intensidad, pero los dispositivos actuales emiten luz de alto brillo en el espectro infrarrojo, visible y ultravioleta.

Debido a su capacidad de operación a altas frecuencias, son también útiles en tecnologías avanzadas de comunicaciones y control. Los ledes infrarrojos también se usan en unidades de control remoto de muchos productos comerciales incluyendo equipos de audio y video.

Existen tres formas principales de conocer la polaridad de un led:

La pata más larga siempre va a ser el ánodo.³

En el lado del cátodo, la base del led tiene un borde plano.

Dentro del led, la plaqueta indica el ánodo. Se puede reconocer porque es más pequeña que el yunque, que indica el cátodo.

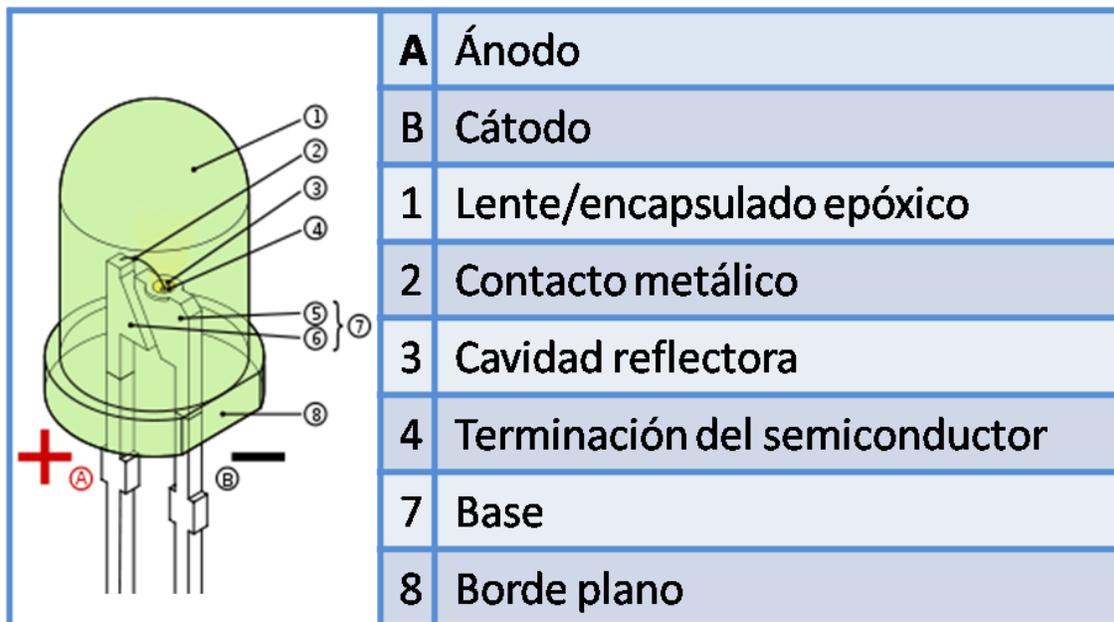


Figura 126 Diodo emisor de luz LED

II.2.13. Relay

El relé (en francés: relais, “relevo”) o relevador es un dispositivo electromagnético. Funciona como un interruptor controlado por un circuito eléctrico en el que, por medio de una bobina y un electroimán, se acciona un juego de uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes. Fue inventado por Joseph Henry en 1835.

Dado que el relé es capaz de controlar un circuito de salida de mayor potencia que el de entrada, puede considerarse, en un amplio sentido, como un amplificador eléctrico. Como tal se emplearon en telegrafía, haciendo la función de repetidores que generaban una nueva señal con corriente procedente de pilas locales a partir de la señal débil recibida por la línea. Se les llamaba "relevadores".

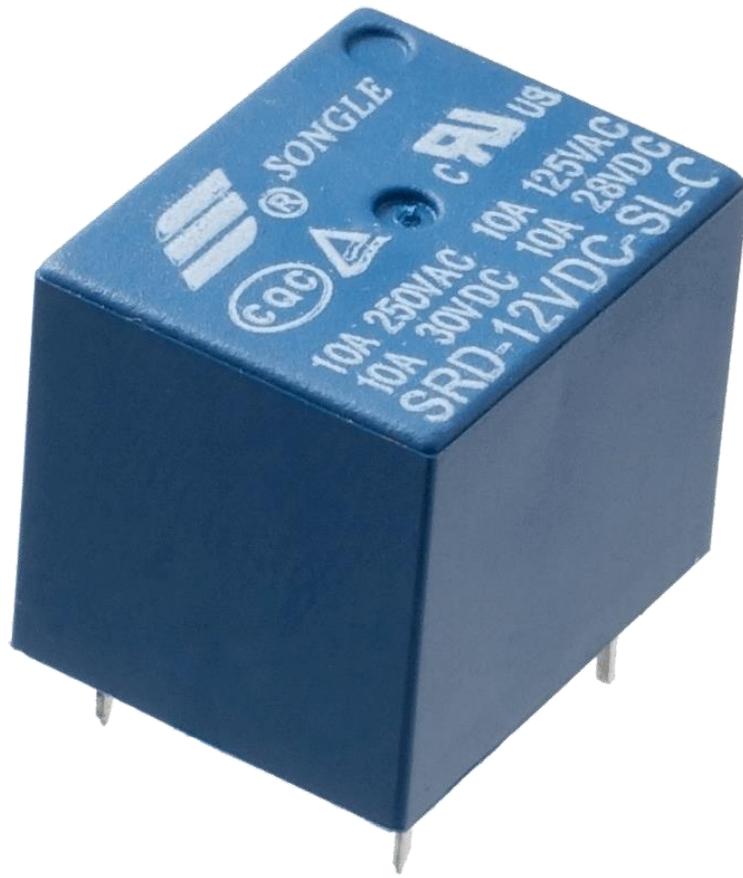


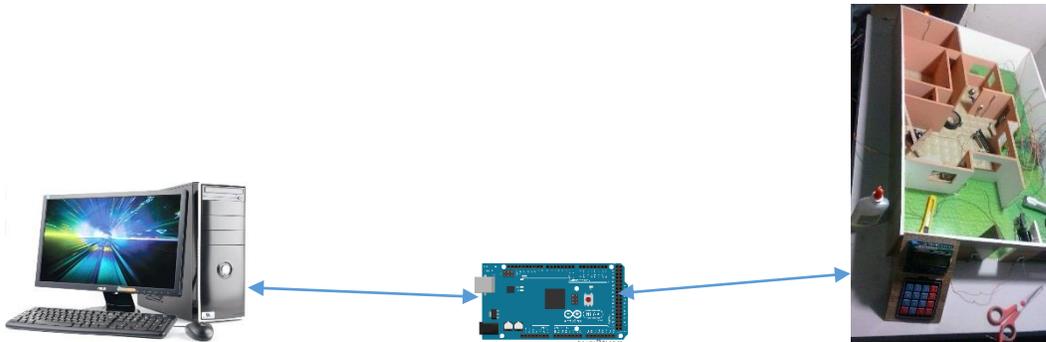
Figura 127 Relay

II.2.14. Diagramas de Conexión

Un diagrama electrónico, también conocido como un esquema eléctrico o esquemático es una representación pictórica de un circuito eléctrico. Muestra los diferentes componentes del circuito de manera simple y con pictogramas uniformes de acuerdo a normas, y las conexiones de alimentación y de señal entre los distintos dispositivos. El arreglo de los componentes e interconexiones en el esquema generalmente no corresponde a sus ubicaciones físicas en el dispositivo terminado.

A diferencia de un esquema de diagrama de bloques o disposición, un esquema de circuito muestra la conexión real mediante cables entre los dispositivos. (Aunque el esquema no tiene que corresponder necesariamente a lo que el circuito real aparenta) -- El tipo de dibujo que sí representa al circuito real se llama negativo (o positivo) de la tablilla de circuito impreso.

II.2.14.1. Diagrama de Conexión Pc-Maqueta-Arduino



II.2.14.2. Diagrama de Conexión DTH11

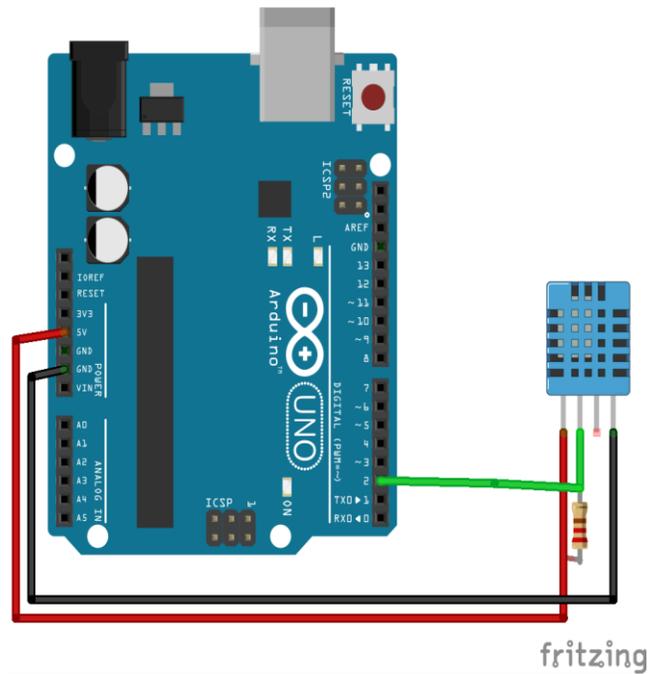


Figura 128 Diagrama de conexión sensor DTH11

II.2.14.3. Diagrama de Conexión Sensor de Llamas

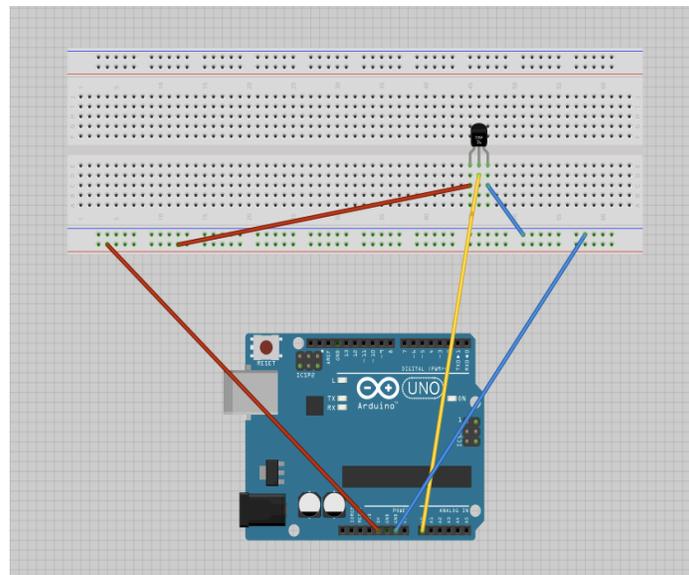


Figura 129 Diagrama de conexión sensor de Llamas

II.2.14.4. Diagrama de Conexión Sensor Ldr

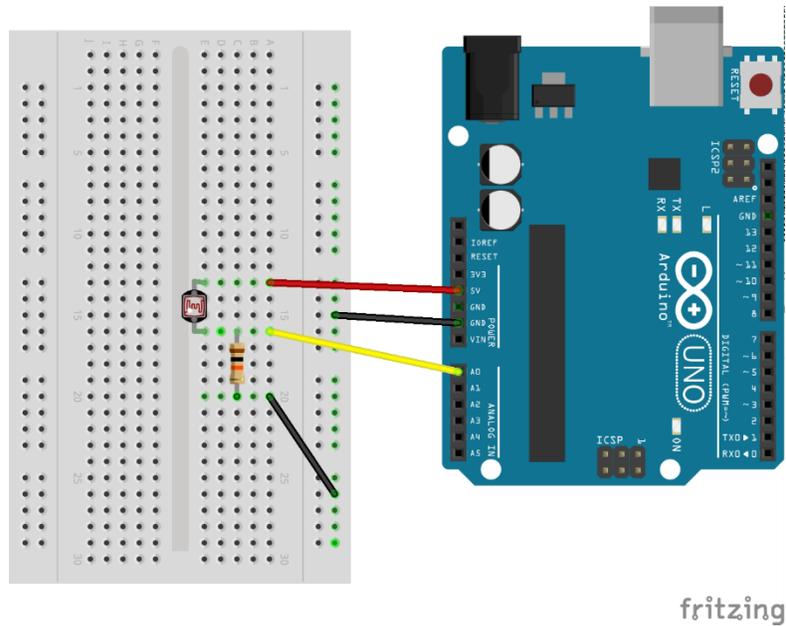


Figura 130 Diagrama de conexión sensor Ldr

II.2.14.5. Diagrama de Conexión Pulsador

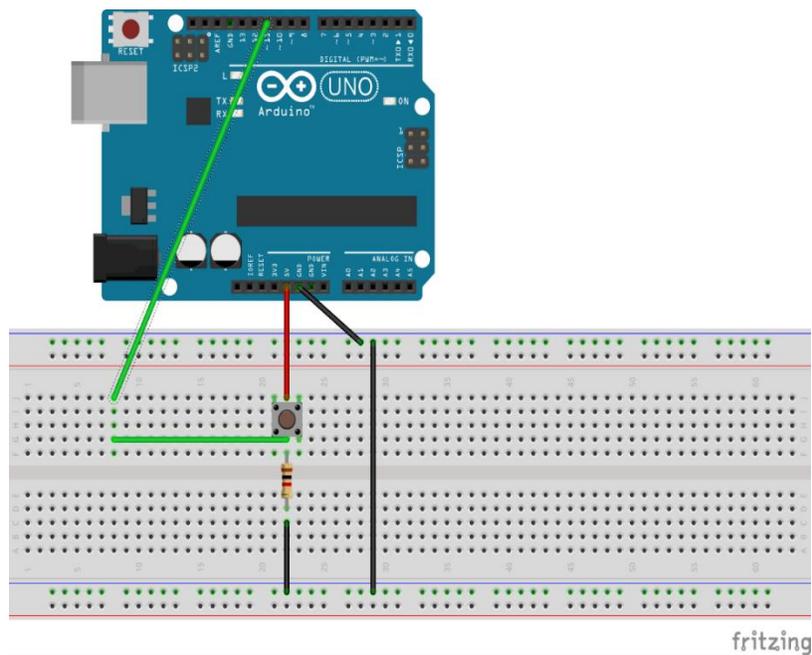


Figura 131 Diagrama conexión pulsador

II.2.14.6. Diagrama de Conexión Puente H L298N

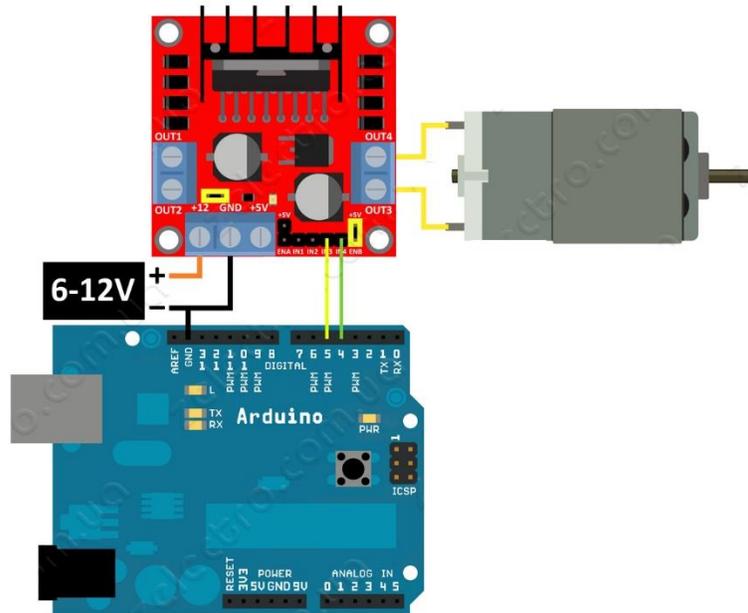
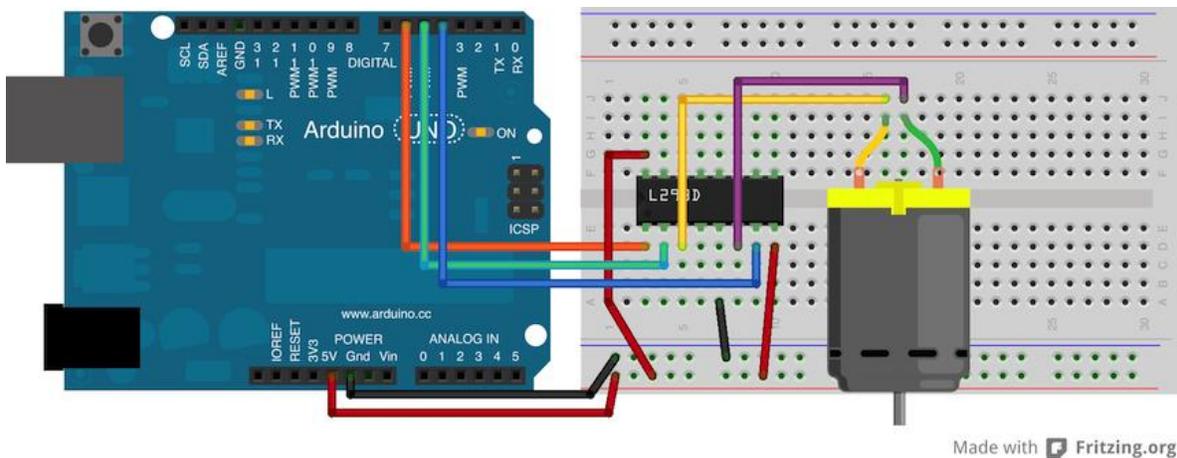


Figura 132 Diagrama de conexión puente h L298N

II.2.14.7. Diagrama de Conexión Puente H L293D



Made with  Fritzing.org

Figura 133 Diagrama de conexión puente H L293D

II.2.14.8. Diagrama de Conexión Led

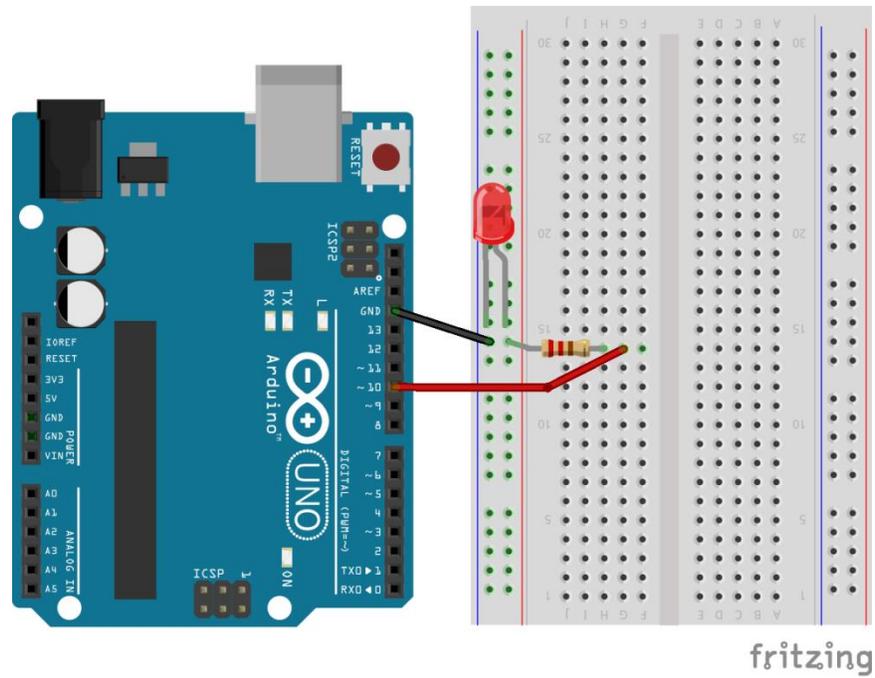


Figura 134 Diagrama conexión LED

II.2.14.9. Diagrama de Conexión Relay

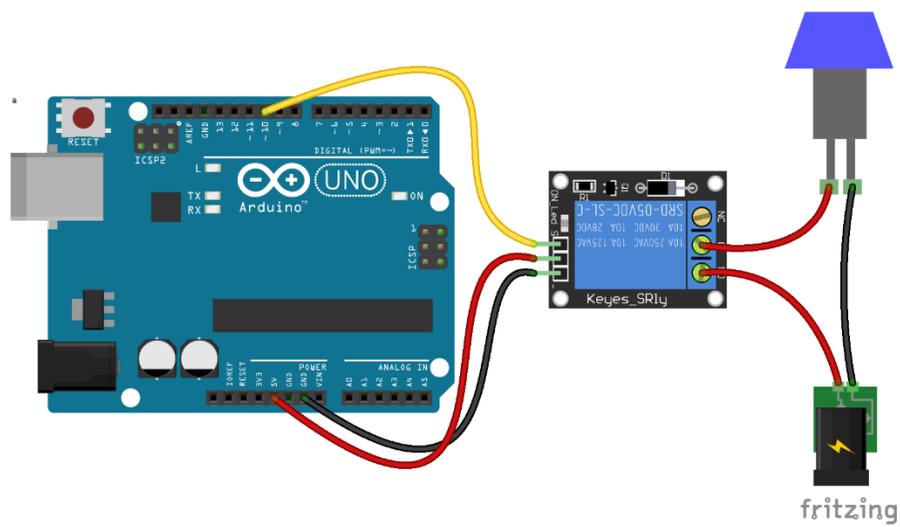


Figura 135 Diagrama de conexión relay

II.2.15. Guía Para el Reciclado de Caja Reductora

Lo primero es quitar los tornillos que permiten remover la tapa metálica. En este modelo hay cuatro, puede que no todas sean iguales.



Una vez abierta, encontraremos que hay una o dos placas llenas de componentes que seguramente también pueden ser aprovechados. Fíjate que tienen trabas o tornillos que permiten removerlas.



Una vez que estas placas han salido, ya tenemos el motor y la caja reductora a la vista (a la izquierda, abajo).



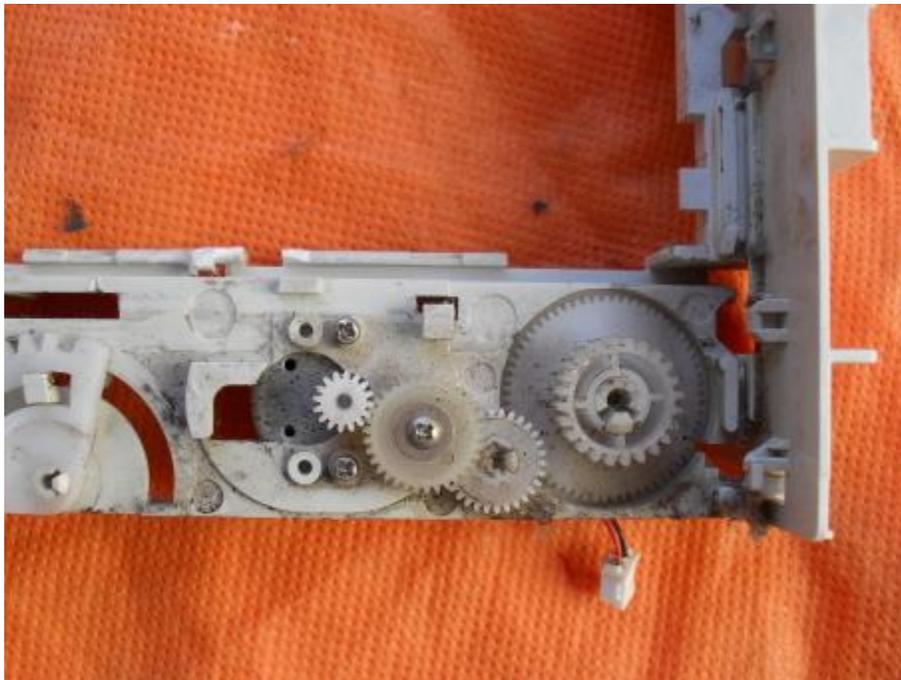
Nuestro objetivo es quedarnos solo con el bastidor plástico que tiene atornillado el motor y los engranajes.



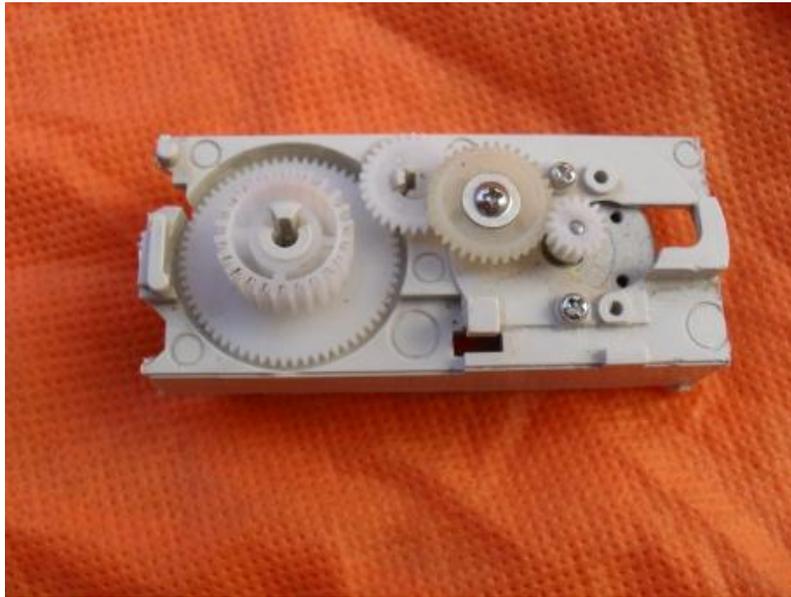
Esto es lo que nos interesa:



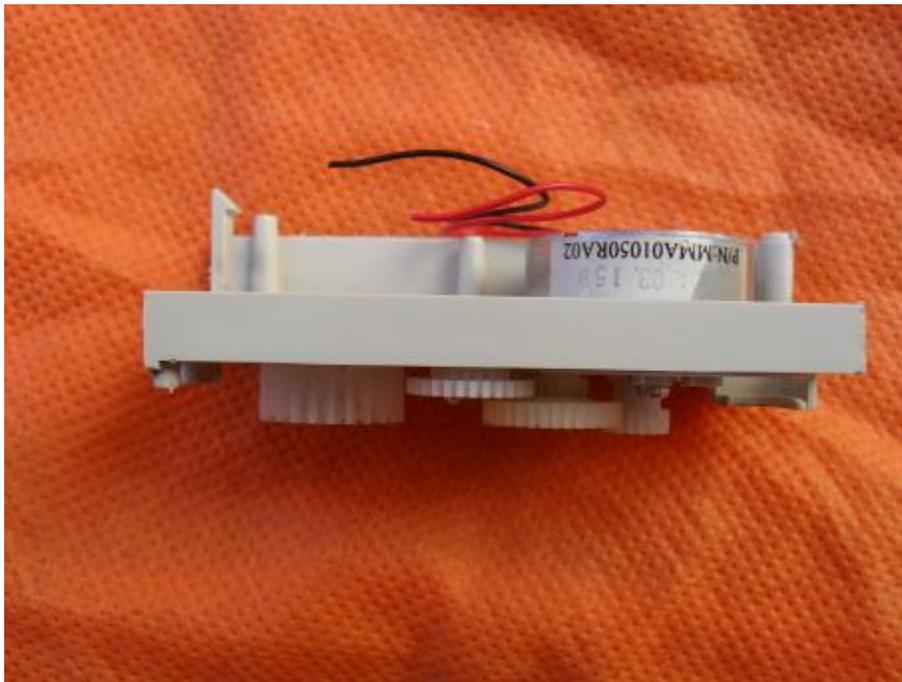
Todo puede ser descartado, solo necesitamos quedarnos con esta parte:

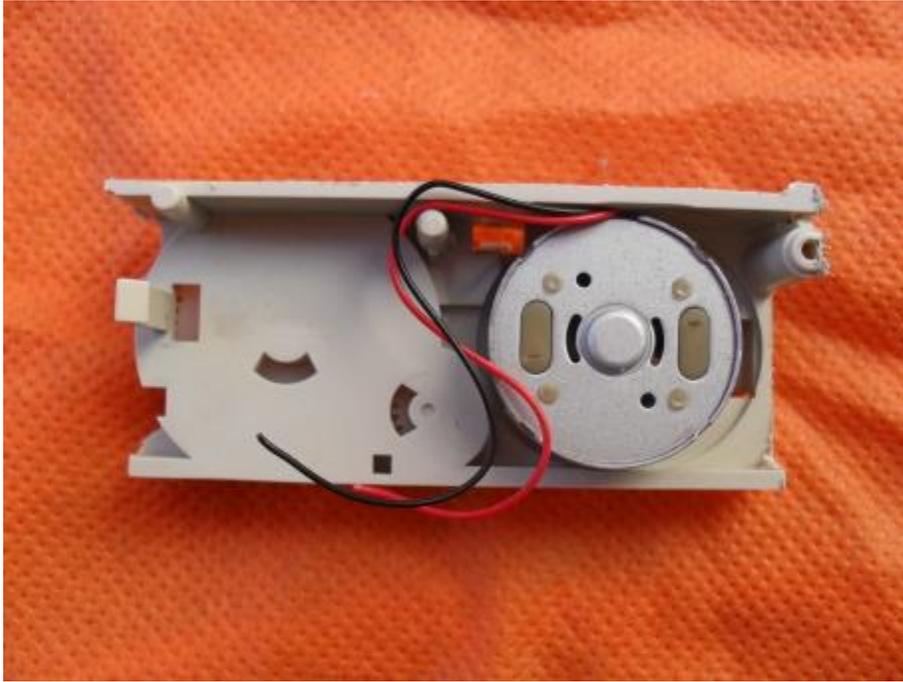


Aquí es donde viene la parte más trabajosa, pero que no lleva más de 10 o 15 minutos: con una sierra de cortar metales, o lo que tengas a mano que te permita cortar plástico, corta el bastidor hasta quedarte con algo así:



Esa es nuestra caja reductora, con motor y todo. Algunas fotos más:





Mecanismo para abrir y cerrar puertas y el portón (reciclado)

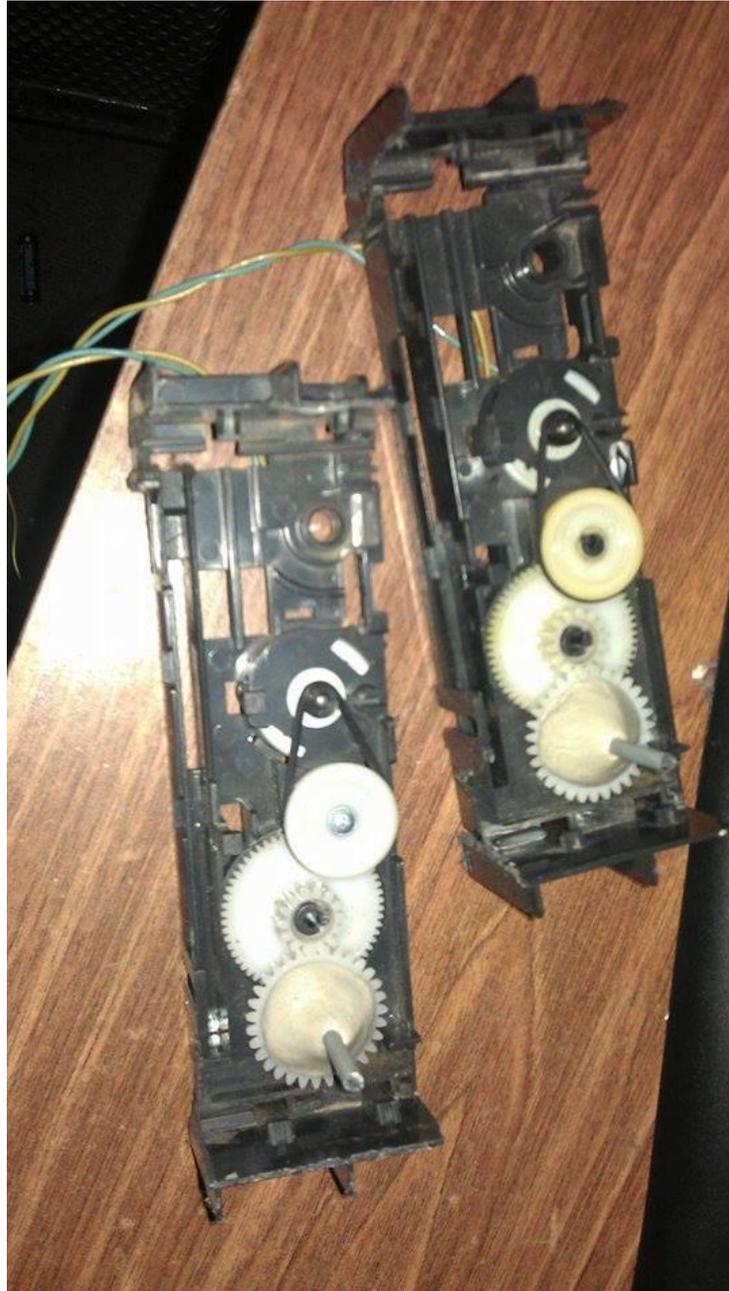
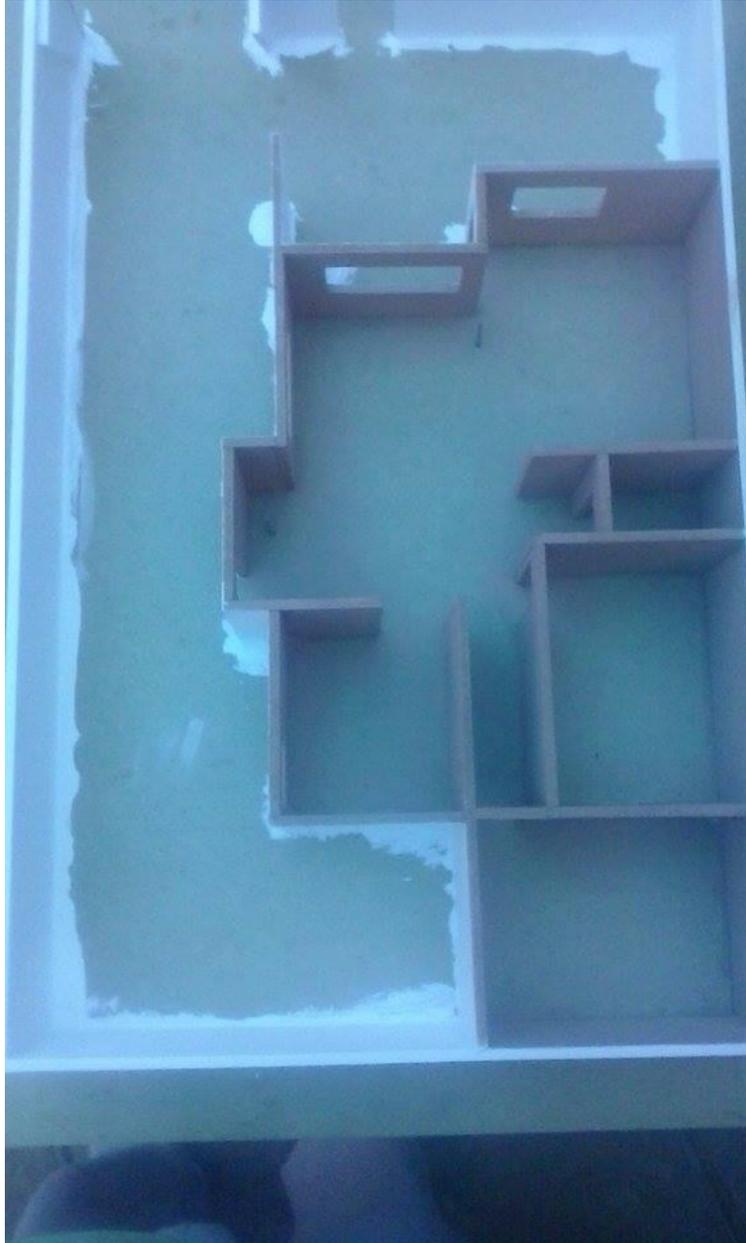
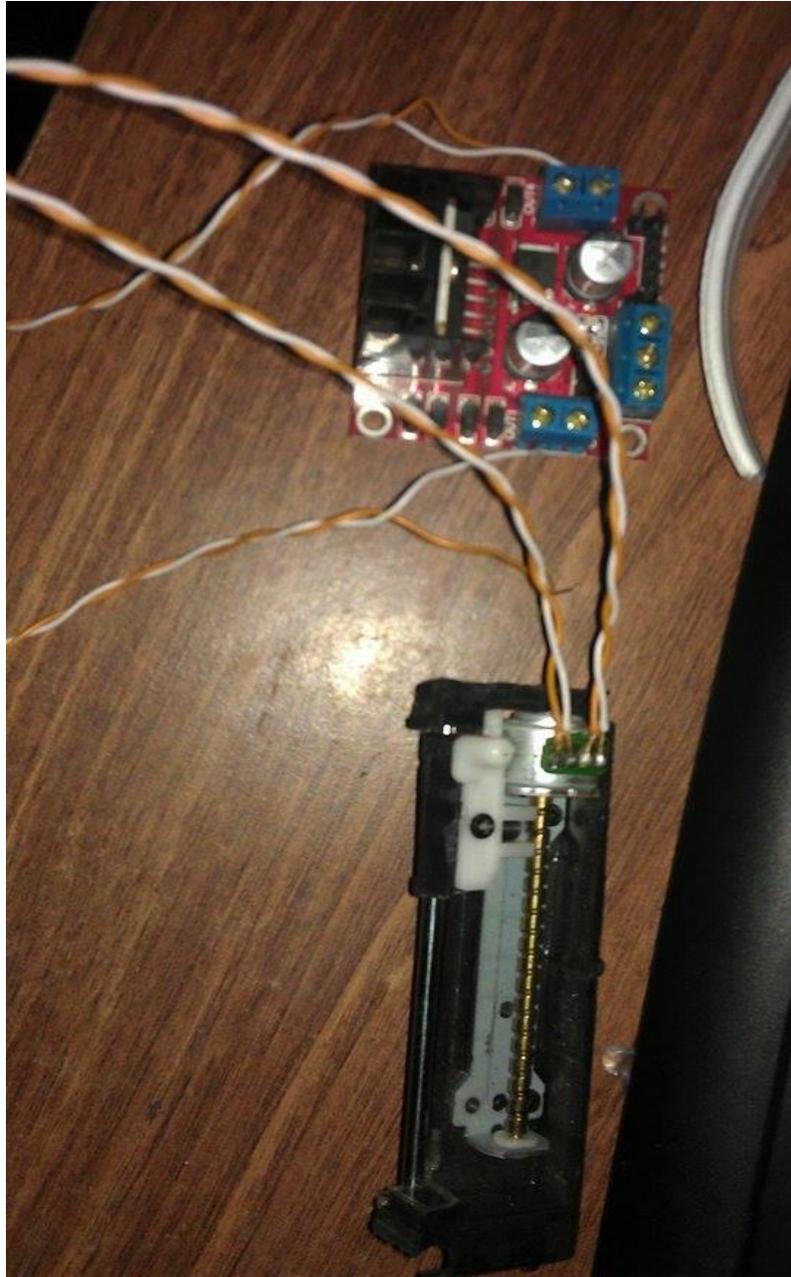


Figura 136 Mecanismo de acción para abrir y cerrar

Pintado de muros y paredes escala 1:25**Figura 137 Pintado de muros y paredes escala 1:25**

Puente h que controla motor de pasos para abrir y cerrar









Vista interior de la maqueta



Tablero de control



Diseño Casi Terminado



Fabricación de Soportes para los LEDS



II.2.16. Lista de Materiales

1	Arduino Mega 2560 REV 3
2	Puente h L298N
3	Puente h L293D
4	Sensor de Humedad y temperatura DTH11
5	Sensor de Llamas
6	Pulsadores
7	Relay
8	Resistencias de Cerámica
9	Led's
10	Cajas Reductoras
11	Ldr
12	Diodos Laser
13	Protoboard
14	Hilos de Cobre
15	Cooler

Tabla 74 Lista de materiales utilizados en la maqueta

II.2.17. Resultados Esperados

Manejo total y/o parcial de todos los componentes instalados en la maqueta de manera correcta desde el sistema web DOMUINO.

II.2.18. Medios de Verificación

Carta del ing. Marcelo Céspedes docente tutor del proyecto.

Presentación de la maqueta el día de la defensa final del trabajo, evidencia física del cumplimiento del componente número dos.

II.2.19. Conclusiones

Terminado el proyecto en su segundo componente los resultados esperados fueron de carácter positivo, se verifica la funcionalidad, la maqueta se integra correctamente con el software DOMUINO.

CAPITULO III

CONCLUSIONES Y RECOMENDACIONES

III. Conclusiones y Recomendaciones

III.1. Conclusiones

En la investigación realizada antes del proyecto, se llegó a la conclusión de la necesidad de contar un sistema que pueda controlar y monitorear los inmuebles desde cualquier dispositivo con accesos a internet.

Para el componente dos del proyecto que es la construcción de una maqueta para evidenciar el correcto funcionamiento del sistema concluyo que; al momento de realizar la maqueta no se contaba en nuestro país con un fácil acceso a los componentes electrónicos tales como placas arduino, sensores, servomotores, pantallas lcd, puentes h, etc. Que son parte fundamental para la culminación del componente dos.

La falta de documentación en cuanto a manejo e implementación de arduino como potencial controlador para su uso en domótica, ya que es un producto en el mercado recientemente consolidado, hace de este trabajo un trabajo de exhaustivas y largas horas de investigación de cómo interactuar con arduino desde el lenguaje java.

Tantas horas de investigación y trabajo se puede evidenciar en el manejo de los dispositivos desde el sistema en tiempo real.

Uno de los inconvenientes y factores que retardaron la conclusión de este proyecto fue que se debió esperar entre 20 a 60 días hábiles para que el proveedor entregue los componentes electrónicos.

Con el desarrollo y culminación de este proyecto, podemos concluir:

- Se ha desarrollado, el Sistema GESTIONAR EL CONTROL Y MONITOREO DE UNA VIVIENDA, USANDO LA TECNOLOGÍA WEB, Y EL HARDWARE DE CÓDIGO ABIERTO ARDUINO, que permite a los usuarios tener el control total y autónomo de su inmueble.
- La utilización de software y hardware de código abierto y de licencia GPL, viene a constituirse, en una opción muy interesante, respecto del software y hardware comercial, que tiene unos costos de licencia elevados.

III.2. Recomendaciones

Las Recomendaciones del proyecto son:

- Para ampliar las capacidades del Sistema GESTIONAR EL CONTROL Y MONITOREO DE UNA VIVIENDA, USANDO LA TECNOLOGÍA WEB, Y EL HARDWARE DE CÓDIGO ABIERTO ARDUINO, se recomienda implementar toda la gama de sensores y actuadores de código abierto para arduino, que representa un inmenso mundo de posibilidades para el control y monitoreo de inmuebles.
- Establecer un presupuesto económico para el equipamiento tecnológico en los centros de salud para poder hacer uso del sistema web y así poder aprovechar los beneficios que el mismo ofrece.

- Para subsanar la falta de dispositivos electrónicos en la construcción del proyecto en su componente dos se recomienda el reciclado de la denominada “Basura Electrónica”, que no es más que el uso de los componentes electrónicos que se pueden reciclar de todos y cada uno de los aparatos electrónicos en desuso, por ejemplo para el mecanismo que abre y cierra las puertas y el portón se reciclo el sistema de la bandeja de expulsión de viejos o dañados lectores de cd o dvd.
Para el sistema exterior de alarma se reciclo el diodo laser que se encuentra en los denominados punteros laser, para la conexión de los diferentes componentes en la protoboard se reciclo alambre de cobre que se encuentra en los cables de red (UTP) en desuso. Así también los diferentes pulsadores que se presentan en la maqueta son reciclados de viejos aparatos electrónicos.
- Para conseguir las placas y sensores en este proyecto se tuvo que hacer un pedido a la república popular de china, desde sus páginas en internet, como: www.aliexpress.com, www.dealextreme.com, www.efrochina.com, etc. Y con pago de tarjeta de crédito.