

**UNIVERSIDAD CATÓLICA SANTO TORIBIO DE MOGROVEJO**  
**FACULTAD DE INGENIERÍA**  
**ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN**



**DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL  
ANDROID ORIENTADA AL ADULTO MAYOR PARA APOYAR  
LA ADHERENCIA AL TRATAMIENTO MÉDICO**

**TESIS PARA OPTAR EL TÍTULO DE  
INGENIERO DE SISTEMAS Y COMPUTACIÓN**

**AUTOR**

**BRANDON JESÚS PÉREZ GUEVARA**

**ASESOR**

**Mgtr. GREGORIO MANUEL LEÓN TENORIO**

**Chiclayo, 2019**

## **DEDICATORIA**

A Dios

A mis padres y hermanos

A mi asesor de tesis, Mgtr. Ing. Gregorio León

## **AGRADECIMIENTOS**

A mis padres y hermanos por apoyarme siempre en todas las decisiones que he tomado a lo largo de la carrera y por impulsarme a ser mejor, educarme y poner su esfuerzo en mí.  
Gracias por creer en mí.

A mi asesor de tesis por acompañarme en esta investigación guiándome y compartiendo sus conocimientos haciéndome ampliar mis horizontes. Gracias a usted, por su tiempo, dedicación y paciencia.

A mis docentes amigos por haberme educado y verme crecer en la carrera.

## RESUMEN

En la población lambayecana se identificó un problema respecto a la adherencia al tratamiento médico en adultos mayores cuando se ha prescrito un régimen de tomas de fármacos de larga o corta duración. Se consideró que esto puede superarse a través del diseño y la implementación de una aplicación móvil que apoye a la adherencia al tratamiento médico y que pueda ayudar a su cumplimiento. Para ello se tuvo que garantizar el cumplimiento de dicho tratamiento trabajando con los asistentes médicos de los pacientes. Esta aplicación ayudó al asistente a llevar un control de la prescripción de su paciente y evolución a través del tiempo, lo que mejoró el cumplimiento de los tratamientos médicos. Además, ayudó también con el control de asistencia a citas médicas. Esta investigación fue de tipo experimental / cuasi - experimental. Se utilizó un muestreo aleatorio simple, se hizo uso de los instrumentos de recolección de datos como encuestas, cuestionarios y entrevistas; algunos ejemplos son: el Test de Morisky & Green, que es una entrevista; Test de cumplimiento SMAQ, que es un cuestionario; entre otros instrumentos validados, documentos y completamente vigentes al año. Además, para medir la satisfacción que tiene el paciente con la aplicación se aplicó una encuesta validada que se relaciona con el nivel de satisfacción del cliente. Esperamos que este proyecto cumpla con las metas y los objetivos trazados y supere las expectativas a corto plazo.

**PALABRAS CLAVE:** adherencia al tratamiento, aplicación móvil, paciente, asistencia, control de tratamiento, test, Morisky & Green, Hermes, farmacia.

## **ABSTRACT**

In the Lambayecan population, a problem identified with respect to adherence to medical treatment in older adults when a long or short-term intake regimen has been prescribed. It was considered that this can be overcome through the design and implementation of a mobile application that supports adherence to medical treatment and can help its compliance. For this, it had to guarantee compliance with said treatment with the medical assistants of the patients. It helped to carry out a control of the prescription of the patient and the evolution over time, which improved the compliance of the physicians of the doctors. In addition, increased the level of assistance for medical appointments. This research is experimental / quasi - experimental. A simple random sampling was used, data collection instruments such as surveys, questionnaires and interviews were used. Some examples were: Morisky & Green Test, which is an interview; Test of Hermes, which is a questionnaire; among other validated instruments, documents and fully in force per year. In addition, to measure the satisfaction that the patient has with the application, a validated survey was applied that is related to the level of client satisfaction. We hope that this project complies with the goals and objectives set and exceeds short-term expectations.

**KEYWORDS:** adherence to treatment, mobile application, patient, assistant, treatment control, test, Morisky & Green, Hermes, pharmacy.

<b>I.</b>	<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>II.</b>	<b>MARCO TEÓRICO .....</b>	<b>7</b>
2.1.	ANTECEDENTES.....	7
2.1.1.	ANTECEDENTES INTERNACIONALES.....	7
2.1.2.	ANTECEDENTES NACIONALES.....	8
2.1.3.	ANTECEDENTES LOCALES.....	9
2.2.	BASES TEÓRICO CIENTÍFICAS .....	10
2.2.1.	ANDROID.....	10
2.2.1.1.	¿Qué es Android? Breve descripción y reseña .....	11
2.2.1.2.	Estructura de Android .....	12
2.2.1.3.	Evolución de Android .....	13
2.2.2.	PROCESO UNIFICADO DE RATIONAL (RUP) .....	14
2.2.2.1.	Ciclo de vida de RUP .....	15
2.2.2.2.	Etapas y fases del desarrollo .....	16
2.2.3.	RECURSOS EMPLEADOS PARA EL DESARROLLO DEL PROYECTO .....	17
2.2.4.	ADHERENCIA AL TRATAMIENTO MÉDICO.....	18
2.2.4.1.	Factores que causan el incumplimiento y la no – adherencia .....	19
2.2.4.2.	Metodologías interpersonales para mejorar la adherencia al tratamiento.....	20
2.2.4.3.	Test que miden la adherencia al tratamiento médico.....	21
2.2.4.4.	Ventajas y desventajas de los Test .....	29
2.2.5.	DEFINICIÓN DE TÉRMINOS BÁSICOS .....	31
<b>III.</b>	<b>METODOLOGÍA .....</b>	<b>32</b>
3.1.	TIPO Y NIVEL DE INVESTIGACIÓN .....	32
3.1.1.	TIPO DE INVESTIGACIÓN .....	32
3.1.2.	NIVEL DE INVESTIGACIÓN .....	32
3.2.	DISEÑO DE INVESTIGACIÓN.....	32
3.3.	POBLACIÓN, MUESTRA Y MUESTREO .....	33
3.3.1.	POBLACIÓN.....	33

3.3.2. MUESTRA.....	33
3.4. CRITERIOS DE SELECCIÓN .....	34
3.5. OPERACIONALIZACIÓN DE VARIABLES.....	34
3.5.1. VARIABLES .....	34
3.5.1.1. Variable independiente .....	34
3.5.1.2. Variable dependiente .....	34
3.5.2. INDICADORES (OPERACIONALIZACIÓN DE VARIABLES) .....	35
3.6. TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS.....	36
3.7. PROCEDIMIENTOS .....	36
3.8. MATRIZ DE CONSISTENCIA .....	37
Objetivo General.....	37
Objetivos Específicos .....	37
3.9. CONSIDERACIONES ÉTICAS.....	38
3.10. METODOLOGÍA DE DESARROLLO.....	38
<b>IV. RESULTADOS .....</b>	<b>39</b>
4.1. RESULTADOS DEL PRE-TEST.....	39
4.2. RESULTADOS RESPECTO AL DESARROLLO: DOCUMENTACION RUP.....	41
4.3. RESULTADOS RESPECTO AL PACIENTE DESPUES DEL USO DE LA APLICACIÓN (POST TEST).....	41
4.3.1. MEJORA DEL NIVEL DE CUMPLIMIENTO DE LOS TRATAMIENTOS MÉDICOS DE LOS PACIENTES .....	42
4.3.2. AUMENTO DEL NIVEL DE ASISTENCIA A CITAS MÉDICAS .....	42
4.3.3. AYUDA AL PACIENTE A LLEVAR UN CONTROL DE SU TRATAMIENTO Y EVOLUCIÓN A TRAVÉS DEL TIEMPO .....	43
4.3.4. AYUDAR AL PACIENTE A LLEVAR UN CONTROL DE SU TRATAMIENTO Y EVOLUCIÓN A TRAVÉS DEL TIEMPO .....	45
<b>V. DISCUSIÓN.....</b>	<b>45</b>
<b>VI. CONCLUSIONES.....</b>	<b>48</b>
<b>VII. RECOMENDACIONES .....</b>	<b>49</b>
<b>VIII. LISTA DE REFERENCIAS .....</b>	<b>50</b>

<b>IX. ANEXOS .....</b>	<b>55</b>
<b>ANEXO N° 01: TEST DE CUMPLIMIENTO SMAQ.....</b>	<b>55</b>
<b>ANEXO N° 02: RESULTADOS DEL TEST DE CUMPLIMIENTO SMAQ .....</b>	<b>56</b>
<b>ANEXO N° 03: ANALISIS DE RIESGOS.....</b>	<b>63</b>
<b>ANEXO N° 06: ACEPTACION POR EXPERTOS.....</b>	<b>67</b>
<b>ANEXO N° 07: ESTRUCTURA TECNOLOGICA.....</b>	<b>68</b>
<b>ANEXO N° 08: MANUAL DE USUARIO .....</b>	<b>70</b>
<b>ANEXO N° 09: ESTRUCTURA RUP.....</b>	<b>92</b>



## ÍNDICE DE TABLAS

<b>TABLA 1.DISEÑO DE CONTRASTACIÓN DE HIPÓTESIS .....</b>	<b>33</b>
<b>TABLA 2.INDICADORES.....</b>	<b>35</b>
<b>TABLA 3.TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS. ....</b>	<b>36</b>
<b>TABLA 4.RESULTADOS DE LAS PREGUNTAS DICOTÓMICAS DEL PRE-TEST.....</b>	<b>40</b>
<b>TABLA 5.RESULTADOS DE LAS PREGUNTAS DICOTÓMICAS DEL PRE-TEST EN PORCENTAJES. ..</b>	<b>40</b>
<b>TABLA 6.RESULTADOS DEL ÍTEM 5 DEL PRE-TEST EN PORCENTAJES. ....</b>	<b>40</b>
<b>TABLA 7.RESULTADOS DEL ÍTEM 6 DEL PRE-TEST EN PORCENTAJES. ....</b>	<b>41</b>

## ÍNDICE DE FIGURAS

<b>FIGURA 1. USUARIOS DE SMARTPHONES E IMPACTO EN EL MUNDO.....</b>	<b>2</b>
<b>FIGURA 2. NÚMERO DE SMARTPHONES VENDIDOS AL USUARIO FINAL A NIVEL MUNDIAL DE 2011 A 2016 (EN MILLONES DE UNIDADES) .....</b>	<b>3</b>
<b>FIGURA 3. ACCESO A SMARTPHONE SEGÚN EDAD.....</b>	<b>3</b>
<b>FIGURA 4. SISTEMAS OPERATIVOS MÁS UTILIZADOS.....</b>	<b>12</b>
<b>FIGURA 5. ESTRUCTURA DE ANDROID .....</b>	<b>13</b>
<b>FIGURA 6. EVOLUCIÓN DE ANDROID .....</b>	<b>14</b>

## I. INTRODUCCIÓN

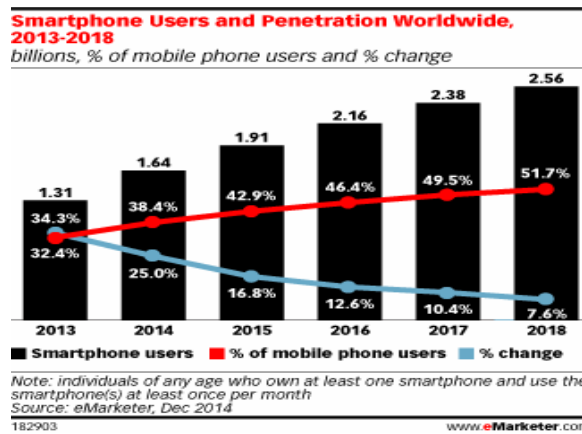
En los últimos años la tecnología y las telecomunicaciones han avanzado a tal punto de involucrarse en la vida cotidiana, especialmente si se habla de medicina. GSM Association [1], una de las más importantes organizaciones de teléfonos móviles, publicó un estudio de mercado denominado Mobile Economy 2017 en el Congreso Mundial de móviles realizado en la Feria de Barcelona en España. Este estudio contiene estadísticas importantes acerca del uso de la tecnología móvil de varios sectores en el mundo incluida Latinoamérica. Se vio que para el 2016, el total de usuarios de un teléfono móvil, Smart o convencional alcanzó el 65% de la población mundial (4.800 millones de personas). En Latinoamérica, el 70% de la población cuenta con un teléfono celular. La mayoría (45%) cuenta con conexión 3G y un 39% cuenta con conexión 2G. Solo el 16% dispone de conexiones 4G. Sin embargo, se pronosticó que para el 2020, las personas que cuenten con tecnología 4G sea un 38%. “Lo móvil es una plataforma global que hoy soporta a dos tercios de toda la población mundial, entregando conectividad e infraestructura cuya energía se basa en las nuevas economías digitales en donde, además, se trabaja en los retos socio económicos modernos”, dice Mats Granryd, director general de GSMA.

Desde los primeros meses de este año se ha escuchado la nueva versión en conexión que sería llamada 5G y se espera que para los próximos años se implemente en gran parte de los dispositivos. Por lo pronto la conexión 4G es normal en los smartphones. GMSA declaro que en el 2016 se identificaron 580 redes 4G que se lanzaron en 188 países, cubriendo el 60% de la población mundial. “Nuestro último reporte de la Economía Móvil revela cómo la ubicuidad y la cercanía de los teléfonos inteligentes y la conectividad de alta velocidad está permitiendo la innovación en áreas como la Inteligencia Artificial, así como manejando la transformación digital “, añade Granryd, citando una inversión en las redes globales desde el 2010 de billones de dólares.

En 2014 eMarker [2], compañía estadounidense dedicada a la investigación de mercado, publicó un artículo donde se reveló que para el 2015 se estimaba un total de 1.91 mil millones de usuarios smartphones en el mundo y para el 2016 esta cifra ascendería a 2.16 mil millones de usuario. Habrían además 2.38 mil millones en 2017 y 2.56 mil millones para el 2018. [Figura 1]

Además de esto, eMarketer presentó una tabla cuantitativa con los países que más consumidores de smartphone tendrán a lo largo de los años. Para el primer lugar, como era de esperarse, se encontraría China (704.1 millones) seguida de U.S (220 millones) y el primer país latinoamericano llegaría al 6to lugar con Brasil (71.9 millones).

**Figura 1. Usuarios de Smartphones e impacto en el mundo**



**Fuente:** eMarketer. 2014. *Smartphone Users and Penetration Worldwide, 2013-2018.*

De esta manera se puede ver que la tendencia a tener un teléfono móvil crece y esto conllevaría al uso de estos por personas desde pequeños hasta adultos. Sin embargo, estas cifras eran una estimación y no fue sino hasta un estudio publicado en 2017 por Statista [3], la base de datos de estadística más grande del mundo, donde se muestra el número de usuarios finales que han adquirido un smartphone. Podemos observar que, en el 2016 aproximadamente 1.500 millones de equipos llegaron a un consumidor final, es decir a manos de una persona que utiliza el smartphone propiamente dicho. [Figura 2]

Así, hemos podido observar cómo la creciente demanda de smartphones ha ido predominando en el mercado de las telecomunicaciones alrededor del mundo.

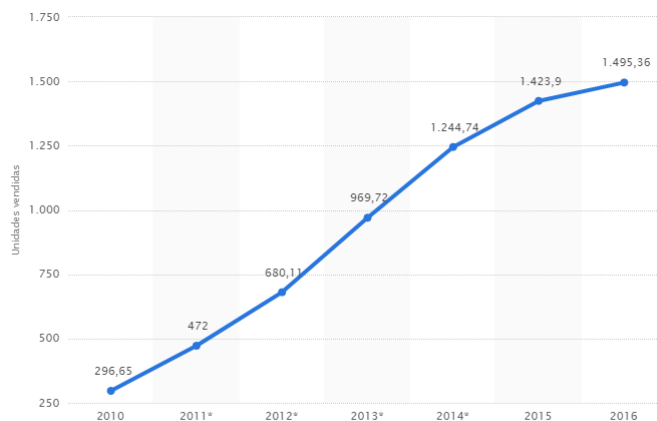
Concretamente en Perú, a finales del año 2016, Tineo [4] publicó en el Diario Gestión un estudio realizado por el IDC (International Data Corporation) donde se proyectaron un aumento en las ventas de teléfonos inteligentes, concretamente entre el 7% y 9% lo que significa una cifra cercana a los 9 millones. Esto supone un aumento del 27% en ventas respecto al 2015 (6.6 millones). Para finales del 2017 se pronostica un aumento del 5% y 7% lo que significa unos 9 millones de smartphones. El vicepresidente de Marketing de Alcatel indicó para Latinoamérica: “Este incremento se debe a que los peruanos optan por un dispositivo de esta clase que destaque por ser un aliado en la vida diaria”.

Respecto a la población a la que va dirigida esta aplicación móvil, según un estudio realizado por OSIPTEL [5] “en el último año, se registró un crecimiento en los usuarios de todas las edades, destacando un avance de 7.5 puntos porcentuales entre las personas de 46 a 50 años, y de 4.7 puntos porcentuales entre los mayores de 51 años.”

En cuanto a las estadísticas que concierne al acceso de un smartphone de las personas entre 36 y 45 años, se ha incrementado de un 22% a un 41.3% (entre

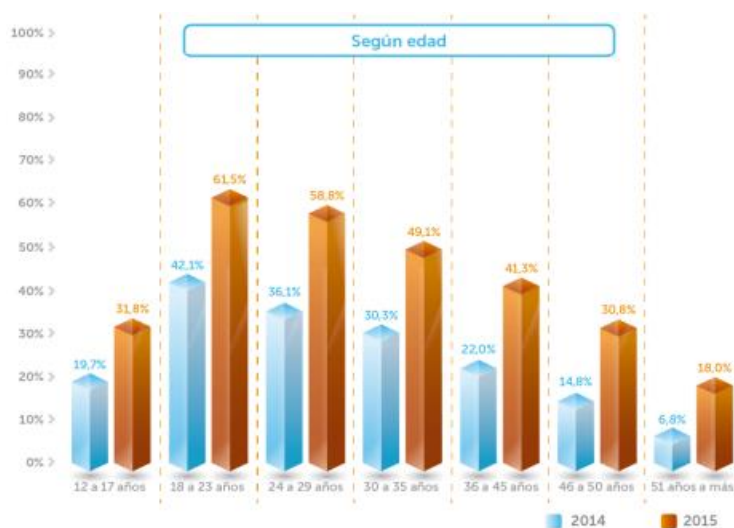
2014 y 2015); en personas de 46 a 50 años de un 14.8% a un 30.8% y en personas de 51 años a más de un 6.8% a un 18%. Lo que indica el creciente uso de smartphones en personas de 35 a 60 años y afirma el uso por ellas, lo que llevaría a poder utilizar también una aplicación móvil.

**Figura 2. Número de smartphones vendidos al usuario final a nivel mundial de 2011 a 2016 (en millones de unidades)**



**Fuente:** Statista. 2017. Número de smartphones vendidos en el mundo al usuario final.

**Figura 3. Acceso a Smartphone según edad**



**Fuente:** OSIPTEL. 2016. Acceso a telefonía móvil se acerca al 100% en Perú

Adicional a esto, según la publicación en un reporte emitido por Compañía Peruana de Estudios de Mercados y Opinión Pública CPI [6], se observa que, mediante una investigación realizada por el Instituto Nacional de Estadística e Informática INEI (2017), se estima que al menos el 18% de la población del Perú

tiene entre 40 – 55 años (5.5% más que las personas de 18 – 24 años representada por el 12.5%) y un 13.3% tiene 56 a más años.

Precisamente en el departamento de Lambayeque, entorno geográfico donde se espera realizar las pruebas una vez desarrollada la presente investigación, existe una población cuantiosa. En el mismo estudio mencionado por el INEI[6], muestra que hasta mediados de este año existen al menos 300 mil personas entre 35 y 60 años. Otro resultado del INEI (2017), publicado por el Ministerio de Salud del Perú [7], muestra que hacia el 2016 la población sólo en Chiclayo, respecto a personas entre el mismo rango de edad, llegaba a la suma de 90338 habitantes, para este año se espera que sobrepase los 100 mil habitantes. Esto indica que existe una población considerable a la que va dirigida esta investigación y que estarían en condiciones de utilizar los productos de la misma.

Existe un problema respecto a esta población y más en el adulto mayor: la adherencia al tratamiento en enfermedades crónicas o no. Según un estudio realizado en la Universidad de Ciencias de Filadelfia por Daniel Hussar y publicado en la web de Merck Sharp & Dohme (MSD) [8], una de las más grandes empresas farmacéutica en el mundo, la adherencia al tratamiento es “el grado en el cual una persona sigue las prescripciones farmacológicas según lo indicado”. Según la Organización Mundial de la Salud OMS[9] se refiere a “el grado en que el comportamiento de una persona —tomar el medicamento, seguir un régimen alimentario y ejecutar cambios del modo de vida— se corresponde con las recomendaciones acordadas de un prestador de asistencia sanitaria”. Sin embargo, en los países desarrollados, por ejemplo, la observancia y seguimiento es apenas al 50% de los pacientes. OMS[10].

Uno de los primeros estudios realizados acerca de la adherencia al tratamiento tuvo lugar en 1990 por Col et. al. [11] donde se determinó que un 58% de los pacientes cometía algunos errores al momento de cumplir con lo prescrito por el médico. Esta cifra es idéntica a lo que American Heart Association (AHA) estima en 59%. Y eso se mantiene hasta la actualidad pues según Sabaté [12] la adherencia a nivel mundial ha tenido cifras alarmantes dado que actualmente se estima que, mediante un estudio estadístico, al menos un 59% de la población mundial no cuenta con una correcta adherencia al tratamiento y esto repercute en hospitalizaciones derivadas del incumplimiento de mismo.

El Dr. Derek Yach, Director Ejecutivo de Enfermedades no Transmisibles y Salud Mental de la OMS, declaró que “el incumplimiento del tratamiento es la principal causa de que no se obtengan todos los beneficios que los medicamentos pueden proporcionar a los pacientes, está en el origen de complicaciones médicas y psicosociales de la enfermedad, reduce la calidad de vida de los pacientes, aumenta la probabilidad de aparición de resistencia a los fármacos y desperdicia recursos asistenciales”. OMS [10].

Hussar en su investigación también menciona un apartado acerca de la adherencia al tratamiento en el adulto mayor donde detalla que la vejez natural de la persona y su deterioro físico y mental provocan que la adherencia al tratamiento se vea disminuida. A su vez resalta mucho la comunicación entre médico – paciente para fortalecer y mejorar la adherencia al tratamiento.

Por su parte, Silva et. al [13] describe que “el 25% de los pacientes mayores de 65 años ingieren seis o más medicamentos y que el 25% de todos los que están en la tercera edad consumen al menos cuatro medicamentos diferentes”. Además, afirma que “de todos los grupos de edad, los ancianos son los que más se benefician con los avances tecnológicos en medicamentos”. He aquí un apoyo a la hipótesis de la investigación descrita posteriormente.

Un estudio realizado en el Hospital Cayetano Heredia publicado por Málaga et. al. [14] analiza el nivel de adherencia al tratamiento médico en este hospital general del Perú. Concretamente se tomó como foco de atención a las personas que sufrían HTA (Hipertensión Arterial) que es una de las enfermedades más extendidas a nivel mundial. El objetivo de este estudio fue determinar la adherencia al tratamiento en pacientes con hipertensión, dado que es una enfermedad con un tratamiento de larga duración y en algunos casos de por vida.

Aquí también se hizo uso de algunos test para medir la adherencia al tratamiento como lo es el cuestionario de Morisky-Green, el cual está certificado y validado para ser empleado en los pacientes. En el grupo muestral se tomaron a 103 personas donde el 80.6% de ellos tenían más de 60 años (adulto mayor), el 69.9% eran mujeres y el 77% tenían un tiempo de enfermedad mayor a 3 años. Los resultados fueron alarmantes, apenas un 37.9% llevaba una adherencia al tratamiento adecuada.

Según Málaga et. al. [14] “en nuestro país, en hospitales de EsSalud en Trujillo y en Lima en el Hospital Nacional Edgardo Rebagliati encontraron tasas de adherencia entre 54% y 63%” y agrega que “la falta de adherencia es un hecho recurrente y un problema global; es justamente la baja adherencia a terapias en enfermedades crónicas, lo que origina morbi-mortalidad que podría ser evitada”.

Así pues, podemos ver que nuestro país no es ajeno a este problema en los pacientes. Pero ¿por qué existe esto? MSD [8] declara como primera causa de la no adherencia al tratamiento a olvidarse de tomar el fármaco. Luego tiene lugar la falta de entendimiento o malinterpretación de las instrucciones, experimentar efectos secundarios, el sabor u olor del fármaco, entre otros. Por su parte, en una publicación de Project Inform [15], una organización estadounidense dedicada a mejorar salud y el empoderamiento de personas con VIH, resalta que en un estudio realizado en pacientes con dicho virus se pudo determinar algunas de las causas por las que los pacientes omitieron las dosis, entre ellas: el 40% simplemente se

olvidó, el 37% estaba dormido durante el momento de la dosis, el 34% se encontraba fuera de casa y el 27% había cambiado su rutina de terapia.

Esta investigación se enfoca en las dos primeras causas: el olvido y la malinterpretación del tratamiento, de manera que sean minimizadas bajo el uso de herramientas que puede proporcionar la metodología, como lo afirmó Silva [13].

Con estos argumentos, se decidió realizar la investigación en un hospital del departamento de Lambayeque para tener una población definida y formada. Entonces, bajo estos argumentos, ¿cómo se podría apoyar la adherencia al tratamiento mediante una aplicación móvil?



## II. MARCO TEÓRICO

### 2.1. Antecedentes

#### 2.1.1. ANTECEDENTES INTERNACIONALES

J. Riobos [16], narra la problemática de la Universidad Autónoma de México donde da constancia de conflictos al momento de gestionar las actividades y dar a conocer y promocionar los eventos llevados en ella con el fin de que todo aquel que desee participar lo pueda hacer. Se aplicó la metodología -, logrando obtener como solución el desarrollo de una aplicación Android que unifique eventos y actividades académicas a base de J-Event. El valor agregado de esta investigación es que a través del producto final se puede brindar al usuario una herramienta que le proporcione información de manera intuitiva de todas aquellas actividades que puedan ser de su interés y no solo de carácter académico sino también de aquellas actividades de carácter extracurricular. Finalmente, el autor concluyo que de esta manera se logra difundir información de manera ordenada, rápida y precisa para los estudiantes. Se tomó en consideración esta tesis ya que su objetivo es unificar eventos e información. De la misma manera, esta investigación busca unificar información de los tratamientos médicos de los pacientes y gestionarlos a través de eventos que son alertas en forma de notificación para cumplir con la adherencia a dicho tratamiento.

Eva Milagros Blanco Delgado [17] narra la problemática al momento de obtener una autenticación para el ingreso a un sistema, por ejemplo; puesto que normalmente las personas tienen la necesidad de desplazarse para identificarse ante una entidad de registro y con cierto grado de seguridad. Se aplicó la metodología -, logran como solución el diseño y desarrollo de una aplicación en plataforma Android para el uso de identidades digitales, autenticación y firmas digitales. El valor agregado de esta investigación es que el usuario puede obtener una identidad digital de la que puede hacer uso en cualquier momento que necesite autenticarse lo que reduce tiempos y agiliza el proceso de comprobar la identidad de una persona. Finalmente, la autora concluyó que es importante contar con un sistema de autenticación para la seguridad de cualquier empresa. Se tomó en consideración esta tesis ya que es desarrollada en plataforma Android y nos brinda un panorama de seguridad a tener en cuenta en el desarrollo del producto acreditable de esta investigación.

Diego Miguel Arribas Marcos [18], narra la problemática de una casa de estudios donde se busca agilizar el proceso de exámenes dando la posibilidad de resolver preguntas simples, múltiples y de autoevaluación en cualquier momento y desde cualquier lugar. Se aplicó la metodología -,

logrando una solución basada en una aplicación Android como gestor de exámenes y calificaciones. El valor agregado de esta aplicación es que permite crear y gestionar exámenes o preguntas de autoevaluación para cualquier materia o curso. De esta manera los alumnos pueden acceder a estos cuestionarios via online desde sus smartphones y pueden responder sus evaluaciones. Además, cuenta con una aplicación web desde donde se puede administrar el sistema. Finalmente, el autor concluyó que esta aplicación permitía mantener a los alumnos en constante evaluación y agilizar el proceso de exámenes y entrega de calificaciones ya que desde la misma aplicación se obtiene el puntaje resultado. Se tomó en consideración esta tesis porque busca hacer las actividades de evaluación de una manera más intuitiva y poder gestionar las actividades de una manera rápida y sencilla.

### **2.1.2. ANTECEDENTES NACIONALES**

Cristina Archilla Manso [19], narra la problemática respecto a la dificultad que tienen las personas mayores al utilizar un smartphone de manera que carecen de la destreza necesaria para manejarlos. Esto conlleva a una exclusión social ya que existe la incapacidad de marcar un número de teléfono hasta incluso enviar un mensaje de texto o encender el teléfono. Se aplicó la metodología -, logrando obtener como solución una aplicación de telefonía móvil para estas personas. El valor agregado de esta investigación es que se desarrolla una aplicación móvil que cambia completamente la apariencia visual del teléfono y lo vuelve como un teléfono antiguo; esto permite pulsar un botón y efectuar una llamada de la forma más sencilla y amigable posible. Finalmente, el autor concluyó que gran cantidad de usuarios que probaron esta aplicación obtuvieron una mayor destreza que sin usarla. Se tomó en consideración esta tesis ya que se enfoca al adulto mayor y su relación con los smartphones y última tecnología. Se logra establecer una relación amigable entre ellos, lo que los conduce a erradicar esa exclusión social.

Pedro Marginal Marina [20], narra la problemática de la necesidad de orientación en el tema de circuitos digitales y sus ejercicios. Se aplicó la metodología -, logrando tener como solución una aplicación Android de problemas de circuitos digitales combinacionales. El valor agregado de esta investigación es ofrecer a los estudiantes un conjunto de ejercicios de síntesis de circuitos que han sido adaptados para ser resueltos en una pantalla de tamaño promedio de un smartphone. Además, permite obtener correcciones de tus ejercicios cuando lo desees, ofrece ayuda para solucionar los problemas y permite compartir las soluciones con otros usuarios para conseguir una interacción constante y relacionar a los estudiantes. Finalmente, el autor concluyó que la aplicación aumento el nivel de

desempeño de los estudiantes respecto al curso de circuitos digitales corroborando el impacto de la aplicación en los mismos. Se tomó en consideración esta tesis ya que se busca tener una interacción constante con el usuario y de esta manera intuitiva aprender del curso.

Alvaro Días Arroyo [21], narra la problemática de la necesidad de usuarios aficionados al deporte por saber y estar al tanto de las estadísticas deportivas de sus equipos y personajes favoritos. Se aplicó la metodología -, logrando como solución una aplicación de estadísticas deportivas en Android. El valor agregado de esta investigación es el análisis y tratamiento masivo de datos relativos a eventos deportivos de fútbol y baloncesto, para obtener estadísticas orientadas a los servicios que brindan las casas de apuestas deportivas. Según el autor “la aplicación recopila datos tales como: Goles, Saques de esquina, Tarjetas, Clasificaciones históricas, Calendario de encuentros, Información de partidos, Información de equipos y contiene las siguientes funcionalidades: Clasificaciones de fútbol de las ligas más importantes a nivel europeo, Clasificaciones de baloncesto de las ligas más importantes a nivel mundial, Estadísticas detalladas de cada equipo en ambas disciplinas, Calendario para mostrar los partidos que hay en el día, Función para filtrar por días el calendario.” Finalmente el autor concluyó que las casas de apuestas utilizan un algoritmo estadístico para ofrecer el servicio de apuestas a sus clientes con escenarios que pueden cambiar en tiempo real y por consiguiente cambiar también las estadísticas. Se tomó en consideración esta tesis ya que utiliza un algoritmo que está siendo implementado en los últimos años y se puede ver localmente, además de manejar estadísticas para obtener información a través de los datos. De esta misma manera el producto acreditable de esta investigación utilizará estadística para brindar reportes.

### **2.1.3. ANTECEDENTES LOCALES**

Almendra Oblitas Guevara [22], narra la problemática que sufren los turistas locales y extranjeros al momento de visitar nuestra región puesto que no reciben información suficiente acerca de eventos y lugares turísticos que cumplan sus expectativas respecto a sus intereses. Se aplicó la metodología -, logrando obtener como solución una aplicación móvil multiplataforma para guiar al turista en su estadía por la región Lambayeque. El valor agregado de esta investigación es el uso de un algoritmo de recomendación que ofrece al usuario eventos y lugares turísticos de acuerdo a sus intereses. Así, se tiene acceso rápido y fácil disminuyendo el tiempo que se utiliza en consultar destinos turísticos, restaurantes y alojamientos. Finalmente, el autor concluyó que los turistas obtuvieron recomendaciones mas certeras respecto al perfil de usuario que crearon y los intereses que establecieron.

Se tomó en consideración esta tesis ya que trata de mejorar alguna actividad desarrollada por cualquier persona en cualquier parte de la localidad, turista o no.

Edgard Gonzalo Estela Vásquez [23], narra la problemática respecto a la falta de comunicación entre todas aquellas personas que actúan como agentes educativos en el colegio San Agustín de Chiclayo. Se aplicó la metodología -, logrando como solución una aplicación móvil de alertas para apoyar la comunicación en la que cualquier evento importante del centro de estudios se pueda publicar entre los integrantes de la familia estudiantil como fiestas de aniversario, reuniones, eventos e incluso incidencias dentro del salón de clase relacionada con algún estudiante. Todos estos reportes llegarían de manera rápida a docentes, directivos y hasta padres de familia que deseen estar al tanto de lo que pasa en las instituciones respecto al nivel, grado y sección de cada uno de los alumnos. El valor agregado de esta investigación es principalmente la comunicación como principal objetivo, manteniendo al tanto de todo a todos los usuarios relevantes. Se tomó en consideración esta tesis ya que se enfoca en un control de incidentes dentro de una aplicación, lo que da una idea de envío de alertas de un smartphone principal a los demás a manera de notificaciones push-up.

Javier Compte Andreu [24], narra la problemática que sufren las personas con Trastorno Espectro Autista respecto a problemas fonéticos y semánticos al momento de leer y comprender. Se aplicó la metodología de la lectura global, logrando obtener como solución una aplicación móvil para apoyar a la lectura global. El valor agregado de esta investigación radica en que la lectura es una vía que sirve como sistema alternativo del lenguaje oral, mejorando problemas fonéticos y semánticos, lo que facilita la expresión y comprensión de sensaciones. Finalmente, el autor concluyó que mediante esta investigación se ha logrado acerca a las personas con TEA a la lectura y ha propiciado un aprendizaje lecto-escritor. Se tomó en consideración esta tesis ya que se ha utilizado la tecnología para atender un problema de personas discapacitadas y apoyar en un aspecto importante para el desarrollo de su persona.

## **2.2. Bases teórico científicas**

### **2.2.1. ANDROID**

En este apartado hablaremos acerca de Android, sus orígenes y su transformación a través del tiempo (versiones del sistema operativo).

### **2.2.1.1. ¿Qué es Android? Breve descripción y reseña**

Android, propiedad de Google, es un sistema operativo y una plataforma de software basado en Linux y desarrollado para ser utilizado en dispositivos móviles (smartphones, tablets, Smart Tv, etc.). Está basado en la filosofía software libre, cuenta con la licencia GNU GPL v2, la cual deja a disposición de los desarrolladores este sistema para ser modificado, por lo que los fabricantes de teléfonos móviles pueden personalizarlo como mejor se ajuste a sus necesidades. Sanz [25]

Fue desarrollado por Android Inc., empresa creada inicialmente por Andy Rubin, Rich Miner, Chris White y Nick Sears; fundada en 2003 en la ciudad de Palo Alto en California.

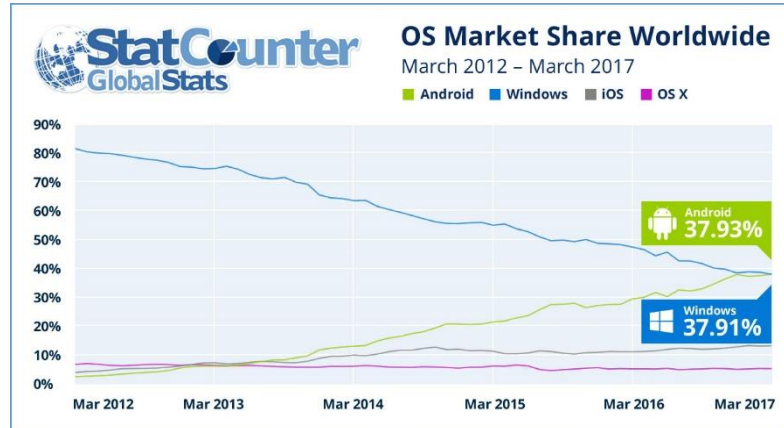
Hacia 2005, Google logra identificar el gran proyecto que tenían en manos y decide comprar a Android Inc., y el 5 de noviembre del 2007 es lanzado oficialmente Android 1.0, llamado Apple Pie y el primer dispositivo en tener Android fue un HTC Dream. Al mismo tiempo se crea la Open Handset Alliance (conglomerado de fabricantes y desarrolladores de software).

Así pues, a través del tiempo, Google ha conseguido abarcar casi la totalidad del mercado de smartphones con este extraordinario sistema operativo. Tal y como lo afirma Android en su página oficial (Android.com), es “el sistema operativo más popular del mundo” y esto abarca desde teléfonos, relojes, televisores y hasta automóviles. De hecho, un artículo periodístico publicado por Lance [26] en el sitio de CNET (la red multinacional de multimedia más grande del mundo), declara que, después de un estudio, en EE. UU., la cuota de mercado de Android creció al 65.5%; en la Unión Europea ascendió hasta el 75.6% y en China se colocó en el 77%. Así, a nivel mundial según Gutierrez [27] para CNET, asegura que “Android controla el 86% del mercado mundial”.

Concretamente en Perú, por un estudio realizado por el Diario Gestión [28] se concluyó en que “el 66.59% de los usuarios de teléfono móviles en el Perú opta por el sistema operativo de Google”. Dado que, para fines del 2013 las ventas de smartphones Android fueron del 66.59% y de IOS el 23.20% y un 5.35% para Windows Phone. En 2016, el Diario Gestión [29] publicó unos estudios realizados por el IMS (Internet Media Services), donde se obtuvo que un 85% personas que navegan en internet lo hacen desde un smartphone Android.

Así pues, vemos la prevalencia de smartphones en el mundo y en el Perú y además que los usuarios prefieren Android por encima de IOS y de cualquier otro sistema operativo conocido.

**Figura 4. Sistemas Operativos más utilizados**



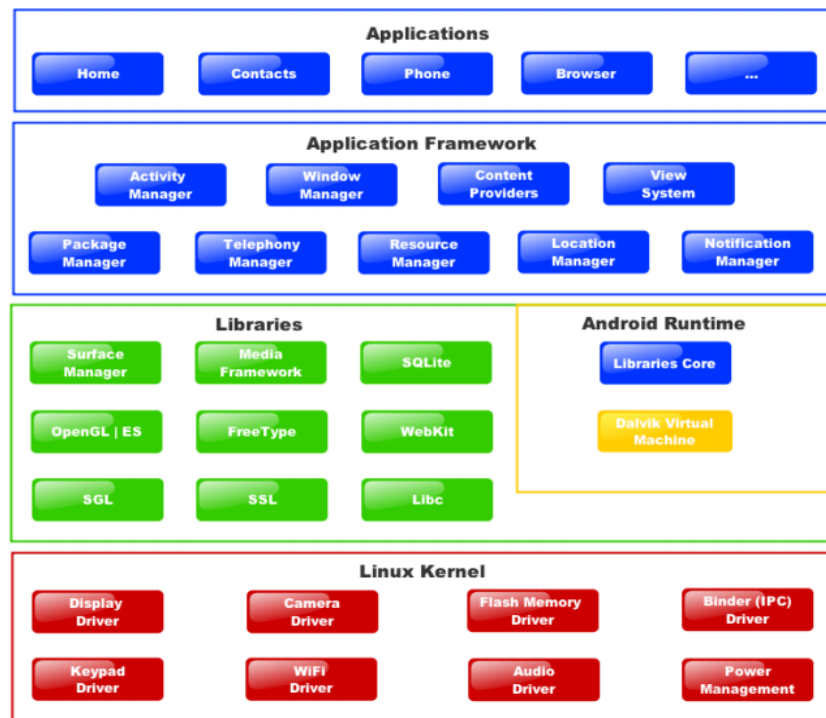
*Fuente: Periódico el País. 2017. Android ya es el sistema operativo más usado del mundo. España*

### 2.2.1.2. Estructura de Android

La estructura de Android según el Departamento de Software de Comunicaciones de la Universidad Carlos III de Madrid [30], está dividida en 5 niveles diferentes:

- **Aplicaciones:** aquí se encuentran todas las aplicaciones incluidas por defecto de Android, las aplicaciones instaladas por el operador telefónico (si lo hubiera), las que instala el usuario, ya sea de la tienda de aplicaciones o de su propio desarrollo. Se ejecutan aquí servicios y se ponen en acción las API y librerías de los niveles anteriores.
- **Framework de aplicaciones:** representa el conjunto de herramientas de desarrollo de cualquier aplicación. Todas las aplicaciones que contenga el smartphone deberán usar la misma API.
- **Librerías:** son un conjunto de líneas de código (generalmente en C/C++) utilizadas por Android y son las que brindan la funcionalidad. Estas librerías y el núcleo basado en Linux representan el “corazón” de Android.
- **Tiempo de ejecución de Android:** conformado por las Core Libraries (Java Classes y Máquina Virtual Dalvik)
- **Kernel Linux 2.6:** contiene los drivers para poder trabajar en conjunto con el hardware.

**Figura 5. Estructura de Android**



- **Fuente:** Universidad Carlos III de Madrid. 2012. Programación en dispositivos móviles portables: Arquitectura Android.

### 2.2.1.3. Evolución de Android

Desde la primera versión del sistema operativo Android, no se ha parado de trabajar para mejorarla y optimizar su funcionamiento. Google ha trabajado muy duro desde 2008 hasta hoy y han conseguido desarrollar concretamente 8 versiones (sin contar las sub versiones). Según la página oficial de Android [31], la evolución de Android a través del tiempo es:

- v1 Apple Pie (2008)
- v1.1 Banana Bread (2009)
- v1.5 Cupcake (2009)
- v1.6 Donut (2009)
- v2.0 – v2.1 Eclair (2009)
- v2.2 – v2.2.3 Froyo (2010)
- v2.3 – v.2.3.7 Gingerbeard (2010)
- v3.0 – v3.2 Honeycomb (2011)
- v4.0 Ice Cream Sandwich (2011)
- v4.1 – v4.3 Jelly Bean (2012)
- v4.4 Kit Kat (2013)
- v5.0 Lollipop (2014)

- v6.0 Marshmallow (2015)
- v7.0 Nougat (2016)
- v8.0 Oreo (2017)

Si bien es cierto, este año se presentó la última versión de Android 8.0 Oreo, un estudio realizado por Android Authority citado en el diario Peru.com [32] revela que el sistema operativo más utilizado por los usuarios de Android es Android 6.0 Marshmallow (31.2%), seguido de Android 5.0 – 5.1 Lollipop con un total de 30.8% de usuarios. Mientras que Android 7.0 y 7.1 llegan a 8.9% y 0.6% respectivamente. Es necesario tener en cuenta esto para la asignación de APIs en el desarrollo del proyecto.

**Figura 6. Evolución de Android**



**Fuente:** Foro de Internet. Consultassmartphones.com

### 2.2.2. PROCESO UNIFICADO DE RACIONAL (RUP)

RUP es una metodología desarrollada inicialmente por Grady Booch, Ivar Jacobson y James Jacobson entre 1998 y 1999. Actualmente tiene soporte por la empresa Rational Software y junto con el Lenguaje Unificado de Modelado (UML) es la metodología más utilizada en el desarrollo de aplicaciones orientados a objetos y esto se debe a la facilidad para adaptarse a la organización y a las necesidades.



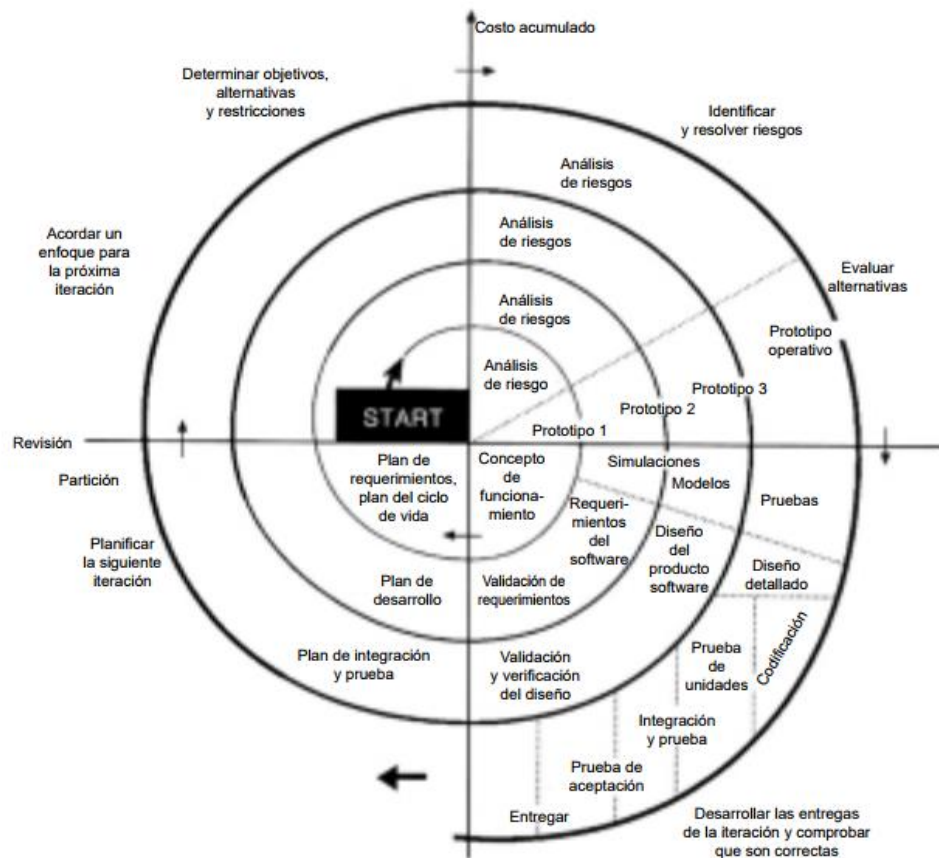
RUP es un conjunto de metodologías adaptables al contexto y necesidades del cliente. Es utilizada desde su año de lanzamiento hasta ahora porque permite documentar todo avance o progreso en el desarrollo de un software. Cabe mencionar que se rige por el Modelo en Espiral.

### 2.2.2.1. Ciclo de vida de RUP

El ciclo de vida de RUP es una implementación del modelo en espiral. Según Castro [33] “el modelo se organiza en un conjunto de iteraciones que puede considerarse a sí mismas como pequeños proyectos que sigue el ciclo de vida completo”.

El modelo en espiral da prioridad a algunas practicas fundamentales del desarrollo de software como la orientación al manejo de riesgos, la orientación al cliente y el desarrollo iterativo.

*Figura 7. Modelo en Espiral*



*Fuente: Mc. Connell. 1997. Desarrollo y Gestión de Proyectos Informáticos. España: Mc. Graw-Hill*

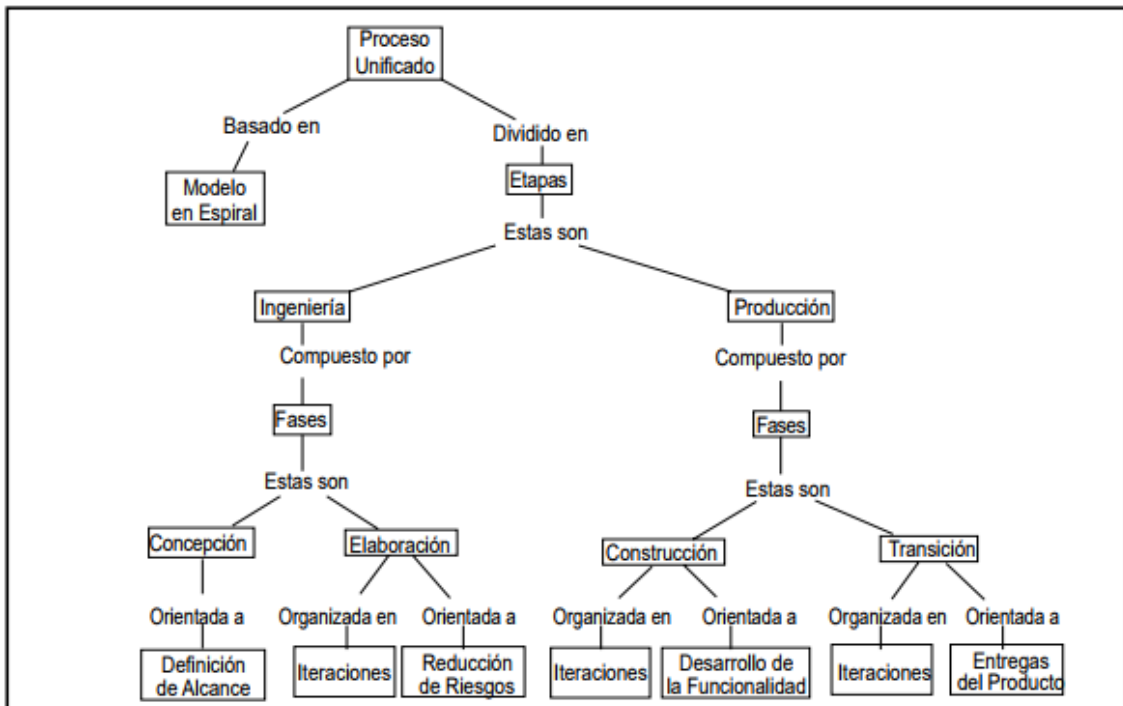
Dado esta representación gráfica del modelo en espiral, RUP toma este modelo y adapta su metodología a él.

### 2.2.2.2. Etapas y fases del desarrollo

Castro [33] define dos grandes etapas que engloban las fases de RUP ya conocidas, estas etapas son: Ingeniería y Producción. La etapa de Ingeniería engloba la fase de Inicio y Elaboración; la etapa de Producción engloba la fase de Construcción y Elaboración.

En la Figura 8 se puede apreciar con más detalle.

**Figura 8. Proceso Unificado de Rational**



Fuente: Castro (2004)

Según Bik Niels [34] las fases de RUP son las siguientes:

- **Inicio:** es la fase de concepción del proyecto donde se define y se acuerda el alcance del mismo con el cliente. Se identifican los riesgos asociados, se propone una vista general de la arquitectura del software y se produce un plan de fases y de iteraciones. Además, cabe resaltar que pasa esta fase se definen los casos de uso a desarrollarse y posteriormente documentarse. Junto con esto se debe plantear también

- el plan de pruebas para evaluar el desarrollo durante todo el ciclo de vida del software, así pues, debe ejecutarse desde la primera iteración de la fase de elaboración y refinarse sucesivamente.
- **Elaboración:** en esta fase se define la arquitectura del software en base a los casos de uso seleccionados y el primer análisis del dominio del problema. Además, se diseña la primera solución del problema y se inicia con el plan de manejo de riesgos (según prioridades). Al final, después de determinar la viabilidad del proyecto, se definen los planes de acción en las estabas de construcción y transición.
  - **Construcción:** se puede dar inicio a la construcción del sistema una vez se haya constituido un conjunto suficiente de requisitos. En esta fase se fusiona el recurso tecnológico y los requisitos para encontrar la mejor manera posible de construir el sistema.
  - **Transición:** en esta fase el grupo de trabajo se asegura del buen funcionamiento del sistema y la disponibilidad de este para trabajar con el cliente. Se busca disminuir los errores y defectos encontrados y brindar capacitaciones a los usuarios finales para la correcta interacción entre el producto y el cliente. Además, se debe verificar que el producto cumpla con todas las funcionalidades definidas en las fases anteriores.

### 2.2.3. RECURSOS EMPLEADOS PARA EL DESARROLLO DEL PROYECTO

- **Java:** Según Steven[35], en una publicación a la página oficial de IBM, el lenguaje Java es un derivado del lenguaje C y sigue sus reglas de sintaxis. Mantiene el paradigma orientado a Objetos (POO). Además de esto, tiene su propia estructura, reglas de sintaxis y forma de programarse.

La estructura más básica de Java comienza en los paquetes, este es el “mecanismo de espacio de nombre del lenguaje Java”. Dentro contienen las clases y en estas a las variables y métodos.

Java presenta los siguientes componentes:

- **Compilador:** verifica el código programado y lo convierte en archivos .class.

- **JVM (Java Virtual Machine):** lee e interpreta archivos .class, los ejecuta y muestra al usuario.
  - **Recolector de basura:** básicamente es un proceso que se realiza en segundo plano y actúa haciendo un seguimiento y qué objetos no están siendo usados por la aplicación, al identificarlos recupera la memoria que utilizan. A este proceso se le llama “gestión de memoria implícita”.
  - **Kit de Desarrollo:** contiene una librería de clases con utilidades predefinidas para el desarrollo de aplicaciones.
  - **Java Runtime Environment (JRE):** contiene las bibliotecas de códigos de la JVM y componentes para programas en ejecución.
- **HTML:** Lenguaje modelador de páginas web. Actúa dando estilo y vista al cliente.
  - **JS:** provee los recursos de animación para la página web.
  - **CSS:** provee los estilos de diseño para la página web.
  - **Frameworks:** diseños predefinidos para trabajar sobre ellos.
  - **Google Maps API:** Se utiliza un API de Google para tener acceso a los mapas. Además, esta API permite tener acceso a la localización del usuario en tiempo real para así poder ofrecerle información relevante de lugares cercanos a él.
  - **Base de datos remota:** será necesaria para almacenar la información de nuestra base de datos y poder acceder a ella. Se contratará un servidor de base de datos.

#### 2.2.4. ADHERENCIA AL TRATAMIENTO MÉDICO

Según la OMS [9], la adherencia al tratamiento se define como “El grado en que el comportamiento de una persona —tomar el medicamento, seguir un régimen alimentario y ejecutar cambios del modo de vida— se corresponde con las recomendaciones acordadas de un prestador de asistencia sanitaria.” Siguiendo la misma línea, Holguín et. al. [36] definen a la adherencia o adhesión al tratamiento como “colaboración y participación proactiva y voluntaria del paciente con su tratamiento, para obtener unas mejores condiciones de salud y de vida, que van más allá del cumplimiento pasivo de las indicaciones del profesional de la salud.”

De acuerdo con Silva [13] la adherencia al tratamiento o a la terapia “se ha definido como el contexto en el cual el comportamiento de la persona coincide con las recomendaciones relacionadas con la salud e incluyen la capacidad del paciente para: asistir a las consultas

programadas (consultorio/hospital), tomar los medicamentos como se prescribieron, realizar los cambios de estilo de vida recomendados y completar los análisis o pruebas solicitadas”. De esta manera también define a la no – adherencia como el incumplimiento de la adherencia.

Silva [13] también señala cuatro factores cruciales que definen la adherencia o no – adherencia al tratamiento. Estos son: “qué tan alta es la prioridad de tomar la medicación, la percepción de gravedad de la enfermedad, la credibilidad en la eficacia de la medicación y aceptación personal de los cambios recomendados”.

#### **2.2.4.1. Factores que causan el incumplimiento y la no – adherencia**

En el transcurso del tiempo donde el paciente está siendo medicado y se le están administrando fármacos para su tratamiento médico, surgen factores, tanto relacionados con el paciente como con el tratamiento en sí, que dificulta la adherencia al mismo. Según Silva [13], tenemos:

- **Factores asociados al paciente:** circunstancias que influyen directamente en el paciente
  - Deterioro sensorial: muchos pacientes tienen dificultad en la vista o el oído y a veces puede que las prescripciones dadas oralmente o de manera escrita no pueda ser interpretada de manera correcta por el paciente. Además de esto, el paciente puede presentar falta de movilidad o dificultad para desplazarse normalmente.
  - Deterioro cognoscitivo y estados alterados del ánimo: produciendo que el paciente no procese bien la información acerca de su tratamiento médico como demencias, depresiones, ansiedad, entre otros.
  - Aspectos de la enfermedad: algunos casos terminan por la adaptación del paciente a los síntomas de su enfermedad acabando en situaciones deplorables. La adherencia es difícil cuando la condición del paciente supone problemas de consciencia y el desempeño social del individuo como por ejemplo las enfermedades mentales.
- **Factor ambiental:** existen pacientes que viven solos (en su mayoría mujeres) y con frecuencia fallan al cumplir

con el tratamiento llegando al extremo de solo consumir la medicina cuando hay alguna afección contundente y de no ser así, dejarlo.

- **Factor asociado al medicamento:** se entiende que existe más alta adherencia al tratamiento cuando el proceso de recuperación es corto y baja cuando el tratamiento es largo. Y algunas las causas más frecuentes de la no – adherencia por causa de los medicamentos son: tomar dosis duplicadas, tomar la dosis en la hora equivocada, omitir la dosis, confundirse de medicamento, entre otros.
- **Factor de interacción médico – paciente:** este factor es muy importante puesto que el cumplimiento también depende gran parte en medida de cómo el médico ha explicado al paciente su tratamiento. Con claridad y de manera concisa se deben establecer los horarios, las medicinas correspondientes, la forma y otras indicaciones relevantes. La adherencia puede verse afectada si el paciente no entiende bien el tratamiento que debe llevar y esto traería consecuencias.

#### 2.2.4.2. Metodologías interpersonales para mejorar la adherencia al tratamiento

Según Silva [13] estas podrían ser algunas de las metodologías para mejorar el tratamiento:

- **Información Efectiva:** el equipo médico a cargo del paciente debe tener en cuenta su nivel mental y socio – cultural para poder explicarle a detalle su tratamiento. Se busca evitar en lo mayor posible el uso de palabras técnicas; sin embargo, si es obligatorio se debe explicar al paciente con la paciencia y el detalle necesarios.
- **Comunicación Asertiva:** la comunicación es primordial para lograr el objetivo. La relación médico – paciente no debe ser autoritaria sino dejar que el paciente intervenga en la toma de decisiones para lograr una toma de conciencia suficiente y apoyar a la adherencia.
- **Educación consciente:** es necesario educar al paciente en su tratamiento médico. Es importante hacerlo en cada visita del paciente así el resultado es una mejor adherencia a las recomendaciones.

- **Ayudas directas para mejorar la adherencia:** existen formas para ayudar al paciente en cumplir con su tratamiento, desde sencillas fichas hasta dispositivos electrónicos que indican con una luz parpadeante cuando el paciente ha olvidado su dosis. También los familiares y los cuidadores personales ayudan en este aspecto. Toda herramienta que facilite la adherencia al tratamiento es útil y más si es electrónica.

#### 2.2.4.3. Test que miden la adherencia al tratamiento médico

Según un estudio realizado por Chamorro et. al.[37] algunos de los test más conocidos para medir la adherencia al tratamiento médico son:

- **Test de cumplimiento comunicado o test de Haynes – Sacket:** consiste en una entrevista directa con el paciente que consta de tres fases. En la primera, discretamente se le conversa al paciente acerca del incumplimiento de algunas personas con sus tratamientos médicos y se trata de crear un ambiente de conversación agradable para que la persona se sienta cómoda y pueda darnos información acerca de cómo lleva su tratamiento. También se le comenta que existen pacientes que tienen dificultades para cumplir estrictamente con lo definido en su tratamiento. Así se llega a la fase dos donde se le pregunta al paciente de una manera más directa: “¿tiene usted dificultades en tomar los suyos?”. Si el paciente asienta es incumplidor sino puede que esté mintiendo por alguna razón, entonces se seguirá insistiendo con: “¿cómo los toma?": todos los días, muchos días, algunos días, pocos días o rara vez. Así se llega a la tercera fase donde se recoge información relevante de la siguiente reflexión dada al paciente: “Muchas personas tienen dificultad en seguir los tratamientos, ¿por qué no me comenta cómo le va a usted?”.

Como podemos ver este es una entrevista entre un especialista y el paciente para recopilar datos importantes y cruciales que marquen la diferencia respecto a lo que se recopila en alguna encuesta.

- **Test de Morisky – Green:** Inicialmente fue diseñado para las personas con HTA, luego se fue amoldando a diferentes enfermedades, crónicas en su mayoría. Se utiliza una vez más la entrevista para captar las respuestas del paciente de manera directa haciendo una conversación agradable para él. Entre palabras y temas relacionados se van haciendo estas 4 preguntas: ¿Olvida alguna vez tomar los medicamentos para tratar su enfermedad? ¿Toma los medicamentos a las horas indicadas? Cuando se encuentra bien, ¿deja de tomar los medicamentos? Si alguna vez le sienta mal, ¿deja usted de tomarla? Las respuestas correctas deben ser No/Sí/No/No.

Cabe resaltar que este método fue utilizado por Jimenez et. al. [38] en su investigación.

- **Test de la comprobación fingida o “bopus pipeline”:** consiste en pedirle al paciente que traiga una muestra de orina para analizarla y medir la cantidad de medicamento que hay en ella; si se resiste, es un paciente incumplidor.
- **Test de Batalla (Test de conocimiento del paciente sobre la enfermedad):** Diseñado inicialmente para la HTA (pero modificable), se trata de un cuestionario donde se mide el conocimiento que tiene el paciente sobre la enfermedad que presenta suponiendo que mientras más conocimiento de su condición tenga el paciente, mayor será su cumplimiento con el tratamiento.

Las preguntas son: ¿Es la HTA una enfermedad para toda la vida? ¿Se puede controlar con dieta y medicación? Cite 2 o más órganos que pueden dañarse por tener la presión arterial elevada.

- **Test de Prochaska – Diclemente:** Existen seis fases consideradas donde los pacientes incumplidores están en las fases de pre-contemplación, contemplación y preparación y los cumplidores se encuentran en las fases de acción y mantenimiento.
- **Test de Hermes:** es un cuestionario que presenta las siguientes preguntas: ¿Puede decirme usted el nombre del medicamento que toma para la HTA? (Sí/No). ¿Cuántos comprimidos de este medicamento debe



tomar cada día? (Sabe/No sabe). ¿Ha olvidado alguna vez tomar los medicamentos? (Nunca/A veces/Muchas veces/Siempre). En las últimas semanas, ¿cuántos comprimidos no ha tomado? (0-1/2 o más) Toma la medicación a la hora indicada? (Sí/No). ¿Ha dejado en alguna ocasión de tomar la medicación porque se encontraba peor tomándola? (No/Sí). Cuando se encuentra bien, ¿se olvida de tomar la medicación? (No/Sí). Cuando se encuentra mal, ¿se olvida de tomar la medicación? (No/Sí).

Las respuestas se miden mediante una escala de valoración y los que alcanzan 3 o 4 puntos son cumplidores.

- **Test de Herrera Carranza:** se trata de un conjunto de preguntas dirigidas a los pacientes que impulsa la contribución a mejorar la vigilancia del cumplimiento terapéutico de la farmacia comunitaria. Las preguntas son:  
¿Sabe para qué su médico le ha prescrito tal(es) medicamento(s)? Me gustaría conocer cómo se siente después de terminar por completo su tratamiento. ¿Sabe cómo tiene que tomar su medicación? ¿Ha puesto en conocimiento de su médico que ha terminado el tratamiento?
- **Test de Cumplimiento SMAQ (The Medication Adherence Questionnaire):** se trata de un cuestionario validado que consta de 6 preguntas orientada a la población infectada con el virus de inmunodeficiencia adquirida (VIH). Las preguntas son las siguientes: 1. ¿Alguna vez olvida tomar la medicación? (Sí/No). ¿Toma siempre los fármacos a la hora indicada? (Sí/No). ¿Alguna vez deja de tomar los fármacos si se siente mal? (Sí/No). ¿Olvidó tomar la medicación durante el fin de semana? (Sí/No). En la última semana, ¿cuántas veces no tomó alguna dosis? (A: ninguna/B: 1-2/C: 3-5/D: 6-10/E: más de 10). Desde la última visita, ¿cuántos días completos no tomó la medicación? (Días: ...).

De acuerdo con lo que el paciente conteste se le considera un cuestionario dicotómico, puesto que la orientación de las respuestas sugiere el incumplimiento

o no. La pregunta 5 puede tratarse de manera proporcional.

- **Escala simplificada para detectar problemas de cumplimiento (ESPA) con el tratamiento antirretroviral:** este cuestionario orientados a las personas con VIH puede ser considerada como la herramienta más precisa de todas que cuenta con 6 preguntas dicotómicas (respuesta correcta supone 1 punto e incorrecta, 0 puntos) y ha sido validada respecto a los registros de dispensación de farmacias, además de contar con un alto grado de sensibilidad, especificidad y eficiencia.
- **The Medication Adherence Report Sale (MARS) (Escala informativa de cumplimiento con la medicación):** es cuestionario más extenso de todos los ejemplares con 30 preguntas para el paciente. Un resultado de 25 puntos o mayor indica un cumplimiento aceptable y de lo contrario, un cumplimiento no óptimo. Thomson et. al. (2000) [39] utilizó este método en su investigación de la esquizofrenia (psicosis) y George et. al (2005) [40] la emplearon para medir el cumplimiento terapéutico en pacientes EPOC.
- **Brief Medication Questionnaire (BMQ) (Breve cuestionario de la medicación):** Ha sido un cuestionario utilizado para diversas enfermedades crónicas como la hipertensión y la diabetes.

#### 2.2.4.4. Ventajas y desventajas de los Test

Test	Ventajas	Desventajas
<b>Haynes Sackets</b>	Este método validado es uno de los más sencillos para utilizar en atención primaria. Es breve, económico y con escasos requisitos de nivel sociocultural para su comprensión. Si el paciente es incumplidor es un método fiable y presenta alto valor predictivo positivo y una especificidad aceptable.	Tiene un bajo valor predictivo negativo. Tiende a sobrestimar el cumplimiento. Posee una baja sensibilidad
<b>Morisky – Green</b>	Es breve y muy fácil de aplicar. Validado y aplicado en numerosas patologías. Puede proporcionar información sobre las causas del incumplimiento. Presenta una alta especificidad, alto valor predictivo positivo y escasos requisitos de nivel sociocultural para su comprensión. Es económico.	Subestima al buen cumplidor y sobrestima el no cumplimiento. Baja sensibilidad. Bajo valor predictivo negativo
<b>Boggus – Pipeline</b>	Método muy sencillo, económico, breve, comprensible y fácil de aplicar. Puede usarse para corroborar el incumplimiento	Tiende a sobrestimar el cumplimiento. No validado
<b>Batalla</b>	Breve, comprensible, económico y muy fácil de aplicar. Presenta una buena sensibilidad. Aplicable y validado para diversas enfermedades	Tiende a sobrestimar el cumplimiento. Baja especificidad y exactitud media
<b>Prohasca – Diclemente</b>	Método muy sencillo, económico, breve y fácil de aplicar. Aplicable a enfermedades crónicas	Tiende a sobrestimar el cumplimiento. No validado. Bajo valor predictivo negativo
<b>Hermes</b>	Método muy sencillo, económico, breve y fácil de aplicar. Se puede aplicar a cualquier enfermedad	Tiende a sobrestimar el cumplimiento. Bajo valor predictivo negativo

<b>Herrera – Carranza</b>	Método muy sencillo, económico, breve y fácil de aplicar. Se puede aplicar a cualquier enfermedad	Puede sobrestimar el cumplimiento. No validado. Bajo valor predictivo negativo
<b>SMAQ</b>	Método validado, muy sencillo, breve, económico y fácil de aplicar. Se puede usar como semicuantitativo, asignándose un porcentaje de cumplimiento	Tiende a sobrestimar el cumplimiento. Bajo valor predictivo negativo
<b>ESPA</b>	Validado para detectar problemas de cumplimiento con el tratamiento antirretroviral en pacientes positivos para el VIH. Presenta elevadas sensibilidad, especificidad y eficiencia. Podría ser adaptado a otras enfermedades debido a su versatilidad	Requiere conocimientos sobre la reposición de medicación y la evolución clínica, lo cual dificulta su aplicación en farmacia
<b>MARS</b>	Es un test autoaplicado. Utilizado en enfermedades crónicas. Validado	Es excesivamente largo y puede ocurrir que muchos pacientes no devuelvan el test o no lo completen adecuadamente. Requiere un buen nivel sociocultural para su comprensión
<b>BMQ</b>	Test autorreferido. No es muy largo y permite analizar el cumplimiento y sus causas. Presenta una gran sensibilidad y permite realizar validaciones de otros tests	Presenta un complejo procedimiento de valoración. Requiere un nivel sociocultural medio para su comprensión

*Fuente: Chamorro, Miguel Ángel, Emilio García, Pedro Amariles, Alfonso Rodríguez y María José Faus. 2008. Revisión de Tests de medición del cumplimiento terapéutico utilizados en la práctica clínica. España*

### 2.2.5. DEFINICIÓN DE TÉRMINOS BÁSICOS

**Plataforma:** es un sistema que sirve como base para trabajar sobre él con módulos de software o hardware que sean compatibles entre sí.

**Linux:** es un sistema operativo creado por Linus Torvalds en 1991. Se refiere a un sistema de código libre disponible para que otros programadores trabajen sobre él. Baig & Aulí [41]

**Sistema operativo:** “es un programa que tiene encomendadas una serie de funciones cuyo objetivo es simplificar el manejo y la utilización del computador, haciéndolo seguro y eficiente”. De Miguel & Perez [42]

**Framework:** o marco de referencia en inglés. Es un conjunto de archivos y directorios que facilitan la creación de aplicaciones, con funcionalidades ya creadas implementadas en un determinado lenguaje de programación. (IEEE, 2015)

**API:** significa Interfaz de Programación de Aplicaciones. En un post digital hecho en la web de TicBeat (perteneciente a Alex Springer España, industria especializada en industria e innovación), se define un API como un conjunto de reglas (código) y especificaciones que rigen a las aplicaciones respecto a la comunicación entre ellas. Las API sirven de interfaz entre aplicaciones diferentes. Merino [43]

**Kernel:** “el corazón del sistema: se encarga de arrancar el sistema y, una vez éste ya es utilizable por las aplicaciones y los usuarios, gestiona los recursos de máquina en forma de gestión de memoria, sistema de ficheros, entrada/salida, procesos e intercomunicación de procesos”. Jorba, en una publicación de la Universidad Abierta de Catalunya [44].

**Librería de clases:** dentro del paradigma de programación orientada a objetos, se refiere a un conjunto de clases que contienen métodos útiles para el programador. Obtenemos esta API al descargar el JDK.

**Adherencia al tratamiento:** según la OMS [9] “es el grado en que el comportamiento de una persona se corresponde con las recomendaciones acordadas de un prestador de asistencia sanitaria.”

**HTA:** Abreviación de Hipertensión Arterial.

**VIH:** Abreviación de Virus de Inmunodeficiencia Humana.

**EPOC:** Abreviación de Enfermedad pulmonar obstructiva crónica.

### **III. Metodología**

#### **3.1. Tipo y nivel de investigación**

##### **3.1.1. Tipo de investigación**

Según Hernández Sampieri [45] una investigación experimental es aquella donde se manipula a la variable dependiente haciendo uso de la variable independiente; es decir, que la primera influye en la segunda para alterar algún aspecto en específico. De esta manera se puede evaluar si la hipótesis planteada es afirmada o negada.

En esta investigación experimental el investigador interviene en un grupo de personas (adulto mayor) a los cuales se les presentará una aplicación móvil para apoyar a su adherencia al tratamiento médico y afirmar o negar la hipótesis planteada anteriormente. En este caso la variable a manipular será la adherencia al tratamiento médico por parte del adulto mayor, la que se verá apoyada o no según la influencia de la aplicación móvil como variable independiente.

##### **3.1.2. Nivel de investigación**

Descriptiva.

#### **3.2. Diseño de investigación**

Dicho lo anterior, según el mismo autor, el diseño que se ajusta a esta investigación es cuasi-experimental de pretest-posttest con grupo de control. Una investigación cuasi – experimental se caracteriza porque es un plan de trabajo donde se busca estudiar y analizar el impacto de las herramientas y los tratamientos empleados y/o procesos de cambio en la población muestra donde se interviene. Además, existe una manipulación de variable independiente sobre la dependiente para establecer una relación causal que pueda afirmar o negar la hipótesis. Por otro lado, existe una no aleatorización en la formación de los grupos (experimental y de control); es decir que, el investigador no interviene en la formación de los grupos de manera que se llegue a grupos naturales e intactos.

Así pues, se tomará un grupo natural e intacto de personas a las cuales se les analizará previamente para determinar su adherencia al tratamiento médico. Una parte del grupo será el Grupo Experimental y el otro el Grupo Control. Al primer grupo se le presentará la aplicación móvil para su uso. Luego de esto se pondrá en marcha la aplicación por un tiempo estimado para luego volver a determinar su adherencia al tratamiento médicos. Al final, se vuelve a analizar la variable dependiente para constatar si se apoyó o no a la

adherencia al tratamiento médico. De esta manera se pondrá a prueba a la hipótesis en el campo real.

El diseño de contrastación de hipótesis es  $G_E O_1 X O_2 - G_C O_3 - O_4$ .

**Tabla 1. Diseño de contrastación de hipótesis**

<b>Grupo</b>	<b>Pre – test</b>	<b>Aplicación</b>	<b>Post Test</b>
Experimental	<b>O1:</b> Analizar la adherencia al tratamiento antes del uso de la app	<b>X:</b> Uso de la app móvil	<b>O2:</b> Analizar la adherencia al tratamiento después del uso de la app
Control	<b>O3:</b> Analizar la adherencia al tratamiento bajo hábitos del paciente	<b>Sin app móvil</b>	<b>O4</b> Analizar la adherencia al tratamiento bajo hábitos del paciente

*Fuente: Hernández Sampieri. 2010. Metodología de la investigación científica. Quinta edición. México: McGraw Hill*

### 3.3. Población, muestra y muestreo

#### 3.3.1. Población

La población está conformada por pacientes del departamento de Cardiología hospital Las Mercedes. Estos pacientes son en su mayoría de la tercera edad o aproximado que pueden sufrir de enfermedades crónicas o no, pero llevan tratamientos de larga duración; es decir toman una o más pastillas en el día o inter-diario.

Son personas que pueden llevar su día a día interactuando con un equipo móvil o de lo contrario lo maneje la persona que lo asista. De esta manera la unidad de investigación son los pacientes.

#### 3.3.2. Muestra

La población de pacientes que cumplen con estas características no es muy extensa. Existen alrededor de 50 pacientes que cumplen con estar próximos a la tercera edad y que ya están en la tercera edad; así que, para tener más diversidad en las unidades de análisis se practicará con los 50 individuos.

### **3.4. Criterios de selección**

El tipo de criterio de selección adoptado para esta investigación es muestreo por conveniencia, perteneciente al método no probabilístico, que describe que los sujetos serán seleccionados dada la conveniente accesibilidad y proximidad de estos para la persona que redacta el documento.

### **3.5. Operacionalización de variables**

Se tiene la siguiente hipótesis: Si diseñamos e implementamos una aplicación móvil Android se puede apoyar la adherencia al tratamiento médico en el adulto mayor.

En ese sentido las variables son las siguientes:

#### **3.5.1. Variables**

##### **3.5.1.1. Variable independiente**

Aplicación móvil Android

##### **3.5.1.2. Variable dependiente**

Adherencia al tratamiento en el adulto mayor



### 3.5.2. Indicadores (Operacionalización de variables)

*Tabla 2.Indicadores.*

<b>Objetivo específico</b>	<b>Indicador(es)</b>	<b>Definición conceptual</b>	<b>Unidad de medida</b>	<b>Instrumento</b>	<b>Definición operacional</b>
Determinar el nivel actual de adherencia al tratamiento médico.	Nivel de adherencia al tratamiento médico	Indica el nivel de adherencia al tratamiento médico de los pacientes antes de usar la aplicación	Porcentaje	Cuestionario	Número de tomas / total de tomas
Mejorar el nivel de cumplimiento de los tratamientos médicos de los pacientes.	Número de horarios generados	Indica el número de tomas en el horario del paciente	Global	Encuesta	Número de tomas a realizar
Aumentar el nivel de asistencia a citas médicas	Nivel de asistencia a citas médicas	Indica el nivel del paciente para asistir a sus citas médicas	Porcentaje	Encuesta	Numero de citas asistidas confirmadas / Total de citas
Ayudar al paciente a llevar un control de su tratamiento y evolución a través del tiempo.	Nivel de cumplimiento con el horario del tratamiento	Muestra el progreso del tratamiento del paciente a través del tiempo	Global	Encuesta	Número de tomas confirmadas / Total de tomas

### 3.6. Técnicas e instrumentos de recolección de datos

El primer método para emplear, para la recolección de datos, es el Hipotético – Deductivo impuesto por Bernal [46] donde se le describe como un procedimiento que parte de unas afirmaciones en calidad de hipótesis y busca afirmar o refutar dicha hipótesis confrontando fichas aseveraciones con los resultados en base a hechos.

El segundo método que se pondrá en uso es el método de la implementación. Según Berndtsoon et. al. [47] este método de implementación supone su uso cuando se desarrollan nuevas aplicaciones de software para un uso determinado ya que esta nueva herramienta se deberá poner a prueba en el campo a fin de demostrar si realmente posee las ventajas propuestas y se obtienen resultados como los esperados.

*Tabla 3. Técnicas e instrumentos de recolección de datos.*

Método	Instrumentos	Justificación	Elementos de la población	Véase
<b>Hipotético - Deductivo</b>	<b>Encuestas</b>	A través de ella se puede conocer y recopilar datos objetivos y contables de fácil interpretación y procesamiento.	Pacientes en tratamiento médico	Anexo 1 y Anexo 2
	<b>Entrevistas</b>	Para conocer más a fondo al paciente y obtener datos que con la encuesta no se pudieron obtener.		Anexo 3 y Anexo 4
<b>Implementación</b>	<b>Encuesta</b>	Para constatar la facilidad de uso e interacción bajo un modelo definido y validado de encuesta (Encuesta para medir el nivel de satisfacción del cliente)		

### 3.7. Procedimientos

Descrito en el marco teórico (página 16).

### 3.8. Matriz de consistencia

PROBLEMA	OBJETIVOS	HIPOTESIS	VARIABLES	METODOLOGÍA
<p>¿cómo se podría apoyar la adherencia al tratamiento mediante una aplicación móvil?</p>	<p><b>Objetivo General</b> Apoyar la adherencia al tratamiento médico en el adulto mayor.</p> <p><b>Objetivos Específicos</b></p> <ul style="list-style-type: none"> <li>- Determinar el nivel actual de adherencia al tratamiento médico.</li> <li>- Mejorar el nivel de cumplimiento de los tratamientos médicos de los pacientes.</li> <li>- Aumentar el nivel de asistencia a citas médicas.</li> <li>- Ayudar al paciente a llevar un control de su tratamiento y evolución a través del tiempo.</li> </ul>	<p>Si diseñamos e implementamos una aplicación móvil Android se puede apoyar la adherencia al tratamiento médico en el adulto mayor.</p>	<p>Independiente: Aplicación móvil Android</p> <p>Dependiente: Adherencia al tratamiento en el adulto mayor</p>	<p>PROCESO UNIFICADO RATIONAL (RUP)</p> <p>Conjunto de metodologías adaptables al contexto y necesidades del cliente.</p> <p>Fases:</p> <ul style="list-style-type: none"> <li>- Inicio</li> <li>- Elaboración</li> <li>- Construcción</li> <li>- Transición</li> </ul>

### 3.9. Consideraciones éticas

Es necesario aclarar que la aplicación no mejorará la salud del paciente haciéndole tener una mejor calidad de vida porque estaría traspasando los límites de la investigación. El objetivo de este proyecto es mejorar la adherencia al tratamiento médico que se refiere a lograr el mejor cumplimiento de lo que prescribe el médico, pero no se puede garantizar si el paciente mejora o empeora, eso dependerá de otros factores como la alimentación, el ambiente, el estado de ánimo, la edad, el avance de la enfermedad, o incluso de los medicamentos que el doctor le asigna al paciente; pero no de la aplicación.

De esta manera, se debe informar de tal consideración a las unidades de análisis para evitar malas interpretaciones y problemas legales futuros. Esta aplicación es de gestión y control, mas no es una cura para los pacientes.

### 3.10. Metodología de desarrollo

Siguiendo la metodología RUP, considerando el entorno de trabajo donde se realizará la investigación, las actividades en cada una de las fases serán las siguientes:

- **Inicio:**
  - Se realizarán unas entrevistas y encuestas a los pacientes para determinar el nivel inicial que tienen de adherencia al tratamiento. Se busca recolectar información importante que pueda orientar las necesidades de los pacientes (son los clientes de la aplicación)
  - Se construirá el primer bosquejo de vista de arquitectura del software y se analizará para evaluar posibles cambios o mejoras.
  - Se determinará los casos de uso que se desarrollarán según la metodología RUP, luego se documentarán.
  - Se desarrollará el plan de pruebas para ir evaluando la aplicación móvil a lo largo del tiempo y conforme crezca en funcionalidad.
- **Elaboración:**
  - Se definirá la versión final de la arquitectura de la aplicación móvil.
  - Se diseñará el bosquejo de la aplicación que es la solución al problema planteado y se analizará para evaluar posibles cambios o mejoras.
  - Se determinará qué tan viable es el desarrollo de esta aplicación.
  - Se construirán el plan de acción para la construcción y la transición.

- **Construcción:**
  - Una vez recolectada la información y procesada, se definirá la versión final del conjunto de requisitos que demanda el paciente.
  - Se dará inicio a la programación de la base de datos, de la web Service y de la app móvil.
  - Se organizan los tiempos para cada una de las actividades.
- **Transición:**
  - Se realizarán las pruebas necesarias y los test de funcionalidad para la aplicación móvil a fin de que esté listo para usarse por el paciente.
  - Se brindará capacitación al paciente mismo o a su asistente para usar correctamente la aplicación.
  - El paciente también evaluará la efectividad de esta aplicación, será nuestro mejor testeador ya que lo usará en el día a día a través del asistente.

#### **IV. RESULTADOS**

##### **4.1. RESULTADOS DEL PRE-TEST**

Para obtener un cálculo aproximado del nivel de adherencia de los pacientes antes de aplicar nuestro producto acreditable, se ha optado por la aplicación de un test validado denominado Test de Cumplimiento SMAQ.

Tal y como está descrito en el marco teórico, el Test de Cumplimiento SMAQ consta de las siguientes preguntas:

- Item 1. ¿Alguna vez olvida tomar la medicación? (Sí/No).
- Item 2. ¿Toma siempre los fármacos a la hora indicada? (Sí/No).
- Item 3. ¿Alguna vez deja de tomar los fármacos si se siente mal? (Sí/No).
- Item 4. ¿Olvidó tomar la medicación durante el fin de semana? (Sí/No).
- Item 5. En la última semana, ¿cuántas veces no tomó alguna dosis? (A: ninguna/B: 1-2/C: 3-5/D: 6-10/E: más de 10).
- Item 6. Desde la última visita, ¿cuántos días completos no tomó la medicación? (Días: ...).

De acuerdo con lo que el paciente conteste se le considera un cuestionario dicotómico, puesto que la orientación de las respuestas sugiere el incumplimiento o no. La pregunta 5 puede tratarse de manera proporcional.

Se aplicó este cuestionario a 50 personas (los resultados en extensión se encuentran en Anexos) se obtuvieron los siguientes resultados.

**Tabla 4. Resultados de las preguntas dicotómicas del Pre-test.**

	<b>Item 1.</b> ¿Alguna vez olvida tomar la medicación?	<b>Item 2.</b> ¿Toma siempre los fármacos a la hora indicada?	<b>Item 3.</b> ¿Alguna vez deja de tomar los fármacos si se siente mal?	<b>Item 4.</b> ¿Olvidó tomar la medicación durante el fin de semana?
<b>NO</b>	<b>14</b>	<b>37</b>	<b>29</b>	<b>12</b>
<b>SI</b>	<b>36</b>	<b>13</b>	<b>21</b>	<b>38</b>

Veamos ahora los resultados representados en porcentaje:

**Tabla 5. Resultados de las preguntas dicotómicas del Pre-test en porcentajes.**

ITEM	N° 1		N°2		N°3		N°4	
PORCENTAJE	72%	28%	26%	74%	42%	58%	76%	24%
CRITERIO	SI	NO	SI	NO	SI	NO	SI	NO
50	36	14	13	37	21	29	38	12

Cabe resaltar que las primeras 4 preguntas se calcularon proporcionalmente ya que son preguntas dicotómicas (a diferencia del ítem 5 y 6). Estas preguntas se interpretan de la siguiente manera.

De las 50 personas a quienes se les aplicó el Test de Cumplimiento SMAQ, el 72% olvidó alguna vez tomar la medicación y el 28% dice no olvidar la toma de su medicamento. Esta es una de las preguntas más importantes porque se refiere netamente al cumplimiento con el tratamiento médico resultante en apenas 28%. Respecto al ítem 2, sólo el 26% de los pacientes toma siempre los fármacos a la hora indicada y el 74% no. Esto significa que el paciente no es puntual al tomar sus medicamentos y hay una margen considerable de tiempo desde la hora en la que debe tomarlo a cuando realmente lo hace. Por otro lado, respecto al ítem 3, solo el 42% de personas deja de tomar el fármaco en caso de sentirse mal. Esto influye también en el nivel de adherencia que se puede lograr con el paciente. Finalmente, respecto al ítem 4, el 76% de las personas olvida tomar lo fármacos los fines de semana. Este último dato es importante para entender en qué días de la semana las personas tienden más a no cumplir con su régimen de toma de medicamentos prescrita.

Respecto al ítem 5, las respuestas varían entre 5 alternativas. El resultado se muestra en la siguiente tabla.

**Tabla 6. Resultados del ítem 5 del Pre-test en porcentajes.**

Item 5. En la última semana, ¿cuántas veces no tomó alguna dosis?
---

50	N°	%
<b>A. Ninguna</b>	7	14%
<b>B. 1-2</b>	21	42%
<b>C. 3-5</b>	11	22%
<b>D. 6-10</b>	5	10%
<b>E. 10 a más</b>	5	10%

Según la Tabla 6 se puede deducir que en la última semana el 14% de pacientes (7 pacientes) no tomaron alguna dosis de su tratamiento prescrito; sin embargo, el 42% (21 pacientes) no tomó de 1 a 2 dosis de su tratamiento. Además, el 22% (11 pacientes) no tomó de 3 a 5 dosis de su tratamiento y el 10% (5 pacientes) no tomó de 6 a 10 dosis y un 10% (5 pacientes) no tomó de 10 a más dosis de su tratamiento.

Respecto al ítem 6, las respuestas se consolidan en un promedio para saber cuántos días completos no se tomó la medicación. Para esto los pacientes ingresaron un número cualquiera estimado por ellos. En promedio se obtuvo que los pacientes dejaron 3,54 días de tomar sus medicamentos.

***Tabla 7. Resultados del ítem 6 del Pre-test en porcentajes.***

Desde la última visita, ¿cuántos días completos no tomó la medicación?	
Promedio	3,54

#### **4.2. RESULTADOS RESPECTO AL DESARROLLO: DOCUMENTACION RUP**

La documentación RUP se encuentra redactada y ordenada en el Anexo 09.

#### **4.3. RESULTADOS RESPECTO AL PACIENTE DESPUES DEL USO DE LA APLICACIÓN (POST TEST)**

Se aclara que, por motivos de tiempo en el desarrollo de esta tesis, se ha implementado y usado la aplicación durante el número de 60 días; usando 2 día para capacitar al asistente médico y lo restante para su uso.

#### 4.3.1. Mejora del nivel de cumplimiento de los tratamientos médicos de los pacientes

El objetivo presente fue plantado con la necesidad de impulsar al paciente al cumplimiento de su tratamiento médico tal cual fue descrito por el médico de cabecera. Inicialmente, los tratamientos se registraban de manera física en una libreta o de manera mental por parte del asistente; esto traía dos problemas, el primero era que el asistente podría olvidar la toma del fármaco a la hora indicada y, segundo, no se podría tener una estimación exacta de cómo va el paciente con su tratamiento.

Para responder a este objetivo se diseñó e implementó la funcionalidad de registrar tratamientos para el paciente en los que el asistente ingresaba datos como el fármaco a tomar, cada cuánto y por cuánto tiempo tomar y la cantidad de dosis a recibir por el paciente.

Este objetivo genera un reporte llamado el horario del paciente donde se puede ver detalles como el tratamiento y el fármaco del tratamiento que va a tomar, además de la hora a la que debe tomarlo. Esto ayudará a que el asistente tenga un registro completo de las tomas que se deben realizar en cada uno de los tratamientos del paciente y mejorar el nivel de cumplimiento.

La sentencia SQL utilizada para obtener este reporte es la siguiente:

```
select id, id_tratamiento, item, id_paciente, desc_periodo from horario
```

El resultado se puede ver en la siguiente figura:

id integer	id_tratamiento smallint	item smallint	id_paciente character(8)	desc_periodo character varying(25)	hora trata time without time zone
171	39	1	16461697	2018-12-09 14:35:43	14:35:43
174	39	1	16461697	2018-12-10 14:35:43	14:35:43
175	39	1	16461697	2018-12-10 22:35:43	22:35:43
177	40	1	16461697	2018-12-08 22:41:48	22:41:48
181	40	1	16461697	2018-12-10 22:41:48	22:41:48
182	40	1	16461697	2018-12-11 10:41:48	10:41:48
183	40	1	16461697	2018-12-11 22:41:48	22:41:48
184	40	1	16461697	2018-12-12 10:41:48	10:41:48
185	40	1	16461697	2018-12-12 22:41:48	22:41:48

#### 4.3.2. Aumentodel nivel de asistencia a citas médicas

El objetivo presente fue plantado con la necesidad de impulsar al paciente al cumplimiento con las citas que han sido programadas por su médico para sus



controles de salud o lo que crea conveniente. Tal cual los tratamientos médicos, las citas médicas se registraban en un cuaderno o libreta, trayendo consigo el mismo riesgo de olvidar la fecha y la hora de la cita médica y la falta de un control periódico.

En los días de prueba de la aplicación se registró 1 cita a la que asistió el paciente en compañía de su asistente.

Este objetivo genera un reporte: controlar la asistencia a las citas médicas del paciente.

La sentencia SQL utilizada para generar este reporte es la siguiente:

```
select c.id_cita, c.dni_paciente , (p.apellido_paterno ||
' ' || p.apellido_materno || ' ' || p.nombre) as paciente,
h.nombre as hospital , c.fecha_cita, c.hora_cita,
c.estado
from cita c
inner join hospital h on c.id_hospital = h.id_hospital
inner join paciente p on c.dni_paciente = p.dni
where dni_paciente = '16461697'
```

Y el resultado de la misma se visualiza en la siguiente figura:

id_cita smallint	dni_paciente character(8)	paciente text	hospital character varying(100)	fecha_cita date	hora_cita time without time zone	estado character(1)
23	16461697	Ruiz Gonzales Hilda	Pacifico	2018-12-07	10:30:00	R

#### 4.3.3. Ayuda al paciente a llevar un control de su tratamiento y evolución a través del tiempo

El objetivo fue planteado con la necesidad de ayudar al paciente a llevar un control de cómo va en el cumplimiento de su tratamiento, por lo que el asistente puede confirmar en la interfaz del sistema la toma del medicamento. Lo que el sistema captura es el momento exacto de la confirmación y guarda registro en la base de datos. Además, se requiere conocer el margen de incumplimiento del paciente, es decir el tiempo que demora entre la hora que recomienda el sistema y la hora que en realidad toma el fármaco.

Este objetivo, ha generado un reporte donde podemos ver detalladamente la actividad del paciente y su cumplimiento a través del tiempo. Primero vemos la hora programada por el sistema para la toma del fármaco (desc\_periodo) y seguido vemos la hora en la que el asistente confirmó la toma del paciente (fecha\_cum). Además de

esto, podemos ver un margen de tiempo para estimar cuánto se demoró el paciente en tomar el fármaco desde la hora que indicó el sistema. De esta manera se puede estimar un promedio promedio de margen de error.

La sentencia SQL utilizada para generar este reporte es la siguiente:

```
select id, id_tratamiento, item, id_paciente,
desc_periodo, fecha_cum,
age((desc_periodo)::timestamp, (fecha_cum)::timestamp )
as margen
from horario
where (desc_periodo)::timestamp <= current_date
order by id
```

Y el resultado de la misma se visualiza en la siguiente figura:

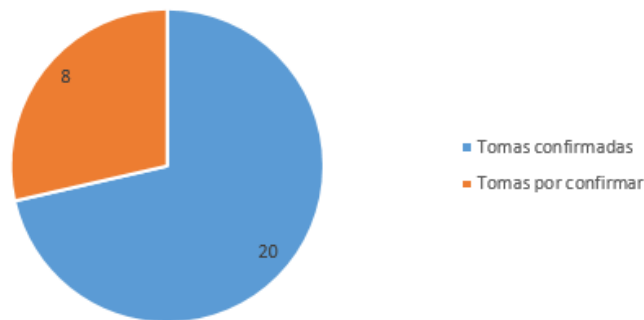
id integer	id_tratamiento smallint	item smallint	id_paciente character(8)	desc_periodo character varying(25)	fecha_cum timestamp without time zone	margen interval
144	28	1	16461697	2018-12-07 20:53:22	2018-12-07 21:00:22	-00:07:00
145	28	1	16461697	2018-12-08 08:53:22	2018-12-08 08:55:22	-00:02:00
146	28	1	16461697	2018-12-08 20:53:22	2018-12-08 20:59:16	-00:05:54
147	28	1	16461697	2018-12-09 08:53:22	2018-12-09 08:55:10	-00:01:48
148	28	1	16461697	2018-12-09 20:53:22	2018-12-09 20:53:01	00:00:21
149	28	1	16461697	2018-12-10 08:53:22	2018-12-10 09:01:10	-00:07:48
158	28	2	16461697	2018-12-08 06:31:41	2018-12-08 06:40:41	-00:09:00
159	28	2	16461697	2018-12-08 14:31:41	2018-12-08 15:31:20	-00:59:39
160	28	2	16461697	2018-12-08 22:31:41	2018-12-08 22:40:14	-00:08:33
161	28	2	16461697	2018-12-09 06:31:41	2018-12-09 08:31:12	-01:59:31
162	28	2	16461697	2018-12-09 14:31:41	2018-12-09 14:45:05	-00:13:24
163	28	2	16461697	2018-12-09 22:31:41	2018-12-09 23:31:14	-00:59:33
164	28	2	16461697	2018-12-10 06:31:41	2018-12-10 08:10:17	-01:38:36
167	39	1	16461697	2018-12-08 06:35:43	2018-12-08 07:00:10	-00:24:27
168	39	1	16461697	2018-12-08 14:35:43	2018-12-08 14:45:54	-00:10:11
169	39	1	16461697	2018-12-08 22:35:43	2018-12-08 22:45:38	-00:09:55
170	39	1	16461697	2018-12-09 06:35:43	2018-12-09 08:01:46	-01:26:03
171	39	1	16461697	2018-12-09 14:35:43	<NULL>	<NULL>
172	39	1	16461697	2018-12-09 22:35:43	2018-12-09 22:45:10	-00:09:27
173	39	1	16461697	2018-12-10 06:35:43	2018-12-10 08:35:10	-01:59:27
176	40	1	16461697	2018-12-08 10:41:48	2018-12-08 10:50:26	-00:08:38
177	40	1	16461697	2018-12-08 22:41:48	<NULL>	<NULL>
178	40	1	16461697	2018-12-09 10:41:48	2018-12-09 10:59:12	-00:17:24
179	40	1	16461697	2018-12-09 22:41:48	2018-12-09 22:45:25	-00:03:37
180	40	1	16461697	2018-12-10 10:41:48	2018-12-10 11:30:12	-00:48:24

- En la columna MARGEN se puede ver cuánto se ha demorado el paciente en tomar el fármaco. Los valores en negativos significan que hubo un retraso y los positivos, un

adelanto. También podemos ver que existen valores nulos, por lo que se entiende que el paciente no tomó el fármaco.

#### 4.3.4. Ayudar al paciente a llevar un control de su tratamiento y evolución a través del tiempo

Dado que en la aplicación móvil se presentan los horarios de las tomas de medicamentos y su cumplimiento/no cumplimiento, el paciente y su médico pueden tener una idea de su evolución a través del tiempo. A partir de esto se puede tener una idea de manera empírica (más allá de la propia consciencia del paciente) de cómo va el progreso del paciente en su tratamiento.



## V. DISCUSIÓN

Según la OMS [9], la adherencia al tratamiento se define como “El grado en que el comportamiento de una persona —tomar el medicamento, seguir un régimen alimentario y ejecutar cambios del modo de vida— se corresponde con las recomendaciones acordadas de un prestador de asistencia sanitaria.” Siguiendo la misma línea, Holguín et. al. [36] definen a la adherencia o adhesión al tratamiento como “colaboración y participación proactiva y voluntaria del paciente con su tratamiento, para obtener unas mejores condiciones de salud y de vida, que van más allá del cumplimiento pasivo de las indicaciones del profesional de la salud.”

De acuerdo con Silva [13] la adherencia al tratamiento o a la terapia “se ha definido como el contexto en el cual el comportamiento de la persona coincide con las recomendaciones relacionadas con la salud e incluyen la capacidad del paciente para: asistir a las consultas programadas (consultorio/hospital), tomar los medicamentos como se prescribieron, realizar los cambios de estilo de vida recomendados y completar los análisis o pruebas solicitadas”. Esto nos da una idea consolidada de que la adherencia al tratamiento se refiere al nivel en que el paciente cumple al pie de la letra las indicaciones de su médico cuando le prescribe un tratamiento médico respecto a alguna afección o enfermedad que se presente.

La discusión sustenta las mejoras realizadas en el contexto del nivel de adherencia al tratamiento médico, específicamente vinculado al apoyo que recibe el paciente por parte del asistente. Es por esta razón que la misma se hará según los indicadores de cada objetivo específico.

Este apartado tendrá la discusión de los indicadores de cada uno de los objetivos específicos, siendo estos:

- **Nivel de adherencia al tratamiento médico**

El proceso que se llevó a cabo para evaluar este indicador fue que se calcula el porcentaje de cumplimiento del paciente respecto a un tratamiento o la totalidad de estos.

Inicialmente en el pre-Test pudimos observar que había mucha falta de cumplimiento por parte de los pacientes que hacía que su nivel de adherencia al tratamiento médico no sea la mejor posible. Las carencias más importantes que pueden ser mencionadas en los resultados del pre-test fueron que el 72% de las personas olvidan tomar la medicación y solo el 26% tomaba los fármacos a la hora indicada. Además, el 42% de las personas encuestadas no tomaron de 1 a 2 dosis y 3,54 días en promedio no tomaron su medicación.

Respecto a esto podemos constatar, mediante los datos de la figura mostrada en los resultados del post test, que los pacientes no dejaron de tomar sus medicamentos. El porcentaje de cumplimiento con cada una de las tomas de fármacos ascendió a un 92%, dejando un 8% como incumplidas es decir que no fueron administradas.

**Diferencia:** las unidades de análisis experimentaron un aumento en la eficiencia y efectividad del control de los tratamientos médicos. Dejando de lado el control empírico para pasar a un control sistematizado, abriendo la posibilidad de lograr un 100% de adherencia al tratamiento médico.

- **Número de horarios generados**

El proceso que se llevó a cabo para evaluar fue el número de registros de horario que genera el sistema para que el paciente tenga una guía del momento correcto de tomar sus fármacos.

Inicialmente no se tenía un registro del horario generado por el asistente para el paciente, todo se controlaba en libretas de manera empírica o por la capacidad de memoria del asistente.

Con el sistema, se generaron automáticamente 42 horarios de diferentes fármacos para diferentes tratamientos que atacan a diferentes afecciones del paciente.

- **Nivel de asistencia a citas médicas**

El proceso que se llevó a cabo para evaluar este indicador fue que se calcula el número de citas médicas agendadas y registradas que han sido realizadas con éxito.

Inicialmente, tal como con los tratamientos médicos, no se tenía un control sistematizado de estas lo que pondría en riesgo el cumplimiento con estas.

Con el sistema implementado, el paciente tiene un control completo de sus citas médicas, lo que ayudaría a elevar su nivel de adherencia al tratamiento médico.

## **VI. CONCLUSIONES**

- 1.** Se logró mejorar el nivel de adherencia del tratamiento médico de los pacientes por medio del registro de citas y tratamientos médicos. Inicialmente los índices que arrojó el pre test indicaban una carencia de adherencia al tratamiento médico sustentada en las estadísticas mostradas en el apartado de resultados. Frente a esto y con la implementación del sistema se logró un cumplimiento del 92% en los días de pruebas que se tuvo con el sistema.
- 2.** Se logró generar horarios automáticos para la toma de fármacos del paciente basado en el fármaco a tomar, la frecuencia de toma (cada 4 horas, cada 8 horas, etc.), y el periodo de días que se extiende el tratamiento. De esta manera se obtiene un horario detallado del momento exacto para tomar el medicamento basándose en lo prescrito por el doctor de cabecera. Esto erradica las equivocaciones en el itinerario que se tenía inicialmente.
- 3.** Se logró registrar la información de las citas médicas para que, del mismo modo que el tratamiento médico, se tenga información detallada y precisa de la misma que permita a la unidad de análisis estar al tanto y no confundirse entre fechas, horas o algún otro factor que traiga consigo la inasistencia a la cita médica.
- 4.** Se logró ayudar al paciente a tener la opción de revisar el progreso y evolución de su tratamiento médico a través del tiempo mediante gráficos de fácil entendimiento.

## VII. RECOMENDACIONES

Se ha considerado ampliar los horizontes del proyecto de investigación para extender la magnitud y relevancia de la información que pueda ser almacenada y, por consiguiente, los reportes que se puedan obtener. Las recomendaciones son las siguientes:

1. **Stock:** Cuando el usuario registre un tratamiento médico, agregar además un stock de fármacos para llevar un control más exacto de las tomas de fármacos y poder estimar la próxima adquisición de medicamentos.
2. **Creación de grupos:** Se podría apoyar al control de los tratamientos médicos de los pacientes si, además de un control por parte del asistente, existe también un control por parte de un grupo de personas. De esta manera, todos los integrantes del grupo interesados en el paciente recibirán alertas si el paciente no está cumpliendo con su tratamiento y podrán ver su progreso.
3. **Registros de mediciones médicas:** además de tener un control del tratamiento médico del paciente, se podría añadir un registro de diferentes aspectos médicos que puedan ser relevantes en el tratamiento o solo de información. Por ejemplo, se podría mantener un registro de peso, presión, niveles de azúcar, hemoglobina, triglicéridos, entre otras mediciones que sean consideradas pertinentes por el usuario.
4. **Expandir la base de datos de los fármacos:** podría ser útil, además de la base de datos nacional de medicamentos, contar con una base de datos de otros países para aquellos fármacos que son traídos del extranjero y son escasos en el país.
5. **Recomendación de precios:** podría ser útil buscar el mejor precio para un medicamento del que tenemos poco o nada de stock y el usuario desea comprar. La aplicación recomendaría los mejores precios en las farmacias más cercanas a la ubicación GPS en la que se encuentra el usuario.
6. **Poder registrar diferentes modalidades de adherencia** como “el tomar el medicamento cuando el paciente sienta alguna afección”, es decir cuando está sujeta a una condición que establezca el médico de cabecera.

## VIII. LISTA DE REFERENCIAS

- [1] GSMA (2017). The mobile Economy [Online]. Barcelona. Available: <https://www.gsmaintelligence.com/research/?file=9e927fd6896724e7b26f33f61db5b9d5&download>
- [2] eMarketer (2014). 2 billion Consumers Worldwide to get Smart(phones) by 2016 [Online]. Available: <https://www.emarketer.com/Article/2-Billion-Consumers-Worldwide-Smartphones-by-2016/1011694>
- [3] Statista. (2017). Número de smartphones vendidos al usuario final a nivel mundial de 2011 a 2016 (en millones de unidades) [Online]. Available: <https://es.statista.com/estadisticas/521667/numero-de-smartphones-vendidos-en-el-mundo-al-usuario-final/>
- [4] R. Tineo (2016, noviembre 22). Los smartphones se disparan en Perú: A fin de año pasarían los 8 millones [Online]. Available: <https://gestion.pe/tecnologia/smartphones-se-disparan-peru-fin-ano-pasarian-8-millones-2175233>
- [5] OSIPTEL (2016, junio). Acceso a telefonía móvil se acerca al 100% en Perú [Online]. Available: [http://www.osiptel.gob.pe/Archivos/Publicaciones/reporte\\_estadistico\\_junio2016/files/asets/common/downloads/Reporte%20Estad.pdf](http://www.osiptel.gob.pe/Archivos/Publicaciones/reporte_estadistico_junio2016/files/asets/common/downloads/Reporte%20Estad.pdf)
- [6] Compañía peruana de estudios de mercado y opinión pública (2017) Perú: población 2017. [Online]. Available: [http://cpi.pe/images/upload/paginaweb/archivo/26/mr\\_poblacion\\_peru\\_2017.pdf](http://cpi.pe/images/upload/paginaweb/archivo/26/mr_poblacion_peru_2017.pdf)
- [7] Ministerio de Salud (2017). Población estimada por edades simples y grupos de edad, según provincia y distrito. Departamentos de Lambayeque - Año 2016 [Online]. Available: <http://www.minsa.gob.pe/estadisticas/estadisticas/Poblacion/PoblacionMarcos.asp?14>
- [8] D. Hussar. Adherencia al tratamiento farmacológico [Online]. Available: <http://www.msmanuals.com/es/hogar/f%C3%A1rmacos/factores-que-influyen-en-la-respuesta-del-organismo-a-los-f%C3%A1rmacos/adherencia-al-tratamiento-farmacol%C3%B3gico>
- [9] Organización Mundial de la Salud (2004). Adherencia a los tratamientos a largo plazo. Pruebas para la acción [Online]. Available: <http://www.paho.org/Spanish/AD/DPC/NC/adherencia-largo-plazo.pdf>
- [10] Organización Mundial de la Salud (2003). El incumplimiento del tratamiento prescrito para las enfermedades crónicas es un problema mundial de gran envergadura [Online] Available: <http://www.who.int/mediacentre/news/releases/2003/pr54/es/>



[11] N. Col, JE. Fanale, P.Kronholm. “The role of medication noncompliance and adverse drug reactions in hospitalizations of the elderly”. JAMA Network (Arch Intern Med), vol 4, no. 150 841-845 pp., 1990.

[12] E. Sabaté (2004). Adherencia a los tratamientos a largo plazo: Pruebas para la acción. Organización Mundial de la Salud/OMS [Online]; traducción de la Organización Panamericana de la Salud/OPS. Aavailable: <http://www.farmacologia.hc.edu.uy/images/WHO-Adherence-Long-Term-Therapies-Spa-2003.pdf>

[13] G. Silva, E. Galeno, J. Correa. “Adherencia al tratamiento: implicaciones de la no – adherencia”. *AMC*. vol. 30, núm. 4 pp. 268-273, 2005. Aavailable: <http://www.redalyc.org/pdf/1631/163113820004.pdf>

[14] G. Málaga, R. Carhuallanqui, G. Diestra, J. Tang. “Adherencia al tratamiento farmacológico en pacientes hipertensos atendidos en un hospital general”, *Revista Médica Herediana*, vol. 21, no. 4, pp. 197-201, octubre-diciembre, 2010. Aavailable: <http://www.redalyc.org/pdf/3380/338038900004.pdf>

[15] Inform Project (2011, enero). “Adherencia: Mantenerse al día con sus medicamentos”, *Project Inform*. pp. 1-5, enero, 2011. Aavailable: [https://www.projectinform.org/pdf/adherence\\_sp.pdf](https://www.projectinform.org/pdf/adherence_sp.pdf)

DE LA 16 A LA 24.

[16] J. Riobos. “Desarrollo de una aplicación Android que unifique eventos y actividades en la UAM (J-Event)”. Madrid: Universidad Autónoma de Madrid, 2017. Aavailable: <https://repositorio.uam.es/handle/10486/679541>

[17] E. Blanco Delgado. “Diseño y desarrollo de una aplicación Android para el uso de identidades digitales, autenticación y firmas digitales en sistemas interactivos”. Madrid: Universidad Autónoma de Madrid, 2014. Aavailable: <http://arantxa.ii.uam.es/~jms/pfcsteleco/lecturas/20140519EvaMilagrosBlancoDelgado.pdf>

[18] D. Arribas Marcos. “Gestor de exámenes y calificaciones en Android”. Madrid: Universidad Autónoma de Madrid, 2016. Aavailable: <https://repositorio.uam.es/handle/10486/669572>

[19] C. Archilla Manso. “Aplicación de telefonía móvil para discapacitados”. Madrid: Universidad Autónoma de Madrid, 2014. Aavailable: <https://repositorio.uam.es/handle/10486/660519>

[20] P. Madrigal Marina. “Aplicación de problemas resueltos de circuitos digitales combinatoriales bajo Android”. Madrid: Universidad Autónoma de Madrid, 2014. Aavailable: <https://repositorio.uam.es/handle/10486/669462>

- [21] A. Díaz Arroyo. “Aplicación Android de estadísticas deportivas”. Madrid: Universidad Autónoma de Madrid, 2017. Aavailable: <https://repositorio.uam.es/xmlui/handle/10486/679359>
- [22] A. Oblitas Guevara. “Aplicación móvil multiplataforma como guía para orientar al turista en su estadía por la región Lambayeque”. Chiclayo: Universidad Católica Santo Toribio de Mogrovejo, 2016.
- [23] E. Estela Vásquez. “Aplicación móvil de alertas para apoyar la comunicación entre los agentes educativos del Colegio San Agustín de Chiclayo”. Chiclayo: Universidad Católica Santo Toribio de Mogrovejo, 2016.
- [24] J. Compte Andreu. “Apoyo a la lectura global mediante dispositivos móviles para personas con TEA”. Madrid: Universidad Autónoma de Madrid, 2017. Aavailable: <https://repositorio.uam.es/handle/10486/679540>
- [25] D. Sanz, M. Saucedo, P. Torralbo. Introducción a Android: Introducción. Primera Edición. Madrid: E.M.E. Editorial, 2015.
- [26] W. Lance (2016). Android le da una paliza a Apple alrededor del mundo [Online]. Aavailable: <https://www.cnet.com/es/noticias/android-apalea-a-apple-alrededor-del-mundo/>
- [27] O. Guttierrez (2016, mayo 11). Sigue siendo el rey: Android controla el 86 por ciento del mercado mundial [Online]. Aavailable: <https://www.cnet.com/es/noticias/android-apalea-a-apple-alrededor-del-mundo/>
- [28] Diario Gestión (2014). Los usuarios peruanos prefieren Android, por encima de IOS [Online]. Aavailable: <https://gestion.pe/tecnologia/usuarios-peruanos-prefieren-android-encima-ios-2085273>
- [29] R. Tineo (2016, noviembre 25). Perú: 9 de cada 10 personas conectadas a internet tienen un smartphone [Online]. Aavailable: <https://gestion.pe/tecnologia/peru-9-cada-10-personas-conectadas-internet-tiene-smartphone-2173139>
- [30] Universidad Carlos III de Madrid (2012). Programación en dispositivos móviles portables: Arquitectura Android [Online]. Aavailable: <https://sites.google.com/site/swcuc3m/home/android/generalidades/2-2-arquitectura-de-android>
- [31] Android (2014). Versiones de Android [Online]. Aavailable: [https://www.android.com/intl/es-419\\_mx/history](https://www.android.com/intl/es-419_mx/history)
- [32] A.Android (2017, junio 7). Android: ¿cuál es la versión más utilizada entre los usuarios? [Online]. Aavailable:

<https://peru.com/epic/epic-mobile/android-cual-version-mas-utilizada-entre-usuarios-noticia-517238>

[33] A. Castro. (2004). Estructura básica del proceso unificado de desarrollo de software. [Online] Aavailable: [http://www.icesi.edu.co/contenido/pdfs/rcaastro\\_estructura-bas-puds.pdf](http://www.icesi.edu.co/contenido/pdfs/rcaastro_estructura-bas-puds.pdf)

[34] B. Niels. (2016). Rational Unifed Process. [Online] Aavailable: [http://www.students.science.uu.nl/~5590329/me/Peer%20Review/Bik,%20N/Assignme ntD\\_3958132\\_Bik\\_DRAFT\\_Review.pdf](http://www.students.science.uu.nl/~5590329/me/Peer%20Review/Bik,%20N/Assignme ntD_3958132_Bik_DRAFT_Review.pdf)

[35] J. Steven (2012). Conceptos básicos del lenguaje Java [Online] Aavailable: <https://www.ibm.com/developerworks/ssa/java/tutorials/j-introtojava1/index.html>

[36] L. Holguín, D. Correa, M. Arrivallaga, D. Cáceres y M. Varela, “Adherencia al tratamiento de hipertensión arterial: efectividad de un programa de intervención biopsicosocial”. *Universitas Psychologica*, Vol 5, No. 3, pp. 511-548. Aavailable: <http://revistas.javeriana.edu.co/index.php/revPsycho/article/view/459>

[37] M. Chamorro, E. García, P. Amariles, A. Rodríguez y M. Faus. Revisión de Tests de medición del cumplimiento terapéutico utilizados en la práctica clínica. España: 2008.

[38] V. Jiménez, G. Amorós, P. Martínez, M. Fernández y M. León. “Estudio descriptivo del cumplimiento del tratamiento farmacológico antihipertensivo y validación del test de Morisky y Green”. *Aten Primaria*, Vol. 5, No. 10, pp. 767-70. Octubre, 1992. Aavailable: <https://www.ncbi.nlm.nih.gov/pubmed/1472599>

[39] Thompson, K., J. Kulkarni & A. Segejew. 2000. “Reliability and validity of a new Medication Adherence Rating Scale (MARS) for the psychoses”, *Schizophrenia Research*, Vol. 3, No. 42, pp. 241-7. Mayo, 2000. Aavailable: <https://www.ncbi.nlm.nih.gov/pubmed/10785582>

[40] George J., D. Kong, R. Thoman, K. Steward. (2005, noviembre). Factors associated with medication nonadherence in patients with COPD [Online]. Aavailable: <https://www.ncbi.nlm.nih.gov/pubmed/16304262>

[41] R. Baig. y F. Aulí. Sistema Operativo GNU/Linux Básico. España: Fundació per a la Universitat Oberta de Catalunya. Primera Edición. Barcelona: noviembre, 2003. Aavailable: <http://softlibre.unizar.es/manuales/linux/868.pdf>

[42] A.De Miguel, P. Perez y F. Perez. (2016). Sistemas operativos [Online] Aavailable: [http://www.elai.upm.es/moodle/pluginfile.php/3574/mod\\_resource/content/1/sistemasoperativosupm.pdf](http://www.elai.upm.es/moodle/pluginfile.php/3574/mod_resource/content/1/sistemasoperativosupm.pdf)

[43] M. Merino (2014, julio 12). ¿Qué es una API y para qué sirve? [Online] Aavailable: <http://www.ticbeat.com/tecnologias/que-es-una-api-para-que-sirve/>

[44] J. Jorba (2014). El Kernel [Online] Available: <https://www.radiosyculturalibre.com.ar/biblioteca/SISTEMAS%20OPERATIVOS/GNU-Linux/Capitaciones/Universidad%20Oberta%20de%20Catalu%C3%B1a%20UOC/5.pdf>

[45] R. Hernández Sampieri, C. Fernández, P. Baptista. Metodología de la investigación científica. Quinta edición. México: McGraw Hill, 2010.

[46] C. Bernal. Metodología de la investigación. Tercera edición. Colombia: Pearson Education, 2010.

[47] M. Berndtsoon, J.Hansoon, B. Olson&B. Lundell. Thesis Projects: a guide for students in computer science and information systems. Suecia: Springer, 2007.

## IX. ANEXOS

### ANEXO N° 01: Test de Cumplimiento SMAQ

Se trata de un cuestionario inicialmente diseñado para personas con VIH, pero adaptable a otras enfermedades:

Las preguntas son las siguientes:

1. ¿Alguna vez olvida tomar la medicación? (Sí/No).
2. ¿Toma siempre los fármacos a la hora indicada? (Sí/No).
3. ¿Alguna vez deja de tomar los fármacos si se siente mal? (Sí/No).
4. ¿Olvidó tomar la medicación durante el fin de semana? (Sí/No).
5. En la última semana, ¿cuántas veces no tomó alguna dosis? (A: ninguna/B: 1-2/C: 3-5/D: 6-10/E: más de 10).
6. Desde la última visita, ¿cuántos días completos no tomó la medicación? (Días: ...).

De acuerdo con lo que el paciente conteste se le considera un cuestionario dicotómico, puesto que la orientación de las respuestas sugiere el incumplimiento o no. La pregunta 5 puede tratarse de manera proporcional.

## **ANEXO N° 02: Resultados del Test de Cumplimiento SMAQ**

### **1. RESULTADOS DEL ITEM 1 AL 4.**

Se agruparon estas preguntas por ser dicotómicas. Las respuestas son las siguientes.

SI/NO
1. ¿Alguna vez olvida tomar la medicación?
2. ¿Toma siempre los fármacos a la hora indicada?
3. ¿Alguna vez deja de tomar los fármacos si se siente mal?
4. ¿Olvidió tomar la medicación durante el fin de semana?

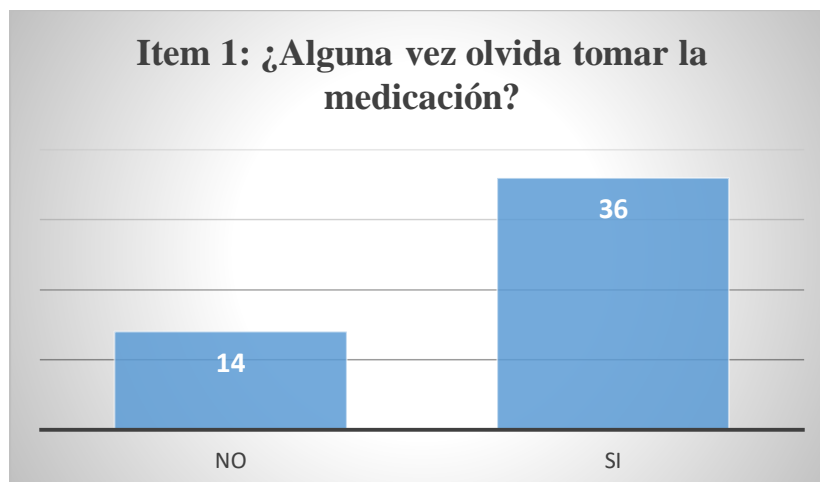
<b>participantes/ ítems</b>	<b>ítem N° 1</b>	<b>ítem N° 2</b>	<b>ítem N° 3</b>	<b>ítem N° 4</b>
participante N° 1	SI	NO	NO	SI
participante N° 2	SI	NO	NO	SI
participante N° 3	SI	NO	NO	SI
participante N° 4	SI	NO	SI	SI
participante N° 5	SI	NO	NO	NO
participante N° 6	NO	SI	SI	SI
participante N° 7	NO	SI	SI	SI
participante N° 8	SI	SI	NO	SI
participante N° 9	NO	NO	NO	NO
participante N° 10	SI	NO	NO	SI
participante N° 11	SI	NO	SI	SI
participante N° 12	NO	NO	NO	NO
participante N° 13	NO	NO	NO	SI
participante N° 14	SI	NO	SI	SI
participante N° 15	SI	NO	SI	SI
participante N° 16	NO	SI	NO	SI
participante N° 17	SI	NO	NO	SI
participante N° 18	SI	SI	NO	SI
participante N° 19	NO	NO	SI	SI
participante N° 20	SI	NO	NO	SI
participante N° 21	SI	NO	SI	NO
participante N° 22	NO	NO	NO	NO
participante N° 23	SI	NO	NO	NO
participante N° 24	SI	SI	NO	SI
participante N° 25	SI	SI	SI	SI
participante N° 26	SI	NO	SI	SI
participante N° 27	SI	SI	NO	SI
participante N° 28	NO	NO	SI	SI
participante N° 29	SI	SI	NO	NO

participante N° 30	NO	SI	NO	SI
participante N° 31	NO	NO	SI	SI
participante N° 32	SI	NO	SI	SI
participante N° 33	NO	NO	SI	SI
participante N° 34	SI	NO	SI	SI
participante N° 35	SI	SI	NO	NO
participante N° 36	SI	NO	SI	NO
participante N° 37	SI	NO	NO	NO
participante N° 38	SI	NO	SI	SI
participante N° 39	SI	NO	SI	SI
participante N° 40	SI	NO	NO	SI
participante N° 41	NO	NO	NO	SI
participante N° 42	SI	NO	NO	SI
participante N° 43	SI	SI	SI	NO
participante N° 44	SI	NO	NO	SI
participante N° 45	SI	NO	SI	SI
participante N° 46	NO	NO	SI	SI
participante N° 47	SI	NO	NO	SI
participante N° 48	SI	SI	NO	SI
participante N° 49	SI	NO	NO	NO
participante N° 50	SI	NO	NO	SI

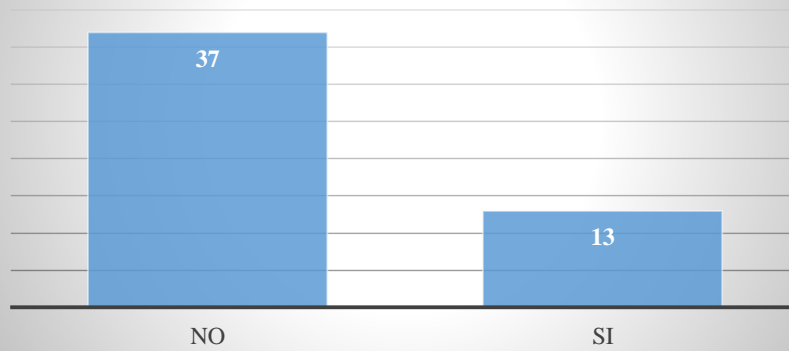
### 1.1. RESULTADOS

	Item 1	Item 2	Item 3	Item 4
<b>TOTAL VECES NO</b>	14	37	29	12
<b>TOTAL VECES SI</b>	36	13	21	38

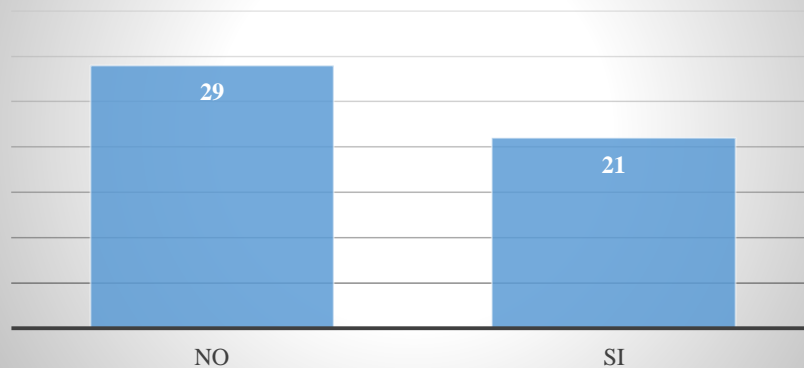
### 1.2. RESULTADOS EXPRESADOS EN GRAFICOS DE BARRAS



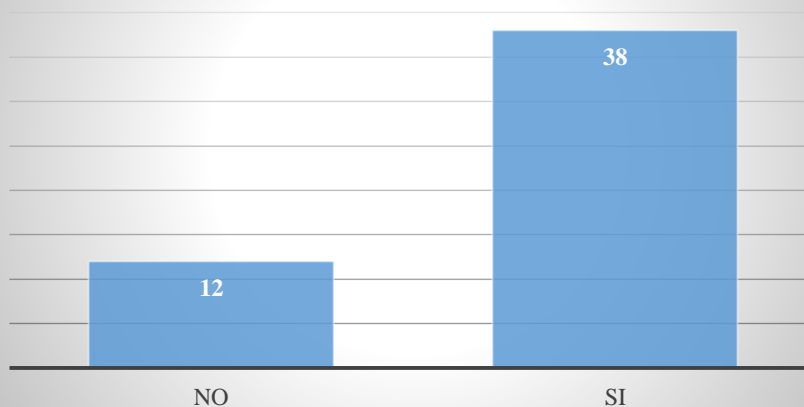
**Item 2: ¿Toma siempre los fármacos a la hora indicada?**



**Item 3: ¿Alguna vez deja de tomar los fármacos si se siente mal?**



**Item 4: ¿Olvidó tomar la medicación durante el fin de semana?**





## 2. RESULTADO DEL ÍTEM 5

Estos son los resultados para el ítem 5: En la última semana, ¿cuántas veces no tomó alguna dosis?

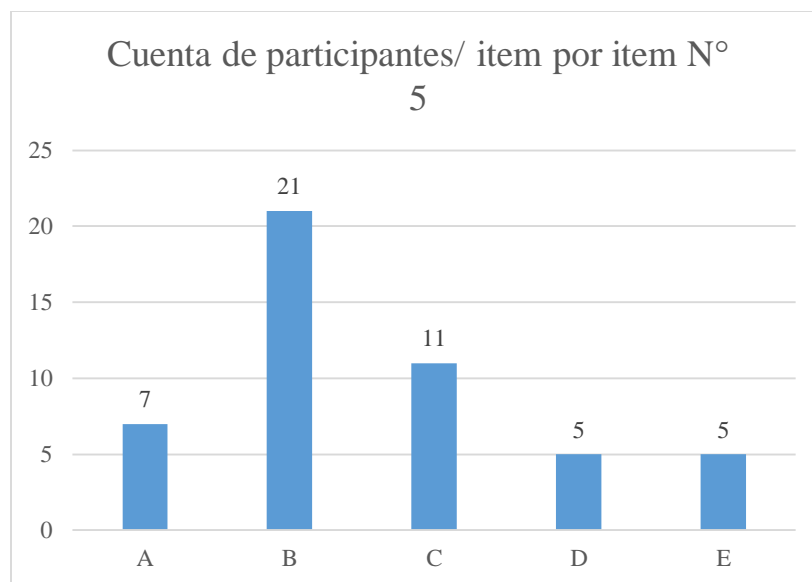
A	B	C	D	E
NINGUNA	1 a 2	3 a 5	6 a 10	más de 10

participantes/ ítem	ítem N° 5
Participante N°1	B
Participante N°2	B
Participante N°3	B
Participante N°4	C
Participante N°5	B
Participante N°6	B
Participante N°7	B
Participante N°8	C
Participante N°9	C
Participante N°10	B
Participante N°11	B
Participante N°12	B
Participante N°13	B
Participante N°14	B
Participante N°15	C
Participante N°16	B
Participante N°17	B
Participante N°18	B
Participante N°19	C
Participante N°20	D
Participante N°21	D
Participante N°22	B
Participante N°23	B
Participante N°24	B
Participante N°25	B
Participante N°26	E
Participante N°27	D
Participante N°28	A
Participante N°29	A
Participante N°30	C
Participante N°31	C
Participante N°32	D
Participante N°33	E

Participante N°34	A
Participante N°35	B
Participante N°36	B
Participante N°37	C
Participante N°38	A
Participante N°39	E
Participante N°40	E
Participante N°41	D
Participante N°42	C
Participante N°43	E
Participante N°44	A
Participante N°45	A
Participante N°46	C
Participante N°47	B
Participante N°48	B
Participante N°49	C
Participante N°50	A

### 2.1.Total de resultados

50	N°	%
A	7	14%
B	21	42%
C	11	22%
D	5	10%
E	5	10%



### 3. RESULTADO DEL ÍTEM 6

Estos son los resultados para el ítem 6: Desde la última visita, ¿cuántos días completos no tomó la medicación?

participantes/item	ítem N°6
Participante N° 1	3
Participante N°2	5
Participante N° 3	3
Participante N° 4	3
Participante N° 5	3
Participante N° 6	3
Participante N° 7	5
Participante N° 8	4
Participante N° 9	2
Participante N° 10	6
Participante N° 11	5
Participante N° 12	3
Participante N° 13	5
Participante N° 14	4
Participante N° 15	3
Participante N° 16	4
Participante N° 17	3
Participante N° 18	2
Participante N° 19	3
Participante N° 20	5
Participante N° 21	3
Participante N° 22	3
Participante N° 23	2
Participante N° 24	6
Participante N° 25	4
Participante N° 26	2
Participante N° 27	2
Participante N° 28	4
Participante N° 29	3
Participante N° 30	6
Participante N° 31	4
Participante N° 32	1
Participante N° 33	2
Participante N° 34	3
Participante N° 35	5
Participante N° 36	4

Participante N° 37	5
Participante N° 38	5
Participante N° 39	3
Participante N° 40	4
Participante N° 41	2
Participante N° 42	3
Participante N° 43	3
Participante N° 44	4
Participante N° 45	2
Participante N° 46	3
Participante N° 47	4
Participante N° 48	3
Participante N° 49	4
Participante N° 50	4

### 3.1.RESULTADO

Desde la última visita, ¿cuántos días completos no tomó la medicación?	
Promedio	3,54

## ANEXO N° 03: ANALISIS DE RIESGOS

### 1. Datos generales

- **Tesista** : Brandon Jesus Perez Guevara
- **Fecha inicial de la tesis** : 19 de marzo de 2018
- **Fecha final de la tesis** : 10 de diciembre de 2018

### 2. Alcance del proyecto

Se desarrolló una aplicación móvil Android para apoyar a la adherencia al tratamiento médico en pacientes adultos mayores, con la finalidad de aumentar los niveles en que los pacientes cumplen al pie de la letra las prescripciones dadas por su médico de cabecera (toma de medicamentos) haciendo uso de notificaciones push-up.

Tiene como características:

- El uso de la tecnología por parte de los adultos mayores.
- Familiarización de los adultos mayores con los smartphones.
- Dar un seguimiento exacto al tratamiento del paciente.
- Obtener datos del progreso del paciente.
- Promover el uso del desarrollo de aplicaciones móviles.

Lo que permite mejorar e incrementar los niveles en que el paciente se adhiere a su tratamiento y que permitirá promover la inclusión tecnológica para esta población ya que son los que menos lo usan.

### 3. Interesados (Stakeholders)

Durante el desarrollo de la presente tesis se ha identificado a los siguientes interesados:

- **Internos**

TABLA 4

INTERESADOS INTERNOS

Interesado	Participación
Investigador	Se ha encargado del desarrollo de la tesis
Asesor	Se ha encargado de guiar el trabajo del investigador
Interesado N	Describir la participación que ha tenido el interesado N en la tesis.

- **Externos**

TABLA 5

INTERESADOS EXTERNOS

Interesado	Participación
Asistente médico	Usuario principal de la aplicación.
Adulto Mayor	Soporte para definir los requerimientos principales del sistema de información
Médico de cabecera	Soporte para definir los reportes que resultarán del sistema de información

#### 4. Beneficios

Los beneficios que se van a obtener con el producto que se ha desarrollado son:

- Mejorar el nivel de cumplimiento de los tratamientos médicos de los pacientes.
- Aumentar el nivel de asistencia a citas médicas
- Ayudar al paciente a llevar un control de su tratamiento y evolución a través del tiempo.
- Incrementar el nivel de satisfacción de pacientes con respecto a la aplicación.

#### 5. Etapas de desarrollo

Para el desarrollo del producto de la presente tesis se ha realizado considerando las etapas de la Metodología del Proceso Unificado Racional o, en sus siglas en inglés, RUP (Rational Unified Process) que consta de las siguientes etapas:

- **Etapas de desarrollo**

- **Actividades realizadas**

- Se realizarán un test a los pacientes para determinar el nivel inicial que tienen de adherencia al tratamiento. Se busca recolectar información importante que pueda orientar las necesidades de los pacientes (son los clientes de la aplicación)
- Se construirá el primer bosquejo de vista de arquitectura del software y se analizará para evaluar posibles cambios o mejoras.
- Se determinará los casos de uso que se desarrollarán según la metodología RUP, luego se documentarán.

- Se desarrollará el plan de pruebas para ir evaluando la aplicación móvil a lo largo del tiempo y conforme crezca en funcionalidad.
  - **Riesgos**
    - Los participantes pueden llenar incorrectamente los test.
    - Descordinación en los horarios para realizar los test.
- **Etapa 2: Elaboración**
  - **Actividades realizadas**
    - Se definirá la versión final de la arquitectura de la aplicación móvil.
    - Se diseñará el bosquejo de la aplicación que es la solución al problema planteado y se analizará para evaluar posibles cambios o mejoras.
    - Se determinará qué tan viable es el desarrollo de esta aplicación.
    - Se construirán el plan de acción para la construcción y la transición.
  - **Riesgos**
    - Los requerimientos no se definieron correctamente.
    - Se añaden requerimientos a destiempo.
    - Se encuentran dificultades en la viabilidad del proyecto.
    - Pérdida de tiempo en la documentación RUP.
- **Etapa 3: Construcción**
  - **Actividades realizadas**
    - Una vez recolectada la información y procesada, se definirá la versión final del conjunto de requisitos que demanda el paciente.
    - Se dará inicio a la programación de la base de datos, de la web Service y de la app móvil.
    - Se organizan los tiempos para cada una de las actividades.
  - **Riesgos**
    - Pérdida de tiempo en arreglos de la base datos y programación.
    - Cambios imprevistos en la aplicación.
    - Falta de conocimientos en la construcción de la aplicación.

- **Etapa 4: Transición**
  - **Actividades realizadas**
    - Se realizarán las pruebas necesarias y los test de funcionalidad para la aplicación móvil a fin de que esté listo para usarse por el paciente.
    - Se brindará capacitación al paciente mismo o a su asistente para usar correctamente la aplicación.
    - El paciente y asistente también evaluarán la efectividad de esta aplicación, serán nuestros mejores testadores ya que lo usarán en el día a día.
  - **Riesgos**
    - El paciente se resista al uso de la aplicación.
    - El asistente no entienda el correcto uso de la aplicación o no le de el uso correcto.
    - El uso de un smartphone para controlar las tomas de medicamentos puede resultar incómoda o muy estricta.



## ANEXO N° 06: ACEPTACION POR EXPERTOS

### EVALUACION POR UN EXPERTO

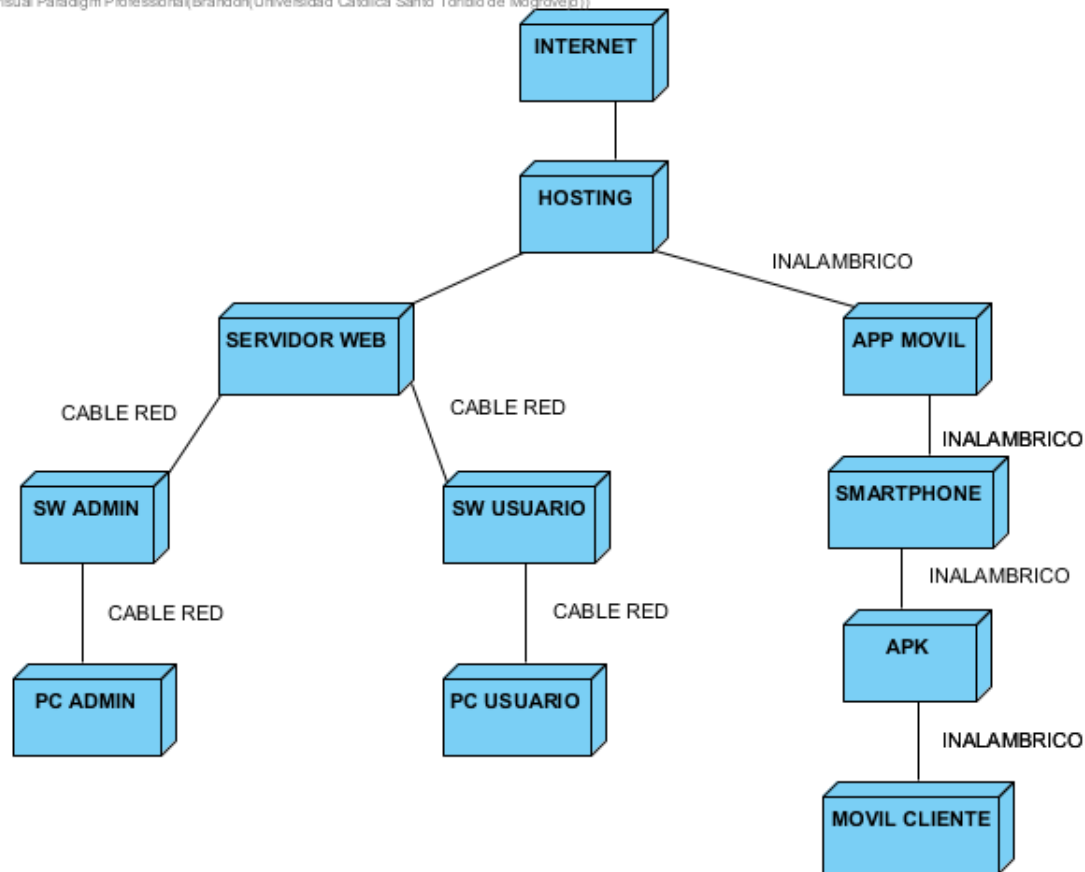
Yo Blasi Dávila Fernando Ruben, identificado con DNI 46201029, declaro haber realizado la revisión y aprobación de los apartados de Resumen, Introducción, Marco teórico, Metodología, Anexos y Bibliografía del proyecto de tesis titulado "Diseño e implementación de una aplicación móvil Android orientada al adulto mayor para apoyar la adherencia al tratamiento médico" con autoría de Brandon Jesús Pérez Guevara, estudiante de ingeniería de sistemas de la Universidad Santo Toribio de Mogrovejo, para los fines que crea conveniente.

   
MINISTERIO DE SALUD  
Fernando Ruben Blasi Dávila  
MEDICO GENERAL  
C.M.P. 144324  
FIRMA

Chiclayo, 7 de diciembre de 2018

## ANEXO N° 07: ESTRUCTURA TECNOLÓGICA

Visual Paradigm Professional (Brandon (Universidad Católica Santo Toribio de Mogrovejo))



### Infraestructura tecnológica detallada

- **Internet:** indispensable para el desarrollo de la investigación puesto que a través de él se compartirán datos de vital importancia.
- **Hosting:** indispensable para el almacenamiento de todos los recursos y paquetes que requiera la aplicación móvil y web para su correcta ejecución en los servidores. También para mantener guardadas copias de seguridad y reportes importantes generados por la aplicación. Para la ejecución de la investigación se utilizó un hosting gratuito, pero es recomendable alquilar un hosting de suscripción pagada para obtener mejor soporte y condiciones de trabajo.
- **Servidor Web:** indispensable para ejecutar la versión web de la aplicación desde donde el administrador puede gestionar todos los apartados de la misma. Es importante recalcar que la aplicación web es solo del administrador.
- **PC Admin:** necesaria para ejecutar y gestionar la aplicación desde el perfil del administrador. Sus características de hardware mínimas son:
  - Procesador: Intel Core i5
  - Memoria RAM: 4GB
  - Sistema Operativo: Windows 10
  - Almacenamiento: 500GB
- **PC Usuario:** necesaria para el uso de la aplicación web desde el perfil del asistente médico.

- Procesador: Intel Core i5
- Memoria RAM: 4GB
- Sistema Operativo: Windows 10
- Almacenamiento: 500GB
- **App móvil:** es el conjunto de componentes que procesados en conjunto permiten la ejecución de la aplicación móvil que será empleada para la investigación.
- **APK:** es el instalador que resulta del compilado y depuración del IDE Android Studio para ser posteriormente instalado en el móvil cliente.
- **Movil Cliente:** indispensable para el asistente médico y el paciente puesto que es la herramienta fundamental de trabajo donde se ingresarán los datos de los tratamientos y citas médicas del paciente. Es el recurso más importante.

## ANEXO N° 08: MANUAL DE USUARIO

### 6. Datos generales

- **Tesista** : Brandon Jesus Perez Guevara
- **Fecha inicial de la tesis** : 19 de marzo de 2018
- **Fecha final de la tesis** : 10 de diciembre de 2018

### 7. PROPOSITO DEL MANUAL

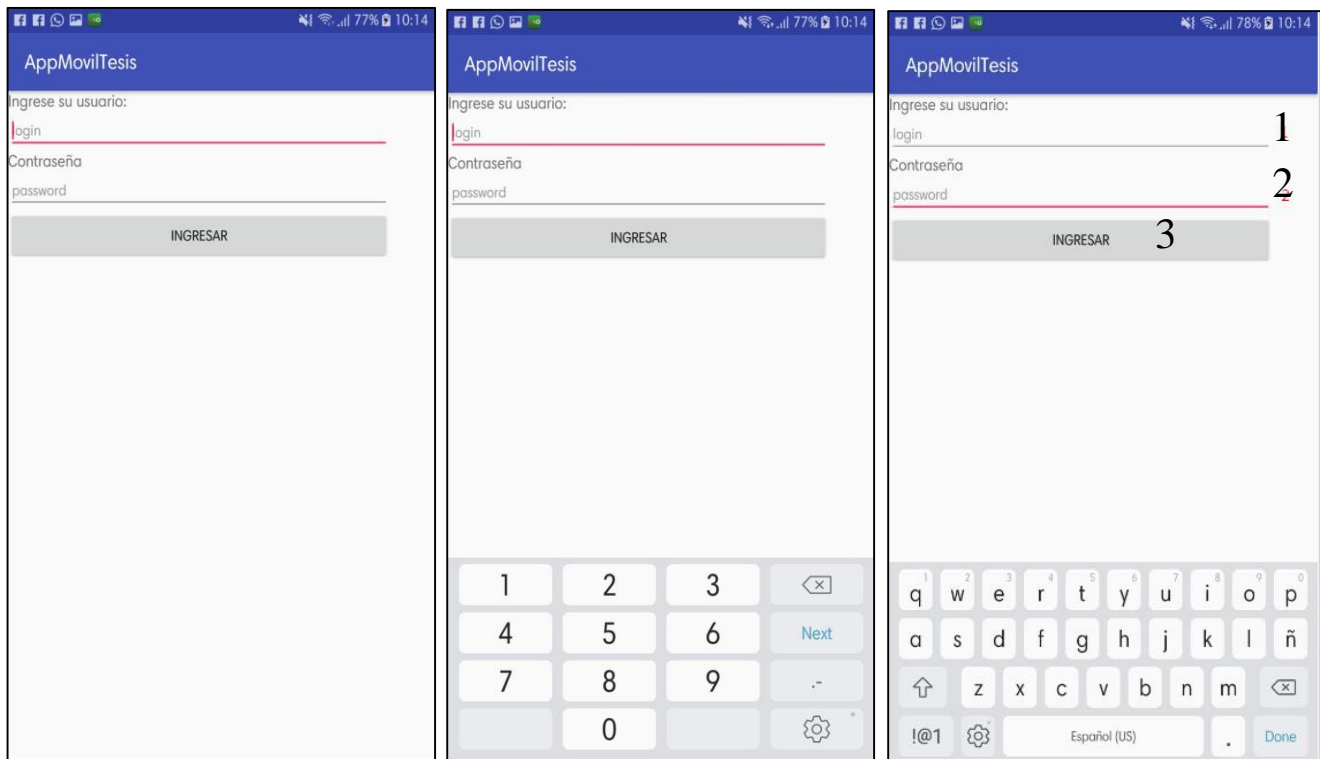
Los manuales de usuario son necesarios para entender qué, cómo y por qué de las interfaces en una aplicación móvil, desarrollando la destreza de navegar en ella con fluidez y sabiendo en cada momento qué se está operando.

Este manual ha sido construido con el objeto de que resulte sencillo de entender para el usuario lector. Contiene el progreso de construcción de todas las pantallas de la aplicación móvil hasta esta etapa de manera concisa y directa.

### 8. ESTRUCTURA DEL MANUAL

#### a. INICIO DE SESIÓN

- **Objetivo de la pantalla de inicio de sesión:** constituye una manera de seguridad de información, teniendo que identificarse mediante un

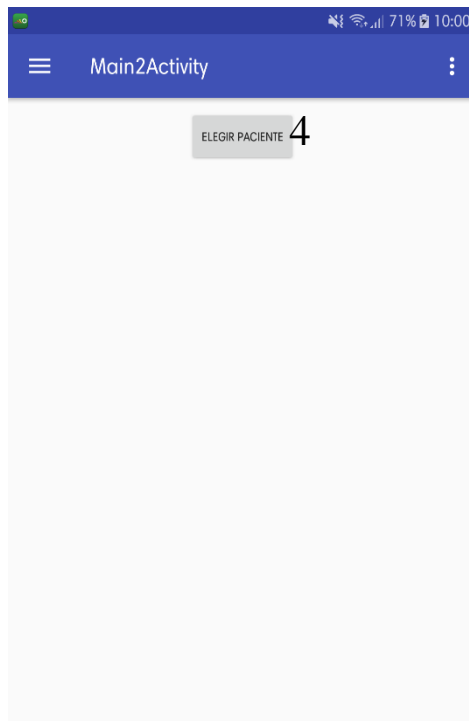


usuario y una contraseña para tener acceso a la información del usuario.

- 1: El usuario debe ingresar su LOGIN que es su número de DNI (8 dígitos).**
- 2: El usuario debe ingresar su CONTRASEÑA que es una cadena que combina letras, números y símbolos y puede ser de hasta 32 caracteres.**
- 3: Este botón comprueba la identidad del usuario que intenta ingresar a la aplicación, si todo está correcto, le da entrada a la pantalla principal. Si no, niega el acceso.**

#### **b. PANTALLA PRINCIPAL**

- **Objetivo de la pantalla principal:** Constituye la pantalla principal de la aplicación desde donde se puede seleccionar un paciente para trabajar con él y sus controles de tratamientos y citas médicas hasta que el tiempo de trabajo expire.

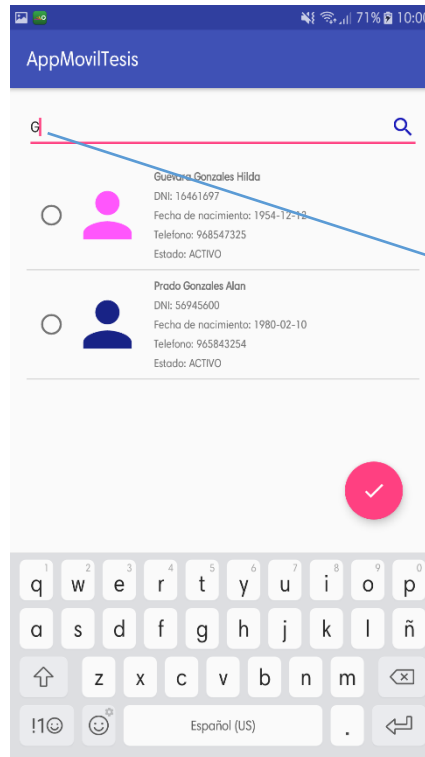


- 4: Este botón apertura la selección del paciente donde se listan todos los pacientes asociados al asistente y este puede elegir entre todos ellos para controlar sus tratamientos médicos.**

Al seleccionar el botón “ELEGIR PACIENTE” se apertura la siguiente interfaz.

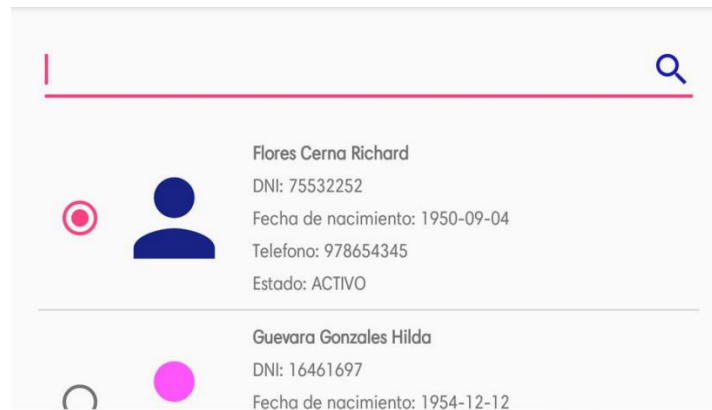


**5:** Este campo se utiliza para seleccionar de una manera más rápida a un paciente y evitar buscarlo en toda la lista. Solo basta con digitar las letras del nombre del paciente y se listarán los pacientes cuyos nombres coinciden con lo ingresado. Tal y como se muestra en la siguiente imagen.



Se ha ingresado la letra “G” y la aplicación ha filtrado todos los nombres de los pacientes donde haya una “G”.

**6:** Este control sirve para seleccionar al paciente con el que se quiere trabajar. Al seleccionar al paciente, dicho control debe cambiar de color de la siguiente manera.



**7:** Este control significa que, después de haber seleccionado el paciente con el que se va a trabajar, se confirma esta selección y el paciente se carga junto con su información listo para ser controlado. Al seleccionar al paciente se debe presentar la interfaz de la siguiente manera.



**8:** Este botón nos lleva a la Gestión de las Citas donde se puede ver información de todas las citas médicas relacionadas con el paciente y los tratamientos de estas.

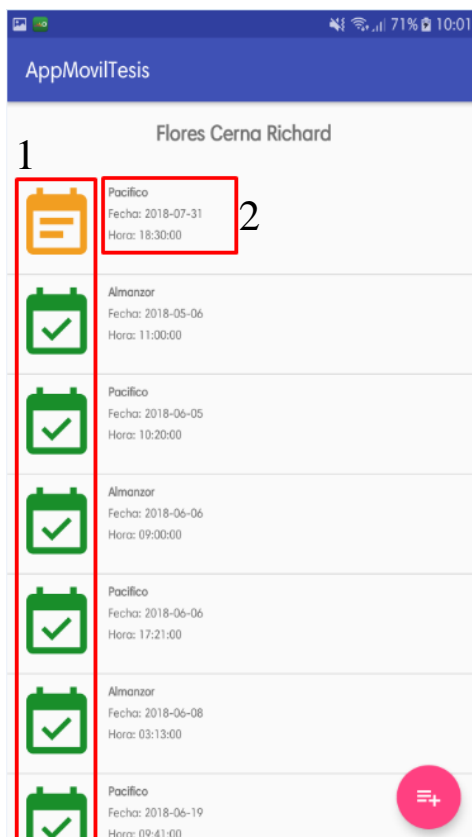
**9:** Este botón nos lleva a ver los tratamientos en general relacionados con el paciente sin estar clasificados por cita médica.

### c. GESTIÓN DE CITAS MEDICAS

- **Objetivo de la Gestión de Citas:** En este conjunto de interfaces el usuario puede gestionar (agregar, editar, cancelar) las citas médicas del paciente. Es importante para tener un registro de las citas médicas pendientes, las que ya se realizaron y las que se cancelaron.

#### i. LISTADO Y OPCIONES





En esta interfaz se muestran las citas asociadas al paciente ordenadas por el “Estado de la Cita” que puede ser “pendiente”, “cancelado” o “realizado”. Las citas se muestran ordenadas por estado, empezando por las pendientes, realizadas y terminando con las canceladas.

**1:** Aquí se encuentra el detalle más resaltante de una cita médica como el hospital donde se llevará a cabo la cita, la fecha y la hora.

**2:** Indica el estado de la cita médica asignada al paciente y se diferencia por medio de colores:

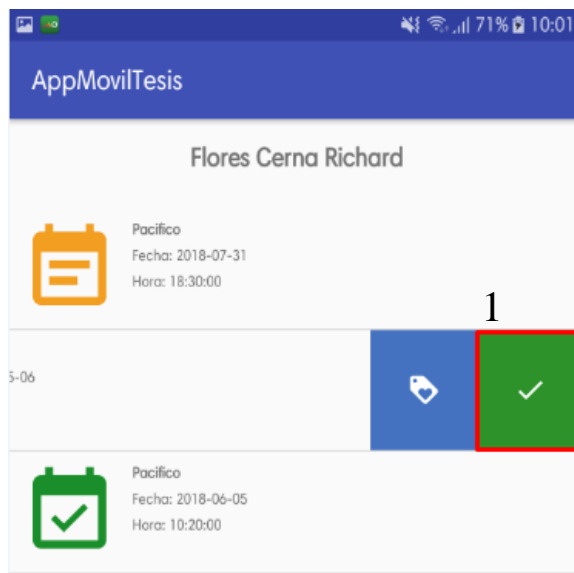
- Verde: Cita realizada. Indica que pueden registrarse tratamientos o gestionar los existentes. Esta cita no se puede cancelar.
- Amarillo: Cita pendiente. Esta cita se puede cancelar.
- Rojo: Cita cancelada. No se puede ejercer ninguna acción sobre una cita cancelada.

## ii. CONTROL DE LOS ESTADOS DE UNA CITA

Existen excepciones cuando se manejan los estados de una cita.

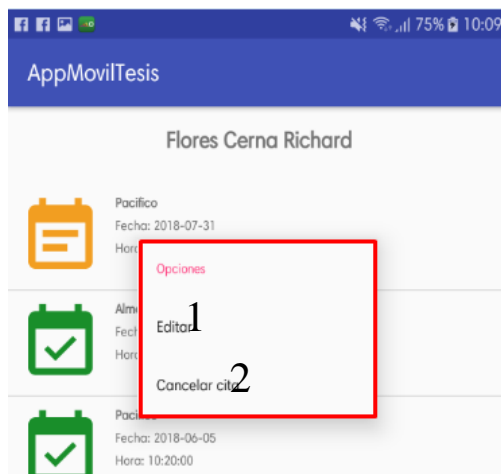
- Excepciones sobre una cita realizada (color verde):
  - No se puede realizar nuevamente.
  - No se puede editar.

- No se puede cancelar.
- Se pueden ver/agregar tratamientos médicos asociados con esa cita.



**1:** Este botón sirve para indicar que una cita pendiente se ha realizado.

- Excepciones sobre una cita pendiente (color amarillo):
  - No puede tener tratamientos asociados.
  - Solo puede ser realizada o cancelada.
  - Puede ser editada respecto al lugar, fecha y hora de cita médica.

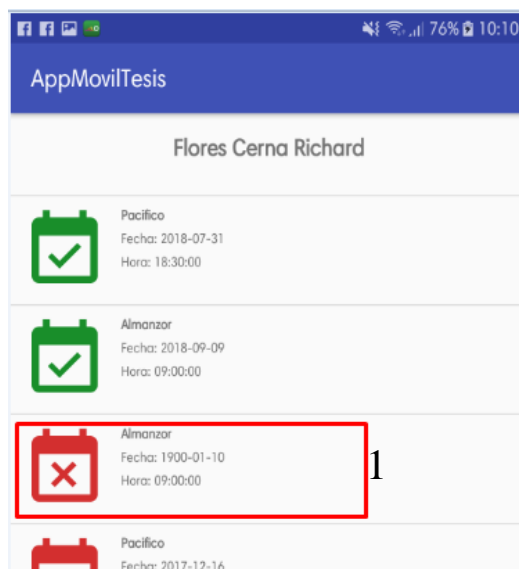


**1:** Este botón sirve para editar una cita médica. Explicado en la sección III. Editar una cita médica.

**2:** Este botón sirve para cancelar una cita médica.

- Excepciones sobre una cita cancelada: color rojo.
  - No puede ser realizada.

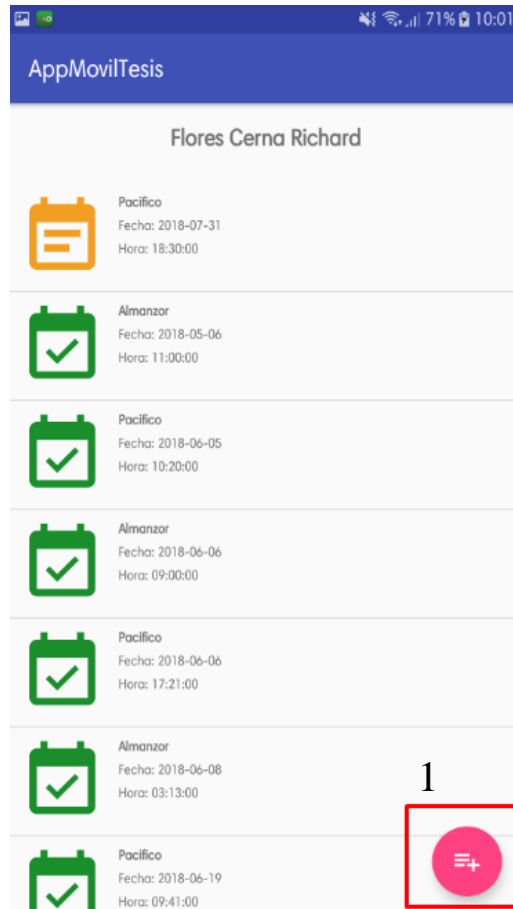
- No puede ser cancelada nuevamente.
- No se pueden ver sus tratamientos porque fue cancelada.



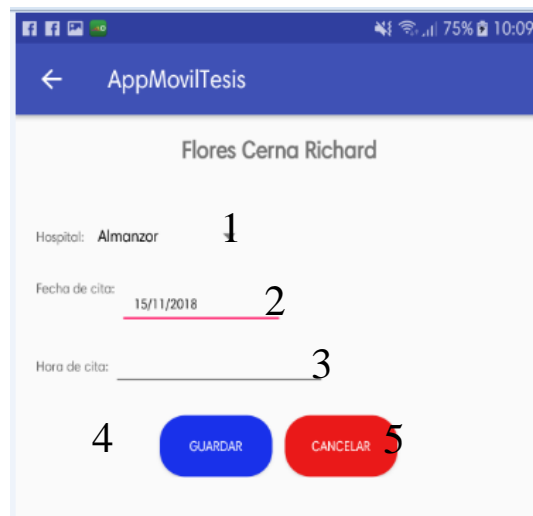
**1:** indicador de una cita cancelada.

### iii. AGREGAR UNA CITA MEDICA

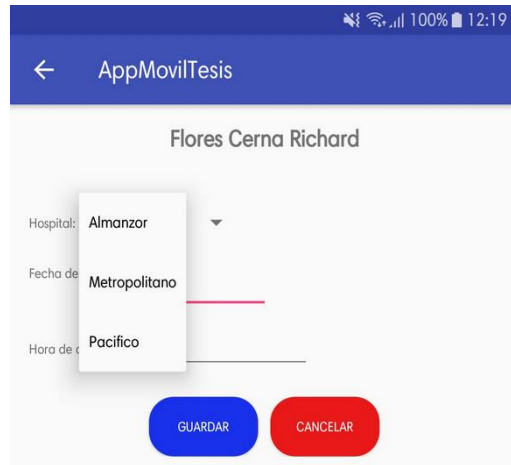
- **Objetivo de la interfaz AGREGAR UNA CITA MEDICA:**  
En esta interfaz el usuario puede agendar una nueva cita médica para su paciente, ingresando información básica de la cita como lugar (hospital), fecha y hora de la cita médica. Es información se guardará y se podrá controlar por el asistente.



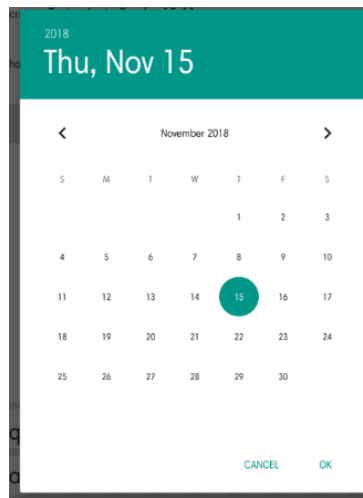
1: Este botón indica que el usuario desea registrar una nueva cita por lo que se apertura una nueva interfaz donde se debe ingresar los datos necesarios para el registro de una nueva cita médica.



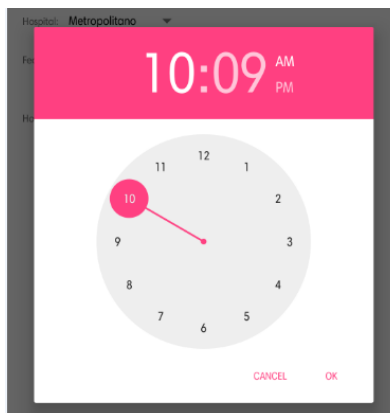
1: Aquí se listan los posibles hospitales donde se puede agendar la cita, el usuario debe elegir entre ellos seleccionando el adecuado.



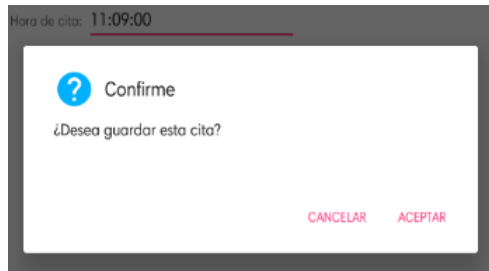
**2:** Aquí puedes elegir la fecha de la cita médica de manera intuitiva calendario de fácil manejo y atractivo diseño.



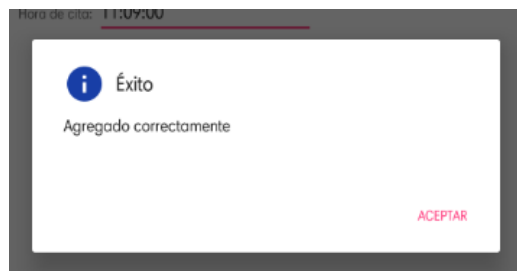
**3:** Aquí puedes elegir la hora de la cita médica de manera intuitiva en un reloj de fácil manejo y atractivo diseño.



**4:** Este botón indica que quieres guardar el registro de la cita médica.



Si el usuario confirma el registro de la cita, aparecerá un mensaje de confirmación como el siguiente:

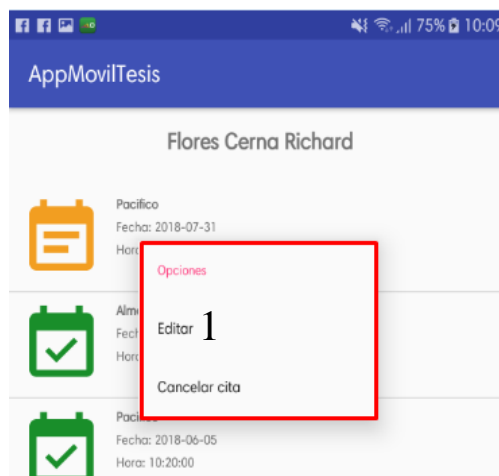


5: Este botón indica que quieres cancelar el registro de la cita médica

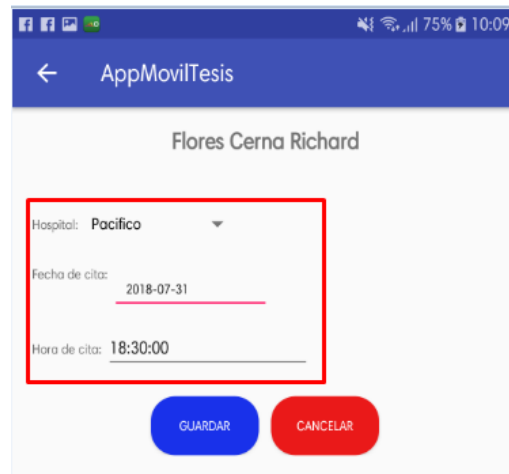
#### iv. EDITAR UNA CITA MEDICA

- **Objetivo de la interfaz EDITAR UNA CITA MEDICA:**  
Esta interfaz tiene por objetivo editar la información de una cita médica pendiente después de ser ingresada en la aplicación. De esta manera se puede corregir algún error que se tuvo al ingresar los datos.

**Para proceder a editar una cita médica, se accede desde el listado de citas, seleccionando una cita pendiente y luego seleccionando editar.**



1: Este es el botón que apertura la interfaz para editar una cita médica.

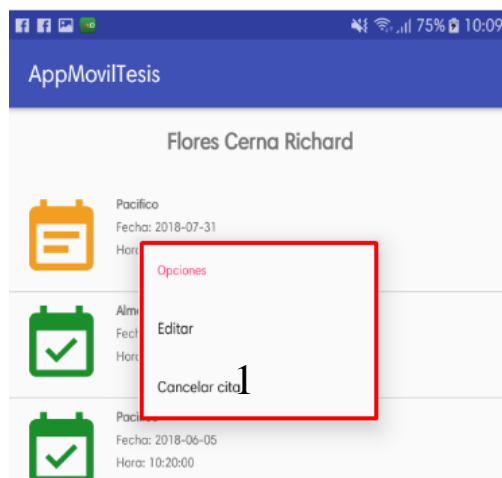


Como se puede ver, la interfaz de editar una cita médica carga los datos de la cita médica y habilita aquellos que pueden ser editados como el hospital, la fecha y la hora de la cita médica. De igual manera a agregar una cita médica, la fecha se muestra en un calendario intuitivo y visual, de igual forma la hora.

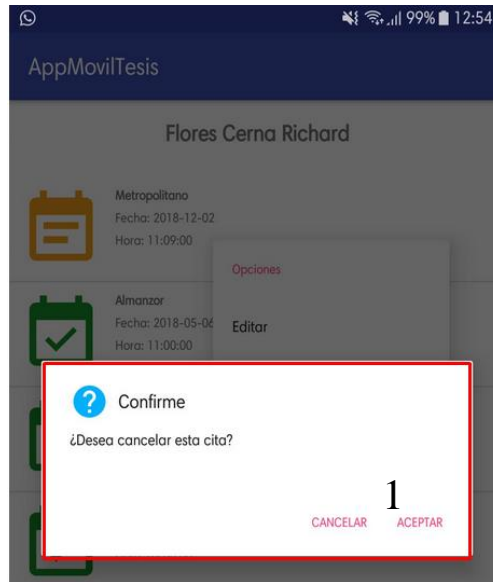
#### v. CANCELAR UNA CITA MEDICA

- **Objetivo de la interfaz CANCELAR UNA CITA MEDICA:** Esta interfaz tiene como objetivo el poder dar de baja a una cita médica y excluirla de las citas médicas pendientes que posee el paciente.

Para cancelar una cita médica, de acuerdo a las excepciones mencionadas anteriormente, se selecciona una cita pendiente. De la siguiente manera:



**1:** Este botón cambia el estado de pendiente a cancelado de la cita médica y por consiguiente cambia el color de la cita a rojo y la coloca al final del listado con las demás citas canceladas.



1: Este botón convierte la cita pendiente en cancelada.

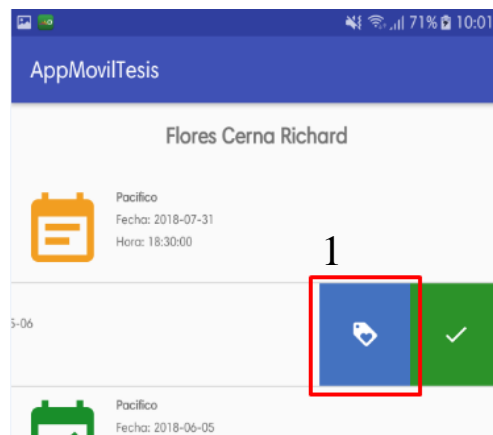
#### d. GESTIÓN DE TRATAMIENTOS

- **Objetivo de la Gestión de Tratamientos:** En este conjunto de interfaces el usuario puede gestionar y controlar los tratamientos médicos del paciente. Es importante para tener un registro de los tratamientos médicos del paciente y el detalle de estos para poder obtener estadísticas acerca del nivel de cumplimiento de estos tratamientos médicos. De esta manera se puede estimar el nivel de cumplimiento del paciente.

#### i. LISTADO Y OPCIONES

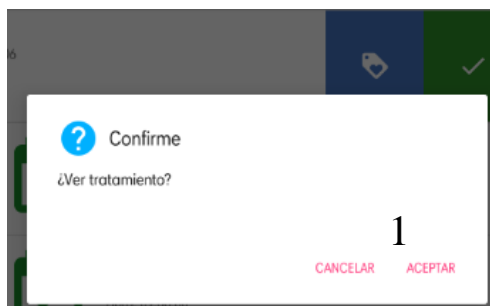
Para tener acceso a la lista de tratamientos se puede optar por dos opciones:

- A través de la interfaz de citas médicas.





**1:** Con este botón se ingresa al listado de tratamientos desde la interfaz de las citas, mostrando los tratamientos que fueron registrados en esa cita médica.



**1:** Con este botón “ACEPTAR” se ingresa a ver los tratamientos asociados a esa cita.

- Ingresando a los tratamientos médicos de manera directa.



**1:** Con este botón “VER TRATAMIENTOS” se ingresa a ver los tratamientos en general de todas las citas del paciente.

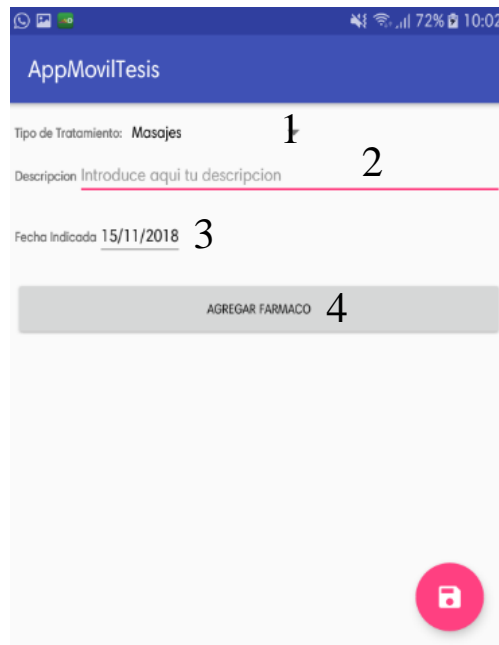
A través de ambos caminos para llegar a los tratamientos, se llega a la misma información



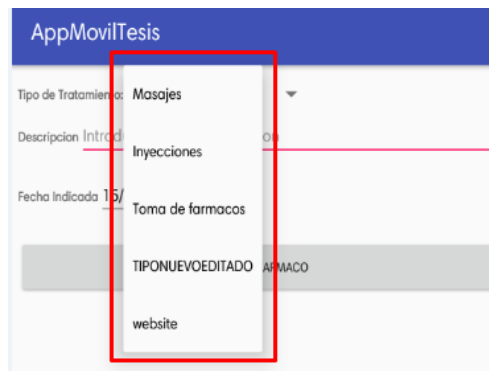
**1:** Aquí se presenta la información de cada uno de los tratamientos asociados al paciente. La información que se muestra es el nombre del tratamiento y la fecha de inicio del tratamiento.

## ii. AGREGAR

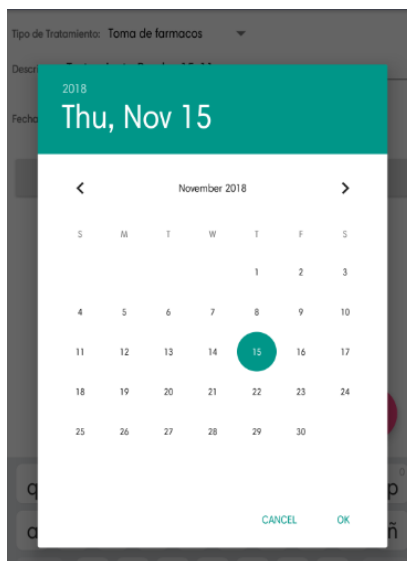
- **Objetivo del formulario de Agregar Tratamiento médico:**  
Este formulario es el mas importante de todos porque es aquí desde donde nace el control al tratamiento médico del paciente por medio del asistente médico. El objetivo de este formulario es guardar los datos del tratamiento médico prescrito por el médico del paciente para controlarlo posteriormente.



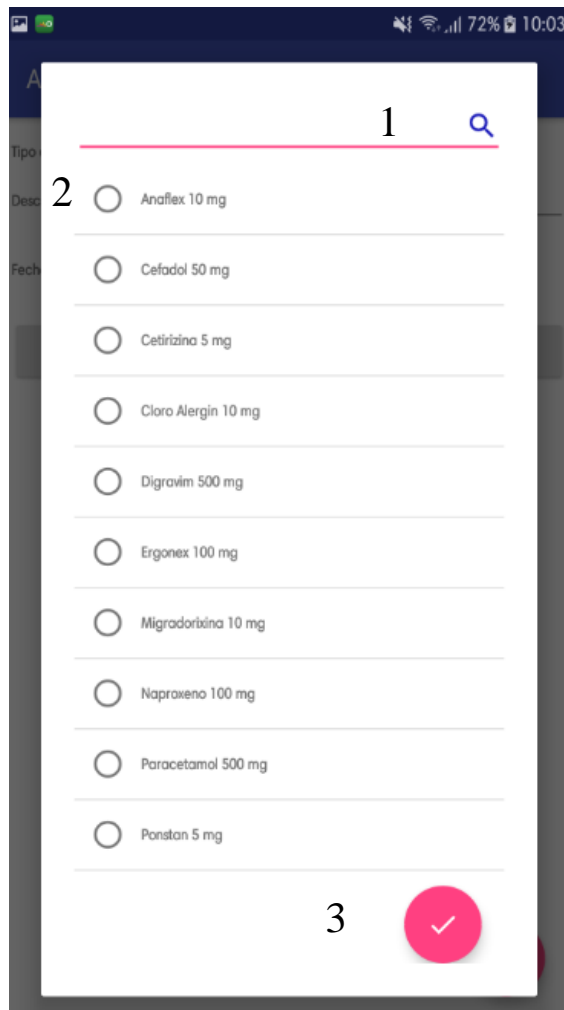
**1:** En este control se selecciona el tipo de tratamiento a registrarle al paciente, entre ellos tenemos toma de medicamentos (el más frecuente), masajes, etc. entre todos los tipos de tratamiento que se contemple. El despliegue de los datos se verá de la siguiente manera.



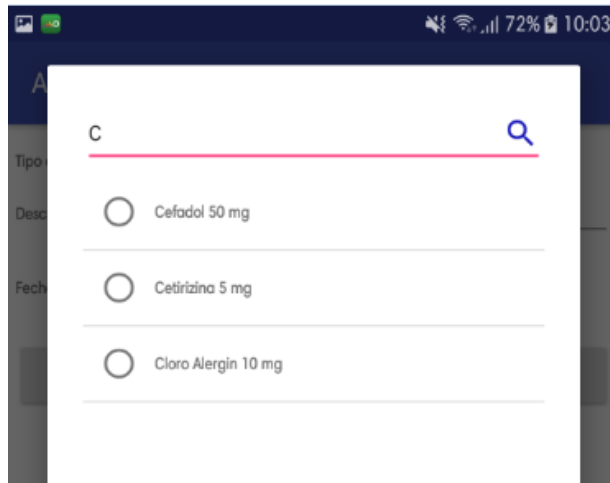
**2:** en este campo se escribe alguna descripción concisa del tratamiento que se va a registrar.  
**3:** al presionar esta fecha se apertura un calendario donde se manera fácil e intuitiva se puede seleccionar la fecha que indica el doctor para iniciar el tratamiento.



**4:** este botón apertura una sección donde se puede buscar el fármaco deseado entre una lista de fármacos posibles. Además, se puede agilizar la búsqueda ingresando letras del medicamento deseado y cada medicamento que coincida irá apareciendo en la lista.

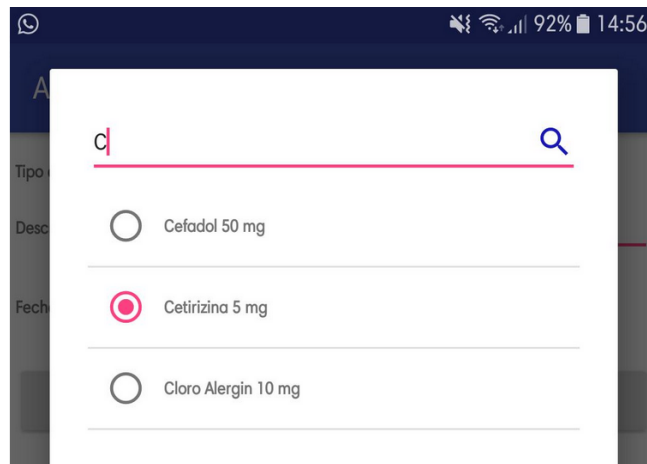


**1:** en esta parte se puede filtrar los medicamentos de acuerdo a las letras ingresadas en dicho campo. El filtrado se puede ver de la siguiente manera:



Como se aprecia en la imagen, se ingreso la letra “C” y se filtraron los medicamentos registrados que empiezan por la letra C, en este caso Cefadol, Cetirizina y Cloro Alergin.

**2:** Una vez ubicado el fármaco deseado se debe seleccionar para luego confirmar la elección.  
Se selecciona de la siguiente manera:



**3:** este botón significa que confirmamos la elección del fármaco por lo que retornamos a la interfaz anterior para llenar más datos del tratamiento y añadir otros fármacos de ser necesario.

The screenshot shows the 'AppMovilTesis' mobile application interface. At the top, the status bar displays signal strength, Wi-Fi, cellular data, 72% battery, and the time 10:03. The app title 'AppMovilTesis' is in a blue header. Below it, the 'Tipo de Tratamiento' is set to 'Toma de farmacos'. The 'Descripcion' field contains 'Tratamiento Prueba 15-11'. The 'Fecha Indicada' is '15/11/2018'. A grey button labeled 'AGREGAR FARMACO' is present. The medication 'Cefadol 50 mg' is selected. Below this, there are four numbered input fields: 1. 'Cantidad' with an empty text input. 2. 'Frecuencia' with a dropdown menu showing 'Cada 1' and a unit dropdown showing 'hora'. 3. 'Periodo' with an empty text input and a unit dropdown showing 'hora'. 4. 'Fecha Indicada' with a text input showing '15/11/2018'. At the bottom right, there is a pink circular button with a white save icon and the number '5' next to it.

**1:** aquí se indicará la cantidad de pastillas del fármaco se van a tomar.

**2:** en esta parte se indicará la frecuencia con la que se va a tomar la cantidad de pastillas del fármaco asignado. **Ejemplo:** cada 1 hora, cada 8 horas, cada 2 horas, cada 1 día.

Descripción Tratamiento Prueba 15-11

Fecha Indicada 15/11/2018

Cada 14

Cada 2

Cada 3

Cada 30

Cada 4

Cada 6

Cada 7

Cada 8

Cefadol 50 mg

Cantidad 1

Frecuencia

Periodo 2

Fecha Indicada 2018

Tipo de Tratamiento: Toma de farmacos

Descripción Tratamiento Prueba 15-11

Fecha Indicada 15/11/2018

AGREGAR FARMACO

Cefadol 50 mg

Cantidad 1

Frecuencia Cada 2

hora

día

semana

mes

año

pruebawebEDITADA

**3:** en esta parte se indicará el periodo durante el cual se va a tomar el fármaco. Pueden ser horas, días, semanas, meses, etc.

Descripción Tratamiento Prueba 15-11

Fecha Indicada 15/11/2018

AGREGAR FARMACO

Cefadol 50 mg

Cantidad 1

Frecuencia Cada 2

Periodo 2

hora

día

semana

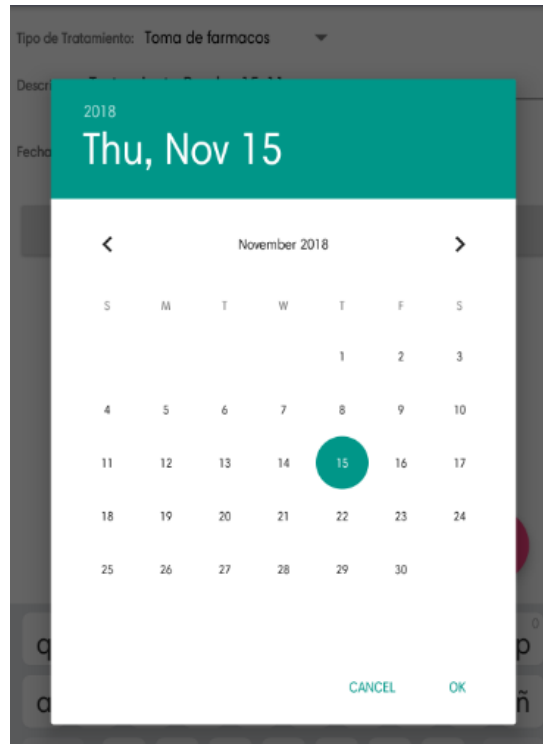
mes

año

pruebawebEDITADA

**4:** En este campo se indicará la fecha indicada por el médico para iniciar la toma de dicho fármaco. Se mostrará nuevamente un calendario fácil de usar e intuitivo.





**5:** al seleccionar este botón el asistente es consciente que se han ingresado todos los datos del tratamiento y está listo para ser registrado.

## ANEXO N° 09: ESTRUCTURA RUP

### ESTRUCTURA RUP

#### 1. CAPITULO I: PLANIFICACIÓN DEL PROYECTO

##### 1.1. PARTICIPANTES DEL PROYECTO

###### A. DOCENTE COORDINADOR

Ing. Gregorio León

###### B. EQUIPO DEL PROYECTO

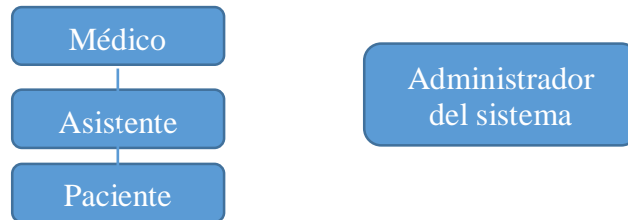
Brandon J. Pérez Guevara

###### C. CONTACTO

bjesusperez@gmail.com

##### 1.2. DESCRIPCIÓN DEL ÁREA

###### A. ORGANIGRAMA ESTRUCTURAL



###### B. FUNCIONES DEL RESPONSABLE

El asistente del paciente debe cumplir las funciones de:

- Controlar las tomas de los medicamentos del paciente.
- Asistirlo en sus necesidades básicas.
- Ir con el paciente a las citas médicas.
- Contar con los fármacos prescritos por el doctor de cabecera.
- Procurar en lo posible que el paciente cumpla en totalidad su tratamiento médico.

###### C. REALIDAD PROBLEMÁTICA

La adherencia al tratamiento fue considerado un problema a nivel mundial desde 2003 por un estudio realizado por la Organización Mundial de la Salud (OMS), donde se obtenía que apenas el 50% de los pacientes llevaban un buen nivel de adherencia al tratamiento. Este término se refiere a el grado en el cual una persona sigue las prescripciones farmacológicas según lo indicado desde tomas de dosis hasta asistencia a citas médicas.

Actualmente existen estudios relacionados con la adherencia al tratamiento como el realizado por Málaga et. al. (2010) dónde analiza el nivel de adherencia en el Hospital General Cayetano Heredia respecto a las personas con HTA (Hipertensión Arterial). Se enfocó en esta población dado que para tratar esta enfermedad se requiere un largo periodo, lo que facilitaría la constante evaluación de la adherencia al tratamiento que desenvuelve el paciente. Los resultados que se obtuvieron fue que apenas el 37.9% de la población completa (103 personas de las cuales el más del 80% eran adultos mayores) lleva una adherencia al tratamiento adecuada. Actualmente más del 30% de la población adulta mayor sufre de hipertensión (INEI, 2015) y el 60% de hipertensos

desconoce su diagnóstico (OMS 2013), lo que convertiría a este tipo de personas en una opción para ser estudiada.

Sin embargo, la hipertensión no es la única enfermedad crónica que afecta a los adultos mayores, por ejemplo, también se ven muchos casos de diabetes. Según el INEI en el 2014 se registró que el 8,2% de la población adulta mayor padece de diabetes. Por grupo de edad, entre quienes tenían de 75 a 79 años, 10,0%, seguida por los de 65 a 69 años (8,8%) y de 70 a 74 años (8,6%). En 2016 el diario La República difundió una declaración de la presidenta de la Asociación de Diabetes del Perú (ADIPER), Angélica Valdivia, quien alertó que en el país existen 400 mil adultos mayores con diabetes.

Estas dos enfermedades no son curables, pero sí controlables lo que demandaría a tener un nivel muy estricto de adherencia al tratamiento médico que sea recibido.

Entonces ¿cómo podríamos apoyar a la adherencia al tratamiento médico?

### **1.3. ANÁLISIS DE SITUACIÓN TECNOLÓGICA**

#### **A. SISTEMAS EXISTENTES**

Existen aplicaciones dedicadas a programar las tomas de las pastillas, pero las interfaces resultan engorrosas y complicadas de llenar para muchas personas. Además, solo se encargan de eso, programar un horario de tomas. Sin embargo, ante esta situación problemática se requiere algunas funcionalidades más que se detallarán en la planificación del sistema propuesto.

Las propuestas actuales son aplicaciones simplemente, lo que se pretende construir es un sistema que integre un panel de control web, una aplicación móvil y una base de datos remota capaz de ser consumida por los dos anteriores.

#### **B. APLICATIVOS UTILIZADOS**

Existen 5 aplicaciones más resaltantes, estas son: RecuerdaMed, Mango Health, Dosecat, MediSafe, Pill Manager. Dedicadas exclusivamente a recordatorios de tomas y no cuentan con portales web desde donde puedas administrar tus tomas. Además la aplicación es personal, un mismo usuario no puede crear horarios de personas diferentes.

#### **C. EQUIPOS (HARDWARE)**

No se cuentan con equipos.

### **1.4. PLANIFICACIÓN DEL SISTEMA PROPUESTO**

#### **A. DESCRIPCIÓN**

Se desarrollará un sistema de información para apoyar a la adherencia al tratamiento médico en adultos mayores con enfermedades de larga duración como hipertensión y diabetes. Este sistema constará de un portal web donde se pueda gestionar, administrar y controlar los datos que se recibe a través de una aplicación móvil que manipulará el usuario por medio de un smartphone con el sistema operativo Android. Además, se contará con una base de datos alojada remotamente a la cual se accederá mediante internet y puede ser consumida tanto por el portal web como por la aplicación móvil.

##### **- Base datos:**

Será desarrollada en Postgres y almacenada en la nube. Estará normalizada y complementada funcionalmente respondiendo a consultas y funciones que requiera el portal web y la aplicación móvil.

##### **- Portal web:**

Será desarrollado en Java, Php, Html y Java Script bajo un framework gratuito que se adaptará para lo que necesita el administrador como gestionar usuarios, verificar como se van guardando los datos, etc.

Además de esto, el médico encargado del paciente puede ver el progreso de su tratamiento, cómo va cumpliendo con sus dosis respecto al tratamiento total. La aplicación web será capaz de emitir reportes llamados gráficos de cumplimiento donde se podrá ver de manera periódica de qué manera se ha ido cumpliendo correctamente con el tratamiento prescrito como también con las citas médicas; ambos reportes de manera separada. De esta manera se puede calcular en qué nivel se cumplió con las tomas de los medicamentos lo que puede dar una idea si el usuario está llevando un buen control y tiene una correcta adherencia.

- **Aplicación móvil:**

Es la parte más importante del sistema pues a través de ella se podrá obtener los resultados esperados para esta investigación. Cada usuario puede ingresar a la aplicación mediante su número de DNI (previamente registrados por el administrador como su población de estudio). Una vez ingresado se pueden añadir pacientes a los que se les puede asignar tratamientos. Los pacientes deben ser registrados únicamente con información relevante para el estudio como su nombre, su género y su edad. Además de esto se pueden registrar y gestionar los aspectos relacionados a la adherencia al tratamiento médico mencionados en el marco teórico. En primer lugar, se podrán gestionar los tratamientos médicos, es decir el registro de un horario de tomas de medicamentos prescrito por el médico. Algunos datos como el nombre del medicamento, el tipo e incluso una foto se pueden registrar. Se denota cada cuánto tiempo se harán las tomas del fármaco y automáticamente la aplicación crea un horario para enviar notificaciones al paciente para realizar su toma respectiva. Las notificaciones deben tener un aspecto y sonido agradables para que sea cómodo interactuar con ella.

En segundo lugar, de la misma manera se podrán gestionar las citas médicas y programarlas para que la aplicación pueda enviar notificaciones periódicamente haciendo acuerdo de dicha cita. Tal y como en la funcionalidad anterior, se puede registrar información acerca del hospital donde se debe de ir, la dirección e incluso mostrarlo en Google Maps mediante la API de Google.

En tercer lugar, se pueden buscar farmacias. Las farmacias serán presentadas como búsqueda en Google Maps de manera interactiva para que el usuario pueda ubicarse rápidamente dónde esté ya sea dentro o fuera de su localidad. Además de esto se recomendará a farmacia más cercana.

Cabe resaltar que, según investigaciones citadas en el marco teórico, hay personas mayores que usan un smartphone; sin embargo, existen muchas otras que no. Por eso, se aclara que la aplicación está más orientada a un paciente y un asistente que a un paciente solo. De esta manera se puede tener un control más claro del paciente ya que se puede evitar que mienta, registre tomas incorrectas o simplemente no haga un correcto uso y no aproveche al máximo la aplicación. Si bien es cierto, existen muchos factores exteriores que puedan interferir de alguna manera con la fiabilidad de la información vertida y recopilada de esta aplicación; sin embargo, para un mejor control y confiabilidad de información, la aplicación

será puesta en manos del asistente que por ética no puede brindar información falsa a la aplicación.

#### B. OBJETIVOS

- Mejorar el nivel de cumplimiento de los tratamientos médicos de los pacientes.
- Aumentar el nivel de asistencia a citas médicas.
- Ayudar al paciente a llevar un control de su tratamiento y evolución a través del tiempo.
- Disminuir el tiempo que demora un paciente en ubicar una farmacia.
- Facilitar al paciente farmacias cercanas en caso de encontrarse en otra localidad.
- Incrementar el nivel de satisfacción de pacientes con respecto a la aplicación.

#### D. PRESUPUESTO

##### Presupuesto del producto acreditable

Tabla1. Presupuesto del producto acreditable

PARTE PRESUP.	DESCRIPCIÓN	CANT.	Unidad de Medida	Precio Unitario (S/.)	Precio Total (S/.)	SUB TOTAL (S/.)	TOTAL (S/.)
2.3	BIENES Y SERVICIOS						6.010,00
2.3.11.1	ALIMENTOS Y BEBIDAS					240,00	
	2.3.11.11 ALIMENTOS Y BEBIDAS PARA CONSUMO HUMANO	30	GLOBAL	8,00	240		
2.3.19.1	MATERIALES Y UTILES DE ENSEÑANZA					250,00	
	2.3.19.11 LIBROS, TEXTOS Y OTROS MATERIALES IMPRESOS	5	GLOBAL	50,00	250		
2.3.21.2	VIAJES DOMESTICOS					600,00	
	2.3.21.21 PASAJES Y GASTOS DE TRANSPORTE	200	GLOBAL	3,00	600		
2.3.22.2	SERVICIOS DE TELEFONIA E INTERNET					4.320,00	
	2.3.22.21 SERVICIO DE TELEFONIA MOVIL	12	GLOBAL	180,00	2.160,00		
	2.3.22.23 SERVICIO DE INTERNET	12	GLOBAL	180,00	2.160,00		
2.3.22.4	SERVICIO DE PUBLICIDAD, IMPRESIONES, DIFUSION E IMAGEN INSTITUCIONAL					400,00	
	2.3.22.44 SERVICIO DE IMPRESIONES, ENCUADERNACION Y EMPASTADO	800	GLOBAL	0,50	400,00		
2.2.23.1	SERVICIOS DE LIMPIEZA, SEGURIDAD Y VIGILANCIA					200,00	
	2.2.23.11 SERVICIOS DE LIMPIEZA E HIGIENE	1	GLOBAL	200,00	200,00		
<b>TOTAL PRESUPUESTO DEL PRODUCTO ACREDITABLE</b>							<b>6.010,00</b>

## Presupuesto tecnológico

**Tabla 2. Presupuesto tecnológico**

PARTE PRESUP.	DESCRIPCIÓN	CANT.	Unidad de medida	Precio Unitario (S/.)	Precio Total (S/.)	Sub Total (S/.)	TOTAL (S/.)
2.6	ADQUISICION DE ACTIVOS NO FINANCIEROS						<b>7.000,00</b>
2.6.32.3	ADQUISICION DE EQUIPOS INFORMATICOS Y DE COMUNICACIONES						<b>7000,00</b>
2.6.32.31	EQUIPOS COMPUTACIONALES Y PERIFERICOS						
	LAPTOP CORE I7 - 16GB RAM MARCA ASUS	1	GLOBAL	5000,00	5000,00		
	SMARTPHONE SAMSUNG GALAXY J1	10	GLOBAL	200,00	2000,00		
<b>TOTAL PRESUPUESTO DEL PROUCTO TECNOLOGICO</b>							<b>7.000,00</b>

## Resumen presupuestal

**Tabla 3. Resumen presupuestal**

ITEM	DESCRIPCIÓN PRESUPUESTO	SUBTOTAL (S/.)
1	TOTAL PRESUPUESTO DEL PROUCTO ACREDITABLE	6,010.00
2	TOTAL PRESUPUESTO TECNOLÓGICO	7,000.00
<b>TOTAL PRESUPUESTO</b>		<b>S/. 13,010.00</b>

### 1.5. FACTIBILIDAD PARA EL DESARROLLO DEL PROYECTO

#### A. FACTIBILIDAD OPERATIVA

En base a investigaciones citadas en el marco teórico se puede garantizar que los usuarios puedan interactuar de manera correcta con la aplicación móvil.

#### B. FACTIBILIDAD TECNOLÓGICA

Actualmente las personas cuentan con un smartphone desde donde pueden interactuar con el sistema.

#### C. FACTIBILIDAD DE FECHAS

Se puede cumplir con el cronograma de actividades.

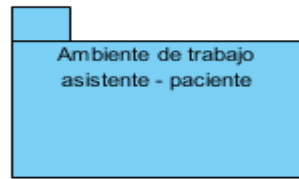
#### D. FACTIBILIDAD FINANCIERA

Se cuenta con factibilidad financiera para desarrollar el sistema dado que la gran parte de la investigación es virtual y el usuario podrá utilizar su propio dispositivo.

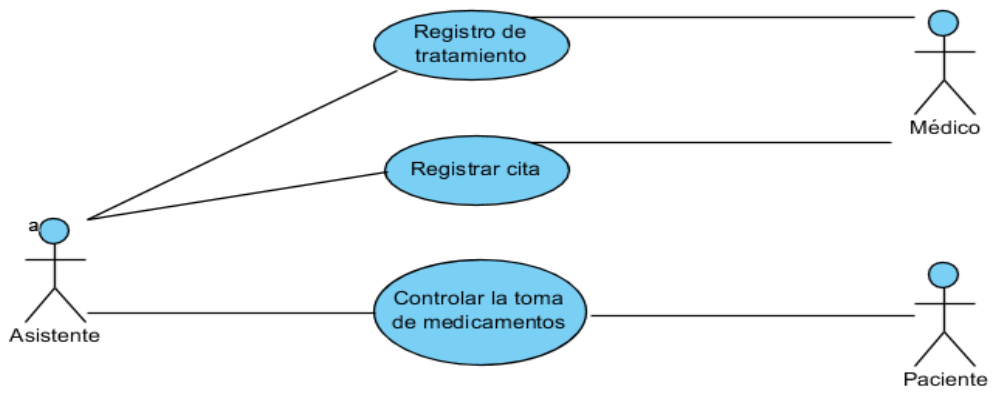
## 2. CAPITULO II: MODELADO DEL NEGOCIO

### 2.1. MODELADO DE NEGOCIO

#### A. DIAGRAMA DE CONTEXTO DEL NEGOCIO



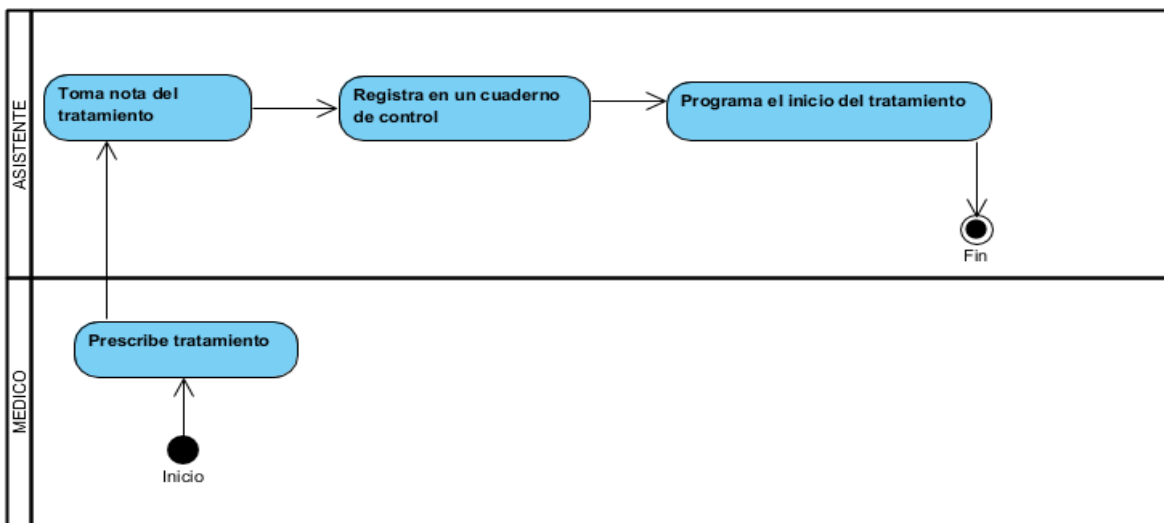
#### B. DIAGRAMA DE CASOS DE USO DE NEGOCIO



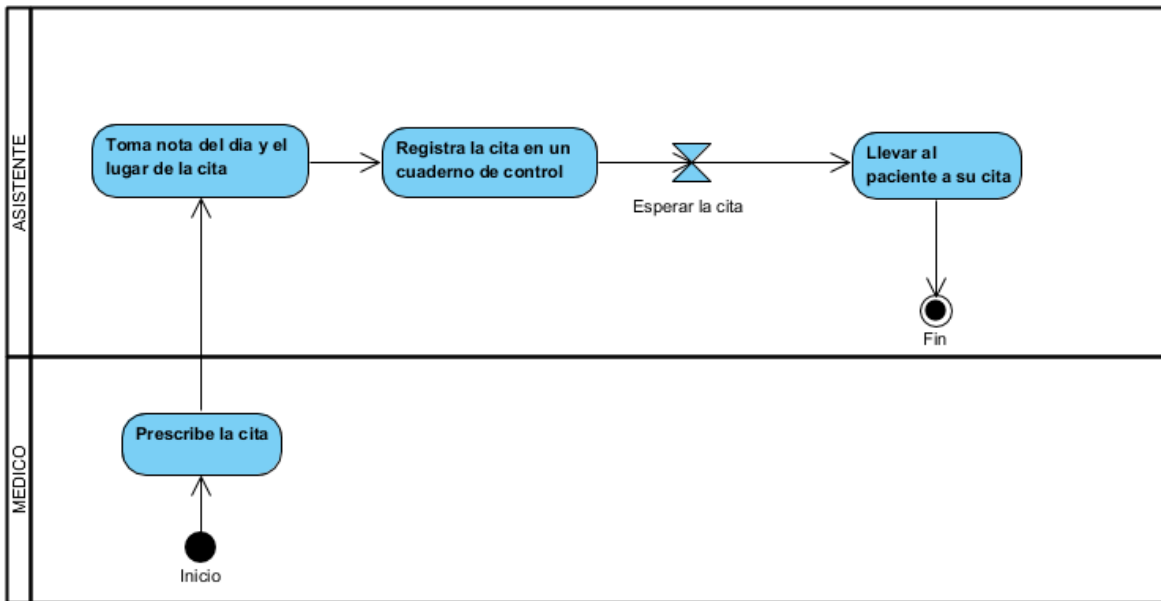
Por cada caso de uso modelar:

#### 1. DIAGRAMA DE ACTIVIDADES

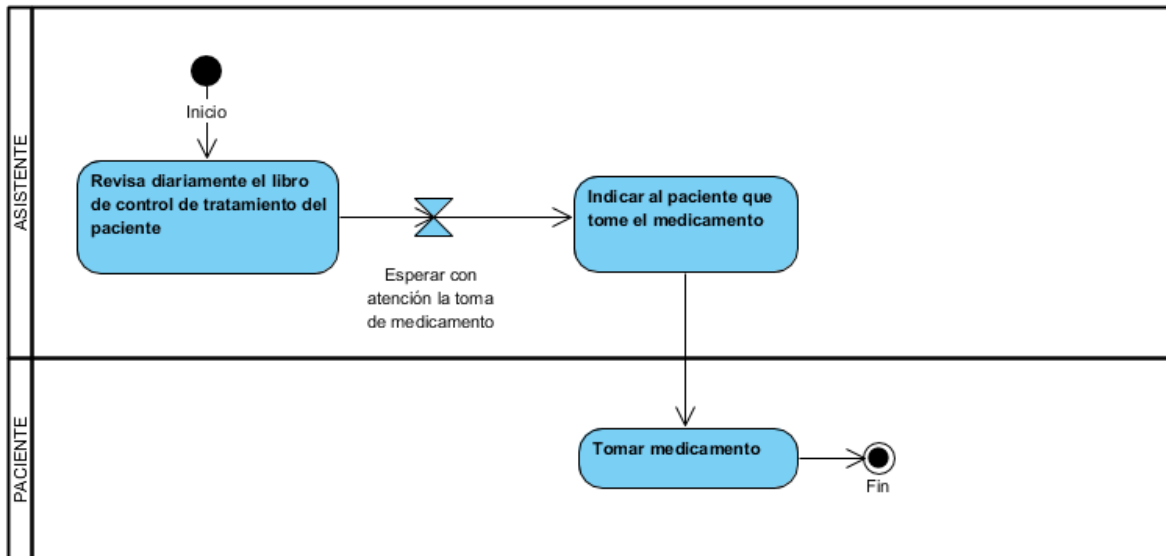
##### a. DACUN Registro Tratamiento



### b. DACUN Registrar Cita



### c. DACUN Controlar la tome de medicamentos





### 3. CAPITULO III: REQUISITOS

#### 3.1. MODELO DE CASOS DE USO (REQUERIMIENTOS)

##### LISTA DE REQUERIMIENTOS:

###### Android

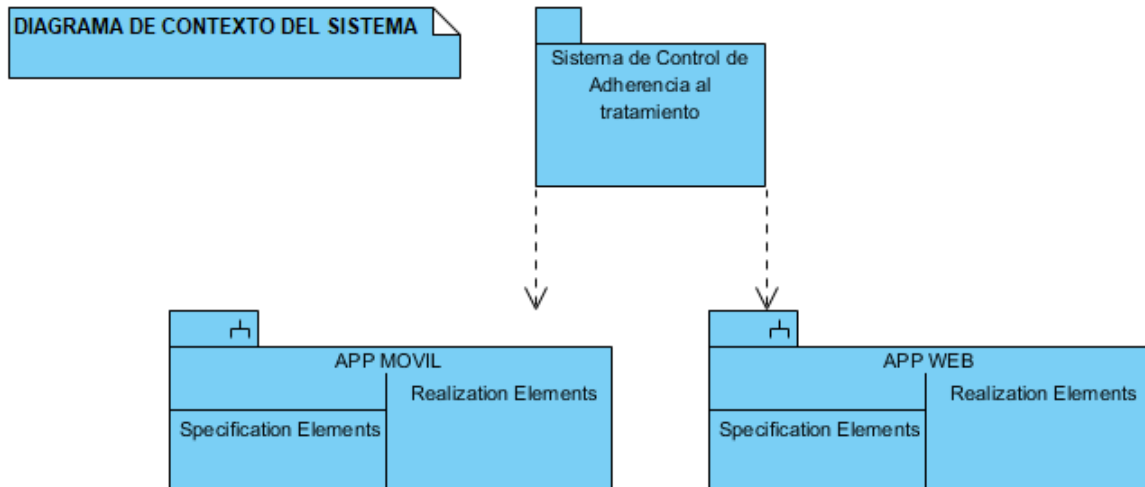
- Se requiere guardar información del **paciente** como su dni, apellidos, nombres, fecha de nacimiento, teléfono, sexo y el estado del paciente (activo o inactivo). Un paciente puede tener una o más enfermedades. A su vez un paciente puede tener uno o más personas de contacto y una persona puede ser contacto de uno o más pacientes. Entre el contacto y el paciente debe haber una relación de parentesco y el contacto debe contar con un número para que pueda ser ubicado en caso de emergencias.
- Un paciente puede ser asistido por uno o más asistentes médico, pero no por más de 1 a la vez. El asistente puede agregar pacientes a su lista. Se requiere guardar información de la fecha de inicio y fin en que el asistente trabaja con el paciente.
- El asistente que gestione la información del paciente debe guardar cita del paciente en los diferentes hospitales o centros médicos donde tenga que ir. De la cita médica se requiere guardar información como: el paciente que asiste a la cita, el hospital donde se hará la cita, la fecha y hora de la cita, el tipo de cita (presencial o no presencial) y el estado de la cita (pendiente, realizado, cancelado).
- En las citas, el doctor puede asignarle tratamientos al paciente, del tratamiento además se desea guardar información como: el tipo del tratamiento, la cita donde se dio el tratamiento y el doctor que lo dictó, la descripción breve del tratamiento y su fecha de inicio. En el tratamiento se pueden dictar 1 o más dosis de toma o aplicación para el paciente. De este detalle se debe guardar información como el fármaco que va a recibir el paciente, cuántos (cantidad) y cada cuanto tiempo (frecuencia de toma) por un número de días determinado (días de toma).
- El tratamiento además esta determinado de un tipo de tratamiento del cual se desea guardar información como el nombre del tipo de tratamiento.
- Se requiere tener control del tratamiento y poder indicar si se cumplió con la toma a la hora y día indicados según el registro del tratamiento.
- El asistente debe tener la posibilidad de buscar las farmacias más cercanas en la aplicación móvil.

###### Web

- Se requiere guardar información de las **enfermedades** de los pacientes como su nombre. Además, estas enfermedades pueden estar activas o inactivas para su selección. Una enfermedad puede presentarse en uno o más pacientes.
- Se requiere guardar información del **parentesco** del contacto del paciente como el nombre del parentesco. Estos parentescos deben ser todos los posibles que existen.

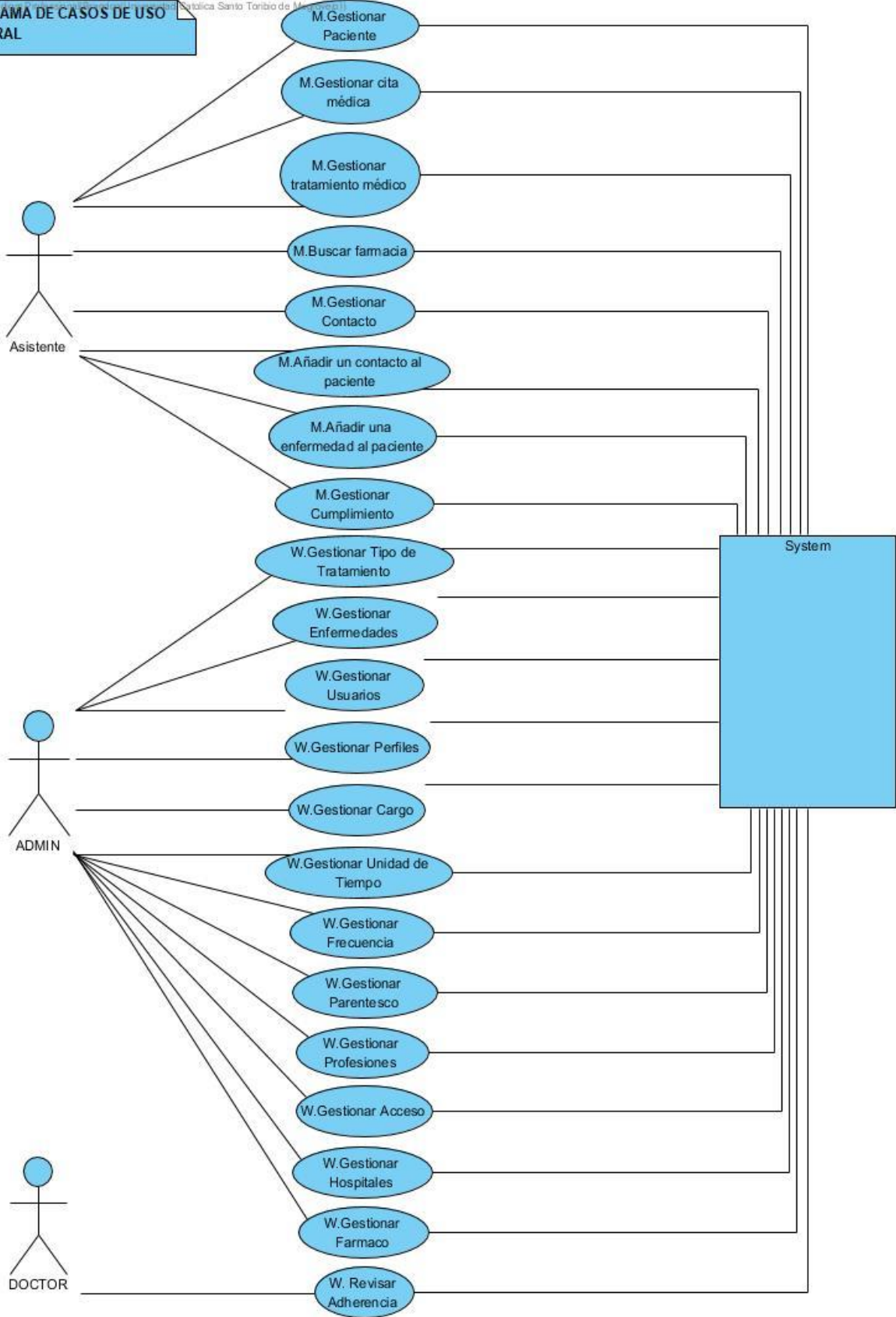
- Se debe guardar información de la persona que actúa como **contacto del paciente**. De ella se debe guardar el dni, apellidos, nombre. Esta persona puede quedar inactiva cuando se requiera.
- Se requiere guardar información del asistente como su dni, la profesión que tiene, el cargo que ocupa, apellidos, nombres, fecha de nacimiento, dirección, teléfono y su estado (activo o inactivo) por si en algún momento deja de trabajar en la empresa.
- Se requiere guardar información de los cargos que desempeña cada usuario que interactúe con el sistema ya sea el doctor, asistente o el administrador, entre otros (si se requiere). Del doctor se requiere guardar información como su dni, la profesión que tiene, el cargo que ocupa, apellidos, nombres, fecha de nacimiento, dirección, teléfono y su estado (activo o inactivo) por si en algún momento deja de trabajar en la empresa.
- Del hospital donde se realiza la cita se requiere guardar información como: el nombre del hospital y la dirección
- Se requiere guardar información de la dosis o fármaco que será aplicado al paciente como su nombre.
- Todo personal registrado debe tener un usuario con un login (dni) y una clave de acceso. Cada personal tiene diferentes accesos según el perfil que tenga tanto en la versión web como Android.
- Cada personal encargado de los pacientes (asistente médico) debe tener sus propios pacientes, cualquier otro asistente no puede acceder a la información de estos.

### A. DIAGRAMA DE CONTEXTO



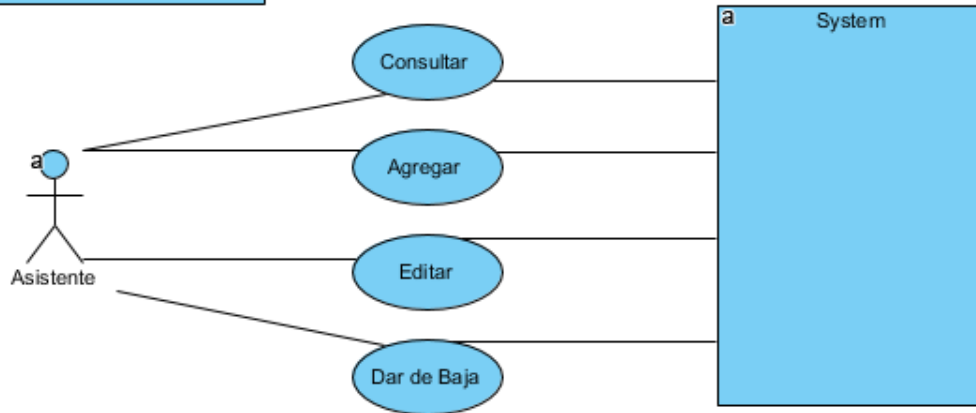
### B. DIAGRAMA DE CASOS DE USO

DIAGRAMA DE CASOS DE USO GENERAL

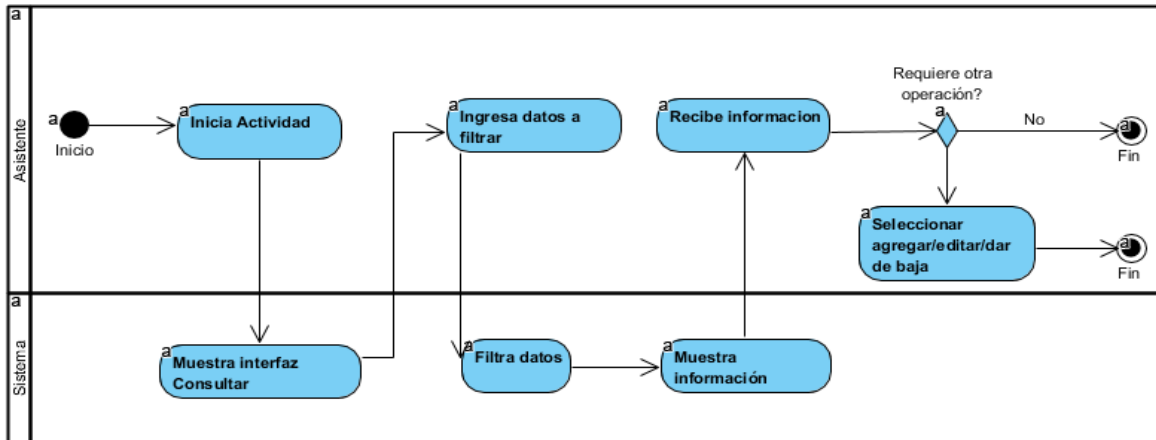


- **M. Gestionar Paciente**
  - **DCU Gestionar Paciente**

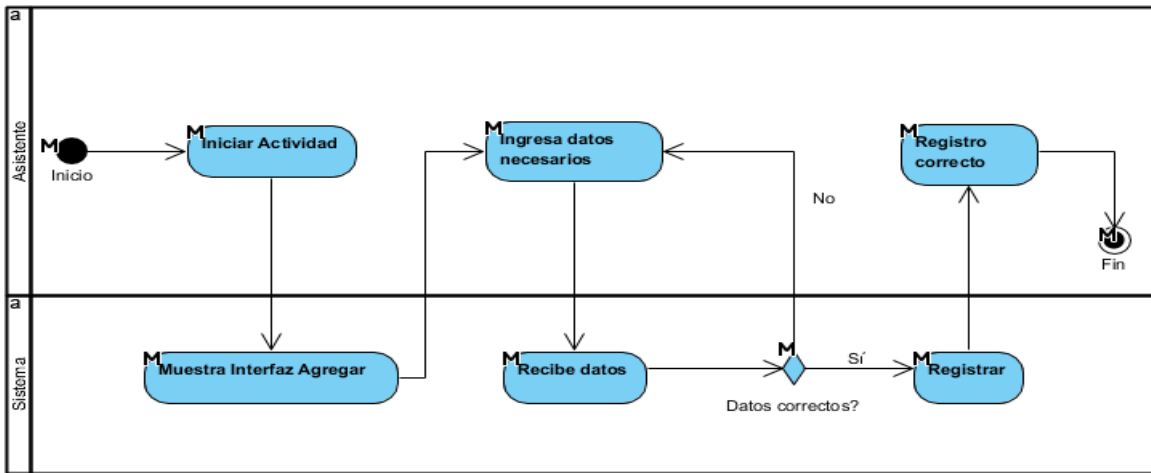
**DIAGRAMA DE CASO DE USO**  
- Gestionar Paciente



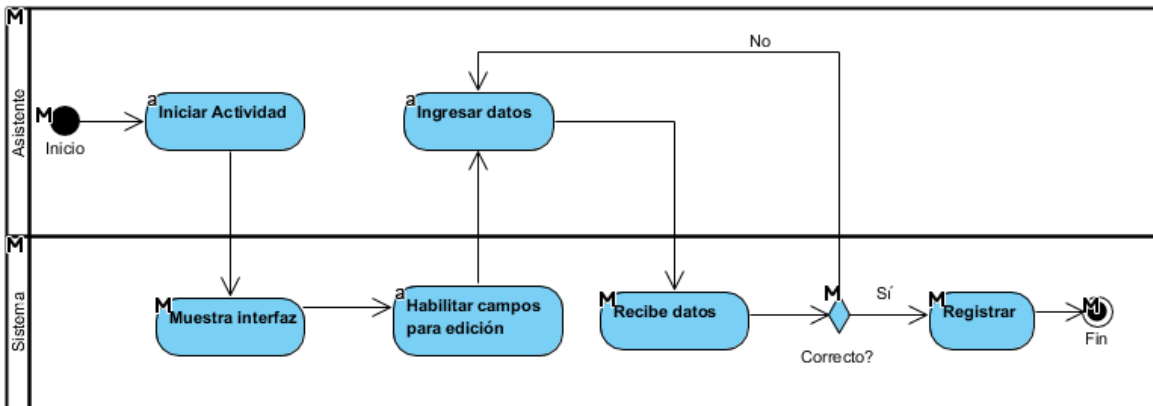
- **DA Gestionar Paciente**
  - **DACU M. Consultar**



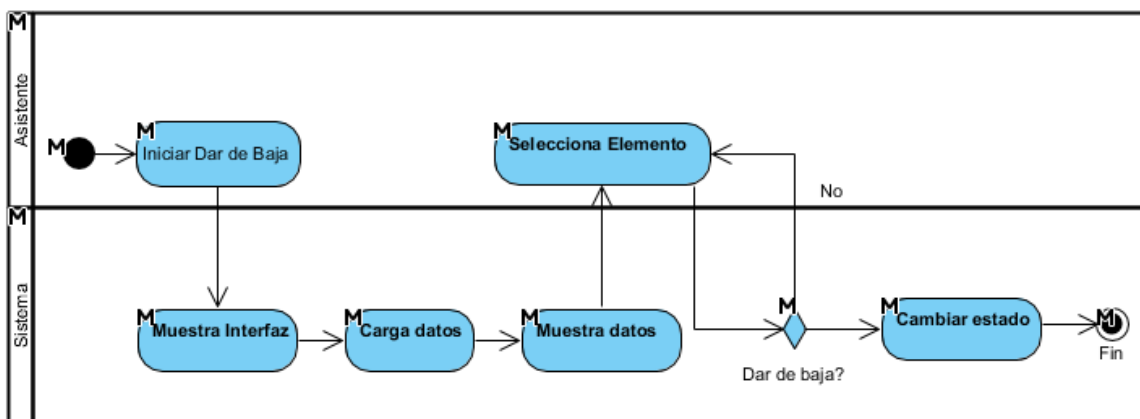
▪ DACU M. Agregar



▪ DACU M. Editar



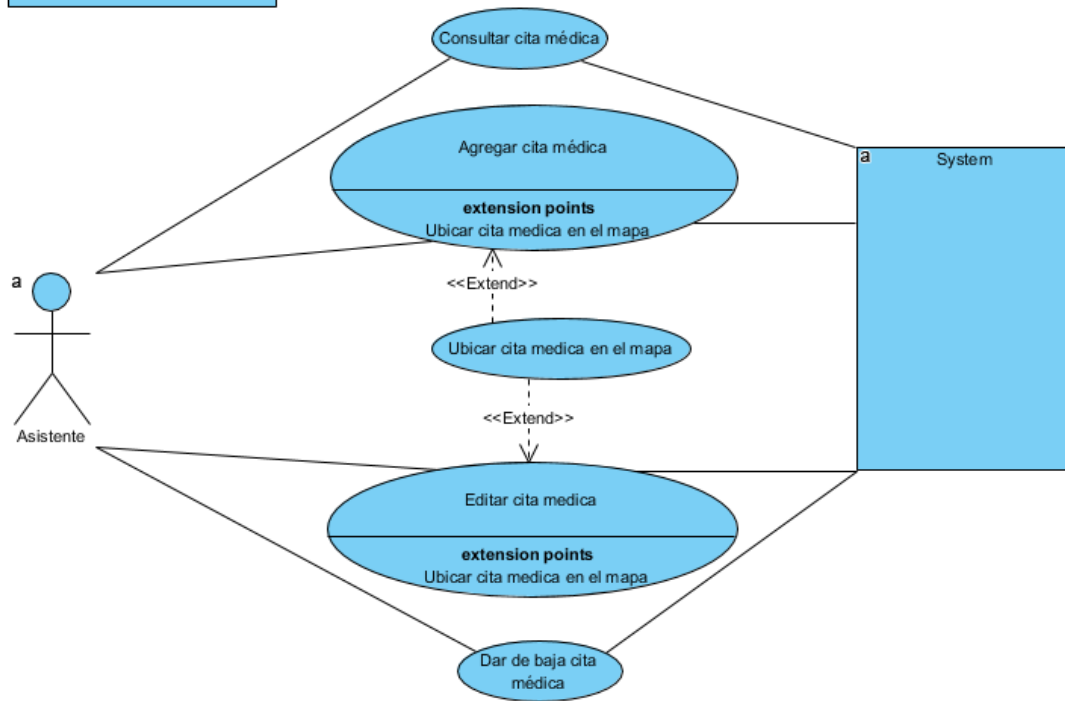
▪ DACU M. Dar de baja



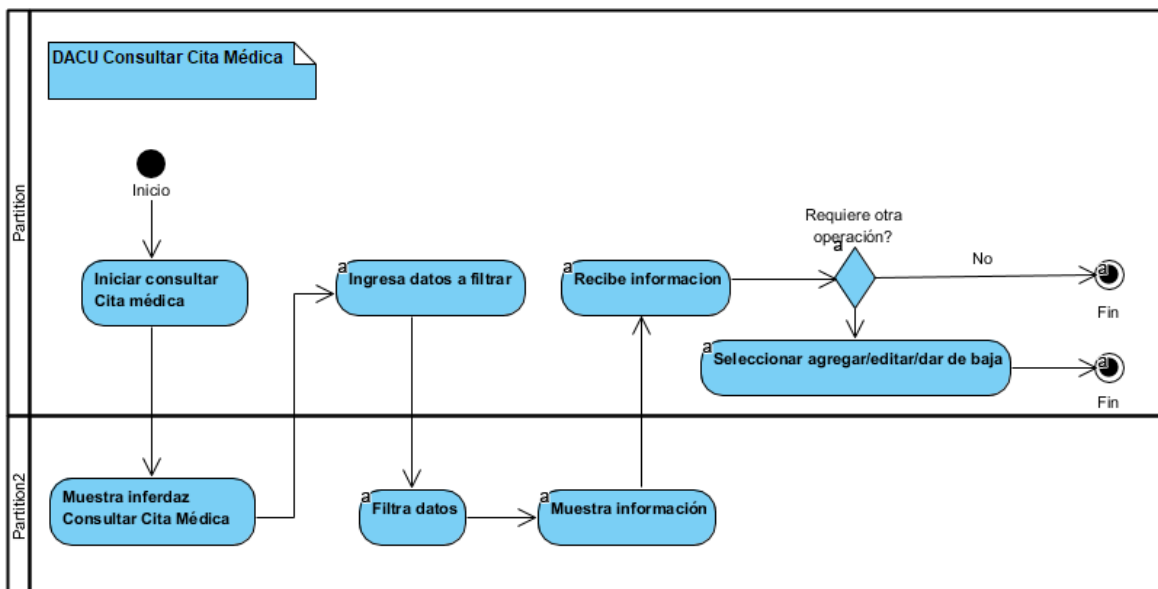
- **DT Gestionar Paciente**
  - **DACU M. Consultar:**
    1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
    2. El sistema carga los datos iniciales y los muestra.
    3. El paciente selecciona un elemento.
    4. El sistema devuelve la información y la muestra.
    5. ¿El asistente tiene otra acción por realizar? No (paso 7)
    6. Fin.
    7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.
  - **DACU M. Agregar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El asistente ingresa los datos requeridos.
    3. El sistema recibe los datos y comprueba si están bien ingresados.
    4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    5. Fin.
    6. El asistente vuelve a ingresar datos.
  - **DACU M. Editar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema habilita los campos disponibles para su edición.
    3. El asistente ingresa los datos requeridos.
    4. El sistema recibe los datos y comprueba si están bien ingresados.
    5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    6. Fin.
    7. El asistente vuelve a ingresar datos.
  - **DACU M. Dar de baja**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema carga los datos iniciales y los muestra.
    3. El paciente selecciona un elemento y elimina
    4. El sistema da de baja al elemento seleccionado
    5. Fin

- **M. Gestionar Cita Médica**
  - **DCU Gestionar Cita Médica**

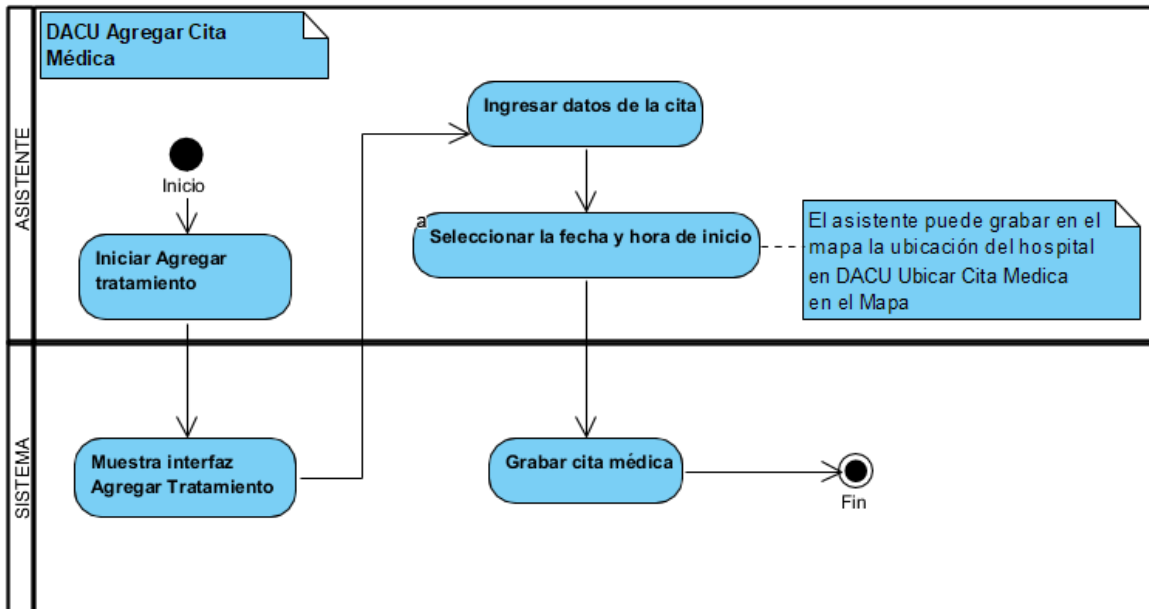
DIAGRAMA DE CASO DE USO - GESTIONAR CITA MÉDICA



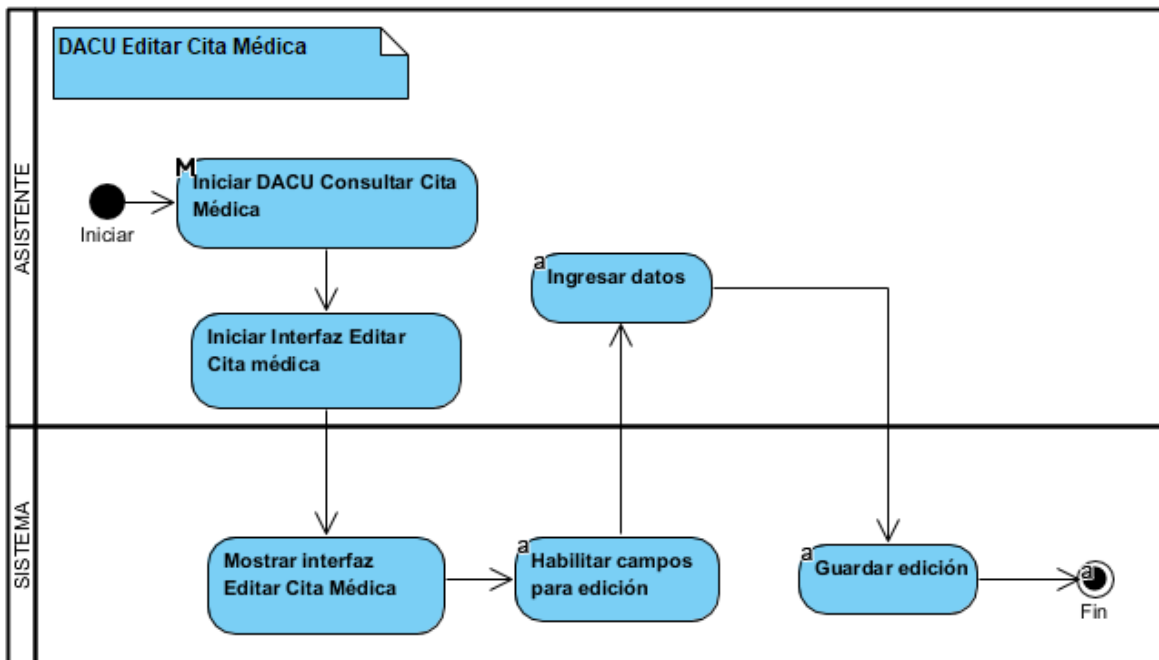
- **DA Gestionar Cita Médica**
  - **DACU M. Consultar**



▪ DACU M. Agregar

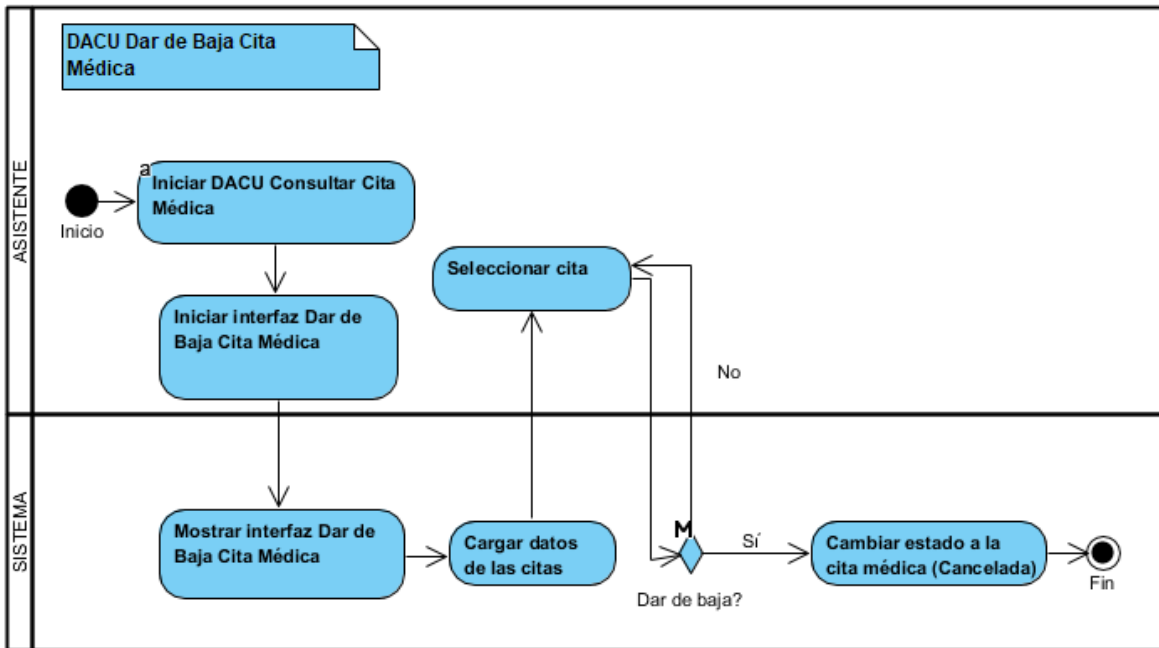


▪ DACU M. Editar



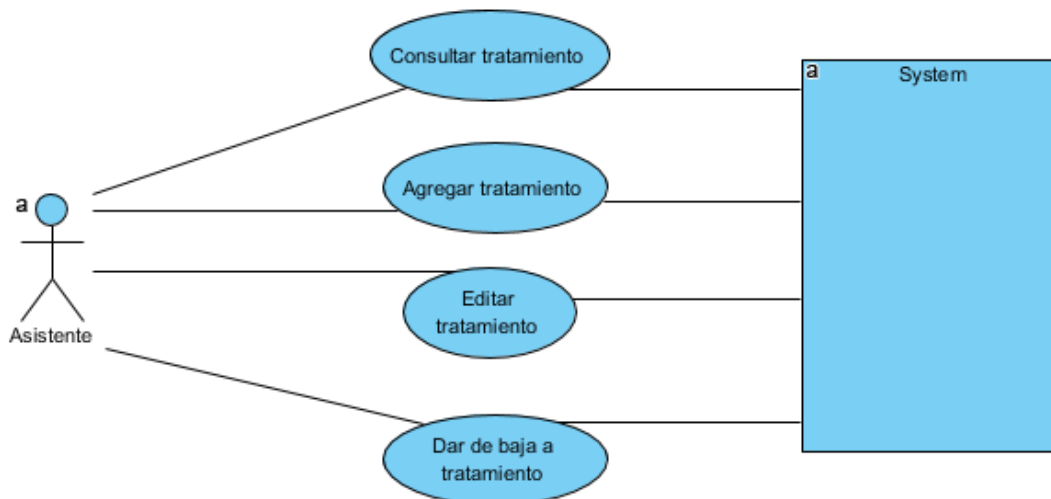


- DACU M. Dar de baja

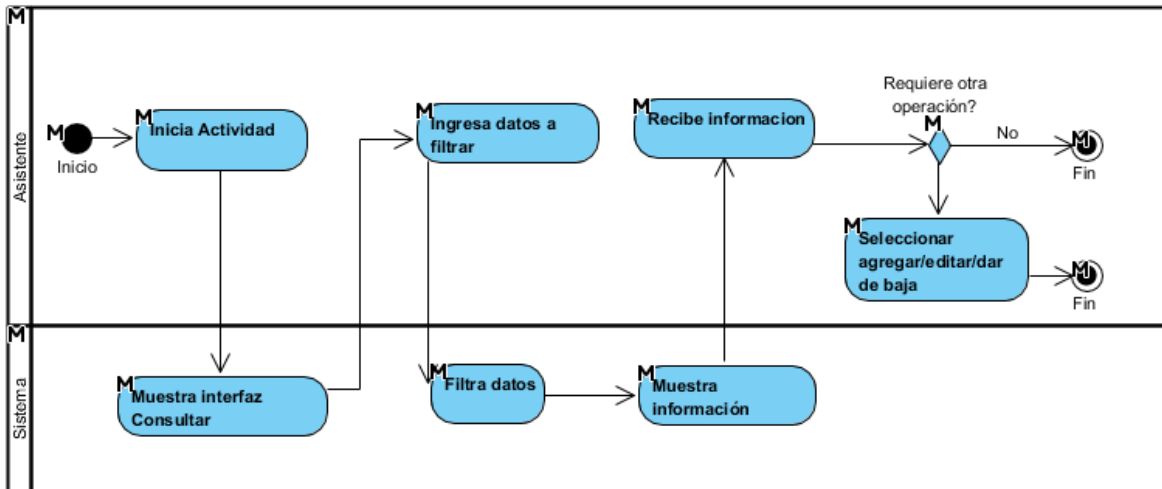


- DT Gestionar Cita Médica
  - M. Gestionar Tratamiento Médico
    - DCU Gestionar Tratamiento Médico

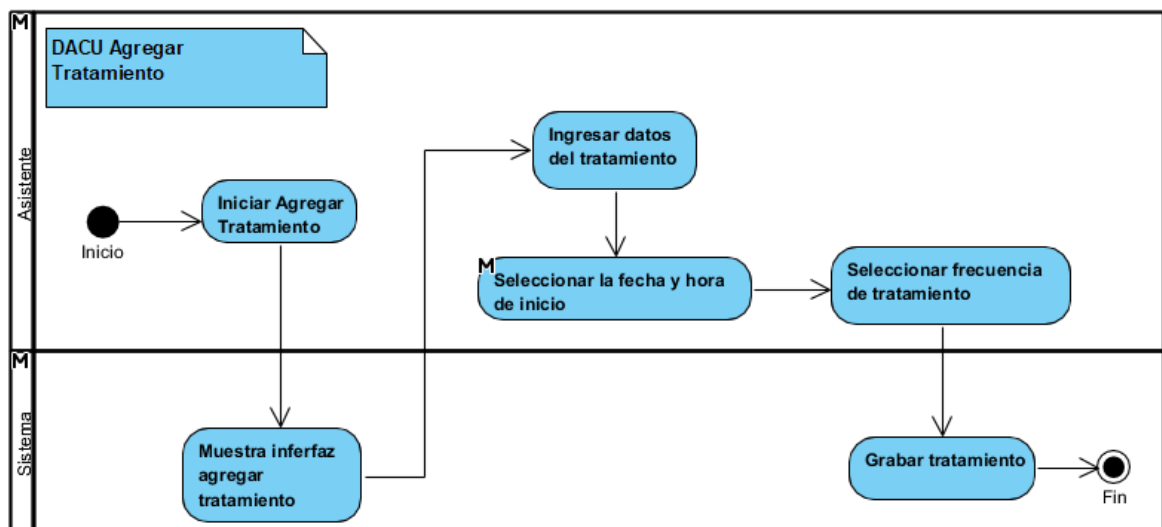
DIAGRAMA DE CASO DE USO  
- Gestionar Tratamiento Médico



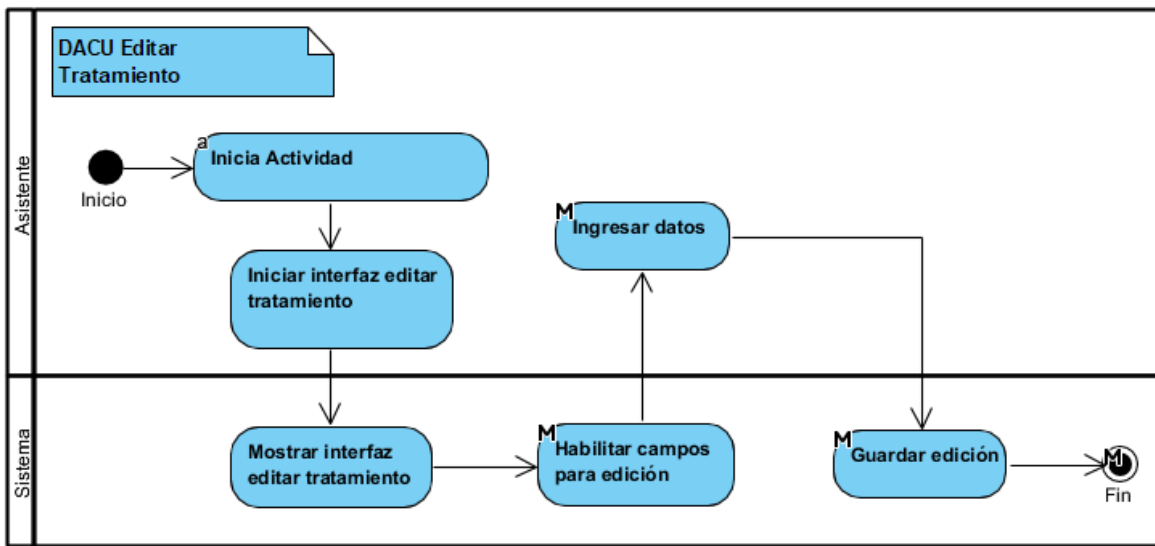
- DA Gestionar Tratamiento Médico
  - DACU M. Consultar



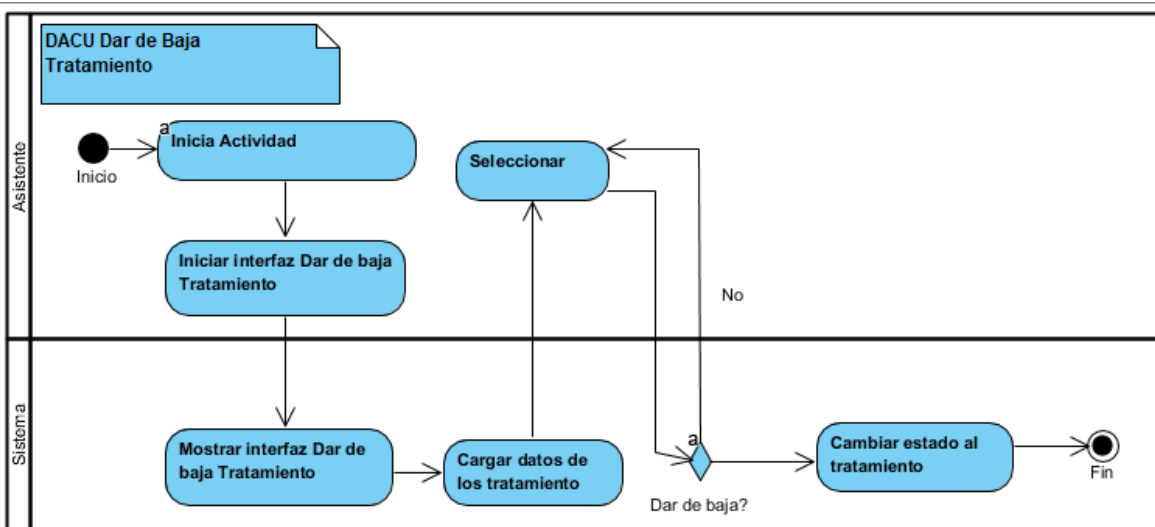
- DACU M. Agregar



- **DACU M. Editar**



- **DACU M. Dar de baja**



- **DT Gestionar Tratamiento Médico**

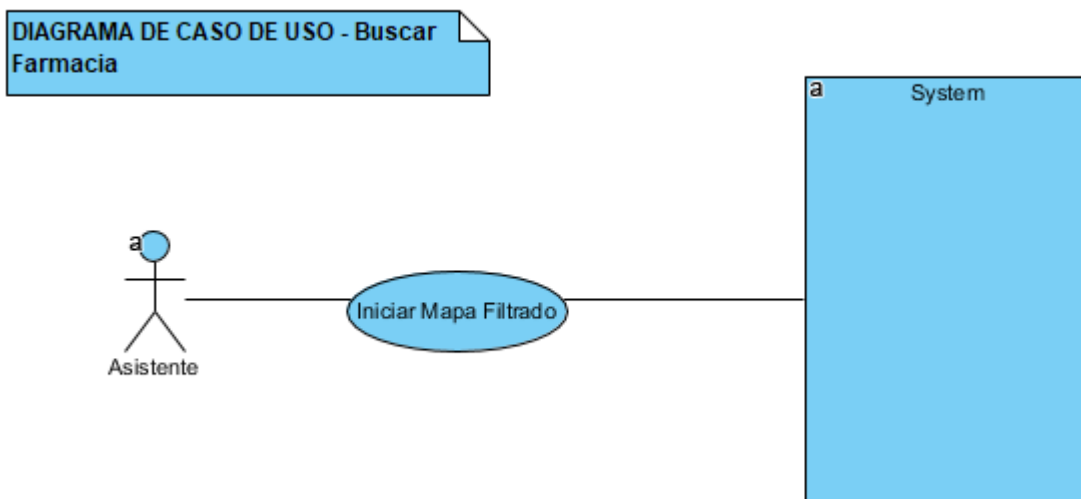
- **DACU M. Consultar**

1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El asistente ingresa los datos para filtrar.
3. El sistema filtra los datos
4. El sistema muestra información.
5. El asistente recibe información.
6. Si el asistente requiere otra operación selecciona entre agregar, editar o dar de baja. Sino Finaliza.

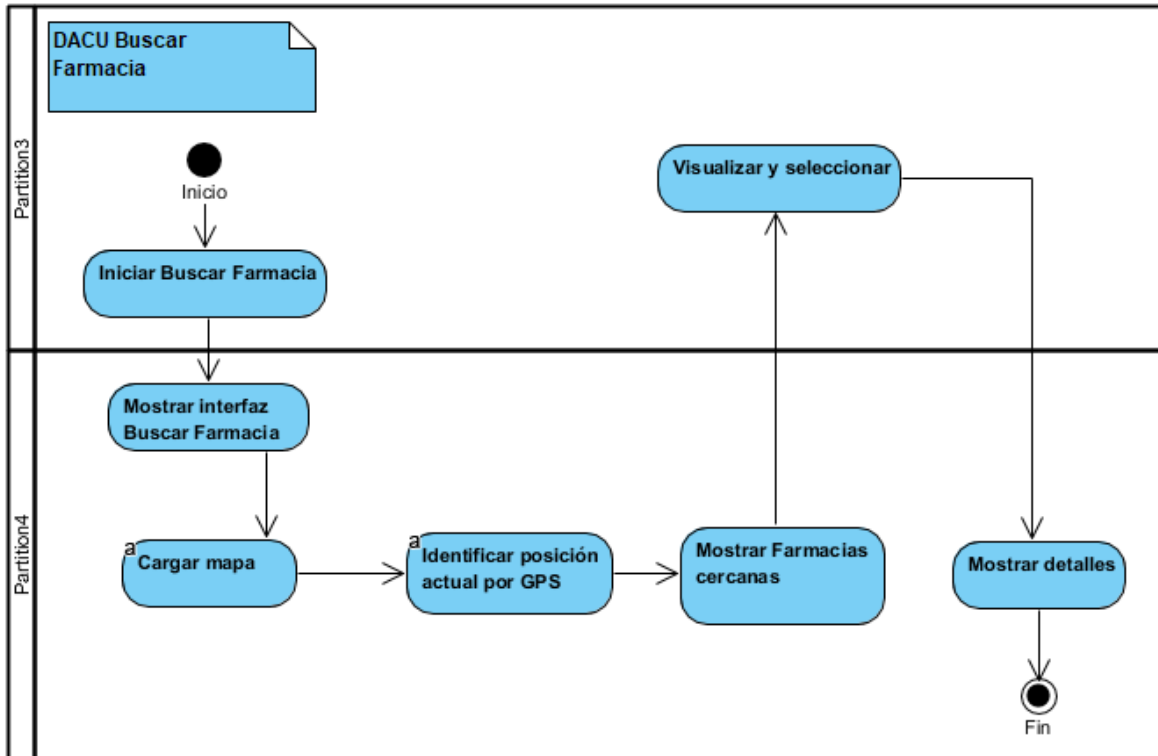
- **DACU M. Agregar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.

2. El asistente ingresa los datos requeridos.
  3. El asistente selecciona la fecha y hora de inicio.
  4. El asistente selecciona los fármacos.
  5. El asistente selecciona la frecuencia del tratamiento.
  6. El sistema recibe los datos y guarda el tratamiento.
  7. Fin
- **DACU M. Editar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema habilita los campos disponibles para su edición.
    3. El asistente ingresa los datos requeridos.
    4. El sistema recibe los datos.
    5. El sistema guarda los datos.
    6. El asistente vuelve a ingresar datos.
  - **DACU M. Dar de baja**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema carga los datos iniciales y los muestra.
    3. El paciente selecciona un elemento y elimina
    4. El sistema da de baja al elemento seleccionado
    5. El sistema cambia de estado al tratamiento (CANCELADO)
    6. Fin
- **M. Buscar farmacia**
    - **DCU Buscar farmacia**



- **DA Buscar farmacia**

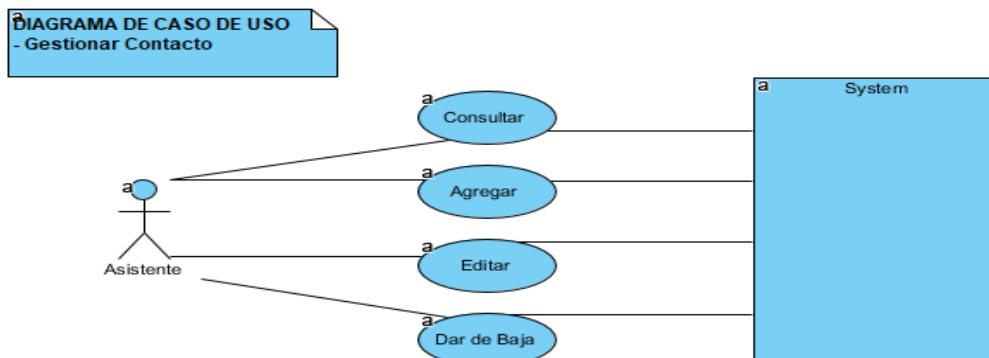


- **DT Buscar farmacia**

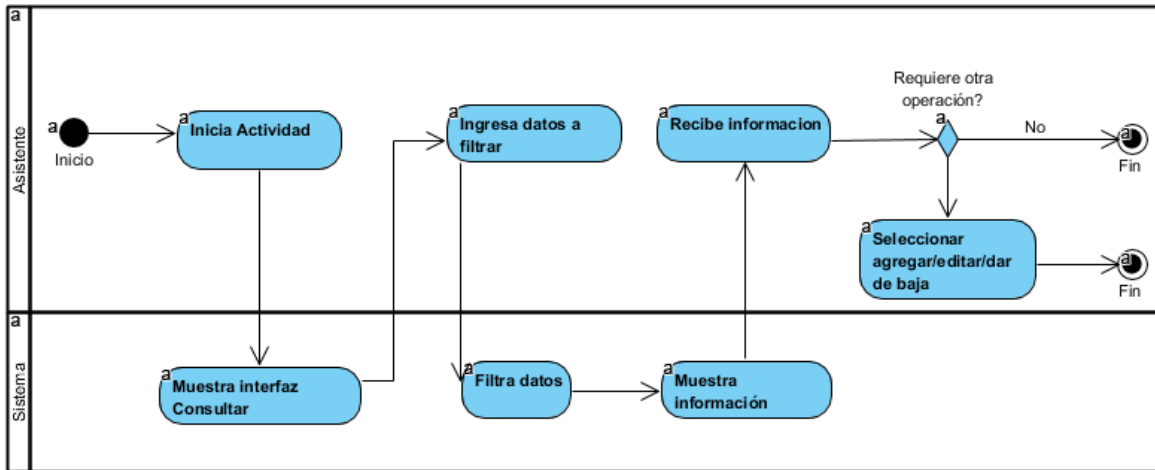
1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El sistema carga el mapa en la interfaz.
3. El sistema identifica la posición GPS del usuario y ubica.
4. El sistema muestra farmacias cercanas a la ubicación del usuario.
5. El asistente visualiza y selecciona una farmacia.
6. El sistema muestra detalles en el mapa.
7. Fin

- **M. Gestionar Contacto**

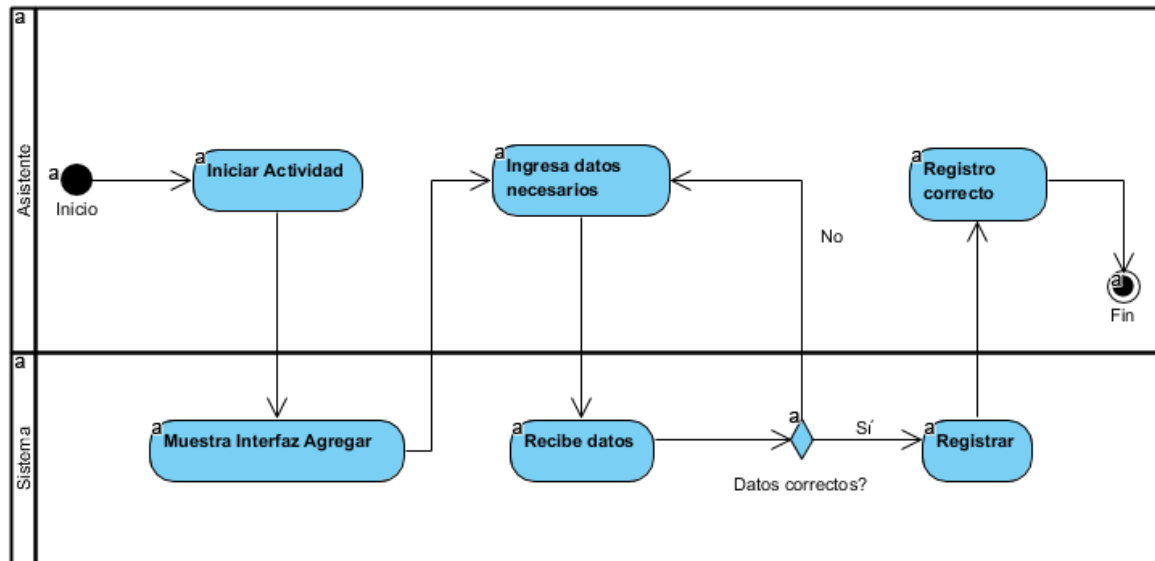
- **DCU Gestionar Contacto**



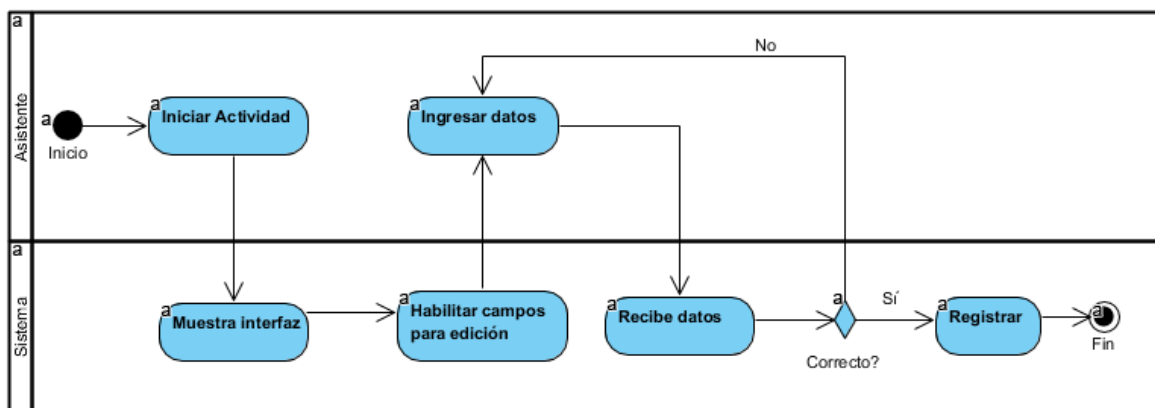
- DA Gestionar Contacto
  - DACU M. Consultar



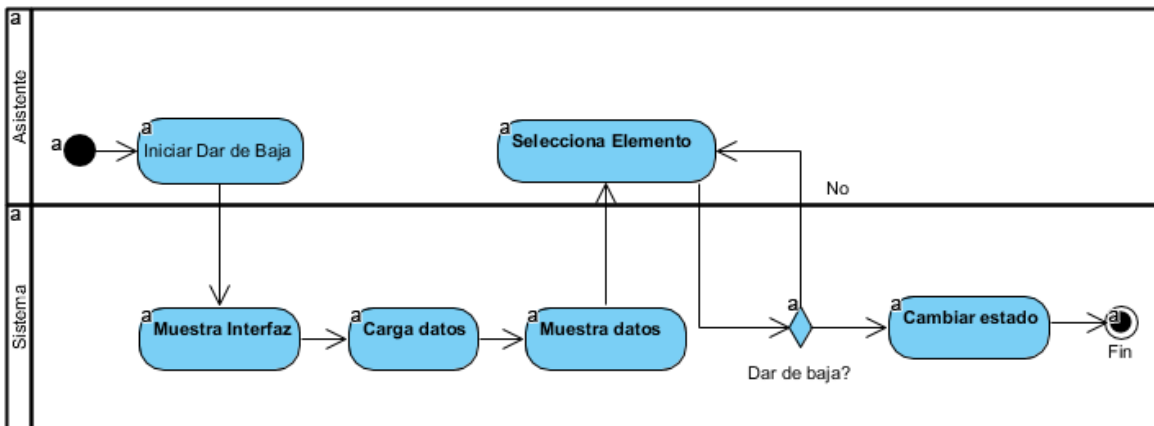
- DACU M. Agregar



- DACU M. Editar



- **DACU M. Dar de baja**



- **DT Gestionar Contacto**

- **DACU M. Consultar:**

1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)
6. Fin.
7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.

- **DACU M. Agregar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El asistente ingresa los datos requeridos.
3. El sistema recibe los datos y comprueba si están bien ingresados.
4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
5. Fin.
6. El asistente vuelve a ingresar datos.

- **DACU M. Editar**

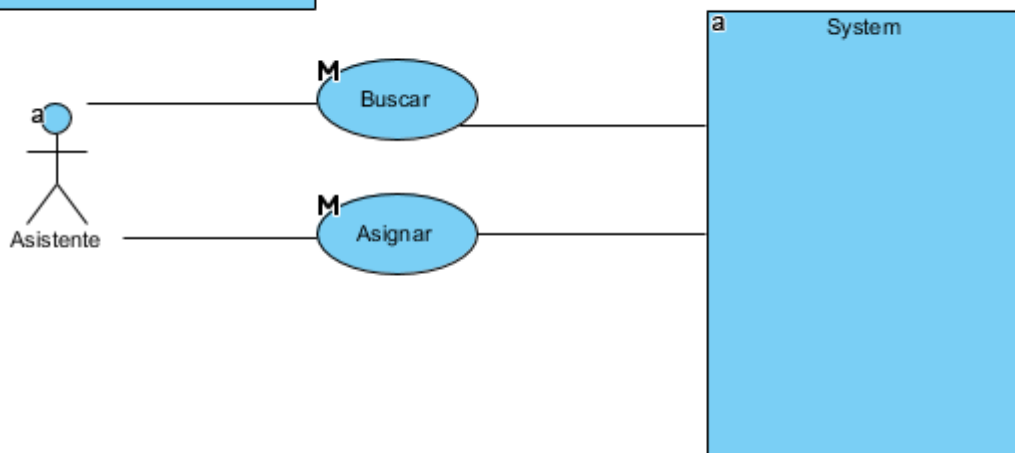
1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El sistema habilita los campos disponibles para su edición.
3. El asistente ingresa los datos requeridos.
4. El sistema recibe los datos y comprueba si están bien ingresados.
5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
6. Fin.
7. El asistente vuelve a ingresar datos.

- **DACU M. Dar de baja**

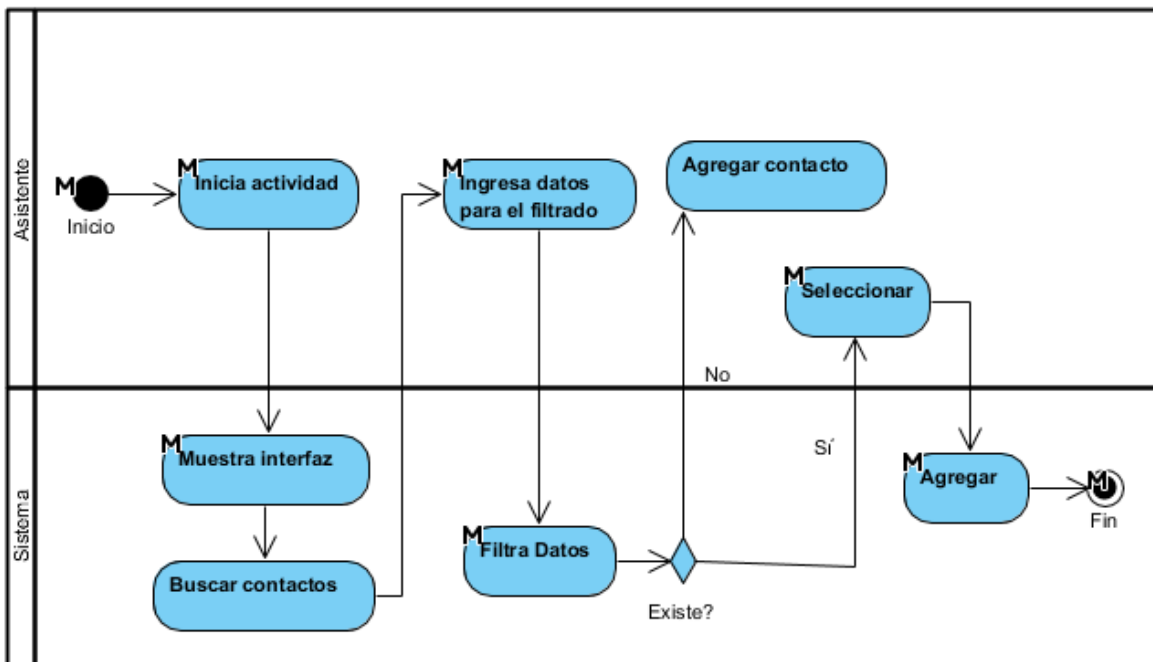
1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento y elimina
4. El sistema da de baja al elemento seleccionado
5. Fin

- **M. Añadir un contacto al paciente**
  - **DCU Añadir un contacto al paciente**

**DIAGRAMA DE CASO DE USO -  
Añadir un contacto a un paciente**



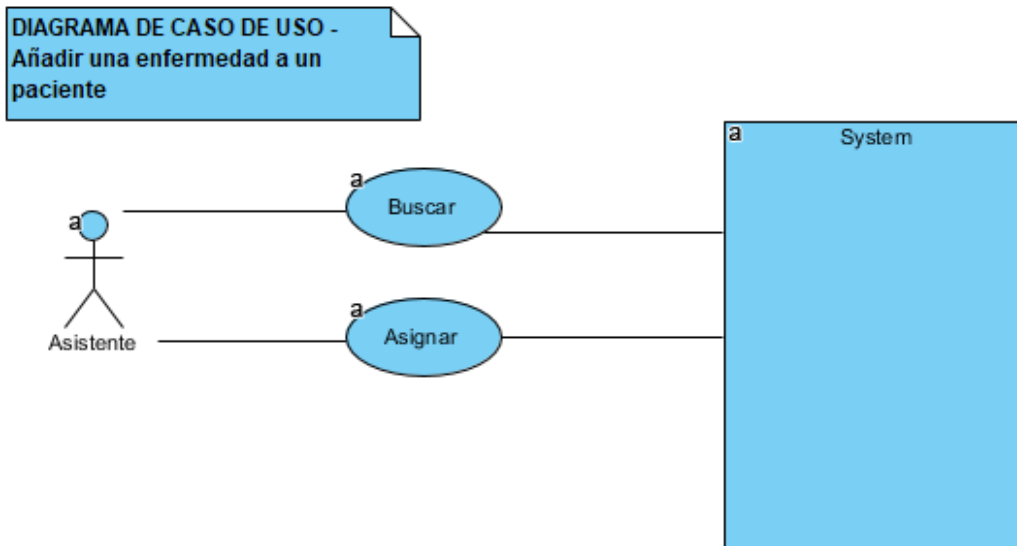
- **DA Añadir un contacto al paciente**



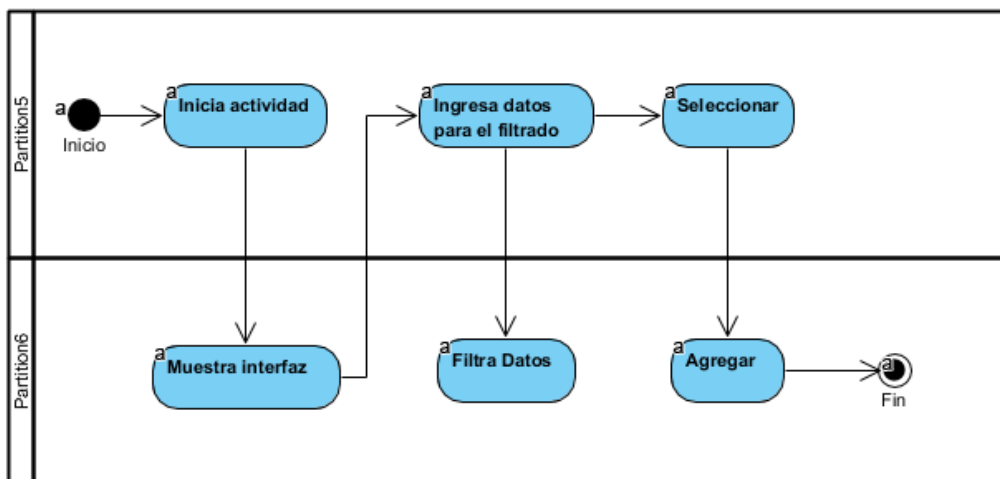


- **DT Añadir un contacto al paciente**
  1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
  2. El sistema empieza a buscar contactos y los muestra
  3. El asistente ingresa datos para filtrar
  4. El sistema filtra datos y muestra resultados
  5. Si el contacto no existe, el paciente lo agrega. Sino selecciona el contacto para asignarle al paciente.
  6. El sistema enlaza el contacto con el paciente.
  7. Fin.

- **M. Añadir una enfermedad al paciente**
  - **DCU Añadir una enfermedad al paciente**

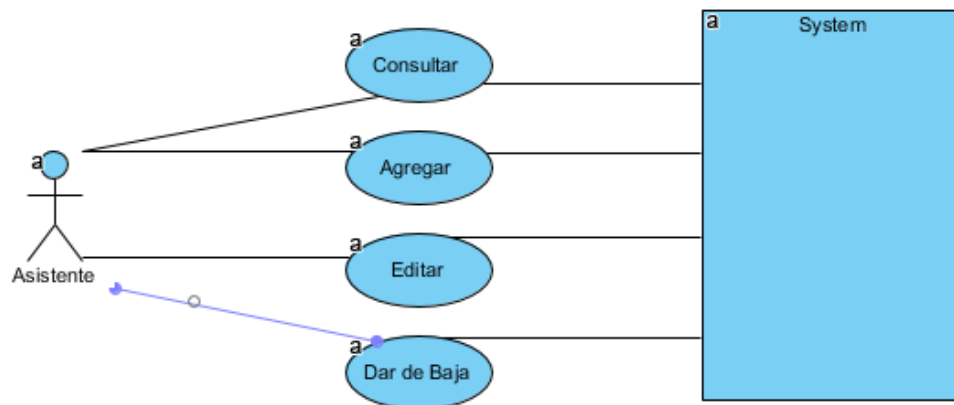


- **DA Añadir una enfermedad al paciente**

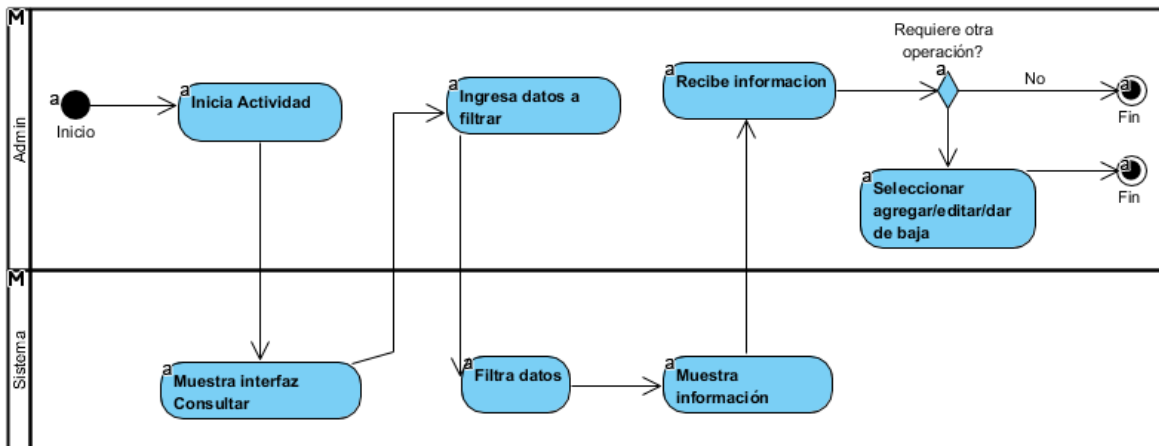


- **DT Añadir una enfermedad al paciente**
  1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
  2. El sistema empieza a buscar enfermedades y las muestra
  3. El asistente ingresa datos para filtrar
  4. El sistema filtra datos y muestra resultados
  5. El asistente selecciona una enfermedad.
  6. El sistema enlaza la enfermedad con el paciente.
  7. Fin.
- **M. Gestionar Cumplimiento**
  - **DCU Gestionar Cumplimiento**
  - **DA Gestionar Cumplimiento**
  - **DT Gestionar Cumplimiento**
- **W. Gestionar tipo de tratamiento**
  - **DCU Gestionar tipo de tratamiento**

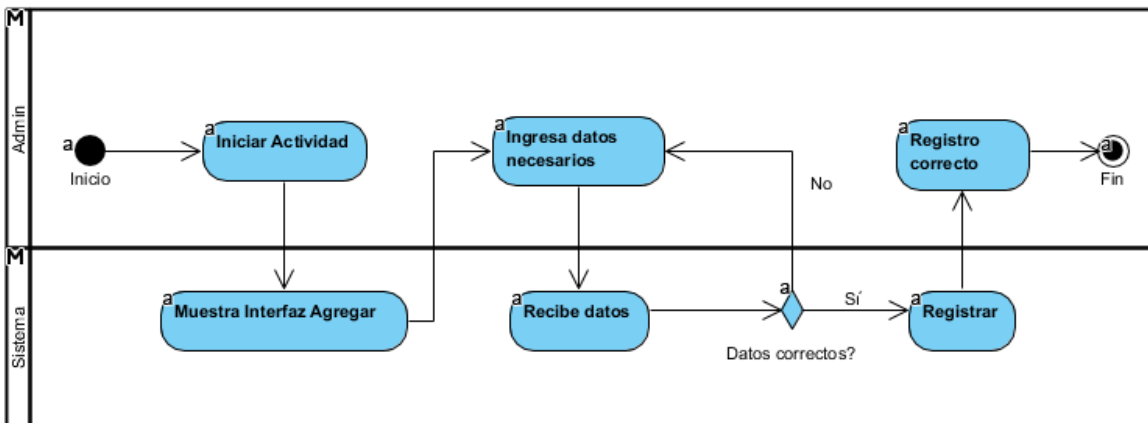
DIAGRAMA DE CASO DE USO -  
Gestionar tipo tratamiento



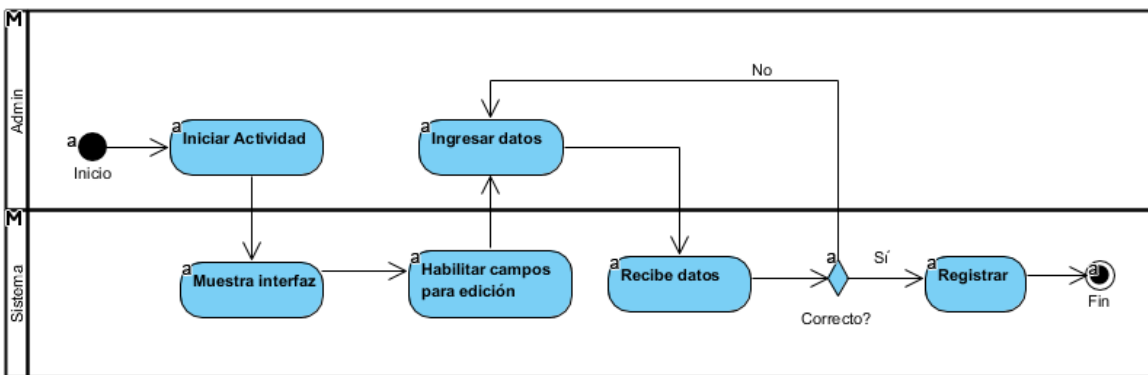
- **DA Gestionar tipo de tratamiento**
  - **DACU M. Consultar**



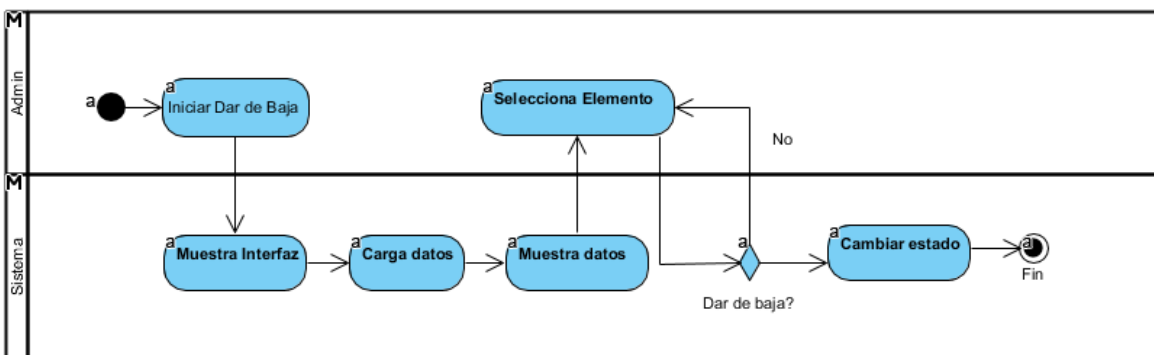
▪ DACU M. Agregar



▪ DACU M. Editar



▪ DACU M. Dar de baja



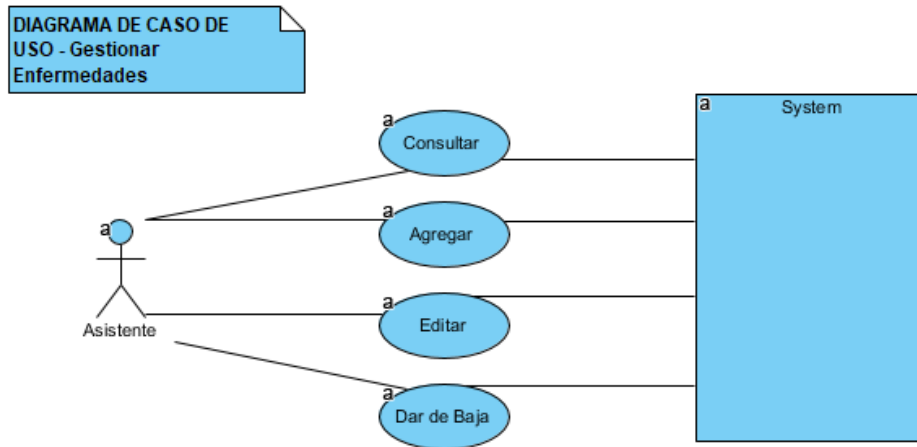
▪ DT Gestionar tipo de tratamiento

▪ DACU W. Consultar:

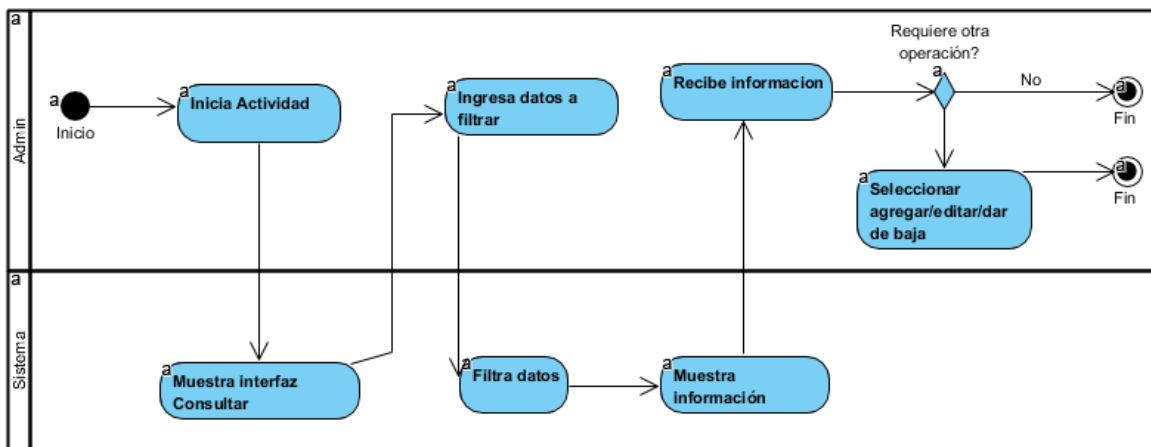
1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)

6. Fin.
  7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.
- **DACU W. Agregar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El asistente ingresa los datos requeridos.
    3. El sistema recibe los datos y comprueba si están bien ingresados.
    4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    5. Fin.
    6. El asistente vuelve a ingresar datos.
  - **DACU W. Editar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema habilita los campos disponibles para su edición.
    3. El asistente ingresa los datos requeridos.
    4. El sistema recibe los datos y comprueba si están bien ingresados.
    5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    6. Fin.
    7. El asistente vuelve a ingresar datos.
  - **DACU W. Dar de baja**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema carga los datos iniciales y los muestra.
    3. El paciente selecciona un elemento y elimina
    4. El sistema da de baja al elemento seleccionado
    5. Fin

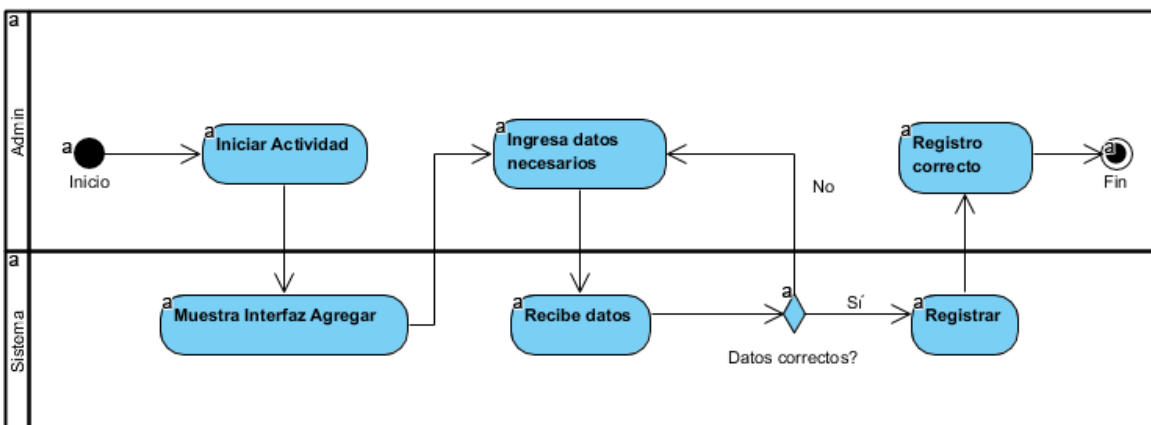
- **W. Gestionar Enfermedad**
  - **DCU Gestionar Enfermedad**



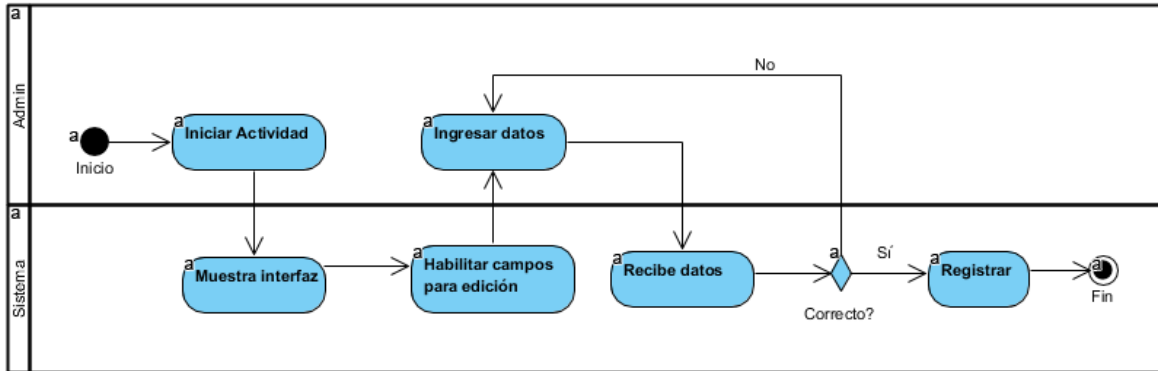
- **DA Gestionar Enfermedad**
  - **DACU M. Consultar**



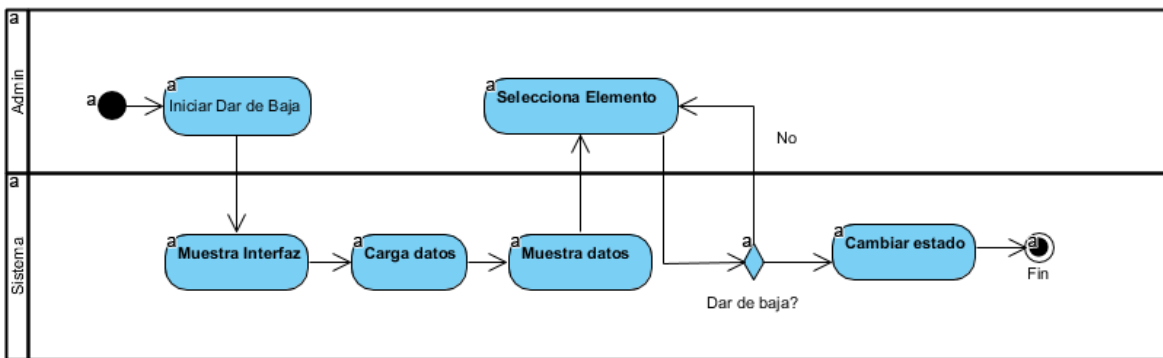
- **DACU M. Agregar**



- **DACU M. Editar**



- **DACU M. Dar de baja**



- **DT Gestionar Enfermedad**

- **DACU W. Consultar:**

1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
  1. El sistema devuelve la información y la muestra.
  2. ¿El asistente tiene otra acción por realizar? No (paso 7)
  3. Fin.

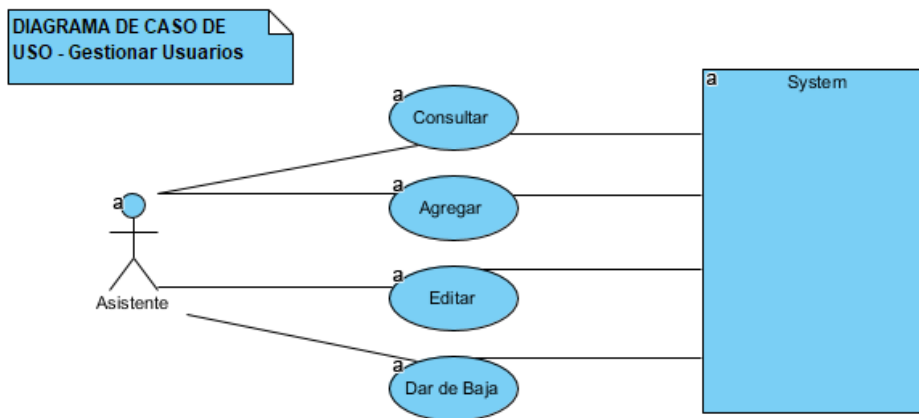
- **DACU W. Agregar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El asistente ingresa los datos requeridos.
  1. El sistema recibe los datos y comprueba si están bien ingresados.
  2. Si están bien ingresados, el sistema guarda los datos. Si no pasa al paso 6.
  3. Fin.
  4. El asistente vuelve a ingresar datos.

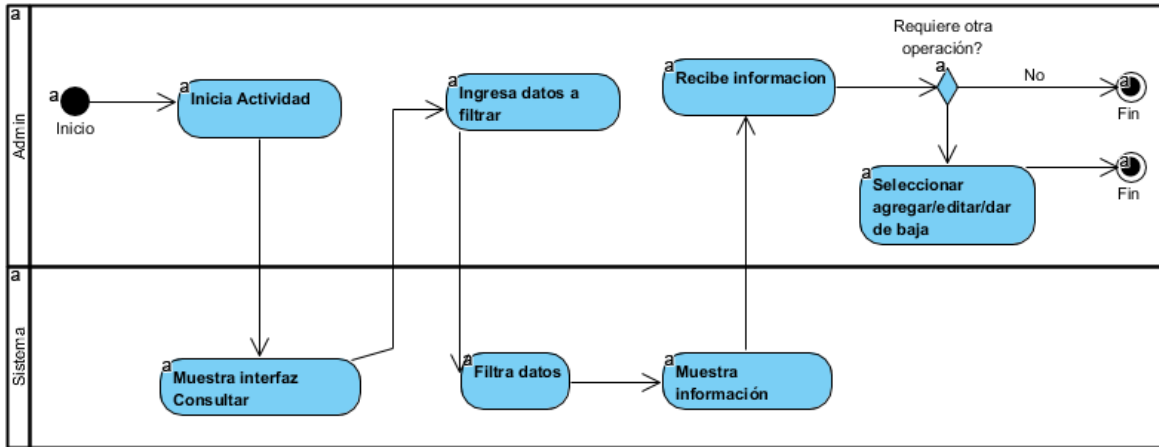
- **DACU W. Editar**
  1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
  2. El sistema habilita los campos disponibles para su edición.
    1. El asistente ingresa los datos requeridos.
    2. El sistema recibe los datos y comprueba si están bien ingresados.
  3. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
  4. Fin.
  5. El asistente vuelve a ingresar datos.
- **DACU W. Dar de baja**
  1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
  2. El sistema carga los datos iniciales y los muestra.
  3. El paciente selecciona un elemento y elimina
  4. El sistema da de baja al elemento seleccionado
  5. Fin

○ **W. Gestionar usuario**

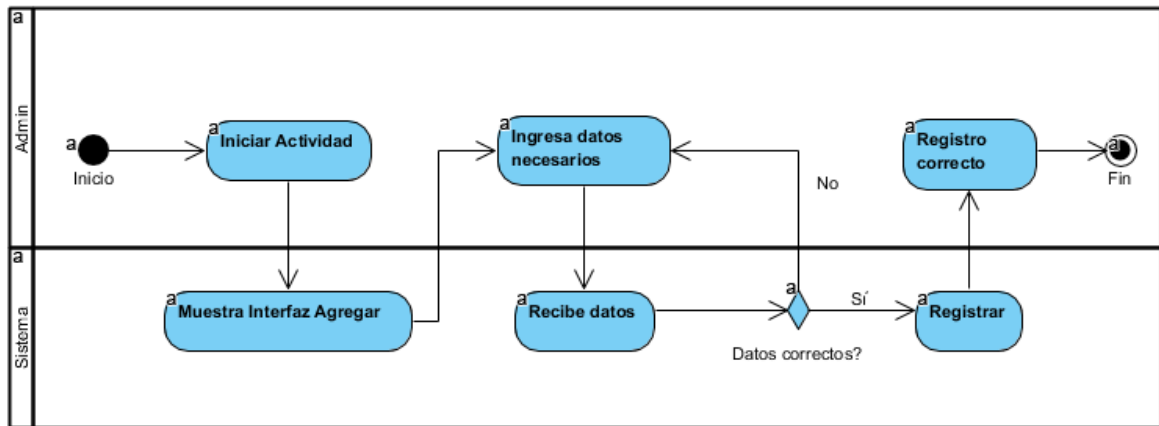
▪ **DCU Gestionar usuario**



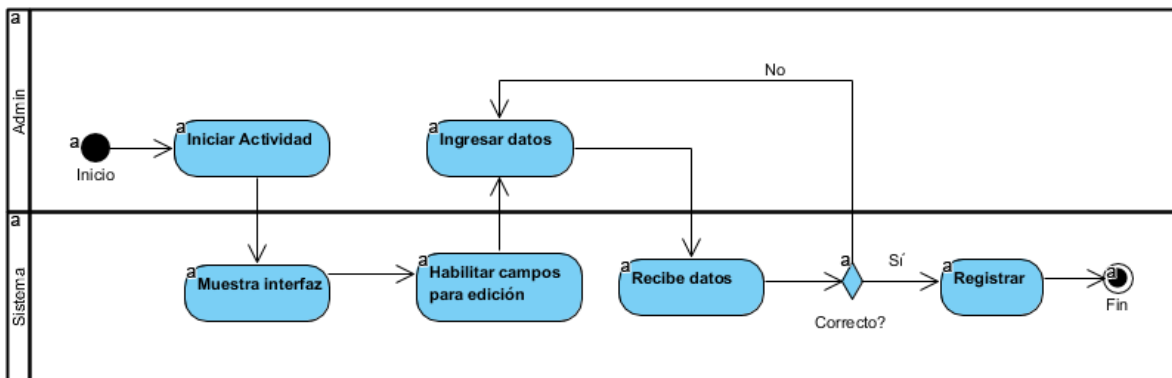
- DA Gestionar usuario
  - DACU M. Consultar



- DACU M. Agregar

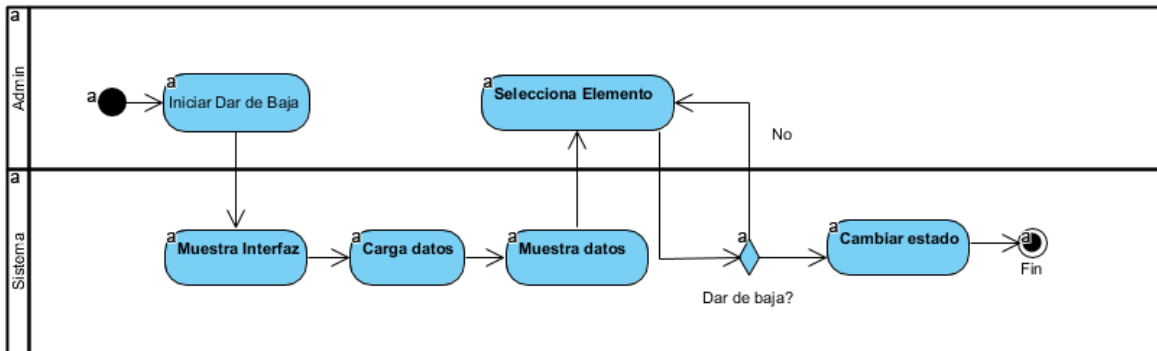


- DACU M. Editar





- **DACU M. Dar de baja**



- **DT Gestionar usuario**

- **DACU W. Consultar:**

1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)
6. Fin.
7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.

- **DACU W. Agregar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El asistente ingresa los datos requeridos.
3. El sistema recibe los datos y comprueba si están bien ingresados.
4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
5. Fin.
6. El asistente vuelve a ingresar datos.

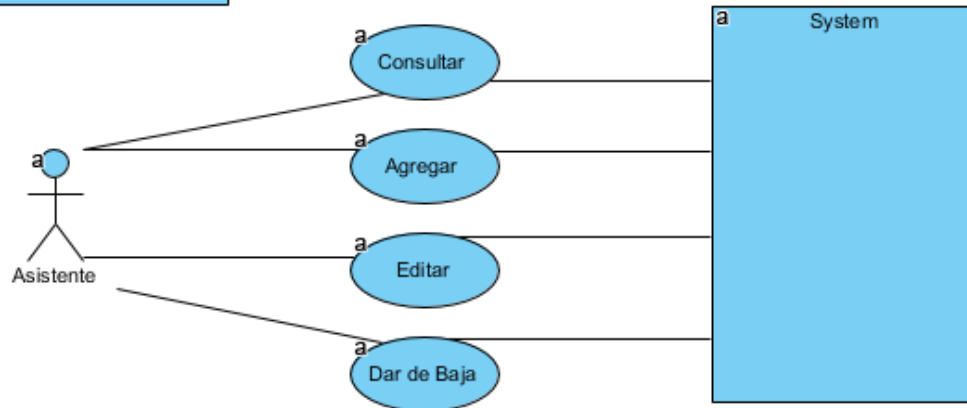
- **DACU W. Editar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El sistema habilita los campos disponibles para su edición.
3. El asistente ingresa los datos requeridos.
4. El sistema recibe los datos y comprueba si están bien ingresados.
5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
6. Fin.
7. El asistente vuelve a ingresar datos.

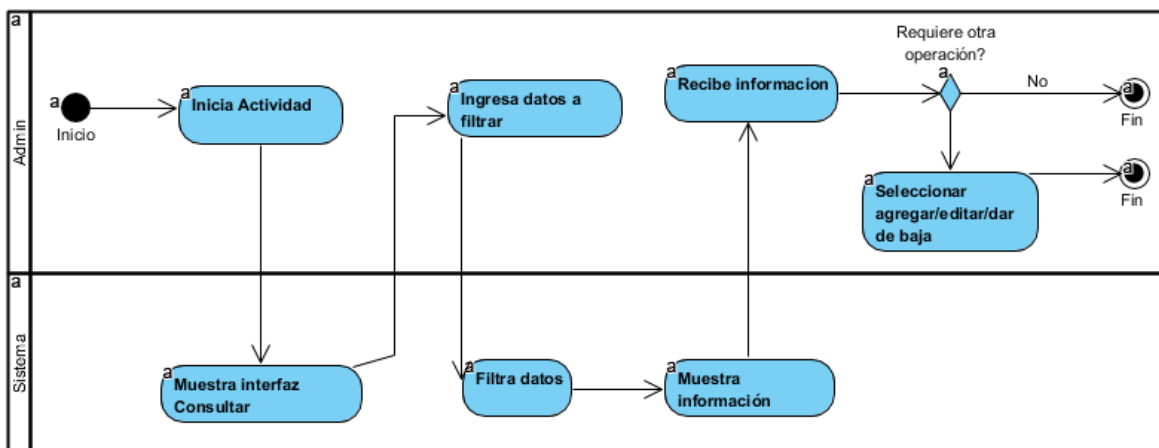
- **DACU W. Dar de baja**
  1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
  2. El sistema carga los datos iniciales y los muestra.
  3. El paciente selecciona un elemento y elimina
  4. El sistema da de baja al elemento seleccionado
  5. Fin

- **W. Gestionar perfiles**
  - **DCU Gestionar perfiles**

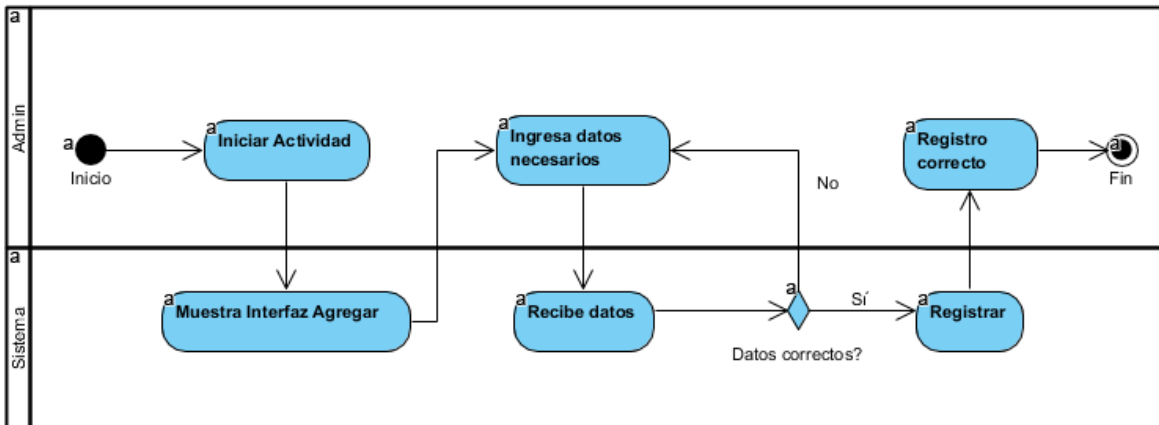
**DIAGRAMA DE CASO DE USO**  
- Gestionar Perfiles



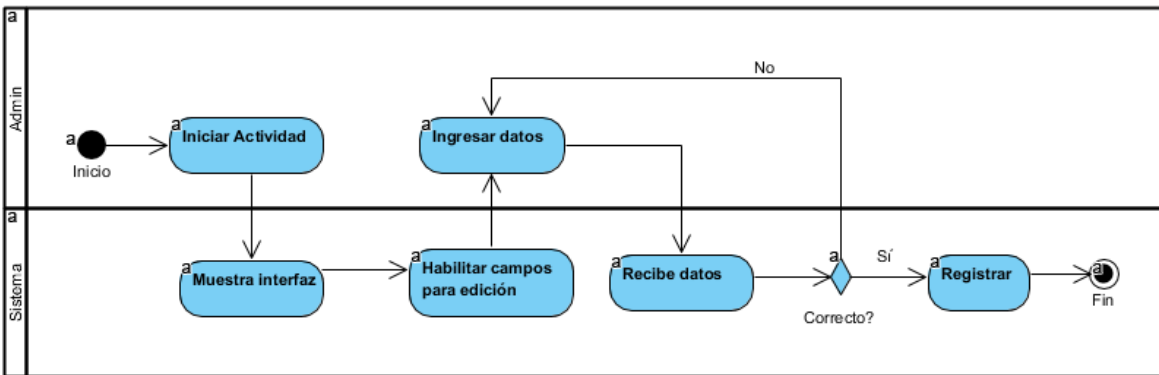
- **DA Gestionar perfiles**
  - **DACU M. Consultar**



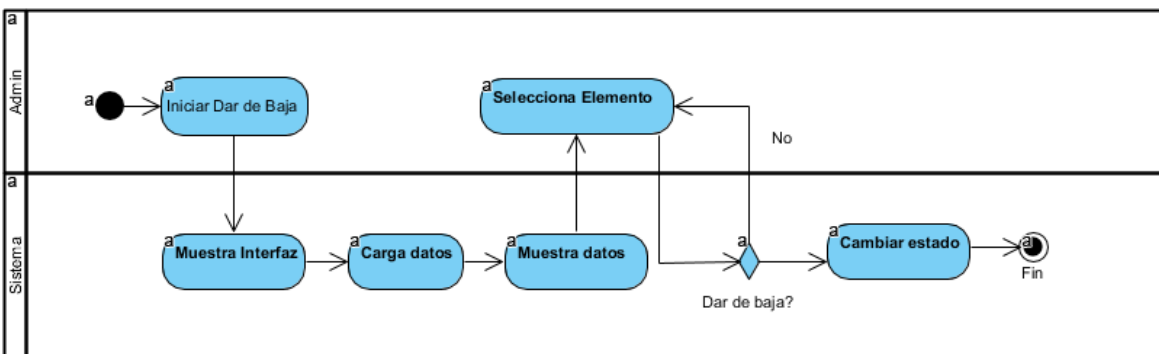
▪ DACU M. Agregar



▪ DACU M. Editar



▪ DACU M. Dar de baja



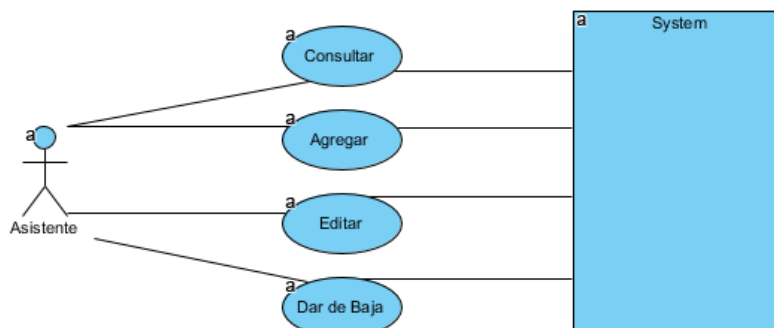
▪ DT Gestionar perfiles

▪ DACU W. Consultar:

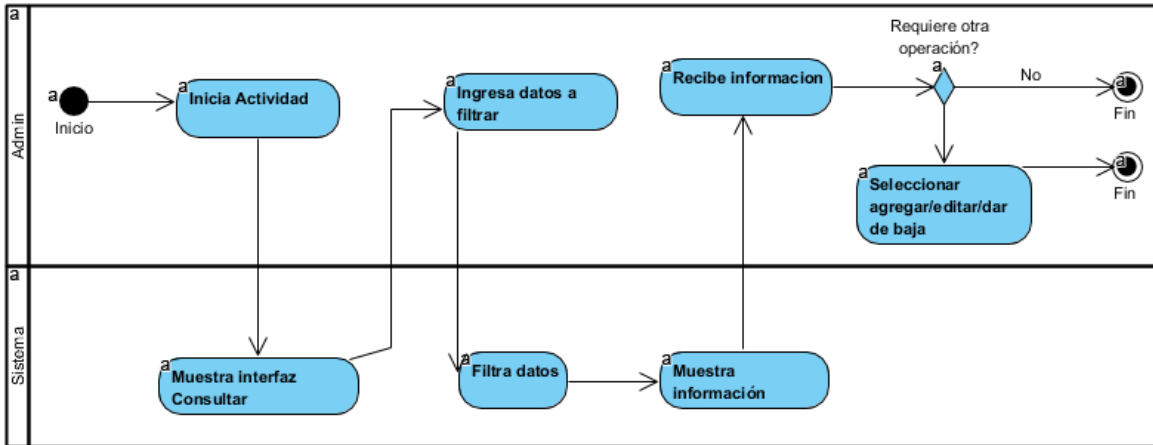
1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)

6. Fin.
  7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.
- **DACU W. Agregar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El asistente ingresa los datos requeridos.
    3. El sistema recibe los datos y comprueba si están bien ingresados.
    4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    5. Fin.
    6. El asistente vuelve a ingresar datos.
  - **DACU W. Editar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema habilita los campos disponibles para su edición.
    3. El asistente ingresa los datos requeridos.
    4. El sistema recibe los datos y comprueba si están bien ingresados.
    5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    6. Fin.
    7. El asistente vuelve a ingresar datos.
  - **DACU W. Dar de baja**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema carga los datos iniciales y los muestra.
    3. El paciente selecciona un elemento y elimina
    4. El sistema da de baja al elemento seleccionado
    5. Fin
- **W. Gestionar Cargo**
    - **DCU Gestionar Cargo**

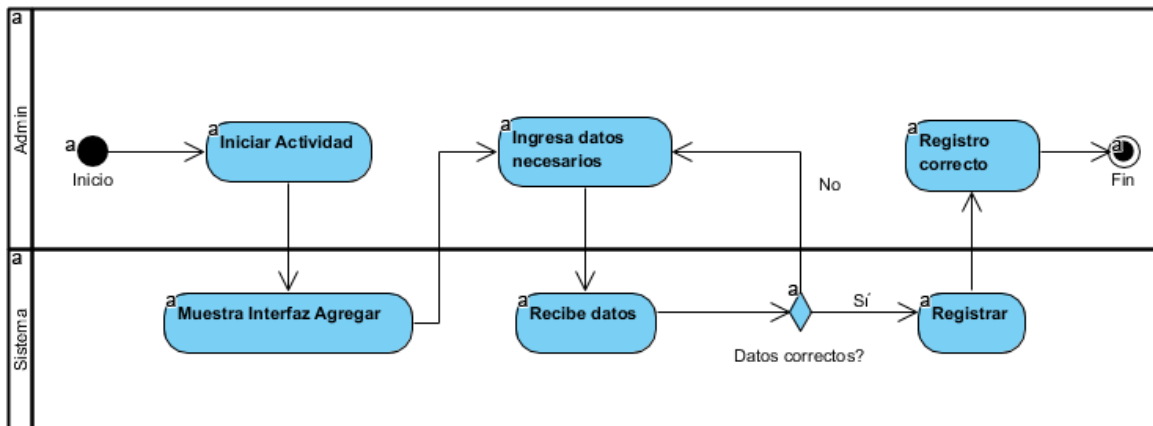
DIAGRAMA DE CASO DE USO - Gestionar Cargo



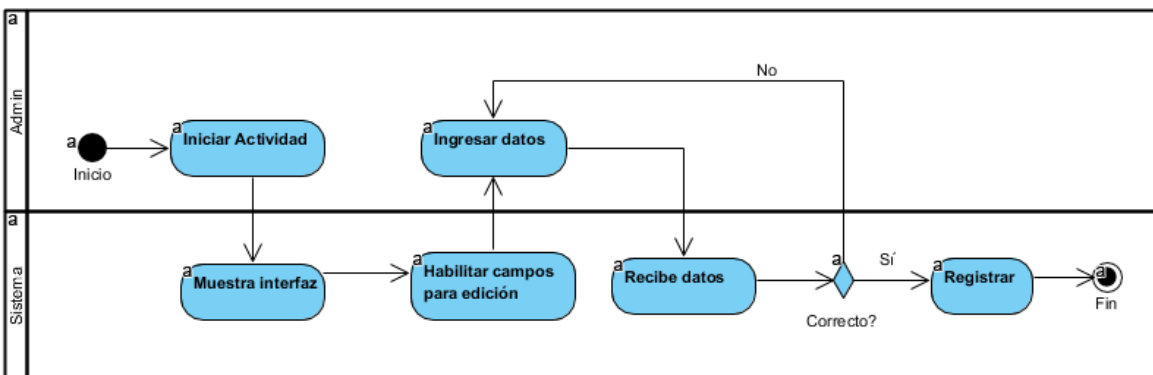
- DA Gestionar Cargo
  - DACU M. Consultar



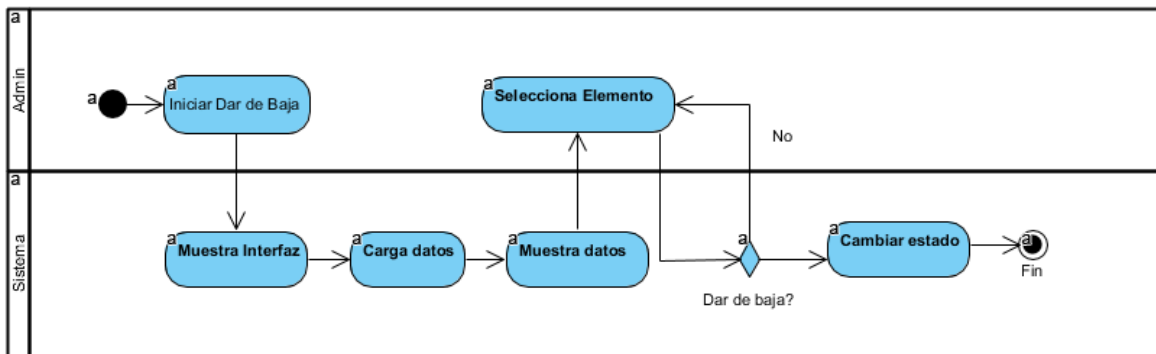
- DACU M. Agregar



- DACU M. Editar



- **DACU M. Dar de baja**



- **DT Gestionar Cargo**

- **DACU W. Consultar:**

1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)
6. Fin.
7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.

- **DACU W. Agregar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El asistente ingresa los datos requeridos.
3. El sistema recibe los datos y comprueba si están bien ingresados.
4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
5. Fin.
6. El asistente vuelve a ingresar datos.

- **DACU W. Editar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El sistema habilita los campos disponibles para su edición.
3. El asistente ingresa los datos requeridos.
4. El sistema recibe los datos y comprueba si están bien ingresados.
5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
6. Fin.
7. El asistente vuelve a ingresar datos.

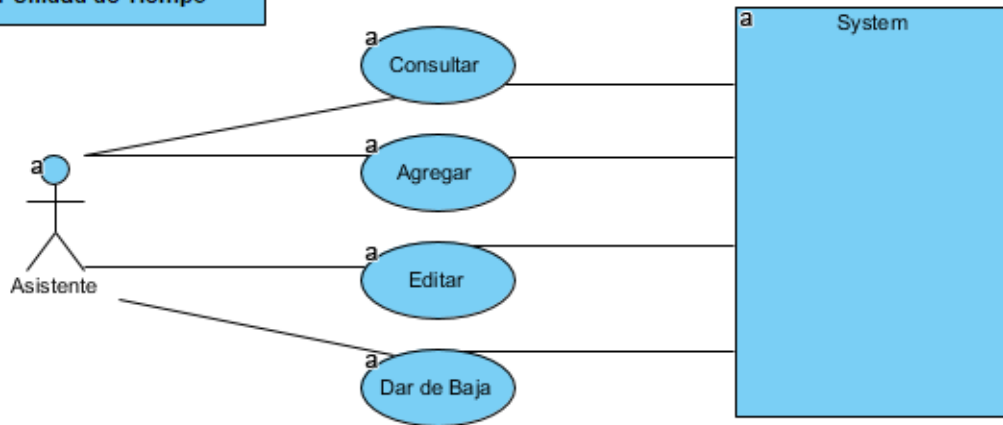
- **DACU W. Dar de baja**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.

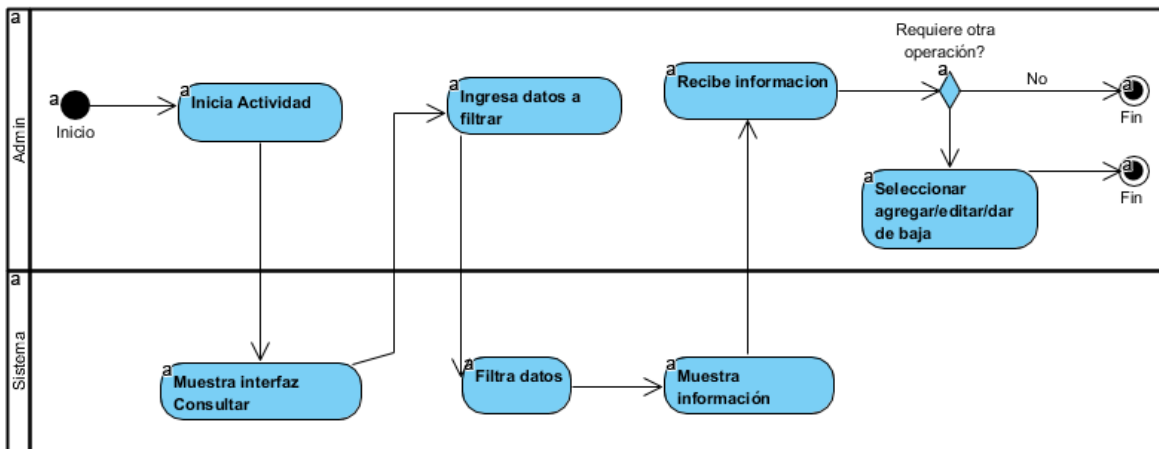
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento y elimina
4. El sistema da de baja al elemento seleccionado
5. Fin

- **W. Gestionar Unidad de Tiempo**
  - **DCU Gestionar Unidad de Tiempo**

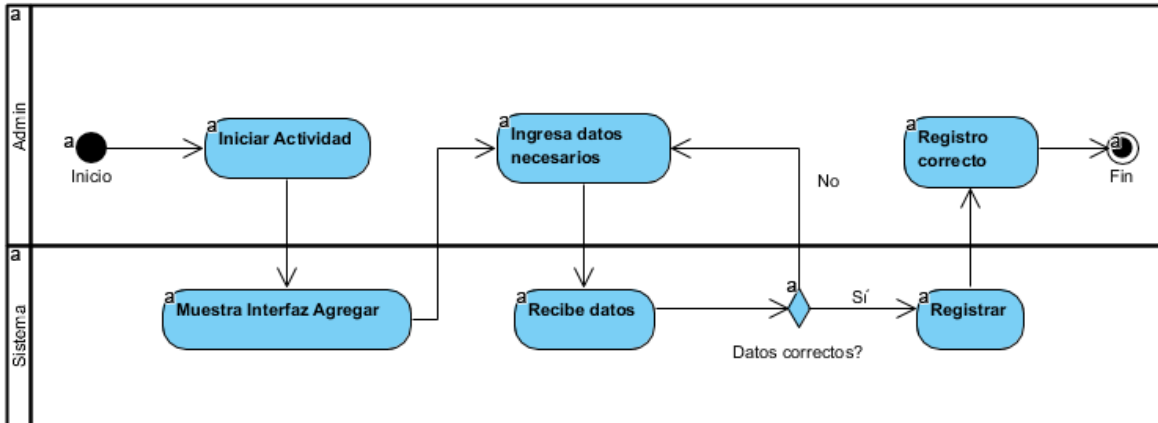
**DIAGRAMA DE CASO DE USO - Gestionar Unidad de Tiempo**



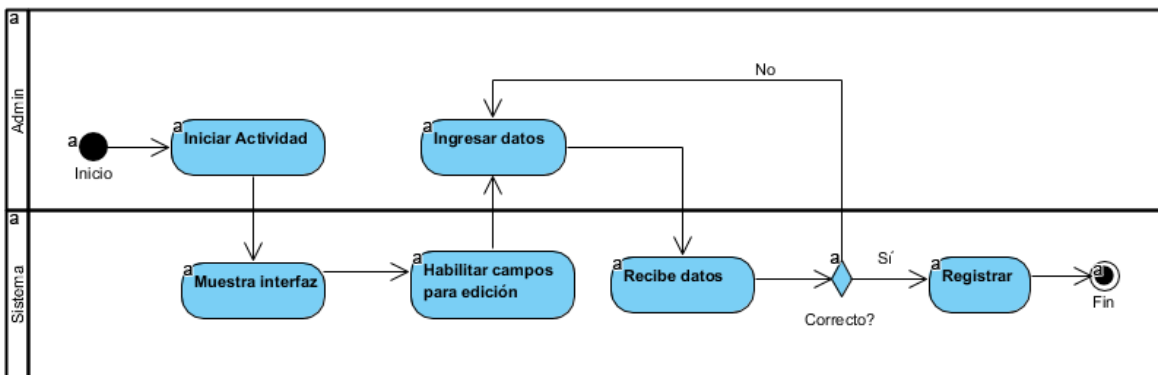
- **DA Gestionar Unidad de Tiempo**
  - **DACU M. Consultar**



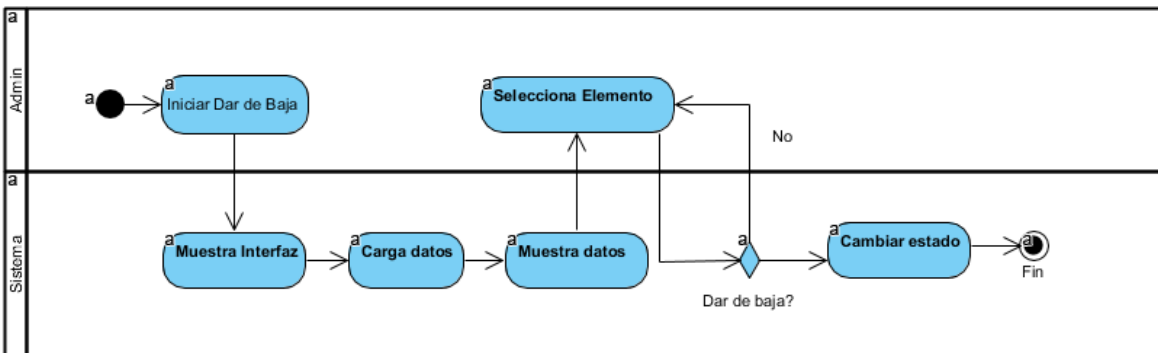
▪ DACU M. Agregar



▪ DACU M. Editar



▪ DACU M. Dar de baja



▪ DT Gestionar Unidad de Tiempo

▪ DACU W. Consultar:

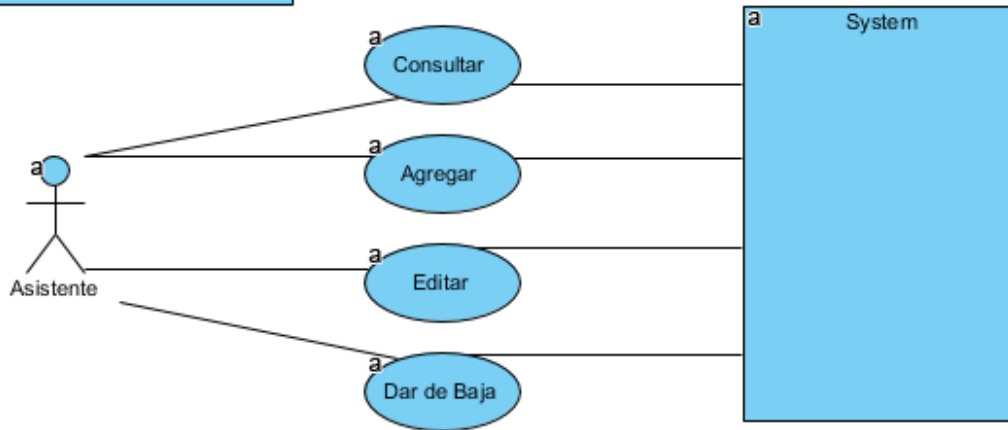
1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)



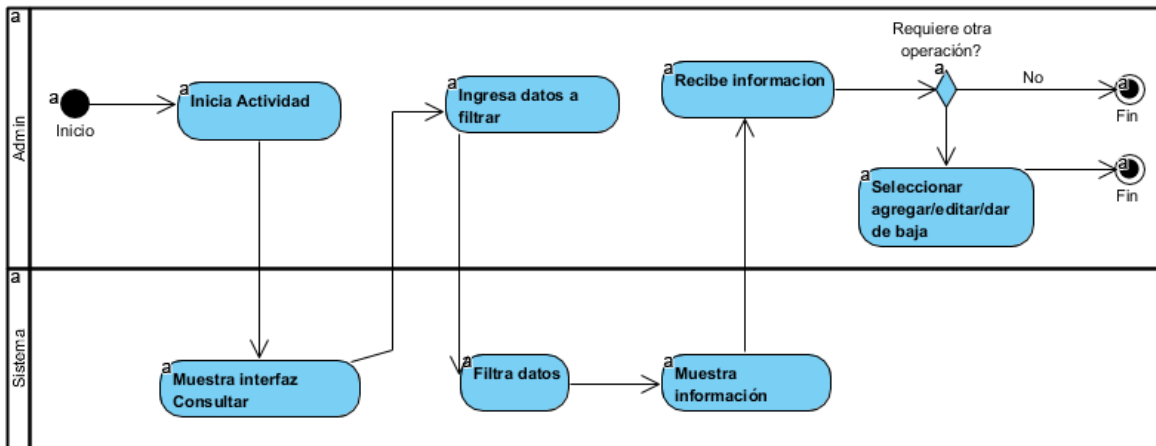
6. Fin.
  7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.
- **DACU W. Agregar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El asistente ingresa los datos requeridos.
    3. El sistema recibe los datos y comprueba si están bien ingresados.
    4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    5. Fin.
    6. El asistente vuelve a ingresar datos.
  - **DACU W. Editar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema habilita los campos disponibles para su edición.
    3. El asistente ingresa los datos requeridos.
    4. El sistema recibe los datos y comprueba si están bien ingresados.
    5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    6. Fin.
    7. El asistente vuelve a ingresar datos.
  - **DACU W. Dar de baja**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema carga los datos iniciales y los muestra.
    3. El paciente selecciona un elemento y elimina
    4. El sistema da de baja al elemento seleccionado
    5. Fin

- **W. Gestionar frecuencia**
  - **DCU Gestionar frecuencia**

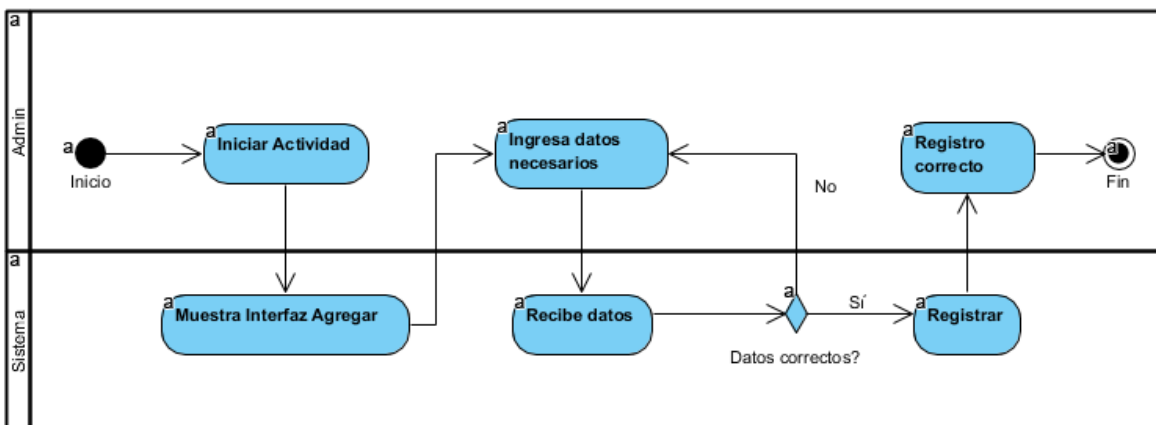
**DIAGRAMA DE CASO DE USO - Gestionar Frecuencia**



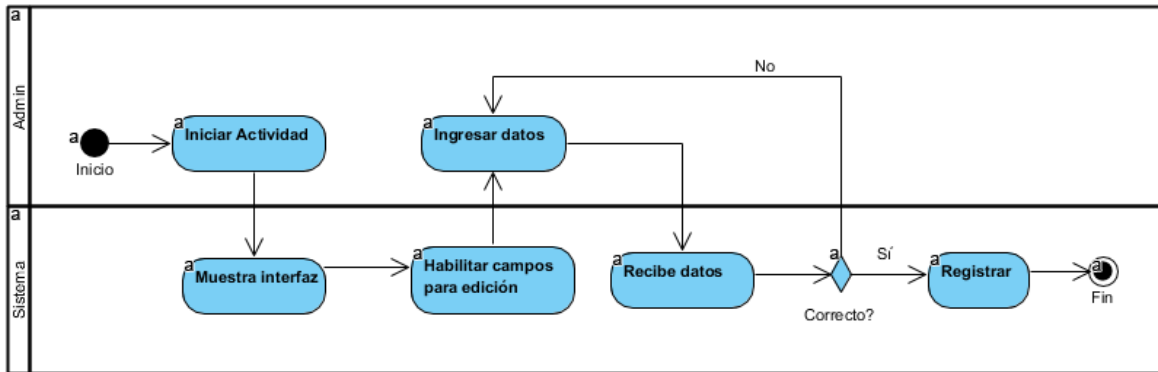
- **DA Gestionar frecuencia**
  - **DACU M. Consultar**



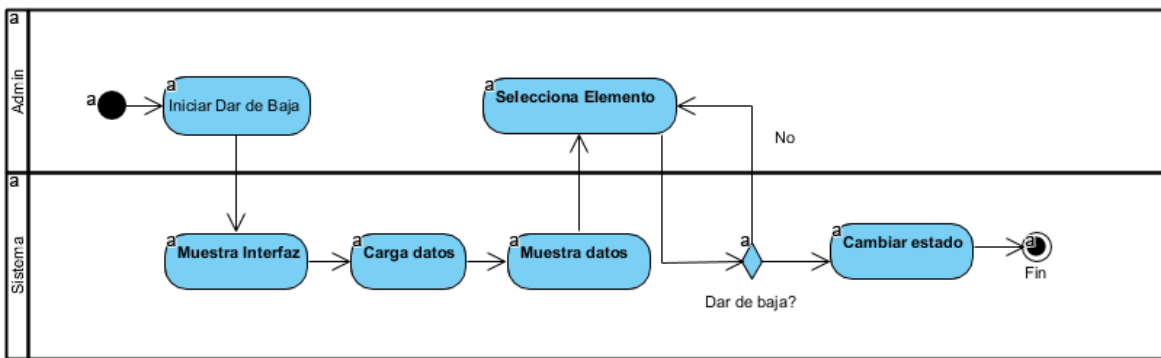
- **DACU M. Agregar**



- **DACU M. Editar**



- **DACU M. Dar de baja**



- **DT Gestionar frecuencia**

- **DACU W. Consultar:**

1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)
6. Fin.
7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.

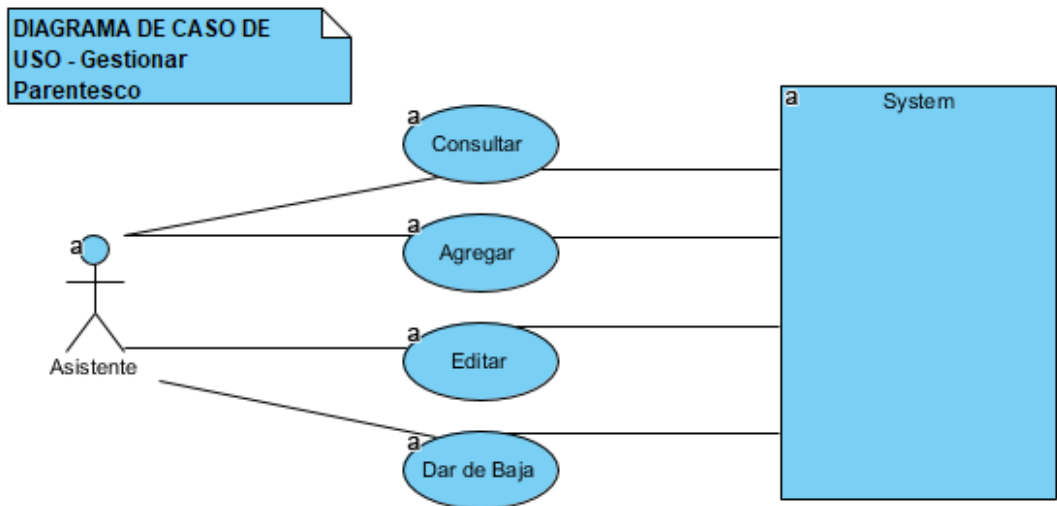
- **DACU W. Agregar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El asistente ingresa los datos requeridos.
3. El sistema recibe los datos y comprueba si están bien ingresados.
4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
5. Fin.
6. El asistente vuelve a ingresar datos.

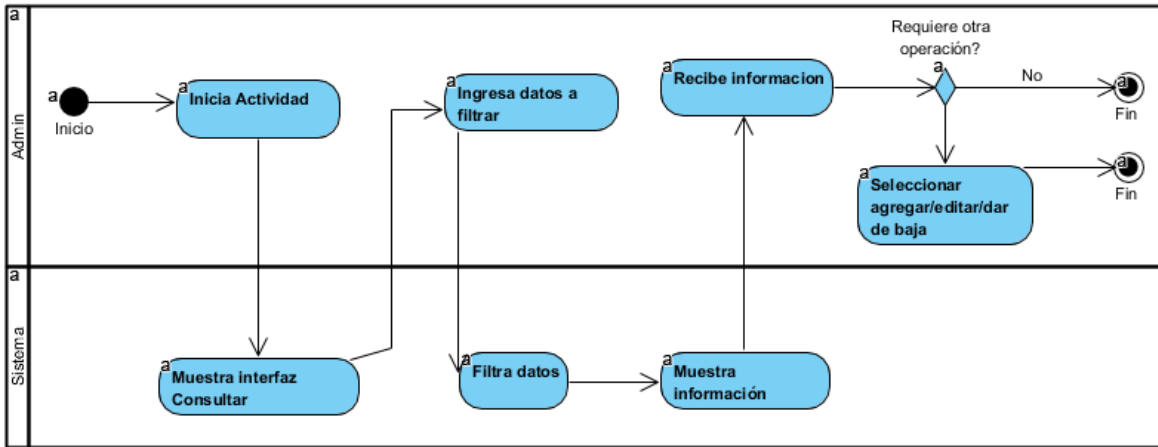
- **DACU W. Editar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
  2. El sistema habilita los campos disponibles para su edición.
  3. El asistente ingresa los datos requeridos.
  4. El sistema recibe los datos y comprueba si están bien ingresados.
  5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
  6. Fin.
  7. El asistente vuelve a ingresar datos.
- **DACU W. Dar de baja**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema carga los datos iniciales y los muestra.
    3. El paciente selecciona un elemento y elimina
    4. El sistema da de baja al elemento seleccionado
    5. Fin

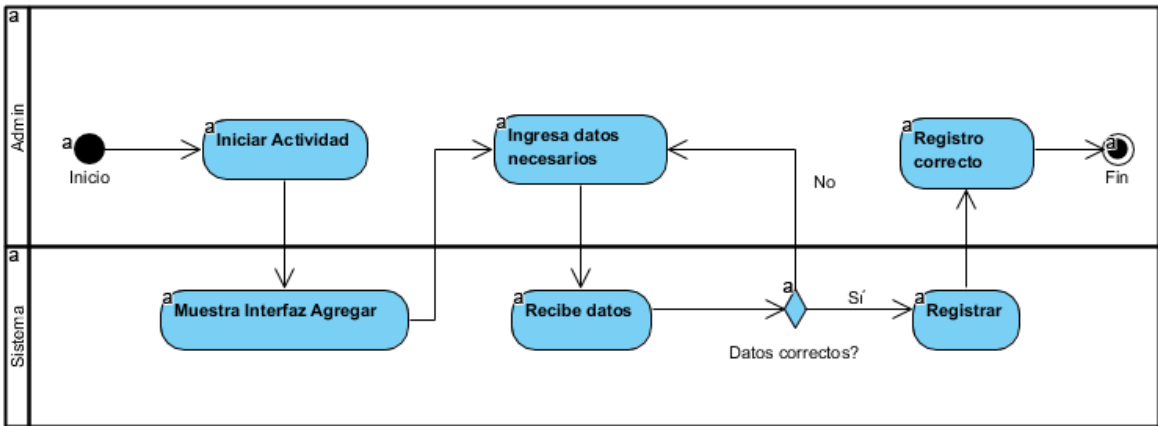
- **W. Gestionar Parentesco**
  - **DCU Gestionar Parentesco**



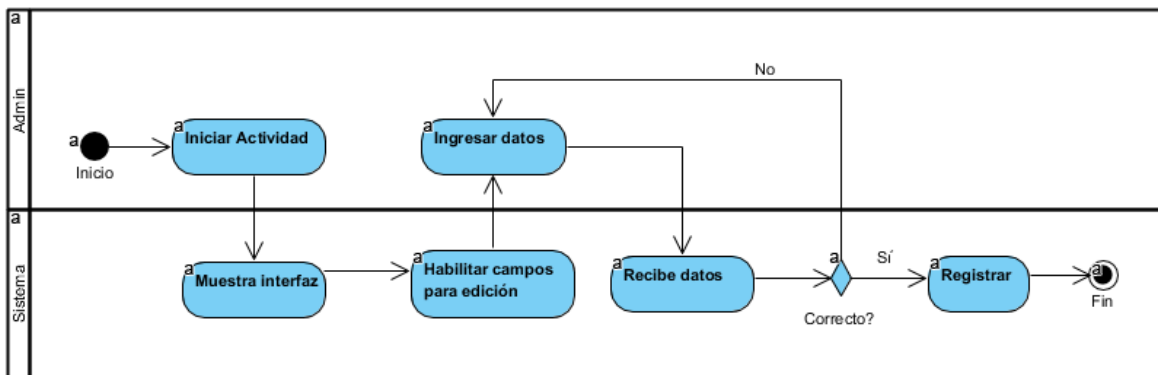
- DA Gestionar Parentesco
  - DACU M. Consultar



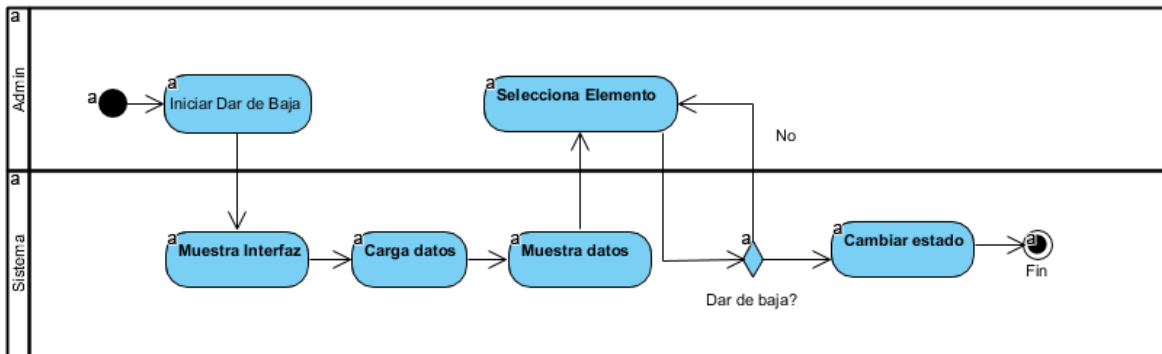
- DACU M. Agregar



- DACU M. Editar



- **DACU M. Dar de baja**



- **DT Gestionar Parentesco**

- **DACU W. Consultar:**

1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)
6. Fin.
7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.

- **DACU W. Agregar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El asistente ingresa los datos requeridos.
3. El sistema recibe los datos y comprueba si están bien ingresados.
4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
5. Fin.
6. El asistente vuelve a ingresar datos.

- **DACU W. Editar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El sistema habilita los campos disponibles para su edición.
3. El asistente ingresa los datos requeridos.
4. El sistema recibe los datos y comprueba si están bien ingresados.
5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
6. Fin.
7. El asistente vuelve a ingresar datos.

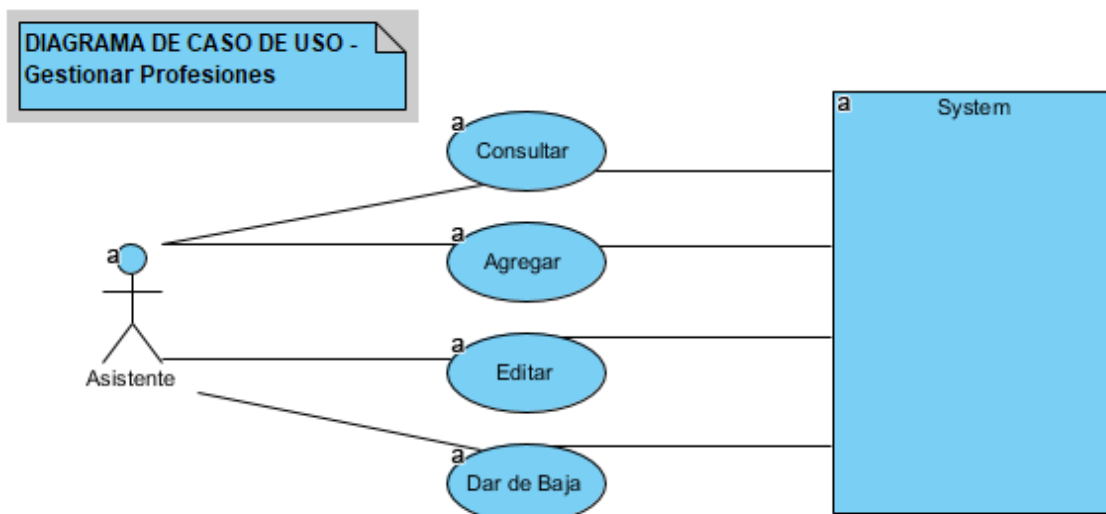
- **DACU W. Dar de baja**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.

2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento y elimina
4. El sistema da de baja al elemento seleccionado
5. Fin

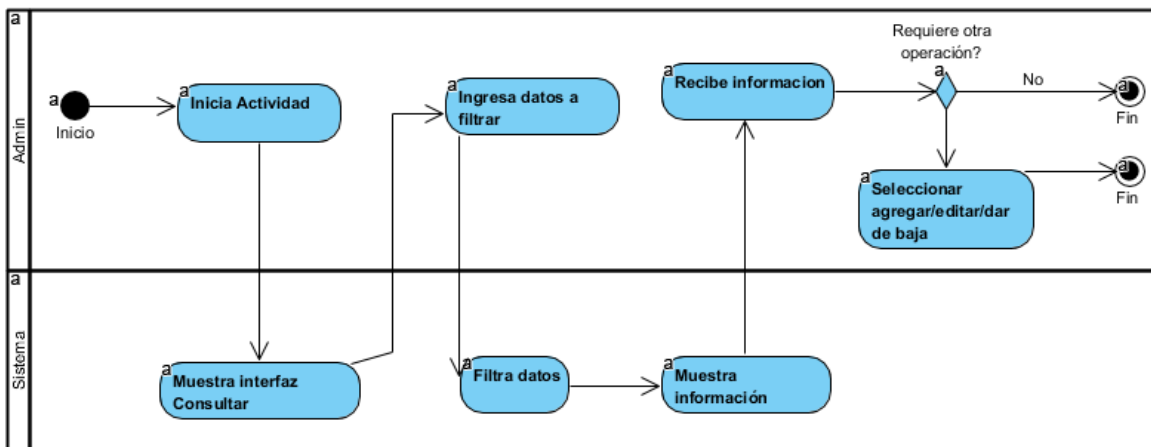
○ **W. Gestionar Profesiones**

▪ **DCU Gestionar Profesiones**

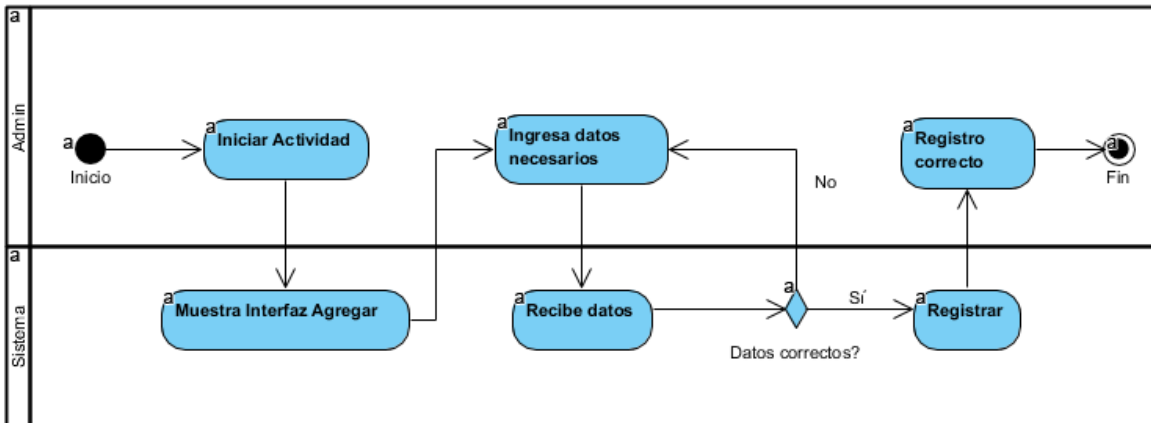


▪ **DA Gestionar Profesiones**

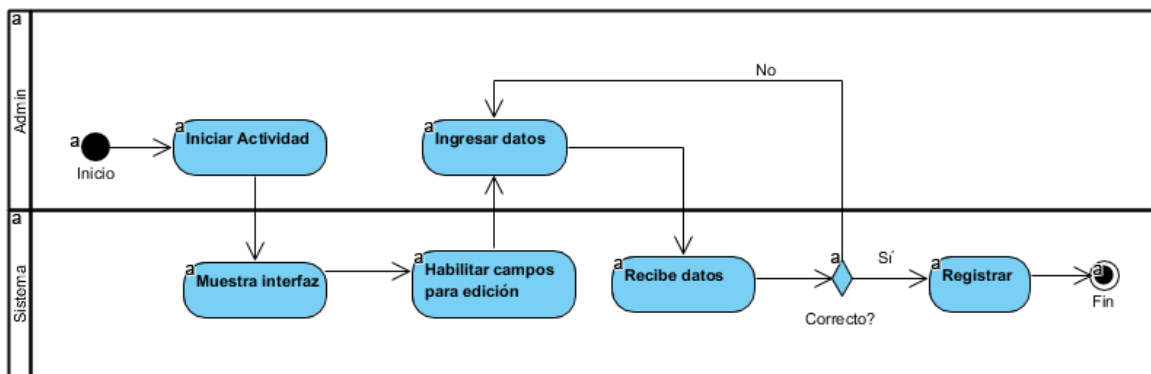
▪ **DACU M. Consultar**



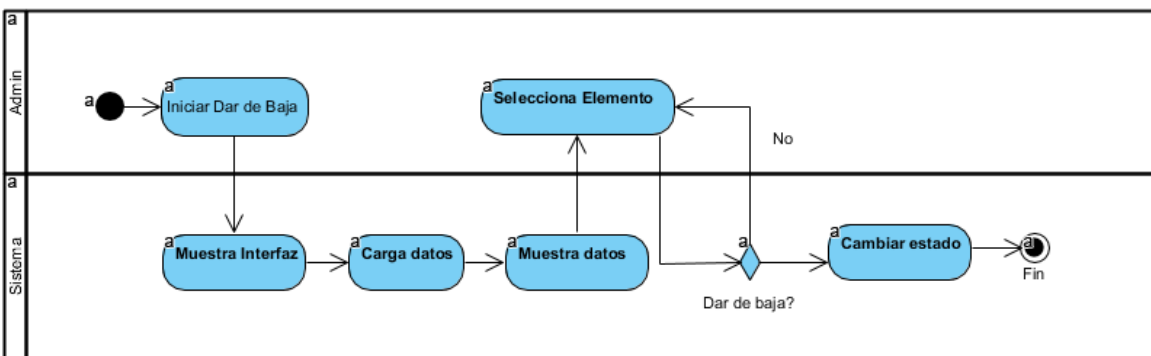
▪ DACU M. Agregar



▪ DACU M. Editar



▪ DACU M. Dar de baja



▪ DT Gestionar Profesiones

▪ DACU W. Consultar:

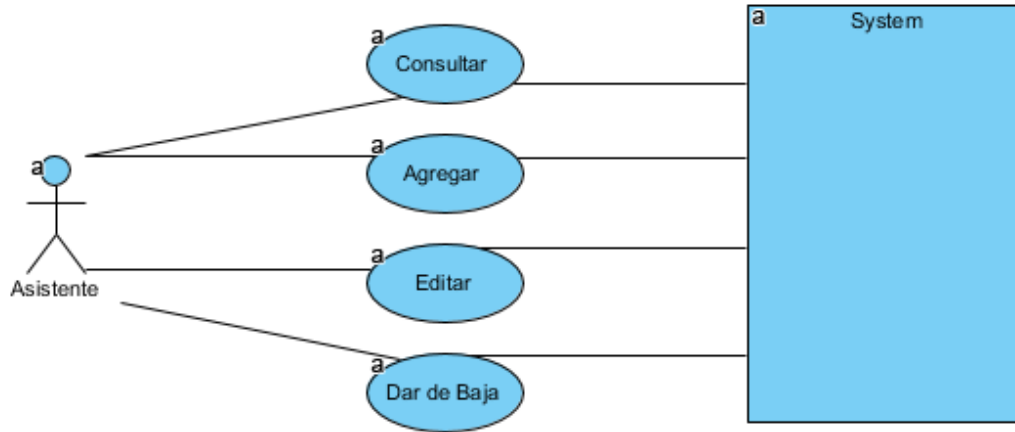
1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)



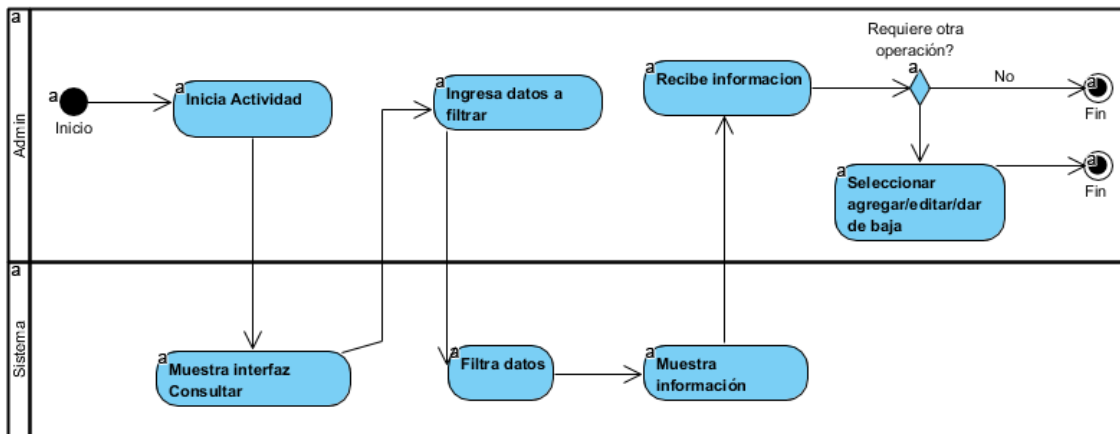
6. Fin.
  7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.
- **DACU W. Agregar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El asistente ingresa los datos requeridos.
    3. El sistema recibe los datos y comprueba si están bien ingresados.
    4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    5. Fin.
    6. El asistente vuelve a ingresar datos.
  - **DACU W. Editar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema habilita los campos disponibles para su edición.
    3. El asistente ingresa los datos requeridos.
    4. El sistema recibe los datos y comprueba si están bien ingresados.
    5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    6. Fin.
    7. El asistente vuelve a ingresar datos.
  - **DACU W. Dar de baja**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema carga los datos iniciales y los muestra.
    3. El paciente selecciona un elemento y elimina
    4. El sistema da de baja al elemento seleccionado
    5. Fin

- **W. Gestionar Acceso**
  - **DCU Gestionar Acceso**

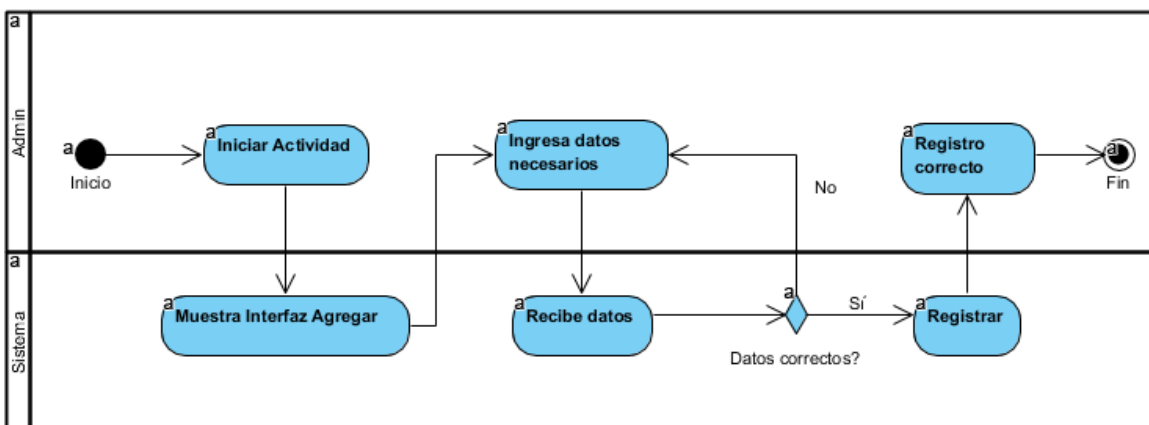
**DIAGRAMA DE CASO DE USO - Gestionar Acceso**



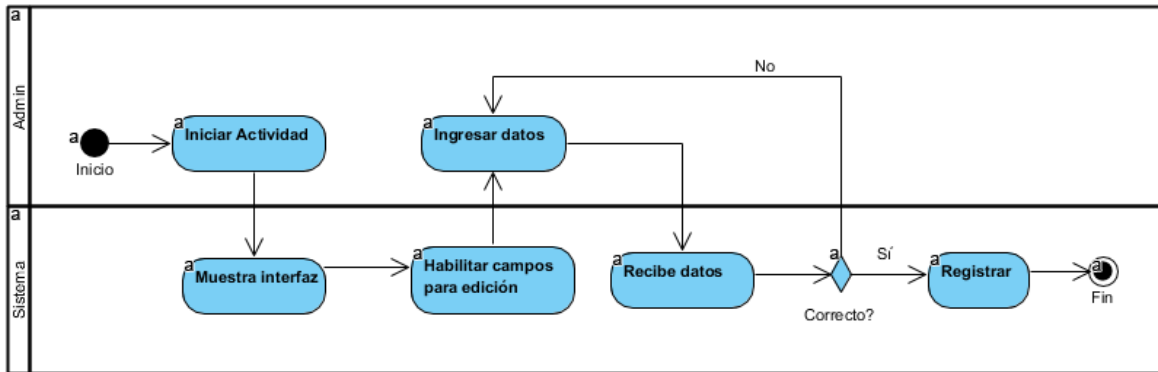
- **DA Gestionar Acceso**
  - **DACU M. Consultar**



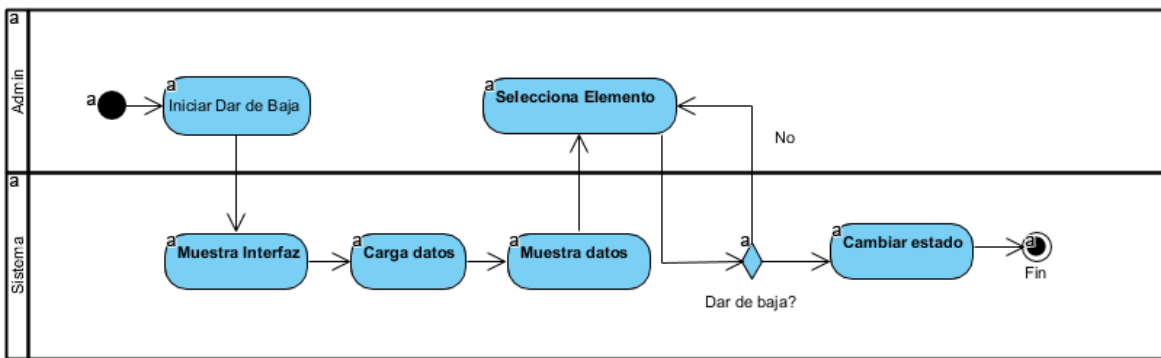
- **DACU M. Agregar**



- **DACU M. Editar**



- **DACU M. Dar de baja**



- **DT Gestionar Acceso**

- **DACU W. Consultar:**

1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)
6. Fin.
7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.

- **DACU W. Agregar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El asistente ingresa los datos requeridos.
3. El sistema recibe los datos y comprueba si están bien ingresados.
4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
5. Fin.
6. El asistente vuelve a ingresar datos.

- **DACU W. Editar**

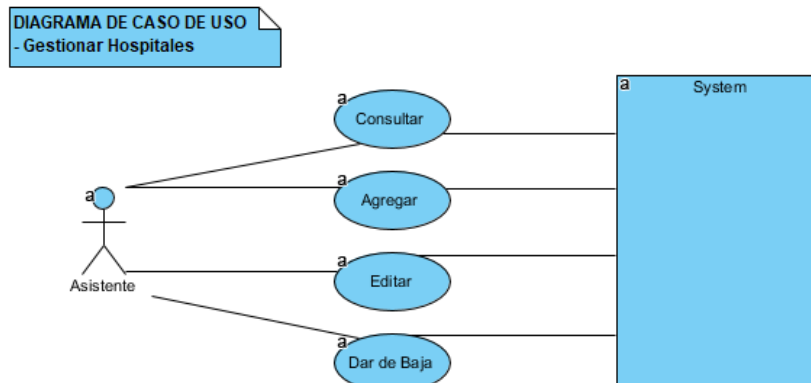
1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El sistema habilita los campos disponibles para su edición.
3. El asistente ingresa los datos requeridos.
4. El sistema recibe los datos y comprueba si están bien ingresados.
5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
6. Fin.
7. El asistente vuelve a ingresar datos.

- **DACU W. Dar de baja**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento y elimina
4. El sistema da de baja al elemento seleccionado
5. Fin

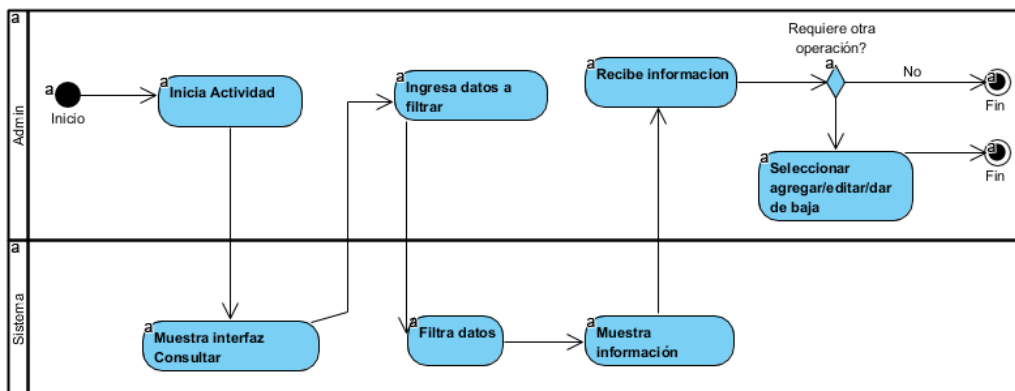
- **W. Gestionar Hospitales**

- **DCU Gestionar Hospitales**

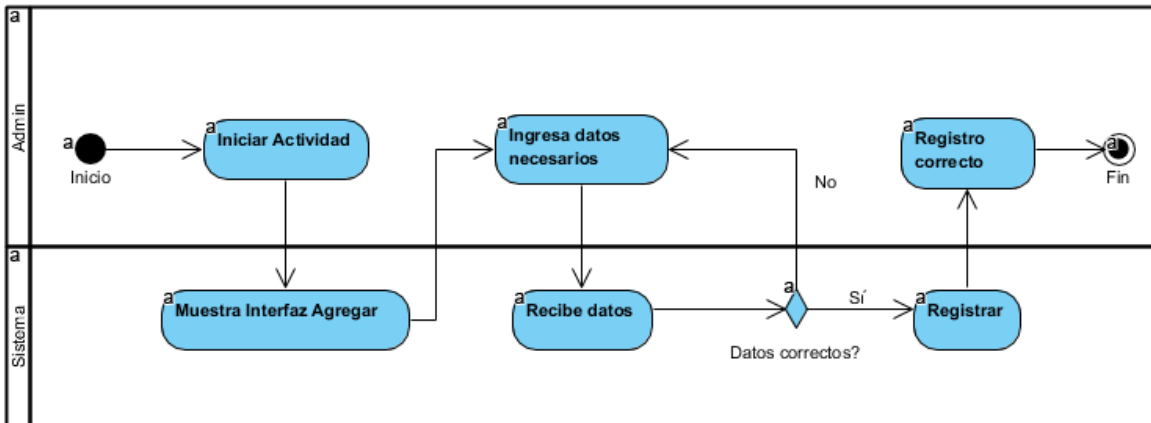


- **DA Gestionar Hospitales**

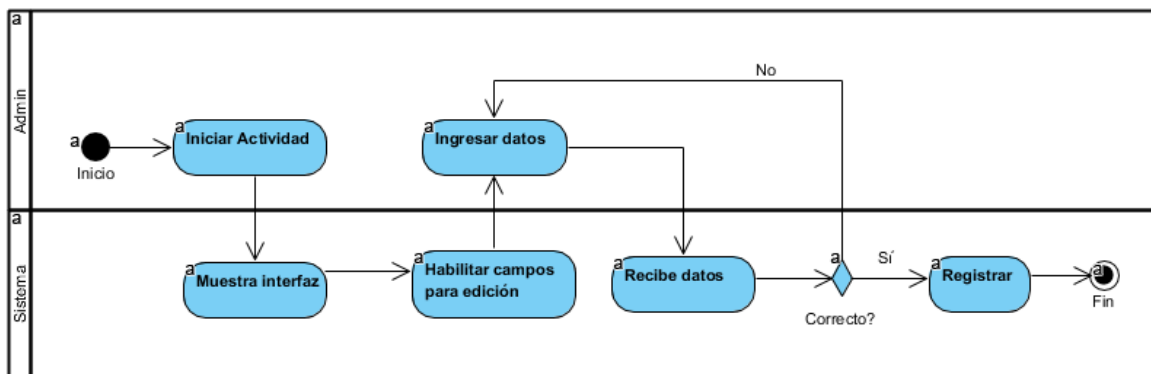
- **DACU M. Consultar**



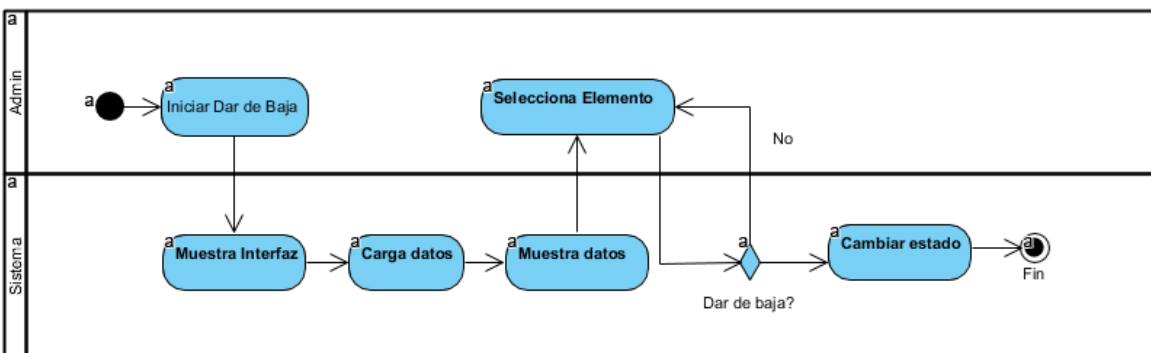
▪ DACU M. Agregar



▪ DACU M. Editar



▪ DACU M. Dar de baja



▪ DT Gestionar Hospitales

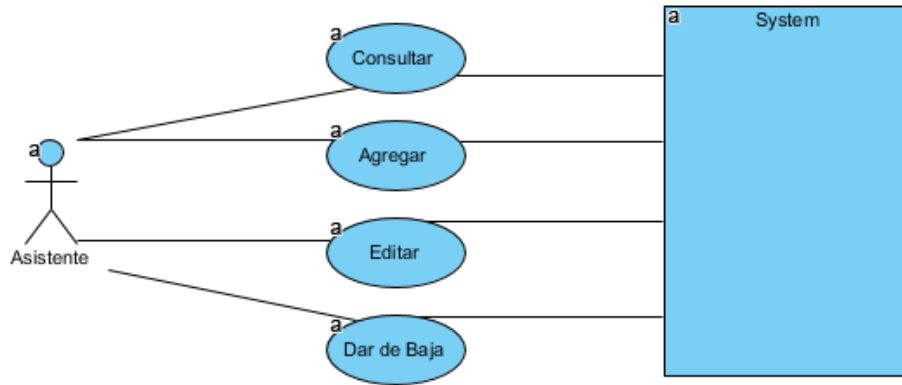
▪ DACU W. Consultar:

1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)

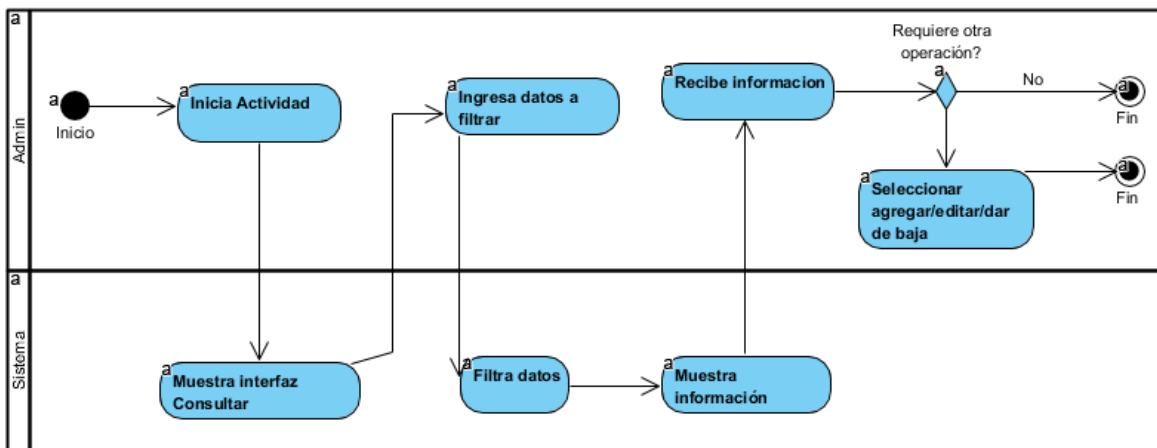
6. Fin.
  7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.
- **DACU W. Agregar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El asistente ingresa los datos requeridos.
    3. El sistema recibe los datos y comprueba si están bien ingresados.
    4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    5. Fin.
    6. El asistente vuelve a ingresar datos.
  - **DACU W. Editar**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema habilita los campos disponibles para su edición.
    3. El asistente ingresa los datos requeridos.
    4. El sistema recibe los datos y comprueba si están bien ingresados.
    5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
    6. Fin.
    7. El asistente vuelve a ingresar datos.
  - **DACU W. Dar de baja**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema carga los datos iniciales y los muestra.
    3. El paciente selecciona un elemento y elimina
    4. El sistema da de baja al elemento seleccionado
    5. Fin

- **W. Gestionar Farmaco**
  - **DCU Gestionar Farmaco**

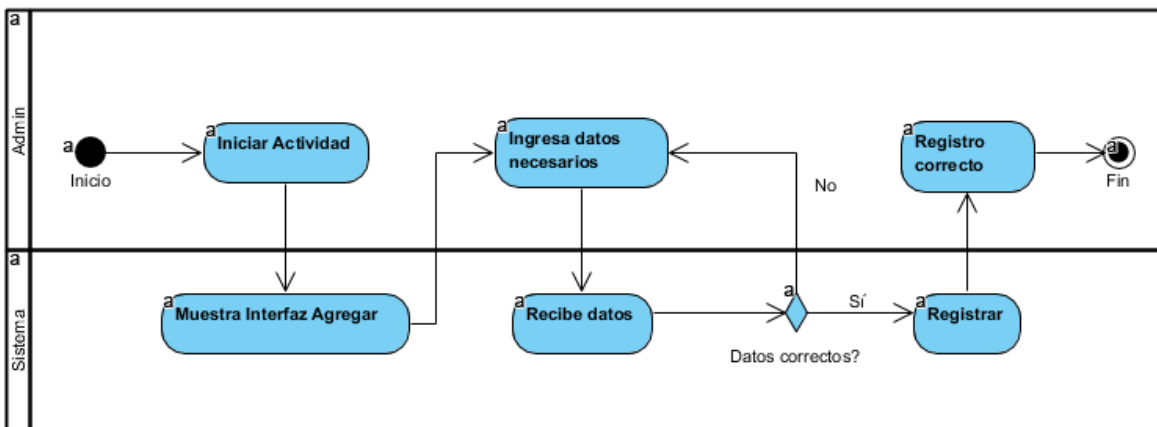
DIAGRAMA DE CASO DE USO - Gestionar Farmaco



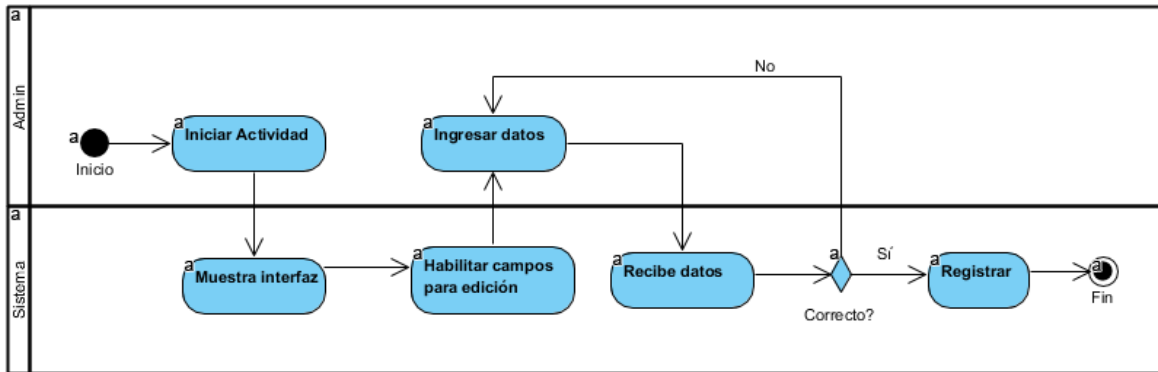
- **DA Gestionar Farmaco**
  - **DACU M. Consultar**



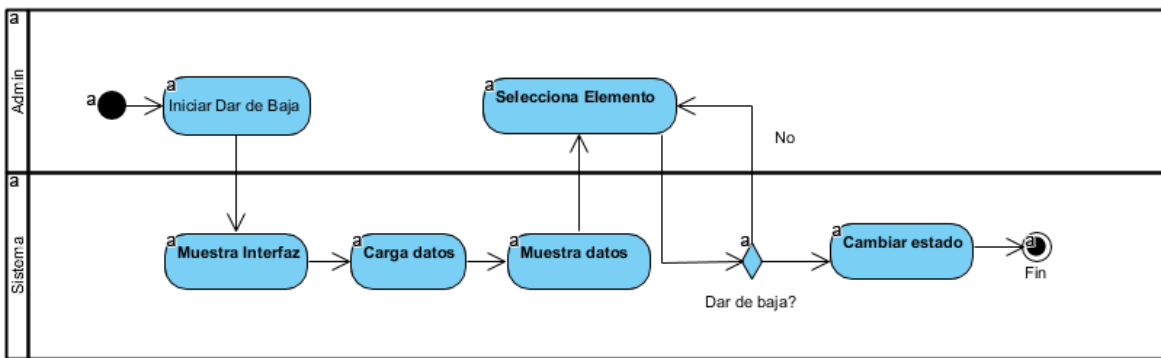
- **DACU M. Agregar**



- **DACU M. Editar**



- **DACU M. Dar de baja**



- **DT Gestionar Farmaco**

- **DACU W. Consultar:**

1. El asistente inicia la actividad y el sistema muestra interfaz respectiva.
2. El sistema carga los datos iniciales y los muestra.
3. El paciente selecciona un elemento.
4. El sistema devuelve la información y la muestra.
5. ¿El asistente tiene otra acción por realizar? No (paso 7)
6. Fin.
7. El asistente selecciona agregar nuevo elemento, editar elemento o eliminar.

- **DACU W. Agregar**

1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
2. El asistente ingresa los datos requeridos.
3. El sistema recibe los datos y comprueba si están bien ingresados.
4. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
5. Fin.
6. El asistente vuelve a ingresar datos.

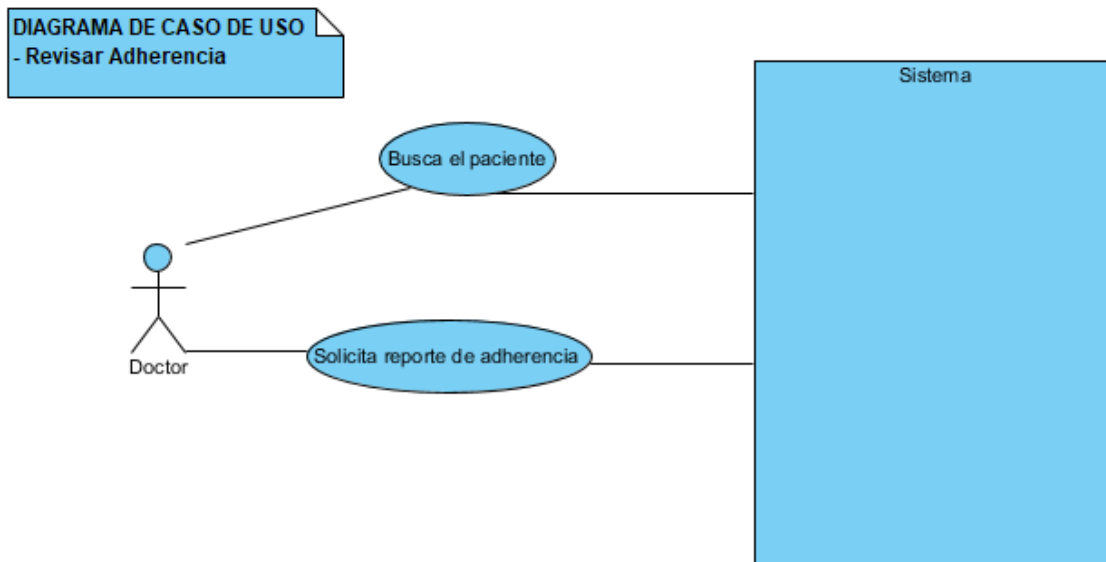
- **DACU W. Editar**



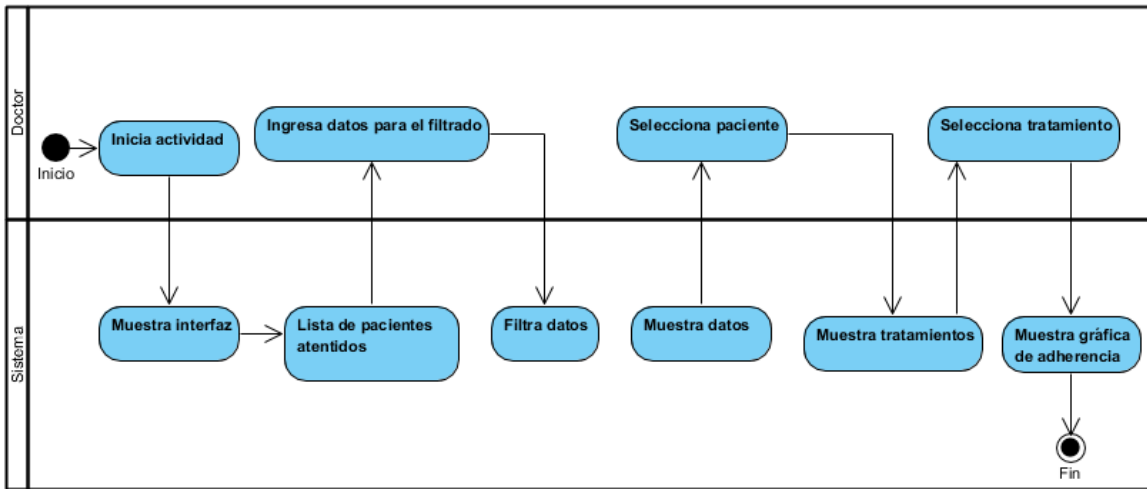
1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
  2. El sistema habilita los campos disponibles para su edición.
  3. El asistente ingresa los datos requeridos.
  4. El sistema recibe los datos y comprueba si están bien ingresados.
  5. Si están bien ingresados, el sistema guarda los datos. Sino pasa al paso 6.
  6. Fin.
  7. El asistente vuelve a ingresar datos.
- **DACU W. Dar de baja**
    1. El asistente inicia la actividad y el sistema muestra la interfaz respectiva.
    2. El sistema carga los datos iniciales y los muestra.
    3. El paciente selecciona un elemento y elimina
    4. El sistema da de baja al elemento seleccionado
    5. Fin

○ **W. Revisar Adherencia**

- **DCU Revisar Adherencia**



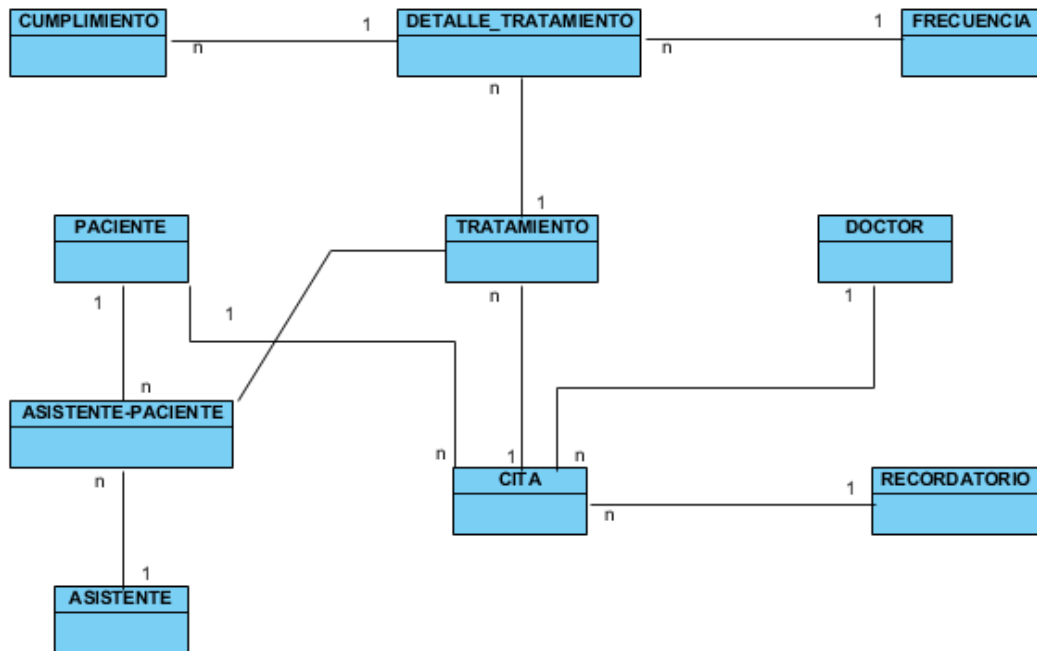
▪ **DA Revisar Adherencia**



▪ **DT Revisar Adherencia**

1. El doctor inicia la actividad y el sistema muestra la interfaz respectiva.
2. El paciente muestra la lista de pacientes atendidos por el doctor.
3. El doctor ingresa datos para el filtrado
4. El sistema filtra los datos.
5. El sistema muestra los resultados del filtrado.
6. El doctor selecciona el paciente.
7. El sistema muestra tratamientos del paciente
8. El doctor selecciona un tratamiento
9. El sistema muestra gráfica de adherencia.

**D. MODELO DE OBJETOS**



### 3.2. DESCRIPCIÓN DE LA ARQUITECTURA

- DACU M. Gestionar Paciente:
- **DACU M. Gestionar Tratamiento**
  - **DACU Consultar Tratamiento:** En esta funcionalidad el usuario podrá consultar información de un tratamiento pudiendo filtrar los datos de diferentes maneras como por ejemplo por la fecha de creación, el nombre del tratamiento, etc.
  - **DACU Agregar Tratamiento:** En esta funcionalidad el usuario podrá registrar un nuevo tratamiento para el paciente, registrar el nombre de los fármacos a consumir, cada cuánto tiempo o con qué frecuencia, por qué tiempo lo hará y si es necesario dar nota de qué doctor prescribió ese tratamiento.
  - **DACU Editar Tratamiento:** En esta funcionalidad el usuario puede editar un tratamiento bajo alguna indicación del doctor o algún factor externo del cuál debe de dar nota al momento de editarlo para dejar registro de cuándo se hizo y por qué.
  - **DACU Dar de baja Tratamiento:** En esta funcionalidad el usuario puede dar de baja a un tratamiento por indicación del médico o algún factor externo del cuál debe de dar nota al momento de dar de baja a un tratamiento. También el tratamiento será dado de baja automáticamente al terminar.
- **DACU M. Gestionar Cita médica:**
  - **DACU Consultar Cita Médica:** En esta funcionalidad el usuario podrá consultar información de una cita como la fecha, el lugar de la cita, entre otros datos; pudiendo filtrar los datos de diferentes maneras como fecha de registro o fecha de la cita, entre otros.
  - **DACU Agregar Cita Médica:** En esta funcionalidad el usuario podrá registrar una cita médica para el paciente donde debe denotar información relevante como la fecha de la cita, el lugar de la cita, el médico, e incluso ubicar el lugar de la cita en un mapa (opcional).
  - **DACU Editar Cita Médica:** En esta funcionalidad el usuario puede editar una cita médica como la fecha o el lugar, el motivo por el cuál sea editada la cita debe añadirse también para dejar registro de cuándo se hizo y por qué.
  - **DACU Dar de baja Cita Médica:** En esta funcionalidad el usuario puede dar de baja a una cita por indicación del médico o algún factor externo del cuál debe de dar nota al momento de dar de baja a la cita. También la cita será dada de baja automáticamente al pasar el día de esta.
  - **DACU Ubicar Cita Médica en el Mapa:** Esta funcionalidad es un adicional de la anterior donde se puede ubicar el lugar de la cita médica en el mapa y la aplicación lo interpretará como coordenadas que serán guardadas en la base de datos. Esta funcionalidad es adicional y no es indispensable para agregar una cita Médica.
- **DACU Buscar Farmacia:** Esta funcionalidad es un adicional para ubicar las farmacias más cercanas cuando el usuario se encuentre en algún lugar

desconocido o haya viajado justo con el paciente y necesite comprar medicamentos. La app móvil debe mostrar la ubicación exacta del usuario mediante el GPS y el API de Google Maps, además debe filtrar los indicadores de lugares para reducirlo a farmacias cercanas.

- **DACU M. Gestionar Contacto:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información del contacto que está asignado al paciente y sus diferentes datos que sean visibles para el asistente.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar los datos de un nuevo contacto que necesite ser enlazado con el paciente.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información habilitada para esta acción por parte del asistente.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja (eliminación lógica – cambio de estado) a un contacto del paciente.
- **DACU M. Añadir un contacto a un paciente:** en esta funcionalidad el usuario podrá enlazar un contacto a un paciente. En caso no esté registrado, puede añadirlo como un nuevo contacto y enlazarlo.
- **DACU M. Añadir una enfermedad a un paciente:** en esta funcionalidad el usuario podrá enlazar una enfermedad a un paciente. En caso no esté registrada se le comunicará a al administrado y este podrá añadirla. El asistente (usuario) NO podrá añadir una enfermedad.
- **DACU M. Gestionar cumplimiento:** En esta funcionalidad el asistente podrá registrar si el paciente tomó la pastilla en el momento que debería. Eso se facilita por la notificación push que se implementará.
- **DACU W. Gestionar Usuarios:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los usuarios.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo usuario al que puede añadirle un login y una clave para el ingreso al sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un usuario. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un usuario denegando el ingreso al sistema por el cambio de estado de Activo a Inactivo.
- **DACU W. Gestionar Perfiles:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los perfiles en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo perfil ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un perfil. El sistema debe habilitar todos los campos disponibles para su edición.

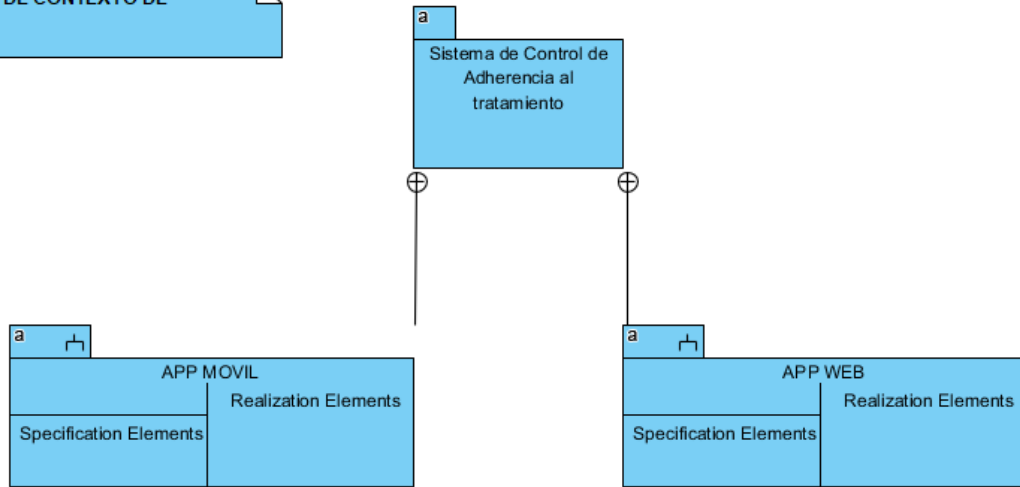
- **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un perfil cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Cargo:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los cargos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo cargo ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un cargo. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un cargo cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Unidad de tiempo:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de las Unidades de Tiempo en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar una nueva unidad de tiempo ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de una unidad de tiempo. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja una unidad de tiempo cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Frecuencia:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de las Frecuencias en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar una nueva frecuencia ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de una frecuencia. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja una frecuencia cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Parentesco:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los parentescos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo parentesco ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un parentesco. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un parentesco cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Profesiones:**

- **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de las profesiones en detalle.
- **DACU Agregar:** En esta funcionalidad el usuario podrá registrar una nueva profesión ingresando los datos requeridos por el sistema.
- **DACU Editar:** En esta funcionalidad el usuario podrá editar información de una profesión perfil. El sistema debe habilitar todos los campos disponibles para su edición.
- **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja una profesión cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Acceso:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los accesos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo acceso ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un acceso. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un acceso cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Hospital:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los hospitales en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo hospital ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un hospital. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un hospital cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Fármaco:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los fármacos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo fármaco ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un fármaco. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un fármaco cambiando su estado de Activo a Inactivo.
- **DACU W. Revisar Adherencia:** en esta funcionalidad el usuario doctor podrá ver el progreso de su paciente respecto a cómo está llevando su tratamiento una vez registrado. Esto se presentará a forma de gráficas en vista web para ser más cómoda su revisión.

## 4. CAPITULO IV: ANÁLISIS

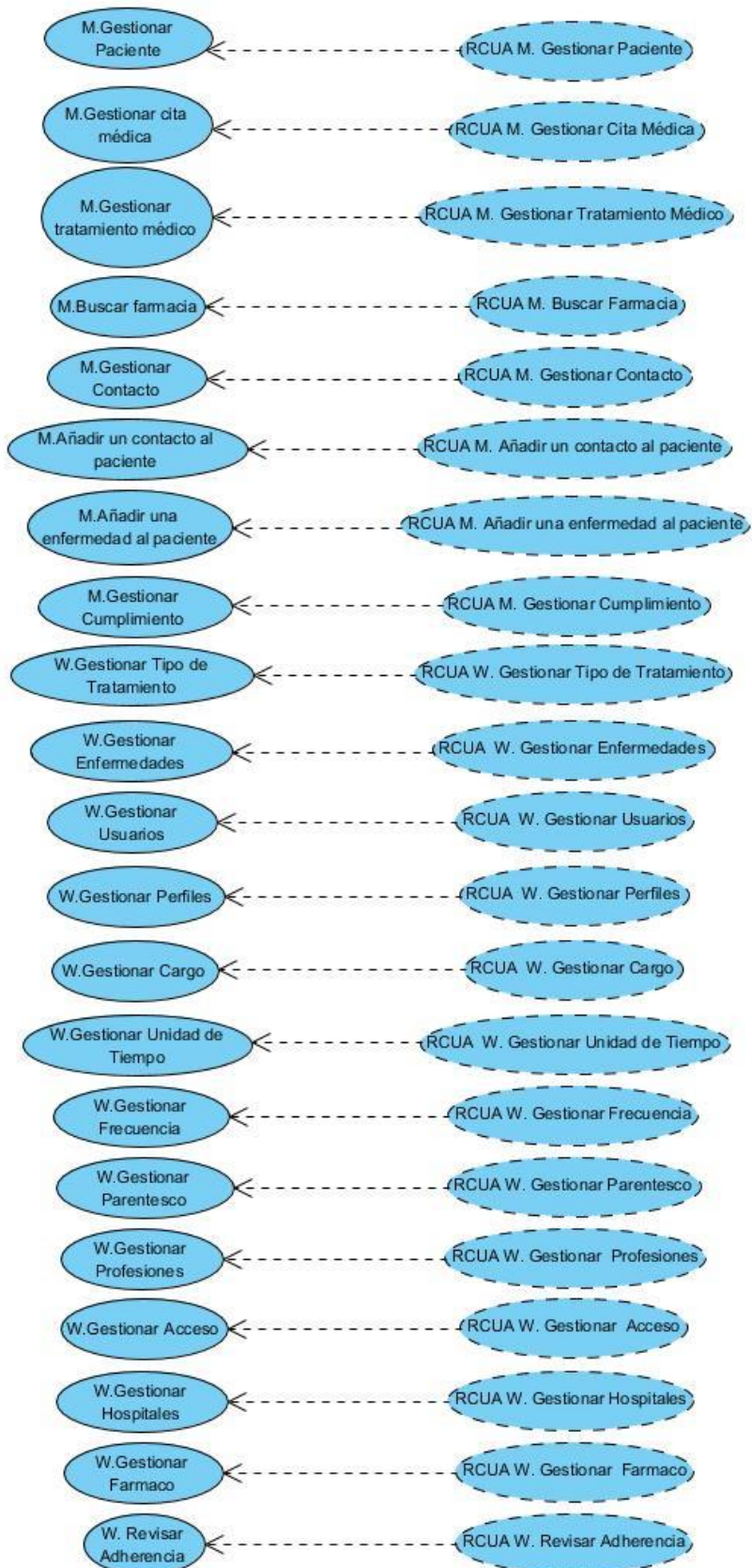
### 4.1.PAQUETE DE ANÁLISIS

DIAGRAMA DE CONTEXTO DE ANÁLISIS



### 4.2.DIAGRAMA DE REALIZACIÓN DE CASOS DE USO DE ANÁLISIS

**DIAGRAMA DE REALIZACION DE CASOS DE USO DE ANALISIS**

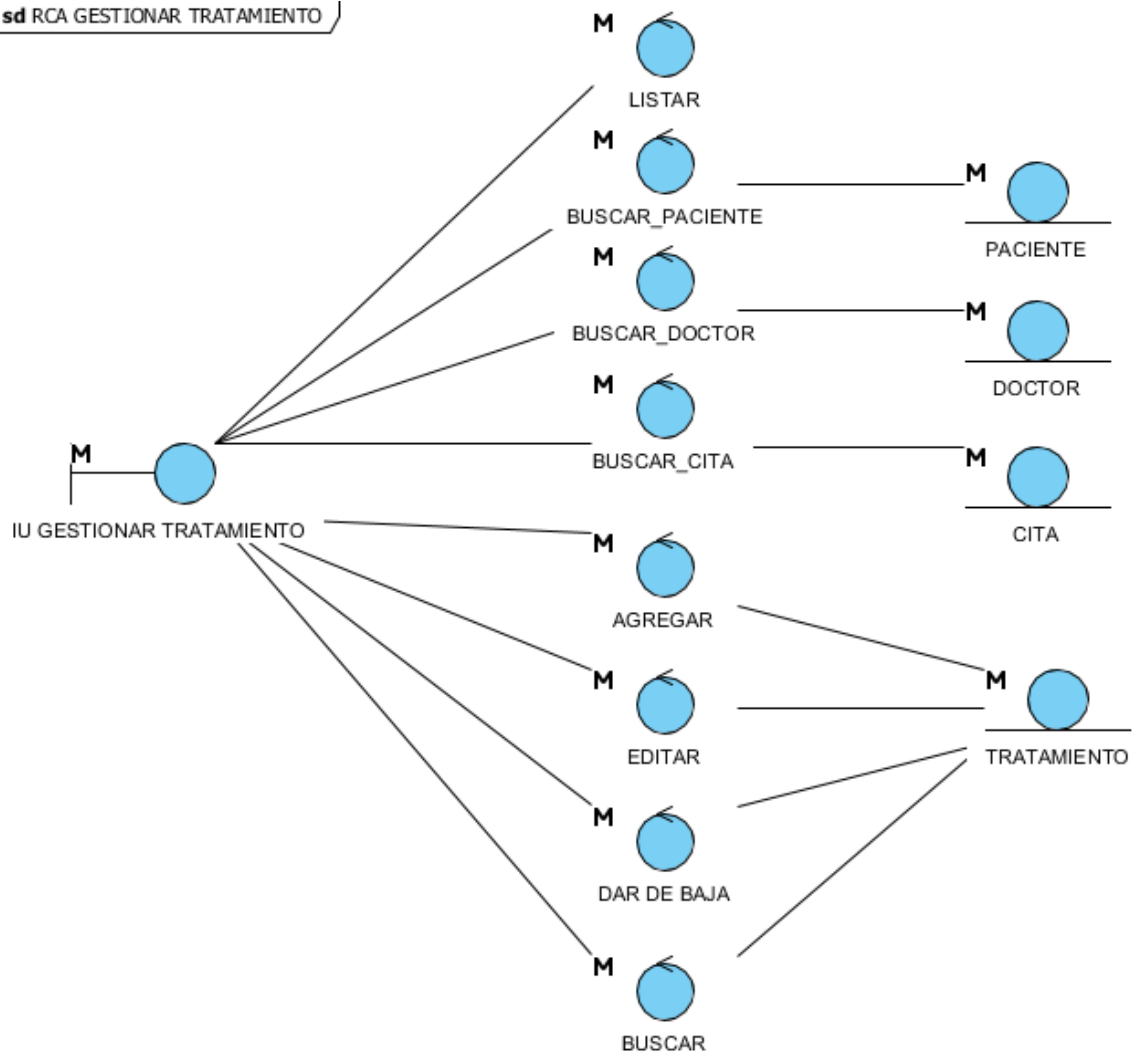




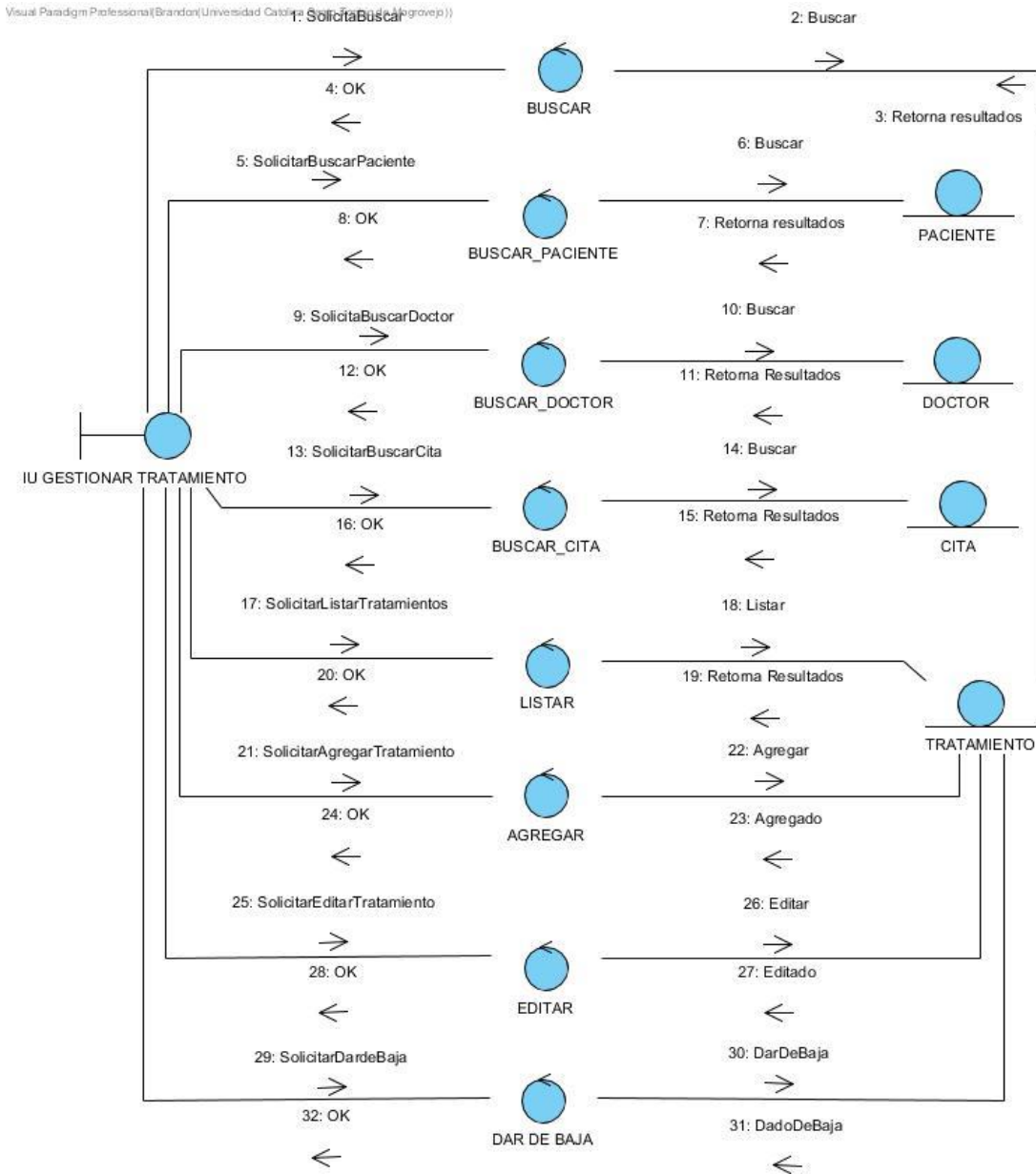
## 4.2.1. RCUA Gestionar Tratamiento

### 4.2.1.1.DCA Gestionar Tratamiento

sd RCA GESTIONAR TRATAMIENTO

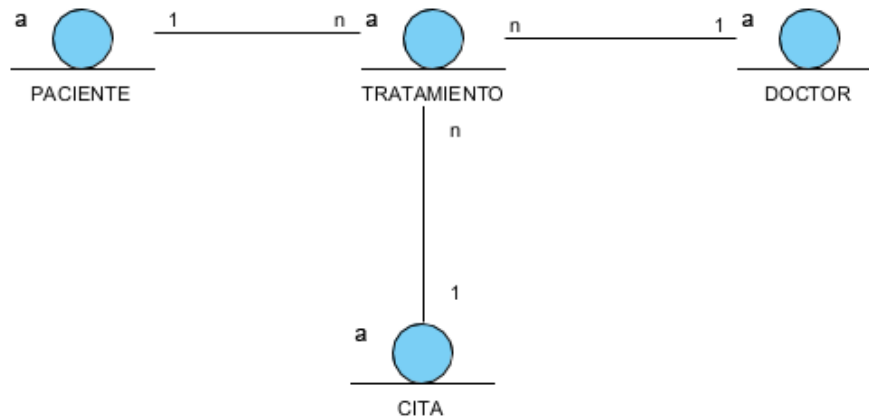


### 4.2.1.2.DCO Gestionar Tratamiento



### 4.2.1.3.DCP Gestionar Tratamiento

sd DC PARCIALES GESTIONAR TRATAMIENTO



### 4.2.1.4.DT Gestionar Tratamiento

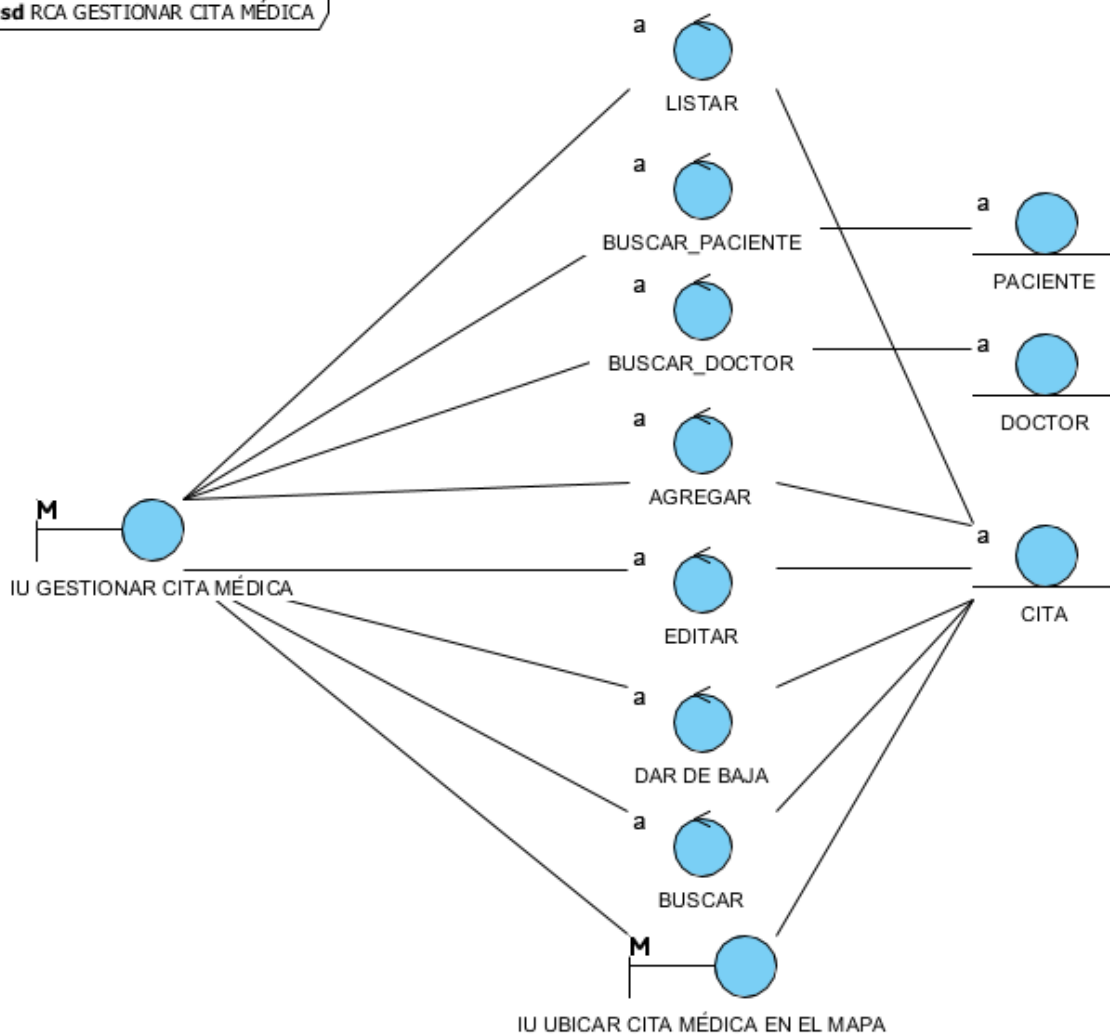
- **Paso 1:** el usuario solicita buscar
- **Paso 2:** El sistema busca
- **Paso 3:** El sistema retorna resultados
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar paciente
- **Paso 6:** El sistema busca el paciente
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita buscar doctor
- **Paso 10:** El sistema busca el doctor
- **Paso 11:** El sistema retorna resultados
- **Paso 12:** OK
- **Paso 13:** El usuario solicita buscar cita
- **Paso 14:** El sistema busca la cita
- **Paso 15:** El sistema retorna resultados
- **Paso 16:** OK
- **Paso 17:** El usuario solicita listar tratamientos
- **Paso 18:** El sistema lista tratamiento
- **Paso 19:** El sistema retorna resultados
- **Paso 20:** OK
- **Paso 21:** El usuario solicita agregar tratamiento
- **Paso 22:** El sistema agrega
- **Paso 23:** El sistema retorna agregado
- **Paso 24:** OK
- **Paso 25:** El usuario solicita editar tratamiento

- **Paso 26:** El sistema edita el tratamiento
- **Paso 27:** El sistema retorna editado
- **Paso 28:** OK
- **Paso 29:** El usuario solicita dar de baja al tratamiento
- **Paso 30:** El sistema da de baja al tratamiento
- **Paso 31:** El sistema retorna un cambio de estado
- **Paso 32:** OK

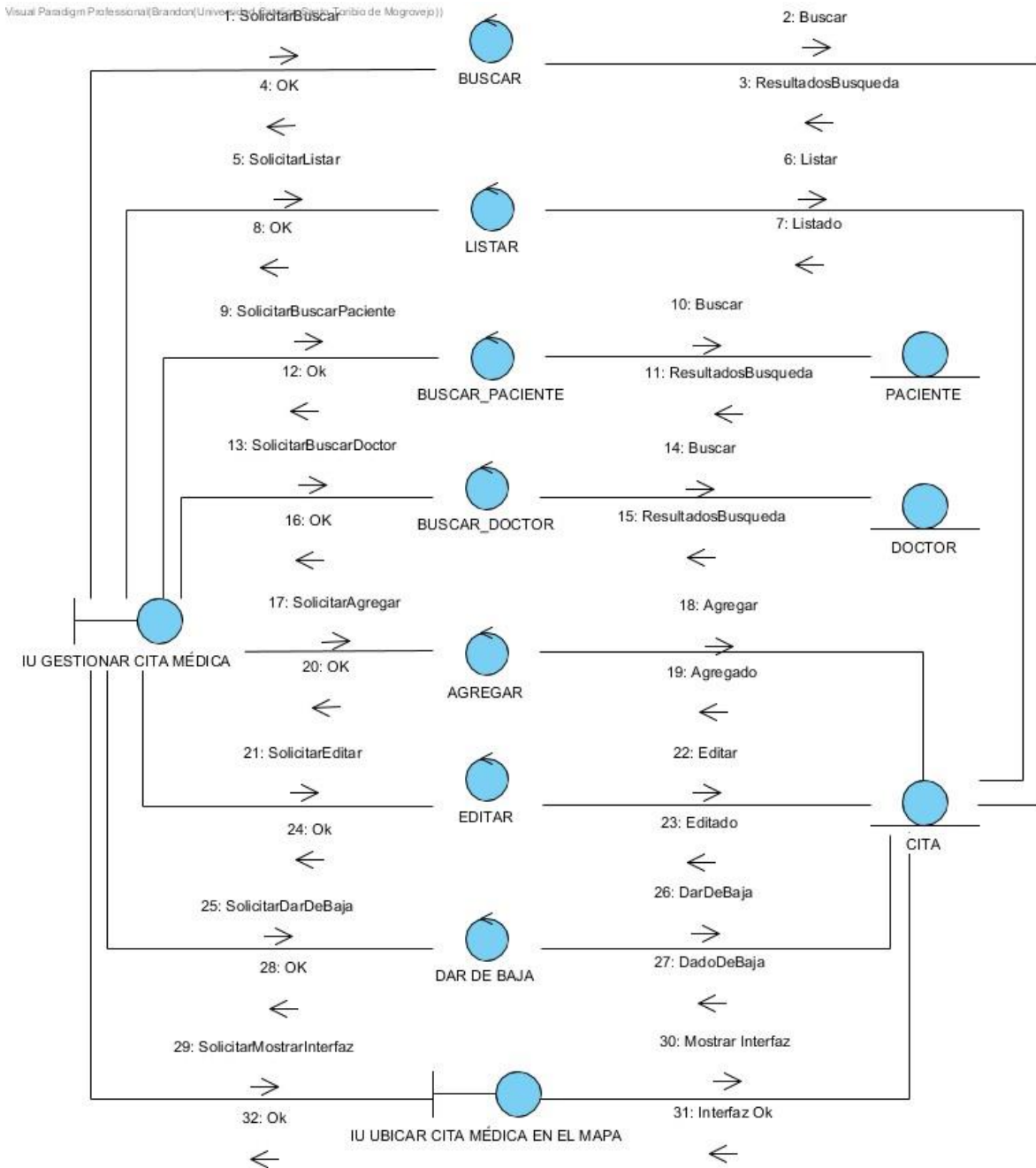
## 4.2.2. RCUA Gestionar Cita Médica

### 4.2.2.1.DCA Gestionar Cita Médica

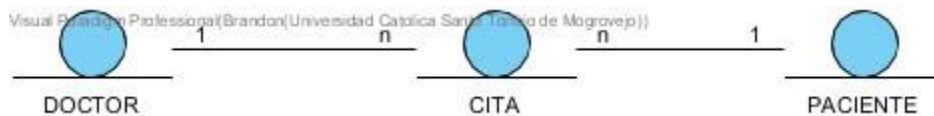
sd RCA GESTIONAR CITA MÉDICA



### 4.2.2.2.DCO Gestionar Cita Médica



### 4.2.2.3.DCP Gestionar Cita Médica



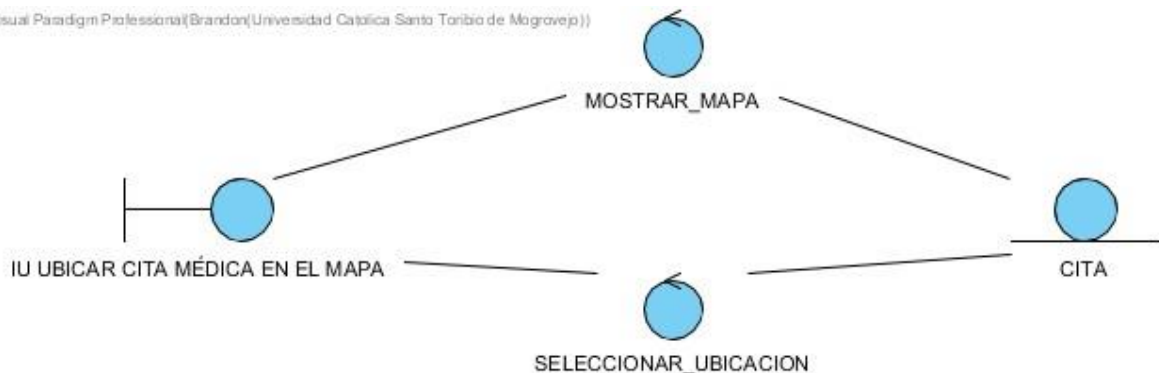
#### 4.2.2.4.DT Gestionar Cita Médica

- **Paso 1:** el usuario solicita buscar
- **Paso 2:** El sistema busca
- **Paso 3:** El sistema retorna resultados
- **Paso 4:** OK
- **Paso 5:** El usuario solicita listar
- **Paso 6:** El sistema lista las citas
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita buscar paciente
- **Paso 10:** El sistema busca el paciente
- **Paso 11:** El sistema retorna resultados
- **Paso 12:** OK
- **Paso 13:** El usuario solicita buscar doctor
- **Paso 14:** El sistema busca el doctor
- **Paso 15:** El sistema retorna resultados
- **Paso 16:** OK
- **Paso 17:** El usuario solicita agregar cita medica
- **Paso 18:** El sistema agrega cita médica
- **Paso 19:** El sistema retorna agregado
- **Paso 20:** OK
- **Paso 21:** El usuario solicita editar cita médica
- **Paso 22:** El sistema edita la cita médica
- **Paso 23:** El sistema retorna editado
- **Paso 24:** OK
- **Paso 25:** El usuario solicita dar de baja a la cita médica.
- **Paso 26:** El sistema da de baja a la cita médica.
- **Paso 27:** El sistema retorna un cambio de estado
- **Paso 28:** OK
- **Paso 29:**

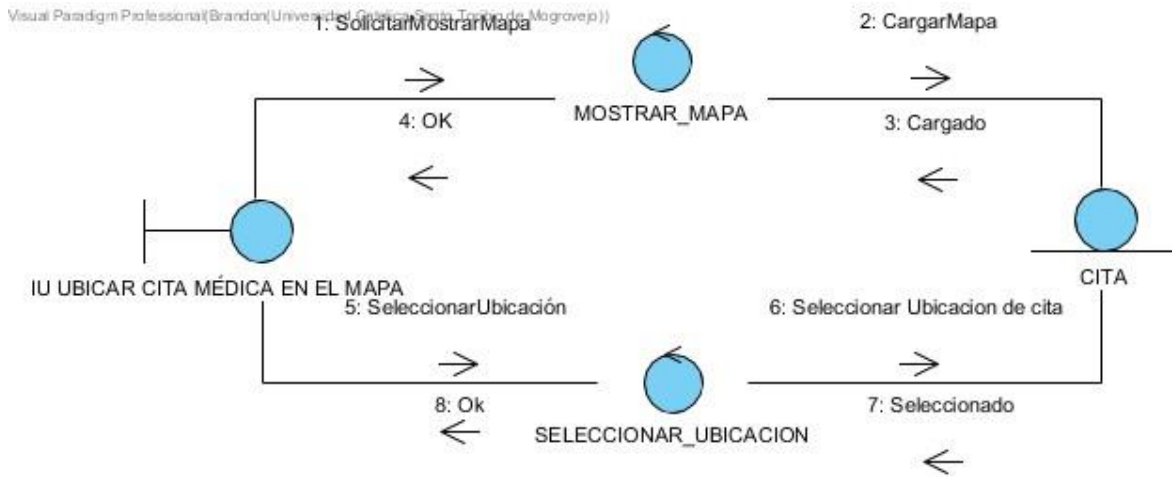
#### 4.2.3. RCUA Ubicar Cita Médica en el mapa

##### 4.2.3.1.DCA Ubicar Cita Médica en el mapa

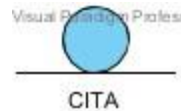
Visual Paradigm Professional (Brandon (Universidad Católica Santo Toribio de Mogrovejo))



### 4.2.3.2.DCO Ubicar Cita Médica en el mapa



### 4.2.3.3.DCP Ubicar Cita Médica en el mapa

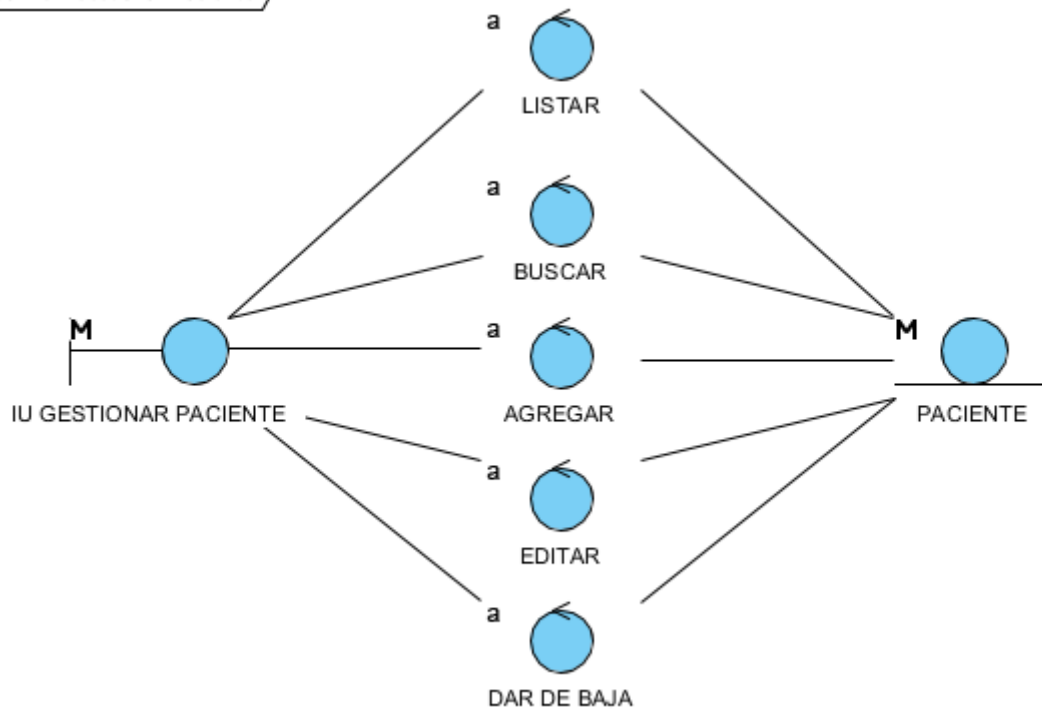


### 4.2.3.4.DT Ubicar Cita Médica en el mapa

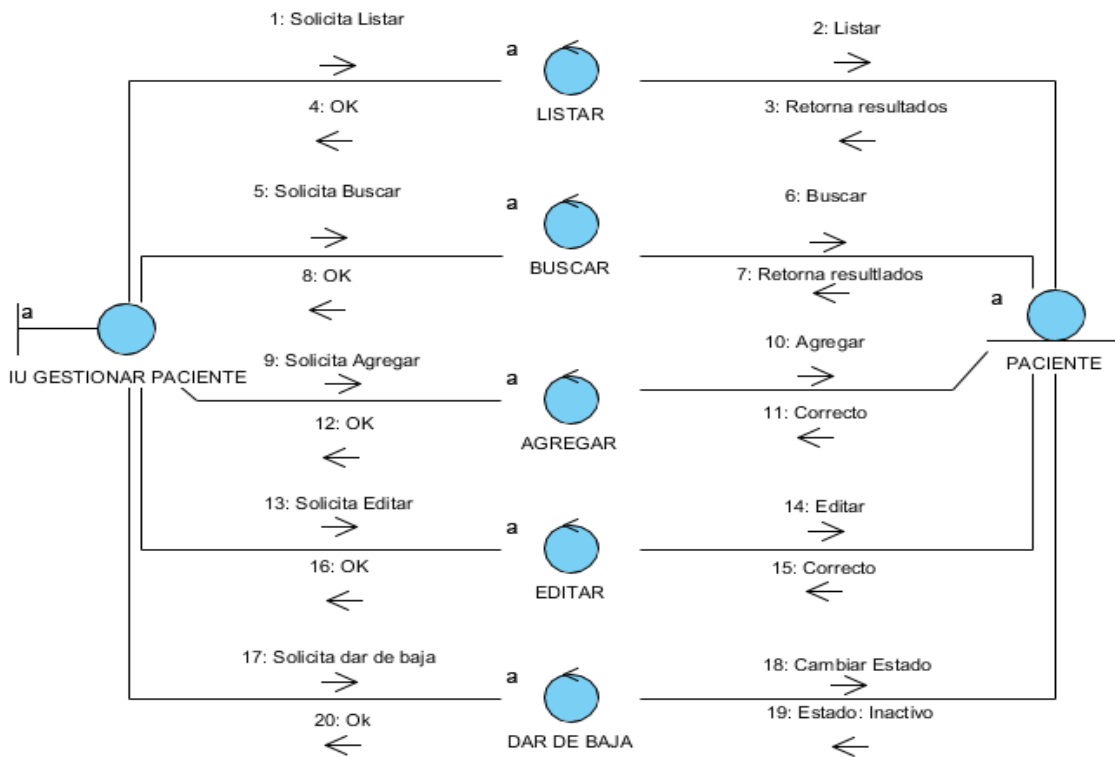
- **Paso 1:** el usuario solicita mostrar mapa
- **Paso 2:** El sistema carga el mapa
- **Paso 3:** El sistema muestra el mapa
- **Paso 4:** OK
- **Paso 5:** El usuario solicita seleccionar una ubicación
- **Paso 6:** El sistema captura la selección
- **Paso 7:** El sistema retorna seleccionado
- **Paso 8:** OK

## 4.2.4. RCUA M. Gestionar Paciente

### 4.2.4.1.DCA Gestionar Paciente



### 4.2.4.2.DCO Gestionar Paciente





#### 4.2.4.3.DCP Gestionar Paciente

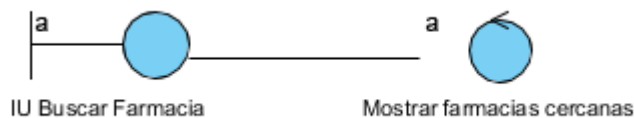


#### 4.2.4.4.DT Gestionar Paciente

- **Paso 1:** El usuario solicita listar
- **Paso 2:** El sistema lista los pacientes
- **Paso 3:** El sistema retorna resultados
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar un paciente
- **Paso 6:** El sistema busca un paciente
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar un paciente
- **Paso 10:** El sistema agrega un paciente
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar un paciente
- **Paso 14:** El sistema edita un paciente
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja al paciente
- **Paso 18:** El sistema cambia el estado de un paciente
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK

#### 4.2.5. RCUA M. Buscar Farmacia

##### 4.2.5.1.DCA Buscar Farmacia



##### 4.2.5.2.DCO Buscar Farmacia

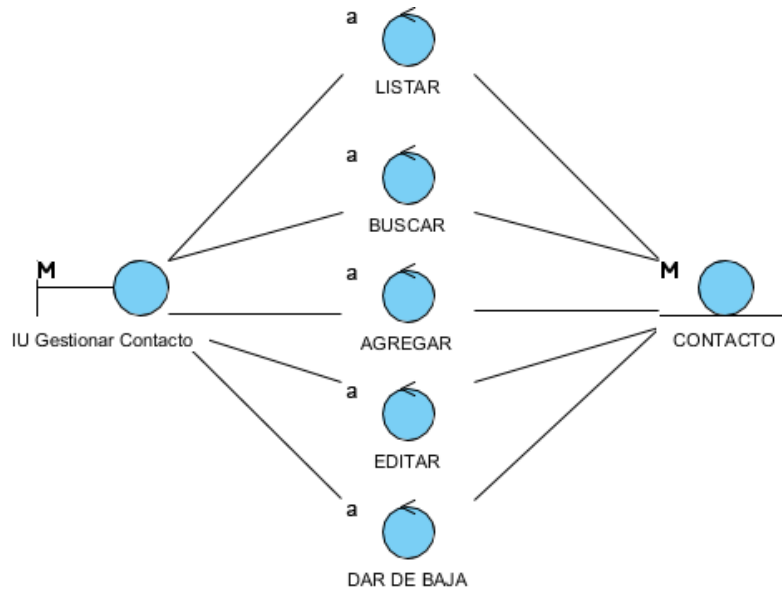


##### 4.2.5.3.DT Buscar Farmacia

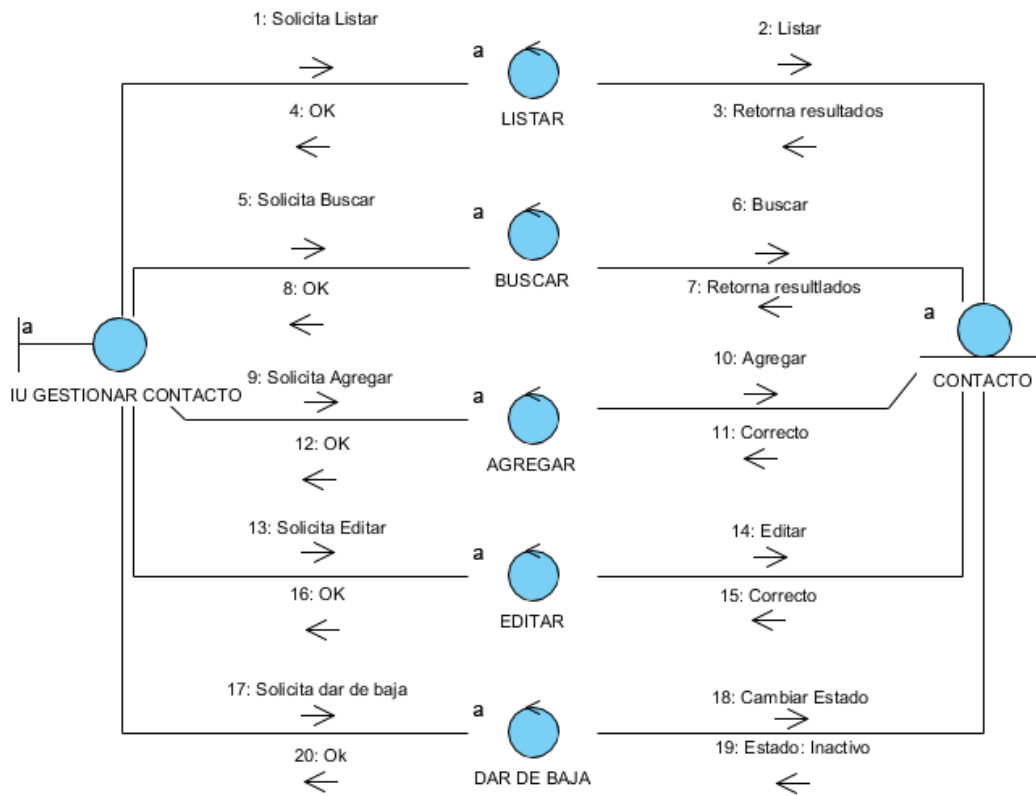
- **Paso 1:** El usuario solicita mostrar farmacias cercanas.
- **Paso 2:** El sistema muestra farmacias cercanas.

## 4.2.6. RCUA M. Gestionar Contacto

### 4.2.6.1.DCA Gestionar Contacto



### 4.2.6.2.DCO Gestionar Contacto



### 4.2.6.3.DCP Gestionar Contacto



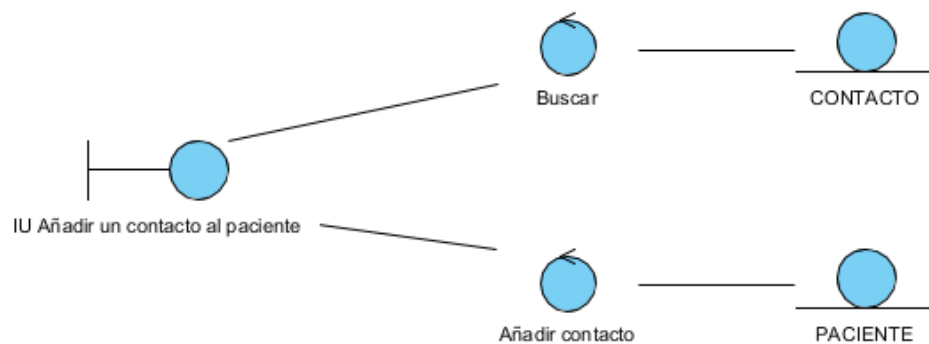
### 4.2.6.4.DT Gestionar Contacto

- **Paso 1:** El usuario solicita listar contacto
- **Paso 2:** El sistema lista los contactos.
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar contacto.
- **Paso 6:** El sistema busca el contacto
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar contacto
- **Paso 10:** El sistema agrega el contacto
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar contacto
- **Paso 14:** El sistema edita el contacto
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a un contacto
- **Paso 18:** El sistema cambia el estado del contacto
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK

### 4.2.7. RCUA M. Añadir un contacto al paciente

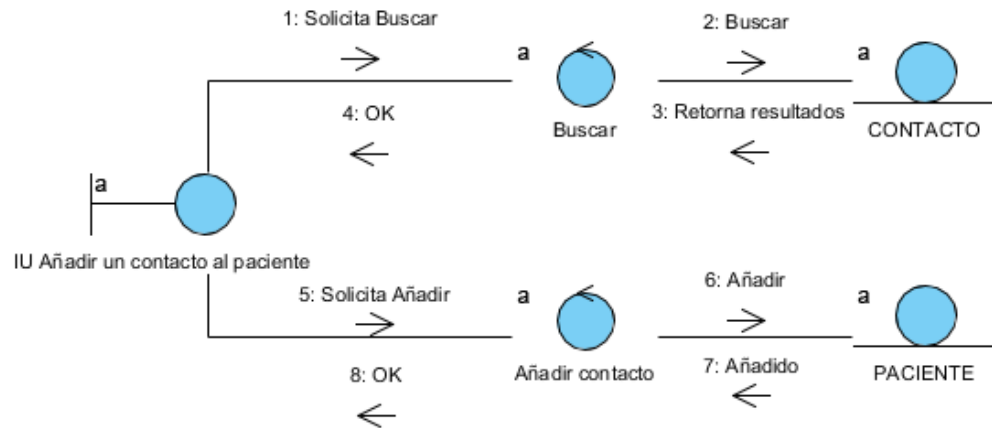
#### 4.2.7.1. DCA M. Añadir un contacto al paciente

sd DCA Anadir un contacto al paciente )



#### 4.2.7.2.DCO M. Añadir un contacto al paciente

sd DCO Añadir un contacto al paciente



#### 4.2.7.3.DCP M. Añadir un contacto al paciente

sd DCP Añadir un contacto al paciente



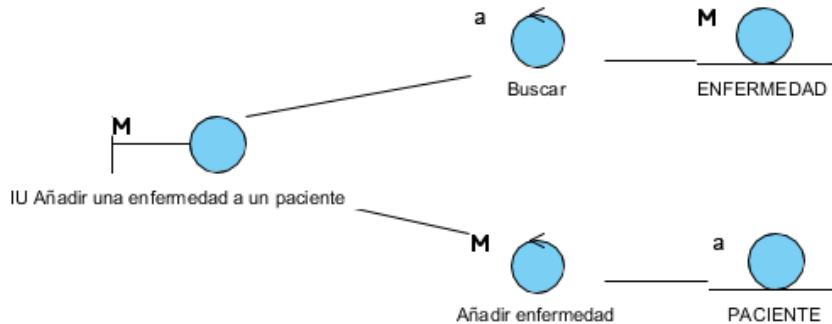
#### 4.2.7.4.DT M. Añadir un contacto al paciente

- **Paso 1:** El usuario solicita buscar contactos
- **Paso 2:** El sistema busca el contacto
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita añadir contacto al paciente
- **Paso 6:** El sistema añade (enlaza) contacto con paciente
- **Paso 7:** El sistema retorna añadido
- **Paso 8:** OK

## 4.2.8. RCUA M. Añadir una enfermedad al paciente

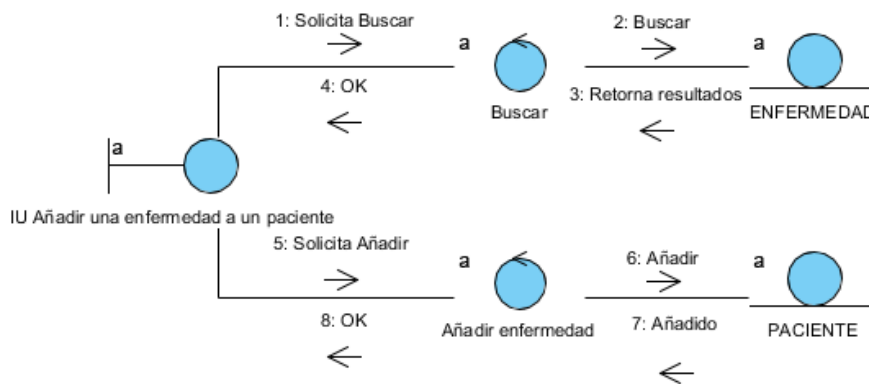
### 4.2.8.1.DCA M. Añadir una enfermedad al paciente

sd DCA Añadir una enfermedad al paciente



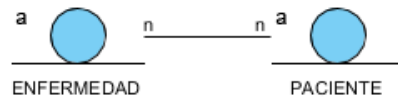
### 4.2.8.2.DCO M. Añadir una enfermedad al paciente

sd DCO Añadir una enfermedad al paciente



### 4.2.8.3.DCP M. Añadir una enfermedad al paciente

sd DCP Añadir una enfermedad al paciente



### 4.2.8.4.DT M. Añadir una enfermedad al paciente

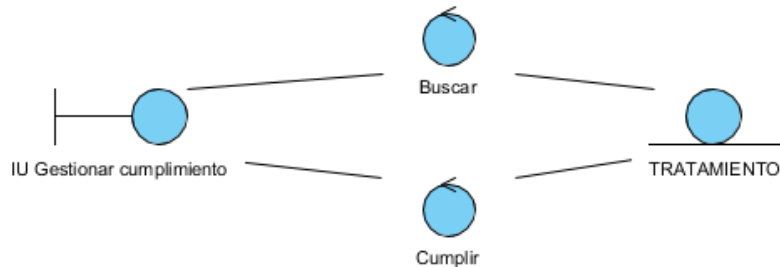
- **Paso 1:** El usuario solicita buscar enfermedades
- **Paso 2:** El sistema busca la enfermedad
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita añadir enfermedad al paciente
- **Paso 6:** El sistema añade (enlaza) enfermedad con paciente
- **Paso 7:** El sistema retorna añadido

- Paso 8: OK

#### 4.2.9. RCUA M. Gestionar Cumplimiento

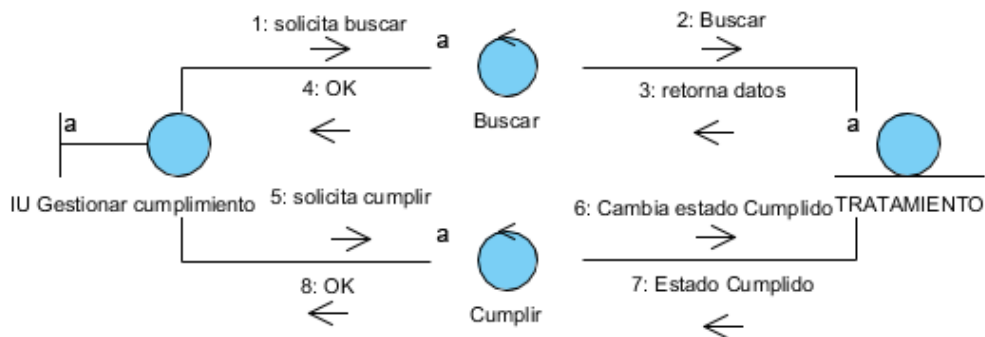
##### 4.2.9.1.DCA Gestionar Cumplimiento

sd DCA Gestionar Cumplimiento



##### 4.2.9.2.DCO Gestionar Cumplimiento

sd DCO Gestionar Cumplimiento



##### 4.2.9.3.DCP Gestionar Cumplimiento

sd DCP Gestionar Cumplimiento

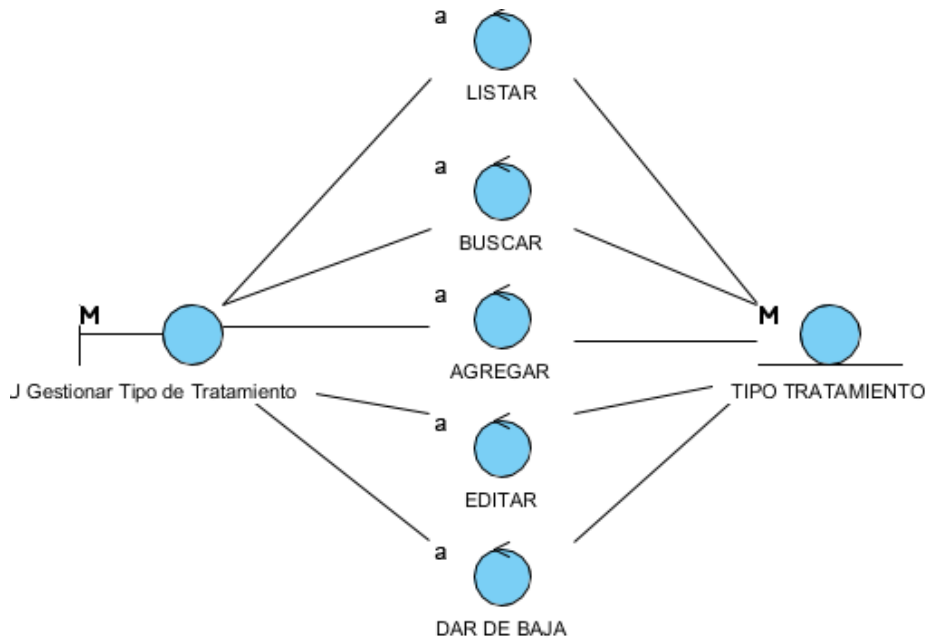


##### 4.2.9.4.DT Gestionar Cumplimiento

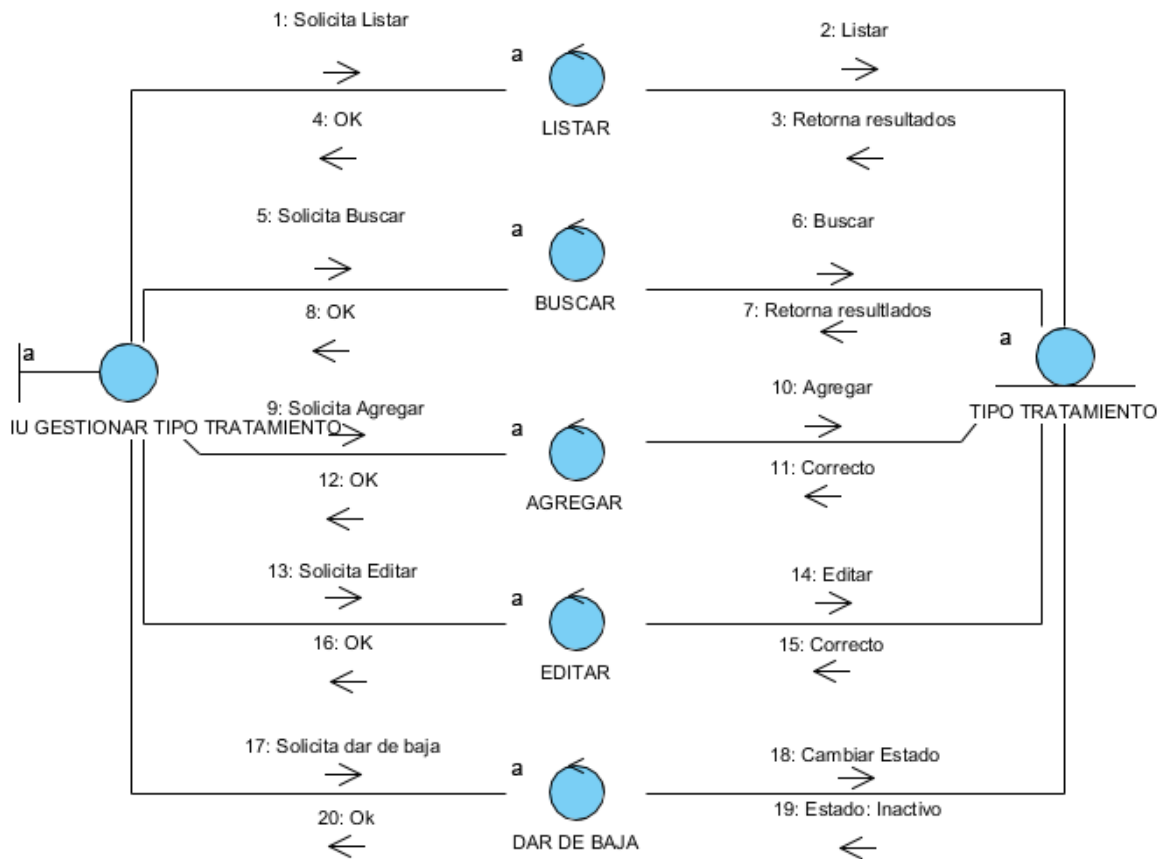
- Paso 1: el usuario solicita buscar el tratamiento del paciente.
- Paso 2: El sistema busca el tratamiento.
- Paso 3: El sistema retorna resultados.
- Paso 4: OK
- Paso 5: El paciente solicita cumplir con el tratamiento.
- Paso 6: El sistema cambia estado de tratamiento.
- Paso 7: Estado tratamiento: Cumplido
- Paso 8: OK

## 4.2.10. RCUA W. Gestionar Tipo de tratamiento

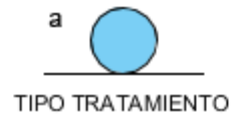
### 4.2.10.1. DCA Gestionar



### 4.2.10.2. DCO Gestionar



#### 4.2.10.3. DCP Gestionar



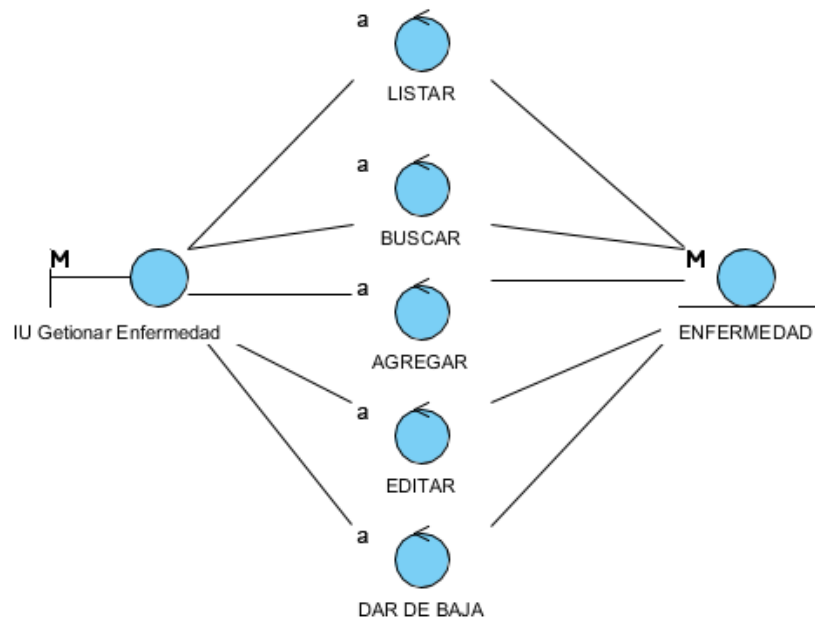
#### 4.2.10.4. DT Gestionar

- **Paso 1:** El usuario solicita listar tipo de tratamientos
- **Paso 2:** El sistema lista los tipos de tratamientos.
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar tipo de tratamientos.
- **Paso 6:** El sistema busca tipo de tratamientos
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar tipo de tratamientos
- **Paso 10:** El sistema agrega el tipo de tratamientos
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar tipo de tratamientos
- **Paso 14:** El sistema edita tipo de tratamientos
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a un tipo de tratamientos
- **Paso 18:** El sistema cambia el estado del tipo de tratamientos
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK

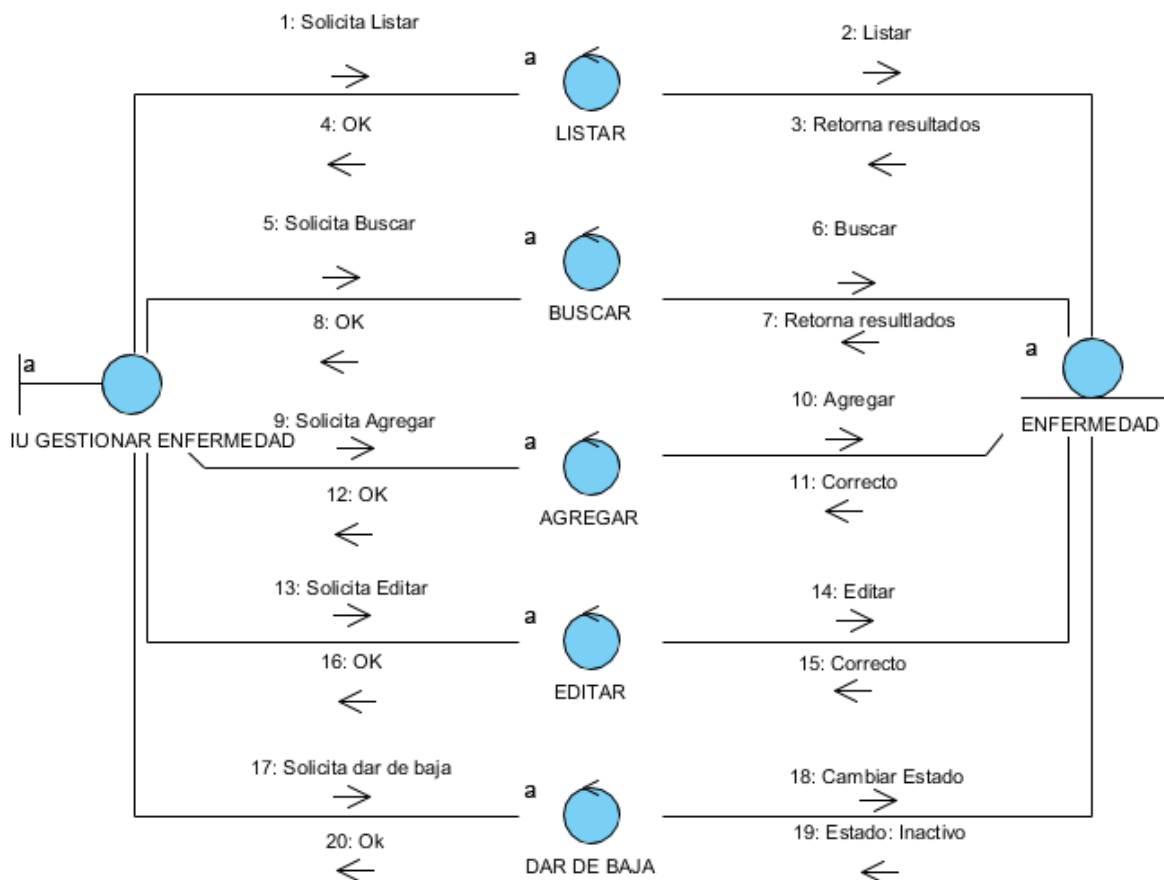


## 4.2.11. RCUA W. Gestionar Enfermedades

### 4.2.11.1. DCA Gestionar



### 4.2.11.2. DCO Gestionar



### 4.2.11.3. DCP Gestionar

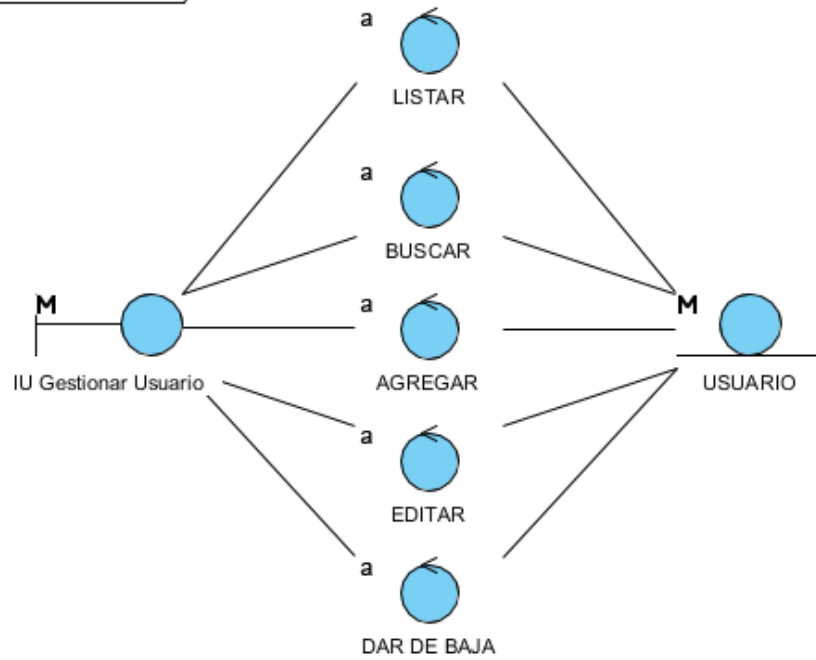


### 4.2.11.4. DT Gestionar

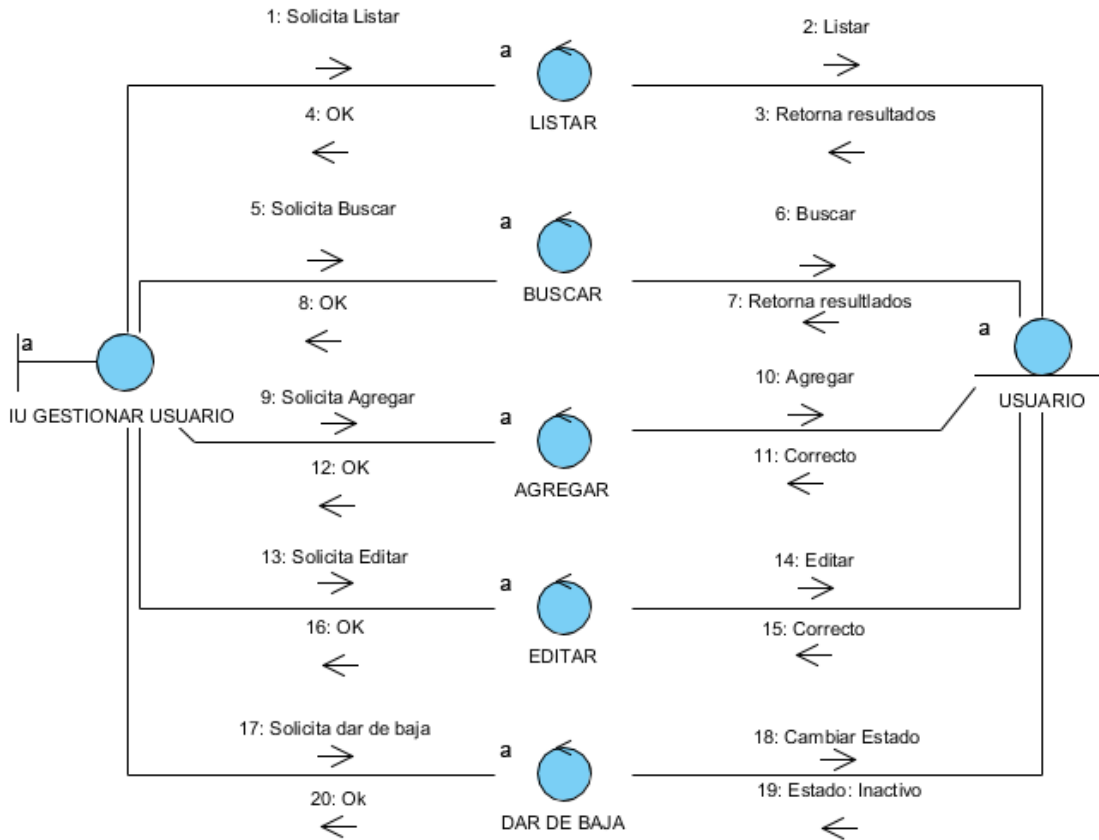
- **Paso 1:** El usuario solicita listar enfermedad
- **Paso 2:** El sistema lista la enfermedad.
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar enfermedad.
- **Paso 6:** El sistema busca la enfermedad
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar enfermedad
- **Paso 10:** El sistema agrega la enfermedad
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar enfermedad
- **Paso 14:** El sistema edita la enfermedad
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a una enfermedad
- **Paso 18:** El sistema cambia el estado de la enfermedad
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK
- 

### 4.2.12. RCUA W. Gestionar Usuarios

#### 4.2.12.1. DCA Gestionar



#### 4.2.12.2. DCO Gestionar



#### 4.2.12.3. DCP Gestionar

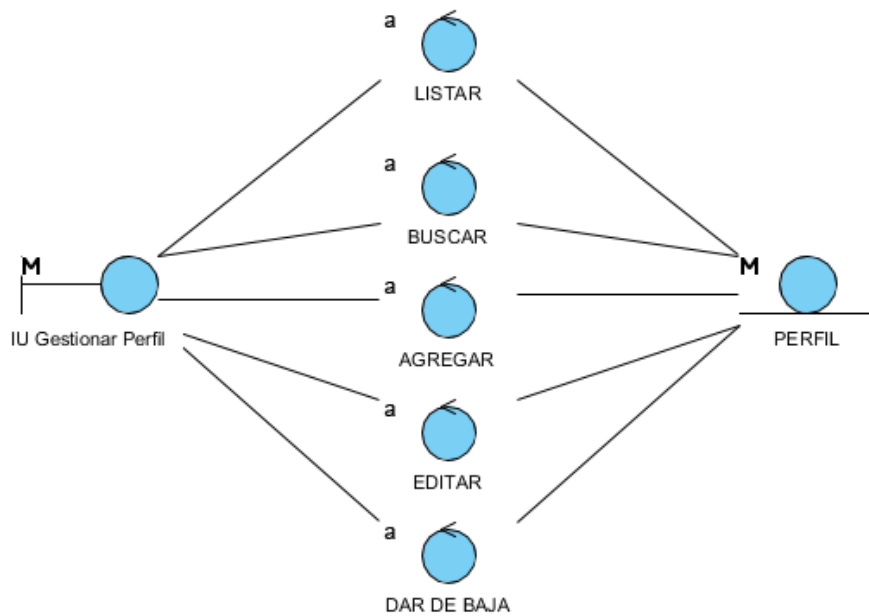


#### 4.2.12.4. DT Gestionar

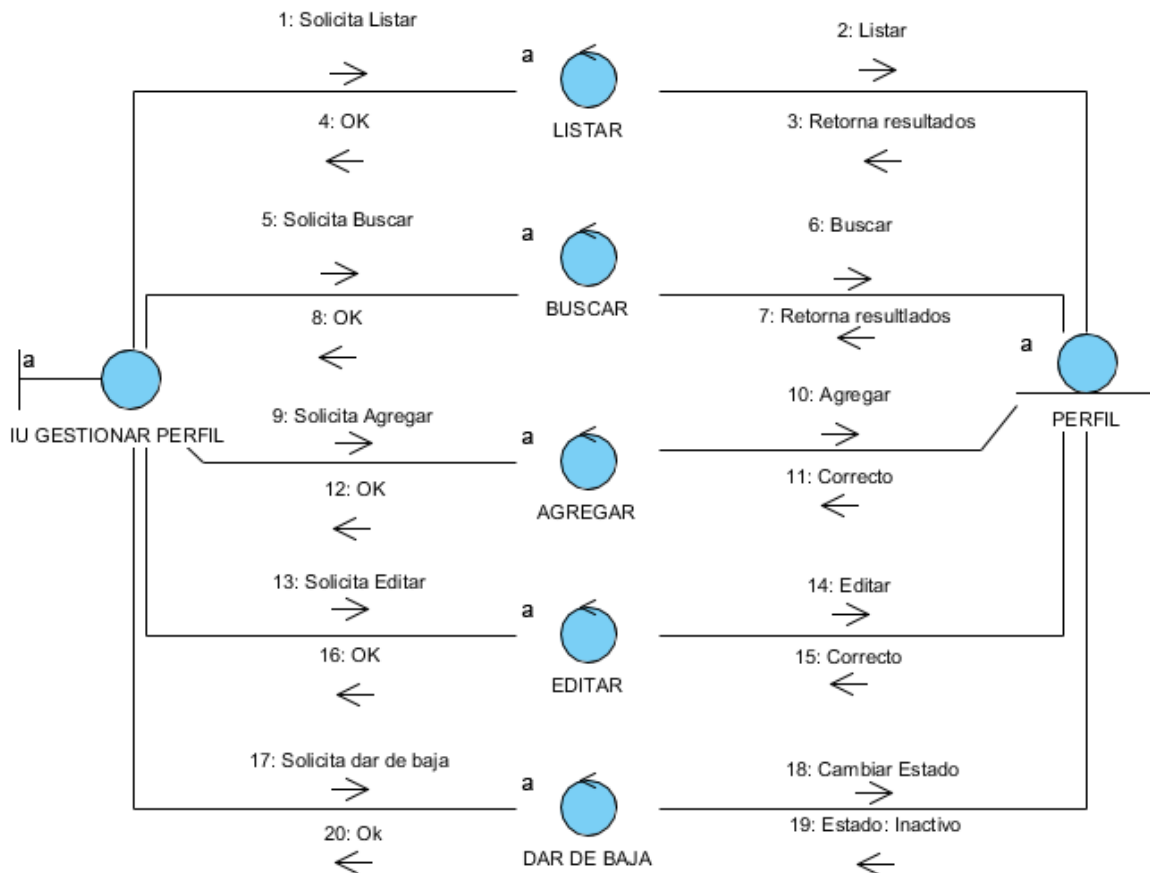
- **Paso 1:** El usuario solicita listar usuario
- **Paso 2:** El sistema lista los usuarios
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar usuarios.
- **Paso 6:** El sistema busca el usuario
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar un usuario
- **Paso 10:** El sistema agrega el usuario
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar el usuario
- **Paso 14:** El sistema edita el usuario
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a un usuario
- **Paso 18:** El sistema cambia el estado del usuario
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK

## 4.2.13. RCUA W. Gestionar Perfiles

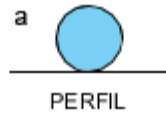
### 4.2.13.1. DCA Gestionar



### 4.2.13.2. DCO Gestionar



#### 4.2.13.3. DCP Gestionar

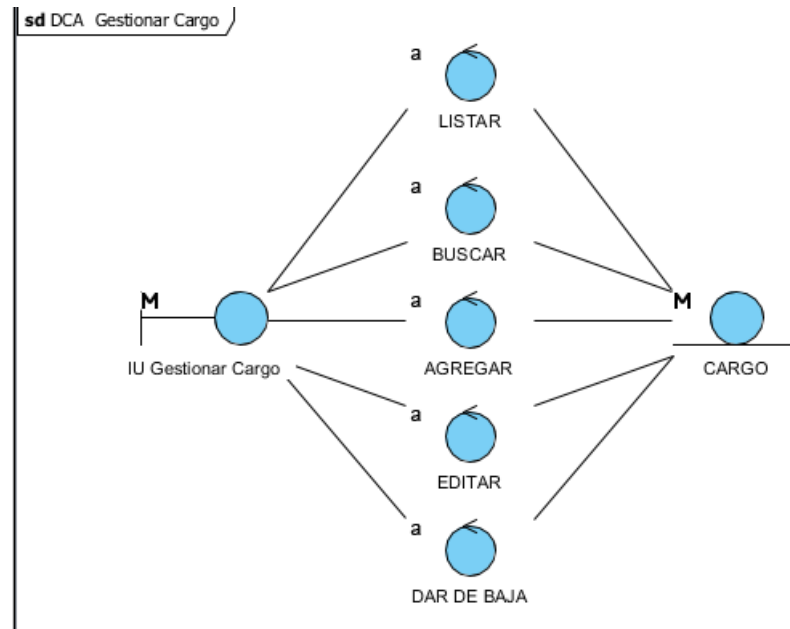


#### 4.2.13.4. DT Gestionar

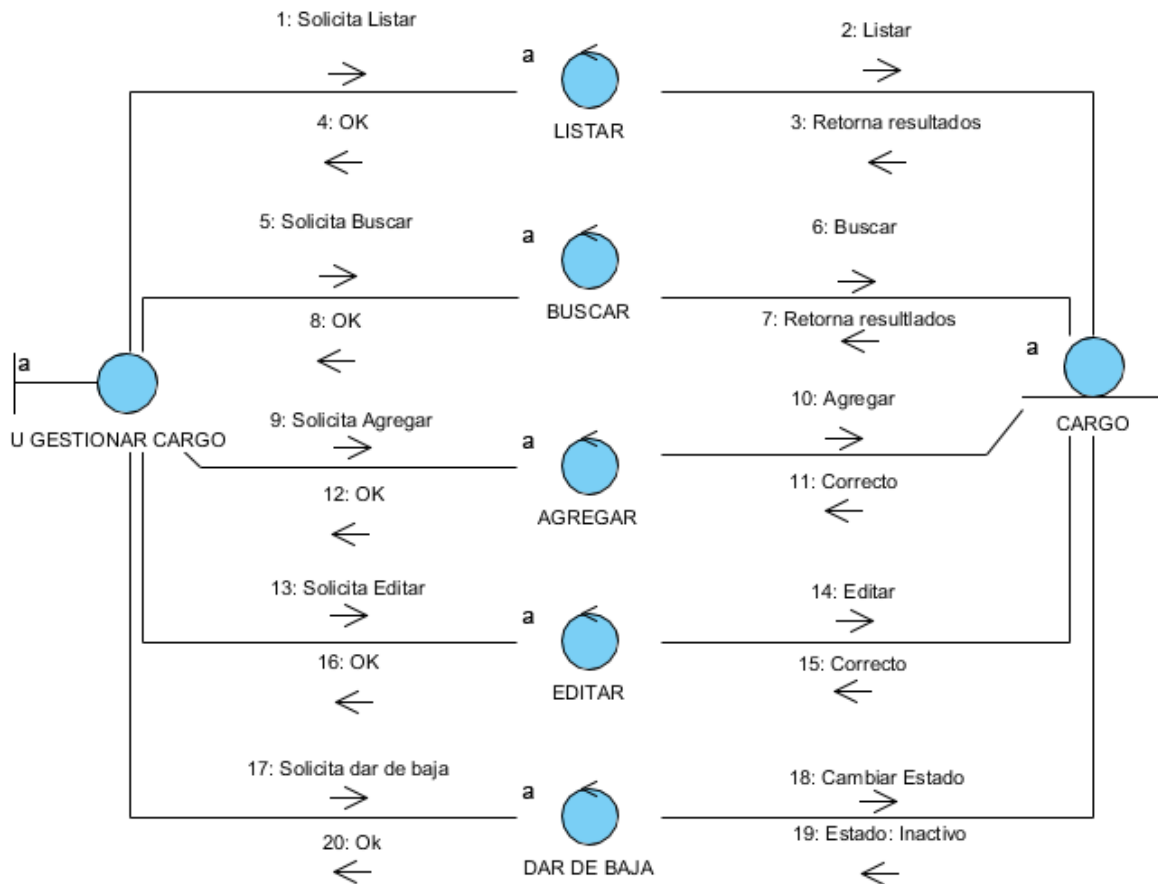
- **Paso 1:** El usuario solicita listar perfil
- **Paso 2:** El sistema lista los perfiles
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar perfiles
- **Paso 6:** El sistema busca el perfil
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar un perfil
- **Paso 10:** El sistema agrega el perfil
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar el perfil
- **Paso 14:** El sistema edita el perfil
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a un perfil
- **Paso 18:** El sistema cambia el estado del perfil
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK

## 4.2.14. RCUA W. Gestionar Cargo

### 4.2.14.1. DCA Gestionar



### 4.2.14.2. DCO Gestionar



#### 4.2.14.3. DCP Gestionar



#### 4.2.14.4. DT Gestionar

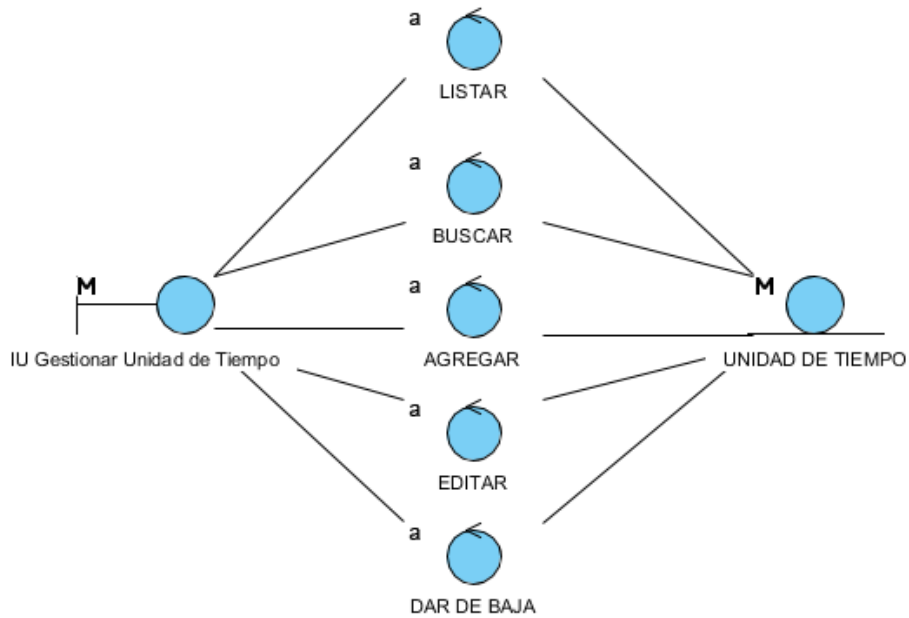
- **Paso 1:** El usuario solicita listar cargo
- **Paso 2:** El sistema lista los cargos
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar cargos
- **Paso 6:** El sistema busca el cargo
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar un cargo
- **Paso 10:** El sistema agrega el cargo
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar el cargo
- **Paso 14:** El sistema edita el cargo
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a un cargo
- **Paso 18:** El sistema cambia el estado del cargo
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK



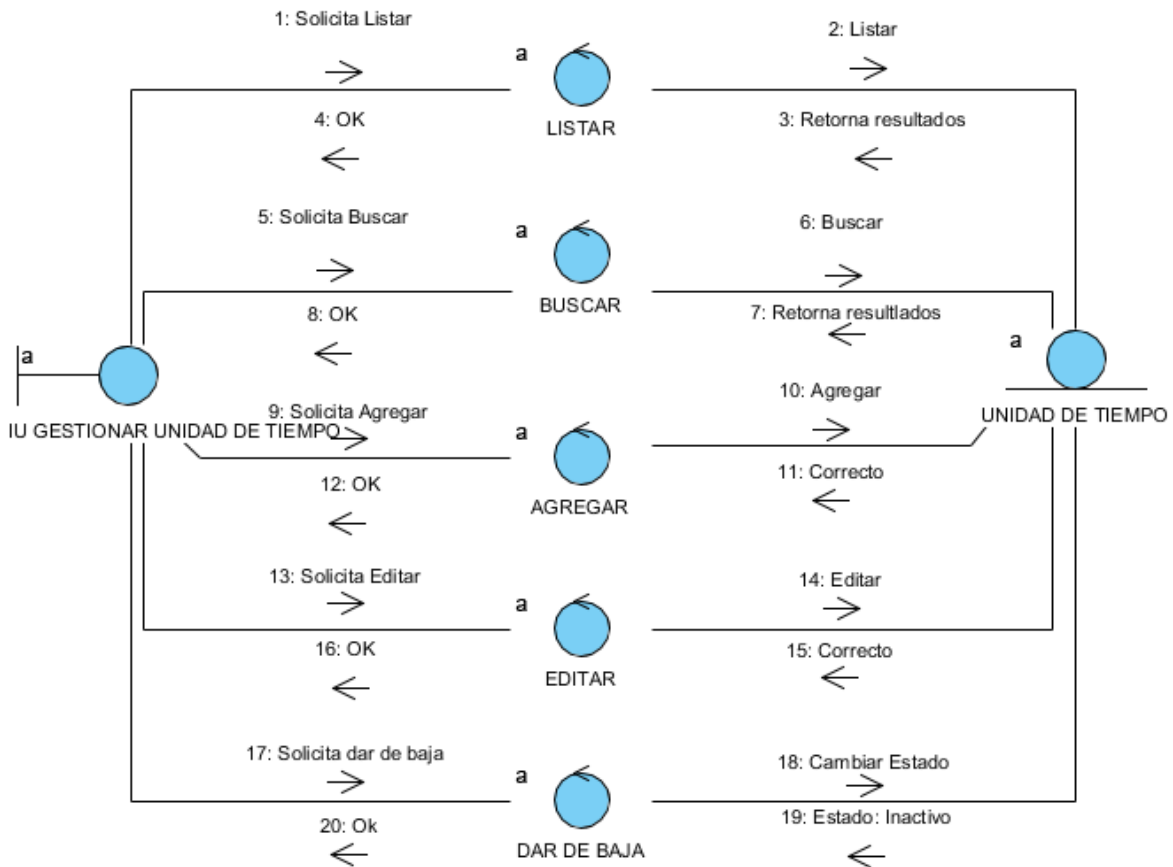
## 4.2.15. RCUA W. Gestionar Unidad de tiempo

### 4.2.15.1. DCA Gestionar

sd DCA Gestionar Unidad de tiempo



### 4.2.15.2. DCO Gestionar



#### 4.2.15.3. DCP Gestionar



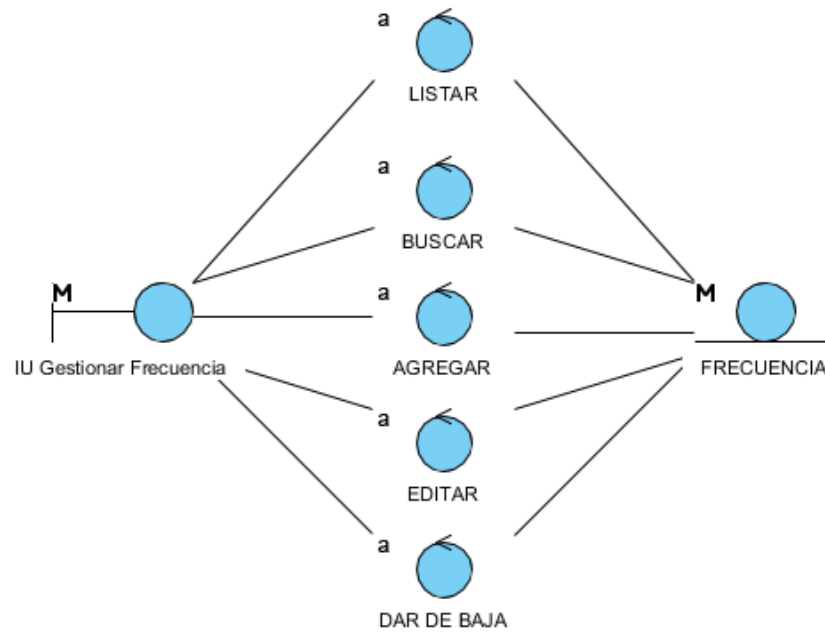
#### 4.2.15.4. DT Gestionar

- **Paso 1:** El usuario solicita listar unidad de tiempo
- **Paso 2:** El sistema lista las unidades de tiempo
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar unidades de tiempo
- **Paso 6:** El sistema busca la unidad de tiempo
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar una unidad de tiempo
- **Paso 10:** El sistema agrega la unidad de tiempo
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar la unidad de tiempo
- **Paso 14:** El sistema edita la unidad de tiempo
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a una unidad de tiempo
- **Paso 18:** El sistema cambia el estado de la unidad de tiempo
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK

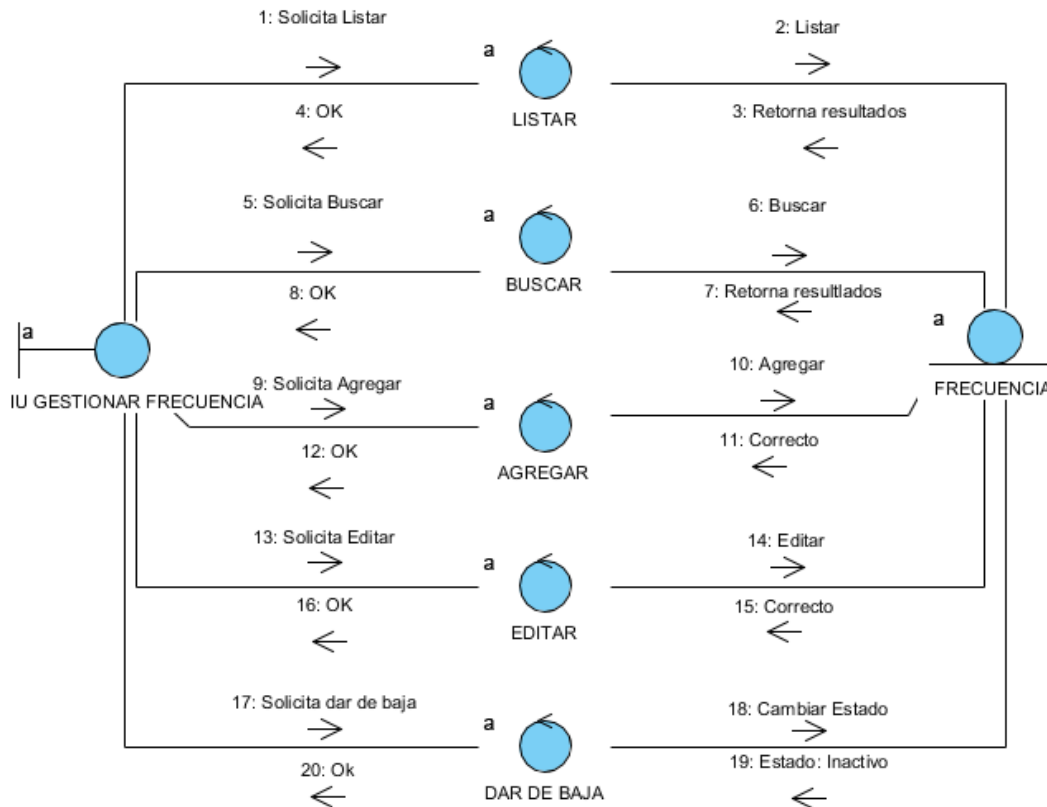
## 4.2.16. RCUA W. Gestionar Frecuencia

### 4.2.16.1. DCA Gestionar

sd DCA Gestionar Frecuencia



### 4.2.16.2. DCO Gestionar



#### 4.2.16.3. DCP Gestionar



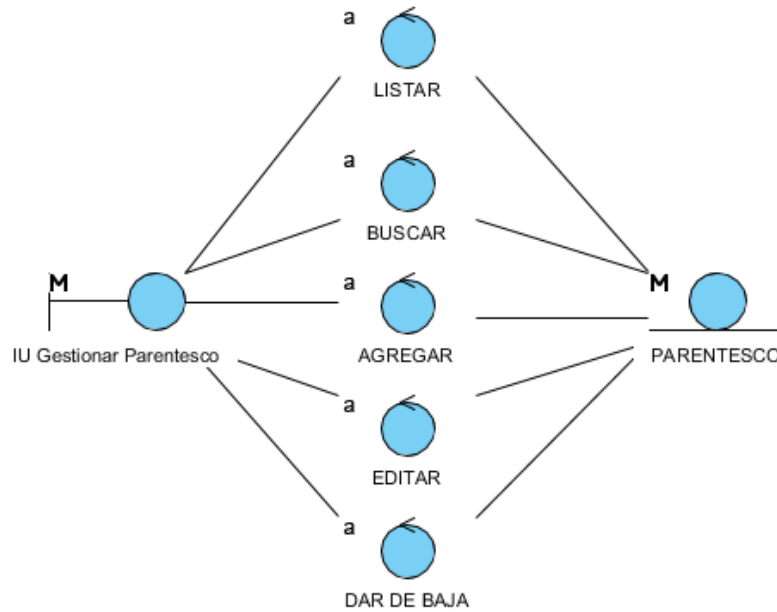
#### 4.2.16.4. DT Gestionar

- **Paso 1:** El usuario solicita listar frecuencia
- **Paso 2:** El sistema lista las frecuencias
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar frecuencias
- **Paso 6:** El sistema busca la frecuencia
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar una frecuencia
- **Paso 10:** El sistema agrega la frecuencia
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar la frecuencia
- **Paso 14:** El sistema edita la frecuencia
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a una frecuencia
- **Paso 18:** El sistema cambia el estado del de la frecuencia
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK

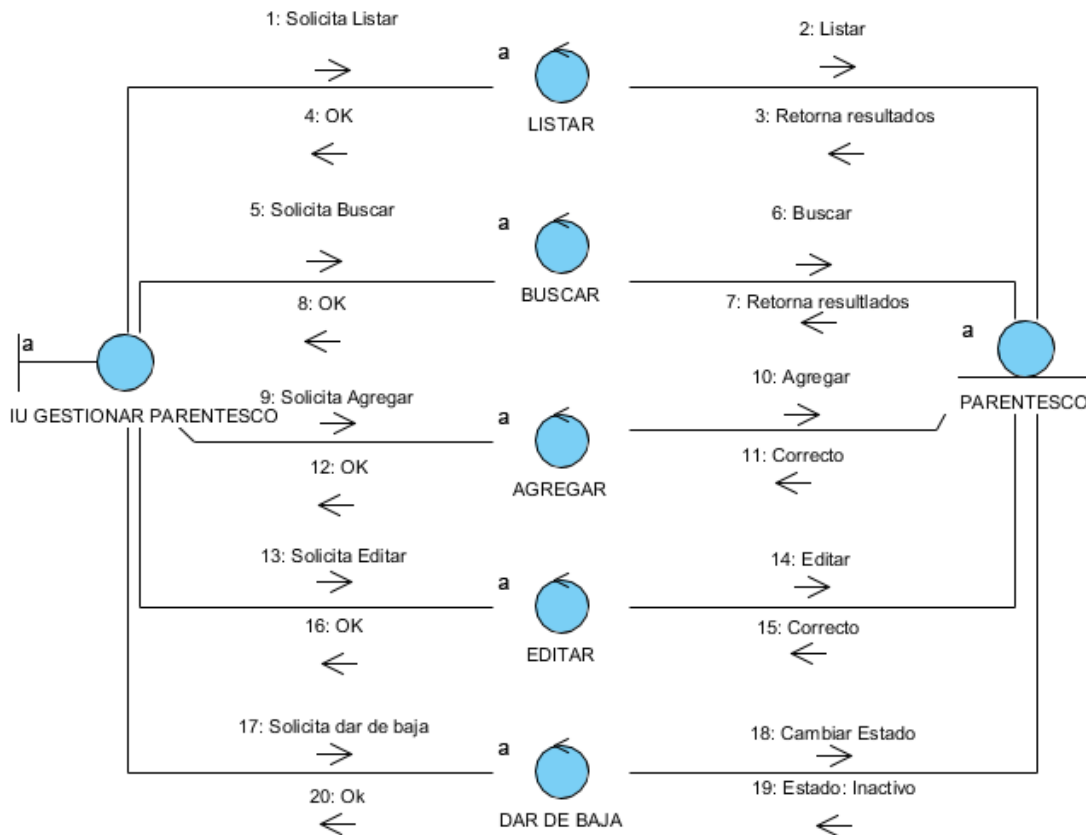
## 4.2.17. RCUA W. Gestionar Parentesco

### 4.2.17.1. DCA Gestionar

sd DCA Gestionar Parentesco



### 4.2.17.2. DCO Gestionar



### 4.2.17.3. DCP Gestionar

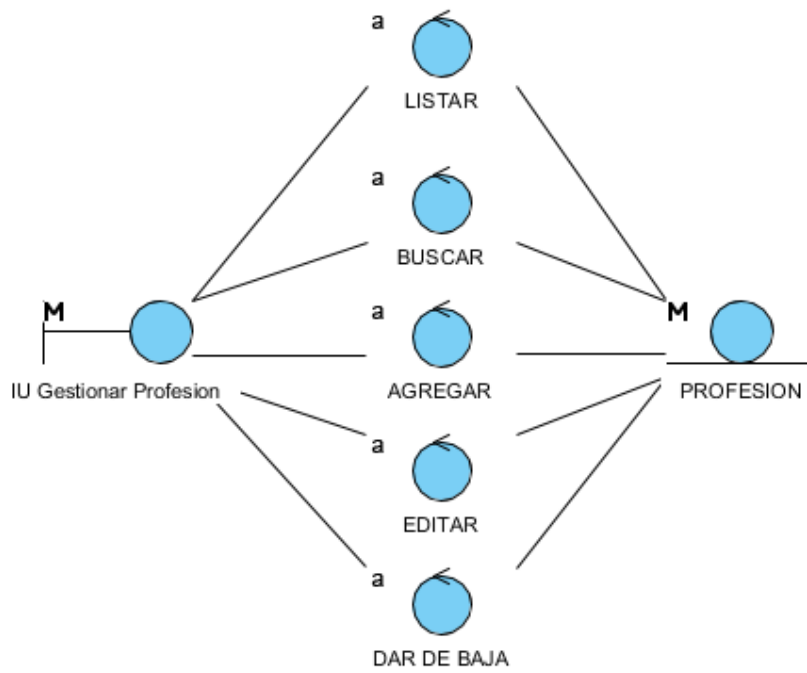


### 4.2.17.4. DT Gestionar

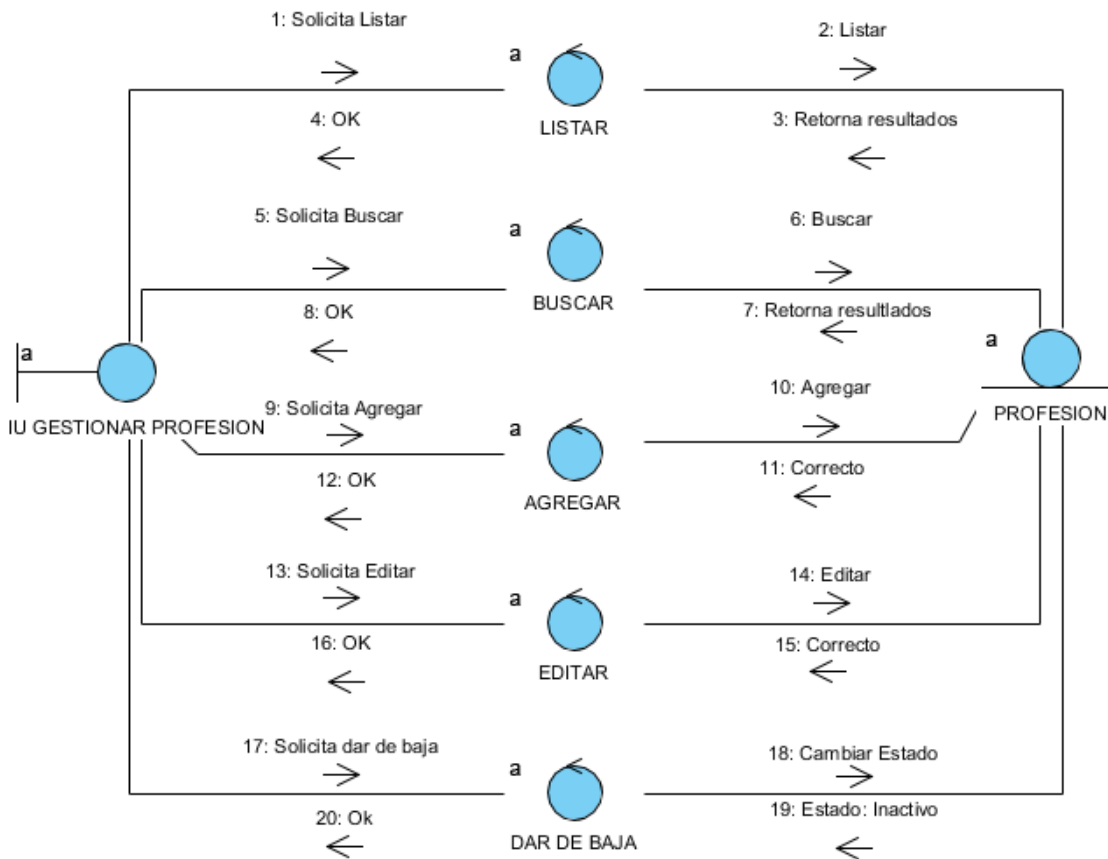
- **Paso 1:** El usuario solicita listar parentesco
- **Paso 2:** El sistema lista los parentescos
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar parentescos
- **Paso 6:** El sistema busca el parentesco
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar un parentesco
- **Paso 10:** El sistema agrega el parentesco
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar el parentesco
- **Paso 14:** El sistema edita el parentesco
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a un parentesco
- **Paso 18:** El sistema cambia el estado del parentesco
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK

### 4.2.18. RCUA W. Gestionar Profesiones

#### 4.2.18.1. DCA Gestionar



#### 4.2.18.2. DCO Gestionar



#### 4.2.18.3. DCP Gestionar



#### 4.2.18.4. DT Gestionar

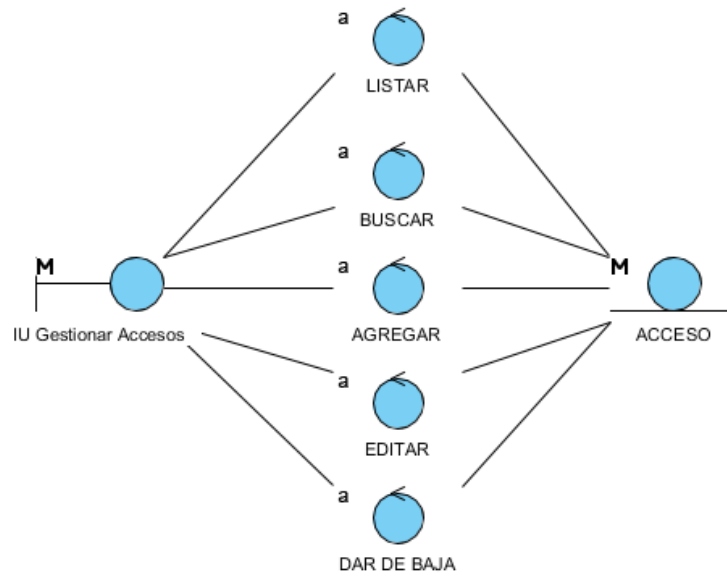
- **Paso 1:** El usuario solicita listar profesiones
- **Paso 2:** El sistema lista las profesiones
- **Paso 3:** El sistema retorna resultados
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar profesiones
- **Paso 6:** El sistema busca la profesión
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar una profesión
- **Paso 10:** El sistema agrega la profesión
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar la profesión
- **Paso 14:** El sistema edita la profesión
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a una profesión
- **Paso 18:** El sistema cambia el estado de la profesión
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK



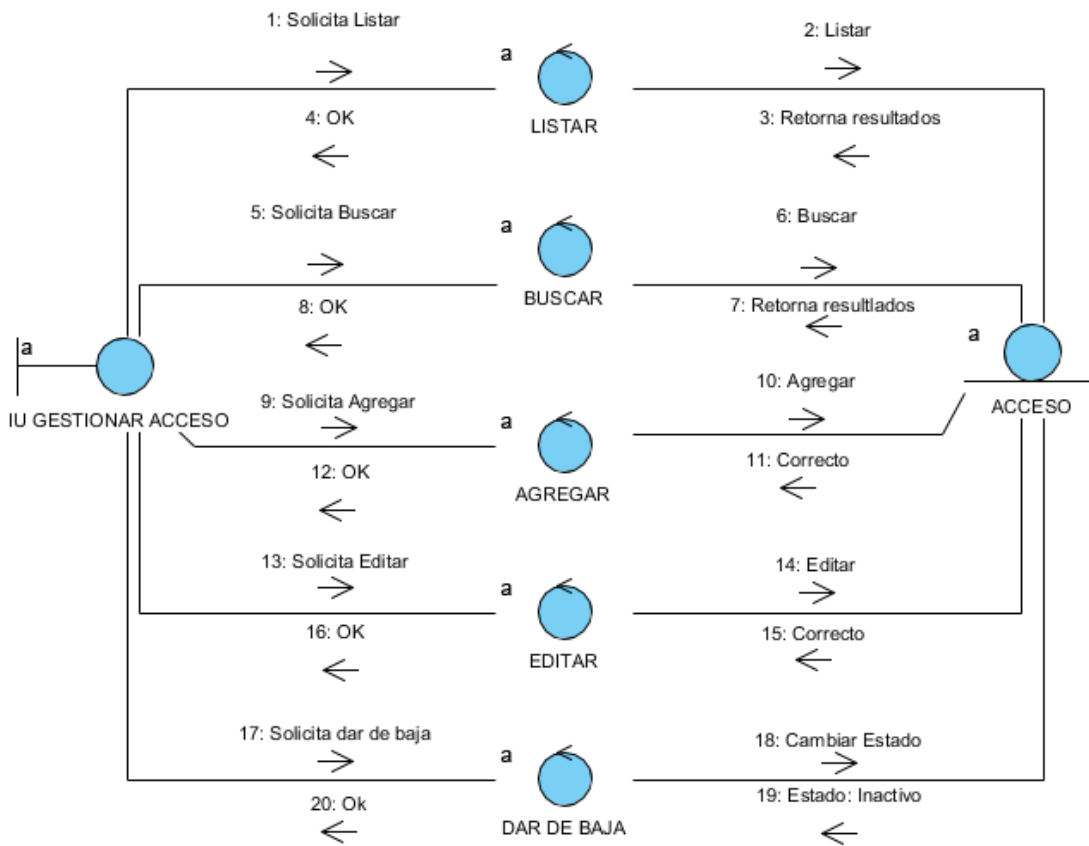
## 4.2.19. RCUA W. Gestionar Acceso

### 4.2.19.1. DCA Gestionar

sd DCA Gestionar Accesos



### 4.2.19.2. DCO Gestionar



#### 4.2.19.3. DCP Gestionar



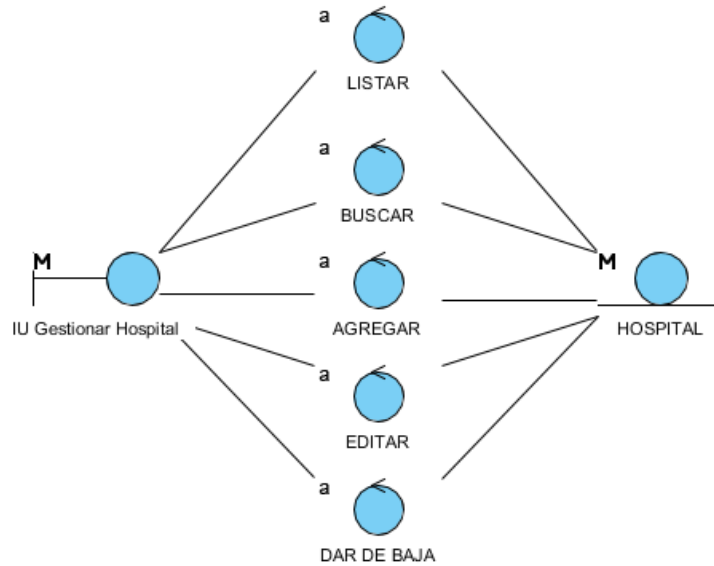
#### 4.2.19.4. DT Gestionar

- **Paso 1:** El usuario solicita listar accesos
- **Paso 2:** El sistema lista los accesos
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar accesos
- **Paso 6:** El sistema busca el acceso
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar un acceso
- **Paso 10:** El sistema agrega el acceso
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar el acceso
- **Paso 14:** El sistema edita el acceso
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a un acceso
- **Paso 18:** El sistema cambia el estado del acceso
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK

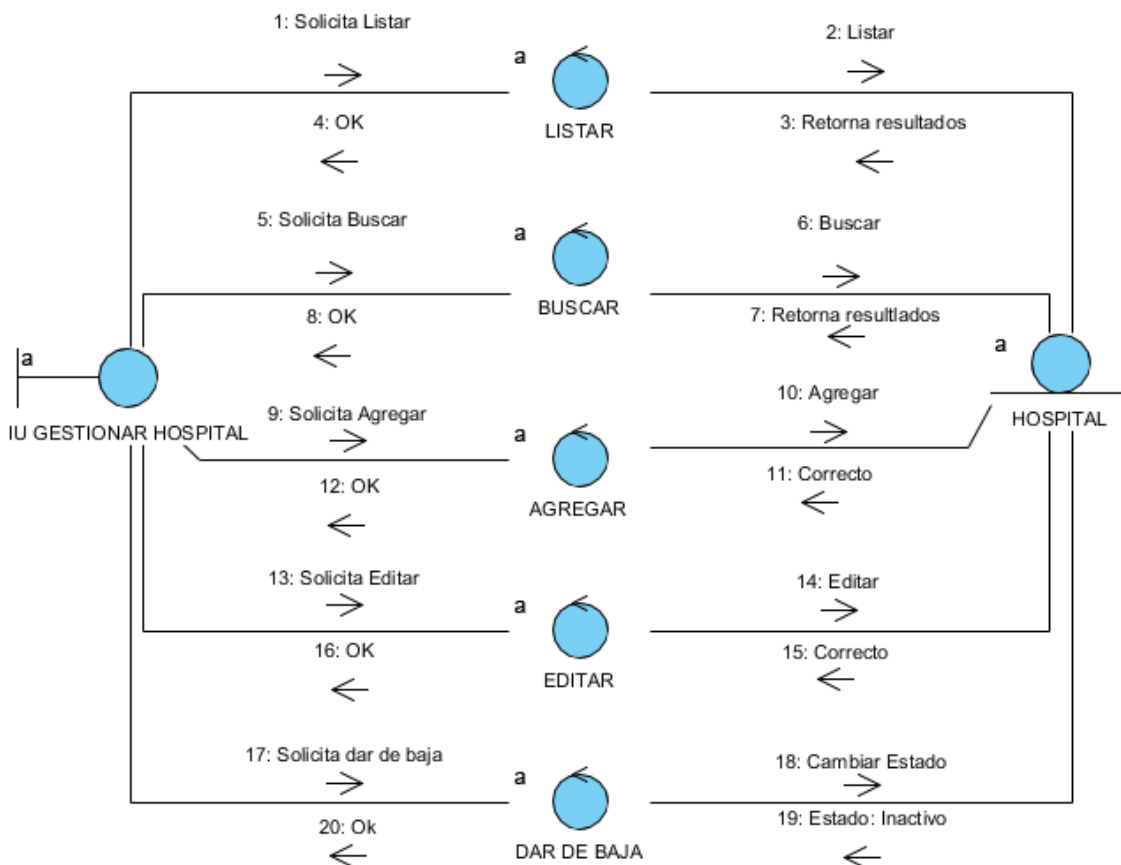
## 4.2.20. RCUA W. Gestionar Hospitales

### 4.2.20.1. DCA Gestionar

sd DCA Gestionar Hospitales



### 4.2.20.2. DCO Gestionar



#### 4.2.20.3. DCP Gestionar



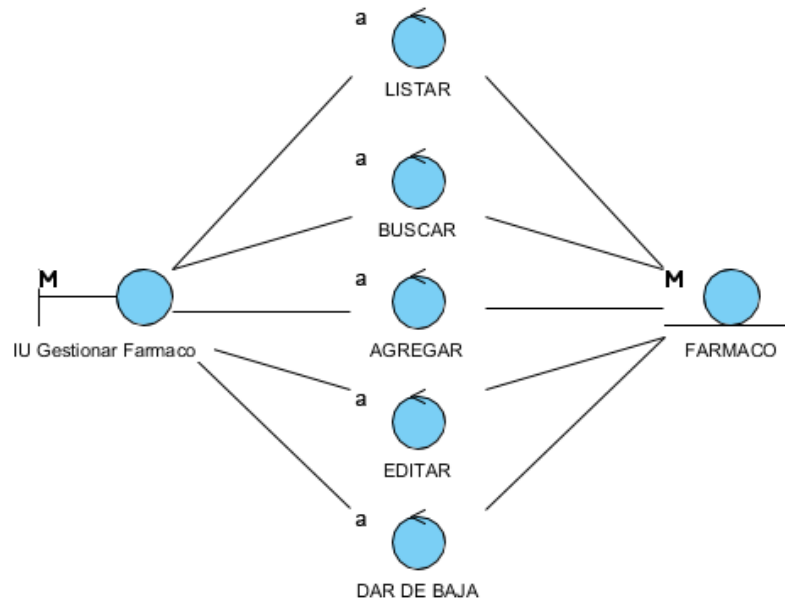
#### 4.2.20.4. DT Gestionar

- **Paso 1:** El usuario solicita listar hospitales
- **Paso 2:** El sistema lista los hospitales
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar hospitales
- **Paso 6:** El sistema busca el hospital
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar un hospital
- **Paso 10:** El sistema agrega el hospital
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar el hospital
- **Paso 14:** El sistema edita el hospital
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a un hospital
- **Paso 18:** El sistema cambia el estado del hospital
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK

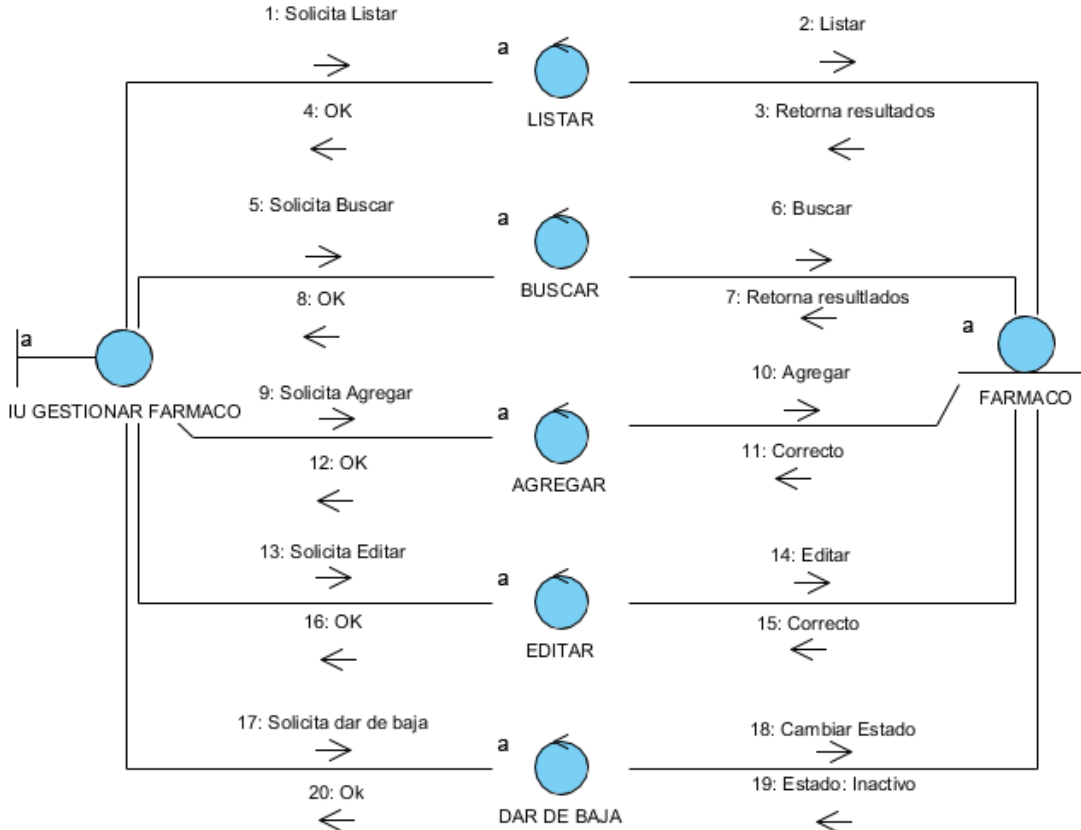
## 4.2.21. RCUA W. Gestionar fármaco

### 4.2.21.1. DCA Gestionar

sd DCA Gestionar Farmaco



### 4.2.21.2. DCO Gestionar



### 4.2.21.3. DCP Gestionar



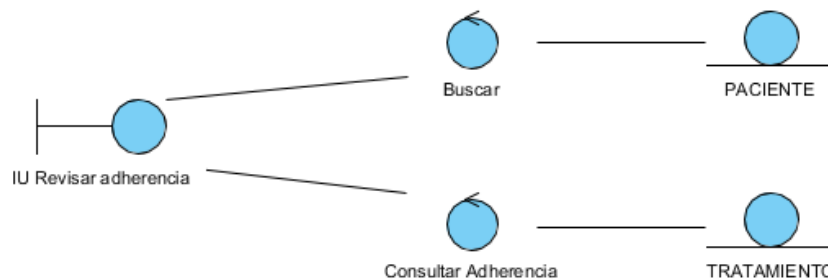
### 4.2.21.4. DT Gestionar

- **Paso 1:** El usuario solicita listar fármaco
- **Paso 2:** El sistema lista los fármacos
- **Paso 3:** El sistema retorna resultados.
- **Paso 4:** OK
- **Paso 5:** El usuario solicita buscar fármacos
- **Paso 6:** El sistema busca el fármaco
- **Paso 7:** El sistema retorna resultados
- **Paso 8:** OK
- **Paso 9:** El usuario solicita agregar un fármaco
- **Paso 10:** El sistema agrega el fármaco
- **Paso 11:** El sistema retorna agregado
- **Paso 12:** OK
- **Paso 13:** El usuario solicita editar el fármaco
- **Paso 14:** El sistema edita el fármaco
- **Paso 15:** El sistema retorna editado
- **Paso 16:** OK
- **Paso 17:** El usuario solicita dar de baja a un fármaco
- **Paso 18:** El sistema cambia el estado del fármaco
- **Paso 19:** El sistema retorna estado inactivo
- **Paso 20:** OK

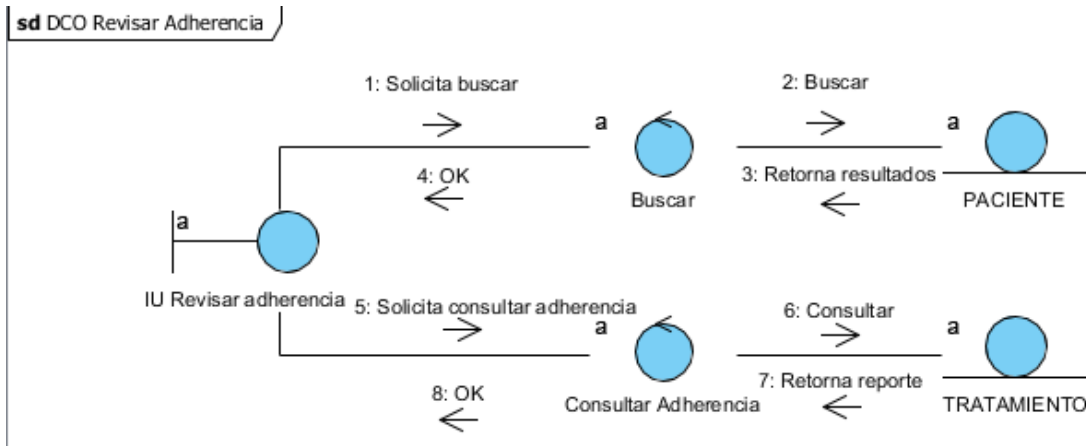
### 4.2.22. RCUA W. Revisar Adherencia

#### 4.2.22.1. DCA Revisar Adherencia

sd DCA Revisar Adherencia

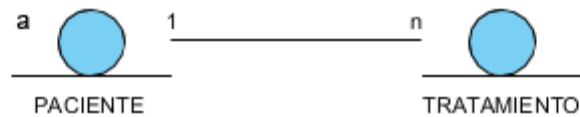


#### 4.2.22.2. DCO Revisar Adherencia



#### 4.2.22.3. DP Revisar adherencia

DCP Gestionar Paciente

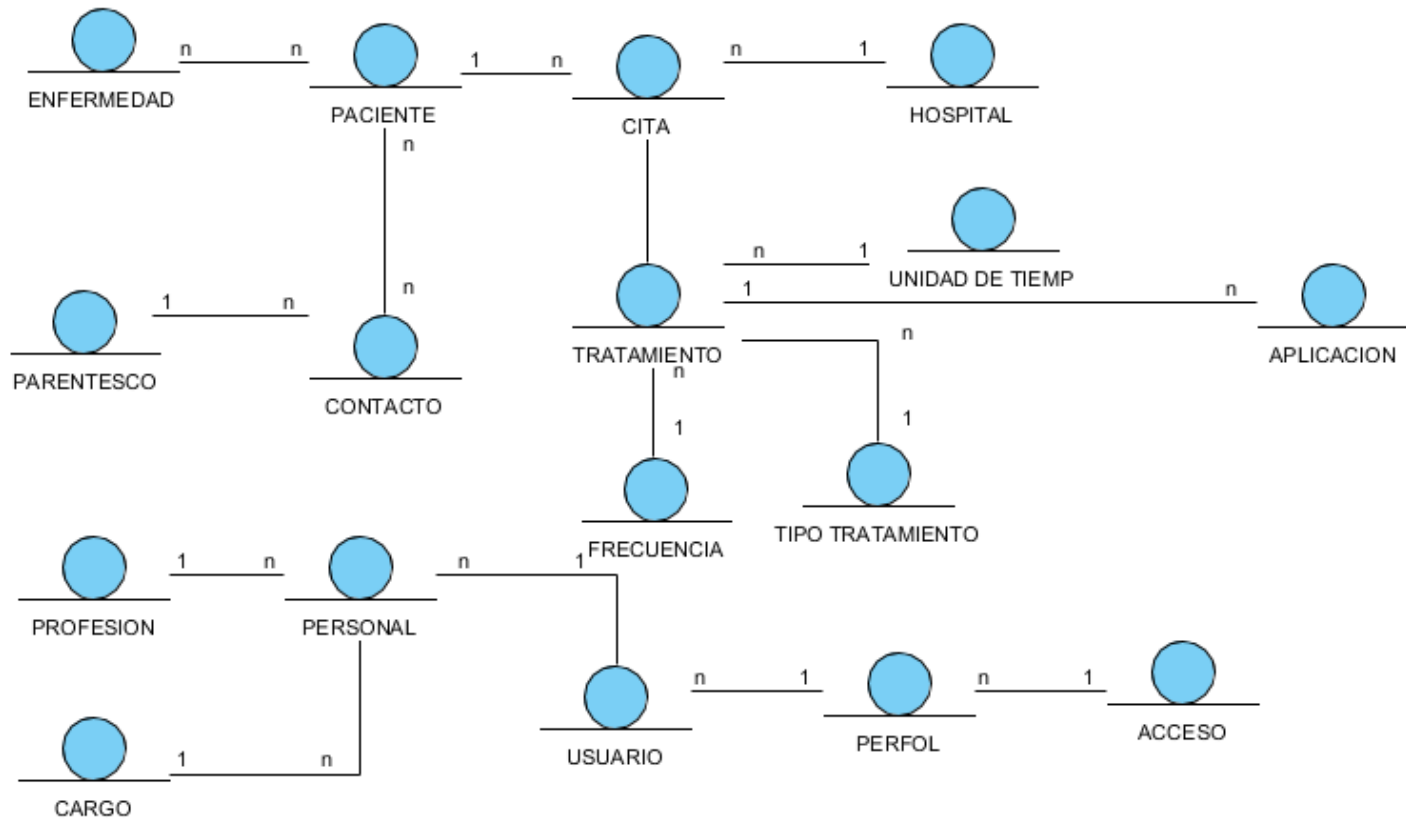


#### 4.2.22.4. DCP Revisar Adherencia

- **Paso 1:** El usuario solicita buscar paciente
- **Paso 2:** El sistema busca el paciente
- **Paso 3:** El sistema retorna resultados
- **Paso 4: OK**
- **Paso 5:** El usuario solicita consultar adherencia del tratamiento
- **Paso 6:** El sistema consulta la adherencia
- **Paso 7:** El sistema retorna reporte
- **Paso 8: OK**

### 4.2.23. DIAGRAMA DE CLASES GENERAL

sd DIAGRAMA GENERAL DE ANALISIS





#### 4.4. DESCRIPCIÓN DE LA ARQUITECTURA

- DACU M. Gestionar Paciente:
- **DACU M. Gestionar Tratamiento**
  - **DACU Consultar Tratamiento:** En esta funcionalidad el usuario podrá consultar información de un tratamiento pudiendo filtrar los datos de diferentes maneras como por ejemplo por la fecha de creación, el nombre del tratamiento, etc.
  - **DACU Agregar Tratamiento:** En esta funcionalidad el usuario podrá registrar un nuevo tratamiento para el paciente, registrar el nombre de los fármacos a consumir, cada cuánto tiempo o con qué frecuencia, por qué tiempo lo hará y si es necesario dar nota de qué doctor prescribió ese tratamiento.
  - **DACU Editar Tratamiento:** En esta funcionalidad el usuario puede editar un tratamiento bajo alguna indicación del doctor o algún factor externo del cuál debe de dar nota al momento de editarlo para dejar registro de cuándo se hizo y por qué.
  - **DACU Dar de baja Tratamiento:** En esta funcionalidad el usuario puede dar de baja a un tratamiento por indicación del médico o algún factor externo del cuál debe de dar nota al momento de dar de baja a un tratamiento. También el tratamiento será dado de baja automáticamente al terminar.
- **DACU M. Gestionar Cita médica:**
  - **DACU Consultar Cita Médica:** En esta funcionalidad el usuario podrá consultar información de una cita como la fecha, el lugar de la cita, entre otros datos; pudiendo filtrar los datos de diferentes maneras como fecha de registro o fecha de la cita, entre otros.
  - **DACU Agregar Cita Médica:** En esta funcionalidad el usuario podrá registrar una cita médica para el paciente donde debe denotar información relevante como la fecha de la cita, el lugar de la cita, el médico, e incluso ubicar el lugar de la cita en un mapa (opcional).
  - **DACU Editar Cita Médica:** En esta funcionalidad el usuario puede editar una cita médica como la fecha o el lugar, el motivo por el cuál sea editada la cita debe añadirse también para dejar registro de cuándo se hizo y por qué.
  - **DACU Dar de baja Cita Médica:** En esta funcionalidad el usuario puede dar de baja a una cita por indicación del médico o algún factor externo del cuál debe de dar nota al momento de dar de baja a la cita. También la cita será dada de baja automáticamente al pasar el día de esta.
  - **DACU Ubicar Cita Médica en el Mapa:** Esta funcionalidad es un adicional de la anterior donde se puede ubicar el lugar de la cita médica en el mapa y la aplicación lo interpretará como coordenadas que serán guardadas en la base de datos. Esta funcionalidad es adicional y no es indispensable para agregar una cita Médica.
- **DACU Buscar Farmacia:** Esta funcionalidad es un adicional para ubicar las farmacias más cercanas cuando el usuario se encuentre en algún lugar

desconocido o haya viajado justo con el paciente y necesite comprar medicamentos. La app móvil debe mostrar la ubicación exacta del usuario mediante el GPS y el API de Google Maps, además debe filtrar los indicadores de lugares para reducirlo a farmacias cercanas.

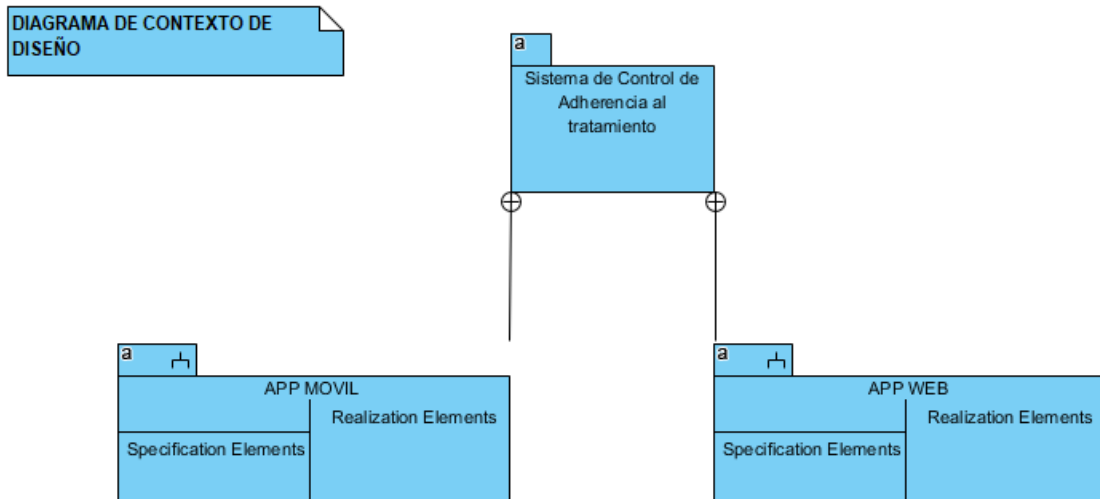
- **DACU M. Gestionar Contacto:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información del contacto que está asignado al paciente y sus diferentes datos que sean visibles para el asistente.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar los datos de un nuevo contacto que necesite ser enlazado con el paciente.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información habilitada para esta acción por parte del asistente.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja (eliminación lógica – cambio de estado) a un contacto del paciente.
- **DACU M. Añadir un contacto a un paciente:** en esta funcionalidad el usuario podrá enlazar un contacto a un paciente. En caso no esté registrado, puede añadirlo como un nuevo contacto y enlazarlo.
- **DACU M. Añadir una enfermedad a un paciente:** en esta funcionalidad el usuario podrá enlazar una enfermedad a un paciente. En caso no esté registrada se le comunicará a al administrado y este podrá añadirla. El asistente (usuario) NO podrá añadir una enfermedad.
- **DACU M. Gestionar cumplimiento:** En esta funcionalidad el asistente podrá registrar si el paciente tomó la pastilla en el momento que debería. Eso se facilita por la notificación push que se implementará.
- **DACU W. Gestionar Usuarios:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los usuarios.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo usuario al que puede añadirle un login y una clave para el ingreso al sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un usuario. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un usuario denegando el ingreso al sistema por el cambio de estado de Activo a Inactivo.
- **DACU W. Gestionar Perfiles:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los perfiles en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo perfil ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un perfil. El sistema debe habilitar todos los campos disponibles para su edición.

- **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un perfil cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Cargo:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los cargos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo cargo ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un cargo. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un cargo cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Unidad de tiempo:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de las Unidades de Tiempo en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar una nueva unidad de tiempo ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de una unidad de tiempo. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja una unidad de tiempo cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Frecuencia:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de las Frecuencias en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar una nueva frecuencia ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de una frecuencia. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja una frecuencia cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Parentesco:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los parentescos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo parentesco ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un parentesco. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un parentesco cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Profesiones:**

- **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de las profesiones en detalle.
- **DACU Agregar:** En esta funcionalidad el usuario podrá registrar una nueva profesión ingresando los datos requeridos por el sistema.
- **DACU Editar:** En esta funcionalidad el usuario podrá editar información de una profesión perfil. El sistema debe habilitar todos los campos disponibles para su edición.
- **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja una profesión cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Acceso:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los accesos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo acceso ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un acceso. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un acceso cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Hospital:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los hospitales en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo hospital ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un hospital. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un hospital cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Fármaco:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los fármacos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo fármaco ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un fármaco. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un fármaco cambiando su estado de Activo a Inactivo.
- **DACU W. Revisar Adherencia:** en esta funcionalidad el usuario doctor podrá ver el progreso de su paciente respecto a cómo está llevando su tratamiento una vez registrado. Esto se presentará a forma de gráficas en vista web para ser más cómoda su revisión.

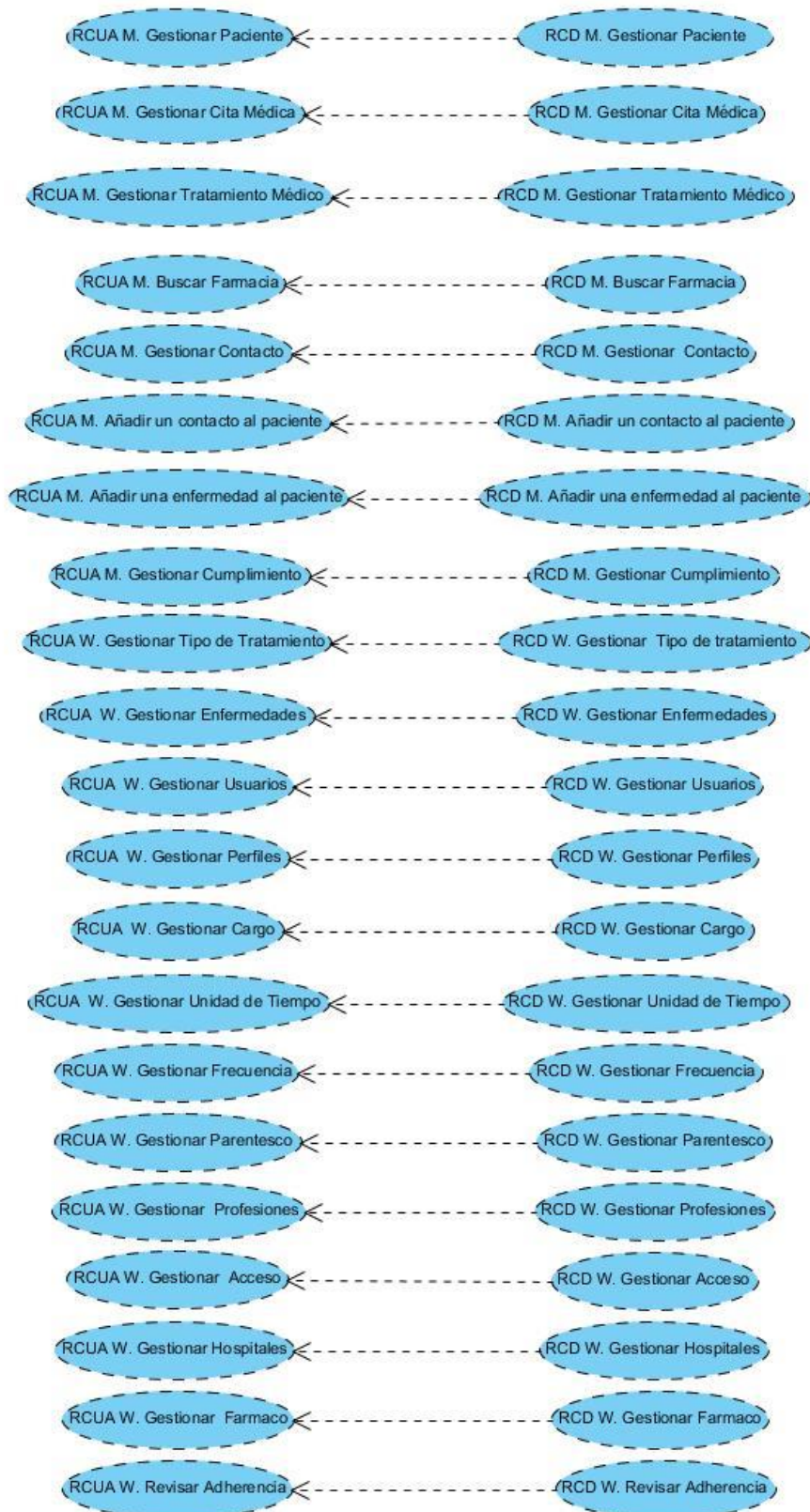
## 5. CAPITULO V: DISEÑO

### 5.1. SUBSISTEMA DE DISEÑO (EN FUNCIÓN A PAQUETES)



### 5.2. DIAGRAMA DE REALIZACIONES DE CASOS DE USO DE DISEÑO

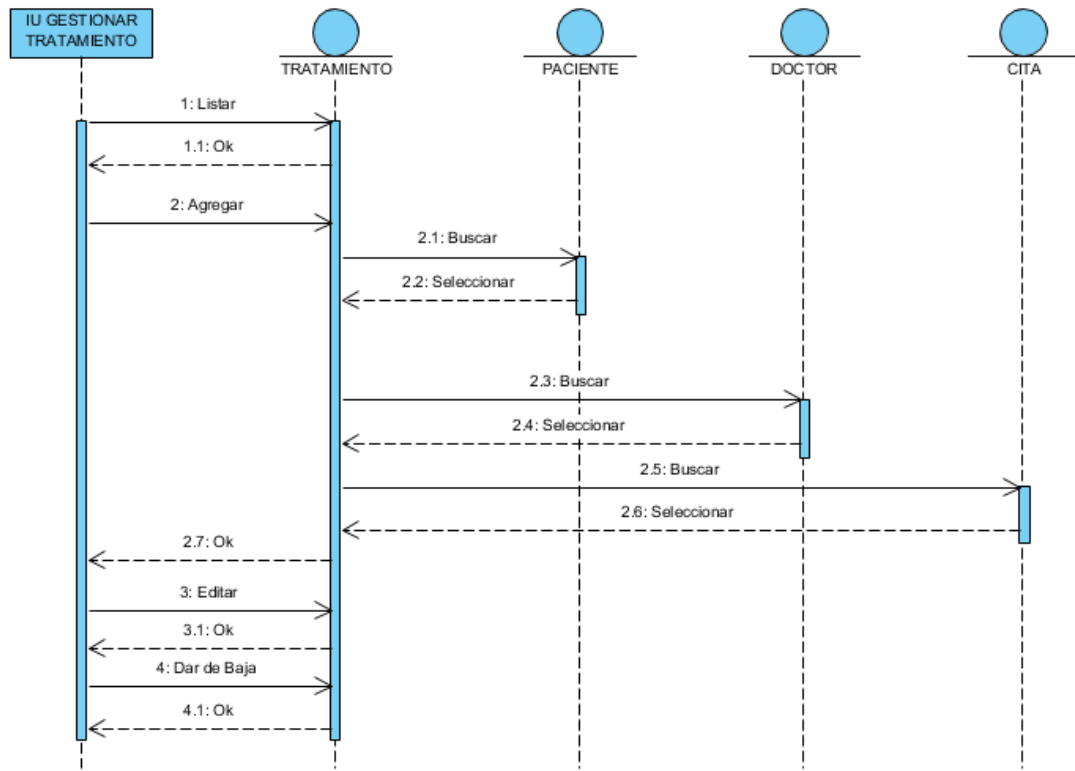
DIAGRAMA DE REALIZACION DE CASOS DE USO DE DISEÑO



### 5.3. DIAGRAMA DE SECUENCIA

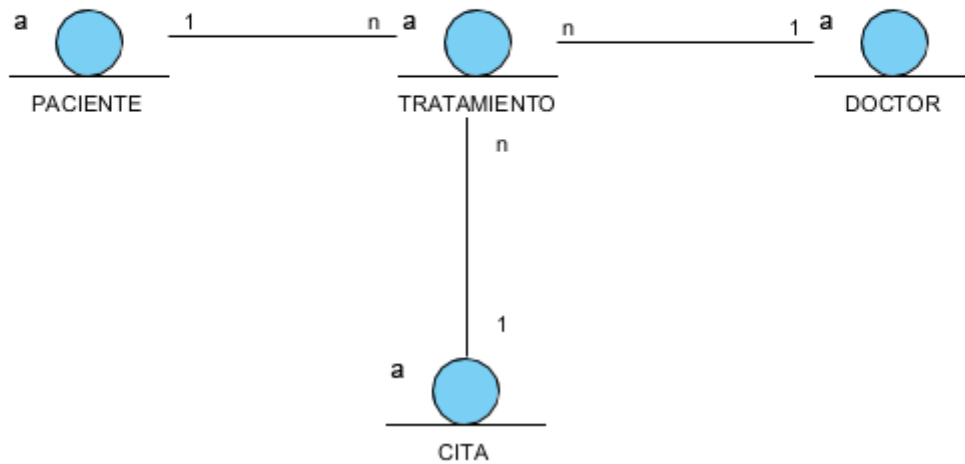
#### 5.3.1. DS GESTIONAR TRATAMIENTO

sd RCUJ GESTIONAR TRATAMIENTO



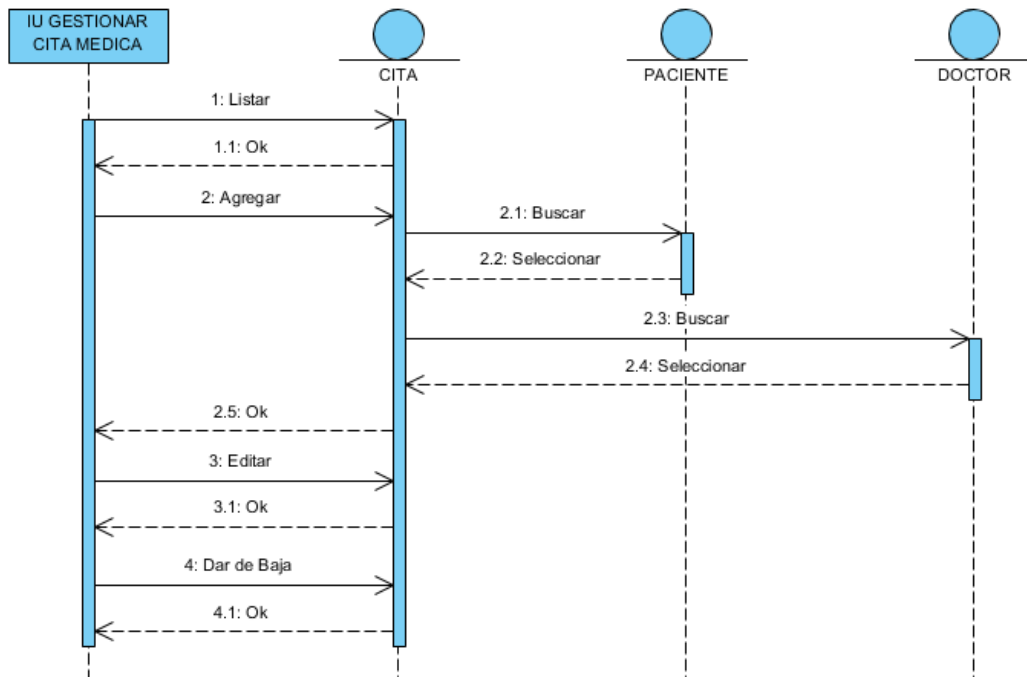
#### 5.3.1.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO

DC PARCIALES GESTIONAR TRATAMIENTO



### 5.3.2. DS GESTIONAR CITA MEDICA

sd RCUD GESTIONAR CITA MEDICA



#### 5.3.2.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO

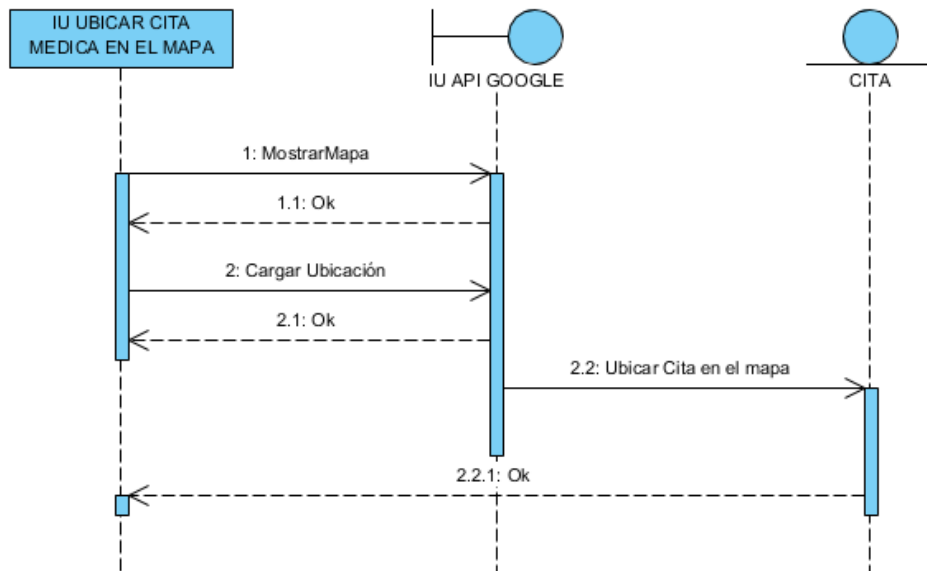
DC PARCIALES GESTIONAR CITA MÉDICA





### 5.3.3. DS UBICAR CITA MEDICA EN EL MAPA

sd RCUJ UBICAR CITA MEDICA EN EL MAPA



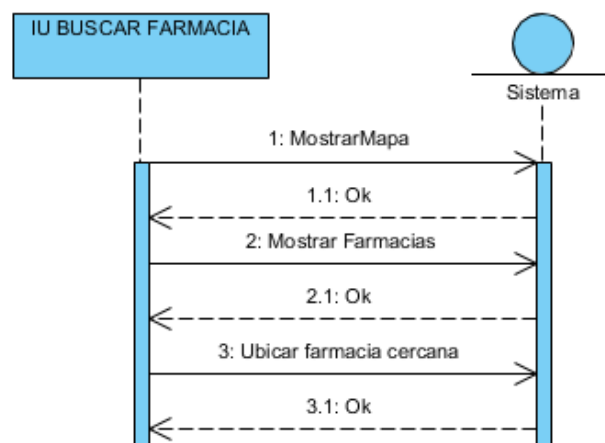
#### 5.3.3.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO

DC PARCIALES UBICAR CITA MEDICA EN EL MAPA



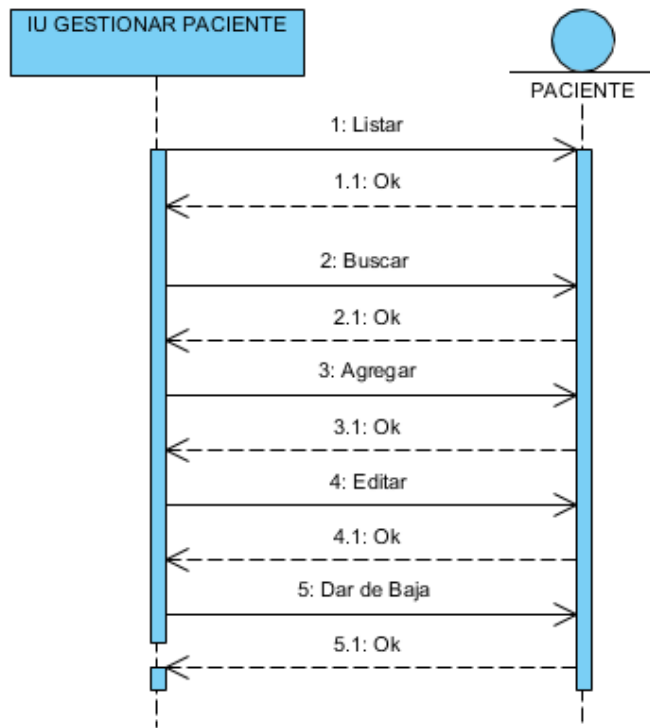
### 5.3.4. DS Buscar Farmacia

sd RCUJ BUSCAR FARMACIA



### 5.3.5. DS M. Gestionar Paciente

sd DS M. Gestionar Paciente

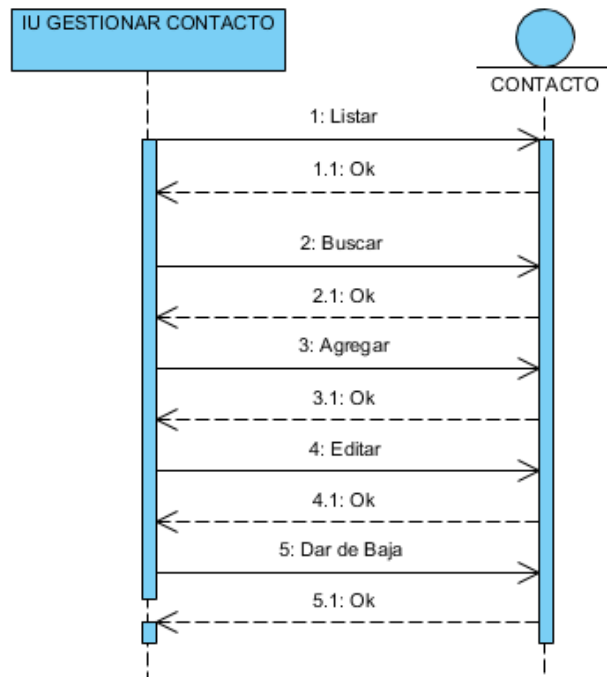


#### 5.3.5.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO



### 5.3.6. DS M. Gestionar Contacto

sd DS Gestionar Contacto

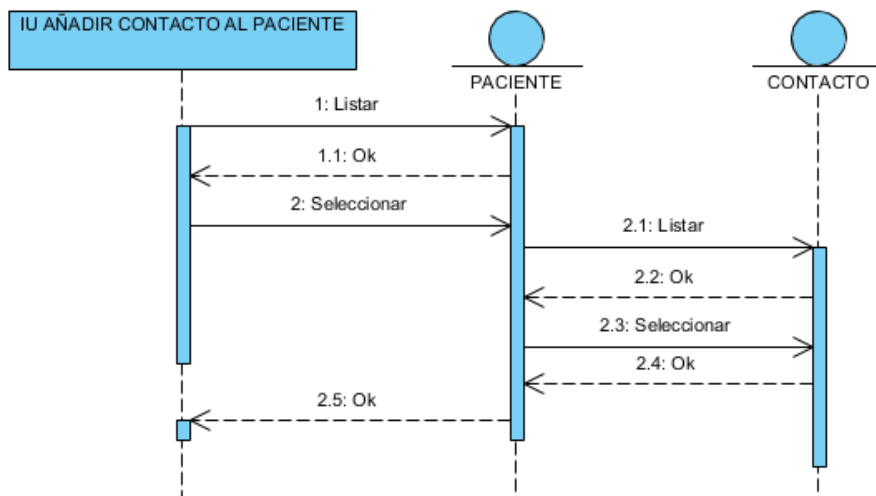


#### 5.3.6.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO



### 5.3.7. DS M. Añadir un contacto al paciente

sd DS M. Añadir un contacto al paciente



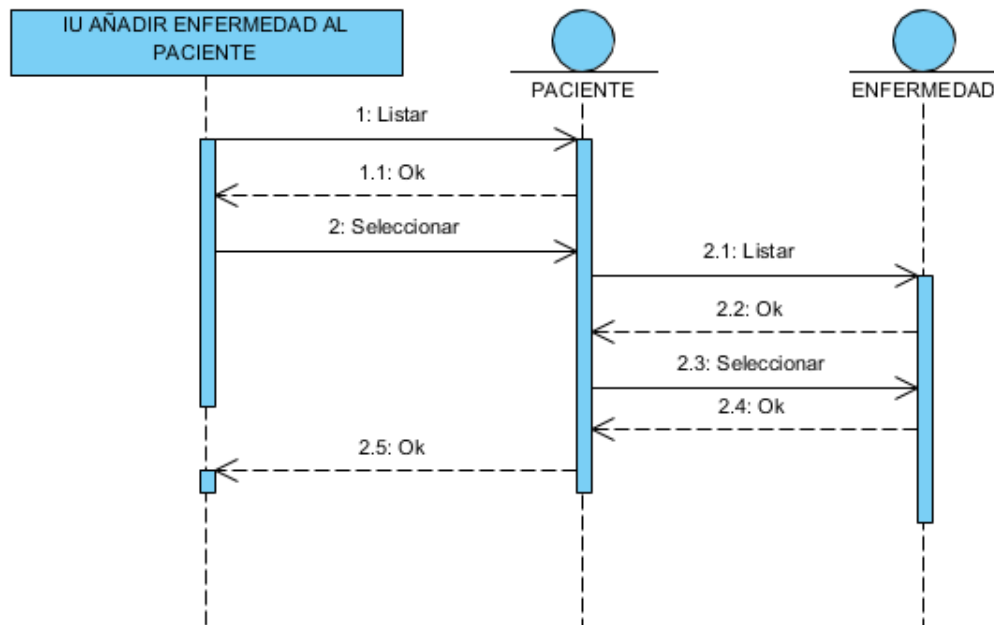
### 5.3.7.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO

sd DCP Añadir un contacto al paciente



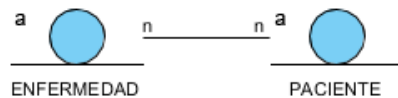
### 5.3.8. DS M. Añadir una enfermedad al paciente

sd DS M. Añadir una enfermedad al paciente



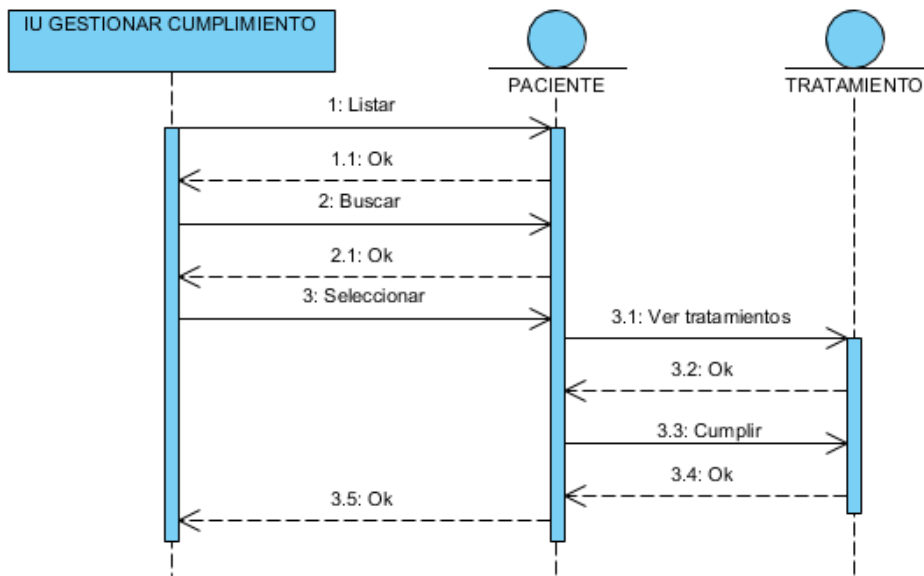
### 5.3.8.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO

sd DCP Añadir una enfermedad al paciente



### 5.3.9. DS W. Gestionar Cumplimiento

sd DS M. Gestionar Cumplimiento)



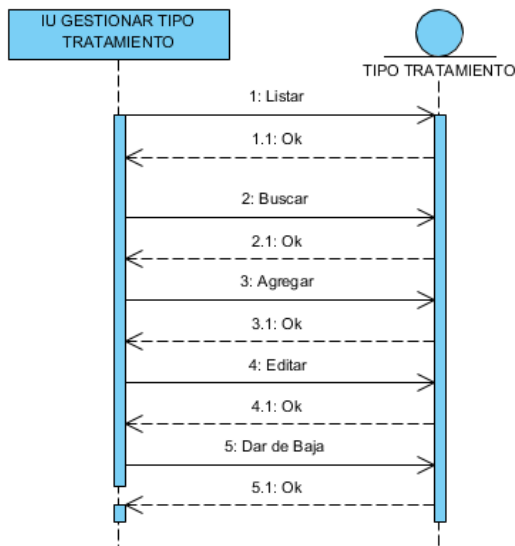
#### 5.3.9.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO

sd DCP Gestionar Cumplimiento)

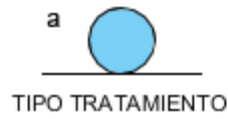


### 5.3.10. DS W. Gestionar Tipo de tratamiento

sd DS W. Gestionar Tipo de tratamiento)

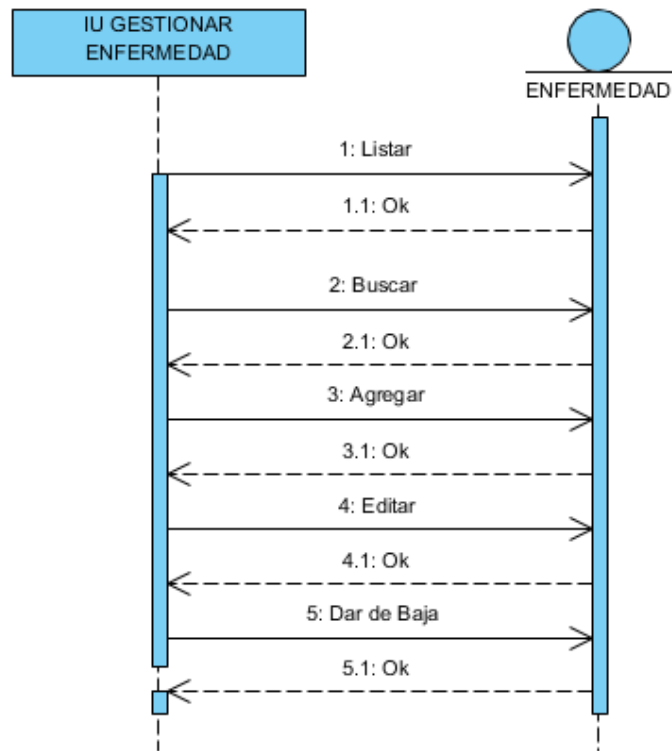


### 5.3.10.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO



### 5.3.11. DS W. Gestionar Enfermedades

sd DS W. Gestionar Enfermedades

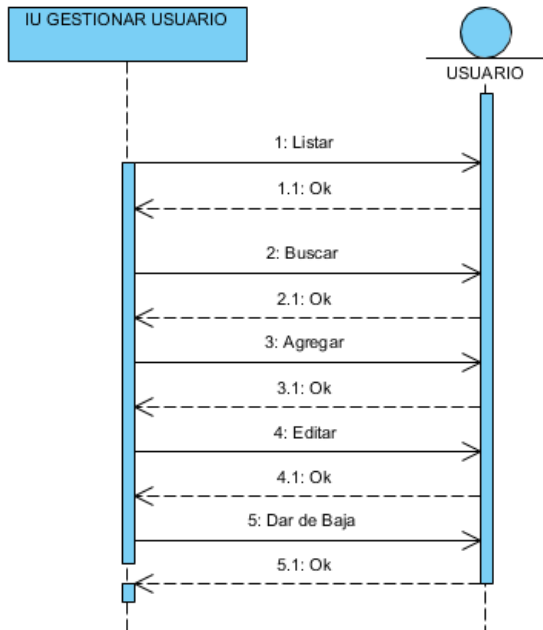


### 5.3.11.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO



### 5.3.12. DS W. Gestionar Usuarios

sd DS W. Gestionar Usuarios

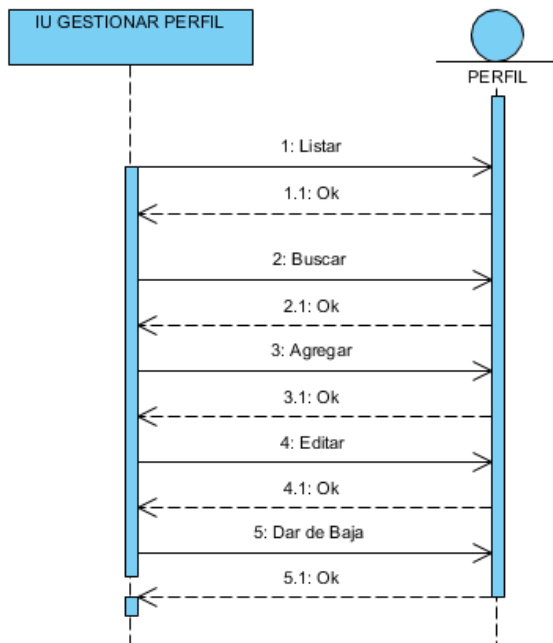


#### 5.3.12.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO

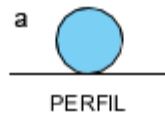


### 5.3.13. DS W. Gestionar Perfiles

sd DS W. Gestionar Perfiles

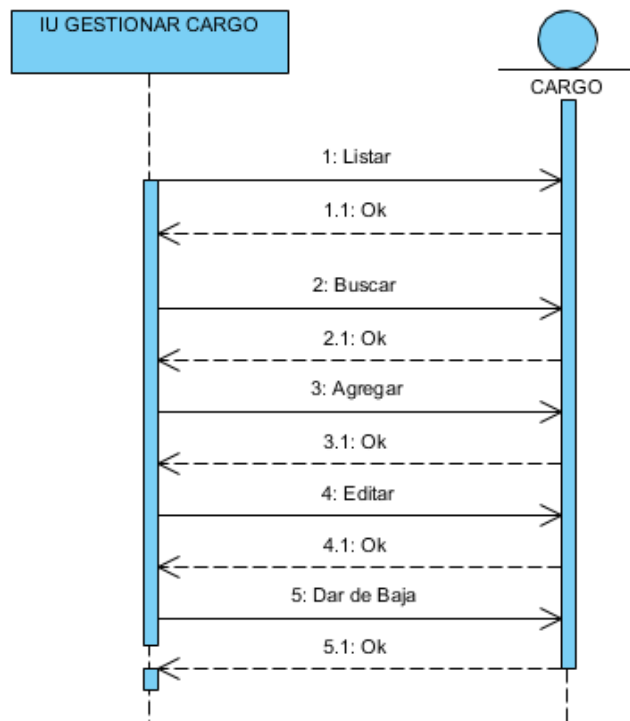


### 5.3.13.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO



### 5.3.14. DS W. Gestionar Cargo

sd DS W. Gestionar Cargo



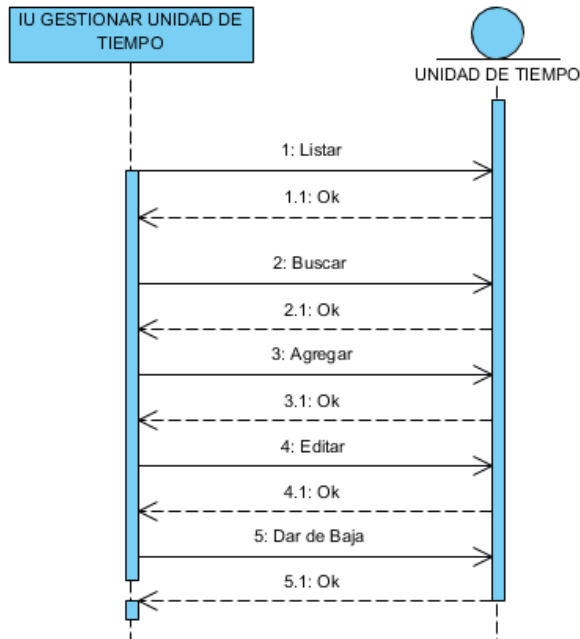
### 5.3.14.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO





### 5.3.15. DS W. Gestionar Unidad de tiempo

sd DS W. Gestionar Unidad de Tiempo

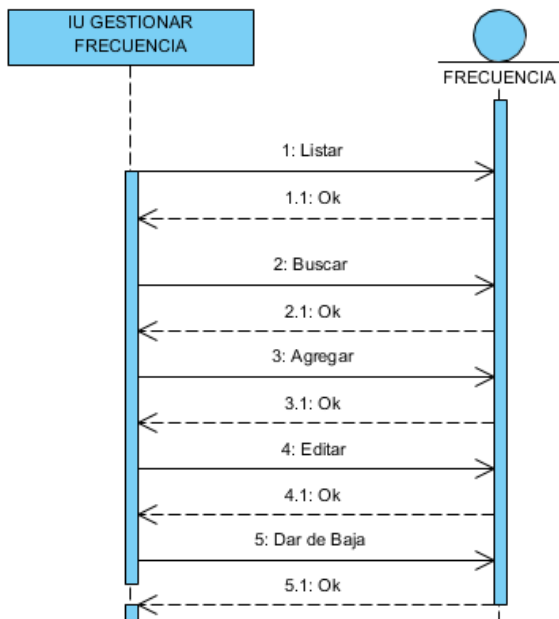


#### 5.3.15.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO

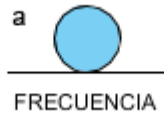


### 5.3.16. DS W. Gestionar Frecuencia

sd DS W. Gestionar Frecuencia

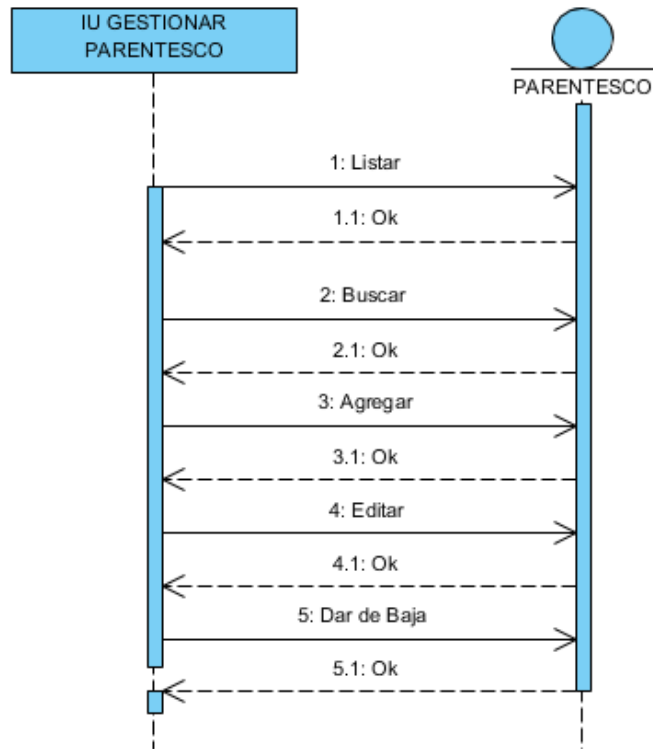


### 5.3.16.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO



### 5.3.17. DS W. Gestionar Parentesco

sd DS W. Gestionar Parentesco

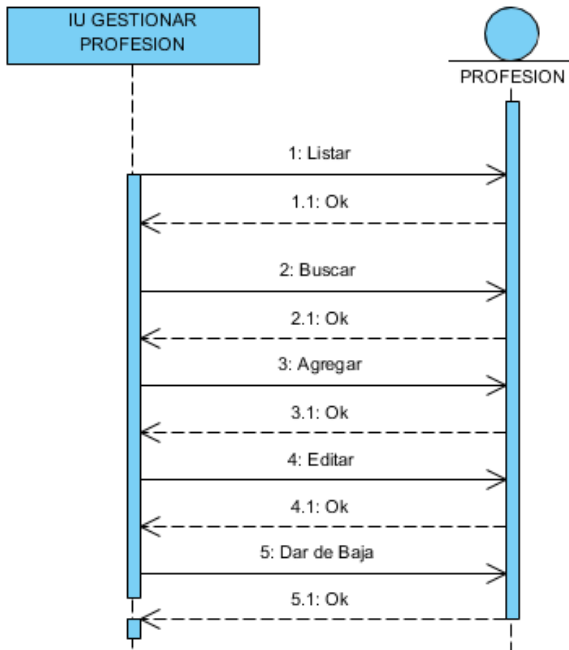


### 5.3.17.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO



### 5.3.18. DS W. Gestionar Profesiones

sd DS W. Gestionar Profesiones

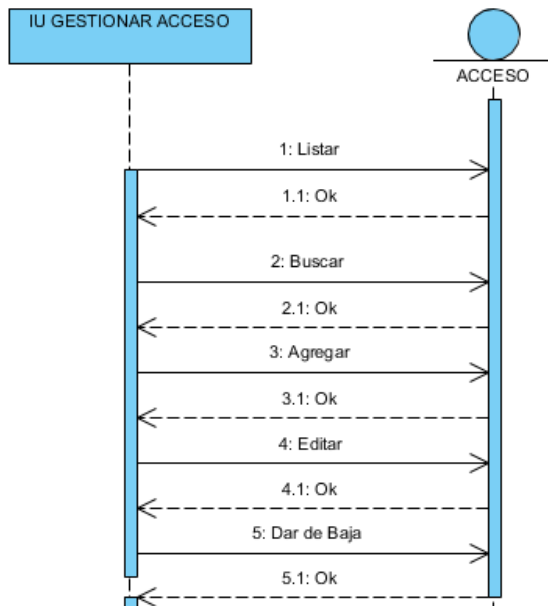


#### 5.3.18.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO



### 5.3.19. DS W. Gestionar Acceso

sd DS W. Gestionar Acceso

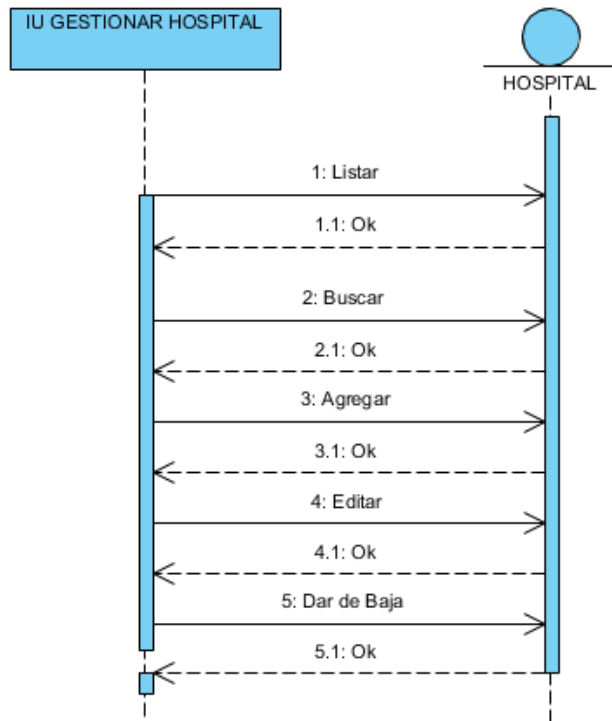


### 5.3.19.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO



### 5.3.20. DS W. Gestionar Hospital

sd DS W. Gestionar Hospital

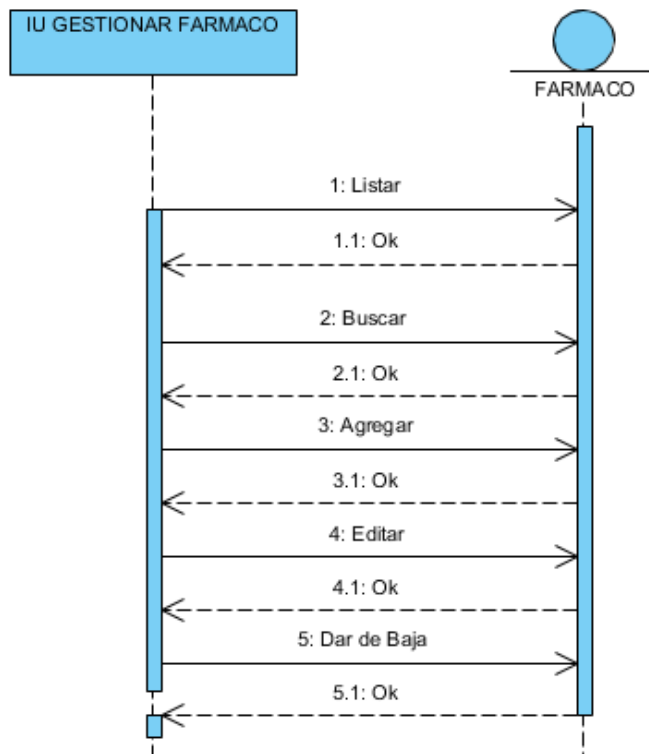


### 5.3.20.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO



### 5.3.21. DS W. Gestionar Fármaco

sd DS W. Gestionar Farmaco

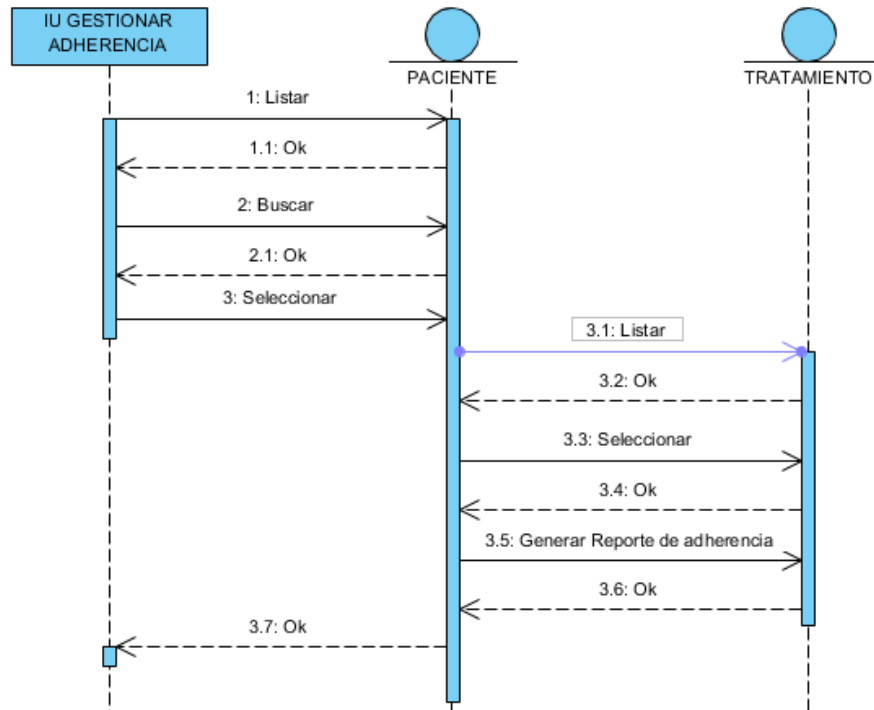


#### 5.3.21.1. DIAGRAMA PARCIAL DE CLASES DE DISEÑO



### 5.3.22. DS W. Gestionar Adherencia

sd DS W. Gestionar Adherencia

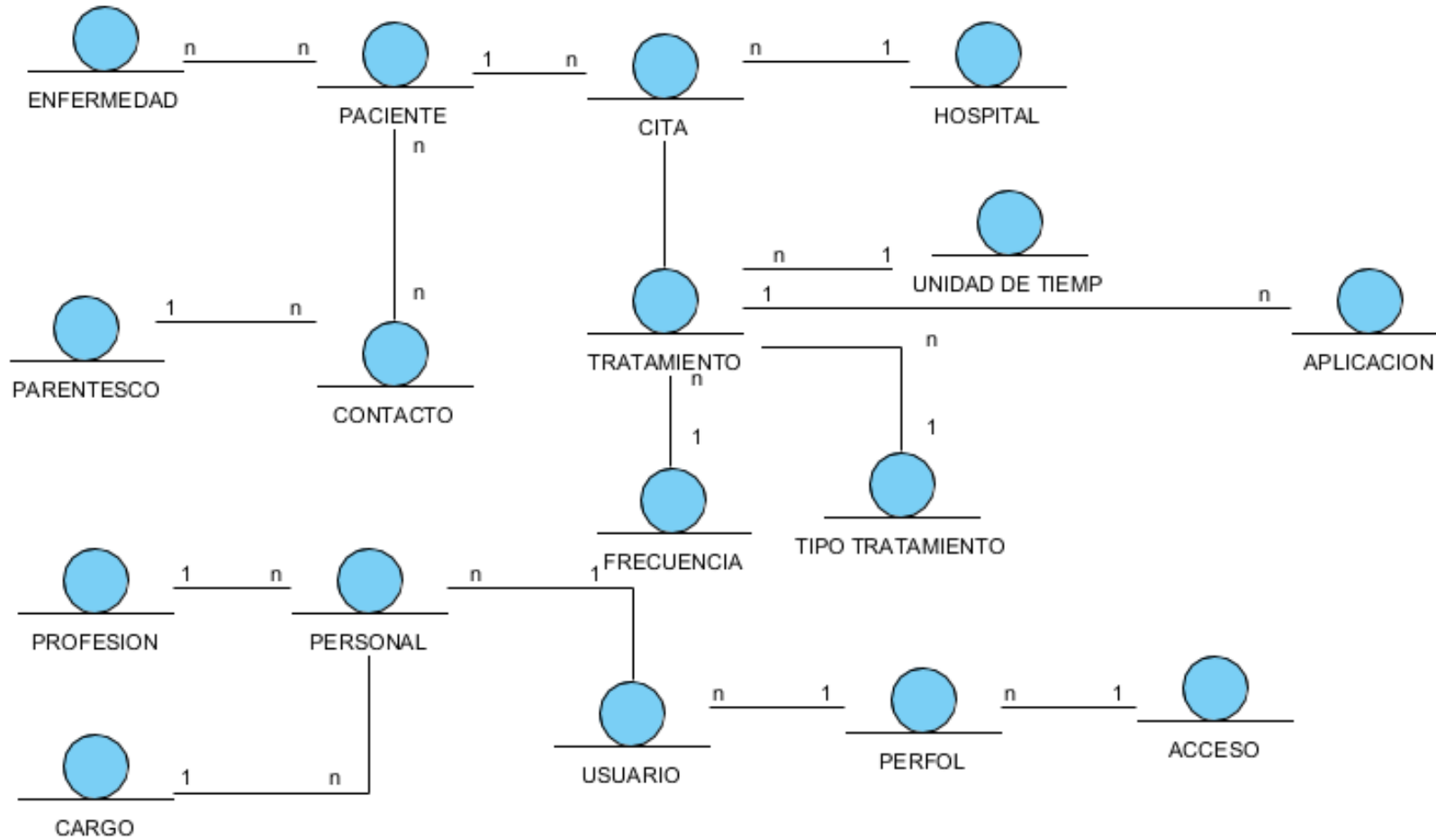


#### 5.3.22.1. DIAGRAMA DE CLASES PARCIALES

DCP Gestionar Paciente

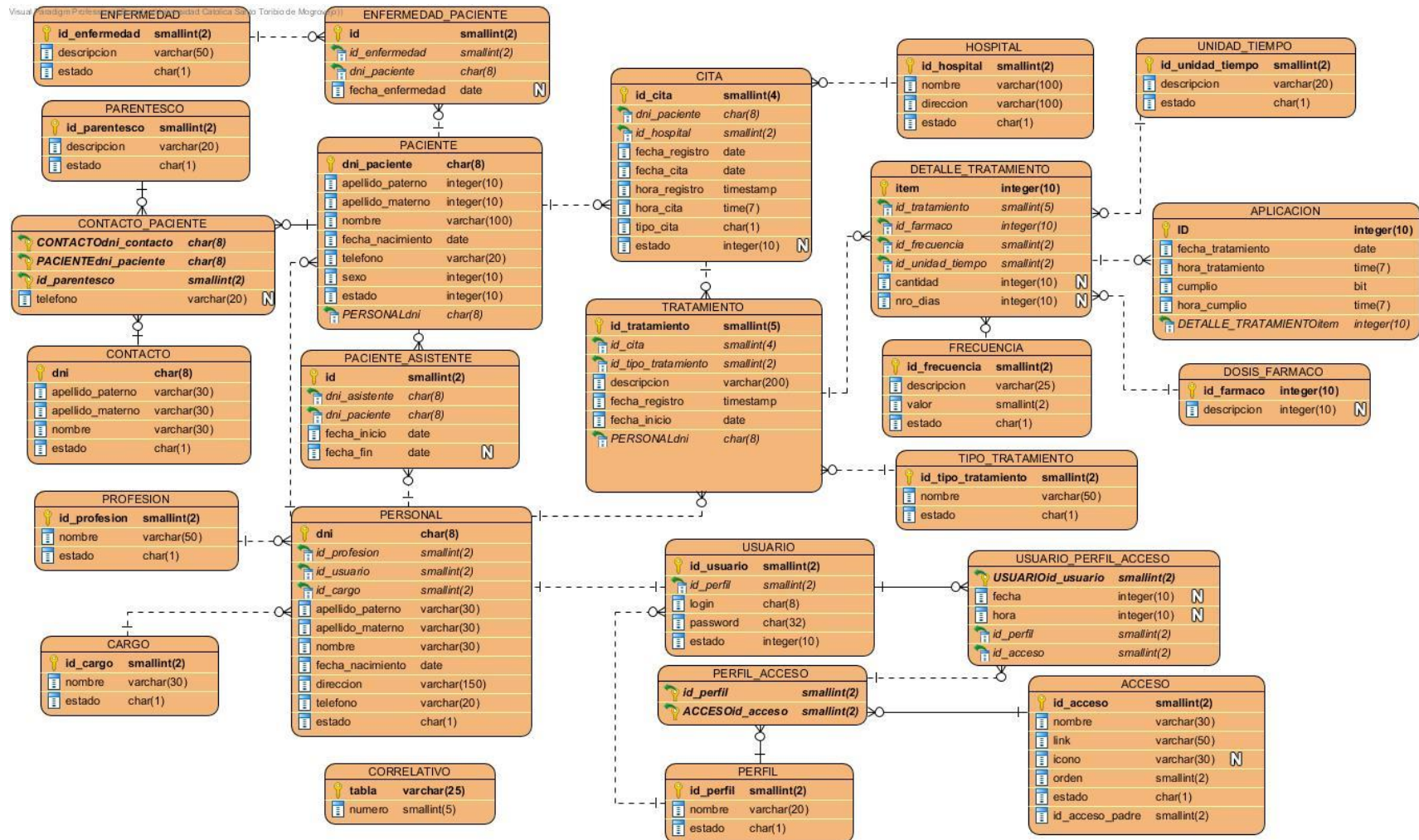


#### 5.4. DIAGRAMA GENERAL DE CLASES DE DISEÑO



## 5.5. DISEÑO DE LA BASE DE DATOS

### 5.5.1. DISEÑO FISICO





### 5.5.2. DISEÑO LOGICO

PARENTESCO	ID_PARENTESCO	DESCRIPCION	ESTADO
PK/FK	PK	-	-
NN/U	NN/U1	NN/U2	NN
TPO DE DATO	smallint	varchar (20)	char(1)
EJEMPLOS	1	Padre	A
	2	Madre	A
	3	Suegro/a	A
	4	Hijo/a	A
	5	Yerno	A
	6	Nuera	A
	7	Abuela/o	A
	8	Nieto/a	A
	9	Hermano/a	A
	10	Cuñado/A	A
	11	Bisabuelo/a	A
	12	Biznieto/a	A
	13	Tio/a	A
	14	Sobrino/a	A

ENFERMEDAD	ID_ENFERMEDAD	DESCRIPCION	ESTADO
PK/FK	PK	-	-
NN/U	NN/U	NN/U	NN
TIPO DE DATO	smallint	varchar(50)	char(1)
EJEMPLO	1	Hipertensión Arterial (HTA)	A
	2	Diabetes	A
	3	Tuberculosis	A
	4	Artrosis	A
	5	Lumbalgia	I

PROFESION	ID_PROFESION	DESCRIPCION	ESTADO
PK/FK	PK	-	-
NN/U	NN/U	NN/U	NN
TIPO DE DATO	smallint	varchar(50)	char(1)
EJEMPLOS	1	Enfermero/a	A
	2	Técnico en Enfermería	A
	3	Médico General	A
	4	Cuidador profesional	I
	5	Terapeuta Físico	A
	6	Gerontólogo	A

	7	Técnico en asistencia al adulto mayor	A
--	---	---------------------------------------	---

HOSPITAL	ID_HOSPITAL	NOMBRE	ESTADO
PK/FK	PK	-	-
NN/U	NN/U	NN/U	NN
TIPO DE DATO	smallint	varchar(50)	char(1)
EJEMPLOS	1	HOSPITAL "LAS MERCEDES" CHICLAYO	A
	2	HOSPITAL REGIONAL LAMBAYEQUE	A
	3	Hospital Almanzor Aguinaga Asenjo	I
	4	Hospital Naylamp	A
	5	Policlínico Chiclayo Oeste	A

DOCTOR	DNI	APELLIDO_PATERNO	APELLIDO_MATERNO	NOMBRE	TELEFONO	ESTADO
PK/FK	PK	-	-	-	-	-
NN/U	NN/U	NN	NN	NN	NN/-	NN
TIPO DE DATO	char(8)	varchar(20)	varchar(20)	varchar(20)	varchar(20)	char(1)
EJEMPLOS	59653744	Perez	López	Jesús	97865477	A
	96543542	López	Dávila	Martín	978654234	A
	11324366	Guzman	Carrasco	Juan	978006543	A
	67546534	Guevara	Gonzales	Luís	978546321	A
	45674309	Silva	Porturas	Ricardo	956485325	I

UNIDAD_MEDIDA	ID_UNIDAD_MEDIDA	DESCRIPCION	ESTADO
PK/FK	PK	-	-
NN/U	NN/U	NN/U	NN
TIPO DE DATO	smallint	varchar(50)	char(1)
EJEMPLO	1	Cucharada	A
	2	Gota	A
	3	Cápsula	A
	4	Ampolla	I
	5	Pastilla	A

TIPO_TRATAMIENTO	ID	DESCRIPCION	ESTADO
PK/FK	PK	-	-

TIPO DE DATO	smallint	varchar(20)	char(1)
EJEMPLO	1	Farmacológico	A
	2	Higiénico	A
	3	Físico	A
	4	De sostén	I
	5	Terapéutico	A

UNIDAD_TIEMPO	ID_FRECUENCIA	DESCRIPCION	ESTADO
PK/FK	PK	-	-
TIPO DE DATO	smallint	varchar(15)	char(1)
EJEMPLO	1	hora(s)	A
	2	dia(s)	A
	3	semana(s)	A

FRECUENCIA	ID_FRECUENCIA	VALOR	ID_UNIDAD_TIEMPO	ESTADO
PK/FK	PK	-	FK	-
TIPO DE DATO	smallint	smallint	smallint	char(1)
EJEMPLO	1	1	1	A
	2	2	1	A
	3	4	1	A
	4	6	1	A
	5	8	1	A
	6	12	1	A
	7	1	2	A
	8	2	2	A
	9	1	3	A

CONTACTO	DNI	APELLIDO_PATERNO	APELLIDO_MATERNO	NOMBRE	ID_PARENTESCO	ESTADO
PK/FK	PK	-	-	-	FK	-
TIPO DE DATO	char(8)	varchar(20)	varchar(20)	varchar(20)	smallint	char(1)
EJEMPLOS	56457656	Guevara	Gonzales	Hilda	2	A

34875580	Perez	Gutierrez	Luis	6	A
45435465	Lopez	Quispe	Timy	8	A
57875893	Perez	Torres	Alan	9	A
47658710	Vilchez	Santisteban	Enrique	9	I

PACIENTE	DNI	APELLIDO_P ATERNO	APELLIDO_M ATERNO	NOMBRE	FECHA_NAC	TELEFONO	SEXO	ESTADO
PK/FK	PK	-	-	-	-	-	-	-
NN/U	NN/U	NN	NN	NN	NN	NN/U	NN	NN
TIPO DE DATO	smallint	varchar(20)	varchar(20)	varchar(20)	DATE	varchar(20)	char(1)	char(1)
EJEMPLOS	76857434	Vilchez	Ramirez	Linda	01/01/1999	978658456	F	A
	58657432	Palma	Gonzales	Pepe	03/02/1963	978607321	M	A
	68795847	Flores	Cerna	Richard	08/05/1954	987654321	M	A
	68553209	Peña	Rodriguez	Julio	04/09/1950	908965456	M	A
	90795743	Sernaqué	Prado	Viviana	01/09/1951	910234654	F	A

<u>ENFERMEDAD - PACIENTE</u>	<u>ID_ENFERMEDAD</u>	<u>DNI_PACIENTE</u>	<u>FECHA_DIAGNOSTICO</u>
PK/FK	PK1/FK1	PK1/FK2	PK1
NN/U	NN/U1	NN/U1	NN/U1
TIPO DE DATO	smallint	char(8)	DATE
EJEMPLO	<u>1</u>	<u>76857434</u>	<u>04/10/2010</u>
	<u>2</u>	<u>76857434</u>	<u>02/11/2011</u>
	<u>3</u>	<u>68795847</u>	<u>10/12/2005</u>
	<u>5</u>	<u>68553209</u>	<u>04/01/2001</u>
	<u>4</u>	<u>90795743</u>	<u>24/10/2008</u>
	<u>5</u>	<u>68553209</u>	<u>11/05/2018</u>

CONTACTO_PACIENTE	DNI_CONTACTO	DNI_PACIENTE	TELEFONO
PK/FK	PK1/FK1	PK1/FK2	-
NN/U	NN/U1	NN/U1	NN
TIPO DE DATO	char(8)	char(8)	varchar(20)
EJEMPLO	56457656	58657432	978547654
	56457656	45435465	981103456
	34875580	76857434	934286543
	34875580	90795743	947584362
	47658710	68553209	968325794

ASISTENTE	DNI	APELLIDO_PATERNO	APELLIDO_MATERNO	NOMBRE	FECHA_NAC	TELEFONO	DIRECCION	ESTADO	ID_PROFESION
PK/FK	PK	-	-	-	-	-	-	-	FK
NN/U	NN/U	NN	NN	NN	NN	NN/-	NN	NN	NN
TIPO DE DATO	smallint	varchar(20)	varchar(20)	varchar(20)	smallint	varchar(20)	varchar(50)	char(1)	smallint
EJEMPLOS	78576543	Martinez	Gomez	Alexandre	01/01/1999	978658456	Graulin10	A	1
	34546577	Gonzales	Basadre	Andres	03/02/1963	978607321	Los Faiques140	A	2
	32437658	Chamberg	Celis	Lucila	08/05/1954	987654321	Bolognesi500	A	1
	67543987	Rojas	Torres	Liliana	04/09/1950	908965456	Balta120	A	4
	69463100	Smith	Bances	María	01/09/1951	910234654	Sarmiento110	A	4

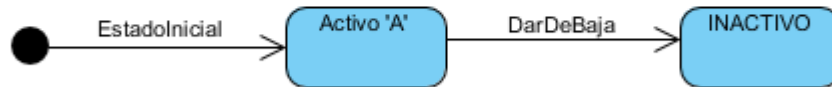
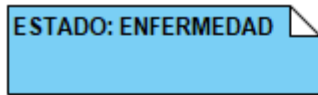
PACIENTE ASISTENTE -	DNI_ASISTENTE	DNI_PACIENTE	FECHA_INICIO	FECHA_FIN
PK/FK	FK	FK	-	-
NN/U	NN	NN	NN	-
TIPO DE DATO	smallint	char(8)	DATE	DATE
EJEMPLO	78576543	76857434	01/03/2008	12/10/2008
	34546577	76857434	20/10/2008	-
	32437658	68795847	05/10/2015	-
	67543987	68553209	20/12/2016	-
	69463100	90795743	16/07/2015	-

CITA	ID_CITA	ID_HOSPITAL	DNI_PACIENTE	FECHA_REGISTRO	HORA_REGISTRO	FECHA_CITA	HORA_CITA	ESTADO	TIPO_CITA
PK/FK	PK	FK1	FK2	-	-	-	-	-	-
NN/U	NN/U	NN	NN	NN	NN	NN	NN	NN	NN
TIPO DE DATO	smallint	char(8)	DATE	DATE	TIME	DATE	TIME	char(1)	char(1)
EJEMPLO	1	1	76857434	07/05/2015	19:33:41.848949-05	10/05/2015	10:00	R	T
	2	1	68795847	01/02/2016	19:50:53.614108-05	10/03/2016	10:00	R	P
	3	2	68553209	04/10/2017	19:33:41.848949-05	10/10/2017	16:00	R	P

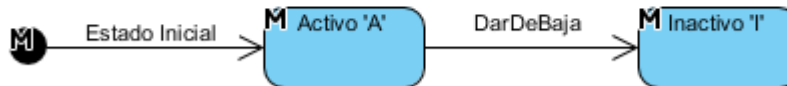
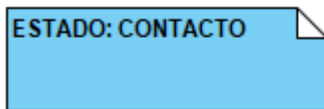
TRATAMIENTO	ID_TRATAMIENTO	DESCRIPCION	ID_CITA	DNI_ASISTENTE	DNI_DOCUMENTOR	ID_TIPO_TRATAMIENTO	FECHA_REGISTRO	FECHA_INICIO	FECHA_FIN_estimado
PK/FK	PK	-	FK	FK	FK	FK	-	-	-
NN/U	NN/U	NN	NN	NN	NN	NN	NN	NN	NN
TIPO DE DATO	smallint	varchar(100)	char(8)	DATE	DATE	smallint	TIME		
EJEMPLO	1	Tratamiento para HTA en segundo Nivel	1	34546577	1	1	10/05/2015	11/05/2015	falta
	2	Tratamiento para Diabetes Crónica	1	34546577	1	1	10/05/2015	11/05/2015	
	3	Tratamiento para la tuberculosis en fase 1 con restriccion de penicillina	2	32437658	2	1	10/03/2016	10/03/2016	
	4	Tratamiento para la Lumbalgia en la espalda baja	3	67543987	2	3	10/10/2017	11/10/2017	

## 5.6. DIAGRAMA DE ESTADOS

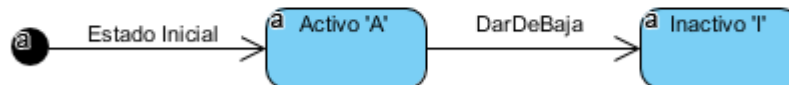
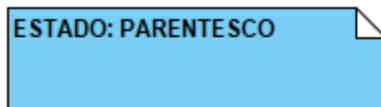
### 5.6.1. Estado: enfermedad:



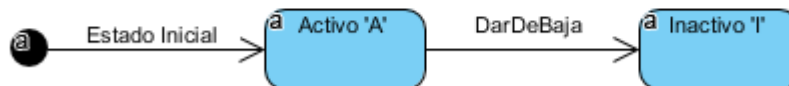
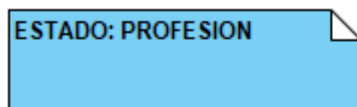
### 5.6.2. Estado: Contacto



### 5.6.3. Estado: Parentesco

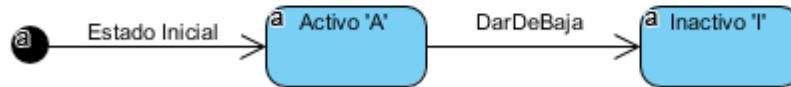
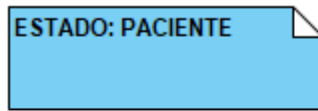


### 5.6.4. Estado: Profesión

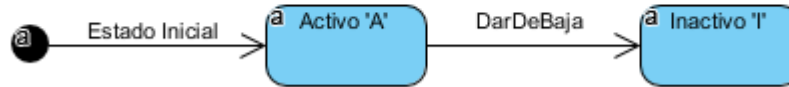
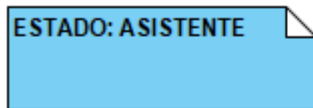




### 5.6.5. Estado: Paciente

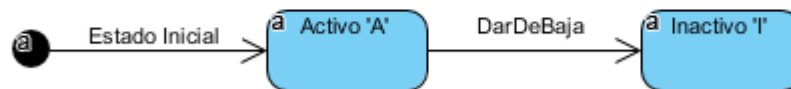
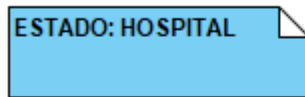


### 5.6.6. Estado: Asistente

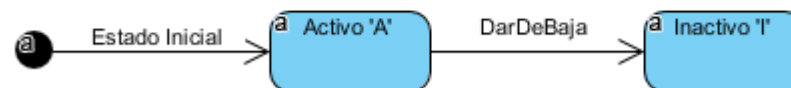
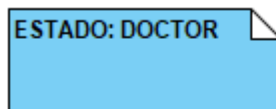


### 5.6.7. Estado: Hospital

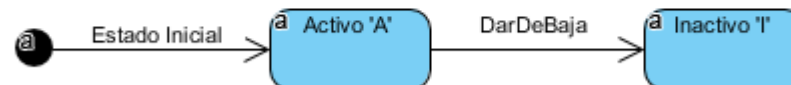
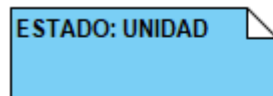
---



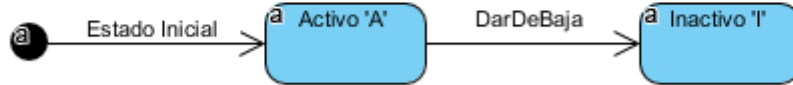
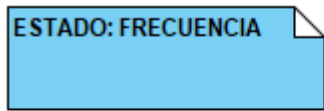
### 5.6.8. Estado: Doctor



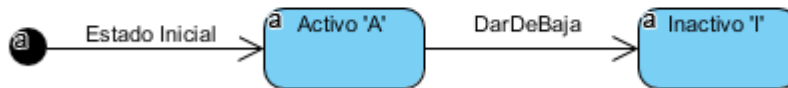
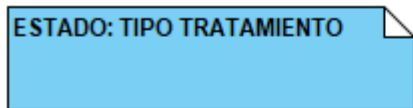
### 5.6.9. Estado: Unidad



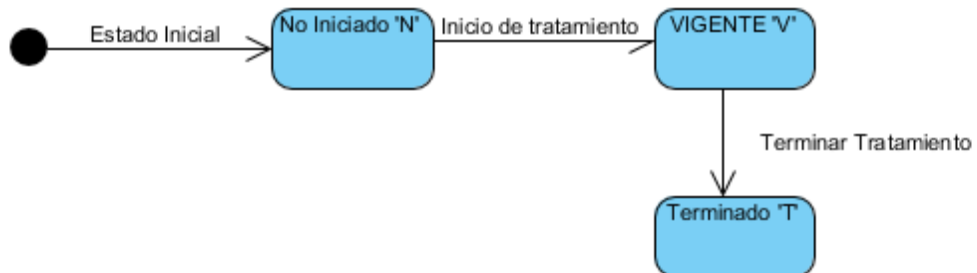
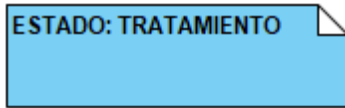
### 5.6.10. Estado: Frecuencia



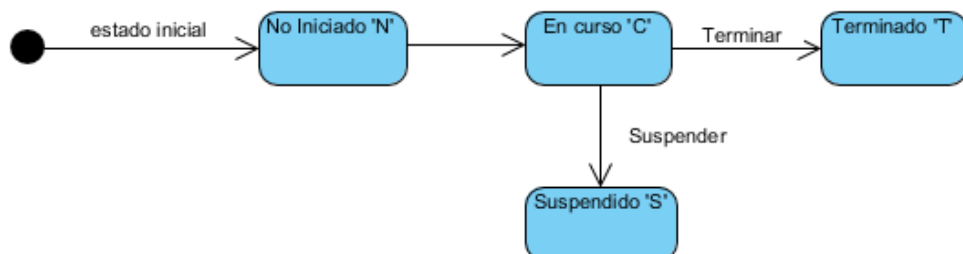
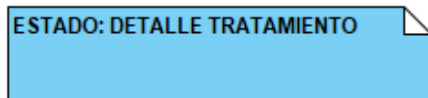
### 5.6.11. Estado: Tipo Tratamiento



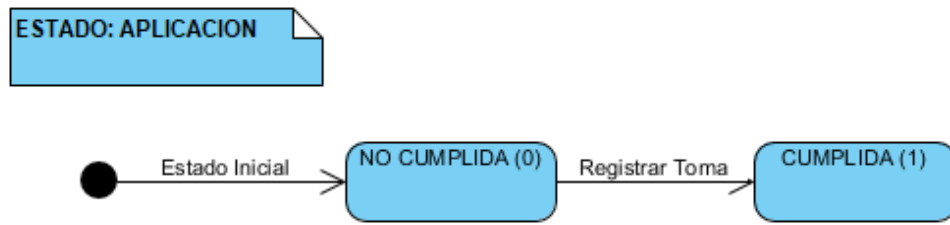
### 5.6.12. Estado: Tratamiento



### 5.6.13. Estado: Detalle tratamiento



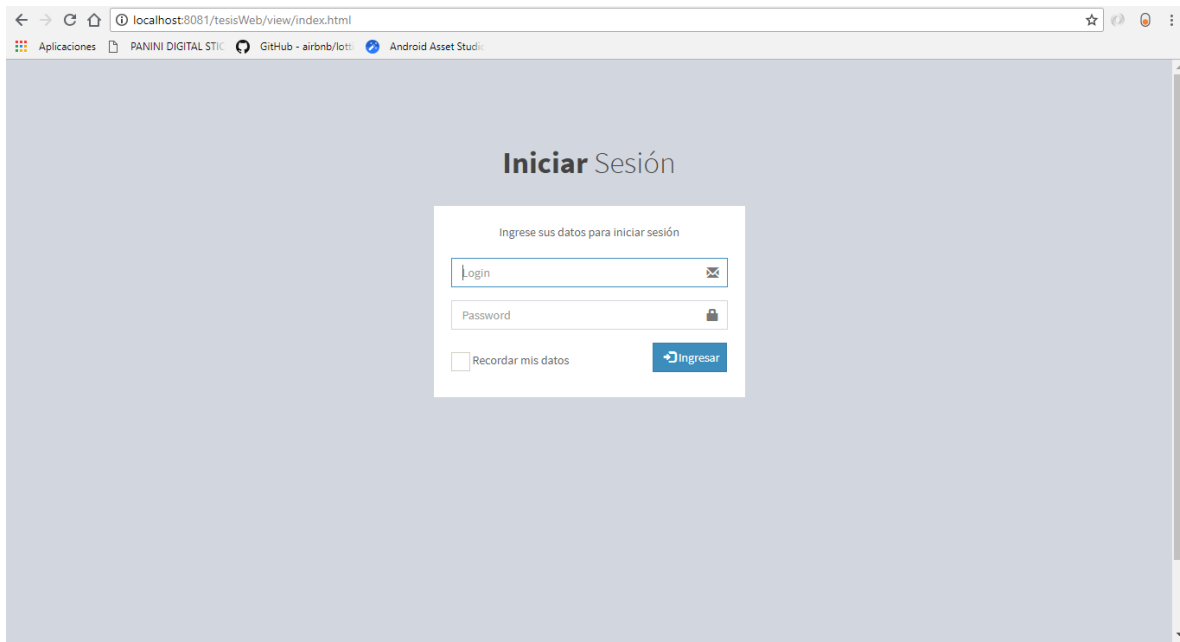
## 5.6.14. Estado Aplicación



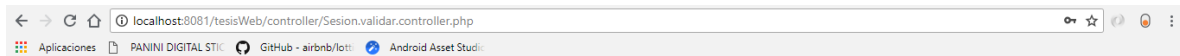
## 5.7. DISEÑO DE INTERFACES

### 5.7.1. WEB

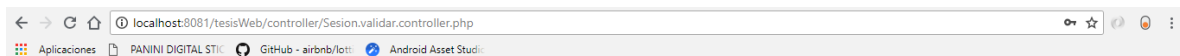
#### 5.7.1.1. PAGINA PRINCIPAL



#### MENSAJES DE ERROR AL INICIAR SESION

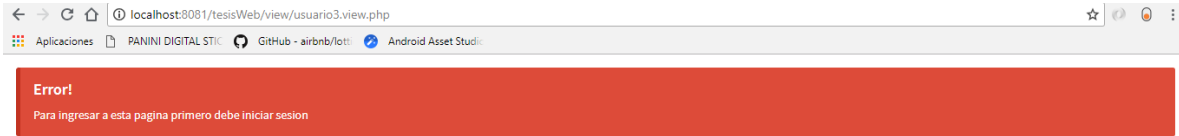


Entendido



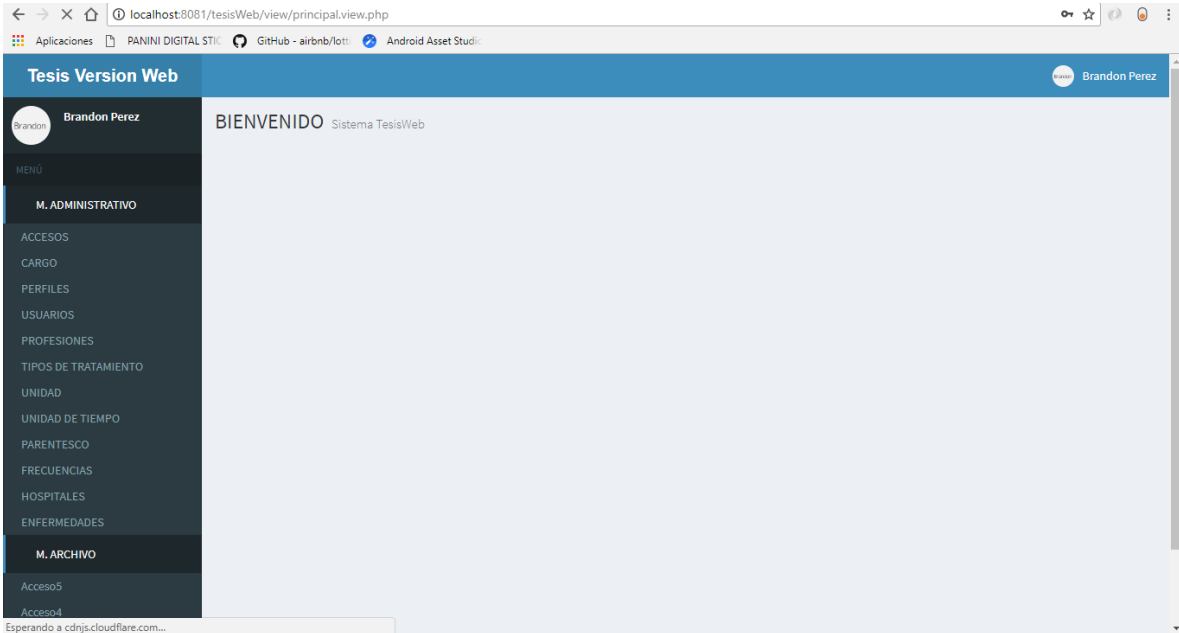
Entendido

# NEGACION DE ACCESO CUANDO SE QUIERE INGRESAR DIRECTAMENTE A UNA DIRECCION SI HABER INICIADO SESION

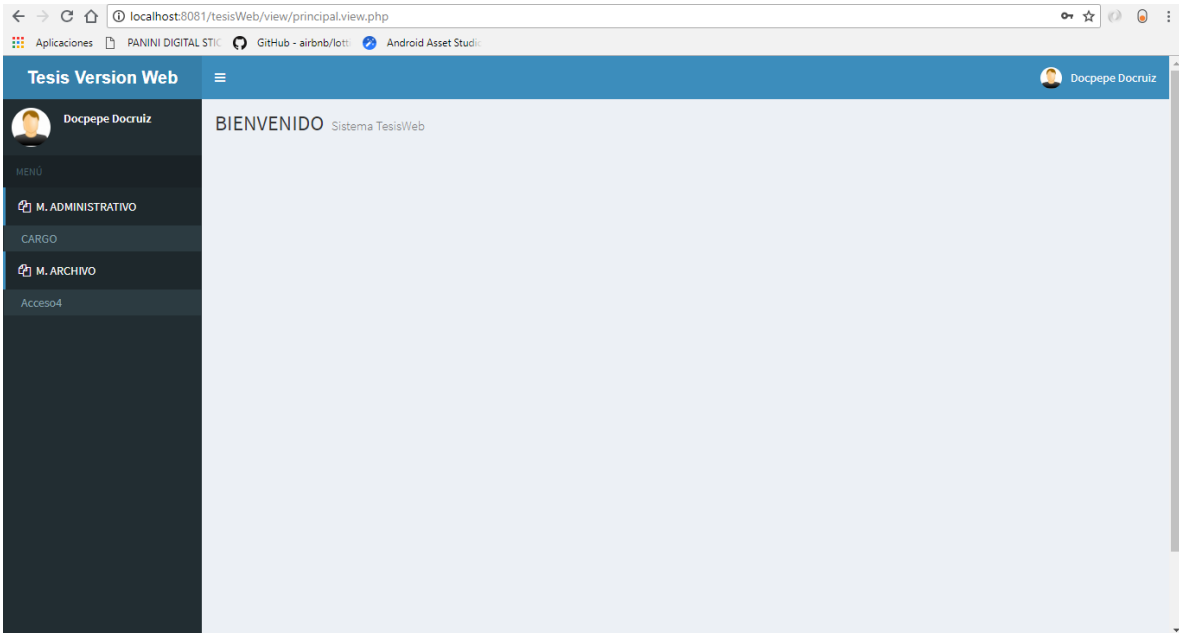


Entendido

## INICIO DE SESION COMO ADMINISTRADOR



## INICIO DE SESION COMO DOCTOR



## 5.7.1.2.ACCESOS

The screenshot displays the 'Tesis Version Web' application interface. The top navigation bar shows the user 'Brandon Perez'. The sidebar menu includes categories like 'M. ADMINISTRATIVO' and 'M. ARCHIVO'. The main content area is titled 'ACCESOS' and features a table with 16 entries. A modal window titled 'Agregar nuevo acceso' is open, allowing the user to add a new access record with fields for 'Código', 'Nombre', 'Link', and 'Icono'. The table below shows the following data:

CODIGO	NOMBRE DEL ACCESO	LINK	ICONO	ESTADO	OPCIONES
5	Acceso4	Acceso4.php	Acceso4EDITADO	A	[Edit] [Delete]
6	Acceso5	Acceso5.php	AccesoWeb	A	[Edit] [Delete]
1	ACCESOS	acceso.view.php	icono	A	[Edit] [Delete]
4	CARGO	cargoview.php	Acceso3	A	[Edit] [Delete]
17	ENFERMEDADES	enfermedad.view.php	icono	A	[Edit] [Delete]
15	FRECUENCIAS	frecuencia.view.php	icono	A	[Edit] [Delete]
16	HOSPITALES	hospital.view.php	icono	A	[Edit] [Delete]
7	M. ADMINISTRATIVO	null	I.ADMINISTRATIVO	A	[Edit] [Delete]
8	M. ARCHIVO	null	I.ARCHIVO	A	[Edit] [Delete]
14	PARENTESCO	parentesco.view.php	icono	A	[Edit] [Delete]

localhost:8081/tesisWeb/view/acceso.view.php

Aplicaciones PANINI DIGITAL STIC GitHub - airbnb/lot: Android Asset Studio

### Tesis Version Web

Brandon Perez

#### ACCESOS

Agregar nuevo acceso

Show 10 entries

CODIGO	NOMBRE	LINK	ICONO	ESTADO	OPCIONES
5	Acceso4	Acceso4.php	Acceso4EDITADO	A	[icon] [icon]
6				A	[icon] [icon]
1				A	[icon] [icon]
4				A	[icon] [icon]
17				A	[icon] [icon]
15	FRECUENCIAS	frecuencia.view.php	icono	A	[icon] [icon]
16	HOSPITALES	hospital.view.php	icono	A	[icon] [icon]
7	M.ADMINISTRATIVO	null	I.ADMINISTRATIVO	A	[icon] [icon]
8	M.ARCHIVO	null	I.ARCHIVO	A	[icon] [icon]
14	PARENTESCO	parentesco.view.php	icono	A	[icon] [icon]

Showing 1 to 10 of 16 entries

Previous 1 2 Next

#### Editar acceso

Código: 5

Nombre: Acceso4

Link: Acceso4.php

Icono: Acceso4EDITADO

Grabar Cerrar

localhost:8081/tesisWeb/view/acceso.view.php

Aplicaciones PANINI DIGITAL STIC GitHub - airbnb/lot: Android Asset Studio

### Tesis Version Web

Brandon Perez

#### ACCESOS

Agregar nuevo acceso

Show 10 entries

CODIGO	NOMBRE	LINK	ICONO	ESTADO	OPCIONES
5	Acceso4	Acceso4.php	Acceso4EDITADO	A	[icon] [icon]
6				A	[icon] [icon]
1				A	[icon] [icon]
4				A	[icon] [icon]
17				A	[icon] [icon]
15	FRECUENCIAS	frecuencia.view.php	icono	A	[icon] [icon]
16	HOSPITALES	hospital.view.php	icono	A	[icon] [icon]
7	M.ADMINISTRATIVO	null	I.ADMINISTRATIVO	A	[icon] [icon]
8	M.ARCHIVO	null	I.ARCHIVO	A	[icon] [icon]
14	PARENTESCO	parentesco.view.php	icono	A	[icon] [icon]

Showing 1 to 10 of 16 entries

Previous 1 2 Next

#### Confirme

¿Esta seguro de eliminar el registro seleccionado?

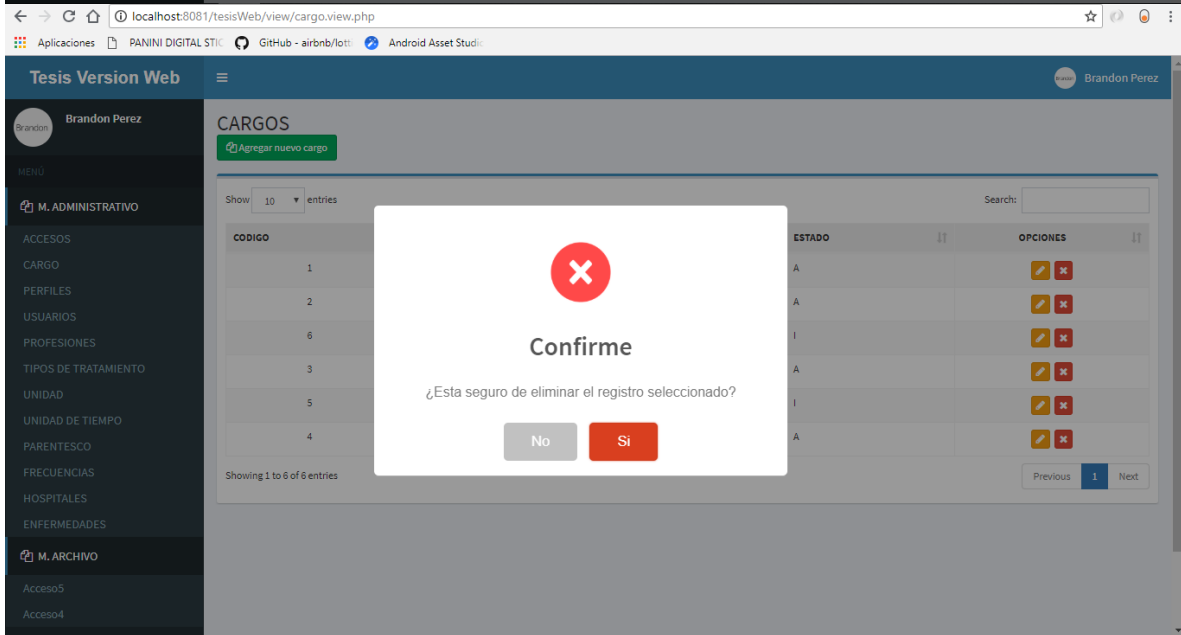
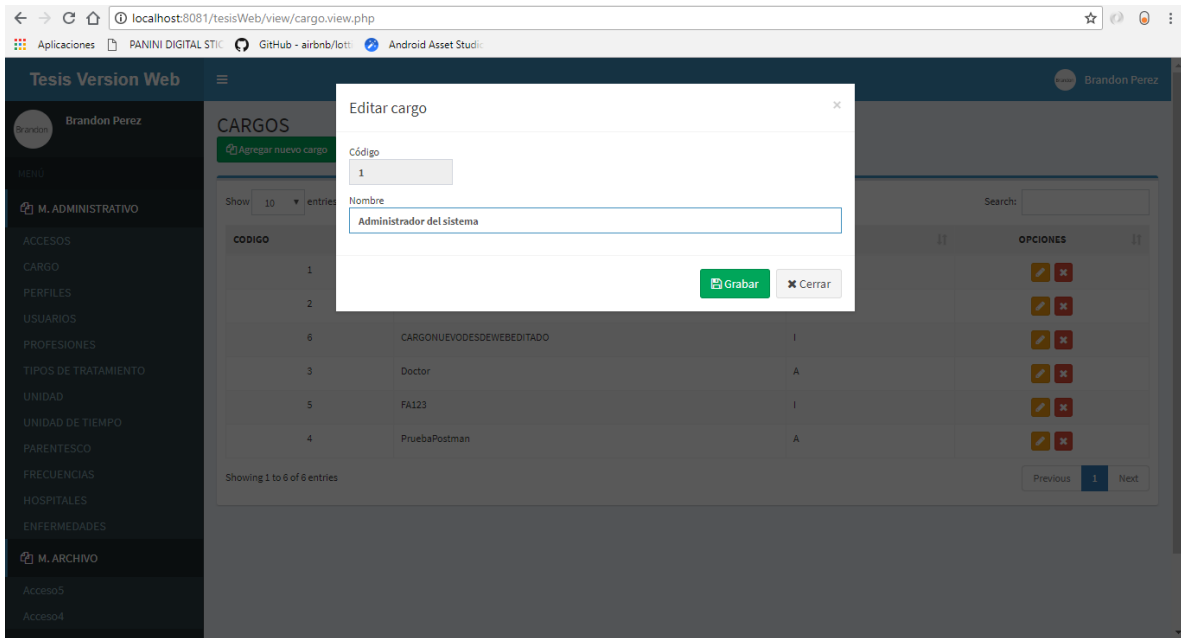
No Si

### 5.7.1.3.CARGO

The image shows two screenshots of a web application interface. The top screenshot displays the 'CARGOS' (Jobs) management page. The page has a dark sidebar with a menu and a main content area. The 'CARGOS' section includes a table with columns for 'CODIGO', 'NOMBRE DEL CARGO', 'ESTADO', and 'OPCIONES'. The table contains six entries. Below the table, there is a pagination control showing 'Showing 1 to 6 of 6 entries' and 'Previous 1 Next' buttons. A green button labeled 'Agregar nuevo cargo' is visible at the top left of the table area.

CODIGO	NOMBRE DEL CARGO	ESTADO	OPCIONES
1	Administrador del sistema	A	[Edit] [Delete]
2	Asistente	A	[Edit] [Delete]
6	CARGONUEVOSEDEWEBEDITADO	I	[Edit] [Delete]
3	Doctor	A	[Edit] [Delete]
5	FA123	I	[Edit] [Delete]
4	PruebaPostman	A	[Edit] [Delete]

The bottom screenshot shows the same 'CARGOS' page but with a modal window open for adding a new job. The modal is titled 'Agregar nuevo cargo' and contains two input fields: 'Código' and 'Nombre'. At the bottom of the modal, there are two buttons: 'Grabar' (Save) and 'Cerrar' (Close).





## 5.7.1.4.PERFILES

The image shows two screenshots of a web application interface. The top screenshot displays the 'PERFILES' section with a table of existing profiles. The bottom screenshot shows a modal window for adding a new profile.

**PERFILES**

Agregar nuevo perfil

Show 10 entries

CODIGO	NOMBRE DEL PERFIL	ESTADO	OPCIONES
1	Admin	A	
2	Asistente	A	
4	Conserje	I	
3	Doctor	A	
5	PRUEBAPOSTAAA	I	
6	pruebawebEDIT	I	
7	SALOME	A	

Showing 1 to 7 of 7 entries

Previous 1 Next

**Agregar nuevo perfil**

Código

Nombre

localhost:8081/tesisWeb/view/perfil.view.php

Tesis Version Web

Brandon Perez

PERFILES

Agregar nuevo perfil

Show 10 entries

CODIGO

CODIGO	Nombre		
1			
2			
4	Conserje	I	
3	Doctor	A	
5	PRUEBAPOSTAAA	I	
6	pruebaweBEDIT	I	
7	SALOME	A	

Showing 1 to 7 of 7 entries

OPCIONES

Grabar Cerrar

Editar perfil

Código: 1

Nombre: Admin

localhost:8081/tesisWeb/view/perfil.view.php

Tesis Version Web

Brandon Perez

PERFILES

Agregar nuevo perfil

Show 10 entries

CODIGO

CODIGO	Nombre		
1			
2			
4			
3			
5			
6			
7			

Showing 1 to 7 of 7 entries

OPCIONES

Confirme

¿Esta seguro de eliminar el registro seleccionado?

No Si

## 5.7.1.5.USUARIOS

USUARIOS

➕ Agregar nuevo usuario

Show: 10 entries

CODIGO	NOMBRE DE PERSONAL	LOGIN	CLAVE	PERFIL	ACCESO	OPCIONES
3	AsistPerez AsistGuevara AsistJesus	12346789	900150983cd24fb0d6963f7d28e17f72	Asistente	A	
2	DocRuiz DocLeiva DocPepe	87654321	250cf8b51c773f3f8dc8b4be867a9a02	Doctor	A	
1	Perez Guevara Brandon	12345678	202cb962ac59075b964b07152d234b70	Admin	A	

Showing 1 to 3 of 3 entries

Previous 1 Next

Título de la ventana

Código

Nombre

Grabar Cerrar

## 5.7.1.6.PROFESIONES

The image shows two screenshots of a web application interface. The top screenshot displays a list of professions, and the bottom screenshot shows a modal for adding a new profession.

**Top Screenshot: List of Professions**

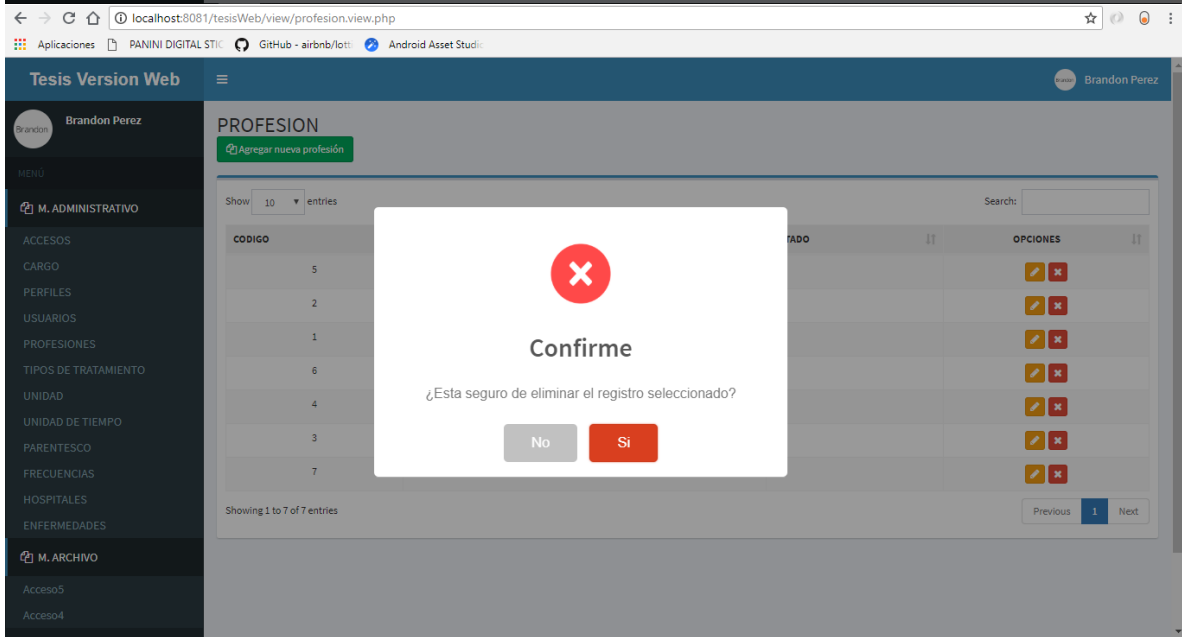
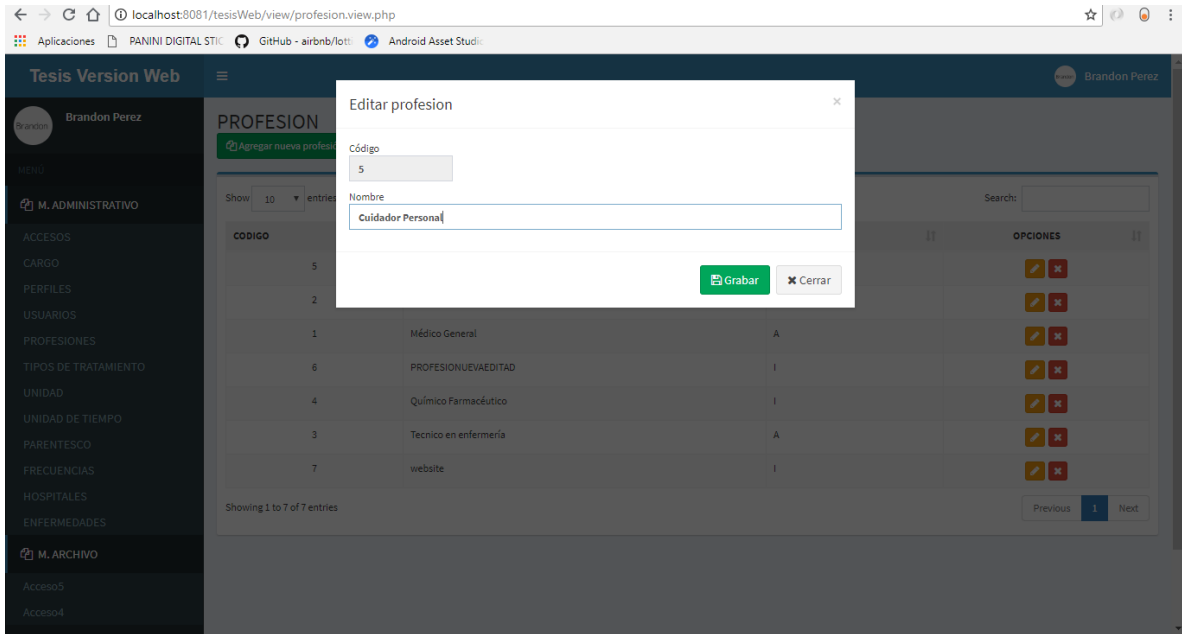
The interface shows a sidebar with navigation options: M. ADMINISTRATIVO, ACCESOS, CARGO, PERFILES, USUARIOS, PROFESIONES, TIPOS DE TRATAMIENTO, UNIDAD, UNIDAD DE TIEMPO, PARENTESCO, FRECUENCIAS, HOSPITALES, ENFERMEDADES, and M. ARCHIVO. The main content area is titled "PROFESION" and includes a button "Agregar nueva profesión". Below this is a table with the following data:

CODIGO	NOMBRE DE PROFESION	ESTADO	OPCIONES
5	Cuidador Personal	I	[Edit] [Delete]
2	Enfermero	A	[Edit] [Delete]
1	Médico General	A	[Edit] [Delete]
6	PROFESIONUEVAEDITAD	I	[Edit] [Delete]
4	Químico Farmacéutico	I	[Edit] [Delete]
3	Tecnico en enfermería	A	[Edit] [Delete]
7	website	I	[Edit] [Delete]

Below the table, it says "Showing 1 to 7 of 7 entries" and includes "Previous" and "Next" navigation buttons.

**Bottom Screenshot: Add New Profession Modal**

The modal is titled "Agregar nueva profesión" and contains two input fields: "Código" and "Nombre". At the bottom of the modal are two buttons: "Grabar" (Save) and "Cerrar" (Close).













## 5.7.1.7. TIPOS DE TRATAMIENTO

The image shows two screenshots of a web application interface. The top screenshot displays a list of treatment types, and the bottom screenshot shows a modal form for adding a new treatment type.

**TIPOS DE TRATAMIENTO**

Agregar nuevo tipo de tratamiento

Show: 10 entries

CODIGO	NOMBRE DEL TIPO DE TRATAMIENTO	ESTADO	OPCIONES
2	Inyecciones	A	 
1	Masajes	A	 
3	PRUEBA EDITADA	I	 
4	TIPONUEVOEDITADO	I	 
5	website	I	 

Showing 1 to 5 of 5 entries

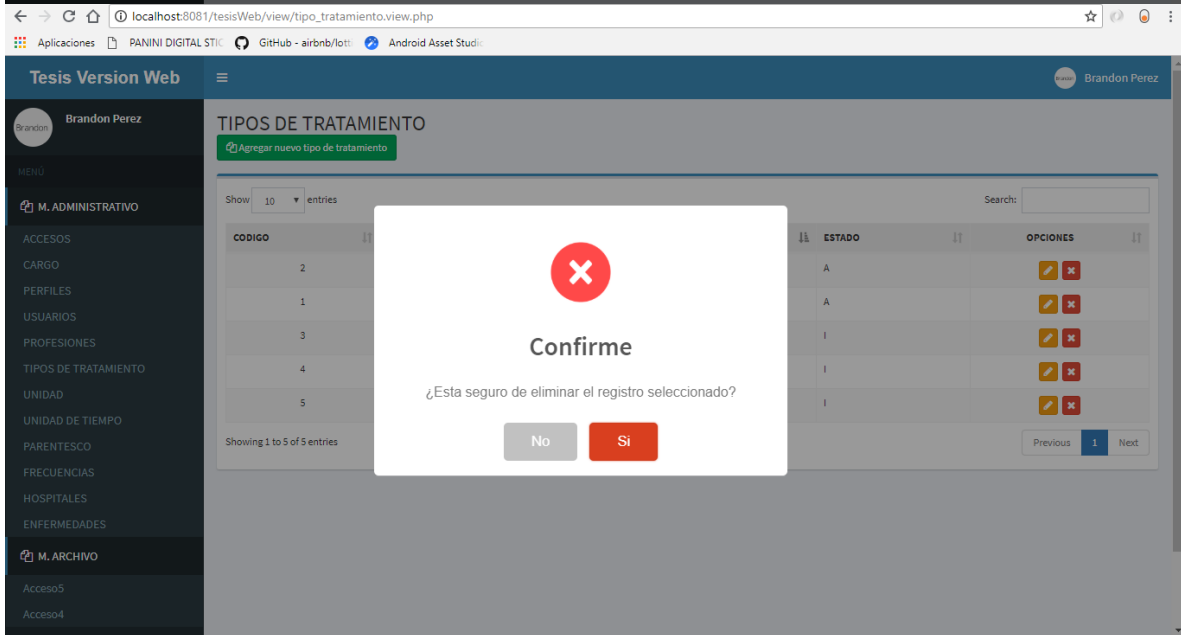
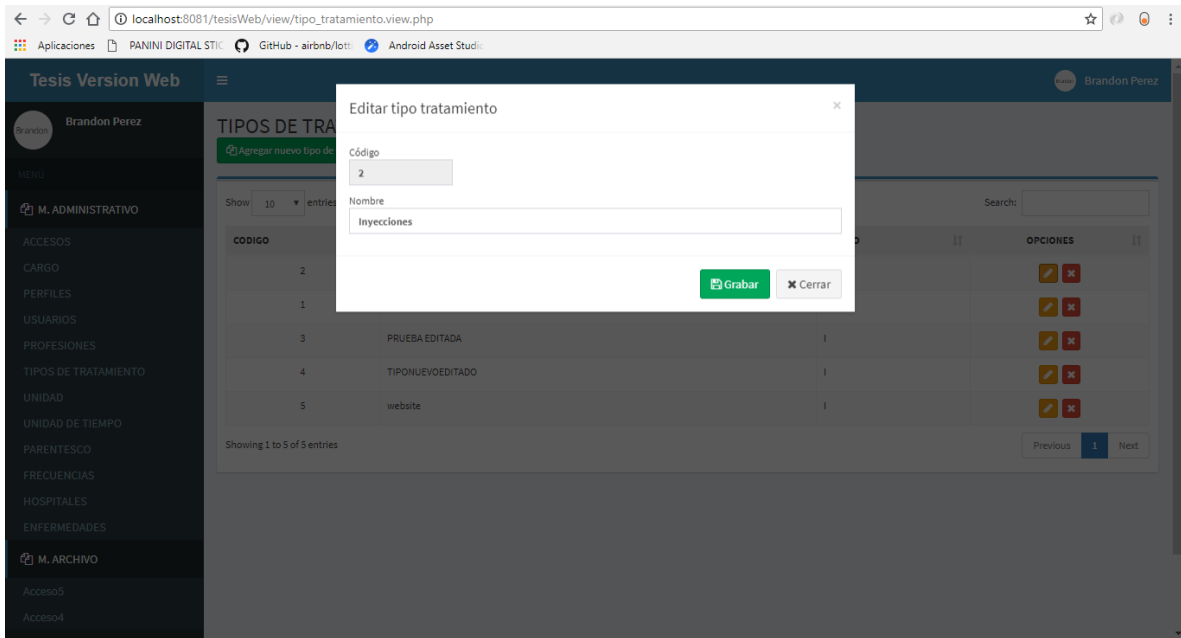
Previous 1 Next

**Agregar nuevo tipo de tratamiento**

Código

Nombre

Grabar Cerrar



## 5.7.1.8.UNIDAD

The image shows two screenshots of a web application interface. The top screenshot displays a list of units, and the bottom screenshot shows a modal for adding a new unit.

**Top Screenshot: UNIDAD List**

The interface shows a sidebar menu with categories like "M. ADMINISTRATIVO" and "M. ARCHIVO". The main content area is titled "UNIDAD" and includes a search bar and a table of units.

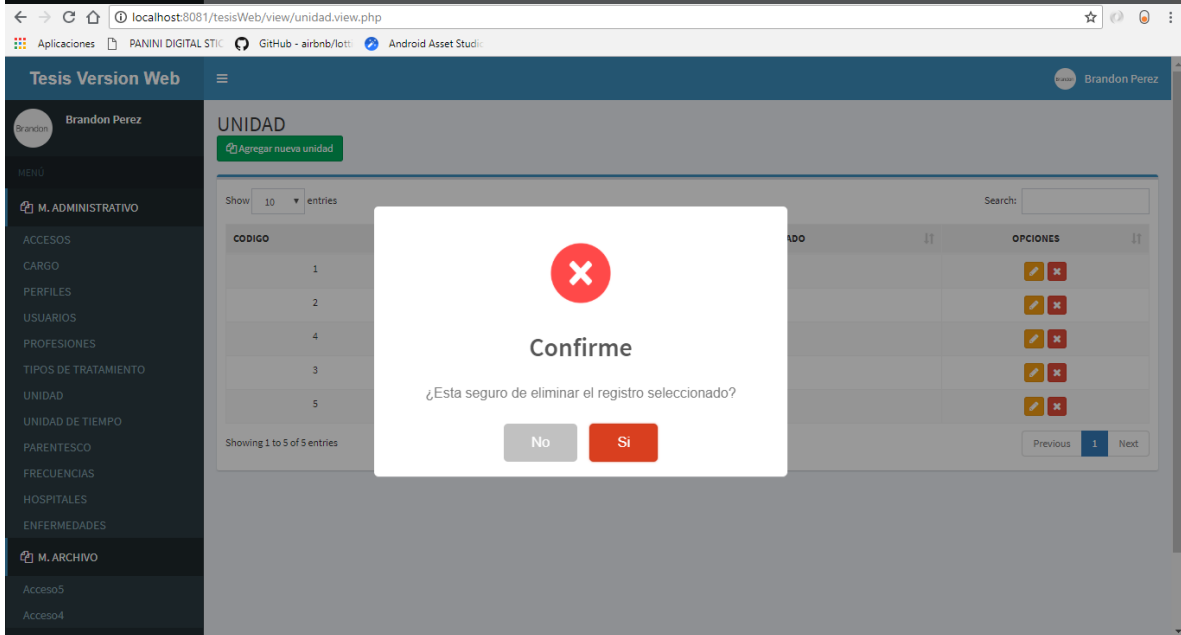
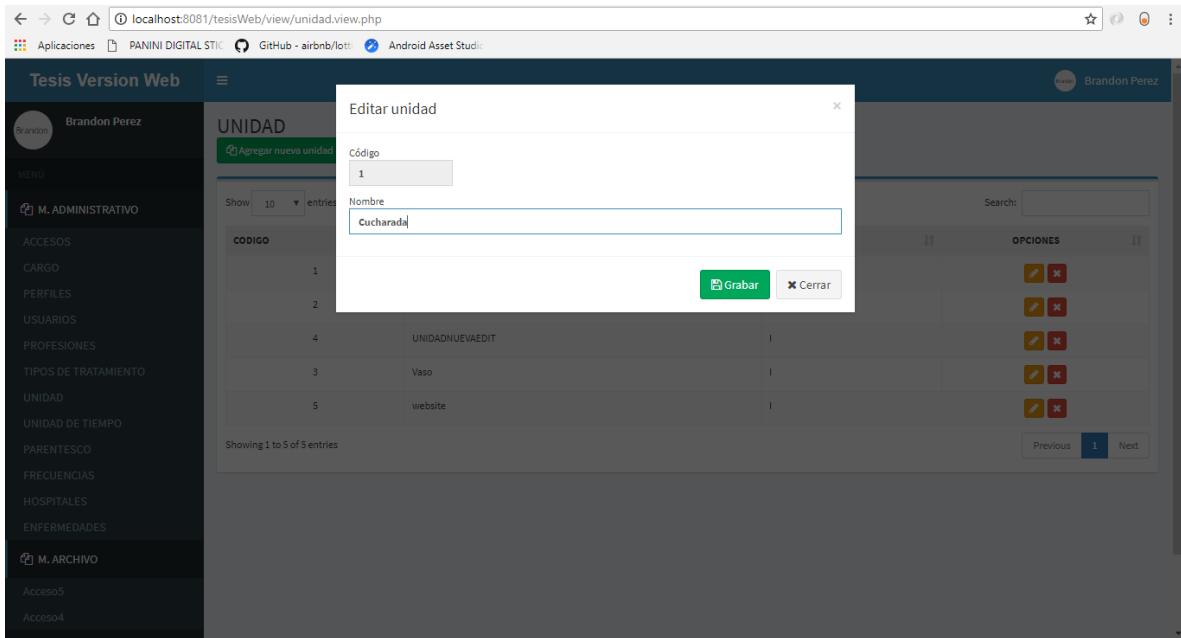
CODIGO	NOMBRE DE LA UNIDAD	ESTADO	OPCIONES
1	Cucharada	A	[Edit] [Delete]
2	Gota	A	[Edit] [Delete]
4	UNIDADNUEVAEDIT	I	[Edit] [Delete]
3	Vaso	I	[Edit] [Delete]
5	website	I	[Edit] [Delete]

Showing 1 to 5 of 5 entries. Navigation: Previous 1 Next.

**Bottom Screenshot: Agregar nueva unidad Modal**

The modal is titled "Agregar nueva unidad" and contains two input fields: "Código" and "Nombre". It has "Grabar" and "Cerrar" buttons.





## 5.7.1.9. UNIDAD DE TIEMPO

The image shows two screenshots of a web application interface. The top screenshot displays a table of time units, and the bottom screenshot shows a modal form for adding a new unit.

**Top Screenshot: UNIDAD DE TIEMPO**

The main content area shows a table with the following data:

CODIGO	NOMBRE DE LA UNIDAD DE TIEMPO	ESTADO	OPCIONES
5	año	I	[Edit] [Delete]
2	día	A	[Edit] [Delete]
1	hora	A	[Edit] [Delete]
4	mes	A	[Edit] [Delete]
6	pruebaveBEDITADA	I	[Edit] [Delete]
3	semana	A	[Edit] [Delete]

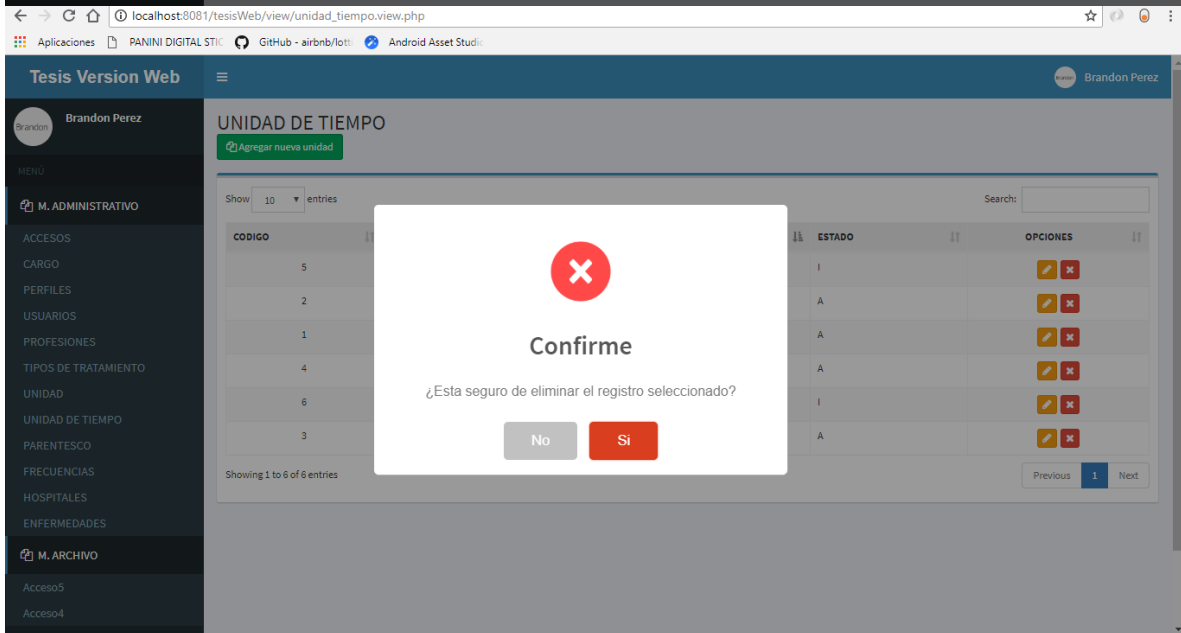
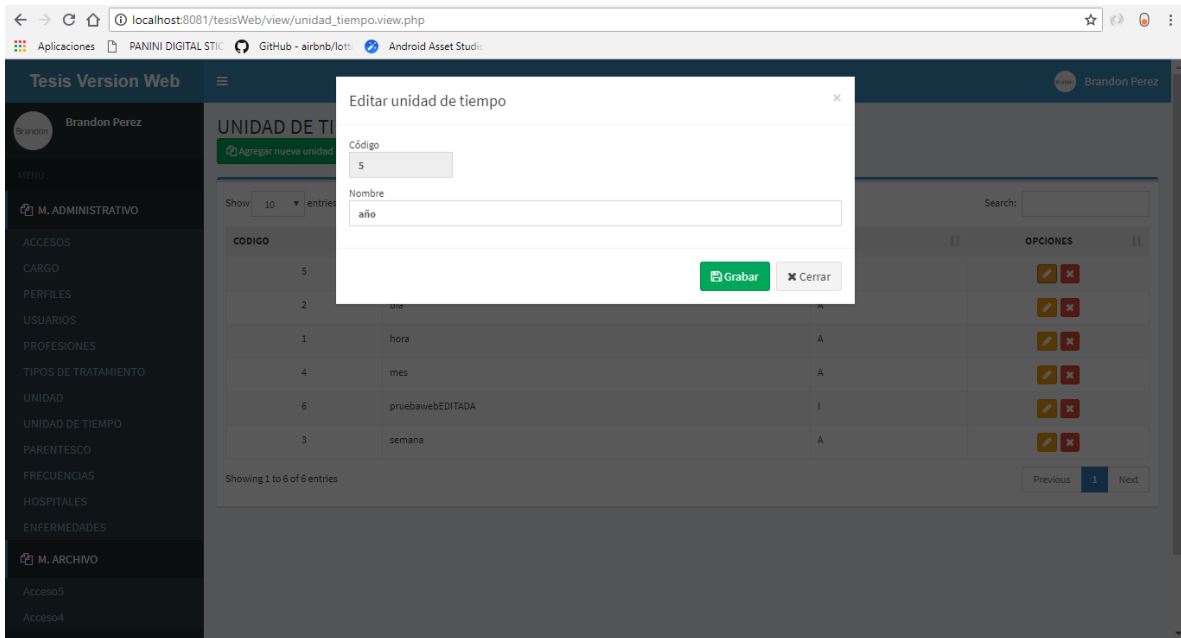
Below the table, it says "Showing 1 to 6 of 6 entries". There are "Previous" and "Next" navigation buttons.

**Bottom Screenshot: Agregar nueva unidad de tiempo**

A modal form is open with the following fields:

- Código:
- Nombre:

At the bottom of the modal, there are two buttons: "Grabar" (Save) and "Cerrar" (Close).



## 5.7.1.10. PARENTESCO

The image shows two screenshots of a web application interface. The top screenshot displays the 'PARENTESCO' section with a table of entries. The bottom screenshot shows a modal form for adding a new entry.

**Top Screenshot: PARENTESCO Table**

URL: localhost:8081/tesisWeb/view/parentesco.view.php

Header: Tesis Version Web | Brandon Perez

Menu: M. ADMINISTRATIVO, M. ARCHIVO

Buttons: Agregar nueva parentesco

Table:

CODIGO	NOMBRE DE PARENTESCO	ESTADO	OPCIONES
2	Mama	A	[Edit] [Delete]
1	Papa	A	[Edit] [Delete]
3	Sobrino	A	[Edit] [Delete]
8	webedit	I	[Edit] [Delete]

Showing 1 to 4 of 4 entries

Navigation: Previous | 1 | Next

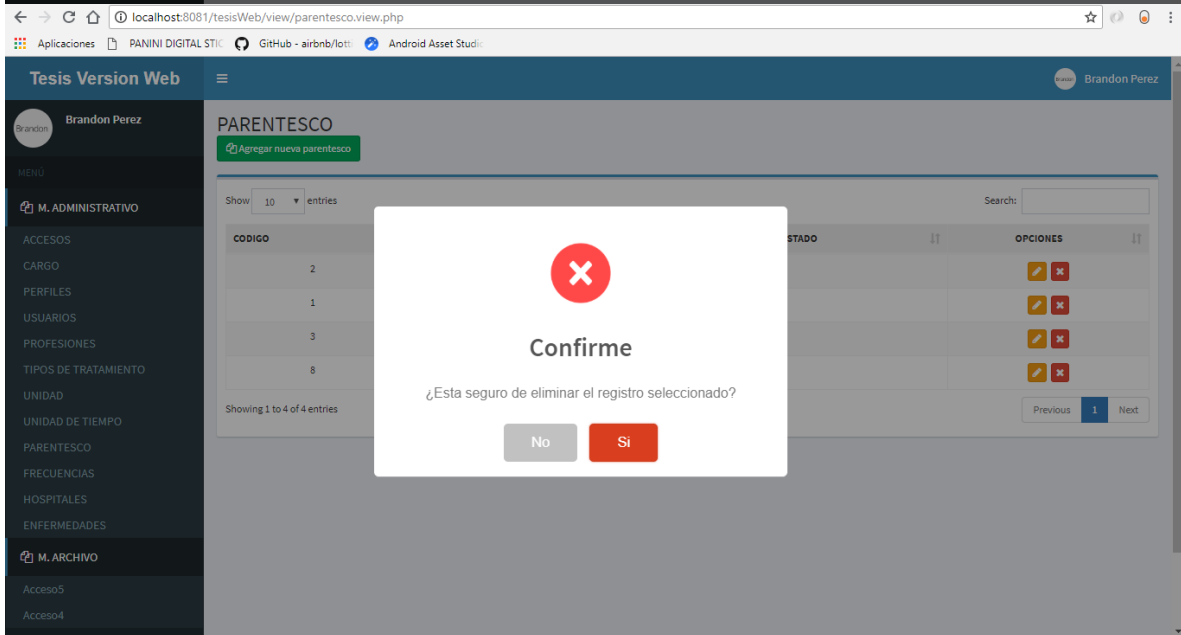
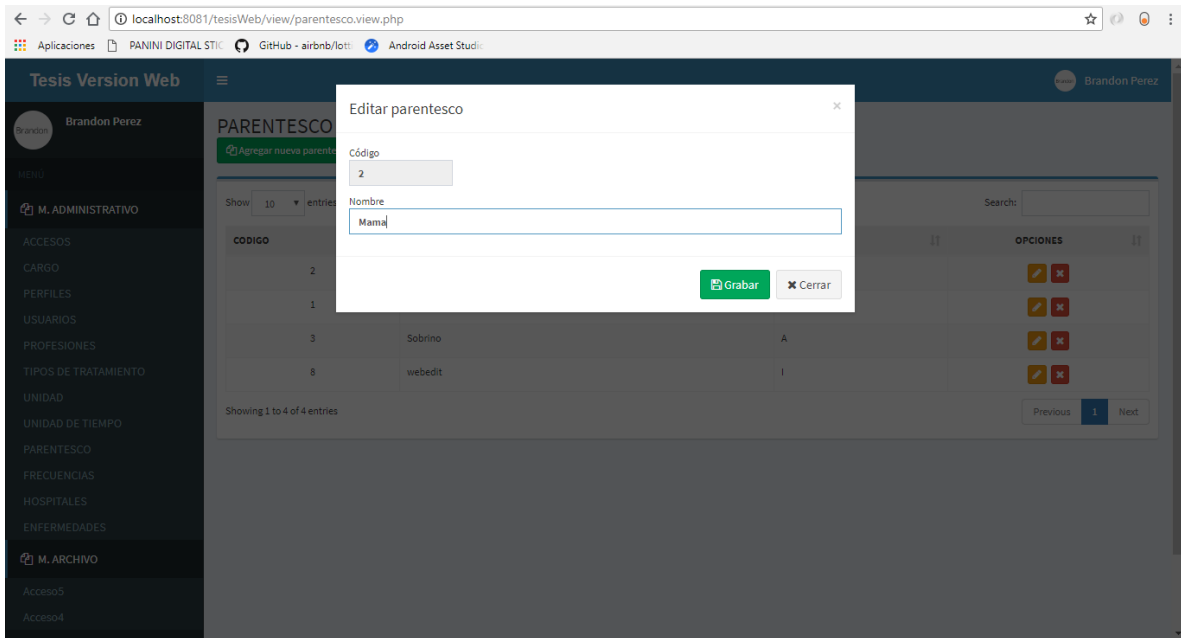
**Bottom Screenshot: Agregar nuevo parentesco Modal**

Modal Title: Agregar nuevo parentesco

Fields:

- Código:
- Nombre:

Buttons: Grabar, Cerrar



## 5.7.1.11. FRECUENCIAS

The image shows two screenshots of a web application interface. The top screenshot displays a table of frequencies, and the bottom screenshot shows a modal form for adding a new frequency.

**Table Data (Top Screenshot):**

CODIGO	NOMBRE DE FRECUENCIA	VALOR NUMÉRICO	ESTADO	OPCIONES
4	Antes de dormir	24	I	[Edit] [Delete]
3	Cada 12	12	A	[Edit] [Delete]
1	Cada 4	4	A	[Edit] [Delete]
2	Cada 8	8	A	[Edit] [Delete]
5	Prueba editada	10	I	[Edit] [Delete]
6	webedit	10	I	[Edit] [Delete]

**Modal Form Data (Bottom Screenshot):**

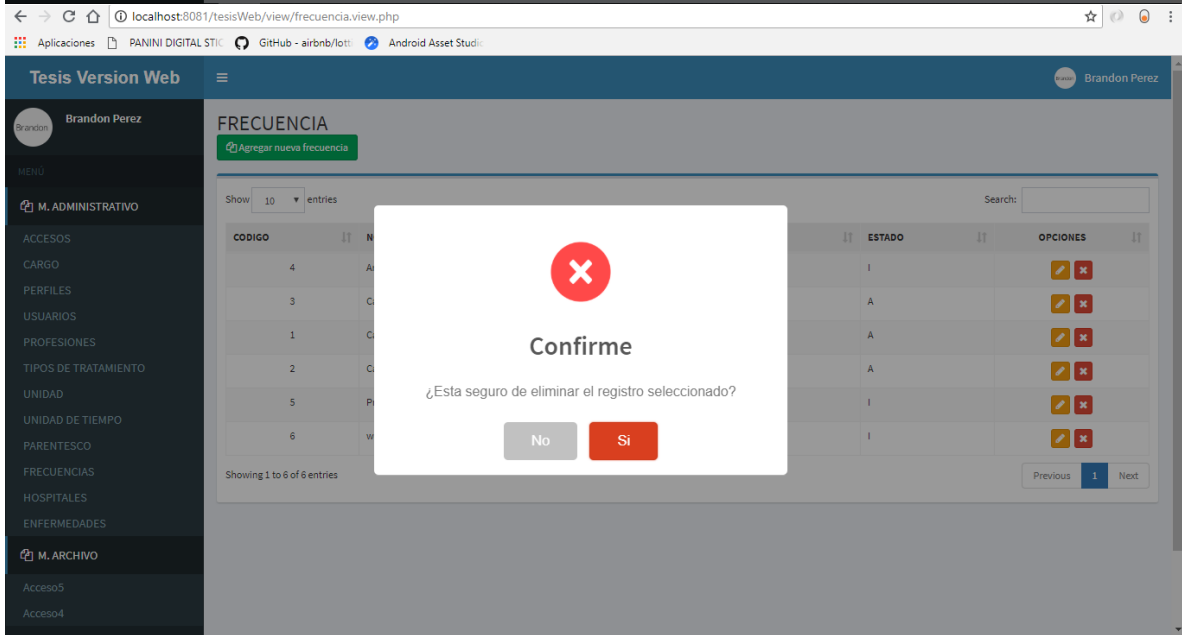
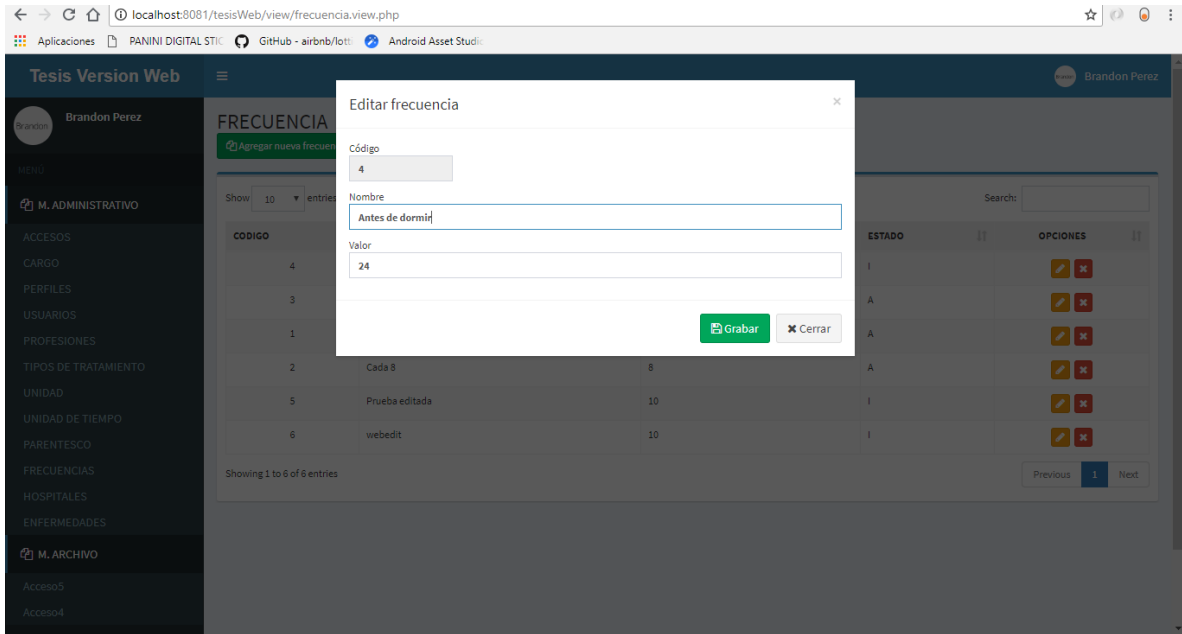
**Agregar nueva frecuencia**

Código:

Nombre:

Valor:

[Grabar] [Cerrar]



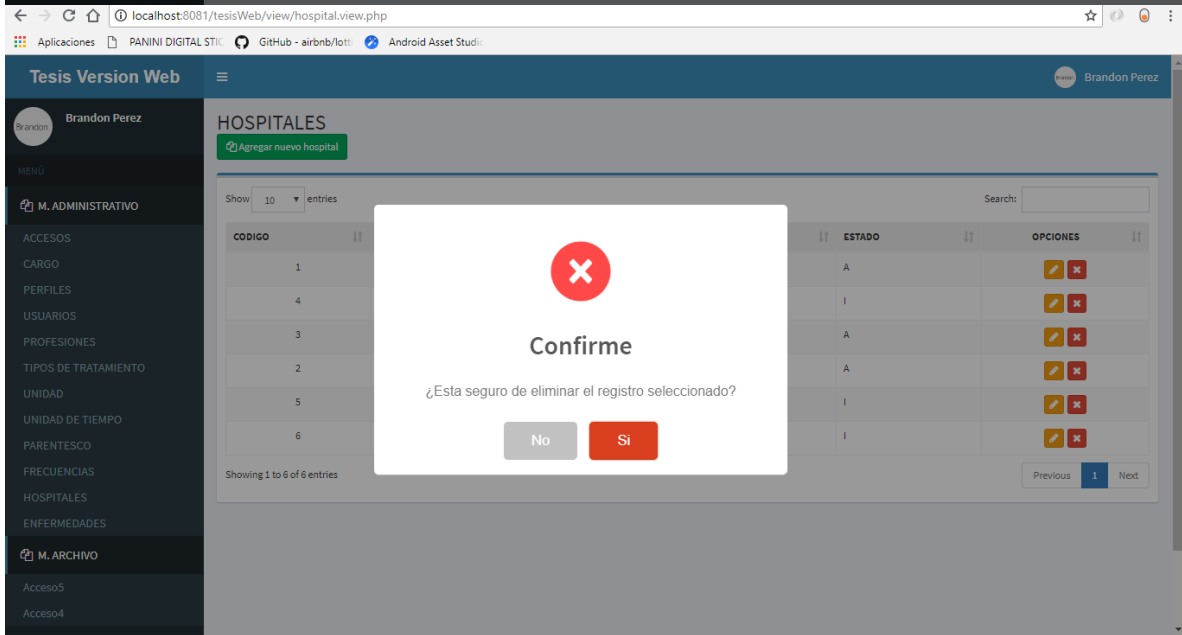
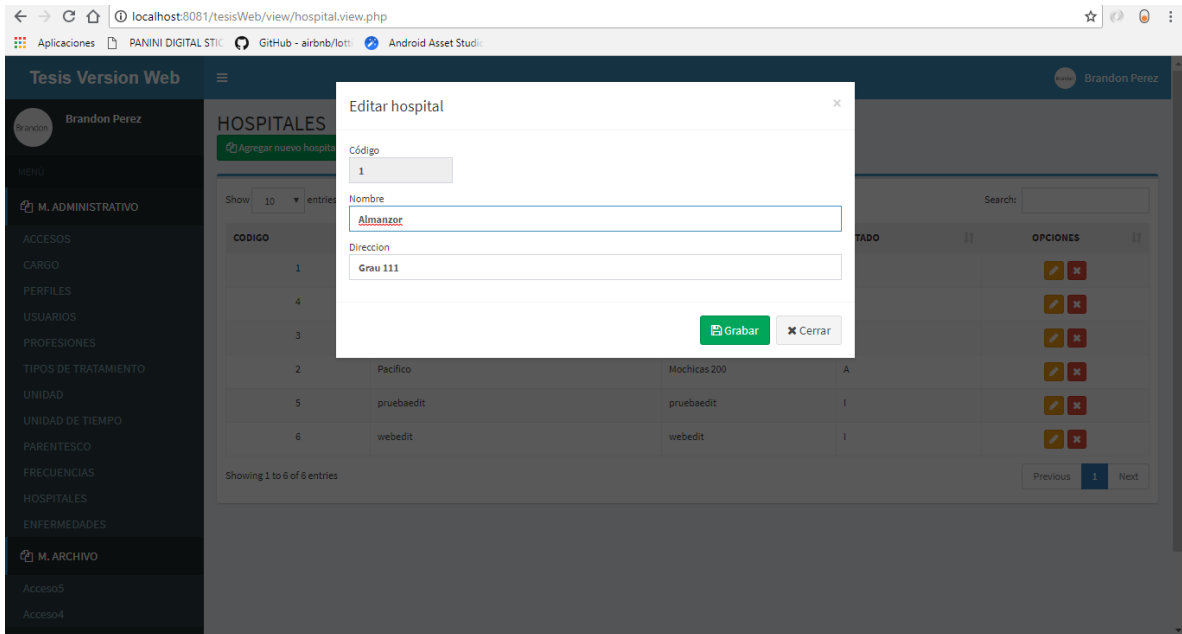
## 5.7.1.12. HOSPITALES

The image shows two screenshots of a web application interface. The top screenshot displays a list of hospitals under the heading "HOSPITALES". The interface includes a sidebar menu with options like "M. ADMINISTRATIVO" and "M. ARCHIVO", and a main content area with a table of hospital records. A green button "Agregar nuevo hospital" is visible at the top of the table.

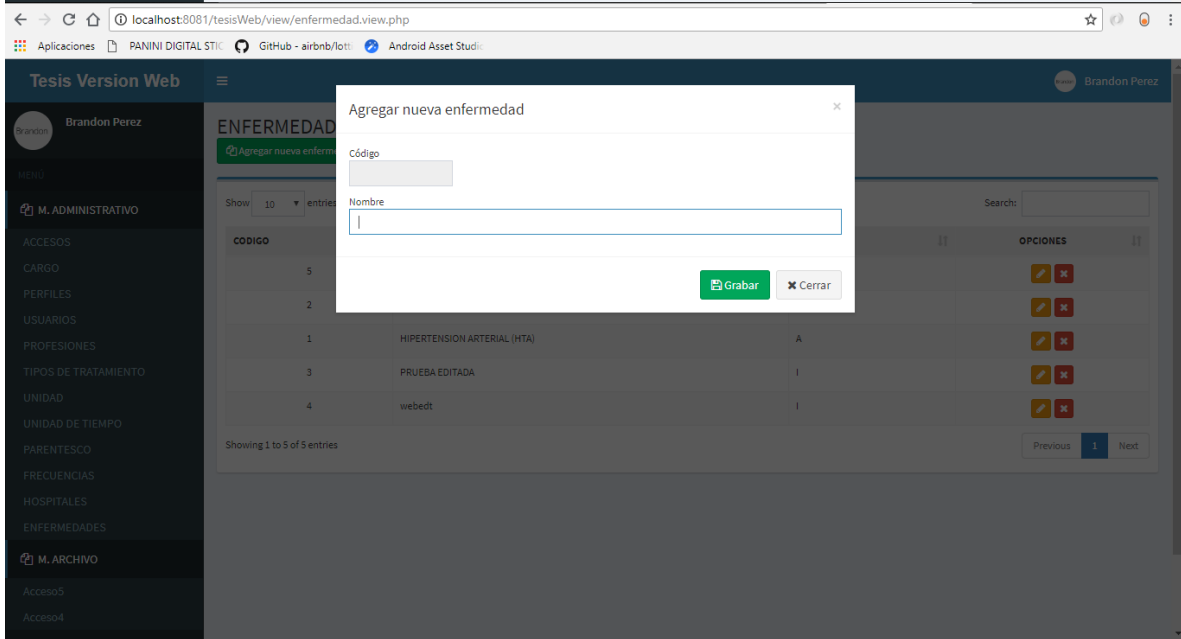
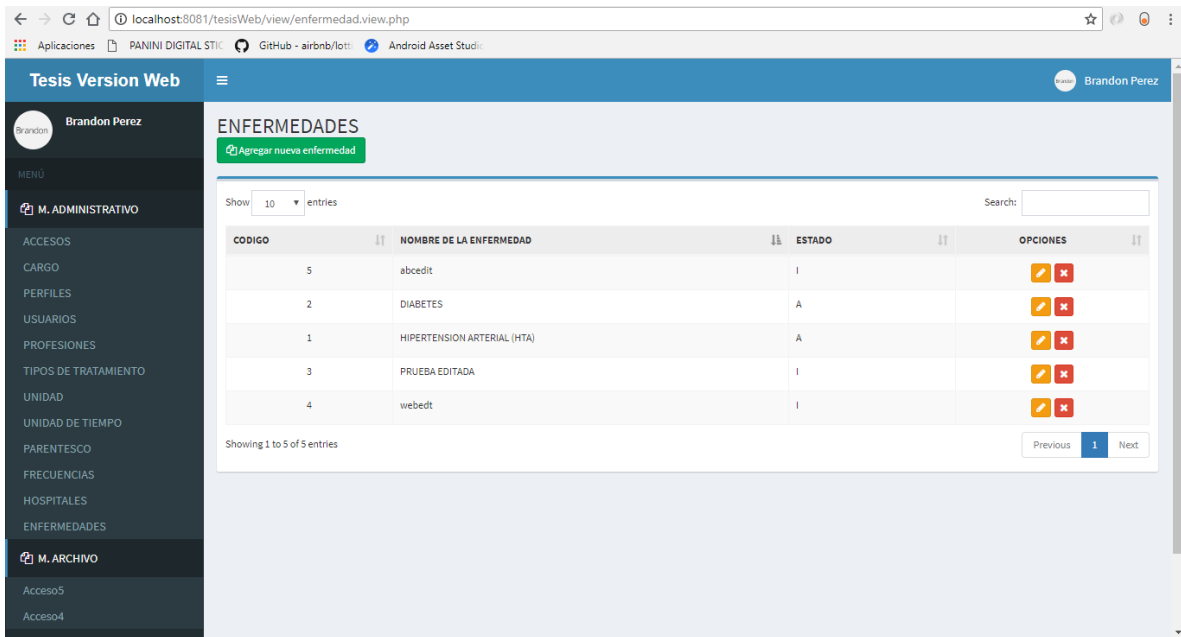
CODIGO	NOMBRE DEL HOSPITAL	DIRECCION	ESTADO	OPCIONES
1	Almanzor	Grau 111	A	[Edit] [Delete]
4	Editado	Editado	I	[Edit] [Delete]
3	Metropolitano	JLS 100	A	[Edit] [Delete]
2	Pacifico	Mochicas 200	A	[Edit] [Delete]
5	pruebaedit	pruebaedit	I	[Edit] [Delete]
6	webedit	webedit	I	[Edit] [Delete]

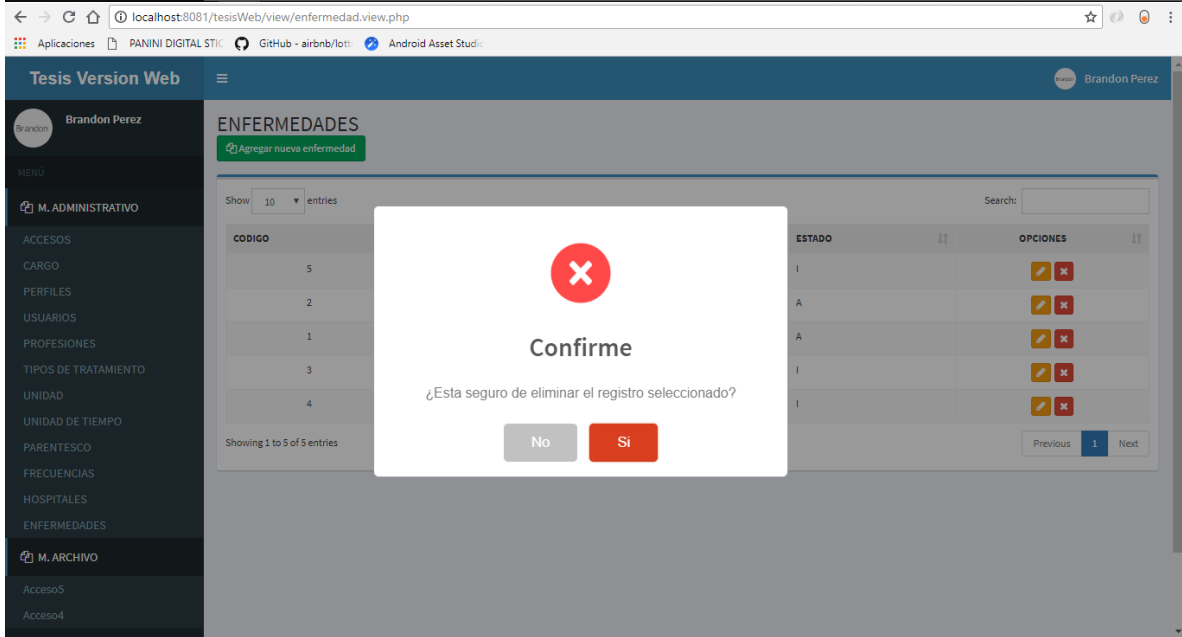
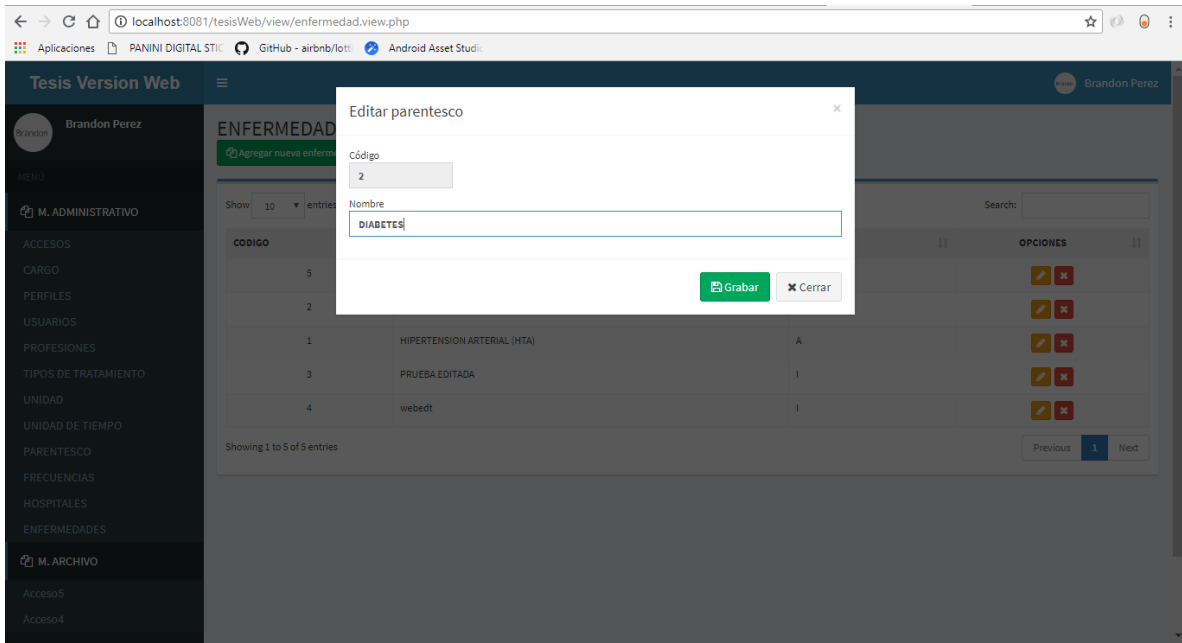
The bottom screenshot shows the same interface but with a modal form titled "Agregar nuevo hospital" open. The form contains three input fields: "Código", "Nombre", and "Direccion". At the bottom of the modal are two buttons: "Grabar" (Save) and "Cerrar" (Close).





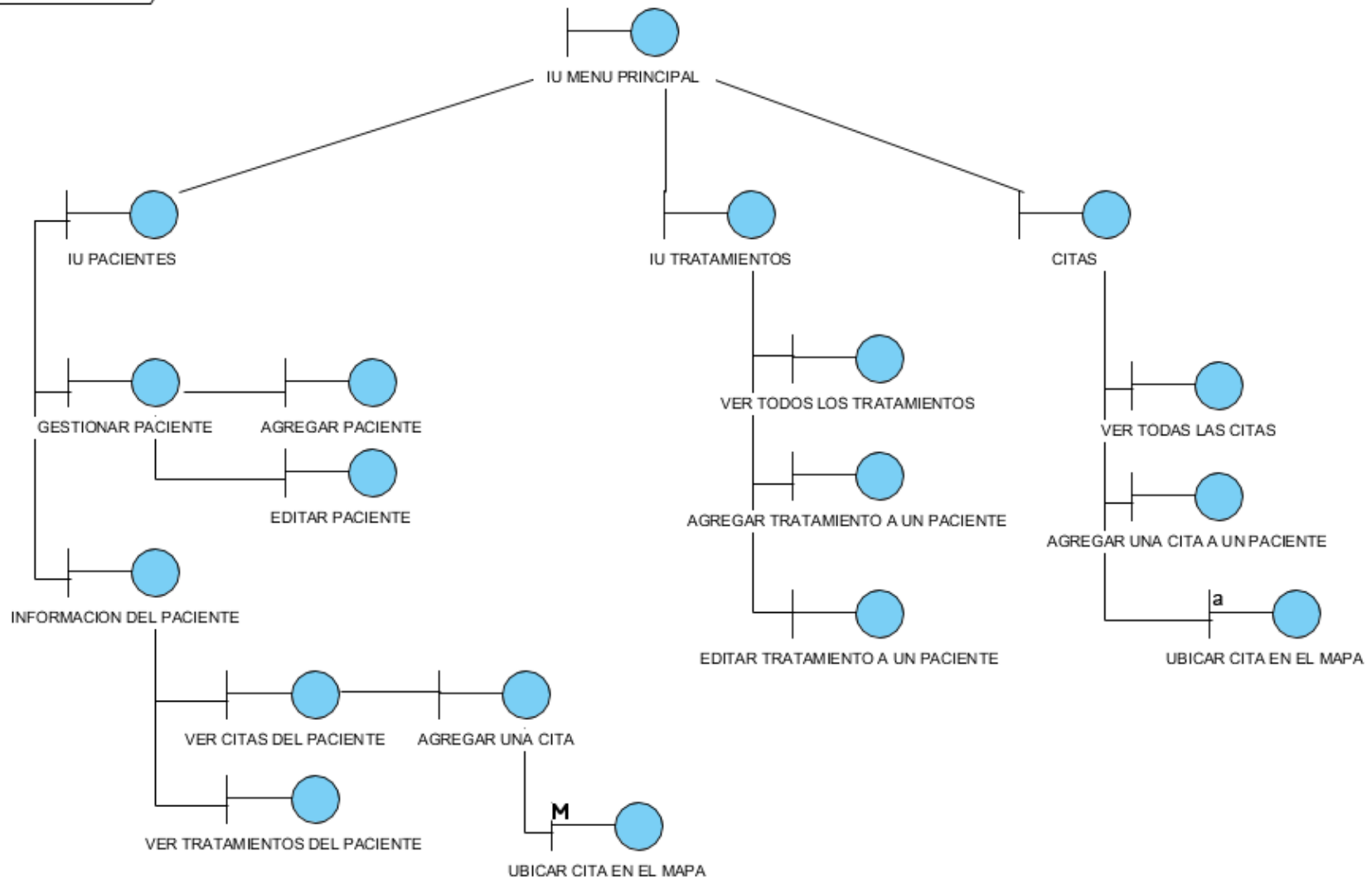
### 5.7.1.13. ENFERMEDADES



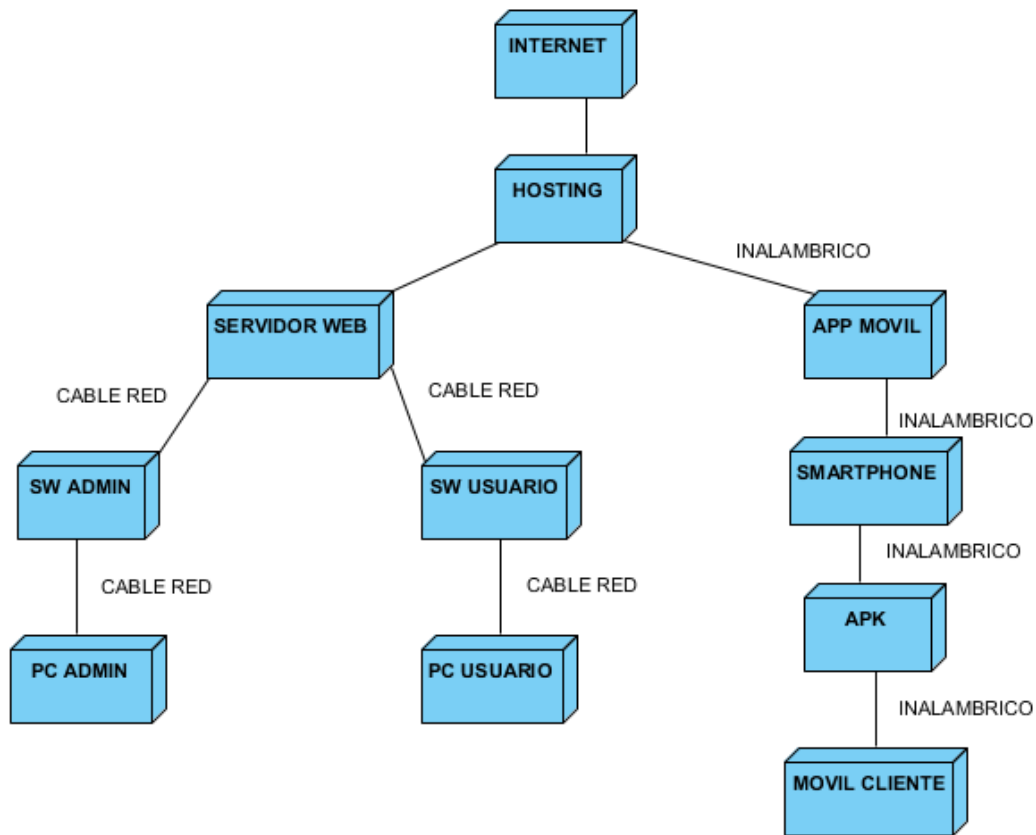


## 5.8. DIAGRAMA DE NAVIGABILIDAD

sd D. NAVIGABILIDAD



## 5.9. DIAGRAMA DE DESPLIEGUE



## 5.10. DESCRIPCION DE ARQUITECTURA

- DACU M. Gestionar Paciente:
- **DACU M. Gestionar Tratamiento**
  - **DACU Consultar Tratamiento:** En esta funcionalidad el usuario podrá consultar información de un tratamiento pudiendo filtrar los datos de diferentes maneras como por ejemplo por la fecha de creación, el nombre del tratamiento, etc.
  - **DACU Agregar Tratamiento:** En esta funcionalidad el usuario podrá registrar un nuevo tratamiento para el paciente, registrar el nombre de los fármacos a consumir, cada cuánto tiempo o con qué frecuencia, por qué tiempo lo hará y si es necesario dar nota de qué doctor prescribió ese tratamiento.
  - **DACU Editar Tratamiento:** En esta funcionalidad el usuario puede editar un tratamiento bajo alguna indicación del doctor o algún factor externo del cuál debe de dar nota al momento de editarlo para dejar registro de cuándo se hizo y por qué.
  - **DACU Dar de baja Tratamiento:** En esta funcionalidad el usuario puede dar de baja a un tratamiento por indicación del médico o algún factor externo

del cuál debe de dar nota al momento de dar de baja a un tratamiento. También el tratamiento será dado de baja automáticamente al terminar.

- **DACU M. Gestionar Cita médica:**
  - **DACU Consultar Cita Médica:** En esta funcionalidad el usuario podrá consultar información de una cita como la fecha, el lugar de la cita, entre otros datos; pudiendo filtrar los datos de diferentes maneras como fecha de registro o fecha de la cita, entre otros.
  - **DACU Agregar Cita Médica:** En esta funcionalidad el usuario podrá registrar una cita médica para el paciente donde debe denotar información relevante como la fecha de la cita, el lugar de la cita, el médico, e incluso ubicar el lugar de la cita en un mapa (opcional).
  - **DACU Editar Cita Médica:** En esta funcionalidad el usuario puede editar una cita médica como la fecha o el lugar, el motivo por el cuál sea editada la cita debe añadirse también para dejar registro de cuándo se hizo y por qué.
  - **DACU Dar de baja Cita Médica:** En esta funcionalidad el usuario puede dar de baja a una cita por indicación del médico o algún factor externo del cuál debe de dar nota al momento de dar de baja a la cita. También la cita será dada de baja automáticamente al pasar el día de esta.
  - **DACU Ubicar Cita Médica en el Mapa:** Esta funcionalidad es un adicional de la anterior donde se puede ubicar el lugar de la cita médica en el mapa y la aplicación lo interpretará como coordenadas que serán guardadas en la base de datos. Esta funcionalidad es adicional y no es indispensable para agregar una cita Médica.
- **DACU Buscar Farmacia:** Esta funcionalidad es un adicional para ubicar las farmacias más cercanas cuando el usuario se encuentre en algún lugar desconocido o haya viajado justo con el paciente y necesite comprar medicamentos. La app móvil debe mostrar la ubicación exacta del usuario mediante el GPS y el API de Google Maps, además debe filtrar los indicadores de lugares para reducirlo a farmacias cercanas.
- **DACU M. Gestionar Contacto:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información del contacto que está asignado al paciente y sus diferentes datos que sean visibles para el asistente.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar los datos de un nuevo contacto que necesite ser enlazado con el paciente.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información habilitada para esta acción por parte del asistente.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja (eliminación lógica – cambio de estado) a un contacto del paciente.
- **DACU M. Añadir un contacto a un paciente:** en esta funcionalidad el usuario podrá enlazar un contacto a un paciente. En caso no esté registrado, puede añadirlo como un nuevo contacto y enlazarlo.

- **DACU M. Añadir una enfermedad a un paciente:** en esta funcionalidad el usuario podrá enlazar una enfermedad a un paciente. En caso no esté registrada se le comunicará a al administrado y este podrá añadirla. El asistente (usuario) NO podrá añadir una enfermedad.
- **DACU M. Gestionar cumplimiento:** En esta funcionalidad el asistente podrá registrar si el paciente tomó la pastilla en el momento que debería. Eso se facilita por la notificación push que se implementará.
- **DACU W. Gestionar Usuarios:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los usuarios.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo usuario al que puede añadirle un login y una clave para el ingreso al sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un usuario. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un usuario denegando el ingreso al sistema por el cambio de estado de Activo a Inactivo.
- **DACU W. Gestionar Perfiles:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los perfiles en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo perfil ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un perfil. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un perfil cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Cargo:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los cargos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo cargo ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un cargo. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un cargo cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Unidad de tiempo:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de las Unidades de Tiempo en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar una nueva unidad de tiempo ingresando los datos requeridos por el sistema.

- **DACU Editar:** En esta funcionalidad el usuario podrá editar información de una unidad de tiempo. El sistema debe habilitar todos los campos disponibles para su edición.
- **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja una unidad de tiempo cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Frecuencia:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de las Frecuencias en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar una nueva frecuencia ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de una frecuencia. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja una frecuencia cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Parentesco:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los parentescos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo parentesco ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un parentesco. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un parentesco cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Profesiones:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de las profesiones en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar una nueva profesión ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de una profesión perfil. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja una profesión cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Acceso:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los accesos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo acceso ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un acceso. El sistema debe habilitar todos los campos disponibles para su edición.

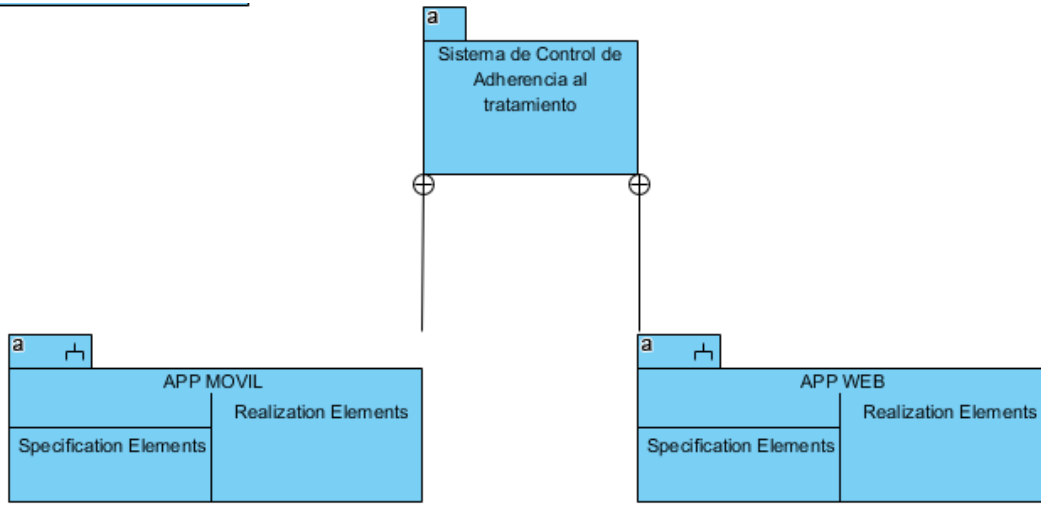


- **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un acceso cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Hospital:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los hospitales en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo hospital ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un hospital. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un hospital cambiando su estado de Activo a Inactivo.
- **DACU W. Gestionar Fármaco:**
  - **DACU Consultar:** En esta funcionalidad el usuario podrá consultar información de los farmacos en detalle.
  - **DACU Agregar:** En esta funcionalidad el usuario podrá registrar un nuevo fármaco ingresando los datos requeridos por el sistema.
  - **DACU Editar:** En esta funcionalidad el usuario podrá editar información de un fármaco. El sistema debe habilitar todos los campos disponibles para su edición.
  - **DACU Dar de Baja:** En esta funcionalidad el usuario podrá dar de baja un fármaco cambiando su estado de Activo a Inactivo.
- **DACU W. Revisar Adherencia:** en esta funcionalidad el usuario doctor podrá ver el progreso de su paciente respecto a cómo está llevando su tratamiento una vez registrado. Esto se presentará a forma de gráficas en vista web para ser más cómoda su revisión.

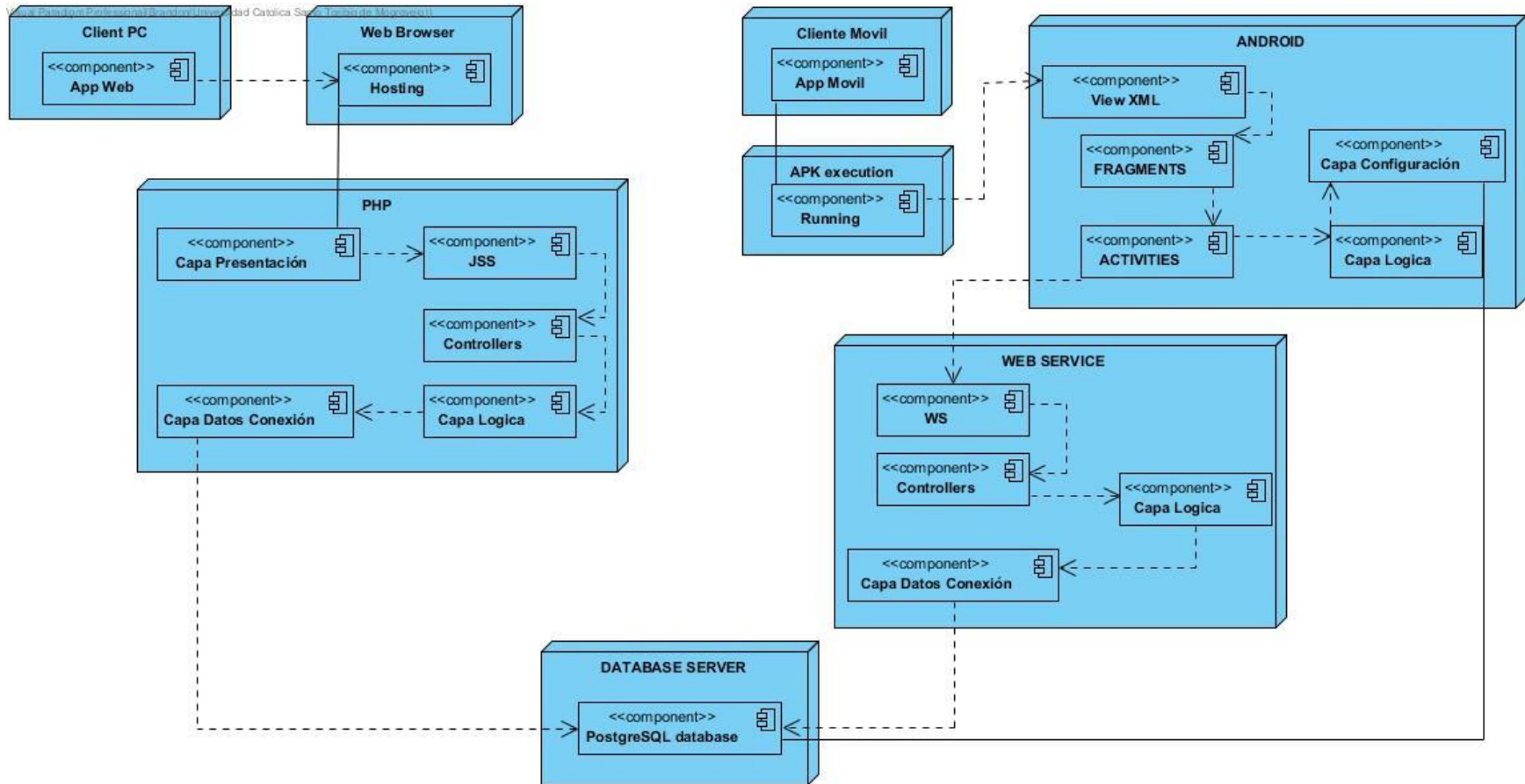
## 6. IMPLEMENTACION

### 6.1.SUBSISTEMAS DE IMLEMENTACIÓN

---



## 6.2. DIAGRAMA DE COMPONENTES



### **6.3. DESCRIPCIÓN DE LA ARQUITECTURA**

## **6.4. LEVANTAMIENTO DE BASE DE DATOS**

### **6.4.1. Creación de la base de datos**

```
CREATE DATABASE bdtesisprueba
```

### **6.4.2. TABLAS**

#### **6.4.2.1. ACCESO**

```
CREATE TABLE public.acceso
(
  id_acceso smallint NOT NULL,
  nombre character varying(30) NOT NULL,
  link character varying(50),
  icono character varying(30),
  estado character(1),
  orden smallint,
  id_acceso_padre smallint,
  CONSTRAINT acceso_pkey PRIMARY KEY (id_acceso)
)
```

#### **6.4.2.2. APLICACIÓN**

```
CREATE TABLE public.aplicacion
(
  id integer NOT NULL,
  id_paciente integer NOT NULL,
  id_farm integer NOT NULL,
  desc_periodo character varying(25) NOT NULL,
  hora_trata time without time zone NOT NULL,
  fecha_trata date NOT NULL,
  fecha_cum timestamp without time zone,
  cumplimiento boolean NOT NULL DEFAULT false,
  CONSTRAINT aplicacion_pkey PRIMARY KEY (id)
)
```

#### **6.4.2.3. CARGO**

```
CREATE TABLE public.cargo
(
  id_cargo smallint NOT NULL,
  nombre character varying(30) NOT NULL,
  estado character(1),
  CONSTRAINT cargo_pkey PRIMARY KEY (id_cargo)
)
```

#### **6.4.2.4. CITA**

```
CREATE TABLE public.cita
(
  id_cita smallint NOT NULL,
  id_hospital smallint,
  dni_paciente character(8),
  fecha_registro date DEFAULT ('now'::text)::date,
  hora_registro time with time zone NOT NULL DEFAULT
('now'::text)::time with time zone,
  fecha_cita date,
  hora_cita time without time zone,
  estado character(1),
  tipo_cita character(1),
)
```

#### **6.4.2.5. CONTACTO**

```
CREATE TABLE public.contacto
(
```

```

dni character(8) NOT NULL,
apellido_paterno character varying(30) NOT NULL,
apellido_materno character varying(30) NOT NULL,
nombre character varying(30) NOT NULL,
estado character(1) NOT NULL,
)

```

#### **6.4.2.6. CONTACTO\_PACIENTE**

```

CREATE TABLE public.contacto_paciente
(
dni_contacto character(8),
dni_paciente character(8),
telefono character varying(20),
id_parentesco smallint,
)

```

#### **6.4.2.7. CORRELATIVO**

```

CREATE TABLE public.correlativo
(
tabla character varying(25) NOT NULL,
numero smallint NOT NULL,
)

```

#### **6.4.2.8. DETALLE\_TRATAMIENTO**

```

CREATE TABLE public.detalle_tratamiento
(
id_tratamiento smallint NOT NULL,
item smallint NOT NULL,
id_farmaco smallint NOT NULL,
id_frecuencia smallint,
id_unidad_tiempo smallint,
cantidad smallint,
periodo smallint,
fecha_inicio timestamp without time zone,
fecha_recomendacion timestamp without time zone,
)

```

#### **6.4.2.9. ENFERMEDAD**

```

CREATE TABLE public.enfermedad
(
id_enfermedad smallint NOT NULL,
descripcion character varying(50) NOT NULL,
estado character(1) NOT NULL,
)

```

#### **6.4.2.10. ENFERMEDAD\_PACIENTE**

```

CREATE TABLE public.enfermedad_paciente
(
id_enfermedad smallint,
dni_paciente character(8),
fecha date,
id smallint NOT NULL,
)

```

#### **6.4.2.11. FARMACO**

```

CREATE TABLE public.farmaco
(
id_farmaco smallint NOT NULL,
nombre character varying(50),
)

```

#### **6.4.2.12. FRECUENCIA**

```

CREATE TABLE public.frecuencia
(
  id_frecuencia smallint NOT NULL,
  descripcion character varying(25) NOT NULL,
  valor smallint,
  estado character(1),
)

```

#### **6.4.2.13. HOSPITAL**

```

CREATE TABLE public.hospital
(
  id_hospital smallint NOT NULL,
  nombre character varying(100) NOT NULL,
  direccion character varying(100) NOT NULL,
  estado character(1) NOT NULL,
)

```

#### **6.4.2.14. PACIENTE**

```

CREATE TABLE public.paciente
(
  dni character(8) NOT NULL,
  apellido_paterno character varying(25),
  apellido_materno character varying(25),
  nombre character varying(25),
  fecha_nacimiento date,
  telefono character varying(15),
  estado character(1) NOT NULL,
  sexo character(1),
)

```

#### **6.4.2.15. PARENTESCO**

```

CREATE TABLE public.parentesco
(
  id_parentesco smallint NOT NULL,
  descripcion character varying(20) NOT NULL,
  estado character(1) NOT NULL,
)

```

#### **6.4.2.16. PERFIL**

```

CREATE TABLE public.perfil
(
  id_perfil smallint NOT NULL,
  nombre character varying(20),
  estado character(1),
)

```

#### **6.4.2.17. PERFIL\_ACCESO**

```

CREATE TABLE public.perfil_acceso
(
  id_perfil smallint,
  id_acceso smallint,
)

```

#### **6.4.2.18. PERSONAL**

```

CREATE TABLE public.personal
(
  dni character(8) NOT NULL,
  apellido_paterno character varying(50),
  apellido_materno character varying(50),
  nombre character varying(50),
  fecha_nacimiento date,
)

```

```

direccion character varying(150),
telefono character varying(20),
estado character(1) NOT NULL,
id_profesion smallint,
id_usuario smallint,
id_cargo smallint,
)

```

#### **6.4.2.19. PROFESION**

```

CREATE TABLE public.profesion
(
  id_profesion smallint NOT NULL,
  nombre character varying(50) NOT NULL,
  estado character(1) NOT NULL,
)

```

#### **6.4.2.20. TIPO\_TRATAMIENTO**

```

CREATE TABLE public.tipo_tratamiento
(
  id_tipo_tratamiento smallint NOT NULL,
  nombre character varying(50),
  estado character(1),
)

```

#### **6.4.2.21. TRATAMIENTO**

```

CREATE TABLE public.tratamiento
(
  id_tratamiento smallint NOT NULL,
  id_cita smallint,
  id_tipo_tratamiento smallint,
  id_personal character(8),
  descripcion character varying(100),
  fecha_registro date DEFAULT ('now'::text)::date,
  fecha_indicada timestamp without time zone,
)

```

#### **6.4.2.22. UNIDAD**

```

CREATE TABLE public.unidad
(
  id_unidad smallint NOT NULL,
  descripcion character varying(20) NOT NULL,
  estado character(1) NOT NULL,
)

```

#### **6.4.2.23. UNIDAD\_TIEMPO**

```

CREATE TABLE public.unidad_tiempo
(
  id_unidad_tiempo smallint NOT NULL,
  descripcion character varying(20),
  estado character(1),
)

```

#### **6.4.2.24. USUARIO**

```

CREATE TABLE public.usuario
(
  id_usuario smallint NOT NULL,
  login character(8) NOT NULL,
  clave character(32) NOT NULL,
  id_perfil smallint,
  estado character(1),
)

```

### **6.5. CÓDIGO DE LAS CLASES DEL SISTEMA**

## 6.5.1. CLASES EN PHP

### 6.5.1.1. ACCESO

```
class Acceso extends Conexion{
    private $id;
    private $nombre;
    private $link;
    private $icono;

    function getId()

    function getNombre()
    function getLink()
    function getIcono()
    function setId($id)
    function setNombre($nombre)
    function setLink($link)
    function setIcono($icono)
    public function listar() {

public function agregar() {
    $sql="insert into acceso(id_acceso, nombre, link, icono,
estado) values (:p_cod, :p_nom, :p_link, :p_icono, 'A')";
        // ASIGNAR LOS VALORES PARA LAS COLUMNAS
    $sentencia->bindParam(":p_cod", $this->getId());
    $sentencia->bindParam(":p_nom", $this->getNombre());
    $sentencia->bindParam(":p_link", $this->getLink());
    $sentencia->bindParam(":p_icono", $this->getIcono());

    $sentencia->execute();

public function editar() {
    $sql = "update acceso set nombre = :p_nom, link =:p_link, icono =
:p_icono where id_acceso = :p_cod";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->bindParam(":p_cod", $this->getId());
    $sentencia->bindParam(":p_nom", $this->getNombre());
    $sentencia->bindParam(":p_link", $this->getLink());
    $sentencia->bindParam(":p_icono", $this->getIcono());

    $sentencia->execute();
}

public function eliminar() {
    $sql = "update acceso set estado = 'I' where id_acceso = :p_cod";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->bindParam(":p_cod", $this->getId());
    $sentencia->execute();
    return true;
}

public function leerdatos($p_cod) {
    $sql="Select id_acceso, nombre, link, icono from acceso where
id_acceso = :p_cod";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->bindParam(":p_cod", $p_cod);
    $sentencia->execute();
    $resultado = $sentencia->fetch(PDO::FETCH_ASSOC);
    return $resultado;
}
}
```

### 6.5.1.2. CARGO



```

    private $id;
    private $nombre;
    function getId()
    function getNombre()
    function setId($id)
    function setNombre($nombre)

public function listar() {

    $sql = " Select id_cargo, nombre, estado from cargo order by 2";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->execute();
    $resultado = $sentencia->fetchAll(PDO::FETCH_ASSOC);
    return $resultado;
}

public function agregar() {
/*INSERTAR EN LA TABLA LABORATORIO*/
$sql="insert into cargo(id_cargo, nombre, estado) values (:p_cod,
:p_nom, 'A')";
$sentencia = $this->dblink->prepare($sql);
// ASIGNAR LOS VALORES PARA LAS COLUMNAS
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->bindParam(":p_nom", $this->getNombre());
$sentencia->execute();
/*INSERTAR EN LA TABLA LABORATORIO*/
}

public function editar() {
$sql = "update cargo set nombre = :p_nom where id_cargo = :p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_nom", $this->getNombre());
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
return true;
}

public function eliminar() {

$sql = "update cargo set estado = 'I' where id_cargo = :p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
return true;
}

```

### 6.5.1.3. ENFERMEDAD

```

    private $id;
    private $descripcion;

    function getId()
    function getDescripcion()
    function setId($id)
    function setDescripcion($descripcion)

public function listar() {
    $sql = " Select id_enfermedad, descripcion, estado from enfermedad
    order by 2";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->execute();
    $resultado = $sentencia->fetchAll(PDO::FETCH_ASSOC);

```

```

return $resultado;
}
public function agregar() {
$sql="Select * from f_generar_correlativo('enfermedad') as nc";
$sentencia = $this->dblink->prepare($sql);
$sentencia->execute();
/*INSERTAR EN LA TABLA LABORATORIO*/
$sql="insert into enfermedad(id_enfermedad, descripcion, estado)
values (:p_cod, :p_nom, 'A')";
$sentencia = $this->dblink->prepare($sql);
// ASIGNAR LOS VALORES PARA LAS COLUMNAS
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->bindParam(":p_nom", $this->getDescripcion());
sentencia->execute();
}

public function editar() {
$sql = "update enfermedad set descripcion = :p_nom where
id_enfermedad = :p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_nom", $this->getDescripcion());
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
return true;
}

public function eliminar() {
$sql = "update enfermedad set estado = 'I' where id_enfermedad =
:p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
return true;
}

```

#### 6.5.1.4. FRECUENCIA

```

private $id;
private $descripcion;
private $valor;

function getId()
function getDescripcion()
function setId($id)
function setDescripcion($descripcion)
function getValor()
function setValor($valor)

public function listar() {
$sql = " Select id_frecuencia, descripcion,valor, estado from
frecuencia order by 2";
$sentencia = $this->dblink->prepare($sql);
$sentencia->execute();

$resultado = $sentencia->fetchAll(PDO::FETCH_ASSOC);
return $resultado;
}

public function agregar() {
$this->dblink->beginTransaction();
/*INSERTAR EN LA TABLA LABORATORIO*/

```

```

$sql="insert into frecuencia(id_frecuencia, descripcion,valor,
estado) values (:p_cod, :p_nom,:p_valor, 'A')";
$sentencia = $this->dblink->prepare($sql);
// ASIGNAR LOS VALORES PARA LAS COLUMNAS
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->bindParam(":p_nom", $this->getDescripcion());
$sentencia->bindParam(":p_valor", $this->getValor());

$sentencia->execute();
}

```

```

public function editar(){
$sql = "update frecuencia set descripcion = :p_nom, valor =
:p_valor where id_frecuencia = :p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_nom", $this->getDescripcion());
$sentencia->bindParam(":p_valor", $this->getValor());
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
        return true;
}

```

```

public function eliminar(){
$sql = "update frecuencia set estado = 'I' where id_frecuencia =
:p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
return true;
}

```

### 6.5.1.5. HOSPITAL

```

    private $id;
    private $nombre;
    private $direccion;
    private $estado;

    function getId()
    function getNombre()
    function getDireccion()
    function getEstado()
    function setId($id)
    function setNombre($nombre)
    function setDireccion($direccion)
    function setEstado($estado)

public function listar() {
    $sql = " Select id_hospital, nombre, direccion, estado from
hospital order by 2";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->execute();
    $resultado = $sentencia->fetchAll(PDO::FETCH_ASSOC);
        return $resultado;
}

public function agregar(){
/*INSERTAR EN LA TABLA LABORATORIO*/
    $sql="insert into hospital(id_hospital, nombre, direccion, estado)
values (:p_cod, :p_nom, :p_dir, 'A')";
    $sentencia = $this->dblink->prepare($sql);
    // ASIGNAR LOS VALORES PARA LAS COLUMNAS
    $sentencia->bindParam(":p_cod", $this->getId());

```

```

$sentencia->bindParam(":p_nom", $this->getNombre());
$sentencia->bindParam(":p_dir", $this->getDireccion());
$sentencia->execute();
}

```

```

public function editar(){
    $sql = "update hospital set nombre = :p_nom, direccion = :p_dir
    where id_hospital = :p_cod";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->bindParam(":p_nom", $this->getNombre());
    $sentencia->bindParam(":p_dir", $this->getDireccion());
    $sentencia->bindParam(":p_cod", $this->getId());
    $sentencia->execute();
    return true;
}

```

```

public function eliminar(){
    $sql = "update hospital set estado = 'I' where id_hospital =
    :p_cod";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->bindParam(":p_cod", $this->getId());
    $sentencia->execute();
    return true;
}

```

### 6.5.1.6. PARENTESCO

```

    private $id;
    private $descripcion;
    function getId()
    function getDescripcion()
    function setId($id)
    function setDescripcion($descripcion)

```

```

public function listar() {
    $sql = " Select id_parentesco, descripcion, estado from parentesco
    order by 2";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->execute();
    $resultado = $sentencia->fetchAll(PDO::FETCH_ASSOC);
    return $resultado;
}

```

```

public function agregar(){
    $sql="insert into parentesco(id_parentesco, descripcion, estado)
    values (:p_cod, :p_nom, 'A')";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->bindParam(":p_cod", $this->getId());
    $sentencia->bindParam(":p_nom", $this->getDescripcion());
    $sentencia->execute();
}

```

```

public function editar(){
    $sql = "update parentesco set descripcion = :p_nom where
    id_parentesco = :p_cod";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->bindParam(":p_nom", $this->getDescripcion());
    $sentencia->bindParam(":p_cod", $this->getId());
    $sentencia->execute();
    return true;
}

```

```

public function eliminar(){

```

```

$sql = "update parentesco set estado = 'I' where id_parentesco =
:p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
return true;
}

```

### 6.5.1.7. PERFIL

```

private $id;
private $nombre;
function getId()
function getNombre()
function setId($id)
function setNombre($nombre)
public function listar() {
$sql = " Select id_perfil, nombre, estado from perfil order by 2";
$sentencia = $this->dblink->prepare($sql);
$sentencia->execute();
$resultado = $sentencia->fetchAll(PDO::FETCH_ASSOC);
return $resultado;
}
public function agregar(){
$sql="Select * from f_generar_correlativo('perfil') as nc";
$sentencia = $this->dblink->prepare($sql);
$sentencia->execute();
$sql="insert into perfil(id_perfil, nombre, estado) values (:p_cod,
:p_nom, 'A')";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->bindParam(":p_nom", $this->getNombre());
$sentencia->execute();
}

public function editar(){
$sql = "update perfil set nombre = :p_nom where id_perfil =
:p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_nom", $this->getNombre());
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
return true;
}

public function eliminar(){
$sql = "update perfil set estado = 'I' where id_perfil = :p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
return true;
}

public function leerdatos($p_cod){
$sql="Select id_perfil, nombre from perfil where id_perfil =
:p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_cod", $p_cod);
$sentencia->execute();
$resultado = $sentencia->fetch(PDO::FETCH_ASSOC);
}

```

### 6.5.1.8. PROFESION

```

    private $id;
    private $nombre;
    function getId()
    function getNombre()
    function setId($id)
    function setNombre($nombre)

public function listar() {
    $sql = " Select id_profesion, nombre, estado from profesion order
    by 2";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->execute();
    $resultado = $sentencia->fetchAll(PDO::FETCH_ASSOC);
    return $resultado;
    }
public function agregar() {
    $sql = "Select * from f_generar_correlativo('profesion') as nc";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->execute();
    /* INSERTAR EN LA TABLA LABORATORIO */
    $sql = "insert into profesion(id_profesion, nombre, estado) values
    (:p_cod, :p_nom, 'A')";
    $sentencia = $this->dblink->prepare($sql);
    // ASIGNAR LOS VALORES PARA LAS COLUMNAS
    $sentencia->bindParam(":p_cod", $this->getId());
    $sentencia->bindParam(":p_nom", $this->getNombre());
    $sentencia->execute();
    }
public function editar() {
    $sql = "update profesion set nombre = :p_nom where id_profesion =
    :p_cod";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->bindParam(":p_nom", $this->getNombre());
    $sentencia->bindParam(":p_cod", $this->getId());
    $sentencia->execute();
    return true;
    }
public function eliminar() {
    $sql = "update profesion set estado = 'I' where id_profesion =
    :p_cod";
    $sentencia = $this->dblink->prepare($sql);

    $sentencia->bindParam(":p_cod", $this->getId());
    $sentencia->execute();
    return true;

```

#### 6.5.1.9. SESION

```

    private $login;
    private $clave;
    function getLogin()
    function getClave()
    function setLogin($login)
    function setClave($clave)

public function obtenerAcceso(){

public function validarSesion(){
    $sql = " select u.id_usuario, per.dni, per.apellido_paterno,
    per.apellido_materno, per.nombre, u.clave, p.id_perfil,p.nombre as
    perfil, per.estado as estado_personal, u.estado as estado_usuario
    from personal per

```

```

        inner join usuario u on u.login = per.dni
        inner join perfil p on p.id_perfil = u.id_perfil
    where
        u.login = :p_login";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->bindParam(":p_login", $this->getLogin());
    //echo $sql;
    $sentencia->execute();
    $resultado = $sentencia->fetch(PDO::FETCH_ASSOC); //
    utilizo fetch cuando se recupera 1 solo registro

    if ($sentencia->rowCount()) {
        // si me devuelve registros entonces el usuario si
    existe
        if ($resultado["clave"] == md5($this->getClave())) { //
    comprobamos que la clave sea la misma
            if ($resultado["estado_personal"] == "A" &&
    $resultado["estado_usuario"] == "A" ) {
                session_name("sistema_tesis_web");
                session_start();
                $_SESSION["s_usuario"] = $resultado["nombre"].
    ' '. $resultado["apellido_paterno"];
                $_SESSION["s_perfil"] = $resultado["perfil"];
                $_SESSION["s_dni"]=$resultado["dni"];
                $_SESSION["s_id_perfil"] =
    $resultado["id_perfil"];
                $this->obtenerAcceso();
                return "SI"; // si ingresa
            } else {
                return "UI"; //usuario inactivo
            }
        } else {
            return "CI"; //clave incorrecta
        }
        } else {
            return "NE";
        }
    }
}

```

### 6.5.1.10. TIPOTRATAMIENTO

```

    private $id;
    private $nombre;

    function getId()
    function getNombre()
    function setId($id)
    function setNombre($nombre)
    public function listar() {
    $sql = " Select id_tipo_tratamiento, nombre, estado from
    tipo_tratamiento order by 2";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->execute();
    $resultado = $sentencia->fetchAll(PDO::FETCH_ASSOC);
    return $resultado;
    }
    public function agregar() {
    /* INSERTAR EN LA TABLA LABORATORIO */
    $sql = "insert into tipo_tratamiento(id_tipo_tratamiento, nombre,
    estado) values (:p_cod, :p_nom, 'A')";
    $sentencia = $this->dblink->prepare($sql);
    // ASIGNAR LOS VALORES PARA LAS COLUMNAS
    }
    }

```

```

$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->bindParam(":p_nom", $this->getNombre());
$sentencia->execute();
/* INSERTAR EN LA TABLA LABORATORIO */
}
public function editar() {
$sql = "update tipo_tratamiento set nombre = :p_nom where
id_tipo_tratamiento = :p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_nom", $this->getNombre());
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
return true;
}
public function eliminar() {
$sql = "update tipo_tratamiento set estado = 'I' where
id_tipo_tratamiento = :p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
return true;
}

```

#### 6.5.1.11. UNIDADMEDIDA

```

private $codigo_unidad_medida;
private $descripcion;

function getCodigo_unidad_medida()
function getDescripcion()
function setCodigo_unidad_medida($codigo_unidad_medida)
function setDescripcion($descripcion)
public function cargarDatos(){
$sql = "select codigo_unidad_medida, descripcion from unidad_medida
order by 2";
$sentencia = $this->dblink->prepare($sql);
$sentencia->execute();
$resultado = $sentencia->fetchAll(PDO::FETCH_ASSOC);
return $resultado;
}

```

#### 6.5.1.12. UNIDADTIEMPO

```

private $id;
private $descripcion;

function getId()
function setId($id)
function setDescripcion($descripcion)

public function listar() {
$sql = " Select id_unidad_tiempo, descripcion, estado from
unidad_tiempo order by 2";
$sentencia = $this->dblink->prepare($sql);
$sentencia->execute();
$resultado = $sentencia->fetchAll(PDO::FETCH_ASSOC);
return $resultado;
}
public function agregar() {
$sql="insert into unidad_tiempo(id_unidad_tiempo, descripcion,
estado) values (:p_cod, :p_nom, 'A')";

$sentencia = $this->dblink->prepare($sql);

```



```

// ASIGNAR LOS VALORES PARA LAS COLUMNAS
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->bindParam(":p_nom", $this->getDescripcion());
$sentencia->execute();
}
public function editar(){
$sql = "update unidad_tiempo set descripcion = :p_nom where
id_unidad_tiempo = :p_cod";
$sentencia = $this->dblink->prepare($sql);
$sentencia->bindParam(":p_nom", $this->getDescripcion());
$sentencia->bindParam(":p_cod", $this->getId());
$sentencia->execute();
return true;
}

public function eliminar(){
    $sql = "update unidad_tiempo set estado = 'I' where
id_unidad_tiempo = :p_cod";
    $sentencia = $this->dblink->prepare($sql);
    $sentencia->bindParam(":p_cod", $this->getId());
    $sentencia->execute();
    return true;
}
}

```

## 6.5.2. CLASES EN ANDROID

### 6.5.2.1. CITA

```

public class Cita {
private int id_cita;
private String hospital;
private String fecha_cita;
private String hora_cita;
private String estado;

public int getId_cita()
public void setId_cita(int id_cita
public String getHospital
public void setHospital(String hospital
public String getFecha_cita
public void setFecha_cita(String fecha_cita
public String getHora_cita
public void setHora_cita(String hora_cita
public String getEstado
public void setEstado(String estado)

```

### 6.5.2.2. HOSPITAL

```

public class Hospital {
private String nombre;
private int id_hospital;

public static ArrayList<Hospital> listaHospitales = new
ArrayList<Hospital>();

public String getNombre()
public void setNombre(String nombre
public int getId_hospital()
public void setId_hospital(int id_hospital) }
}

```

### 6.5.2.3. PACIENTE

```

public class Paciente {
private String dni;

```

```

private String nombre;
private String fecha_nac;
private String telefono;
private String estado;
private String sexo;
private boolean seleccionado;

public String getDni()
public void setDni(String dni
public String getNombre()
public void setNombre(String nombre
public String getFecha_nac
public void setFecha_nac(String fecha_nac
public String getTelefono()
public void setTelefono(String telefono
public String getEstado()
public void setEstado(String estado
public String getSexo
public void setSexo(String sexo
public boolean isSelected()
}

```

#### 6.5.2.4. SESION PREFERENCES

```

private static final String NOMBRE_PREFERENCIA =
"appTesis";
private static final String INICIO_SESION =
"inicioSesion";
private static final String DNI = "DNI";
private static final String NOMBRE = "NOMBRE";
private static final String dniPaciente = "DNIPACIENTE";
private static final String nombrePaciente =
"NOMBREPACIENTE";
private static final String id_cita = "-1";
private static final String HOSPITAL = "HOSPITAL";
private static final String HORA = "HORA";
private static final String FECHA = "FECHA";

public SesionPreferences(Context context){
this.appTesis = context;
preferencias =
appTesis.getSharedPreferences(NOMBRE_PREFERENCIA,privado);
editor = preferencias.edit();
}

public void iniciarSesion(String dni){
editor.putBoolean(INICIO_SESION, true);
editor.putString(dni, "");
editor.commit();
}

public void cerrarSesion
public void validarSesion
public String obtenerDni
public void guardarNombre(String nombre
public String obtenerNombre
public void guardarDniPaciente(String dni
public String obtenerDniPaciente()

```

```

public void guardarNombrePaciente(String nombre)
public String obtenerNombrePaciente()
public void guardarIdCita(String id)
public String obtenerIdCita()
public void guardarHospital(String hos)
public String obtenerHospital()
public void guardarHoraCita(String hora)
public String obtenerHoraCita()
public void guardarFechaCita(String fecha)
public String obtenerFechaCita()

```

## 6.6. CÓDIGO DE LOS PROCEDIMIENTOS PRINCIPALES

### 6.6.1. PROCEDIMIENTOS PRINCIPALES EN PHP

#### 6.6.1.1. AGREGAR CITA

```

    $id_hospital = $_POST["id_hospital"];
    $dni = $_POST["dni_paciente"];
    $fecha = $_POST["fecha_cita"];
    $hora = $_POST["hora_cita"];
    $fechaFinal = explode("/", $fecha);
    $id_cita = $_POST["id_cita"];
    $obj = new Cita();
    /*if ($tipoOperacion == "agregar") {*/
        $obj->setDni_paciente($dni);
        $obj->setId_hospital($id_hospital);
        $obj->setFecha_cita($fechaFinal[2]."-".$fechaFinal[1]."-".
$.fechaFinal[0]);
        $obj->setHora_cita($hora);
    if($id_cita == "-1"){
        $resultado = $obj->fnAgregar();
        if ($resultado) {
Funciones::imprimeJSON(200, "Agregado correctamente", "");
        }
    }else{
        $obj->setId($id_cita);
        $resultado = $obj->fnEditar();
        if ($resultado) {
Funciones::imprimeJSON(200, "Cita actualizada correctamente", "");
        }
    }
    /*} else { //EDITAR
        if (!isset($_POST["p_cod"]) || empty($_POST["p_cod"])) {
Helper::imprimeJSON(500, "Falta completar datos para editar", "");
exit();
        }
        $cod_pe = $_POST["p_cod"]; // espero que envíe el
codigoLaboratorio con el nombre de p_cod_lab que se enviará desde
el script
        $obj->setId($cod_pe);
        $obj->setNombre($nombre);
        $obj->setDireccion($dir);

        $resultado = $obj->editar();

        if ($resultado) {
Helper::imprimeJSON(200, "Editado correctamente", "");
        }
    }*/
} catch (Exception $exc) {

```

```
Funciones::imprimeJSON(200, "Error", $exc->getMessage());
}
```

### 6.6.1.2. SESION

```
$objSesion = new Sesion();
$objSesion->setLogin($login);
$objSesion->setClave($clave);
$resultado = $objSesion->fnIniciarSesion();
```

## 6.6.2. PROCEDIMIENTOS PRINCIPALES EN ANDROID

### 6.6.2.1. MANTENIMIENTO DE CITAS

```
public class AgregarEditarCitaActivity extends AppCompatActivity
{
    private static final String CERO = "0";
    private static final String DOS_PUNTOS = ":";
    public final Calendar c = Calendar.getInstance();
    //Variables para obtener la hora hora
    int horaH;
    int minuto;
    private String dniPaciente,nombrePaciente,id_cita,
hospitalLeer, horaLeer, fechaLeer;
    TextView txtNombrePaciente;
    EditText txtFecha, txtHora;
    Button btnGuardar,btnCancelar;
    Spinner spHospital;
    String id_hospital,hora,fecha;
    int ano=0,mes=0,dia=0;
    SesionPreferences sesion;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_agregar_editar_cita);
        sesion = new SesionPreferences(this);
        sesion.validarSesion();

        /*Bundle p = this.getIntent().getExtras();
        dniPaciente = p.getString("dni");
        nombrePaciente = p.getString("nombre");
        id_cita = p.getString("id_cita");*/
        dniPaciente = sesion.obtenerDniPaciente();
        nombrePaciente = sesion.obtenerNombrePaciente();
        id_cita = sesion.obtenerIdCita();
        horaH = c.get(Calendar.HOUR_OF_DAY);
        minuto = c.get(Calendar.MINUTE);

        if(!id_cita.equalsIgnoreCase("-1")){
            hospitalLeer = sesion.obtenerHospital();
            horaLeer = sesion.obtenerHoraCita();
            fechaLeer = sesion.obtenerFechaCita();
            horaH =
Integer.parseInt(sesion.obtenerHoraCita().substring(0,2));
            minuto =
Integer.parseInt(sesion.obtenerHoraCita().substring(3,5));
        }
        txtNombrePaciente =
(TextView)findViewById(R.id.txtNombrePacienteCita);
        txtNombrePaciente.setText(nombrePaciente);

        txtFecha = (EditText)findViewById(R.id.txtFechaCita);
        Date fechaActual = new Date();
        SimpleDateFormat formatoFecha = new
```

```

SimpleDateFormat("dd/MM/yyyy");
txtFecha.setText(formatoFecha.format(fechaActual));

txtFecha.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        fecha();
    }
});
txtHora = (EditText)findViewById(R.id.txtHoraCita);
txtHora.setOnFocusChangeListener(new
View.OnFocusChangeListener() {
    @Override
    public void onFocusChange(View view, boolean b) {
        hideKeyboard(view);
    }
});
txtHora.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        hideKeyboard(view);
        obtenerHora();
    }
});
spHospital = (Spinner)findViewById(R.id.spHospital);
new cargarSpinnerHospital().execute();
btnGuardar = (Button)findViewById(R.id.btnGuardar);
btnCancelar = (Button)findViewById(R.id.btnCancelar);
btnGuardar.setOnClickListener(new View.OnClickListener()
{
    @Override
    public void onClick(View view) {
        boolean r =
Funciones.mensajeConfirmacion(AgregarEditarCitaActivity.this,
"Confirme", "¿Desea guardar esta cita?");
        if (r){
            id_hospital =
String.valueOf(Hospital.listaHospitales.get(spHospital.getSelected
ItemPosition()).getId_hospital());
            fecha = txtFecha.getText().toString();
            hora = txtHora.getText().toString();
            new guardarCita().execute();
        }
    }
});
btnCancelar.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        boolean r =
Funciones.mensajeConfirmacion(AgregarEditarCitaActivity.this,
"Confirme", "¿Desea salir?");
        if (r){
            AgregarEditarCitaActivity.this.finish();
        }
    }
});
}
private void fecha(){
    Date fechaActual = new Date();
    SimpleDateFormat formatoFecha = new

```

```

SimpleDateFormat("dd/MM/yyyy");
//txtFecha.setText(formatoFecha.format(fechaActual));
if(sesion.obtenerFechaCita().contains("FECHA")){
    dia =
Integer.parseInt(String.valueOf(formatoFecha.format(fechaActual)
.substring(0,2)));
    mes =
Integer.parseInt(String.valueOf(formatoFecha.format(fechaActual)
.substring(3,5)) -1;
    ano =
Integer.parseInt(String.valueOf(formatoFecha.format(fechaActual)
.substring(6,10)));
    }else{
        dia =
Integer.parseInt(sesion.obtenerFechaCita().substring(9,10));
        mes =
Integer.parseInt(sesion.obtenerFechaCita().substring(6,7)) -1;
        ano =
Integer.parseInt(sesion.obtenerFechaCita().substring(0,4));
    }
    DatePickerDialog datePickerDialog = new
DatePickerDialog(this, R.style.Theme_AppCompat_Light_Dialog,new
DatePickerDialog.OnDateSetListener() {
        @Override
        public void onDateSet(DatePicker view, int year, int
monthOfYear, int dayOfMonth) {
            String mes = "";
            if(monthOfYear<9){
                mes = "0"+(monthOfYear+1);
            }else{
                mes = String.valueOf(monthOfYear+1);
            }
            String dia = "";
            if(dayOfMonth<10){
                dia = "0"+dayOfMonth;
            }else{
                dia = String.valueOf(dayOfMonth);
            }
            txtFecha.setText(dia+"/"+mes+"/"+year);
        }
    }
    ,ano,mes,dia);
    datePickerDialog.show();

}

public class guardarCita extends AsyncTask<Void, Void,
String> {
    @Override
    protected String doInBackground(Void... params) {
        // TODO: attempt authentication against a network
service.

        String ws = Funciones.URL_WS +
"cita.agregar.editar.php";
        //id_cita, id_hospital, dni_paciente, fecha_cita,
hora_cita);
        HashMap parametros = new HashMap<String, String>();
        parametros.put("id_hospital", id_hospital);
        parametros.put("dni_paciente", dniPaciente);
        parametros.put("fecha_cita", fecha);
        parametros.put("hora_cita", hora);
        parametros.put("id_cita", id_cita);
    }
}

```

```

        String resultado = new
Funciones().getHttpContent(ws, parametros);
        return resultado;
    }
    @Override
    protected void onPostExecute(final String resultado) {
        try {
            JSONObject jsonObject = new
JSONObject(resultado);
            String mensaje =
jsonObject.getString("mensaje");
            int estado = jsonObject.getInt("estado");
            if(estado == 200){
                if(mensaje.equalsIgnoreCase("Error")){
Funciones.mensajeError(AgregarEditarCitaActivity.this, "Error",
jsonObject.getString("datos"));
                }else{
Funciones.mensajeInformacion(AgregarEditarCitaActivity.this,
"Éxito", mensaje);

                    Intent intent = new Intent();
                    /*Bundle p = new Bundle();
                    dniPaciente = p.getString("dni");
                    nombrePaciente = p.getString("nombre");
                    p.putString("dni",dniPaciente);
                    p.putString("nombre", dniPaciente);*/
                    dniPaciente =
sesion.obtenerDniPaciente();
                    nombrePaciente =
sesion.obtenerNombrePaciente();
                    setResult(RESULT_OK,intent);
                    finish();
                }
            }
        } catch (Exception e){
        }
    }
}

public class cargarSpinnerHospital extends AsyncTask<Void,
Void, String> {
    @Override
    protected String doInBackground(Void... params) {
        // TODO: attempt authentication against a network
service.

        String ws = Funciones.URL_WS +
"hospital.listar.php";
        HashMap parametros = new HashMap();
        String resultado = new
Funciones().getHttpContent(ws, parametros);
        return resultado;
    }
    protected void onPostExecute(final String resultado) {

        if ( ! resultado.isEmpty() ) {
            try {
                JSONObject json = new JSONObject(resultado);

                int estado = json.getInt("estado");
                if (estado==200){
                    JSONArray jsonArray =
json.getJSONArray("datos");
                    Hospital.listaHospitales.clear();
                }
            }
        }
    }
}

```

```

        String arrayDepartamentos[] = new
String[jsonArray.length()];
        for (int i = 0; i < jsonArray.length();
i++) {
                JSONObject objItem =
jsonArray.getJSONObject(i);
                Hospital objHospital= new
Hospital();
                objHospital.setId_hospital(objItem.getInt("cod"));
                objHospital.setNombre(objItem.getString("nombre"));
                Hospital.listaHospitales.add(objHospital);
                arrayDepartamentos[i]=objItem.getString("nombre");
                ArrayAdapter<String> adapter = new
ArrayAdapter<String>(AgregarEditarCitaActivity.this,android.R.la
yout.simple_spinner_dropdown_item,arrayDepartamentos);
                spHospital.setAdapter(adapter);
                if(!id_cita.equalsIgnoreCase("-1")){
Funciones.selectedItemSpinner(spHospital,hospitalLeer);
                txtHora.setText(horaLeer);
                txtFecha.setText(fechaLeer);
                }
        }
        }catch (Exception e){
        //
Toast.makeText(EditorPerfilFragment.this.getContext(),
e.getMessage(), Toast.LENGTH_SHORT).show();
        }
    }
}

public void hideKeyboard(View view) {
    InputMethodManager inputMethodManager
=(InputMethodManager) getSystemService (Activity.INPUT_METHOD_SERV
ICE);

inputMethodManager.hideSoftInputFromWindow(view.getWindowToken()
, 0);
}

private void obtenerHora(){
    TimePickerDialog recogerHora = new
TimePickerDialog(this, new TimePickerDialog.OnTimeSetListener()
{
        @Override
        public void onTimeSet(TimePicker view, int
hourOfDay, int minute) {
            //Formateo el hora obtenido: antepone el 0 si
son menores de 10
            String horaFormateada = (hourOfDay < 10)?
String.valueOf(CERO + hourOfDay) : String.valueOf(hourOfDay);
            //Formateo el minuto obtenido: antepone el 0 si
son menores de 10
            String minutoFormateado = (minute < 10)?
String.valueOf(CERO + minute):String.valueOf(minute);
            //Obtengo el valor a.m. o p.m., dependiendo de
la selección del usuario
            //Muestro la hora con el formato deseado
            txtHora.setText(horaFormateada + DOS_PUNTOS +
minutoFormateado + ":00");
        }
    }
}

```



```

        //Estos valores deben ir en ese orden
        //Al colocar en false se muestra en formato 12 horas
y true en formato 24 horas
        //Pero el sistema devuelve la hora en formato 24
horas
        }, horaH, minuto, false);

        recogerHora.show();
    }
    @Override
    public void onBackPressed() {
        super.onBackPressed();
        VerCitasActivity.id_cita = this.id_cita;
        VerCitasActivity.dniPaciente = this.dniPaciente;
        VerCitasActivity.nombrePaciente = this.nombrePaciente;
    }
}

```

#### 6.6.2.2. LISTADO DE PACIENTES

```

public class PacienteF extends Fragment implements
SwipeRefreshLayout.OnRefreshListener, View.OnClickListener{
    SwipeRefreshLayout swipeContenedor;
    ListView lvListado;
    ArrayList<Paciente>listaDatos;
    PacienteAdapter adaptadorPaciente;
    ProgressDialog pDialog;
    SesionPreferences sesion;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup
container,
                                Bundle savedInstanceState) {

        View rootView = inflater.inflate(R.layout.fragment_paciente,
container, false);
sesion = new SesionPreferences(PacienteF.this.getActivity());
        /*Configura los controles*/
swipeContenedor = (SwipeRefreshLayout)
rootView.findViewById(R.id.swipe_container);
swipeContenedor.setColorScheme(android.R.color.holo_red_light,
                                android.R.color.holo_green_light,
                                android.R.color.holo_orange_light,
                                android.R.color.holo_blue_bright);

swipeContenedor.setOnRefreshListener(this);
lvListado = (ListView) rootView.findViewById(R.id.lvListado);
        /*Configura los controles*/

        //Inicializar con datos en blanco
listaDatos = new ArrayList<Paciente>();
adaptadorPaciente = new PacienteAdapter(getContext(),
listaDatos);
lvListado.setAdapter(adaptadorPaciente);
        registerForContextMenu(lvListado);
        //Inicializar con datos en blanco

        //Iniciando la descarga de datos y mostrando el
resultado

```

```

pDialog = new ProgressDialog(getContext());
pDialog.setProgressStyle(ProgressDialog.STYLE_SPINNER);
pDialog.setMessage("Cargando...");
pDialog.setCancelable(false);
pDialog.show();

new CargarListaPaciente().execute();
//Iniciando la descarga de datos y mostrando el resultado
return rootView;
}

@Override
public void onRefresh() {

if (! new Funciones().isOnline(this.getContext())){

        Snackbar snackbar = Snackbar
                .make(swipeContenedor, "Sin conexión",
Snackbar.LENGTH_LONG)
                .setAction("VOLVER A INTENTAR", new
View.OnClickListener() {
@Override
public void onClick(View view) {
                onRefresh();
            }
        });

        snackbar.setActionTextColor(Color.RED);

        View sbView = snackbar.getView();
        TextView textView = (TextView)
sbView.findViewById(android.support.design.R.id.snackbar_text);
        textView.setTextColor(Color.WHITE);

        snackbar.show();

swipeContenedor.setRefreshing(false);
return;
    }

        System.out.println("Refrescando");
new CargarListaPaciente().execute();
swipeContenedor.setRefreshing(true);

    }

@Override
public void onClick(View view) {

    }

@Override
public void onCreateContextMenu(ContextMenu menu, View v,
ContextMenu.ContextMenuInfo menuInfo) {
    /*Inicio: instanciar lista (en Values > Strings) para que
aparezca en listview*/

if(v.getId() == R.id.lvListado){
        AdapterView.AdapterContextMenuInfo info =

```

```

(AdapterView.AdapterContextMenuInfo)menuInfo;

menu.setHeaderTitle(listaDatos.get(info.position).getNombre());
String[] menuItems =
getResources().getStringArray(R.array.menuPaciente);
for(int i= 0; i<menuItems.length ; i++){
    menu.add(Menu.NONE, i, i, menuItems[i]);
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    AdapterView.AdapterContextMenuInfo info =
(AdapterView.AdapterContextMenuInfo) item.getMenuInfo();

int menuItemIndex = item.getItemId();

switch (menuItemIndex){
case 0: //Editar
break;
case 1: // eliminar
break;
case 2: //Ver citas
String dni = listaDatos.get(info.position).getDni();
Intent i = new Intent(this.getActivity(),
VerCitasActivity.class);
Bundle p = new Bundle();
p.putString("dni",dni);

p.putString("nombre",listaDatos.get(info.position).getNombre());
i.putExtras(p);
startActivity(i);

break;
}
return true;
}

public class CargarListaPaciente extends AsyncTask<Void, Void,
Integer> {
@Override
protected Integer doInBackground(Void... params) {
Integer retorno = 0;

try {
String urlListado = Funciones.URL_WS +
"pacientews.listar.php";
HashMap parametros = new HashMap<String,
String>();
parametros.put("dni", sesion.obtenerDni());
String resultado = new
Funciones().getHttpContent(urlListado,parametros);
cargarListaDatos(resultado);
retorno = 1; // Satisfactoriamente
} catch (Exception e) {
e.printStackTrace();
}

return retorno;
}

@Override
protected void onPostExecute(Integer retorno) {
if (retorno == 1) {
adaptadorPaciente.setListaDatos(listaDatos);
}
}
}

```

```

swipeContenedor.setRefreshing(false);
pDialog.dismiss();
    } else {
        Toast.makeText(PacienteF.this.getContext(), "Ha
ocurrido un error al cargar los datos de la web service",
Toast.LENGTH_SHORT).show();
    }
}
}
private void cargarListaDatos(String resultado) {
try {

        JSONObject json = new JSONObject(resultado);
        JSONArray jsonArray = json.getJSONArray("datos");

listaDatos.clear();

for (int i = 0; i < jsonArray.length(); i++) {
        JSONObject jsonData =
jsonArray.getJSONObject(i);

        Paciente item = new Paciente();

        item.setDni(jsonData.getString("dni"));
        item.setNombre(jsonData.getString("nombre"));
item.setFecha_nac(jsonData.getString("fecha_nacimiento"));
        item.setTelefono(jsonData.getString("telefono"));
        item.setEstado(jsonData.getString("estado"));
        item.setSexo(jsonData.getString("sexo"));
listaDatos.add(item);
    }
} catch (Exception e) {
    System.out.println("Error: " + e.getMessage());
}
}
@Override
public void onCreate(@Nullable Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
/*Configurar la librería para guardar en el cache*/
ImageLoaderConfiguration.Builder config = new
ImageLoaderConfiguration.Builder(this.getContext());
    config.threadPriority(Thread.NORM_PRIORITY - 2);
    config.denyCacheImageMultipleSizesInMemory();
    config.diskCacheFileNameGenerator(new
Md5FileNameGenerator());
    config.diskCacheSize(50 * 1024 * 1024); // 50 MiB
config.tasksProcessingOrder(QueueProcessingType.LIFO);
    config.writeDebugLogs(); // Remove for release app
ImageLoader.getInstance().init(config.build());

if (Build.VERSION.SDK_INT > Build.VERSION_CODES.GINGERBREAD) {
    StrictMode.ThreadPolicy policy = new
StrictMode.ThreadPolicy.Builder().permitAll().build();
    StrictMode.setThreadPolicy(policy);
}
}
}
}

```

## 6.7. CÓDIGO DE CONEXIÓN A BASE DE DATOS

### 6.7.1. PHP

```
<?php
```

```

require_once 'configuracion.php';
require_once '../util/functions/Helper.class.php';
class Conexion{
    protected $dblink;

    public function __construct() {
        $this->abrirConexion();
        //echo "conexión abierta";
    }
    public function __destruct() {
        $this->dblink = NULL;
        //echo "Conexión cerrada";
    }
    protected function abrirConexion(){
        $servidor =
"pgsql:host=".SERVIDOR_BD.";port=".PUERTO_BD.";dbname=".NOMBRE_BD;
        $usuario = USUARIO_BD;
        $clave = CLAVE_BD;

        try {
            $this->dblink = new PDO($servidor, $usuario, $clave);
            $this->dblink->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
        } catch (Exception $exc) {
            Helper::mensaje($exc->getMessage(), "e");
        }
        return $this->dblink;
    }
}

```

## 6.7.2. ANDROID

```

public class Funciones {

    public static final String URL_WS =
"http://192.168.137.1:8081/tesis2/webservice/";

    public String getHttpContent(String requestURL, HashMap<String,
String> postDataParams) {
        URL url;
        String response = "";
        try {
            url = new URL(requestURL);
            HttpURLConnection conn = (HttpURLConnection)
url.openConnection();
            conn.setReadTimeout(15000);
            conn.setConnectTimeout(15000);
            conn.setRequestMethod("POST");
            conn.setDoInput(true);
            conn.setDoOutput(true);

            OutputStream os = conn.getOutputStream();
            if (postDataParams != null){
                BufferedWriter writer = new BufferedWriter(
new OutputStreamWriter(os, "UTF-8"));
                writer.write(getPostDataString(postDataParams));
                writer.flush();
                writer.close();
            }
            os.close();
            int responseCode=conn.getResponseCode();

```

```
        System.out.println("RRRR: " + responseCode);
String line;
        BufferedReader br=new BufferedReader(new
InputStreamReader(conn.getInputStream(), "iso-8859-1"), 1024);
while ((line=br.readLine()) != null) {
            response+=line;
        }
} catch (FileNotFoundException e2){
    e2.printStackTrace();
} catch (Exception e) {
    e.printStackTrace();
}

return response;
}
```