

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 7



Análisis y Diseño del Módulo Centro Coordinador de
Emergencia Médica Nacional

Trabajo para optar por el Título de Ingeniero en Ciencias Informáticas

Autora: Yasmín Caridad Pérez Díaz

Tutores: Ing. Maidelis Pulido Morera

Ing. Luis Mariano Reyna Soler

Ciudad de La Habana, Julio 2008

“Año 50 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaro que soy la única autora de este trabajo y autorizo a la Facultad 7 de la Universidad de las Ciencias Informáticas (UCI) a hacer uso del mismo en su beneficio.

Para que así conste firmamos la presente a los 4 días del mes de Julio del año 2008.

Firma del Autor

Yasmín Caridad Pérez Díaz

Firma de la Tutora

Maidelis Pulido Morera

Firma del Tutor

Luis Mariano Reyna Soler

DATOS DE CONTACTO

Ing. Maidelys Pulido Morera: Profesor recién graduado de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas (UCI). Imparte las asignaturas Ingeniería de Software I y II en la Facultad 7.

Correo electrónico: mpulido@uci.cu

Ing. Luis Mariano Reyna Soler: Profesor graduado de Ingeniería Informática en la CUJAE en el curso 2005 – 2006. Imparte las asignaturas Introducción a la Programación y Programación I desde hace dos años en la Facultad 7. Posee la categoría docente de Instructor. Es jefe de las asignaturas Introducción a la Programación y Programación I en la Facultad 7.

Correo electrónico: lreyna@uci.cu

Dedico este trabajo:

*A mami y papi por ser tan especiales, por darme la vida y educarme como lo han hecho.
Los quiero mucho.*

A mis dos hermanitos Michel y papo, siempre los tengo presente.

*A mis abuelos, a Agui a quien quiero y le debo todo lo que soy, a mi abuelita linda Mima,
que aunque ya no este físicamente se que me guía y me da fuerzas
cada día para seguir adelante.*

A mi tío Tatico, por apoyarme siempre y quererme como a una hija, yo también te adoro.

A Liana, Sergio y Dieguito.

*A Coco y José, mis segundos padres, por ayudarme en todo momento y
orientarme el camino correcto.*

A todos mis primos, mi familia y amigos por su apoyo y confianza.

Quiero agradecer:

A mis padres, que siempre han estado junto a mí en todo momento y me han llevado por el buen camino en la vida, son los mejores padres del mundo.

*A mis hermanos y abuelos por cuidarme y velar por mí.
A Agui y Mima, esas personitas que siempre lucharon por verme hecha una profesional,
hoy su sueño se hace realidad.*

A Tatico, mi salvador, por preocuparse por mis problemas y ayudarme a resolverlos.

*A mi novio, Yoandry, por la paciencia y comprensión, por el amor y la ternura que me ha dado.
Por hacerme feliz y acompañarme en los momentos más difíciles de mi vida.*

A mis tíos y demás familiares que han ayudado en la realización de este trabajo.

A mis Tutores, por el esfuerzo y empeño para que todo saliera bien.

A Maidelis quien me apoyó muchísimo.

A todos los profes de la Facultad por la formación que han dado.

*A mis amigos de toda la vida, los de la universidad y aquellos que no por estar lejos han dejado de serlo,
dándome aliento siempre para seguir adelante.*

A todos, de corazón: Gracias.

Actualmente una de las metas del país es informatizar el Sistema Nacional de Salud (SNS), para lograr una mejor gestión de la información en las instituciones médicas, en beneficio de la sociedad. Como parte de ello se desarrolla este trabajo cuyo objetivo es modelar el diseño de una aplicación web para el Centro Coordinador de Emergencia Médica Nacional que facilite la gestión de la información en este centro.

Para dar cumplimiento al objetivo planteado se hizo una investigación sobre los sistemas similares a nivel nacional e internacional. Se utilizó como metodología de desarrollo RUP, usando UML para construir, visualizar y especificar los artefactos; generándose estos con la herramienta Visual Paradigm. Este sistema utilizará información de componentes existentes en SISalud para su funcionamiento.

El sistema para el Centro Coordinador de Emergencia Médica Nacional permitirá gestionar la información que se maneja dentro de esta institución, así como registrar y gestionar los traslados interprovinciales, controlar los móviles que viajan de una provincia a otra, así como emitir partes estadísticos sobre el trabajo en el centro.

PALABRAS CLAVES

Centro Nacional de Urgencias Médicas, Centro Coordinador de Emergencia Médica Nacional, Negocio, Requerimientos de Software, Diseño.

Índice

INTRODUCCIÓN	1
Capítulo I: Fundamentación Teórica.	6
1.1 Conceptos Básicos Relacionados con el Problema Planteado.	6
1.2 Sistemas Automatizados Vinculados al Campo de Acción.	7
1.3 Tendencias y Tecnologías Actuales.	11
1.3.1 Metodologías de Desarrollo.	11
1.3.2 UML (Lenguaje Unificado de Modelado).....	21
1.3.3 Herramientas CASE.	22
1.3.4 Arquitectura.	23
1.3.5 Lenguaje de Programación.....	25
1.3.6 Sistema Gestor de Bases de Datos (SGBD).....	25
Capítulo II: Características del Sistema.	27
2.1 Flujo Actual de los Procesos Involucrados en el Campo de Acción.	27
2.2 Objeto de Automatización.....	28
2.3 Modelado del Negocio.	29
2.4 Propuesta de Sistema.	41
2.4.1 Especificación de Requerimientos de Software.	42
2.4.2 Dependencias y Relaciones con otros Sistemas.	48
2.5 Definición de los Casos de Uso.	48
2.5.1 Organización de los Elementos: Casos de Uso en Paquetes.	49
2.5.2 Descripción Textual de los Casos de Uso.....	56
Capítulo III: Diseño del Sistema.	67
3.1 Modelo de Diseño.....	67
3.1.1 Patrones.	67

ÍNDICE

3.1.2 Diagramas de Clases del Diseño.....	69
3.1.3 Diagramas de Interacción.....	80
CONCLUSIONES.....	91
RECOMENDACIONES	92
REFERENCIAS BIBLIOGRÁFICAS	93
BIBLIOGRAFÍA	95
ANEXOS	98
Anexo 1: Diagrama de Actividades.	98
Anexo 2: Modelo Pacientes para Traslado a Provincia.	104
GLOSARIO DE TÉRMINOS	105

INTRODUCCIÓN.

La creación del Sistema Nacional de Salud ha sido uno de los avances más novedosos desde los inicios de la atención médica especializada en Cuba. Se clasifica según los niveles de atención a la población en: nivel de Atención Primaria¹, nivel de Atención Secundaria² y el nivel de Atención Terciaria³. El Ministerio de Salud Pública (MINSAP), es el órgano rector del sistema Nacional de Salud; encargado de dirigir, ejecutar y controlar la aplicación de la política del Estado y del Gobierno en cuanto a la Salud Pública, el desarrollo de las Ciencias Médicas y la Industria Médico Farmacéutica. **(MINSAP, 2008)**

La medicina cubana ha obtenido durante varias décadas resultados importantes mediante un arduo trabajo, lo que le ha reportado gran prestigio a nivel nacional e internacional. Para mantenerlo y garantizar la excelencia de los servicios brindados en el área de salud, es necesario hacer los cambios pertinentes que permitan la transformación tecnológica, donde el empleo de la informática juega un papel determinante.

Las computadoras, han concedido a los profesionales de la medicina la posibilidad de mejorar su trabajo, en cuanto a eficiencia y rapidez. Son utilizadas, con el objetivo de apoyar la investigación científica, la docencia médica, la asistencia a enfermos, tratamientos y el procesamiento de estadísticas. La combinación de las telecomunicaciones y las técnicas para el procesamiento de datos permiten que las actividades médicas maximicen sus logros.

El gobierno cubano tiene la meta de informatizar varias de sus esferas, entre ellas el Sistema Nacional de Salud. Esto permitirá mejorar la gestión de sus actividades, así como la atención al paciente, el control y la administración de la información, logrando beneficiar en gran medida a la población. De manera general, las instituciones del país deberán alcanzar un elevado nivel de informatización, de

¹ Representa el primer nivel de contacto de la familia y la comunidad con el Sistema Nacional de Salud. Se refiere a consultorios, policlínicos.

² Se refiere a los servicios prestados en hospitales generales, pediátricos y ginecobstétricos.

³ Se refiere a la atención médica prestada en determinados centros por la atención especializada que se requiere, como Institutos de Cirugía Cardiovascular, Neurocirugía, Gastroenterología, entre otros centros de investigación categorizados de referencia nacional y en algunos casos de referencia internacional.

manera que haya un incremento de la calidad de los servicios y una mayor satisfacción del personal que allí labora.

Como apoyo a este proceso de informatización, la Universidad de las Ciencias Informáticas y dentro de ella la Facultad # 7 ha puesto en marcha una serie de proyectos productivos apoyando el uso del software libre para la realización y desarrollo de productos informáticos para la salud.

El Sistema Integrado de Urgencias Médicas (SIUM) surge en 1997 como parte del Sistema Nacional de Salud, a partir de aquí comenzó un proceso de reordenamiento progresivo y lógico de los sistemas de urgencia en el país. **(DTCUBA, 2005)**

Su misión es organizar las urgencias y emergencias médicas desde Consultorios, Policlínicos, Terapias Hospitalarias y Centros de Coordinación de Ambulancia Provinciales y el Centro Coordinador de Emergencia Médica Nacional. En la siguiente figura se muestra como parte de la estructura organizativa del SIUM una pirámide con varios subsistemas de atención al paciente, desde la base del Sistema Nacional de Salud hasta el último escalón en la atención al ciudadano grave.



Estructura organizativa del SIUM.

El Centro Coordinador de Emergencia Médica Nacional (CCEMN) ha decidido realizar transformaciones en sus servicios en cuanto a infraestructura y tecnología lo que permitirá brindar una respuesta más rápida y certera en la atención al paciente grave. Actualmente este centro realiza la coordinación y expedición de móviles, el traslado de pacientes a instituciones y hospitales; atiende eventos y actividades de primer nivel y situaciones de desastres y contingencias.

Estos procesos son solicitados a través de una llamada telefónica, que es recogida y gestionada de manera manual, lo que puede ocasionar deficiencias en los servicios y poco control de los recursos que disponen. Además los partes estadísticos no reflejan toda la información necesaria y existen deficiencias para generar informes de balances mensuales o anuales, ya que se dificulta el manejo de los registros diarios que son anotados en planillas, y en muchos casos pueden deteriorarse al manipularlos con frecuencia. Además puede ocurrir la pérdida de demandas, demoras en la ejecución de estas, quejas de la población por incumplimiento de recogidas de pacientes y falta de control de los móviles que viajan sin la debida autorización del Centro Coordinador de Emergencia Médica Nacional.

Por otra parte, el déficit de herramientas para el control y almacenamiento de información y datos importantes influye en la toma de decisiones y provoca resultados que en muchas ocasiones no reflejan la realidad.

Considerando lo analizado anteriormente sobre la situación actual del Centro Coordinador de Emergencia Médica Nacional se plantea como **problema científico** de la investigación ¿Cómo facilitar la gestión de la información de las emergencias médicas en el Centro Coordinador de Emergencia Médica Nacional?

El **objeto de estudio** de la investigación es: El proceso de gestión de la información en el Centro Nacional de Urgencias Médicas (SIUM) donde el **campo de acción** se enmarca específicamente en el proceso de gestión de la información del Centro Coordinador de Emergencia Médica Nacional.

Como **objetivo general** para darle solución al problema planteado se propone diseñar una aplicación Web que facilite la gestión de la información de las emergencias médicas en el Centro Coordinador de Emergencia Médica Nacional.

Para dar cumplimiento al objetivo planteado deben realizarse las siguientes **tareas**:

1. Analizar los sistemas informáticos existentes en Cuba y el mundo, que permitan la gestión de las emergencias médicas.

2. Valorar tecnologías, metodologías y herramientas de desarrollo de software utilizadas en la actualidad.
3. Valorar la selección de la metodología, las herramientas y el lenguaje de modelado a utilizar.
4. Realizar entrevistas a los especialistas del proceso de gestión de las emergencias médicas en el Centro Coordinador de Emergencia Médica Nacional.
5. Modelar las condiciones actuales que rigen el proceso de las emergencias médicas realizadas en el Centro Coordinador de Emergencia Médica Nacional.
6. Analizar las necesidades de funcionamiento del sistema informático, describiendo la Especificación de Requisitos del Software.
7. Analizar la propuesta de arquitectura definida por el MINSAP (Orientada a Servicios y Basada en Componentes) con la cual diseñar el Módulo Centro Coordinador de Emergencia Médica Nacional.
8. Valorar la integración con otros componentes ya existentes en el SISalud.
9. Obtener el Modelo de Sistema del Módulo Centro Coordinador de Emergencia Médica Nacional.
10. Definir los patrones de diseño a utilizar.
11. Obtener el Modelo de Diseño del Módulo Centro Coordinador de Emergencia Médica Nacional.

Como **resultados esperados** se desarrollará el diseño de un software para el Centro Coordinador de Emergencia Médica Nacional (CCEMN), obteniendo los modelos correspondientes en cada flujo de trabajo por los que se transitó, permitiendo realizar la implementación del mismo a partir de dichos modelos. De esta forma se dotará al Centro Coordinador de Emergencia Médica Nacional de un sistema informático que gestione los procesos que en él se realizan.

El documento se encuentra estructurado en 3 capítulos, que incluyen la investigación realizada de tecnologías y sistemas similares existentes en Cuba y el mundo, el modelado del sistema propuesto, así como el diseño del mismo. Cuenta además con un glosario de términos en el que se describen conceptos utilizados durante el desarrollo de la investigación.

El documento está distribuido de la siguiente forma:

En el Capítulo 1 se refleja la investigación de conceptos relacionados con el campo de acción del trabajo, abarcando dentro de estos los referentes al proceso de gestión de las emergencias y urgencias médicas en el Centro Coordinador de Emergencia Médica Nacional. Además, se realiza una

investigación acerca de las tecnologías y metodologías utilizadas actualmente en el desarrollo de software, donde se describen sus características, así como la propuesta a utilizar para la realización del sistema.

En el Capítulo 2 se muestran de manera detallada los aspectos relacionados a los procesos del negocio, además de la propuesta de sistema. En este capítulo se generan los artefactos que componen los flujos de trabajo de RUP; Modelado del Negocio y Gestión de Requerimientos, reflejando además las reglas del negocio, los diagramas de actividades, descripciones de casos de uso y diagramas, así como el listado de requerimientos funcionales y no funcionales.

En el Capítulo 3 se encuentran los aspectos relacionados con el Diseño de la aplicación, mostrándose los artefactos de este flujo de trabajo; diagramas de clases del diseño y diagramas de secuencia.

Capítulo I: Fundamentación Teórica.

En este capítulo se analizan los conceptos más importantes asociados al contexto del problema, se expone el resultado de búsquedas y análisis realizados vinculados a los procesos de gestión de información en el Centro Coordinador de Emergencia Médica Nacional, sistemas existentes a nivel nacional e internacional así como el estado del arte de técnicas, metodologías de desarrollo de software, tendencias y tecnologías usadas en la actualidad, justificando de igual forma las utilizadas en la investigación.

La información obtenida en este análisis ha sido consultada y recopilada a través de revisiones bibliográficas, entrevistas y otras fuentes de información.

1.1 Conceptos Básicos Relacionados con el Problema Planteado.

A continuación se muestran algunos conceptos importantes relacionados con el problema planteado, los cuales permiten una mejor comprensión de los aspectos a tratar en el trabajo.

Demanda: Solicitud de atención médica a través del servicio de emergencia móvil.

Móvil ó Ambulancia: Vehículo para el transporte de heridos y enfermos. Hospital ambulante.

Emergencia médica: Es una situación de peligro o desastre que requiere una acción inmediata. **(Villatoro, 2005)**

Emergencia móvil: Es toda la red de vehículos destinados a socorrer en caso de emergencia, como ambulancias o cualquier otro tipo de transporte sanitario.

Prioridades en las emergencias médicas:

- Primera prioridad: Urgencias con afectación vital o peligro vital potencial que generan un peligro vital inmediato. Requieren de atención inmediata.
- Segunda prioridad: Urgencia verdadera con requerimientos hospitalarios. Paciente con enfermedad o lesiones que de no atenderse en un tiempo mediano puede generar complicaciones fatales.

- Tercera prioridad: Urgencia banal o sin riesgos. Pacientes que requieren ser vistos de urgencia pero sin peligro vital mediato o tardío.

Test diagnósticos: Término utilizado para referirse a todo tipo de pruebas cuyo objetivo es efectuar un diagnóstico. Pueden ser de laboratorio (hematológicas, químicas, serológicas, etc.), radiográficas (rayos X, ultrasonidos, tomografías, resonancias magnéticas, etc.).

1.2 Sistemas Automatizados Vinculados al Campo de Acción.

Con el fin de aumentar la capacidad y eficiencia en la atención a la población, tanto a nivel nacional como internacional, se han realizado numerosos productos informáticos. Actualmente Cuba cuenta con varios productos instalados en instituciones de salud; en bancos de sangre y hospitales generalmente, de ellos, no existe ninguno para la gestión de las emergencias médicas en el Centro Coordinador de Emergencia Médica Nacional.

A continuación se muestran algunos de los sistemas vinculados al campo de acción de este trabajo.

GEMCIA - Sistema de Gestión de Emergencias y Urgencias Médicas.

Es un sistema encargado de la gestión para dar soporte operativo y de toma de decisiones a todos los procesos y tareas que intervienen en los servicios de emergencias y urgencias médicas. Su objetivo principal es cubrir los tres procesos básicos de las empresas de emergencias y urgencias médicas.

(StartNet, S/A)

- EL Proceso Operativo Funcional.
- El Proceso Administrativo/Contable.
- El Proceso de Control y Toma de decisiones.

La herramienta cuenta con varios módulos, desde la recepción de la llamada, despacho del móvil específico, control de los Stock de medicamentos e insumos, tiempo de ocupación de móviles, así como módulos Administrativos/Contables que hacen posible llevar adelante toda la gestión de padrones, convenios, facturación general mensual, cuenta corriente de afiliados, proveedores, etc.

Algunos módulos del sistema

Módulo de *Dispatcher*:

- Recepción.
- Despacho de móviles.
- Manejo de turnos de personal.
- Préstamos de nebulizadores.
- Emisión de presupuestos y facturación.
- Emisión de cronograma para la enfermería a domicilio.
- Emisión de cronograma para traslados urbanos, interurbanos y eventos.

Módulo de Manejo de Turnos de Personal:

- Alta, baja y modificación de turnos del personal.

Módulo de Farmacia:

- Alta, baja y modificación de medicamento e insumos.
- Stock de medicamento.
- Stock de insumos.
- Stock de medicamento e insumos por móvil, equipo y enfermería en base.

Módulo Administrativo/Contable.

Administrativo – Clientes.

- Cuenta corriente de socios.
- Cobro de cupones (amortización en cuenta corriente, anulación, baja etc.).
- Listados varios de facturas de socios.
- Gestión de cobradores.
- Listados de deudores por períodos y cobradores.
- Facturación a clientes (empresas y consumidores finales).
- Gestión de cobros de deudores.

Valoración del Sistema

El Sistema de Gestión de Emergencias y Urgencias Médicas (GEMCIA) ha sido desarrollado por la empresa *StartNet* ubicada en Córdoba, Argentina. Esta se dedica al desarrollo de productos

informáticos ofreciendo una amplia gama de ellos, cubriendo así las necesidades de las diferentes empresas que solicitan servicios.

Esta es una empresa privada la cual desarrolla productos propietarios, muy costosos, de difícil alcance por numerosos países del mundo, entre los que se encuentra Cuba. Por otra parte el sistema GEMCIA no resuelve los problemas existentes en el Centro Coordinador de Emergencia Médica Nacional, pues no se adapta a las características del sistema de salud cubano. Además ofrece servicios de facturación, cuenta corriente de afiliados, entre otros, que no se corresponden con la estructura del sistema de salud cubano.

SICUM - Sistema Informático para el Control de la Urgencia Médica.

Sistema para controlar de forma automatizada la atención al paciente dentro de la red Nacional del Sistema Integrado de Urgencias Médicas (SIUM). Este forma parte del sistema para la atención primaria a la salud. **(Pompa Sourd, 2001)**

El sistema brinda información acerca de:

- Solicitudes o demandas de rescate de un paciente.
- Información sobre las ambulancias y su recorrido.
- Características del personal autorizado para viajar.
- Características del paciente.
- Estado al comienzo y fin del traslado.
- Problemas de salud que presenta.
- Accionar del personal de rescate.
- Parámetros vitales del paciente.

Ofrece además servicios para la atención en consultorios de urgencias, policlínicos, así como características de la consulta, diagnóstico y conducta médica para cada diagnóstico y seguimiento del estado del paciente durante el ingreso en la sala de observaciones (medicamentos suministrados, análisis de laboratorio). Ofrece reportes estadísticos para medir el funcionamiento de la atención primaria, estadísticas que muestran el desempeño y efectividad de la red del SIUM como sistema.

Es una aplicación de escritorio desarrollada en el lenguaje de programación *Borland Delphi* y que utiliza como gestor de base de datos SQL server.

Valoración del Sistema

SICUM es uno de los sistemas existentes en Cuba, desarrollado en el año 2001 y encargado de controlar las urgencias médicas. Este sistema no cumple con las expectativas actuales del Centro Nacional de Urgencias Médicas (SIUM) y dentro de este del Centro Coordinador de Emergencia Médica Nacional pues a lo largo de todos estos años desde la creación de dicha aplicación, el SIUM ha sufrido numerosos cambios en cuanto a infraestructura y organización de varias de sus esferas, es por ello que los requerimientos actuales de dicha institución no son acordes a los desarrollados en el Sistema de Control de Urgencias Médicas en el año 2001.

Este sistema no cumple con la propuesta de integración con los componentes existentes en el Sistema de Información para la Salud, el cual es un requisito fundamental para la producción de software de esta rama en la actualidad, ya que ofrece numerosos servicios de gran importancia.

Registros Estadísticos Informáticos del Sistema Integrado de Urgencias Médicas.

En la provincia de Las Tunas fue desarrollado un producto informático implementado con el lenguaje de programación *Borland Delphi* para llevar el control estadístico de los datos referentes a todos los pacientes atendidos en el Sistema Integrado de Urgencias Médicas en la provincia, el cual permite realizar estudios periódicos acerca de incidencias en las urgencias y las emergencias del territorio e integrar los niveles de atención primaria y secundaria de salud. **(Rojas, 2001)**

Valoración del Sistema

Este sistema no resuelve el problema planteado en este trabajo, solo es una vía para solucionar algunas de las deficiencias estadísticas en la provincia de Las Tunas. El sistema descrito anteriormente no gestiona las demandas y no controla los móviles del centro, factor esencial para el buen funcionamiento de la institución.

Actualmente el Centro Coordinador de Emergencia Médica Nacional requiere de un sistema que le gestione toda la información referente a las urgencias y emergencias médicas, incluyendo el control de los móviles, además de permitir emitir estadísticas acerca del trabajo diario, mensual o anual y este

sistema no ofrece ese tipo de funcionalidades por lo que no es conveniente el uso del mismo para darle solución al problema existente.

1.3 Tendencias y Tecnologías Actuales.

En esta sección se muestran algunas de las tendencias y tecnologías actuales para el desarrollo de aplicaciones informáticas. Dentro de las que se puede mencionar las metodologías de desarrollo, incluyendo en estas las llamadas metodologías ágiles, además de otras clasificaciones, así como el lenguaje de modelado y algunas herramientas CASE.

1.3.1 Metodologías de Desarrollo.

Una Metodología de Desarrollo de Software es un conjunto de procedimientos, técnicas, herramientas y un soporte de documentos que ayuda a los desarrolladores a realizar un nuevo software. Una metodología puede seguir uno o varios modelos de ciclo de vida, esta indica cómo hay que obtener los distintos productos parciales y finales en el desarrollo de un software. **(García, S/A)**

El resultado final de un proyecto de software es un producto donde intervienen las personas a través de un proceso de desarrollo que guía los esfuerzos de estas, a modo de plantilla, explicando los pasos necesarios para terminar el proyecto.

No existe una metodología universal para hacer frente con éxito a un proyecto de desarrollo de software. Toda metodología debe ser adaptada al contexto del proyecto (recursos técnicos y humanos, tiempo de desarrollo y tipo de sistema).

Lo más importante antes de elegir la metodología que se usará para la implementación de un software, es determinar el alcance que tendrá el mismo y ver cual es la más conveniente para la aplicación.

Criterios de Selección de Metodologías de Desarrollo de Software.

Debe ajustarse a los objetivos: Cada aproximación al desarrollo de software está basada en unos objetivos. Por ello la metodología que se elija debe recoger el aspecto filosófico de la aproximación deseada, es decir que los objetivos generales del desarrollo deben estar implementados en la metodología de desarrollo. **(Creative Commons, 2007)**

Debe cubrir el ciclo entero de desarrollo del software: Para ello la metodología debe cubrir todas las etapas de desarrollo del software.

Debe integrar las distintas fases del ciclo de desarrollo.

Debe incluir la realización de validaciones: La metodología debe detectar y corregir los errores cuanto antes. Uno de los problemas más frecuentes y costosos es el aplazamiento de la detección y corrección de problemas en las etapas finales del proyecto. Cuanto más tarde sea detectado el error más caro será corregirlo.

Debe especificar claramente los responsables de los resultados: Debe especificar claramente quienes son los participantes de cada tarea a desarrollar, debe detallar de una manera clara los resultados de los que serán responsables.

Debe poder enseñarse: Cada integrante del equipo de desarrollo debe entender las técnicas específicas de la metodología, los procedimientos organizativos y de gestión que la hacen efectiva, las herramientas automatizadas que soportan la metodología y las motivaciones que subyacen en ella.

Debe estar soportada por herramientas CASE⁴: La metodología debe estar soportada por herramientas automatizadas que mejoren la productividad, tanto del ingeniero de software en particular, como la del desarrollo en general. El uso de estas herramientas reduce el número de personas requeridas y la sobrecarga de comunicación, además de ayudar a producir especificaciones y diseños con menos errores, más fáciles de probar, modificar y usar.

Metodologías Ágiles.

Las metodologías ágiles son estrategias de desarrollo de software que promueven prácticas que son adaptativas en vez de predictivas, centradas en las personas o en los equipos, iterativas, orientadas hacia prestaciones y hacia la entrega, de comunicación intensiva y que requieren que el cliente se involucre de forma directa. **(Montejano, 2006)**

Lo fundamental en este tipo de metodología es:

- La comunicación entre los usuarios y los desarrolladores.

⁴ CASE: Computer Aided Software Engineering (Ingeniería de Software Asistida por Computadora).

- La simplicidad al desarrollar y codificar los módulos del sistema.
- La retroalimentación concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales.

Entre las metodologías ágiles más usadas en la actualidad se pueden nombrar:

- *XP eXtreme Programming.*
- *Scrum.*
- *Feature Driven Development.*
- *Adaptive Software Development.*

A continuación se dará un breve resumen de algunas de estas metodologías para tener conocimiento general de cada una de ellas.

XP eXtreme Programming.

Es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizada para proyectos de corto plazo y equipo pequeño. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. **(Blanqué, 2008)** Se caracteriza por:

Pruebas unitarias: Se basa en las pruebas realizadas a los principales procesos, es decir, consiste en hacer pruebas de las fallas que pudieran ocurrir. Es como adelantarse a obtener los posibles errores.

Re-fabricación: Se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.

Programación en pares: Una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo.

La metodología XP propone:

- Empezar poco a poco e ir añadiendo funcionalidades con retroalimentación continua.
- Que el manejo del cambio se convierta en parte sustantiva del proceso.

- Que el costo del cambio no dependa de la fase o etapa.
- No introducir funcionalidades antes de ser necesario.
- Que el cliente o el usuario se convierta en miembro del equipo.

Feature Driven Development.

Esta metodología ágil de desarrollo de software define un proceso iterativo que consta de 5 pasos. Las iteraciones son cortas con duración hasta de dos semanas, centrándose, fundamentalmente, en las fases de diseño e implementación del sistema partiendo de una lista de características que debe reunir el software. **(H. Canós, et al., S/A)**

Adaptive Software Development.

Las características principales de esta metodología son: iterativo, orientado a los componentes de software más que a las tareas y tolerante a los cambios. Propone un ciclo de vida de tres fases: especulación, colaboración y aprendizaje. En la primera de ellas se inicia el proyecto y se planifican las características del software; en la segunda se desarrollan las características y finalmente en la tercera se revisa su calidad y se entrega al cliente. La revisión de los componentes sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo. **(Behr, 2008)**

Scrum.

Esta metodología define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Está especialmente indicada para proyectos con un rápido cambio de requisitos. Posee dos características fundamentales, estas son: desarrollo de software mediante iteraciones, denominadas *sprints* (el resultado de cada *sprint* es un incremento ejecutable que se muestra al cliente), con una duración de 30 días y reuniones a lo largo del proyecto, destacando la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración. **(Mousqués, 2004)**

No es una metodología de análisis, ni de diseño, como podría ser RUP, es una metodología de gestión del trabajo. Por otra parte SCRUM puede ser aplicado a distintos modelos de calidad.

Ciclo de vida de *Scrum*.

- Pre-juego, Planeamiento: El propósito es establecer la visión, definir expectativas y asegurarse la financiación. Las actividades son la escritura de la visión, el presupuesto, el registro de

acumulación o retraso del producto inicial y los ítems estimados, así como la arquitectura de alto nivel, el diseño exploratorio y los prototipos.

- Pre-juego, Montaje: El propósito es identificar más requerimientos y priorizar las tareas para la primera iteración. Las actividades son planificación, diseño exploratorio y prototipos.
- Juego, Desarrollo: El propósito es implementar un sistema listo para entrega en una serie de iteraciones. Las actividades son un encuentro de planeamiento de carreras cortas en cada iteración, la definición del registro de acumulación de carreras cortas y los estimados, y encuentros diarios de *Scrum*.
- Pos-juego, Liberación. El propósito es el despliegue operacional. Las actividades son documentación, entrenamiento, mercadeo y venta.

Otras Metodologías de Desarrollo de Software.

Microsoft Solution Framework (MSF).

Esta es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas. (**Microsoft Solutions Framework, 2007**)

Características de MSF.

- Escalable: Puede organizar equipos tan pequeños como de 3 o 4 personas, así como también, proyectos que requieren 50 personas o más.
- Flexible: Es utilizada en el ambiente de desarrollo de cualquier cliente.
- Tecnología agnóstica: Puede ser usada para desarrollar soluciones basadas sobre cualquier tecnología.

MSF se compone de varios modelos encargados de planificar las diferentes partes implicadas en el desarrollo de un proyecto: Modelo de Arquitectura del Proyecto, Modelo de Equipo, Modelo de Proceso, Modelo de Gestión del Riesgo, Modelo de Diseño de Proceso y finalmente el Modelo de

Aplicación.

- **Modelo de Arquitectura del Proyecto:** Diseñado para acortar la planificación del ciclo de vida. Este modelo define las pautas para construir proyectos empresariales a través del lanzamiento de versiones.
- **Modelo de Equipo:** Este modelo ha sido diseñado para mejorar el rendimiento del equipo de desarrollo. Proporciona una estructura flexible para organizar los equipos de un proyecto. Puede ser escalado dependiendo del tamaño del proyecto y del equipo de personas disponibles.
- **Modelo de Proceso:** Diseñado para mejorar el control del proyecto, minimizando el riesgo, y aumentar la calidad acortando el tiempo de entrega. Proporciona una estructura de pautas a seguir en el ciclo de vida del proyecto, describiendo las fases, las actividades, la liberación de versiones y explicando su relación con el Modelo de Equipo.
- **Modelo de Gestión del Riesgo:** Diseñado para ayudar al equipo a identificar las prioridades, tomar las decisiones estratégicas correctas y controlar las emergencias que puedan surgir. Este modelo proporciona un entorno estructurado para la toma de decisiones y acciones valorando los riesgos que puedan provocar.
- **Modelo de Diseño del Proceso:** Diseñado para distinguir entre los objetivos empresariales y las necesidades del usuario. Proporciona un modelo centrado en el usuario para obtener un diseño eficiente y flexible a través de un enfoque iterativo. Las fases de diseño conceptual, lógico y físico proveen tres perspectivas diferentes para los tres tipos de roles: los usuarios, el equipo y los desarrolladores.
- **Modelo de Aplicación:** Diseñado para mejorar el desarrollo, el mantenimiento y el soporte, proporciona un modelo de tres niveles para diseñar y desarrollar aplicaciones software. Los servicios utilizados en este modelo son escalables, y pueden ser usados en un solo ordenador o incluso en varios servidores.

Rational Unified Process (RUP).

RUP es una metodología de desarrollo de software, que pretende implementar las mejores prácticas en ingeniería de software, con el objetivo de asegurar la producción de software de calidad, dentro de plazos y presupuestos predecibles. La metodología RUP es más adaptable para proyectos de largo plazo. Define QUIÉN debe hacer QUÉ, CUÁNDO y CÓMO debe hacerlo. **(Gomez, 2007)**

Esta divide el proceso de desarrollo en ciclos, donde se obtiene un producto al final de cada uno de ellos. Estos ciclos se dividen en 4 fases, cada fase concluye con un hito bien definido donde deben tomarse ciertas decisiones.

Las fases de esta metodología son las siguientes:

- Inicio: El objetivo en esta etapa es determinar la visión del proyecto, los objetivos del software, se describe el negocio y se delimita el proyecto describiendo su alcance con la identificación de los casos de uso del sistema.
- Elaboración: En esta etapa el objetivo fundamental es definir la arquitectura del sistema. Ya en esta fase se obtiene una aplicación ejecutable que responde a los casos de uso más significativos arquitectónicamente.
- Construcción: En esta fase el objetivo es llegar a obtener la funcionalidad operativa del producto, es decir un producto listo para su utilización que está documentado, tiene un manual de usuario y ha pasado las pruebas.
- Transición: El objetivo es llegar a obtener un *release* del producto listo para su instalación en las condiciones reales. Esta etapa puede implicar también reparación de errores.

Las principales características de esta metodología son:

- Dirigido por casos de uso: Los casos de uso reflejan lo que el cliente necesita, lo cual se capta al modelar el negocio y se representa a través de los requerimientos. Luego los casos de uso guían el proceso de desarrollo, ya que como resultado de los diferentes flujos de trabajo, los modelos que se obtienen representan la realización de los casos de uso, es decir, estos se van haciendo por cada caso de uso.

- Centrado en la arquitectura: La arquitectura muestra la visión común del sistema. RUP se desarrolla mediante iteraciones, comenzando por los casos de uso relevantes desde el punto de vista de la arquitectura.
- Iterativo e incremental: Cada fase se desarrolla en iteraciones, una iteración implica actividades de todos los flujos de trabajo, aunque de algunos más que de otros. RUP propone dividir el trabajo en partes más pequeñas o mini proyectos, donde cada uno de ellos es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en los flujos de trabajo, y los incrementos, al crecimiento del producto.

El proceso unificado de desarrollo (RUP) puede describirse en dos dimensiones (Figura 1.1).

Horizontalmente se representa el tiempo y muestra el aspecto dinámico del proceso, expresado en términos de ciclos, fases, iteraciones, y metas.

Verticalmente se representa el aspecto estático del proceso; como está descrito en términos de actividades, artefactos, trabajadores y flujos de trabajo.

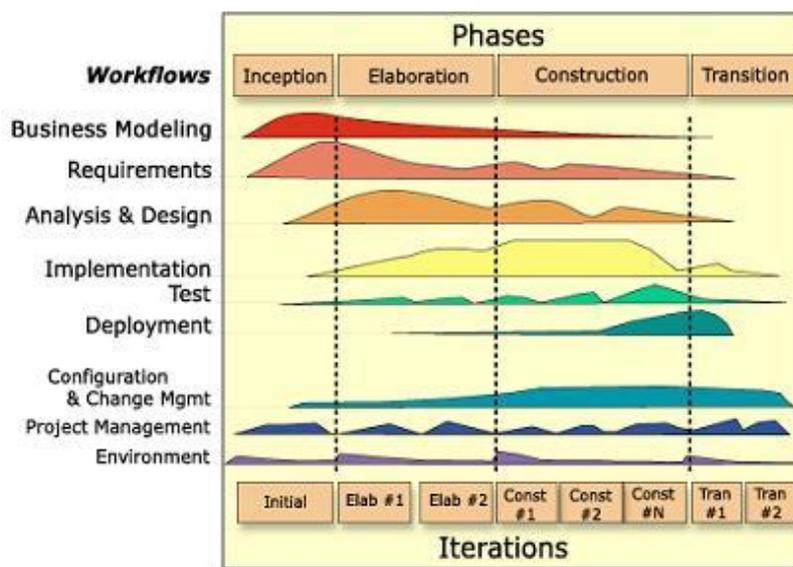


Figura 1. 1 Proceso Unificado de Desarrollo.

En RUP se han agrupado las actividades en grupos lógicos definiéndose 9 flujos de trabajo principales. Los 6 primeros son conocidos como flujos de ingeniería y los tres últimos como de apoyo.

Modelado del Negocio: En este flujo de trabajo se llega a un mejor entendimiento de la organización donde se va a implantar el producto, es decir las necesidades del negocio. Sus objetivos son:

- Comprender la estructura y la dinámica de la organización en la cual se va a implantar un sistema.
- Comprender los problemas actuales de la organización e identificar las mejoras potenciales.
- Asegurar que los consumidores, usuarios finales y desarrolladores tengan un entendimiento común de la organización.
- Derivar los requerimientos del sistema que va a soportar la organización.

Gestión de Requerimientos: Este es uno de los flujos de trabajo más importantes, porque en él se establece qué tiene que hacer el sistema que se va a construir, es decir, se trasladan las necesidades del negocio a un sistema a automatizar. Sus objetivos son:

- Establecer y mantener un acuerdo entre clientes y otros *stakeholders* sobre lo que el sistema podría hacer.
- Proveer a los desarrolladores un mejor entendimiento de los requisitos del sistema.
- Definir el ámbito del sistema.
- Proveer una base para la planeación de los contenidos técnicos de las iteraciones.
- Proveer una base para estimar costos y tiempo de desarrollo del sistema.
- Definir una interfaz de usuarios para el sistema, enfocada a las necesidades y metas del usuario.

Análisis y Diseño: En este flujo de trabajo se traducen los requisitos a una especificación que describe cómo implementar el sistema. Este flujo tiene como objetivo:

- Transformar los requisitos al diseño del futuro sistema.
- Desarrollar una arquitectura para el sistema.
- Adaptar el diseño para que sea consistente con el entorno de implementación, diseñando para el rendimiento.

Implementación: En este flujo de trabajo se implementan las clases y objetos en los diferentes ficheros. Además cada programador es responsable de probar las unidades que produzca. El resultado final de este flujo de trabajo es un sistema ejecutable. Tiene como objetivo:

- Definir la organización del sistema en términos de subsistemas de implementación organizados en capas.
- Implementar los elementos de diseño en términos de elementos de implementación (ficheros fuentes, binarios, ejecutables y otros).
- Probar los componentes desarrollados independientemente como unidades.
- Integrar los resultados producidos por desarrolladores independientes o equipos en un sistema ejecutable.

Pruebas: Este flujo de trabajo es el encargado de la evaluación del software que se está desarrollando, se debe aplicar en todo el ciclo de vida, para que cuando el producto esté terminado no se le encuentren errores que impliquen comenzar todo de nuevo. Tiene como objetivo:

- Encontrar y documentar defectos en la calidad del software.
- Asesorar la calidad del software percibida.
- Provee la validación de los supuestos realizados en el diseño y especificación de requisitos por medio de demostraciones concretas.
- Verificar las funciones del producto de software según lo diseñado.
- Verificar que los requisitos tengan su apropiada implementación.

Despliegue: Este flujo de trabajo es el encargado de la distribución e instalación del producto.

Flujos de control de apoyo:

Gestión de cambio y configuración: Este flujo de trabajo es el encargado de mantener la integridad de todos los artefactos que se crean en el proceso, así como de mantener información del proceso evolutivo que han seguido y el control de los cambios que puedan ocurrir.

Gestión de proyecto: Es el flujo encargado de gestionar objetivos, riesgos, horarios, recursos y restricciones para desarrollar un producto que sea acorde a los requisitos de los clientes y los usuarios.

Entorno de desarrollo: Este flujo de trabajo es el encargado de dar soporte al proyecto con las adecuadas herramientas, procesos y métodos. Brinda una especificación de las herramientas que se van a necesitar en cada momento, así como define la instancia concreta del proceso a seguir.

El Proceso Unificado de desarrollo está basado en componentes y utiliza el lenguaje unificado de modelado (UML) para preparar todos los esquemas de un sistema de software. Este es una parte esencial de RUP.

1.3.2 UML (Lenguaje Unificado de Modelado).

UML es un lenguaje para especificar, construir, visualizar y documentar los artefactos (información que se utiliza o produce mediante un proceso de software). Este lenguaje de modelado no es una guía para realizar el análisis y diseño orientado a objeto, es decir, no es un proceso, es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos. **(Hernández, S/A)**

El modelado visual permite manejar la complejidad de los sistemas a analizar o diseñar. Un modelo es expresado en un lenguaje de modelado compuesto por vistas, elementos de modelos y un conjunto de mecanismos generales o reglas que indican como utilizar los elementos. Las vistas muestran diferentes aspectos del sistema modelado. Una vista es una abstracción que se puede definir como una fotografía completa del sistema. Los diagramas son la representación gráfica de un conjunto de elementos describiendo el contenido de una vista y visualizan un sistema desde diferentes perspectivas.

UML intenta solucionar el problema de propiedad de código que ocurre con muchos desarrolladores. Al implementar un lenguaje de modelado común para todos los desarrollos se crea una documentación común de igual manera, que cualquier desarrollador con conocimientos de UML será capaz de entender, independientemente del lenguaje utilizado para este. **(García, 2005)**

Su utilización es independiente del lenguaje de programación y de las características de los proyectos, ya que UML ha sido diseñado para modelar cualquier tipo de proyectos, tanto informáticos como de arquitectura, o de cualquier otro ramo.

De forma general las principales características de UML son:

- Lenguaje unificado para la modelación de sistemas.

- Tecnología orientada a objetos.
- El cliente participa en todas las etapas del proyecto.
- Corrección de errores viables en todas las etapas.
- Aplicable para tratar asuntos de escala inherentes a sistemas complejos de misión crítica, tiempo real y cliente/servidor.

Ventajas de UML

- Mayor rigor en la especificación.
- Permite realizar una verificación y validación del modelo realizado.
- Se pueden automatizar determinados procesos y permite generar código a partir de los modelos y a la inversa (a partir del código fuente generar los modelos). Esto permite que el modelo y el código estén actualizados, con lo que siempre se puede mantener la visión en el diseño, de más alto nivel, de la estructura de un proyecto.

1.3.3 Herramientas CASE.

Las herramientas *CASE* (*Computer Aided Software Engineering*, Ingeniería de Software Asistida por Computadoras) son un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del ciclo de desarrollo de un software. El uso de estas herramientas mejora la calidad de las aplicaciones desarrolladas y se aumenta de igual manera la productividad, reduciendo el coste de las mismas. Estas herramientas permiten además a los analistas tener más tiempo para el análisis y diseño y minimizar el tiempo para codificar y probar.

(StartMedia, S/A)

Rational Rose.

Rational Rose es una herramienta para el modelado visual mediante UML de sistemas de software, con plataforma independiente que ayuda a la comunicación entre los miembros del equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Permite especificar, analizar, diseñar el sistema antes de codificarlo.

Dentro de sus características fundamentales se tiene que:

- Mantiene la consistencia de los modelos del sistema de software.

- Chequeo de la sintaxis UML.
- Generación de documentación automáticamente.
- Generación de código a partir de los modelos.
- Ingeniería inversa (crear modelos a partir de código).

Rational Rose permite completar los siguientes flujos fundamentales del Proceso Unificado (RUP):

- Modelado del negocio (completo).
- Captura de requisitos (parcial).
- Análisis y diseño (completo).
- Implementación (como ayuda).
- Control de cambios y gestión de configuración (parcial).

Visual Paradigm.

Visual Paradigm es una herramienta para el modelado de sistemas informáticos que brinda un conjunto de facilidades. Es de gran utilidad para el desarrollo sobre lenguajes como PHP, pues posee autogeneración de código en dicho lenguaje.

Esta herramienta define su diseño centrado en casos de uso y enfocado al negocio, lo que permite generar un software de mayor calidad. Otra de las características es la realización de ingeniería tanto directa como inversa. Por otra parte se puede decir que es una herramienta colaborativa, es decir, soporta múltiples usuarios trabajando sobre el mismo proyecto.

Genera la documentación del proyecto automáticamente en varios formatos como Web o .pdf, y permite control de versiones. Nos ofrece modelo y código que permanece sincronizado en todo el ciclo de desarrollo, disponibilidad de múltiples versiones, para cada necesidad, disponibilidad de integrarse en los principales *IDEs* y disponibilidad en múltiples plataformas.

1.3.4 Arquitectura.

La arquitectura de software se define como la estructura de los componentes de un programa o sistema, sus interrelaciones, y los principios y reglas que gobiernan su diseño y evolución en el tiempo. Aporta además una visión abstracta de alto nivel, postergando el detalle de cada uno de los módulos definidos a pasos posteriores del diseño. Establece los fundamentos para que analistas, diseñadores y

programadores trabajen en una línea común que permita alcanzar los objetivos y necesidades del sistema.

Arquitectura Basada en Componentes.

La arquitectura de software de una aplicación basada en componentes consiste en un número pequeño de componentes específicos de la aplicación (que se diseñan específicamente para ella), que hacen uso de otros muchos componentes prefabricados que se acoplan entre sí para proporcionar los servicios que se necesitan en la aplicación.

En la tecnología de componentes la interfaz constituye el elemento básico de interconectividad. Cada componente debe describir de forma completa las interfaces que ofrece, así como las interfaces que requiere para su operación. Y debe operar correctamente con independencia de los mecanismos internos que utilice para soportar la funcionalidad de la interfaz. **(Bolso, 2004)**

Arquitectura Orientada a Servicios (SOA).

La arquitectura orientada a servicios define la utilización de servicios para dar soporte a los requerimientos de software del usuario. SOA proporciona una metodología y un marco de trabajo para documentar las capacidades de negocio y puede dar soporte a las actividades de integración y consolidación. Al contrario de las arquitecturas orientadas a objetos, las SOAs están formadas por servicios de aplicación débilmente acoplados y altamente interoperables. **(Solutions Group, S/A)**

Patrón arquitectónico. Modelo vista controlador.

La arquitectura de software, está muy relacionada al diseño y la implementación del sistema. Es el resultado de ensamblar un cierto número de elementos arquitectónicos de forma adecuada para satisfacer los requerimientos de un sistema, tanto funcionales como no funcionales; por ejemplo, de confiabilidad, portabilidad, y disponibilidad. **(Catalani, 2007)**

El Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

- Modelo: Representación específica de la información con la cual el sistema opera. Componente encargado del acceso a datos.

- Vista: Presenta el modelo en un formato adecuado para interactuar, usualmente se corresponde con la interfaz de usuario.
- Controlador: Responde a eventos, acciones del usuario e invoca cambios en el modelo y probablemente en la vista. El controlador puede procesar los datos y mostrarlos, es autosuficiente.

La ventaja del patrón Modelo Vista Controlador radica en la posibilidad de representar un mismo modelo en más de una vista, además posibilita la reutilización de código o la incorporación de este en programas más grandes.

1.3.5 Lenguaje de Programación.

Para el desarrollo de la aplicación se utilizará como lenguaje de programación PHP (*Hypertext Preprocessor*), este es un lenguaje del lado del servidor, gratuito y multiplataforma. Posee una gran librería de funciones, mucha documentación y es muy fácil de utilizar.

Entre las capacidades de este lenguaje se tiene:

- Muy fácil de aprender.
- Compatibilidad con las bases de datos *MySQL*, *mSQL*, *Oracle*, *Informix*, y *ODBC*.
- Funciones para el envío de correo electrónico, *upload* de archivos.
- Se caracteriza por ser un lenguaje muy rápido.
- Soporta en cierta medida la orientación a objeto. Clases y herencia.
- Posee documentación en su página oficial la cual incluye descripción y ejemplos de cada una de sus funciones.
- Es una alternativa de fácil acceso para todos.
- No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

1.3.6 Sistema Gestor de Bases de Datos (SGBD).

MySQL es un sistema gestor de base de datos rápido y fiable que posee muy buena integración con PHP y resulta muy adecuada para aplicaciones dinámicas basadas en Internet. *MySQL* es muy rápido, confiable, robusto y fácil de usar tanto para volúmenes de datos grandes como pequeños. **(Aramayo, S/A)**

Ventajas:

- Mayor rendimiento y velocidad.
- Mejores utilidades de administración (*backup*, recuperación de errores).
- En caso de fallos, no suele perder información ni corromper los datos.
- Mejor integración con PHP.
- No hay límites en el tamaño de los registros.
- Mejor control de acceso usuarios-tablas-permisos
- Consume muy pocos recursos de CPU y memoria.

1.4 Tecnologías y Herramientas a Utilizar.

Después de analizar las características fundamentales de las herramientas y tecnologías descritas en este capítulo se ha decidido desarrollar una aplicación web, utilizando como lenguaje de programación PHP con el *framework CodeIgniter*, el cual usa el patrón arquitectónico Modelo Vista Controlador. Además el sistema se implementará con una arquitectura basada en componentes y orientada a servicios, haciendo uso de *MySQL* como gestor de bases de datos.

La aplicación se realizará utilizando la metodología de desarrollo de software definida por RUP, en la cual se utiliza UML como lenguaje de modelado, garantizando así la elaboración de todas las fases de un producto de software de calidad y orientado a objeto.

Para la confección de los diagramas y demás artefactos se empleará *Visual Paradigm*, herramienta muy completa y de amplias potencialidades.

En este capítulo, se hizo referencia a los principales conceptos relacionados con el dominio del problema. Además, se realizó un estudio sobre los principales sistemas existentes a nivel nacional e internacional que se encuentran relacionados con el campo de acción del trabajo. También se analizaron un conjunto de tecnologías y herramientas, definiendo las que se utilizarán para el desarrollo de esta aplicación, así como una breve fundamentación de la utilización de cada una de ellas.

Capítulo II: Características del Sistema.

En este capítulo se describe la situación actual en el Centro Coordinador de Emergencia Médica Nacional, así como el modelo de negocio estudiado, actores, trabajadores y la descripción de los casos de uso del negocio. Además en el presente capítulo se hace una propuesta del sistema a desarrollar describiendo sus principales características, se especifican los requisitos funcionales y no funcionales, así como los casos de uso del sistema y su descripción.

2.1 Flujo Actual de los Procesos Involucrados en el Campo de Acción.

En estos momentos toda la gestión de la información asociada a los procesos que ocurren en el Centro Coordinador de Emergencia Médica Nacional se lleva a cabo de manera manual. Esta situación dificulta el funcionamiento óptimo de la institución, así como en muchos casos se producen demoras en la atención a los pacientes, pues se hace muy engorroso el trabajo del personal que allí labora. Por otra parte la búsqueda de información para la emisión de partes puede ser compleja, además en muchos casos el gran volumen de información almacenada en archivos puede extraviarse o duplicarse. Se hace difícil brindar determinada información estadística puesto que no hay forma de llevarla manualmente.

Descripción de los Procesos de Negocio.

Los principales procesos que ocurren en el Centro Coordinador de Emergencia Médica Nacional son: traslado de pacientes de institutos a hospitales, coordinación de emergencias dentro de la ciudad, traslados interprovinciales, eventos y actividades de primer nivel y situaciones de desastres y contingencias.

Traslado de pacientes de institutos a hospitales.

En este proceso se lleva a cabo la coordinación y expedición de móviles para los traslados entre instituciones y hospitales, es decir ya sean test diagnósticos o altas hospitalarias. El proceso es activado por un médico especialista a través de una llamada telefónica. Luego se recogen los datos correspondientes al paciente y se informa que será atendido, estos casos no se archivan como casos de emergencia, sino que se planifican para dar salida en el horario correspondiente, teniendo en cuenta la disponibilidad de camas para el traslado.

Toda la información es recogida por la enfermera coordinadora, la cual confecciona la historia clínica ambulatoria y asigna el móvil que hará el traslado.

Coordinación de emergencias dentro de la ciudad.

Este proceso es realizado fundamentalmente por el Centro de Coordinación Provincial de Ciudad de la Habana pero en ocasiones por cercanía u otras eventualidades es asumido por el Centro Coordinador de Emergencia Médica Nacional. El proceso es activado a través de una llamada telefónica; por personal médico y las diferentes entidades de Estado y Gobierno, así como por el MININT, MINFAR, entre otras y por la población solo en algunos casos. Luego se recogen los datos de la demanda y se activa el móvil de acuerdo a la prioridad.

Coordinación de traslados interprovinciales.

El proceso se inicia con una llamada telefónica procedente de los Centros Coordinadores Provinciales para pedir autorización de salida de un móvil de una provincia y hacer entrada en otra, ya sea cercana o colindante. Esta solicitud debe realizarse el día anterior y en casos de emergencia en el momento de salida del móvil. La información es recibida por el expedidor del Centro Coordinador de Emergencia Médica Nacional, y autorizada por el médico jefe de la guardia. (**Anexo 4**)

En caso de los traslados para la Ciudad de la Habana, el móvil debe pasar por el Centro Coordinador de Emergencia Médica Nacional para ser verificado y darle entrada a la ciudad, luego para su salida, este debe pasar nuevamente por dicho centro, pues existe un listado de demandas de pacientes por provincias con necesidad de traslado desde la Ciudad de la Habana hacia estas y se le va dando respuesta en dependencia de la disponibilidad de las ambulancias del Centro Coordinador de Emergencia Médica Nacional y también con las de las provincias que arriban a la ciudad. Es por ellos que se inspeccionan a la salida y se decide si llevan pacientes de retorno a su provincia o no.

2.2 Objeto de Automatización.

Con el fin de solucionar los problemas que se están presentando en el Centro Coordinador de Emergencia Médica Nacional se propone realizar una aplicación web, minimizando de esta forma los requisitos no funcionales del sistema. Se tiene como objeto de automatización inicial la autenticación de los usuarios al sistema, dando la posibilidad de asignar los permisos correspondientes a cada usuario según la labor que realiza, posibilitando así una mayor confidencialidad, seguridad y control de la información que se manipula.

La aplicación permitirá automatizar los procesos descritos anteriormente, así como la gestión y almacenamiento de toda esta información. Debe permitir una mayor eficiencia en los servicios prestados a la población, rapidez en la atención al paciente y menor tiempo de rescate. También permitirá generar reportes e informes de estadísticas de balance mensual y anual de los que actualmente no se lleva control. Además permitirá mantener información actualizada sobre el control de los móviles.

Por otra parte todos estos datos importantes serán almacenados en una base de datos, por lo que estarán disponibles el tiempo que se crea conveniente para su uso, dando la posibilidad de un mejor manejo y manipulación de la información almacenada.

2.3 Modelado del Negocio.

Modelo del negocio: Describe los procesos de un negocio y su interacción con elementos internos, es decir, describe las funciones que este realiza. Comprende la descripción de los actores, trabajadores y casos de uso que intervienen.

Actor del negocio: Es el rol que juega algo o alguien que interactúa con el negocio para beneficiarse de sus resultados.

Trabajadores del negocio: Son los que actúan directamente dentro del negocio.

Casos de uso del negocio: Son los procesos que se llevan a cabo en el negocio.

Las entidades del negocio: Representan a los objetos que los trabajadores manipulan o inspeccionan, producen o utilizan durante la realización de los casos de uso del negocio.

Diagrama de actividad: Es un tipo especial de diagrama de estados que muestra el flujo de actividades dentro del negocio, donde se representan los actores, trabajadores y entidades del negocio, así como las actividades realizadas por ellos y sus relaciones. (**Anexo 1**)

Modelo de objetos del negocio: Este modelo describe cómo colaboran los trabajadores y las entidades del negocio dentro del flujo de trabajo.

Justificación de Actores y Trabajadores del Negocio.

Actores del Negocio.

Actores del negocio	Justificación
Individuo	Cualquier persona que necesite de los servicios de atención médica móvil.
Jefe del Centro Coordinador	Persona que se encuentra al frente de la institución, encargado de supervisar todo el trabajo que en él se realiza.

Trabajadores del Negocio.

Trabajadores del negocio	Justificación
Enfermera Coordinadora	Persona encargada de recoger la demanda y procesarla hasta la asignación y puesta en marcha de la ambulancia, así como dar seguimiento a la misma.
Médico Jefe de la Guardia	Encargado de autorizar la salida de los móviles y controlar el trabajo en la mesa coordinadora.
Expedidor	Encargado de registrar los datos de los traslados interprovinciales, entrega de hoja de ruta, dar salida a los móviles y supervisarlos, así como planificar los traslados de pacientes a provincia.

Diagrama de Casos de Uso del Negocio.

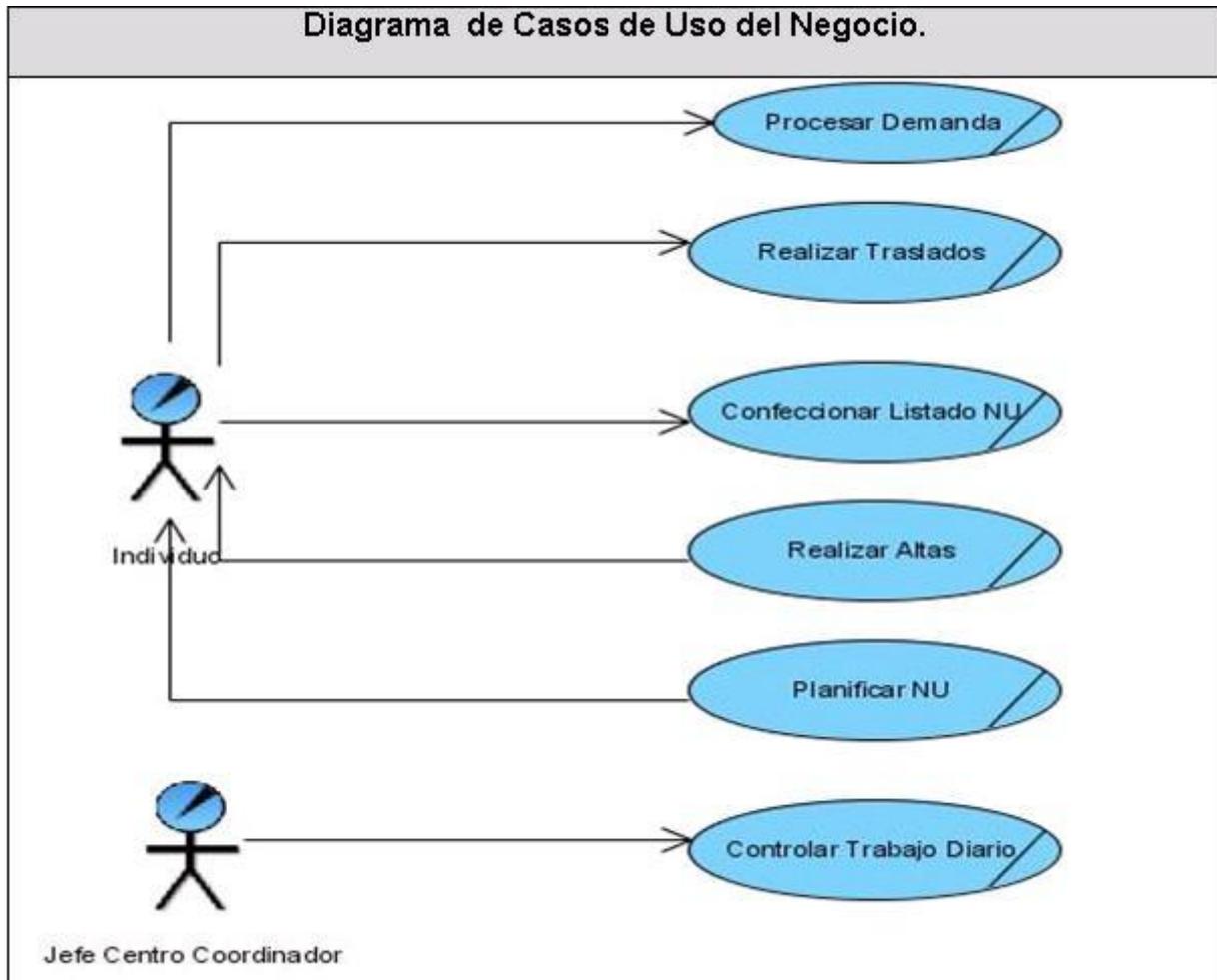


Figura 2. 1 Diagrama de Casos de Uso del Negocio.

Modelo de Objeto.

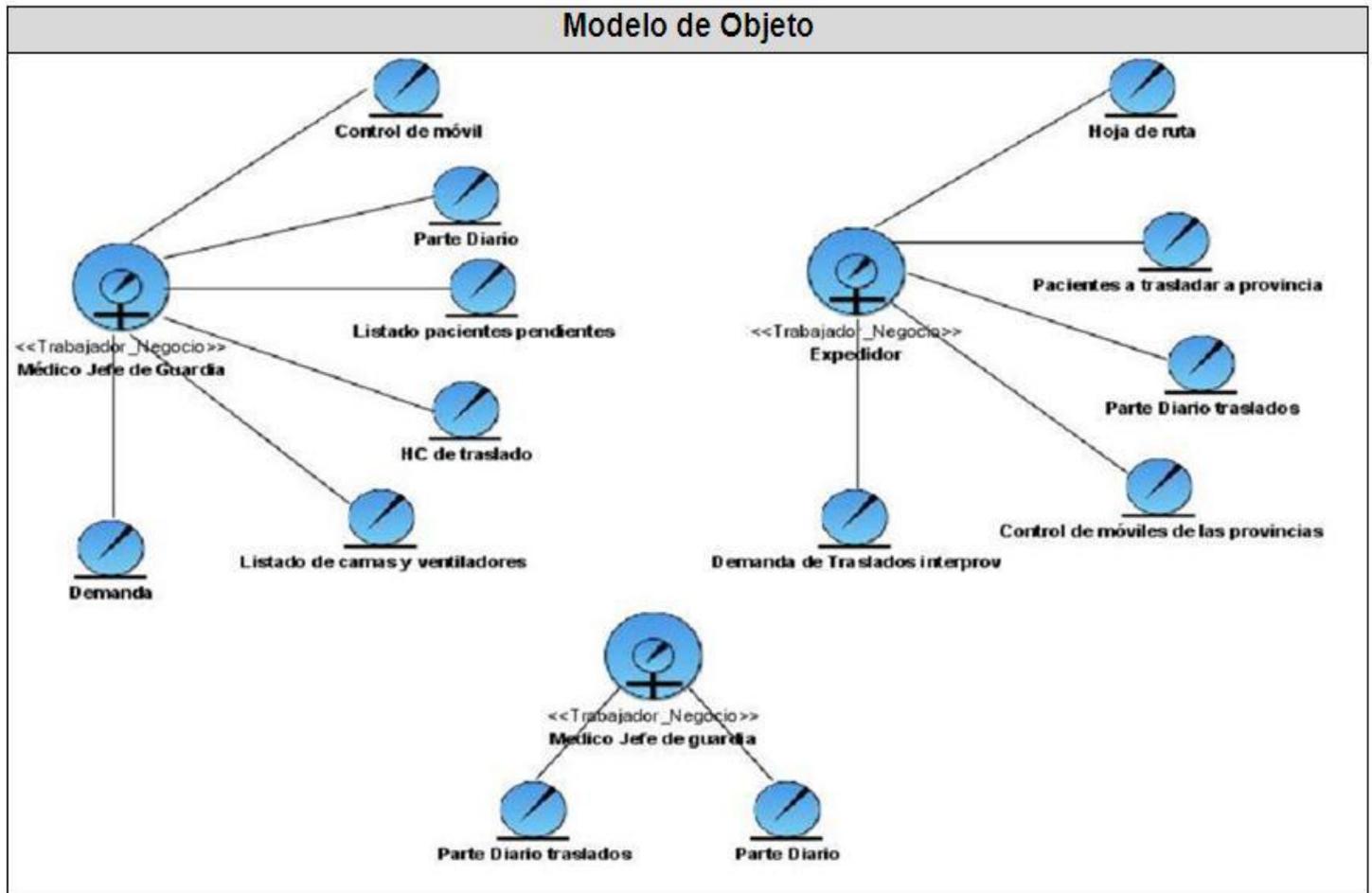


Figura 2. 2 Modelo de Objeto.

Descripción de los Casos de Uso del Negocio.

Caso de Uso:	Procesar Demanda
Actores:	Individuo(Inicia)
Trabajadores:	Enfermera Coordinadora, Médico Jefe de Guardia.
Propósito:	Atender una solicitud de servicios de atención médica móvil para salvar vidas.
Resumen:	El caso de uso (CU) se inicia cuando un individuo llama al Centro Coordinador de Emergencia Médica Nacional para activar una demanda. La enfermera coordinadora recoge los datos necesarios, asigna tipo de ambulancia, se

	autoriza la salida del móvil y luego se pone este en marcha, dándole seguimiento al mismo mediante claves, a través de la planta, finalizando así el caso de uso.
Precondiciones:	Se realice una llamada telefónica.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1- El individuo llama al CCEMN.	1.1- La enfermera coordinadora solicita datos necesarios para la demanda.
2- El individuo informa los datos solicitados.	<p>2.1- La enfermera coordinadora anota los datos.</p> <p>2.2 -La enfermera coordinadora verifica si es una emergencia o un alta.</p> <p>Ver Flujo Alterno paso 1</p> <p>2.3- Si es una emergencia se llena la HC de traslado.</p> <p>2.4- Asigna la ambulancia y se decide el móvil que realizará la demanda.</p> <p>Ver Flujo Alterno paso 2</p> <p>2.4- Si el móvil está en la base, el médico jefe de guardia autoriza la salida del mismo.</p> <p>2.5- La enfermera coordinadora realiza seguimiento a la demanda mediante claves a través de la planta de esta forma finaliza el CU.</p>
Flujos Alternativos de los Eventos	
Acción del Actor	Respuesta del Negocio

<p>Flujo Alternativo paso 1</p> <p>1- Si es un alta.</p> <p>2- La persona se retira, finalizando así el CU.</p>	<p>1.1- La enfermera coordinadora le informa a la persona que será atendido más tarde.</p>
<p>Flujo Alternativo paso 2</p>	<p>1- Si el móvil no está en la base, este localiza la ambulancia por planta que va a realizar el caso.</p> <p>2- Le informa todos los datos del caso y finaliza así el caso de uso.</p>
<p>Mejoras</p>	<ol style="list-style-type: none"> 1. Recoger otros datos importantes que actualmente no se tienen para llevar control estadístico. 2. Toda la información sería guardada en una base de datos (BD) a la cual se podría acceder en cualquier momento y trabajar con ella. 3. Permitiría realizar reportes importantes para el centro Coordinador de Emergencia Médica Nacional de manera fácil. 4. Mejoraría el control de los móviles y el trabajo con ellos. 5. Aparece el controlador, encargado de dar seguimiento a la demanda.
<p>Prioridad</p>	<p>Responde al principal objetivo de automatización al resolver en gran medida con los problemas actuales.(Crítico)</p>

<p>Caso de Uso:</p>	<p>Registrar Traslados</p>
<p>Actores:</p>	<p>Individuo(Inicia)</p>
<p>Trabajadores:</p>	<p>Expedidor y Jefe del Centro Coordinador</p>
<p>Propósito:</p>	<p>Registrar los traslados interprovinciales y autorizar la salida de estos.</p>
<p>Resumen:</p>	<p>El caso de uso (CU) se inicia cuando un individuo llama al Centro Coordinador de Emergencia Médica Nacional para activar una demanda de traslado hacia otra provincia. El expedidor recoge los datos necesarios y el CCEMN autoriza la salida del móvil de la provincia para entrar a otra, finalizando así el caso de uso.</p>
<p>Precondiciones:</p>	<p>Se realiza una llamada telefónica.</p>
<p>Flujo Normal de Eventos</p>	

Acción del Actor	Respuesta del Negocio
1- El Individuo llama al CCEMN.	1.1- Expedidor solicita datos necesarios para la demanda.
2- El individuo informa los datos solicitados.	2.1- Expedidor anota los datos. 2.2- El Jefe del Centro Coordinador de Emergencia Médica Nacional autoriza la salida del móvil. 2.3- Verifica origen y destino del traslado. <p style="text-align: center;">Ver Flujo Alternativo paso 1</p> 2.4- Si es un traslado para la Ciudad de la Habana. 2.5- Supervisa este al llegar al CCEMN y verifica retornos. <p style="text-align: center;">Ver Flujo Alternativo paso 2</p> 2.6- Si hay retornos, se verifica capacidad. <p style="text-align: center;">Ver Flujo Alternativo paso 3</p> 2.7- Si hay capacidad, el expedidor coordina traslados de pacientes a la provincia origen, finalizando así el caso de uso.
Flujos Alternativos de los Eventos	
Acción del Actor	Respuesta del Negocio
<p style="text-align: center;">Flujo Alternativo paso 1</p> 1- Si el traslado no es para la Ciudad de la Habana.	1.1- El Jefe del Centro Coordinador de Emergencia Médica Nacional autoriza la salida del móvil, finalizando así el caso de uso.
<p style="text-align: center;">Flujo Alternativo paso 2</p> 1- Si no hay retornos	1.1- El expedidor despide el móvil a su provincia origen, finalizando así el caso de uso.

Flujo Alternativo paso 3 1- Si no hay capacidad.	1.2- El expedidor despide el móvil a su provincia de origen, finalizando así el CU.
Mejoras	<ol style="list-style-type: none"> 1. Mejor control de los móviles que se trasladan de una provincia a otra. 2. Se elimina el teléfono para la activación de la demanda de traslado interprovincial. 3. Recoger datos importantes con fines estadísticos, que actualmente no se llevan para un mejor control. 4. Los datos serían guardados en una BD para luego ser gestionados en beneficio del centro. 5. Brindar reportes importantes para la institución y el país.
Prioridad	Responde al principal objetivo de automatización al resolver gran parte de los problemas actuales.(Crítico)

Caso de Uso:	Controlar Trabajo Diario
Actores:	Jefe del Centro Coordinador
Trabajadores:	Enfermera Coordinadora, Médico Jefe de Guardia y Expedidor.
Propósito:	Controlar el trabajo diario del centro.
Resumen:	El caso de uso (CU) inicia cuando el jefe del centro coordinador solicita documentos e informes al médico jefe de guardia, este a su vez le hace la solicitud a la enfermera coordinadora y al expedidor para tener control de una serie de parámetros importantes para el centro y para el país, cada responsable confecciona la información y se la brinda al jefe del CCEMN, finalizando así el CU.
Precondiciones:	<ol style="list-style-type: none"> 1. Demandas registradas. 2. Documento de control del móvil confeccionado y actualizado.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1- El jefe del centro coordinador solicita	1.1- El médico jefe de guardia informa solicitud a la

información.	<p>enfermera coordinadora y expedidor del centro.</p> <p>1.2- La enfermera coordinadora y el expedidor revisan documentos y confeccionan la información.</p> <p>1.3- El médico jefe de guardia revisa los documentos y envía información solicitada.</p>
Flujos Alternativos de los Eventos	
Acción del Actor	Respuesta del Negocio
-	
Mejoras	<ol style="list-style-type: none"> 1. Facilitar la gestión de toda la información del centro. 2. Mejor control de los recursos. 3. Los datos serían guardados en una BD para luego ser gestionados en beneficio del centro. 4. Brindar reportes importantes para la institución y el país. 5. Mejorar la seguridad y privacidad de la información. 6. El trabajo se hace más cómodo.
Prioridad	Responde al principal objetivo de automatización al resolver gran parte de los problemas actuales.(Crítico)

Caso de Uso:	Confeccionar Listado NU
Actores:	Individuo(Inicia)
Trabajadores:	Enfermera Coordinadora
Propósito:	Confeccionar el listado de test diagnósticos y turnos médicos planificados.
Resumen:	El caso de uso (CU) se inicia cuando un individuo llama al Centro Coordinador de Emergencia Médica Nacional para solicitar el servicio de ambulancia para un traslado a institutos y hospitales para realizar un test diagnóstico, turno médico o alta hospitalaria, la enfermera coordinadora solicita y recoge los datos e informa que será atendido, finalizando así el caso de uso.
Precondiciones:	Se realiza una llamada telefónica.
Flujo Normal de Eventos	

Acción del Actor	Respuesta del Negocio
1- El Individuo llama al CCEMN.	1.1- La enfermera coordinadora solicita los datos.
2- El individuo informa los datos solicitados.	2.1 - Expedidor anota los datos. 2.2- La enfermera coordinadora recoge los datos del paciente. 2.3- La enfermera coordinadora informa que será atendido, finalizando así el caso de uso.
Mejoras	1. La confección de un listado previo con todos los casos de no urgencias almacenados en una base dato, para ser accedidos en cualquier momento y puedan ser consultados. 2. La obtención de información importante para la confección de reportes estadísticos importantes.
Prioridad	Responde al principal objetivo de automatización al resolver gran parte de los problemas actuales.(Critico)

Caso de Uso:	Realizar Altas
Actores:	Individuo
Trabajadores:	Enfermera Coordinadora, Médico Jefe de Guardia
Propósito:	Dar salida a móviles para el cumplimiento de una demanda.
Resumen:	El caso de uso (CU) se inicia cuando una vez llegada las 4 de la tarde la enfermera coordinadora se dispone a dar cumplimiento a las solicitudes de altas de pacientes. Se verifica si el paciente continua en la institución a la cual se debe recoger, en caso positivo se asigna un móvil, el cual debe ser autorizado por el médico jefe de guardia y se le da un seguimiento a través de claves, en caso negativo no se envía el móvil, finalizando así el caso de uso.
Precondiciones:	El listado de las actas confeccionado y registrado en las demandas.
Flujo Normal de Eventos	

Acción del Actor	Respuesta del Negocio
	<p>1.1- Al llegar las 4:00pm la enfermera coordinadora verifica si la demanda ha sido cancelada.</p> <p>Ver Flujo Alternativo paso 1</p> <p>1.2- Si no fue cancelada, la enfermera coordinadora asigna la ambulancia y se decide el móvil que realizará la demanda.</p> <p>Ver Flujo Alternativo paso 2</p> <p>1.3- Si el móvil está en la base, el médico jefe de guardia autoriza la salida del mismo.</p> <p>1.4- La enfermera coordinadora realiza seguimiento a la demanda mediante claves a través de la planta de esta forma finaliza el CU.</p>
Flujos Alternativos de los Eventos	
Acción del Actor	Respuesta del Negocio
Flujo Alternativo paso 1	1- Si la demanda fue cancelada, termina el proceso de la demanda, finalizando así el caso de uso.
Flujo Alternativo paso 2	<p>2.1- Si el móvil no está en la base, este localiza la ambulancia por planta que va a realizar el caso.</p> <p>2.2- Le informa todos los datos del caso, procede al seguimiento de la demanda a través de la planta mediante claves, finaliza así el caso de uso.</p>
Mejoras	<ol style="list-style-type: none"> 1. Se organizaría de una mejor forma el proceso de realizar las actas. 2. Se logra tener un mejor control de los móviles sobre la función que están realizando.

Prioridad	Responde al principal objetivo de automatización al resolver gran parte de los problemas actuales.(Crítico)
------------------	---

Caso de Uso:	Planificar NU
Actores:	Individuo
Trabajadores:	Enfermera Coordinadora, Médico Jefe de guardia
Propósito:	Dar salida a los móviles para cumplir con las demandas planificadas del día anterior.
Resumen:	El caso de uso se inicia, una vez confeccionado el listado de las NU el día anterior, se procede a la coordinación y organización de dichas demandas según los horarios registrados, asigna el móvil, se le da salida a este y se procede al seguimiento de la demanda, finalizando así el caso de uso.
Precondiciones:	El listado de las no urgencias ya confeccionado, registrado en Confeccionar Listado NU.

Flujo Normal de Eventos

Acción del Actor	Respuesta del Negocio
.	<p>1.1-Al llegar el día siguiente la enfermera coordinadora decide el móvil que realizará la demanda</p> <p>Ver Flujo Alterno paso 1</p> <p>1.2- Si el móvil está en la base, el médico Jefe de Guardia autoriza la salida del mismo.</p> <p>1.4- La enfermera coordinadora realiza seguimiento a la demanda mediante claves a través de la planta de esta forma finaliza el CU.</p>

Flujos Alternativos de Eventos

Acción del Actor	Respuesta del Negocio
------------------	-----------------------

<p>Flujo Alternativo paso 1</p>	<p>2- Si el móvil no está en la base, este localiza la ambulancia por planta que va a realizar el caso.</p> <p>2.1- Le informa todos los datos del caso, procede al seguimiento de la demanda a través de la planta mediante claves, finaliza así el caso de uso.</p>
<p>Mejoras</p>	<ol style="list-style-type: none"> 1. Se organizaría de una mejor forma el proceso de planificación de las no urgencias. 2. Se logra tener un mejor control de los móviles sobre la función que están realizando. 3. Se tiene la información guardada en una base de datos, la cual puede accederse en cualquier momento y trabajar con ella. 4. Permite emitir reportes.
<p>Prioridad</p>	<p>Responde al principal objetivo de automatización al resolver gran parte de los problemas actuales.(Crítico)</p>

2.4 Propuesta de Sistema.

El Ministerio de Salud Pública en Cuba lleva adelante desde hace algunos años un proceso de perfeccionamiento en las unidades municipales y en todas las instalaciones del sector de la salud, para garantizar una mejor asistencia a la población y calidad en los servicios prestados. Como parte de estas transformaciones se han desarrollado en el país varias aplicaciones con el fin de informatizar el sistema de salud cubano.

La aplicación desarrollada para el Centro Coordinador de Emergencia Médica Nacional formará parte de este perfeccionamiento, facilitando la gestión de la información de los procesos llevados a cabo en la institución. Permitirá recoger los datos de un traslado interprovincial desde la provincia o la nación, autorizar la salida de este de su provincia, así como tener control de los móviles que viajan de un lugar a otro. Posibilitará darle entrada al móvil en caso de arribar a la Ciudad de la Habana, registrar y gestionar los pacientes a trasladar para su provincia, además de insertar y planificar altas, turnos médicos y test diagnósticos solicitados dentro de la Ciudad de la Habana, asignar móviles a cada caso

de demanda solicitado y gestionar todos los procesos que en él ocurren, así como ofrecerá la posibilidad de emitir reportes estadísticos sobre el control del trabajo diario.

Será una aplicación Web que estará hospedada en los servidores de Infomed a la cual se podrá acceder con una PC cliente desde cualquier parte del país siempre y cuando se cuenten con los permisos pertinentes. El sistema se relaciona con varios componentes existentes en SISalud (Registro de Unidades de Salud, Registro del Personal de Salud, Registro de Ubicación, Registro de Ciudadanos, además del SAAA para la autenticación de los usuarios).

2.4.1 Especificación de Requerimientos de Software.

Los requerimientos de software son condiciones o capacidades que debe tener el sistema para satisfacer las necesidades de un cliente, estos deben ser especificados por escrito. A continuación se muestra un listado de los requerimientos tanto funcionales como no funcionales de la aplicación.

Requerimientos Funcionales.

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir, indican que es lo que el software debe hacer, especifican como debe comportarse el sistema en situaciones particulares y como debe ser el comportamiento de entrada y salida del sistema.

RF 1: Gestionar Demanda.

RF 1.1: Insertar Demanda.

RF 1.2: Modificar Demanda.

RF 1.3: Listar Demanda Emergencia.

RF 1.4: Buscar Planificación Demanda.

RF 1.5: Buscar Altas.

RF 2: Gestionar Demanda Traslado Interprovincial.

RF 2.1: Insertar Demanda Traslado Interprovincial.

RF 2.2: Modificar Demanda Traslado Interprovincial.

RF 2.3: Visualizar Demanda Traslado Interprovincial.

RF 3: Buscar Demanda Traslado Interprovincial.

RF 3.1 Buscar Demanda Sin Permiso.

RF 3.2: Buscar Demanda Aceptada.

RF 3.3: Buscar Entradas Traslado Interprovincial.

RF 3.4: Buscar Traslado Interprovincial para Entradas.

RF 4: Gestionar Paciente Demanda Traslado Interprovincial.

RF 4.1: Insertar Paciente Demanda Traslado Interprovincial.

RF 4.2: Listar Pacientes Demanda Traslado Interprovincial.

RF 4.3: Eliminar Pacientes Demanda Traslado Interprovincial.

RF 4.4: Modificar Pacientes Demanda Traslado Interprovincial.

RF 5: Gestionar Demanda Paciente a Provincia.

RF 5.1: Insertar Demanda Paciente a Provincia.

RF 5.2: Visualizar Demanda Paciente a Provincia.

RF 5.3: Eliminar Demanda Paciente a Provincia.

RF 5.4: Modificar Demanda Paciente a Provincia.

RF 5.5: Insertar Cumplimiento Demanda Paciente a Provincia.

RF 6: Buscar Demanda Paciente a Provincia.

RF 7: Gestionar Entrada.

RF 7.1: Insertar Entrada.

RF 7.2: Insertar Pacientes Entrada.

RF 7.3: Modificar Pacientes Entrada.

RF 7.4: Eliminar Pacientes Entrada.

RF 8: Insertar Salida.

RF 9: Insertar Permiso.

RF 10: Gestionar Usuario.

RF 10.1: Insertar Usuario.

RF 10.2: Insertar Rol.

RF 10.3: Eliminar Rol.

RF 10.4: Eliminar Usuario.

RF 10.5: Buscar Usuario en el Registro de Ciudadano.

RF 10.6: Listar Usuario.

RF 11: Gestionar Enfermedad Trazadora.

RF 11.1: Insertar Enfermedad Trazadora.

RF 11.2: Eliminar Enfermedad Trazadora.

RF 11.3: Modificar Enfermedad Trazadora.

RF 11.4: Listar Enfermedad Trazadora.

RF 12: Gestionar Marca Móvil.

RF 12.1: Insertar Marca Móvil.

RF 12.2: Eliminar Marca Móvil.

RF 12.3: Modificar Marca Móvil.

RF 12.4: Listar Marca Móvil.

RF 13: Gestionar Clave.

RF 13.1: Insertar Clave.

RF 13.2: Eliminar Clave.

RF 13.3: Modificar Clave.

RF 13.4: Listar Clave.

RF 14: Gestionar Centro Coordinador.

RF 14.1: Insertar Centro Coordinador.

RF 14.2: Eliminar Centro Coordinador.

RF 14.3: Modificar Centro Coordinador.

RF 14.4: Listar Centro Coordinador.

RF 15: Autenticar Usuario.

RF 16: Generar Parte Traslados Provincia.

RF 16.1: Generar Parte Traslado Interprovincial.

RF 16.2: Buscar Parte Traslado Interprovincial.

RF 16.3: Imprimir Parte Traslado Interprovincial.

Requerimientos no Funcionales.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Estas propiedades son las características que hacen al producto atractivo, usable, rápido o confiable.

RNF 1: Apariencia o interfaz externa

- La interfaz del sistema no contiene numerosas imágenes para evitar demoras en la respuesta a cualquier acción del usuario.
- La misma será sencilla, amigable e intuitiva, de fácil navegación por parte del usuario. Estará diseñada para una óptima visualización siendo adaptable a cualquier resolución.

RNF 2: Apariencia o interfaz interna

- Los componentes del sistema serán desarrollados siguiendo el principio de Alta Cohesión y Bajo Acoplamiento, y los que sean reutilizables en los diferentes módulos del sistema serán desarrollados como Servicios Web XML que interactuarán con otros componentes a través de SOAP.

RNF 3: Usabilidad

- La aplicación Web deberá facilitar la interacción usuario – sistema con el objetivo de evitar rechazo en el uso de la misma, y guiará mediante mensajes al usuario en las diferentes acciones que realice.
- El usuario deberá poseer conocimientos básicos del manejo de computadoras y estar familiarizado con la gestión y el flujo de la información en el área para la cual se desarrolla el sistema informático.

RF 4: Soporte

- La aplicación Web contará con una ayuda donde el usuario podrá suplir las dudas que se le puedan presentar durante la utilización de la misma. El usuario del módulo deberá recibir un adiestramiento previo en la utilización del sistema con el fin de que pueda explotar las prestaciones del sistema sin contratiempos ocasionados por la falta de preparación técnica.

RF 5: Portabilidad

- El sistema informático está desarrollado sobre una plataforma Web y podrá ser utilizado desde cualquier sistema operativo, recomendándose para su uso Windows o distribuciones de Linux.

RF 6: Rendimiento

- El tiempo de respuesta de una petición al servidor y la velocidad de procesamiento de la información deben ser rápidas para evitar retrasos en la actualización de datos, así como en la toma de decisiones.

RF 7: Seguridad

RF 7.1: Confidencialidad

- La información que brinda el sistema estará protegida contra el acceso de usuarios no autorizados. Solamente los administradores del sistema podrán realizar cambios en la configuración y en la información.

RF 7.2: Disponibilidad

- El sistema estará accesible cada vez que los usuarios del mismo lo requieran.

RF 8: Software

RF 8.1: Software servidor

- El servidor debe contar con sistema operativo Linux/Debian 4 Etch, Apache.

RF 8.2: Software Cliente

- Para utilizar la aplicación Web será necesaria una computadora con el sistema operativo Windows o Linux en cualquiera de sus versiones, recomendándose Windows XP o superior y Ubuntu 7.10 o superior. Además se podrá acceder desde cualquier navegador Web recomendándose Internet Explorer 6 o superior y Firefox 2.0 o superior.

RF 9: Hardware

- Para garantizar el correcto funcionamiento de la aplicación se necesitan como requerimientos mínimos una computadora con un procesador Pentium II o superior, una memoria RAM de 512 MB o más, un disco duro de 10 GB o más y una tarjeta de red a 128Mbps o más.

RF 10: Restricciones en el diseño y la implementación

- Se utilizarán los patrones de diseño establecidos.
- La lógica de presentación constituirá una capa independiente de la lógica de negocio, centrandó su función en la interfaz de usuario y validaciones simples de la entrada de datos.
- Para el análisis y el diseño del sistema deberá ser utilizada la metodología RUP, usando el lenguaje de modelado UML y como herramienta para llevarlo a cabo el Visual Paradigm.
- Para la implementación se utilizará como lenguaje de programación PHP 5 y como herramienta de desarrollo el Zend Studio.

2.4.2 Dependencias y Relaciones con otros Sistemas.

El sistema de gestión de información para el Centro Coordinador de Emergencia Médica Nacional tiene dependencia del Sistema de Auditoria, Autenticación y Autorización (SAAA) para el acceso a la aplicación, al hacer uso del servicio de autenticación, interactúa con el Registro de Ubicación, pues es necesario para la aplicación el uso de los municipios y provincias de Cuba. Además, la aplicación interactúa con el Registro de Ciudadanos, para registrar el ciudadano en la aplicación y luego pueda ser autorizado a acceder a las funcionalidades de esta según el rol que se le asigne.

Por otra parte utiliza el Registro de Unidades de Salud y el Registro del Personal de Salud; el primero con el objetivo de conocer las diferentes unidades de salud de Cuba y el segundo para comprobar el registro profesional de cualquier persona que trabaje en el Ministerio de Salud Pública. Además el sistema hace uso del servicio de disponibilidad de ambulancia que ofrece el Módulo de Operaciones para asignar un móvil a una petición de rescate o atención médica.

2.5 Definición de los Casos de Uso.

Trabajadores del negocio	Justificación
Enfermera coordinadora CCEMN	Es la encargada de recoger los datos de las demandas de pacientes para traslado a provincia.
Persona	Generalización de los actores Enfermera coordinadora CCEMN y del Controlador, encargada de gestionar las demandas (solicitudes de atención dentro de la Ciudad de

	la Habana).
Enfermera Coordinadora	Generalización de los actores Enfermera coordinadora CCEMN y el actor Enfermera coordinadora del CCPEM encargada de insertar los traslados interprovinciales.
Trabajador	Generalización de los actores Enfermera Coordinadora CCPEM y del Coordinador encargado de gestionar las demandas planificadas y los pacientes de ella.
Coordinador	Responsable de gestionar pacientes traslado interprovincial, insertar permiso, listar móviles de las provincias e insertar salida.
Médico Jefe de la guardia	Persona que se encuentra al frente del Centro Coordinador de Emergencia médica Nacional, responsable de autorizar la salida de los móviles, además dirige y controla el trabajo en el centro y es el encargado de generar los partes para el control del trabajo diario.
Administrador	Es el encargado de gestionar los nomencladores o codificadores utilizados en el sistema y darle soporte a la aplicación.

2.5.1 Organización de los Elementos: Casos de Uso en Paquetes.

Los paquetes son usados para organizar y manipular la complejidad de los modelos cuando estos alcanzan un tamaño considerable, agrupando los casos de uso en subconjuntos más pequeños. Este ofrece una vista más detallada del modelo de casos de uso del sistema.

Para agrupar los casos de uso en paquetes se debe tener en cuenta que los elementos estén estrechamente relacionados entre sí.

Dependencias de paquetes: Es la relación que se establece entre paquetes, significa que los elementos de un paquete conocen de alguna forma los del otro y estos a su vez se encuentran acoplados.

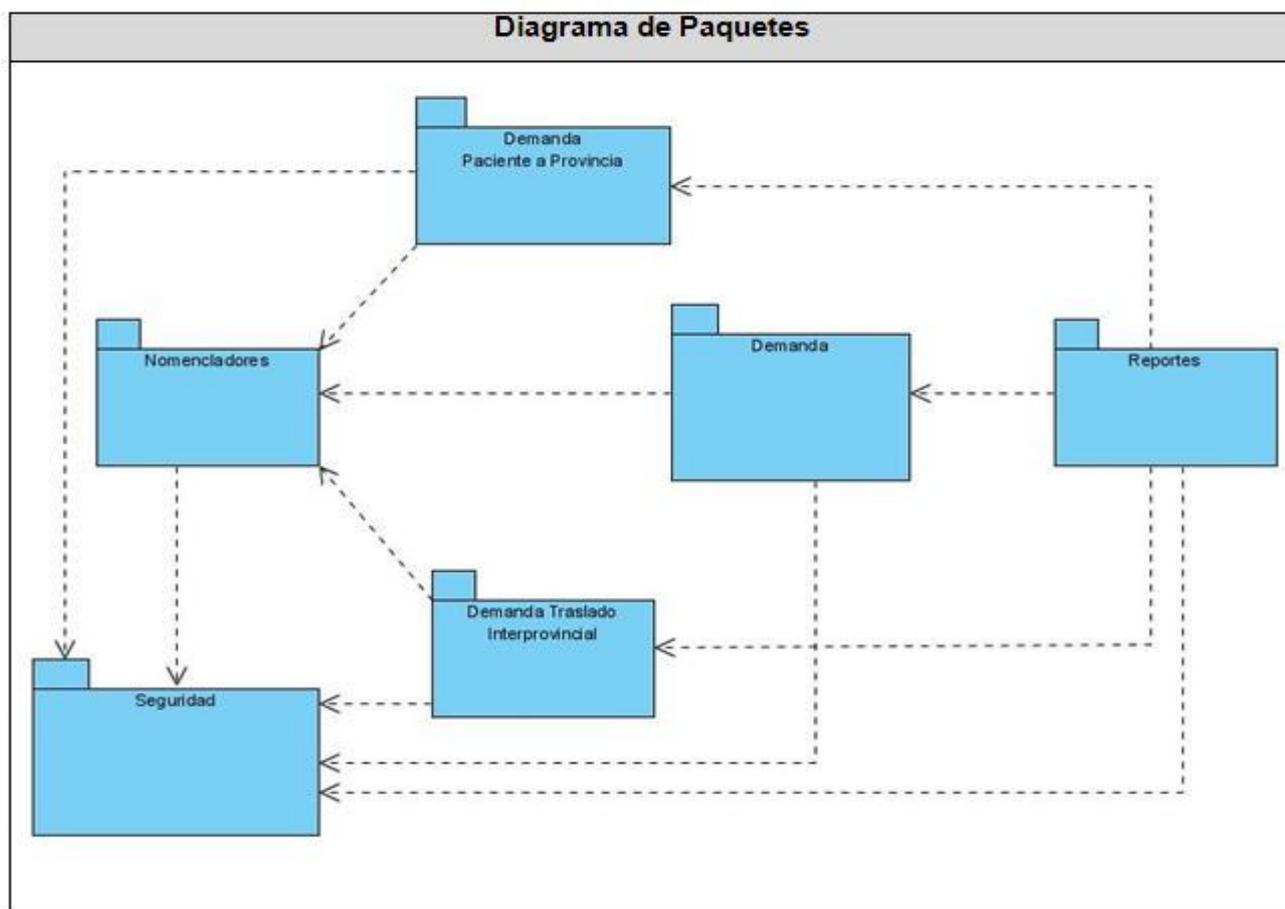


Figura 2. 3 Diagrama de Paquetes.

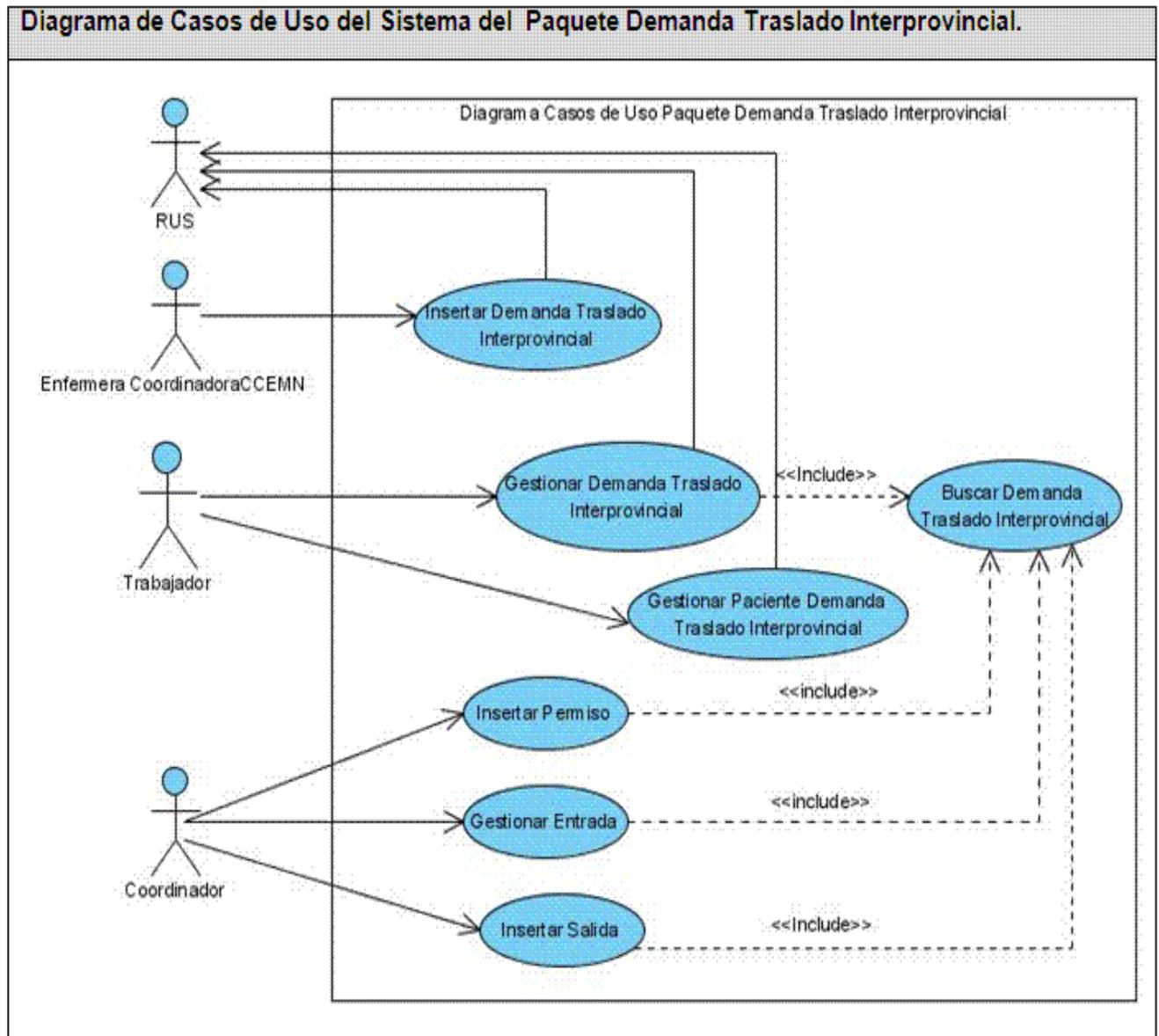


Figura 2. 4 Diagrama de Casos de Uso del Sistema. Paquete Demanda Traslado Interprovincial.

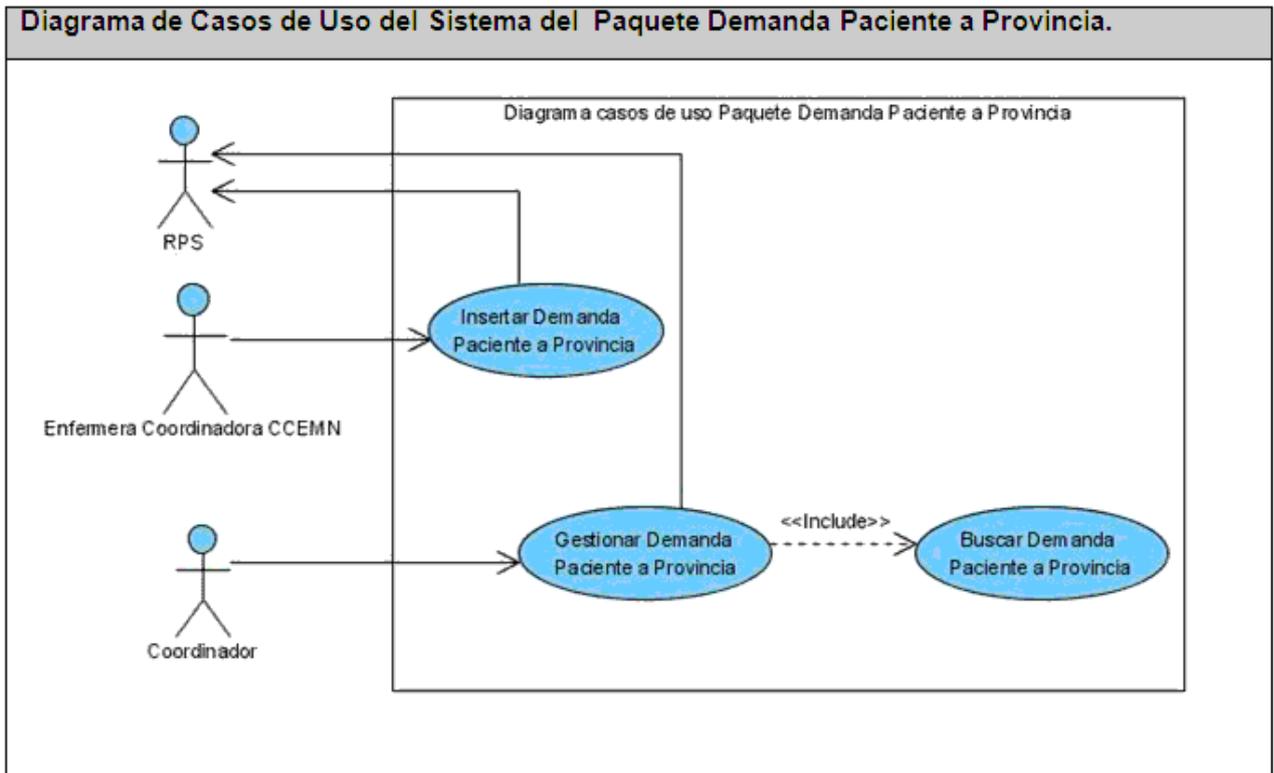


Figura 2. 5 Diagrama de Casos de Uso del Sistema. Paquete Demanda Paciente a Provincia.

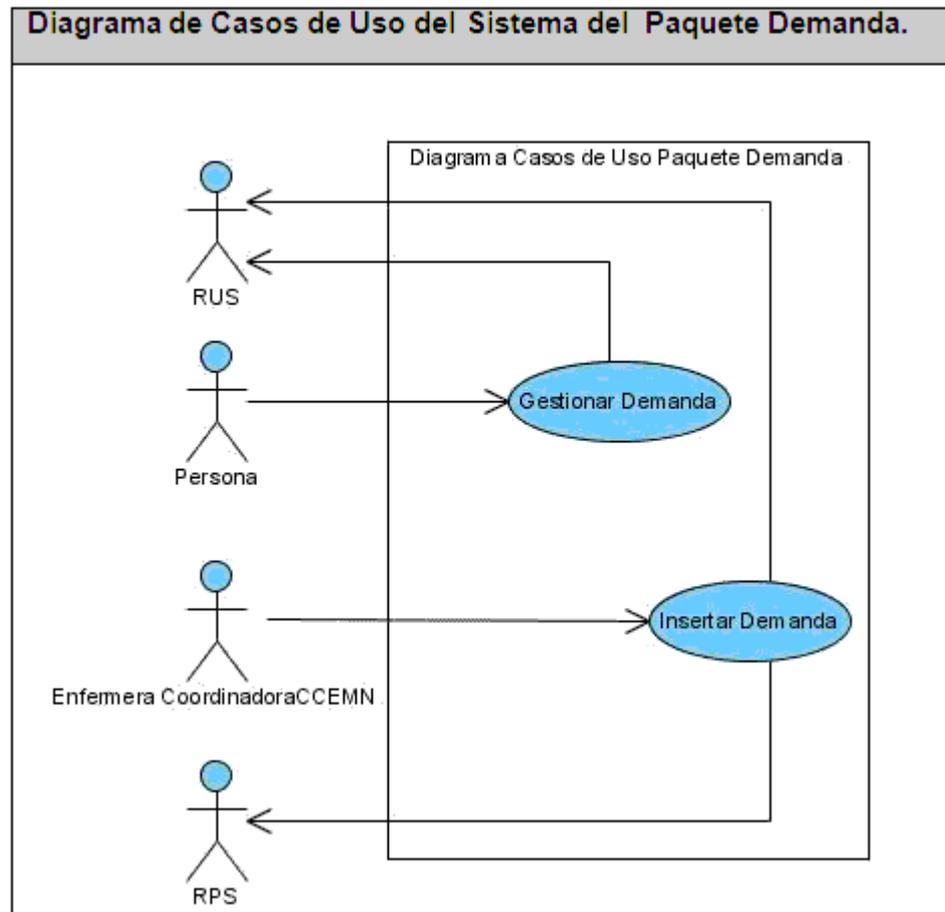


Figura 2. 6 Diagrama de Casos de Uso del Sistema. Paquete Demanda.

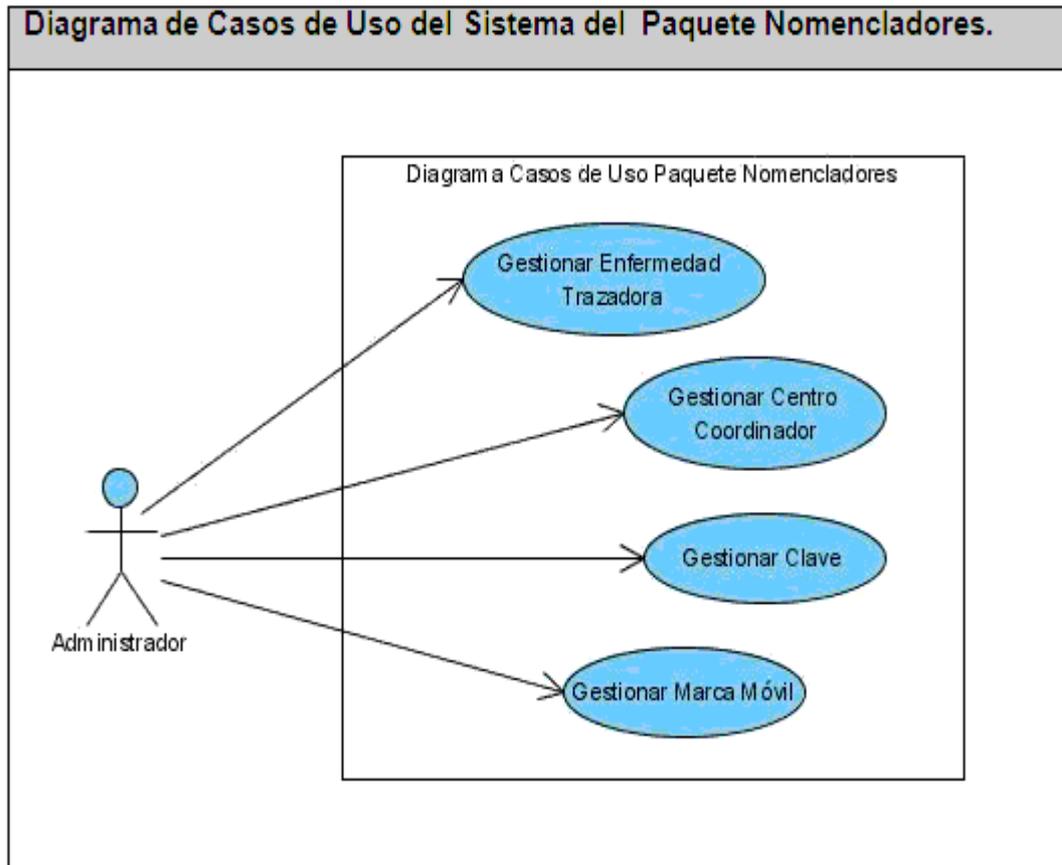


Figura 2. 7 Diagrama de Casos de Uso del Sistema. Paquete Nomencladores.

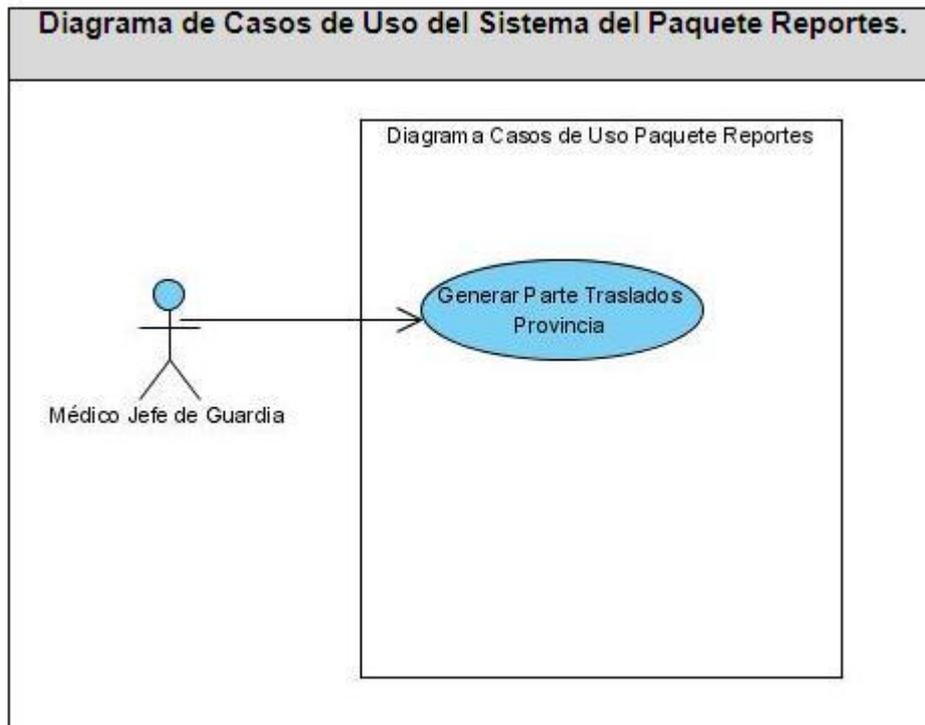


Figura 2. 8 Diagrama de Casos de Uso del Sistema. Paquete Reportes.

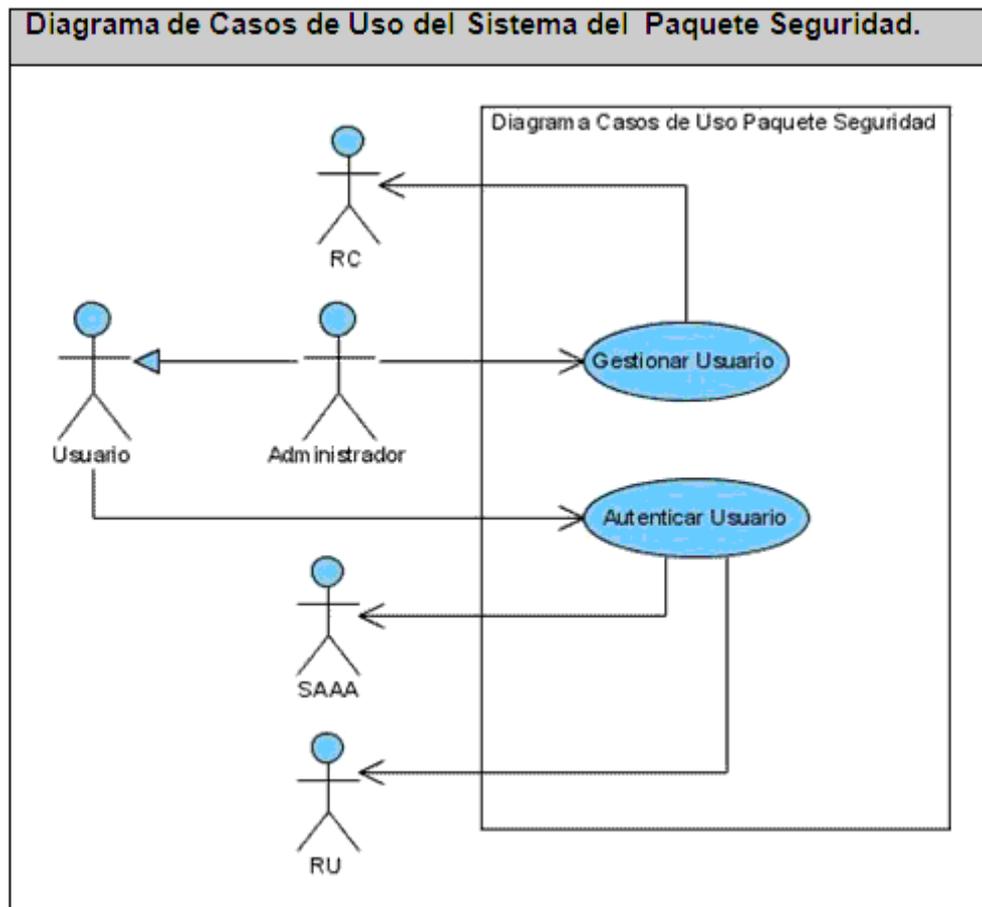


Figura 2. 9 Diagrama de Casos de Uso del Sistema. Paquete Seguridad.

2.5.2 Descripción Textual de los Casos de Uso.

Caso de Uso:	Insertar Demanda Traslado Interprovincial.
Actores:	Enfermera Coordinadora CCEMN.
Propósito:	Permitir registrar las solicitudes de demandas de traslados interprovinciales.
Resumen:	El caso de uso se inicia cuando la Enfermera Coordinadora del CCEMN selecciona la opción Insertar Traslado Interprovincial de la página principal, el sistema muestra una interfaz con los datos a insertar, guardando estos una vez registrados y actualizando de igual forma la base de datos de la

	aplicación, finalizando así el caso de uso.
Precondiciones:	La Enfermera Coordinadora CCEMN debe estar autenticada en el sistema.
Poscondiciones:	Demanda traslado interprovincial insertada y actualizada la base de datos.
Referencias:	RF 2: RF 2.1, RF 15.
Prioridad:	Crítico.

Caso de Uso:	Gestionar Demanda Traslado Interprovincial.
Actores:	Trabajador.
Propósito:	Permitir gestionar la información referente a las demandas de traslados interprovinciales (Modificar Demanda Traslado Interprovincial y Visualizar Demanda Traslado Interprovincial).
Resumen:	El caso de uso se inicia cuando el Trabajador necesita gestionar las demandas de traslado interprovincial. El Trabajador debe buscar primero las demandas de traslado interprovincial, luego selecciona la opción deseada, pudiendo modificar y visualizar estas, finalizando así el caso de uso.
Precondiciones:	El Trabajador debe estar autenticado en el sistema. Las demandas deben ser buscadas primero. (Ejecutar el Caso de Uso Buscar Demandas Traslado interprovincial) Ventana de Gestionar Demanda Traslado Interprovincial abierta y lista.
Poscondiciones:	Demanda Traslado Interprovincial modificada y actualizada la base de datos. Demanda Traslado Interprovincial visualizada.

Referencias:	RF 2, RF 3, RF 15.
Prioridad:	Crítico.

Caso de Uso:	Buscar Demanda Traslado Interprovincial.
Actores:	Trabajador.
Propósito:	Buscar las demandas de traslado interprovincial.
Resumen:	El caso de uso se inicia cuando el Trabajador necesita buscar las demandas de traslado interprovincial realizadas al Centro Coordinador de Emergencia Médica Nacional, este escoge la opción “Buscar Demandas Traslado Interprovincial”, introduce el criterio de búsqueda y el sistema muestra un listado con las demandas realizadas, finalizando así el caso de uso.
Precondiciones:	El Trabajador debe estar autenticado en el sistema. Demandas de traslado interprovincial insertadas en la aplicación.
Poscondiciones:	Listado de Demanda Traslado Interprovincial.
Referencias:	RF 3, RF 15.
Prioridad:	Crítico.

Caso de Uso:	Insertar Demanda Paciente a Provincia.
Actores:	Enfermera Coordinadora CCEMN.
Propósito:	Permitir registrar las solicitudes de demandas de pacientes a provincia.
Resumen:	El caso de uso se inicia cuando la Enfermera Coordinadora del CCEMN selecciona la opción Insertar Demanda Paciente a Provincia de la página

	principal, el sistema muestra una interfaz con los datos a insertar, guardando estos una vez registrados y actualizando de igual forma la base de datos de la aplicación, finalizando así el caso de uso.
Propósito:	Permitir registrar las solicitudes de Demanda Paciente a Provincia.
Precondiciones:	La Enfermera Coordinadora CCEMN debe estar autenticada en el sistema.
Poscondiciones:	Demanda de Pacientes a Provincia y actualizada la base de datos.
Referencias:	RF 5: RF 5.1, RF 15.
Prioridad:	Crítico.

Caso de Uso:	Gestionar Demanda Paciente a Provincia.
Actores:	Coordinador.
Propósito:	Permitir gestionar la información referente a las demandas de pacientes a provincia (Modificar Demanda Paciente a Provincia, Eliminar Demanda Paciente a Provincia, Insertar Cumplimiento Demanda Paciente a Provincia y Visualizar Demanda Paciente a Provincia).
Resumen:	El caso de uso se inicia cuando el Coordinador necesita gestionar las demandas de pacientes a provincia. El Coordinador debe busca primero las demandas de pacientes a provincia, pudiendo luego modificarlas, visualizarlas, eliminarlas e insertarle cumplimiento una vez asignado el móvil para el retorno hacia su provincia, finalizando así el caso de uso.
Precondiciones:	El Coordinador debe estar autenticado en el sistema. Las demandas deben ser buscadas primero (Ejecutar el Caso de Uso Buscar Demanda Paciente a Provincia). Ventana de Gestionar Demanda Paciente a Provincia abierta y lista.

Poscondiciones:	Demanda Paciente a Provincia modificada y actualizada la base de datos. Demanda Paciente a Provincia visualizada. Demanda Paciente a Provincia eliminada y actualizada la base de datos. Demanda Paciente a Provincia con cumplimiento insertado y actualizada la base de datos.
Referencias:	RF 5, RF 6, RF 15.
Prioridad:	Crítico.

Caso de Uso:	Buscar Demanda Paciente a Provincia.
Actores:	Coordinador.
Propósito:	Buscar Demanda Paciente a Provincia.
Resumen:	El caso de uso se inicia cuando el Coordinador necesita buscar las demandas de paciente a provincia, este escoge la opción “Buscar Demandas Pacientes a Provincias”, introduce el criterio de búsqueda y el sistema muestra un listado con las demandas de paciente a provincia, finalizando así el caso de uso.
Precondiciones:	El Coordinador debe estar autenticado en el sistema. Demandas de paciente a provincia insertadas.
Poscondiciones:	Listado de Demandas de Paciente a Provincia.
Referencias:	RF 6, RF 15.
Prioridad:	Crítico.

Caso de Uso:	Gestionar Paciente Demanda Traslado Interprovincial.
Actores:	Trabajador.
Propósito:	Permitir gestionar la información referente a los pacientes de las demandas de traslado interprovincial (Insertar Paciente Demanda Traslado Interprovincial, Modificar Paciente Demanda Traslado Interprovincial, listar y eliminar Paciente Traslado Interprovincial).
Resumen:	El caso de uso se inicia cuando el Trabajador necesita gestionar los pacientes de las demandas de traslado interprovincial. El Trabajador busca primero las demandas de traslado interprovincial, dando la posibilidad de seleccionar la opción “Gestionar Pacientes”, pudiendo luego insertar, modificar y eliminar los pacientes, finalizando así el caso de uso.
Precondiciones:	Trabajador debe estar autenticado en el sistema. Buscar demandas de traslado interprovincial. (Ejecutar el Caso de Uso Buscar Demandas Traslado interprovincial) Ventana de Gestionar Demanda Traslado Interprovincial abierta y lista.
Poscondiciones:	Listado de Pacientes Demanda Traslado Interprovincial. Paciente Demanda Traslado Interprovincial insertado y actualizada la base de datos. Paciente Demanda Traslado Interprovincial modificado y actualizada la base de datos. Paciente Demanda Traslado Interprovincial eliminado y actualizada la base de datos.
Referencias:	RF 2, RF 3, RF 4, RF 15.
Prioridad:	Crítico.

Caso de Uso:	Gestionar Entrada.
Actores:	Coordinador.
Propósito:	Permitir gestionar la información referente a las entradas realizadas a los móviles que viajan hacia la Ciudad de La Habana una vez llegada a la misma y a sus pacientes (Insertar Entrada, Insertar Paciente Entrada, Modificar Paciente Entrada y Eliminar Paciente Entrada).
Resumen:	El caso de uso se inicia cuando el Coordinador necesita insertar una entrada a una demanda de traslado interprovincial. Este escoge la opción Efectuar Entrada de la página principal, dando la posibilidad de buscar primero las demandas de traslado interprovincial, luego de realizar la búsqueda, el sistema brinda la posibilidad de insertar la entrada, pudiendo además gestionar los pacientes de esta, en caso de ser necesario, es decir insertar un nuevo paciente a la entrada, modificar un paciente o eliminarlo en caso deseado. Luego de realizar cualquier operación se actualiza la base de datos y finalizando así el caso de uso.
Precondiciones:	Coordinador debe estar autenticado en el sistema. Buscar demandas de traslado interprovincial. (Ejecutar el Caso de Uso Buscar Demandas Traslado interprovincial) Ventana con listado de Demandas Traslado Interprovincial abierta y lista.
Poscondiciones:	Entrada insertada. Paciente de la entrada insertado. Paciente de la entrada modificado. Paciente de la entrada eliminado.
Referencias:	RF 2: RF 2.1, RF 3: RF 3.4, RF 7, RF 15.
Prioridad:	Crítico.

Caso de Uso:	Gestionar Enfermedad Trazadora.
Actores:	Administrador.
Propósito:	Permitir gestionar el nomenclador Enfermedades Trazadoras, es decir insertar una nueva enfermedad trazadora a la aplicación, modificar una ya existente, así como listarlas y eliminarlas.
Resumen:	El caso de uso se inicia cuando el Administrador del sistema necesita insertar, listar, eliminar y modificar las enfermedades trazadoras, este escoge la opción Gestionar Nomenclador de la página principal, luego la aplicación brinda una interfaz con un buscador para seleccionar el nomenclador al cual desea gestionar (insertar, eliminar o modificar), después de realizada la búsqueda, se ofrece un listado con las enfermedades existentes, dando la posibilidad de realizar la operación deseada. Se puede insertar una nueva enfermedad, eliminar y modificar una ya existente, el sistema actualiza la base de datos y finaliza así el caso de uso.
Precondiciones:	Administrador debe estar autenticado en el sistema.
Poscondiciones:	Enfermedad Trazadora insertada. Enfermedad Trazadora modificada. Enfermedad Trazadora eliminada.
Referencias:	RF 11, RF 15.
Prioridad:	Crítico.

Caso de Uso:	Gestionar Usuario.
Actores:	Administrador.
Propósito:	Permitir gestionar los usuarios de la aplicación, es decir adicionar un nuevo usuario a la aplicación una vez buscado en el Registro de Ciudadano, listar los usuarios de la aplicación, eliminar el usuario, asignarle un rol a este, así como eliminarlo.
Resumen:	El caso de uso se inicia cuando el Administrador del sistema desea gestionar los usuarios de la aplicación. El administrador selecciona la opción Gestionar Usuario, el sistema ofrece la posibilidad de buscar usuarios en el Registro de Ciudadanos, dando la posibilidad de adicionar un usuario al sistema. Para eliminar un usuario, asignarle un rol o eliminarlo, el administrador escoge la opción Listar Usuarios de la página principal, este brinda una interfaz con los usuarios existentes en la aplicación, dando la posibilidad de eliminar un usuario, eliminarle un rol a este o asignarle uno nuevo. Al realizar cualquier operación se actualiza la base de datos y finaliza así el caso de uso.
Precondiciones:	Administrador debe estar autenticado en el sistema.
Poscondiciones:	Rol eliminado. Rol insertado. Usuario insertado. Usuario eliminado. Búsqueda de usuarios del registro de ciudadanos realizada. Listado de usuarios de la aplicación.
Referencias:	RF 10, RF 15.
Prioridad:	Crítico.

Caso de Uso:	Generar Parte Traslado Provincia.
Actores:	Médico Jefe de Guardia.
Resumen:	El caso de uso se inicia cuando el Médica Jefe de Guardia solicita generar el parte para el control de los traslados a provincia realizados por el Coordinador de Emergencia Médica Nacional, este escoge la opción deseada, finalizando así el caso de uso.
Propósito:	Permitir generar el Parte Traslado Provincia y de Traslados Interprovinciales para el control de las actividades realizadas en la institución.
Precondiciones:	Demandas de traslado interprovincial insertadas. Demandas de paciente a provincia insertadas. Información acerca de los móviles que viajan de una provincia a otra registrados, así como el control de la entrada de los móviles a las provincias.
Poscondiciones:	Reporte Parte Traslado Provincia para el control del trabajo en la institución generado (posibilidad de imprimir). Reporte Parte Traslado Interprovincial para el control del trabajo en la institución generado (posibilidad de imprimir).
Referencias:	RF 17, RF 15.
Prioridad:	Secundario.

En este capítulo, después de realizar un estudio sobre la situación actual en el Centro Coordinador de Emergencia Médica Nacional se propone realizar una aplicación informática que mejore el funcionamiento de dicha institución. Para ello se obtuvo el modelo del negocio, definiéndose actores, trabajadores y procesos en el mismo, así como los requerimientos funcionales y no funcionales del sistema basados en las necesidades del cliente.

Capítulo III: Diseño del Sistema.

En este capítulo, se describen los elementos más importantes correspondientes a la etapa de diseño del sistema, generándose los artefactos necesarios que contribuyen a la implementación del mismo. Se mostrarán los diagramas de clases y secuencia correspondientes al diseño de los casos de uso más significativos.

3.1 Modelo de Diseño.

El diseño tiene como propósito formular los modelos que se centran en los requisitos no funcionales y en el dominio de la solución. Se basa en que el modelo del sistema es una colección de objetos que cooperan entre sí, donde cada objeto es una instancia de una clase en una jerarquía de clases. Además impone una estructura del sistema que se debe conservar lo más exacto posible cuando se le da forma al mismo.

Clases del Diseño: Una clase de diseño es una abstracción de una clase o construcción en la implementación del sistema.

3.1.1 Patrones.

Son modelos a seguir, surgen de la experiencia de seres humanos al tratar de solucionar problemas reales, capturan la experiencia existente y probada para promover buenas prácticas.

Patrones de Diseño.

Los patrones de diseño son soluciones simples y elegantes a problemas específicos y comunes del diseño orientado a objetos. Son soluciones basadas en la experiencia y se ha demostrado que funcionan. Facilitan la reutilización del conocimiento experto como componentes de diseño y mejoran así la documentación, comprensión y comunicación del diseño final.

Los patrones de diseño, facilitan el aprendizaje al programador inexperto, pudiendo establecer parejas problema-solución. Además, ayudan a especificar las interfaces, identificando los elementos claves en ellas y las relaciones existentes. **(Larman, S/A)**

GRASP: Patrones para la Asignación de Responsabilidades.

Alta cohesión.

Asignar responsabilidades de manera que la cohesión permanezca alta, es decir mantener la complejidad manejable. Cada elemento del diseño debe realizar una única labor dentro del sistema. La información que almacena una clase debe ser coherente y estar muy relacionada con la misma.

Beneficios

- Mejoran la claridad del diseño.
- Simplificación del cambio.
- Genera bajo acoplamiento.
- Facilita la reutilización.

Bajo acoplamiento.

Es asignar una responsabilidad de manera que el acoplamiento permanezca bajo. Una clase tiene bajo acoplamiento si esta no depende de demasiados otros elementos. La idea es tener las clases lo menos interrelacionadas posible, de tal forma que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión en el resto de las clases, potenciando la reutilización.

Beneficio

- No se afectan por cambio en otros componentes.
- Fáciles de entender por separado.
- Facilita la reutilización.

Controlador.

Asignar la responsabilidad de controlar el flujo de eventos del sistema a clases específicas. Esto facilita la centralización de actividades (validaciones, seguridad). Mantiene un modelo de alta cohesión debido a que el controlador delega estas actividades en otras clases.

Este patrón sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado.

Beneficio

- Mayor potencial de los componentes reutilizables.

Creador.

El patrón creador ayuda a identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases. La nueva instancia deberá ser creada por la clase que:

- Tiene la información necesaria para realizar la creación del objeto.
- Usa directamente las instancias creadas del objeto.
- Almacena o maneja varias instancias de la clase.

Beneficio

- Bajo acoplamiento.

Experto.

Este patrón es el principio básico de asignación de responsabilidades. Indica que la responsabilidad de la creación de un objeto debe recaer sobre la clase que conoce toda la información necesaria para crearlo (atributos).

Beneficios

- Conservación del encapsulamiento.
- Bajo acoplamiento.
- Alta cohesión.

3.1.2 Diagramas de Clases del Diseño.

Los diagramas de clases de diseño exponen un conjunto de interfaces, colaboraciones y sus relaciones. Se utilizan para modelar la vista de diseño estática de un sistema. Son importantes para visualizar, especificar, documentar modelos estructurales y construir sistemas ejecutables aplicando ingeniería directa e inversa. Cada caso de uso cuenta con un diagrama de clases del diseño y pueden organizarse en paquetes o subsistemas para hacer más fácil su comprensión.

Caso de Uso Insertar Demanda Traslado Interprovincial.

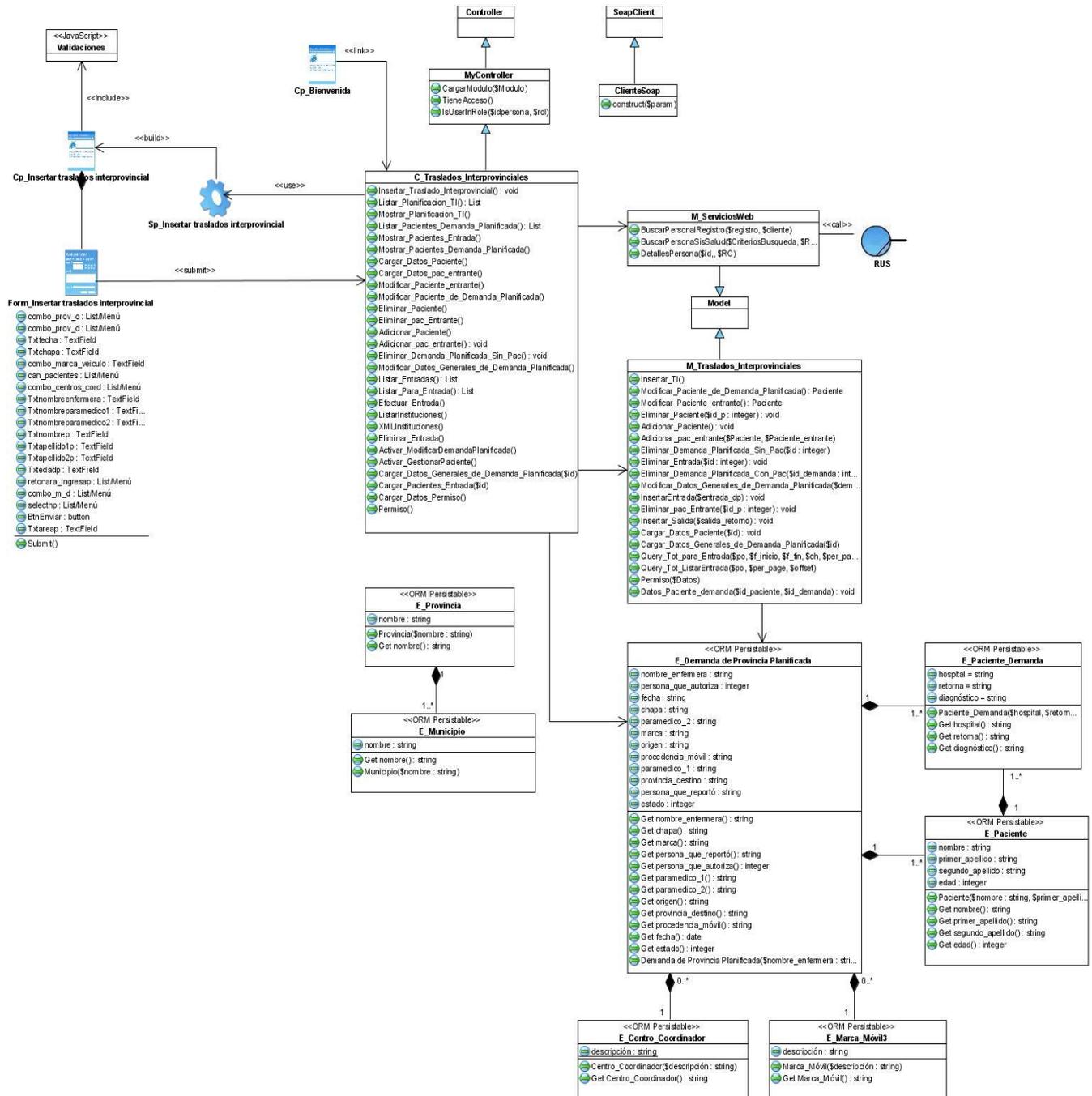


Figura 3. 1 Caso de Uso Insertar Demanda Traslado Interprovincial.

Caso de Uso Gestionar Demanda Traslado Interprovincial.

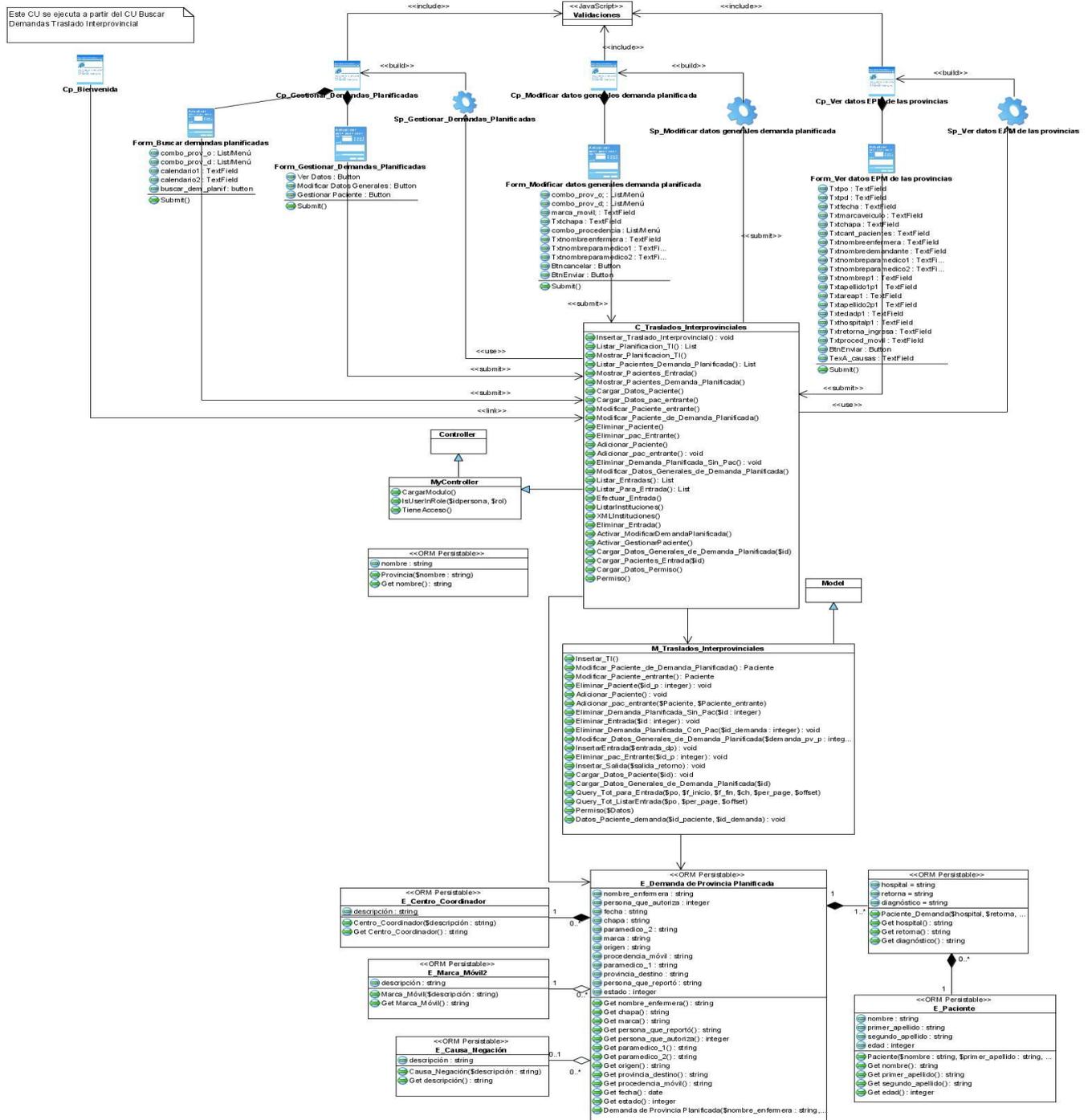


Figura 3. 2 Caso de Uso Gestionar Demanda Traslado Interprovincial.

Caso de Uso Buscar Demanda Traslado Interprovincial.

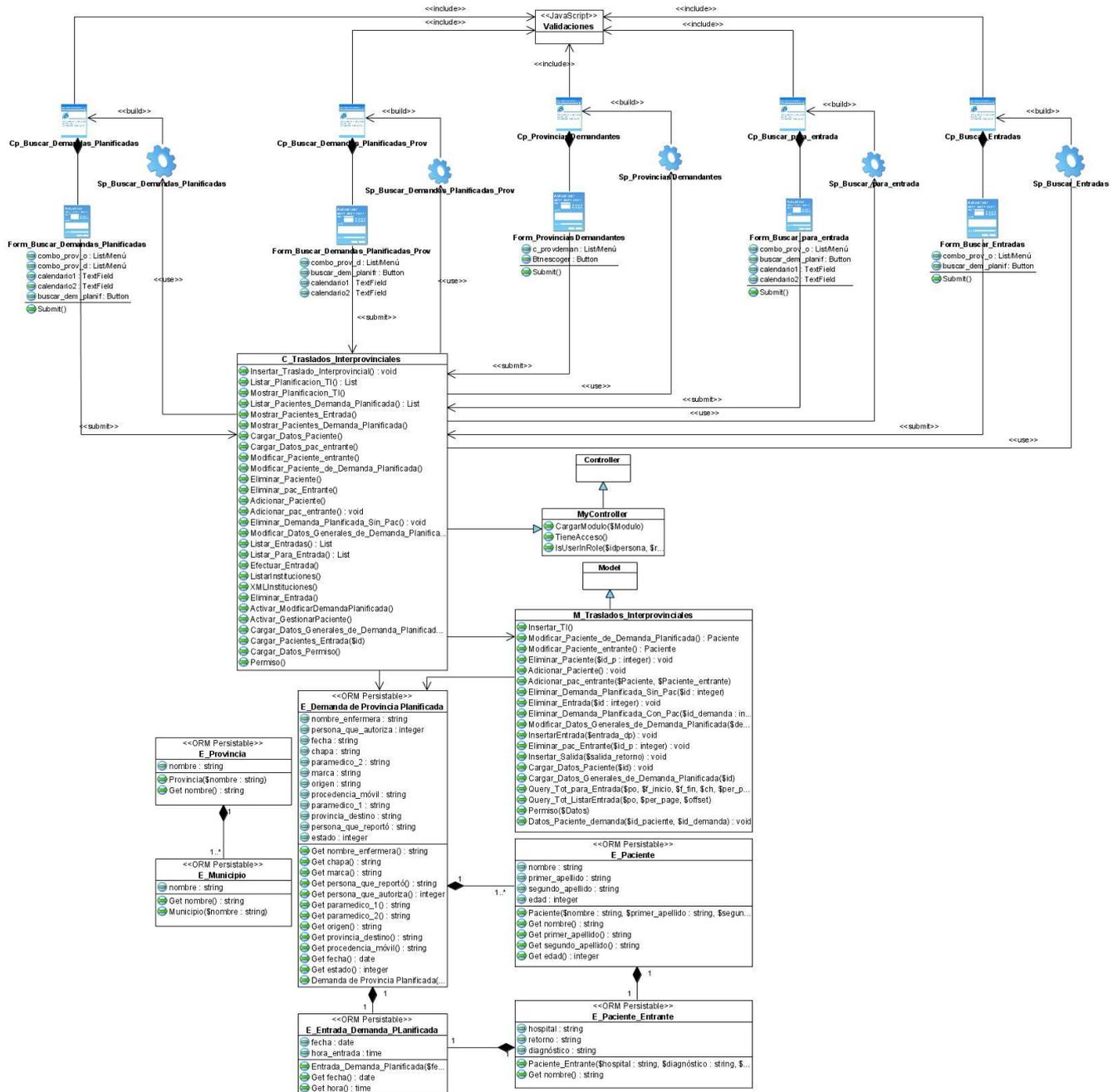


Figura 3. 3 Caso de Uso Buscar Demanda Traslado Interprovincial.

Caso de Uso Insertar Demanda Paciente a Provincia.

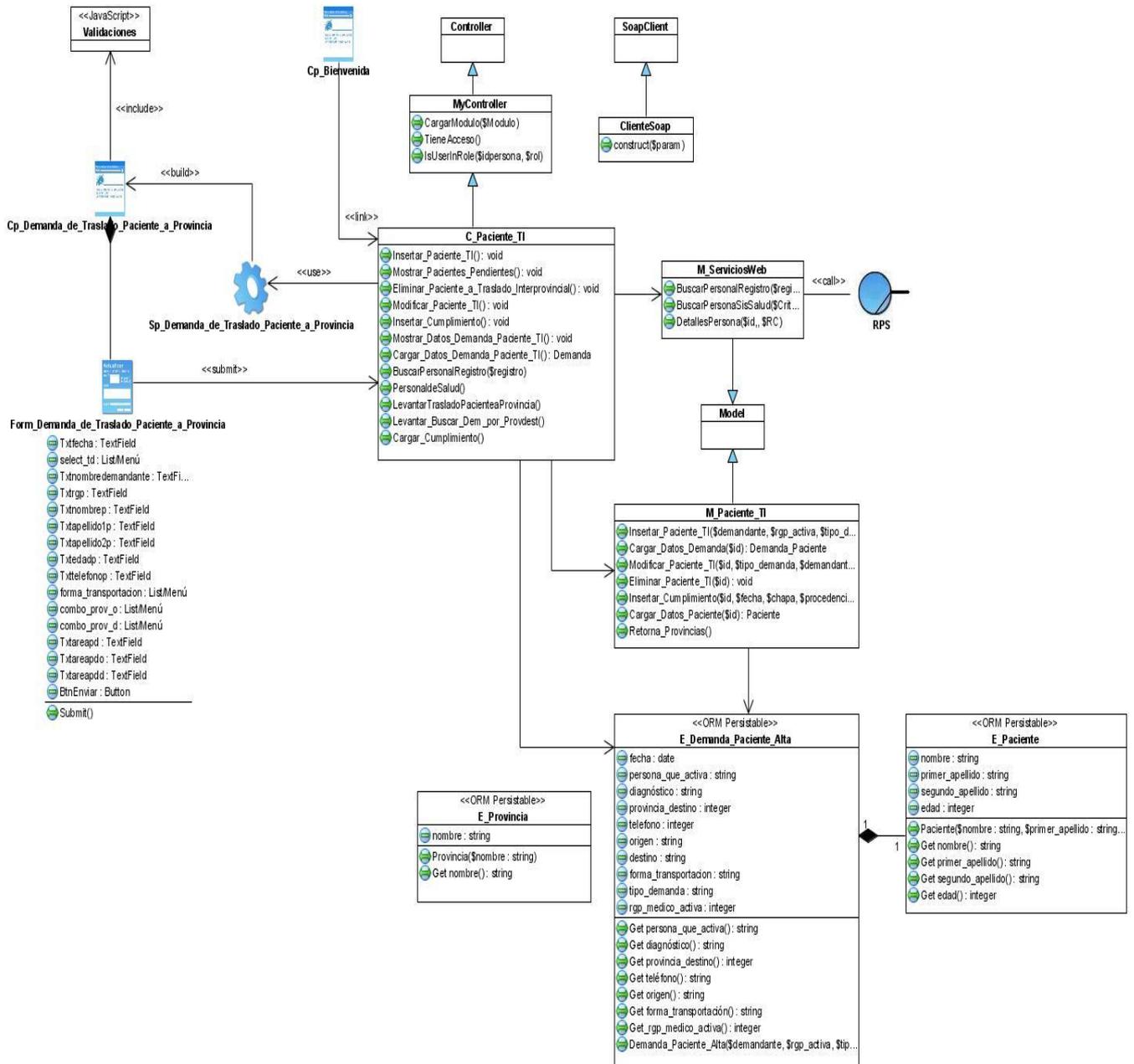


Figura 3. 4 Caso de Uso Insertar Demanda Paciente a Provincia.

Caso de Uso Gestionar Demanda Paciente a Provincia.

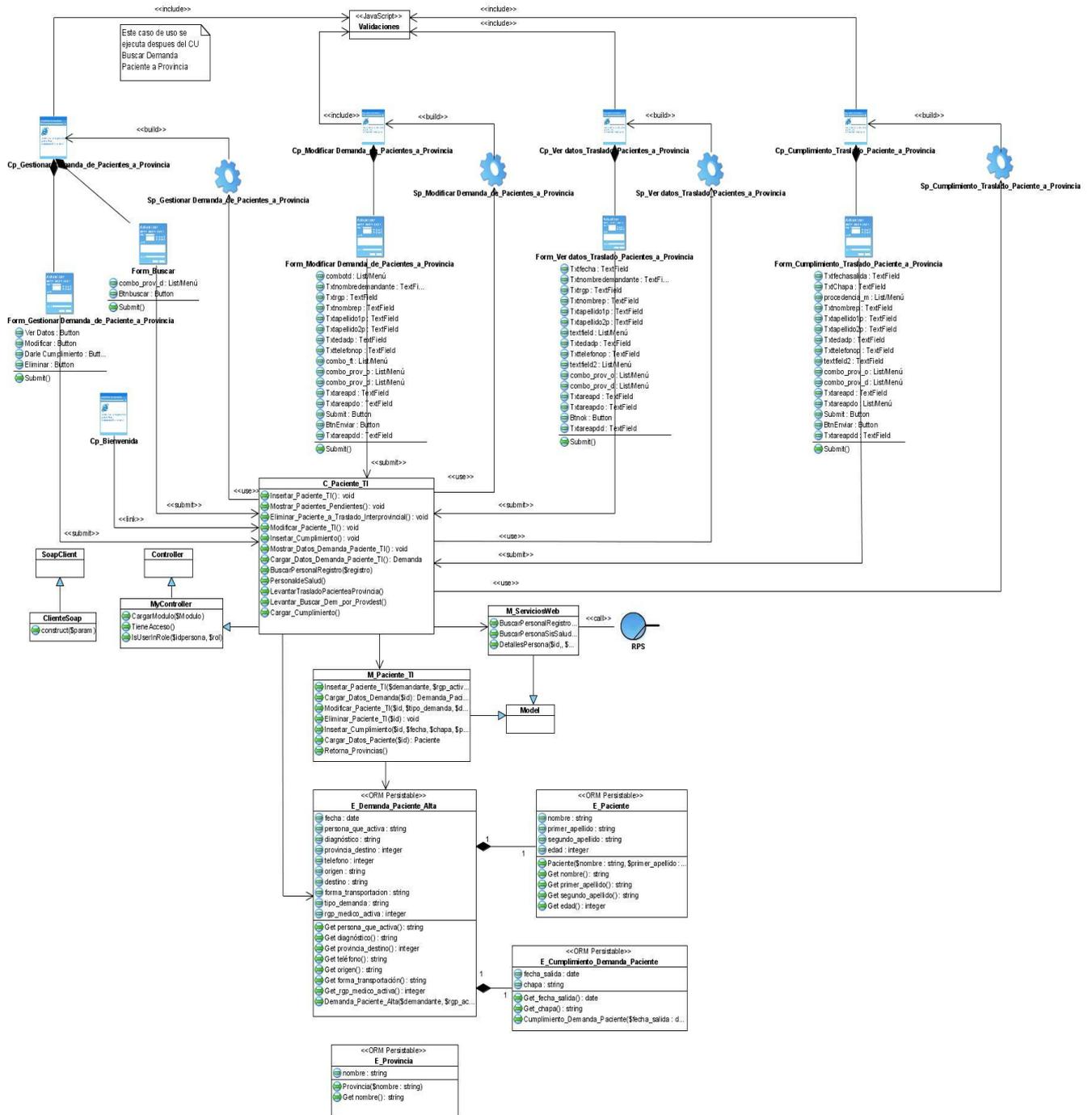


Figura 3. 5 Caso de Uso Gestionar Demanda Paciente a Provincia.

Caso de Uso Buscar Demanda Paciente a Provincia.

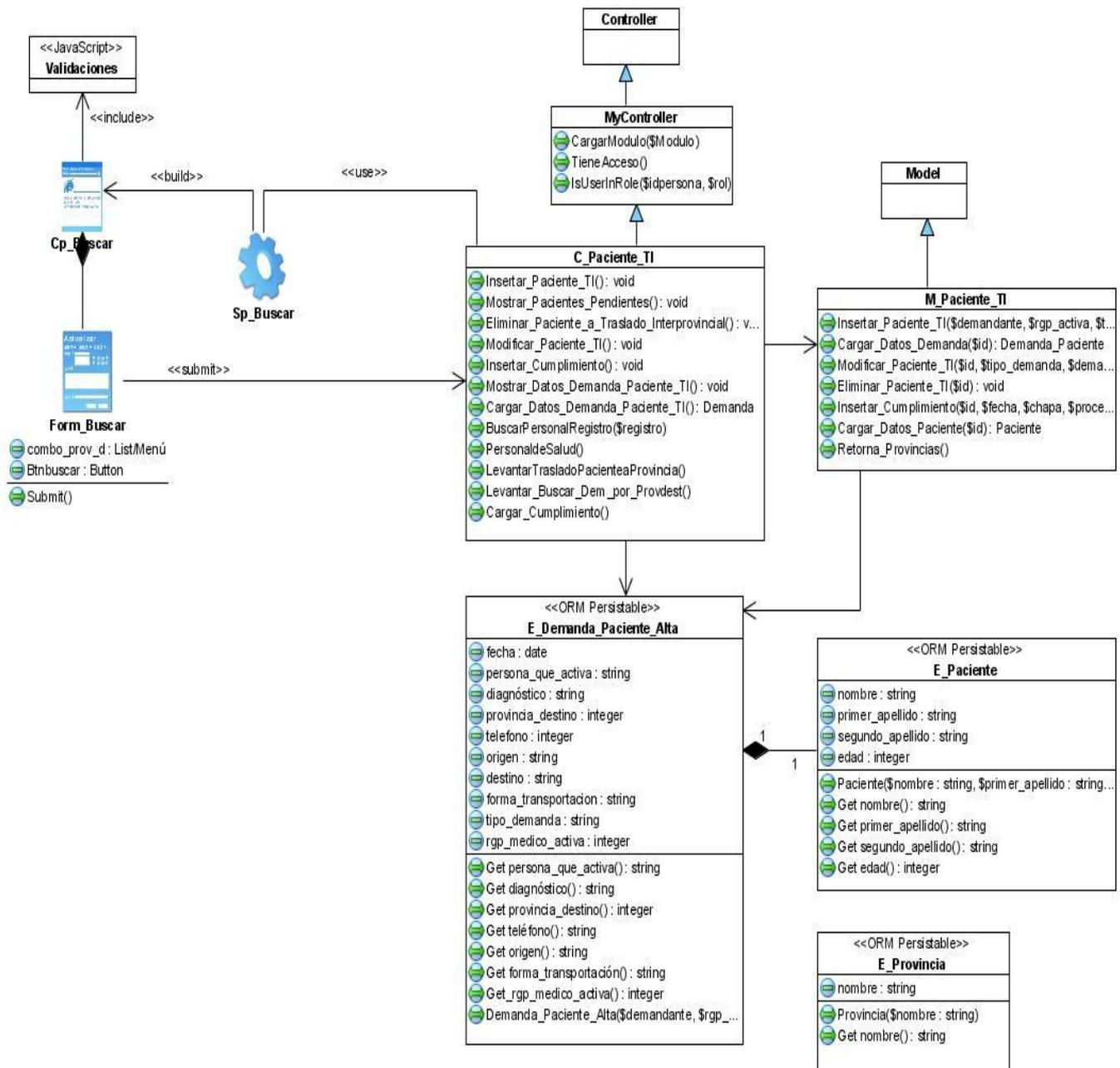


Figura 3. 6 Caso de Uso Buscar Demanda Paciente a Provincia.

Caso de Uso Gestionar Paciente Demanda Traslado Interprovincial.

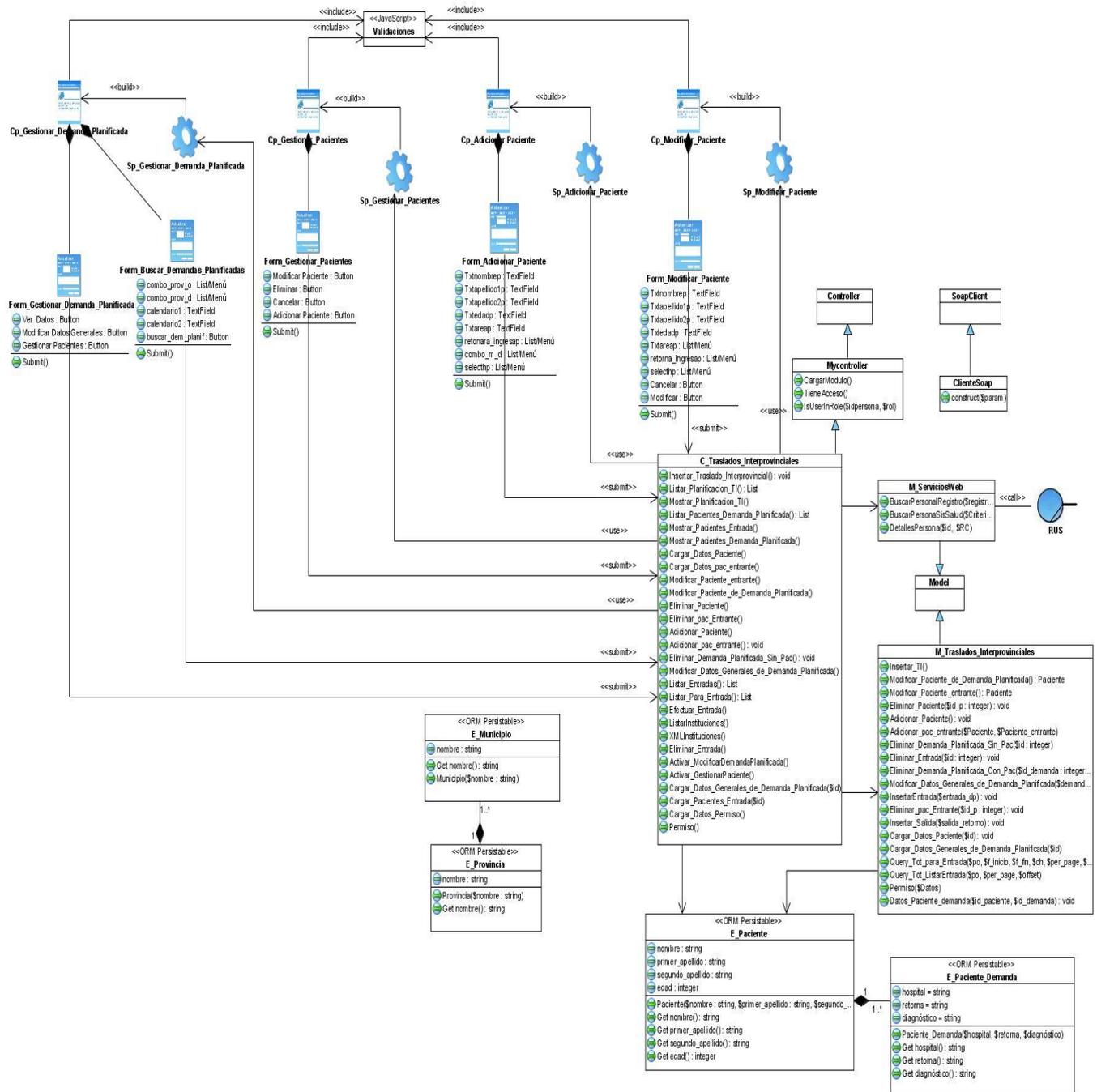


Figura 3. 7 Caso de Uso Gestionar Paciente Demanda Traslado Interprovincial.

Caso de Uso Gestionar Entrada.

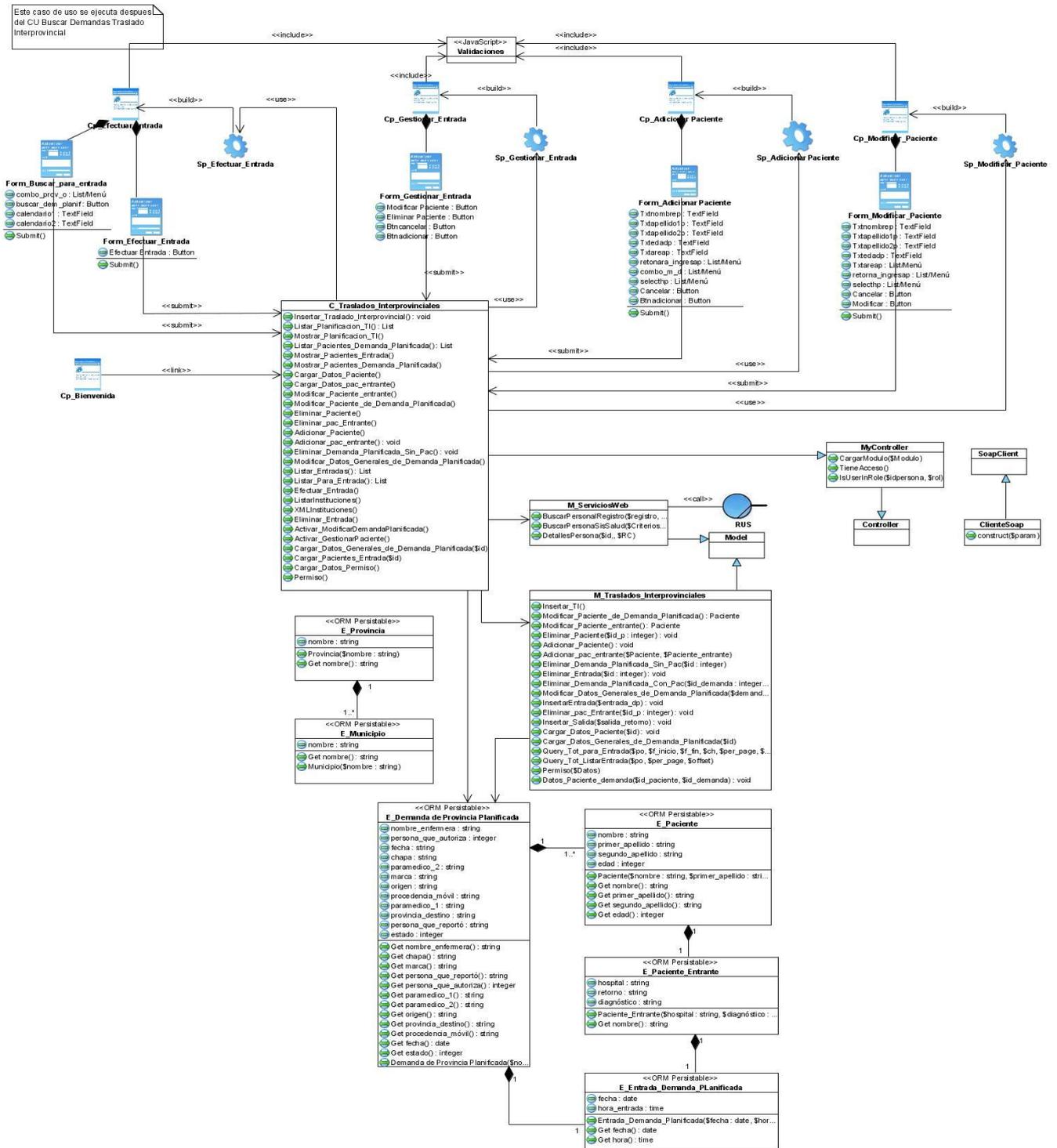


Figura 3. 8 Caso de Uso Gestionar Entrada.

Caso de Uso Gestionar Enfermedad Trazadora.

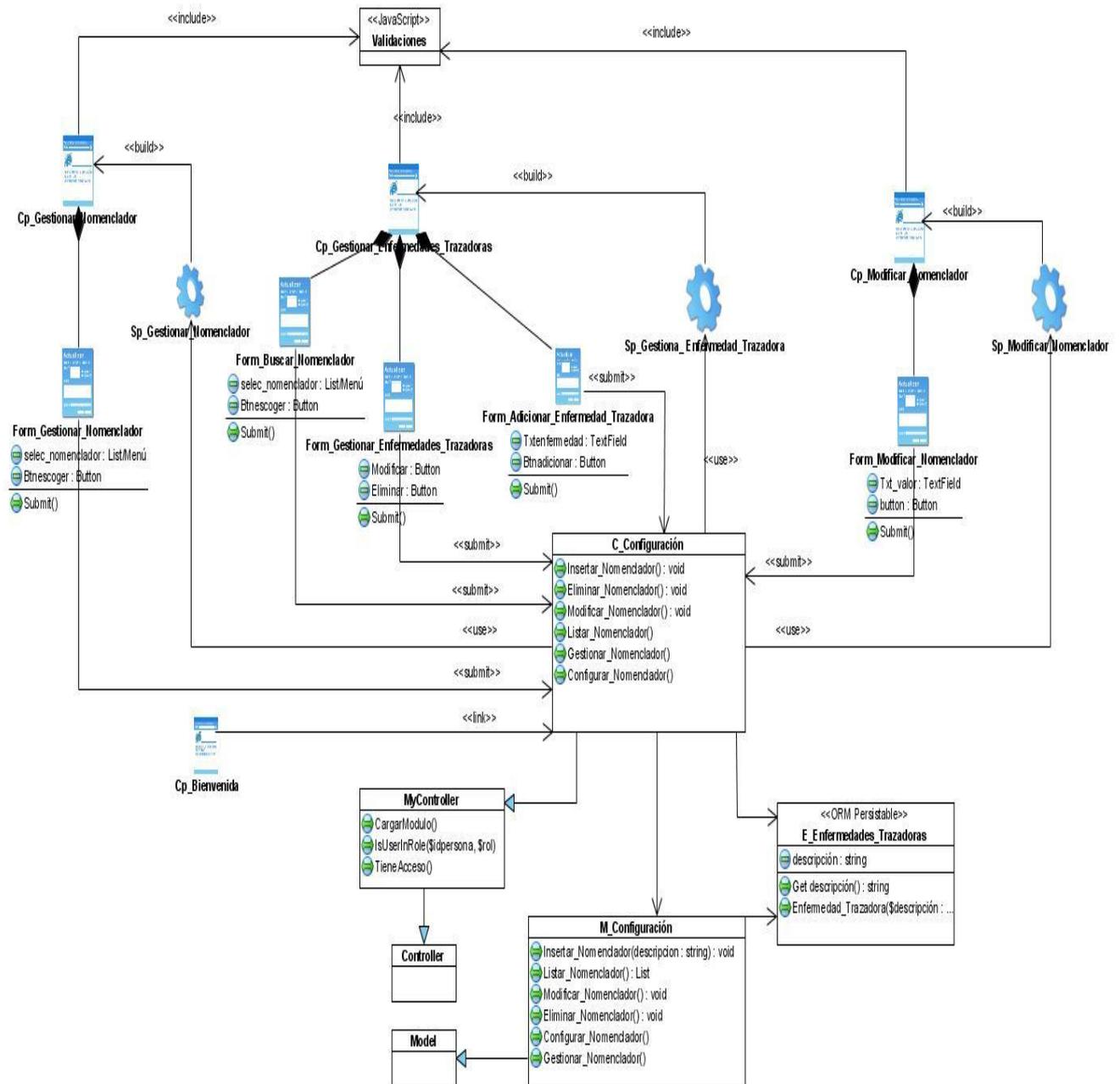


Figura 3. 9 Caso de Uso Gestionar Enfermedad Trazadora.

Caso de Uso Gestionar Usuario.

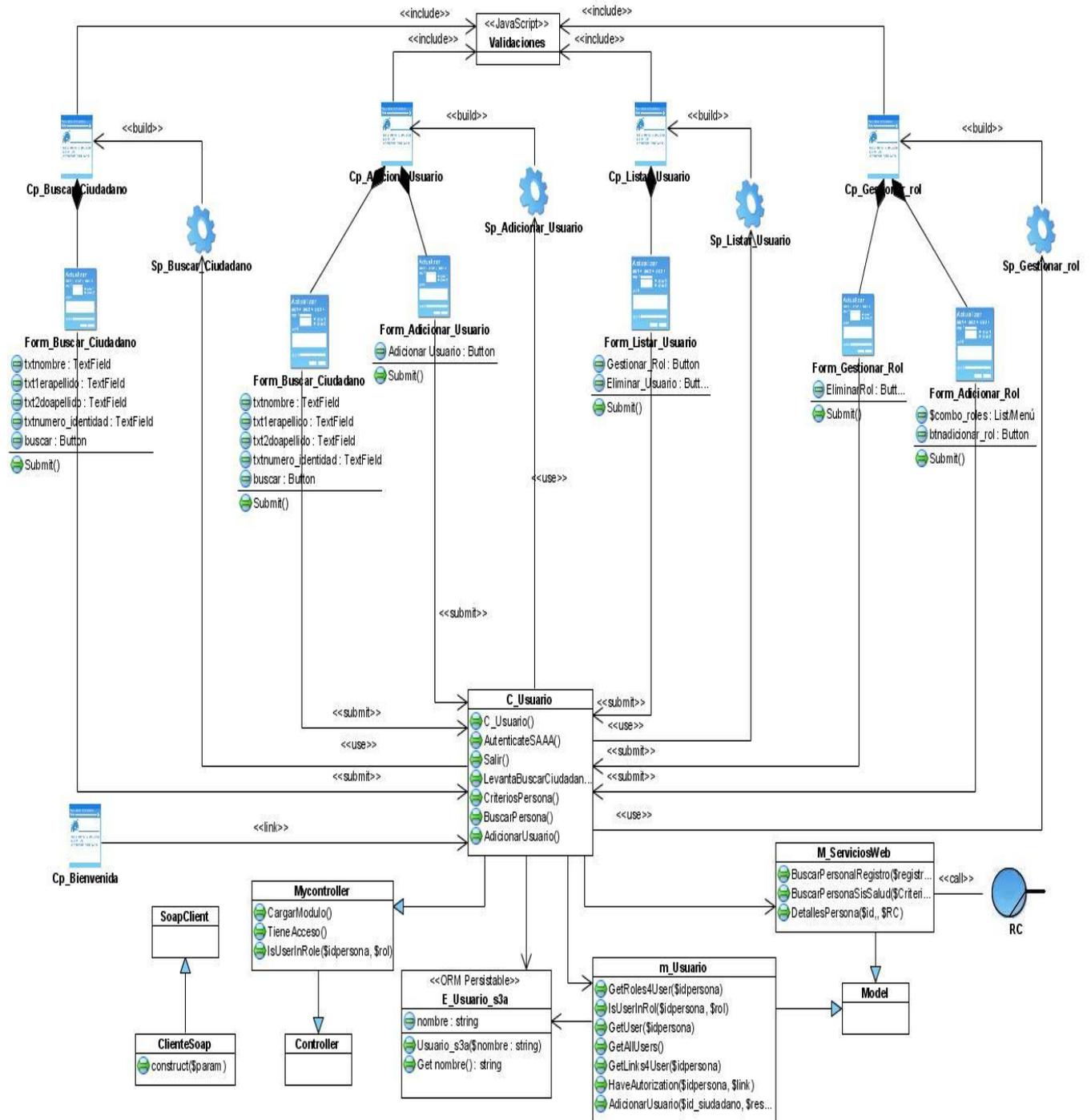


Figura 3. 10 Caso de Uso Gestionar Usuario.

3.1.3 Diagramas de Interacción.

Un diagrama de interacción es un artefacto del diseño que permite asignar las responsabilidades a cada clase, muestra: actores, objetos de las clases, eventos, orden de los eventos y pueden contener además notas y restricciones. Estos diagramas son muy útiles para visualizar, especificar, construir y documentar la dinámica entre dos objetos. Hay dos tipos de diagrama de interacción: Diagramas de Secuencia y Diagramas de Colaboración.

Los diagramas de secuencias se componen de 4 elementos fundamentales: el curso de acción, los objetos, mensajes y métodos (operaciones). Estos diagramas son el núcleo del modelo dinámico, destacan el orden temporal de los mensajes, y muestran todos los cursos alternos que pueden tomar los casos de uso durante un escenario concreto. Dan una idea cronológica de cómo ocurren las interacciones y cada objeto viene dado por una barra vertical donde el tiempo transcurre de arriba hacia abajo.

Caso de Uso Buscar Demanda Paciente a Provincia.

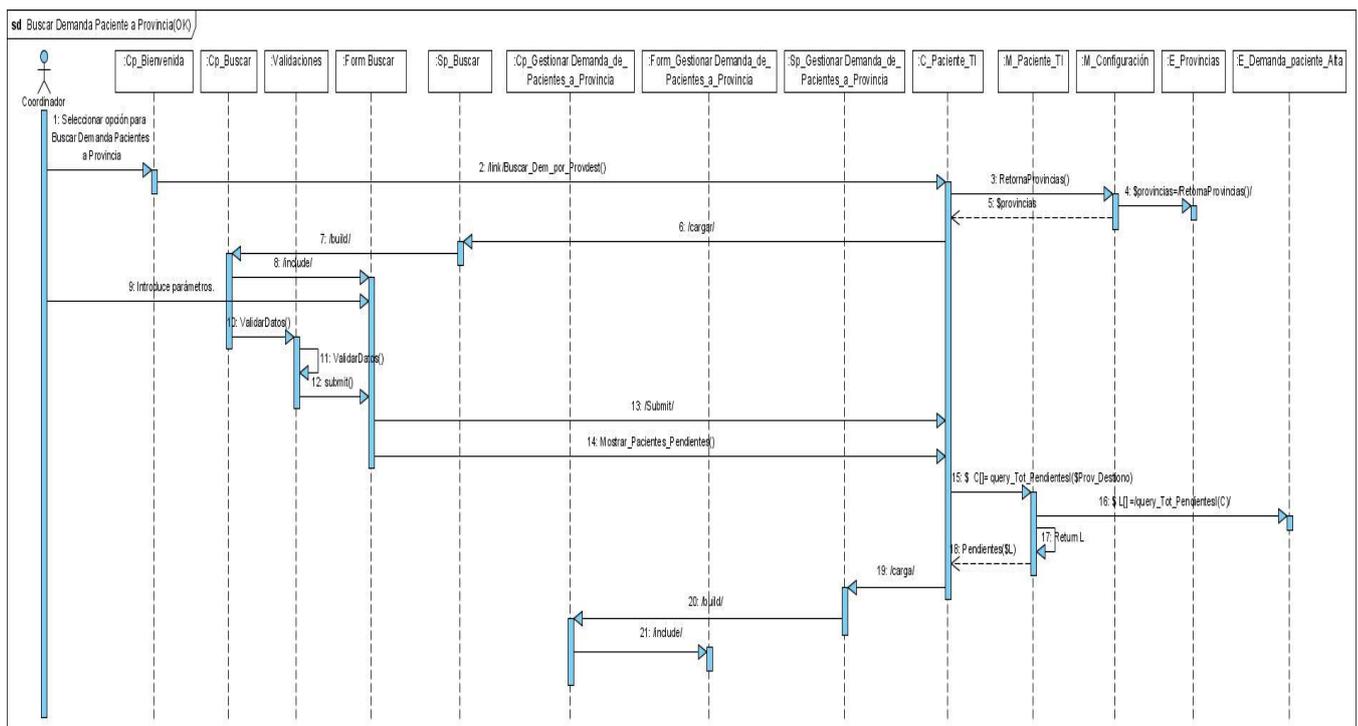


Figura 3. 11 Caso de Uso Buscar Demanda Paciente a Provincia.

Caso de Uso Insertar Demanda Paciente a Provincia.

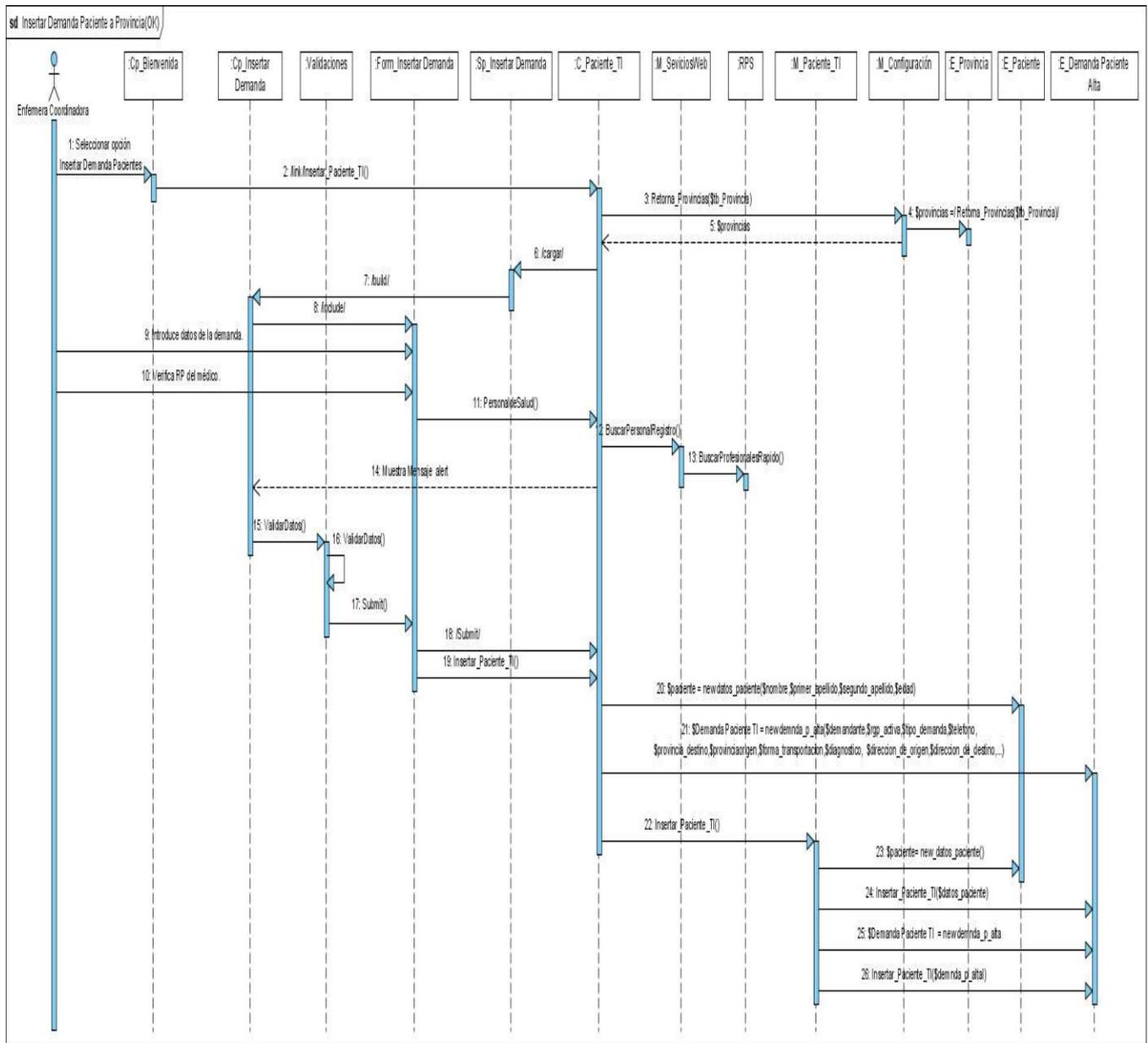


Figura 3. 12 Caso de Uso Insertar Demanda Paciente a Provincia.

Escenario Modificar Demanda Paciente a Provincia. Caso de Uso Gestionar Demanda Paciente a Provincia.

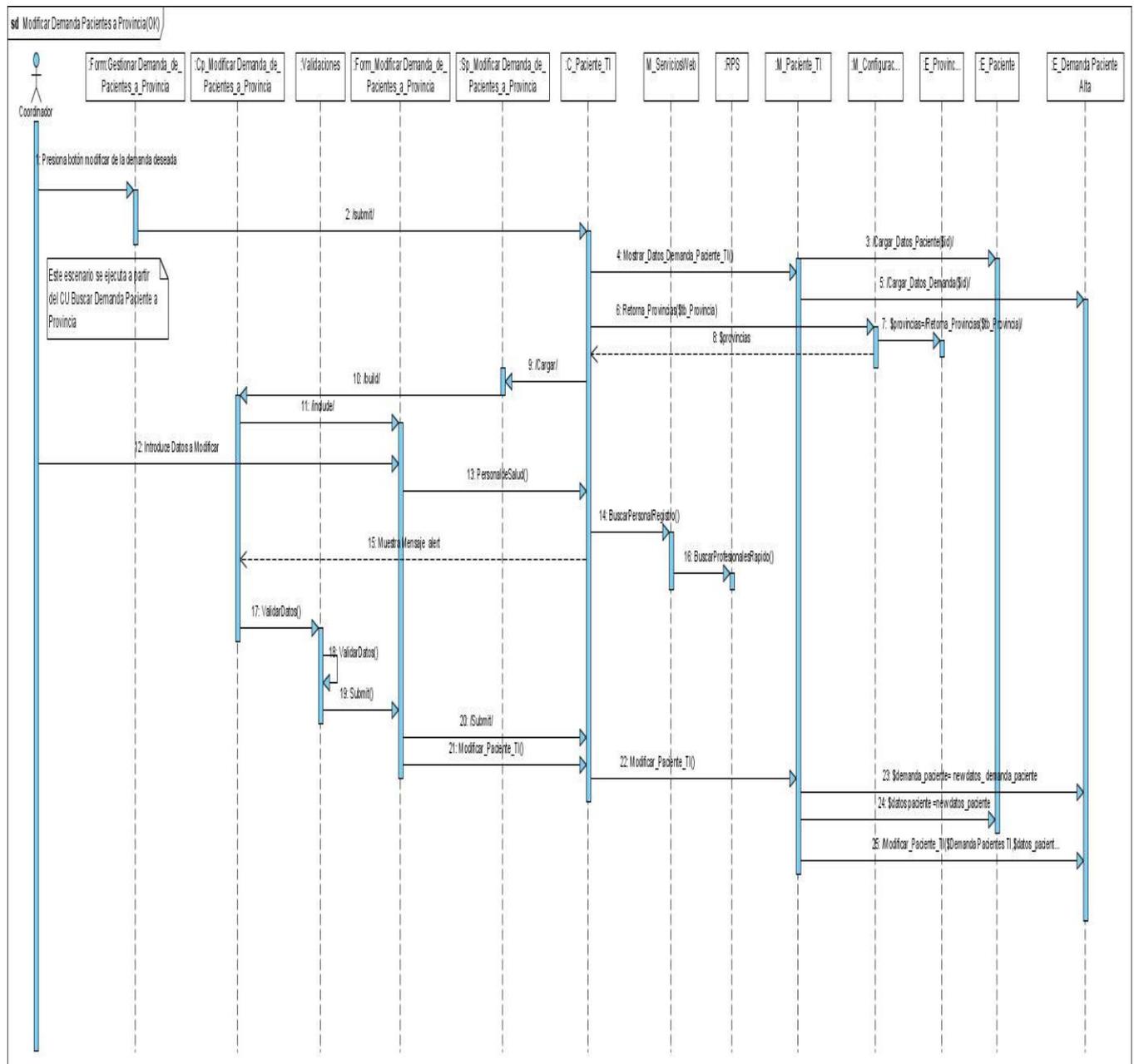


Figura 3. 13 Escenario Modificar Demanda Paciente a Provincia. Caso de Uso Gestionar Demanda Paciente a Provincia.

Escenario Buscar Demandas Traslado Interprovincial para Entradas. Caso de Uso Buscar Demanda Traslado Interprovincial.

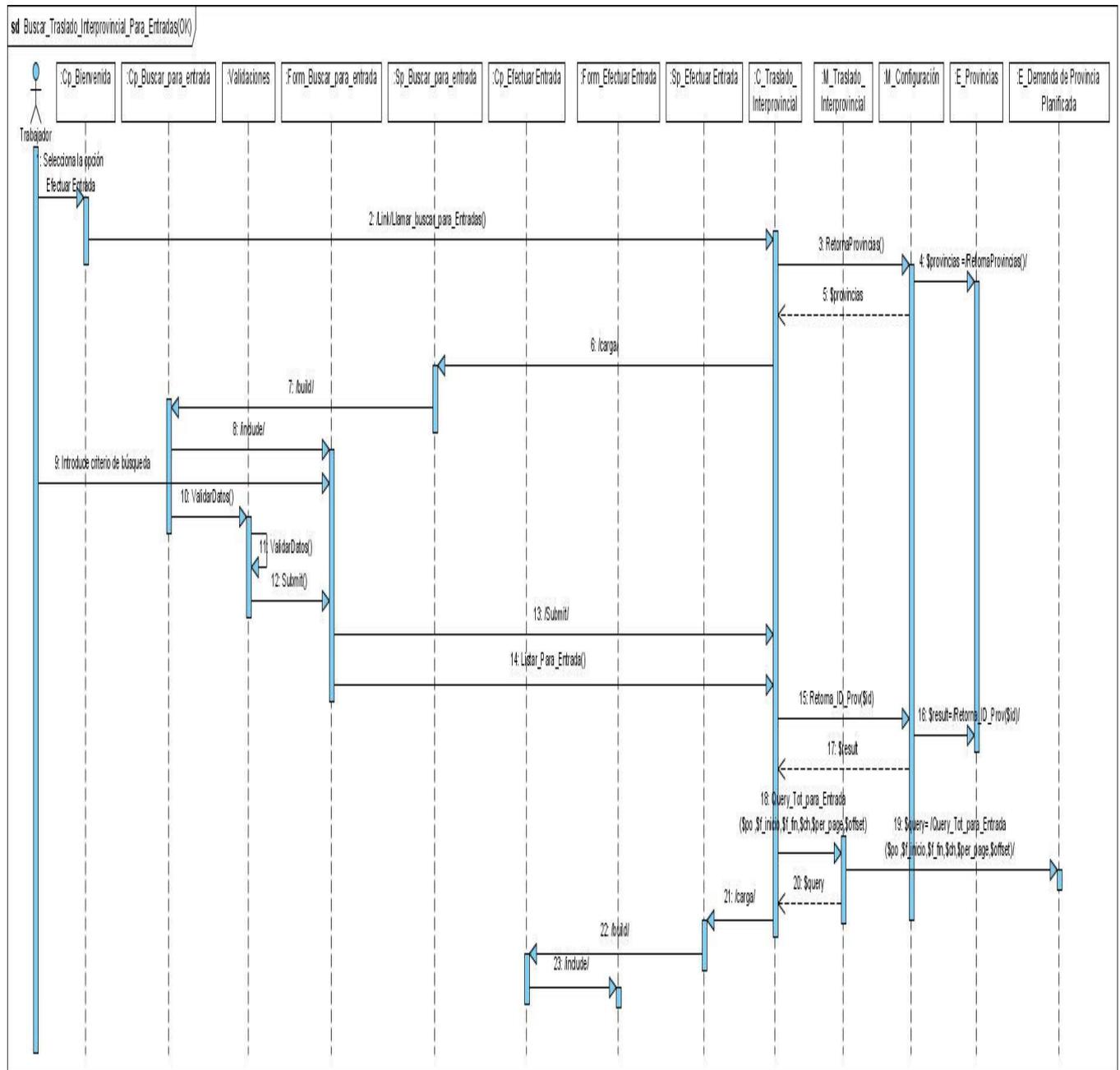


Figura 3. 14 Escenario Buscar Demandas Traslado Interprovincial para Entradas. Caso de Uso Buscar Demanda Traslado Interprovincial.

Escenario Insertar Entrada. Caso de Uso Gestionar Entrada.

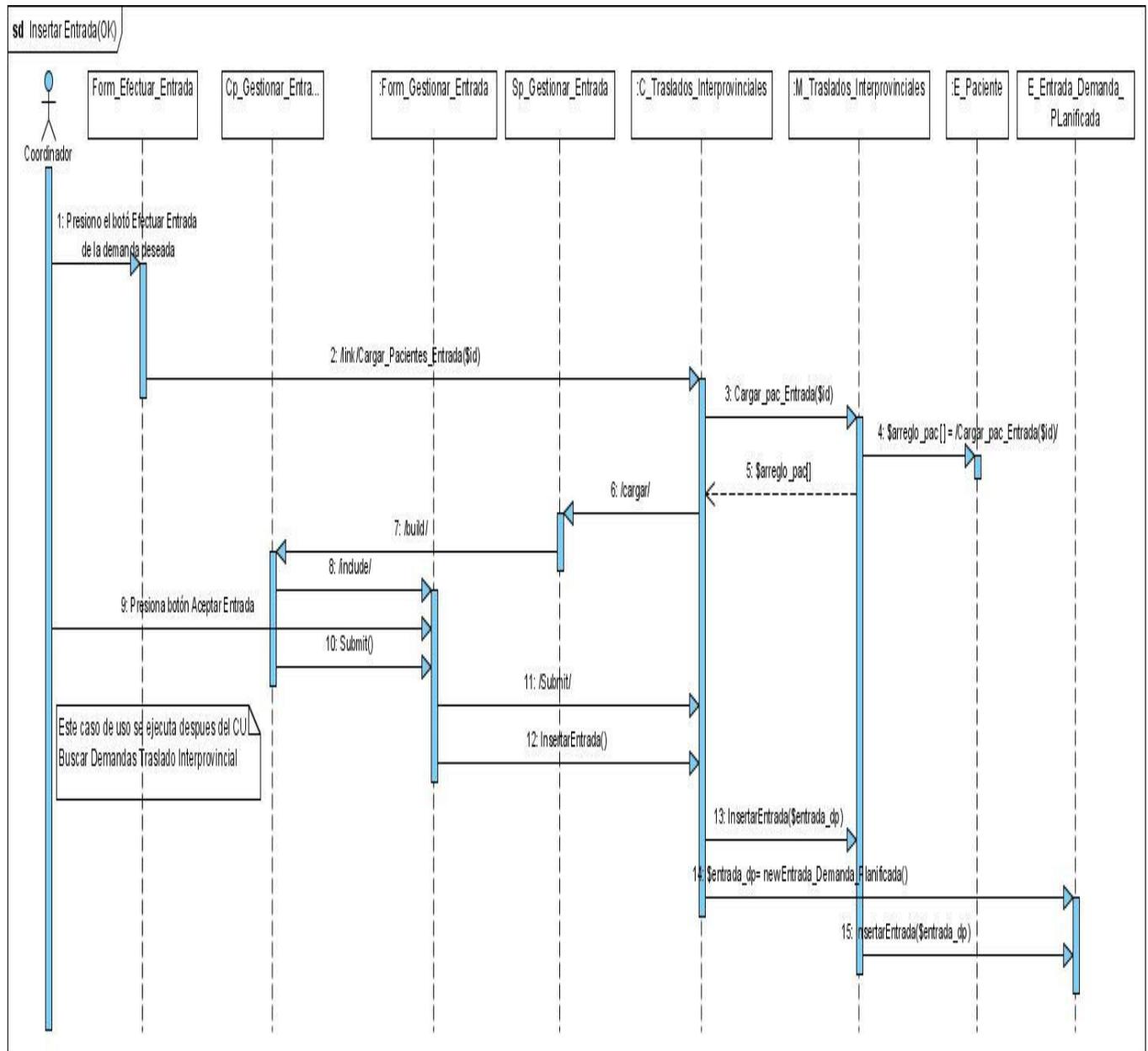


Figura 3. 15 Escenario Insertar Entrada. Caso de Uso Gestionar Entrada.

Escenario Buscar Demanda Traslado Interprovincial. Caso de Uso Buscar Demanda Traslado Interprovincial.

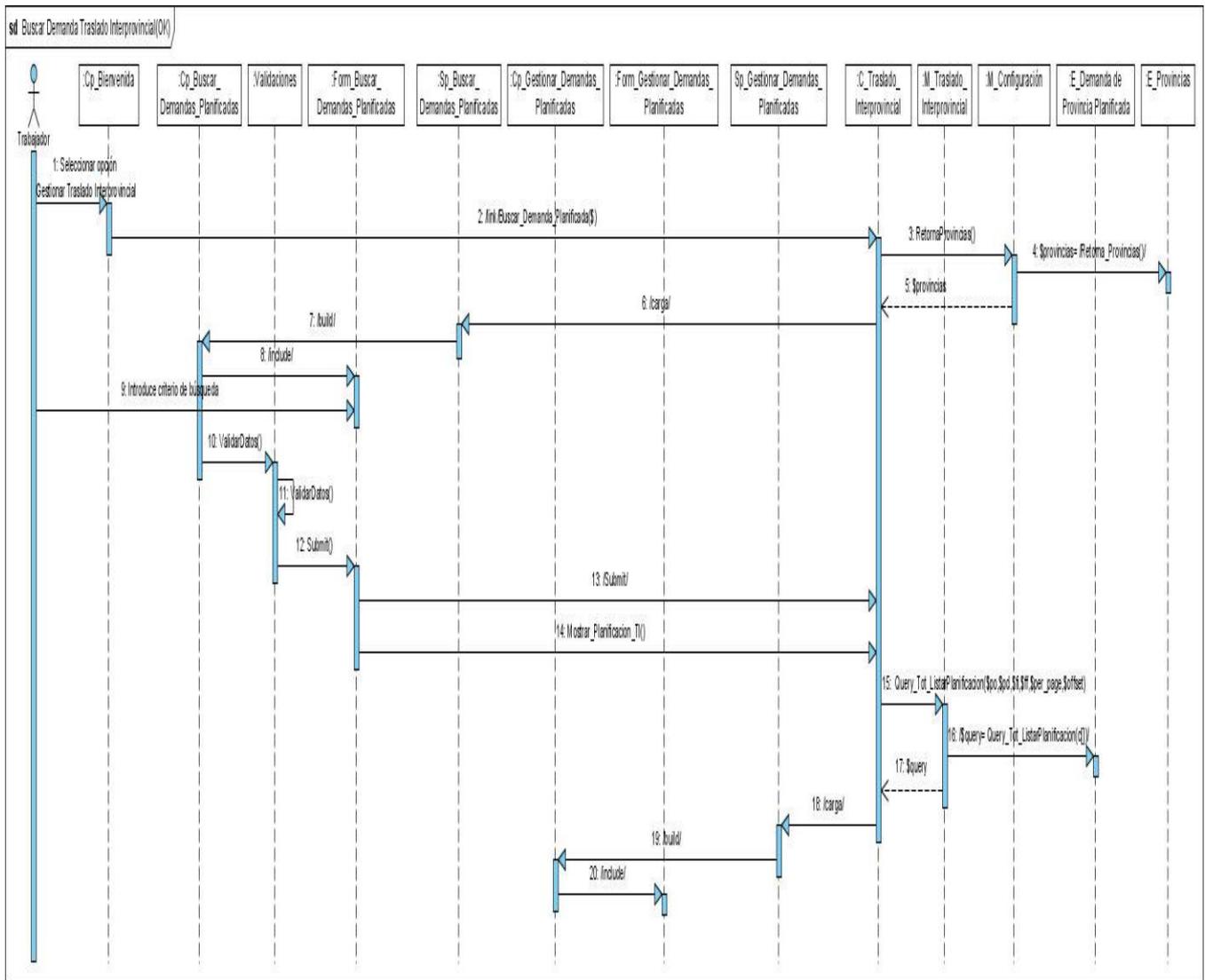


Figura 3. 16 Escenario Buscar Demanda Traslado Interprovincial. Caso de Uso Buscar Demanda Traslado Interprovincial.

Caso de Uso Insertar Demanda Traslado Interprovincial.

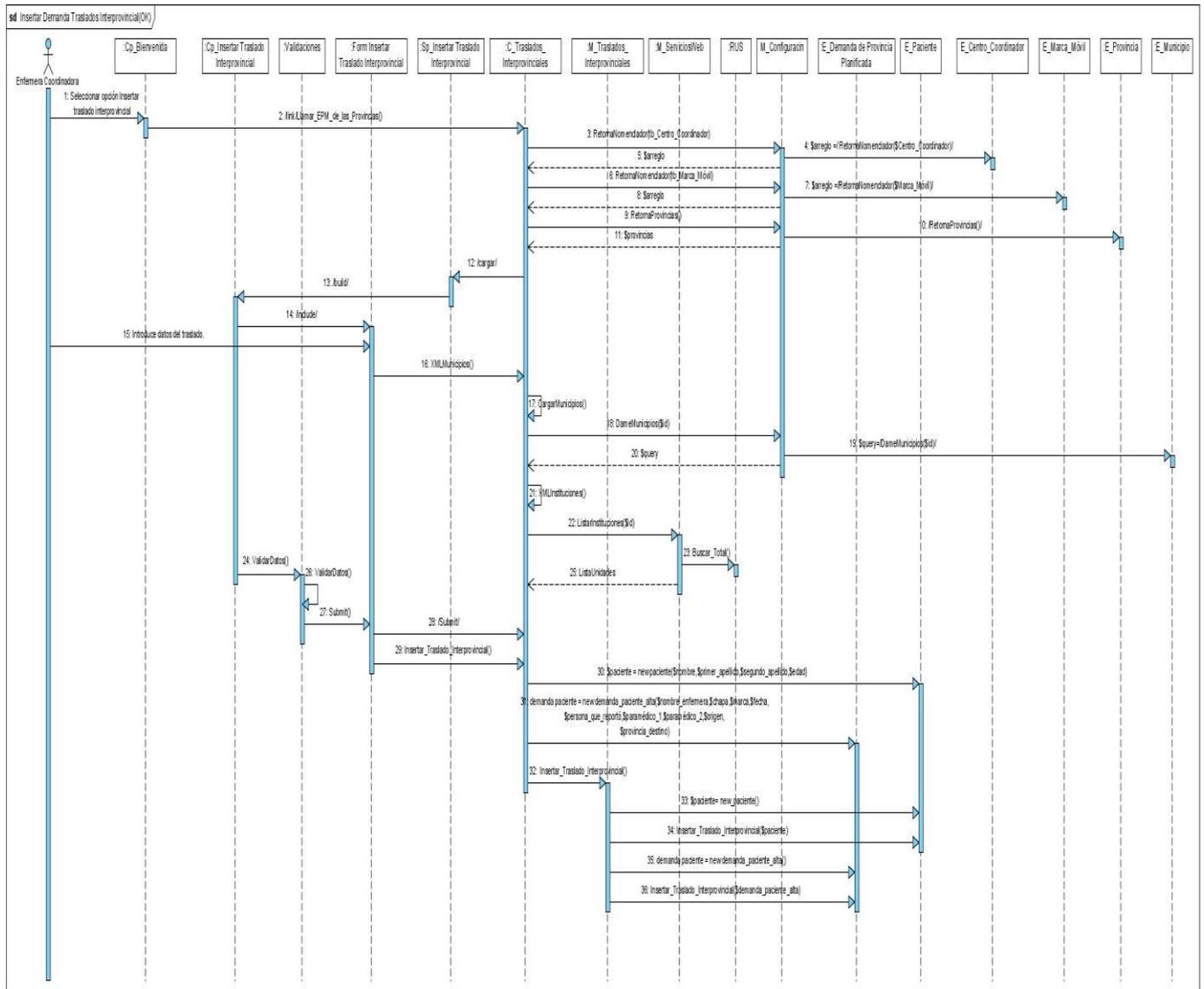


Figura 3. 17 Caso de Uso Insertar Demanda Traslado Interprovincial.

Escenario Modificar Demanda Traslado Interprovincial. Caso de Uso Insertar Demanda Traslado Interprovincial.

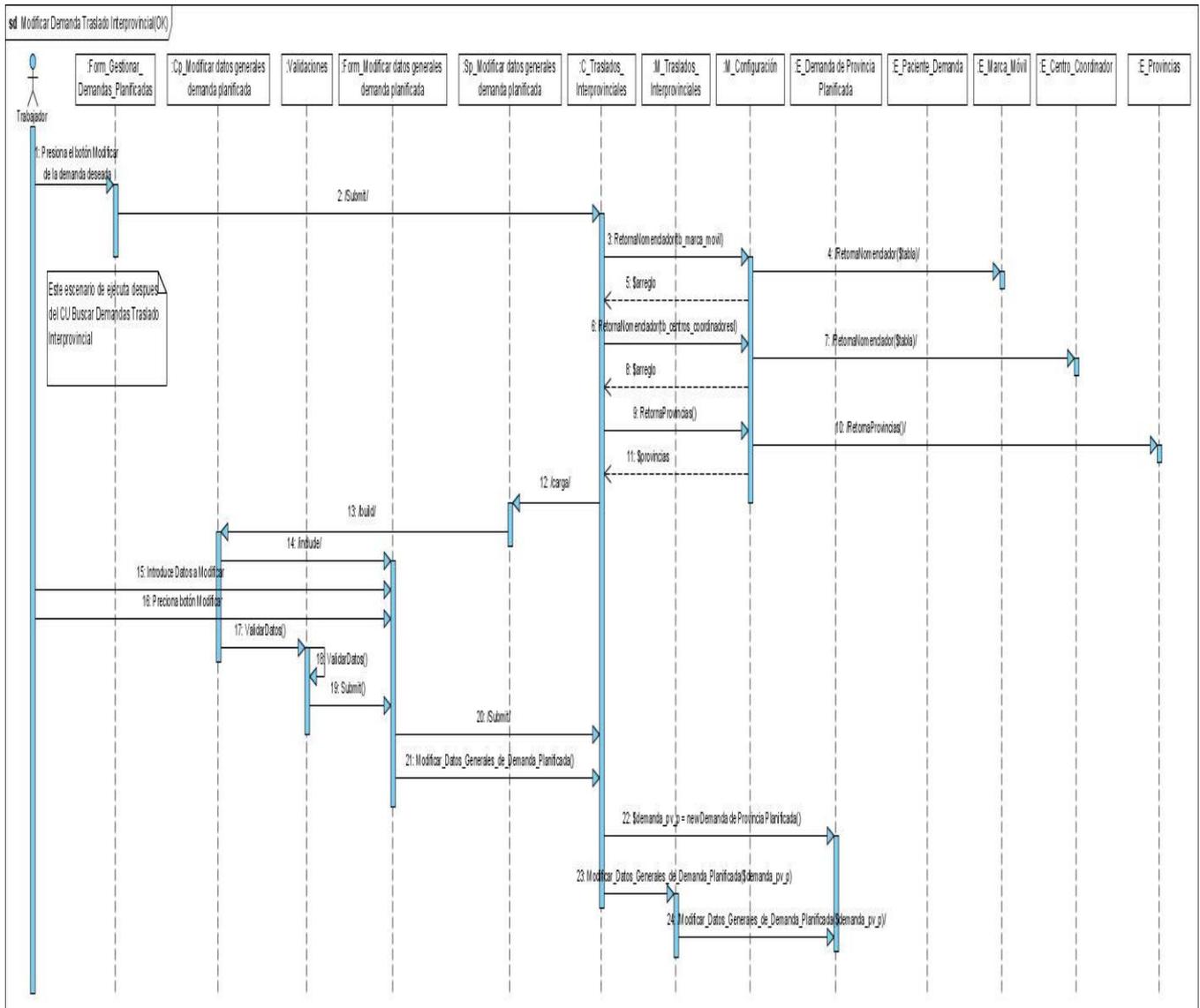


Figura 3. 18 Escenario Modificar Demanda Traslado Interprovincial. Caso de Uso Insertar Demanda Traslado Interprovincial.

Escenario Insertar Enfermedad Trazadora. Caso de Uso Gestionar Enfermedad Trazadora.

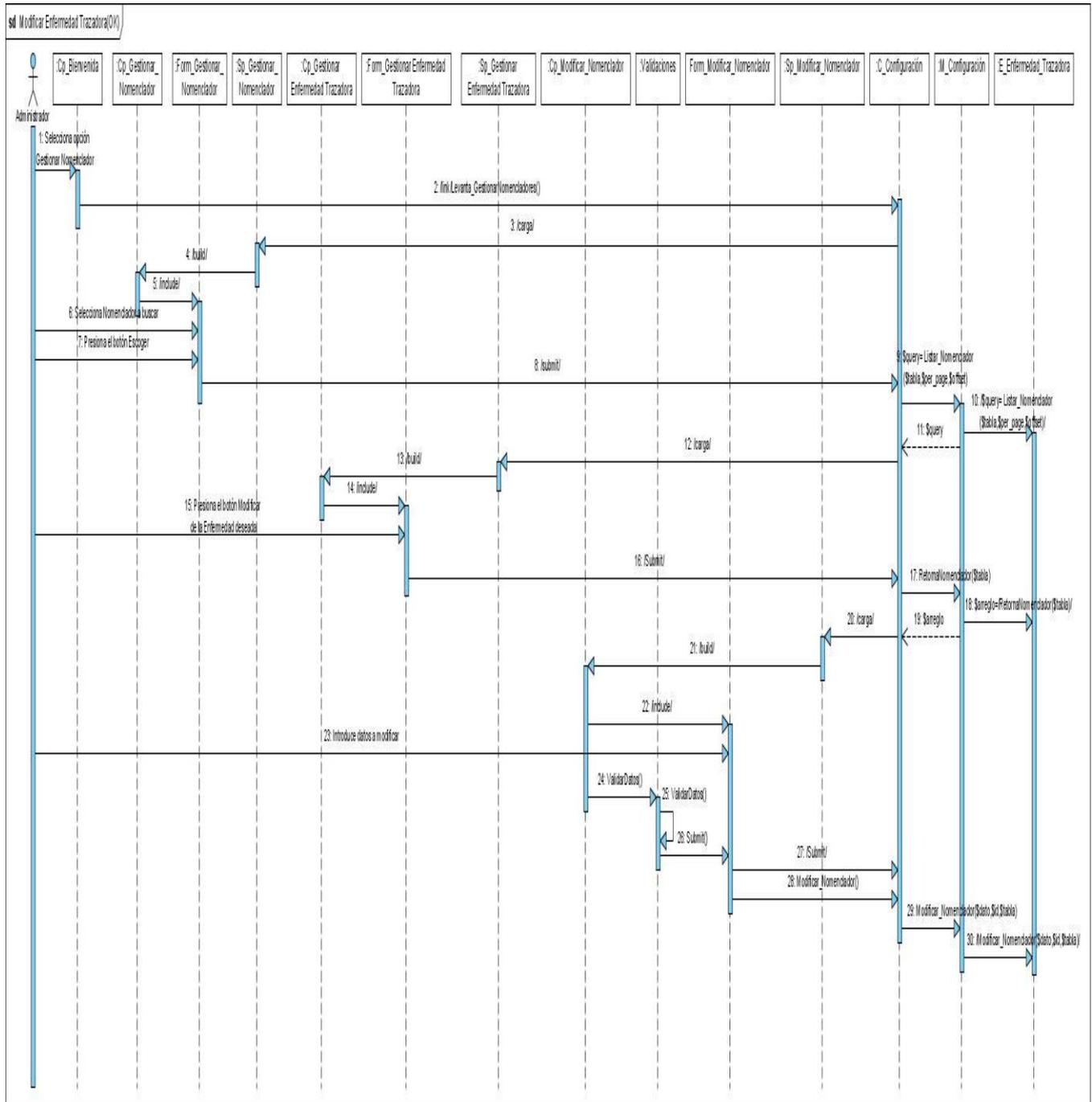


Figura 3. 19 Escenario Insertar Enfermedad Trazadora. Caso de Uso Gestionar Enfermedad Trazadora.

Caso de uso Generar Parte Traslado Interprovincial.

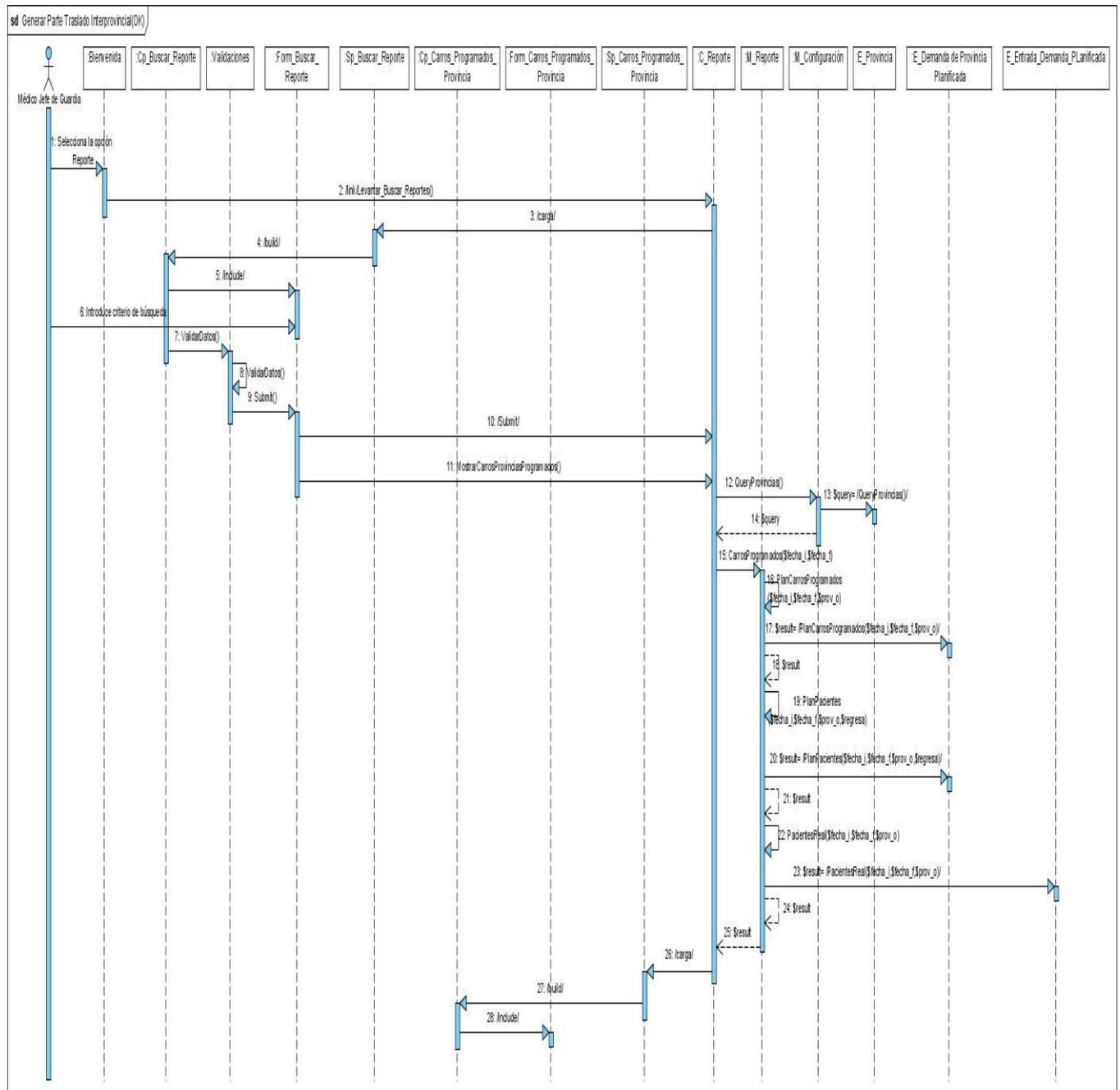


Figura 3. 20 Caso de uso Generar Parte Traslado Interprovincial.

Al concluir este capítulo se obtiene el modelo de diseño de cada caso de uso del sistema, generándose los artefactos correspondientes a este flujo de trabajo, como por ejemplo los diagramas de clases del diseño y los diagramas de interacción lográndose de esta forma materializar los requerimientos del cliente y obtener una visión de las funcionalidades del sistema.

CONCLUSIONES

La investigación realizada permitió arribar a las siguientes conclusiones:

- Se realizó un estudio detallado del proceso de gestión de la información de las emergencias médicas en el Centro Coordinador de Emergencia Médica Nacional.
- Se modelaron los flujos de trabajo propuestos por RUP: modelado del negocio, requerimientos análisis y diseño, obteniéndose los artefactos que se generan en cada uno de estos flujos.
- Se realizó la integración con los siguientes componentes de SISalud; Registro de Unidades de Salud, Registro del Personal de Salud, Registro de Ciudadanos, Registro de Ubicación y el SAAA para la autenticación, así como con componentes del Sistema Informatizado para el Centro Nacional de Urgencias Médicas.

RECOMENDACIONES

Una vez concluida la investigación, se realizan las siguientes recomendaciones:

- Identificar nuevas funcionalidades que respondan a las necesidades del cliente para versiones posteriores del producto.
- Diseñar e implementar funcionalidades para el control de las claves en el seguimiento de los móviles.
- Valorar la posibilidad de integrar la aplicación con sistemas de la facultad o del país que brinden información acerca de las camas y los ventiladores en las terapias hospitalarias.
- El uso de sistemas de ubicación (GPS) para un mejor control de los móviles que se encuentran trabajando.

REFERENCIAS BIBLIOGRÁFICAS

1. **Aramayo, Cristian. S/A.** MySQL. [En línea] S/A.
http://www.salnet.com.ar/inv_mysql/pag01_intro.htm#que_es_mysql.
2. **Ariza Rojas, Maribel y Molina García, Juan Carlos. 2004.** Introducción a principios básicos del desarrollo de software basado en componentes. 2004.
3. **Behr, Miguel. 2008.** Concierto. [En línea] 2008.
4. **Blanqué, Javier. 2008.** BR Consulting. [En línea] 2008.
<http://brconsulting.info/portal/articulos/metodologias-de-desarrollo/extreme-programming---xp.html>.
5. **Bolso, Rodrigo. 2004.** Desarrollo.net. [En línea] 17 de Junio de 2004.
<http://www.dotnetjunkies.com/WebLog/desarrollonet/archive/2004/06/17/16855.aspx>.
6. **Catalani, Exequiel. 2007.** Arquitectura Modelo/Vista/Controlador. [En línea] 20 de Agosto de 2007. <http://exequielc.wordpress.com/2007/08/20/arquitectura-modelovistacontrolador/>.
7. **DTCUBA. 2005.** Boletín Semanal de la Industria Turística Cubana. [En línea] 13 de Junio de 2005. <http://www.dtcuba.com/shownews.aspx?c=15767&ref=dtcnews>.
8. **García Rubio, Félix Óscar. S/A.** Metodologías de desarrollo de software. S/A.
9. **García, Joaquin. 2005.** IngenieroSoftware. [En línea] 7 de Mayo de 2005.
<http://www.ingenierosoftware.com/analisisydiseno/uml.php>.
10. **Gomez Gallego, Juan Pablo. 2007.** Scribd. [En línea] 16 de Septiembre de 2007.
11. **Gracia, Joaquin. 2005.** Patrones de Diseño. [En línea] 27 de Mayo de 2005.
<http://www.ingenierosoftware.com/analisisydiseno/patrones-diseno.php>.
12. **H. Canós, José, Letelie, Patricio y Penadé, Carmen. S/A.** Metodologías Ágiles en el Desarrollo de Software. S/A.
13. **Larman, Craig. S/A.** UML y Patrones. S/A.

14. **Microsoft Solutions Framework. 2006.** GPI Consultores. [En línea] 2006.
<http://www.gpicr.com/msf.aspx>.
15. **Ingeniería de Software. 2007.** Introducción Ingeniería de Software. [En línea] 16 de Octubre de 2007.
16. **MINSAP. 2008.** Informe a la 1ra Conferencia Latinoamericana sobre investigación e innovación en salud. 2008.
17. **Montejano, Germán. 2006.** Metodologías de desarrollo de software ágiles. 2006.
18. **Mousqués, Gastón. 2004.** Desarrollo Agil de Software. 2004.
19. **Pompa Sourd, Frank, y otros. 2001.** Sistema Informático para la Atención Primaria de Salud. La Habana : s.n., 2001.
20. **Rojas, Xiomara. 2001.** Registros Estadísticos Informáticos del Sistema Integrado de urgencias Médicas. La Habana : s.n., 2001.
21. **Solutions Group. S/A.** Arquitectura Orientada a Servicios. Bogotá : s.n., S/A.
22. **StartMedia. S/A.** Herramientas CASE. [En línea] S/A.
23. **StartNet. S/A.** Ingeniería en Desarrollos Informáticos. [En línea] S/A.
<http://www.stnt.com.ar/sistema-emergencias-medicas.asp>.
24. **Villatoro, Alejandro Martínez. 2005.** definiciones Básicas en Medicina de Urgencia. México : s.n., 2005.

BIBLIOGRAFÍA

- Aramayo, Cristian. S/A. MySQL. [En línea] S/A.**
http://www.salnet.com.ar/inv_mysql/pag01_intro.htm#que_es_mysql.
- Ariza Rojas, Maribel y Molina García, Juan Carlos. 2004.** Introducción a principios básicos del desarrollo de software basado en componentes. 2004.
- Behr, Miguel. 2008.** Concierto. [En línea] 2008.
- Blanqué, Javier. 2008. BR Consulting. [En línea] 2008.**
<http://brconsulting.info/portal/articulos/metodologias-de-desarrollo/extreme-programming---xp.html>.
- Bolso, Rodrigo. 2004. Desarrollo.net. [En línea] 17 de Junio de 2004.**
<http://www.dotnetjunkies.com/WebLog/desarrollonet/archive/2004/06/17/16855.aspx>.
- Casal, Julio Terreros. 2008. msdn. [En línea] 2008.**
http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ_3985/default.aspx.
- Catalani, Exequiel. 2007. Arquitectura Modelo/Vista/Controlador. [En línea] 20 de Agosto de 2007.**
<http://exequielc.wordpress.com/2007/08/20/arquitectura-modelovistacontrolador/>.
- Ciberaula. 2006. Patrones de Diseño en aplicaciones Web. [En línea] 2006.**
http://java.ciberaula.com/articulo/disenio_patrones_j2ee/.
- Cibercuba. 2008. Medicina y Salud en Cuba. [En línea] Marzo de 2008.**
http://salud.cibercuba.com/presta_servicios_de_calidad_sistema_integrado_de_urgencias_medicas_en_granma_cuba.
- DTCUBA. 2005. Boletín Semanal de la Industria Turística Cubana. [En línea] 13 de Junio de 2005.**
<http://www.dtcuba.com/shownews.aspx?c=15767&ref=dtcnews>.
- Escambray. 2007. Periódico de la Provincia Sancti Spiritus. [En línea] 18 de Septiembre de 2007.**
<http://www.escambray.cu/Esp/ss/Epasajes061023937.htm>.
- Fernández Escribano, Gerardo. 2002.** Introducción a XP Programming. 2002.
- Figuroa, Pablo. S/A.** Metodología de desarrollo de software Orientado por Objetos. S/A.

García Rubio, Félix Óscar. S/A. Metodologías de desarrollo de software. S/A.

García, Joaquin. 2005. IngenieroSoftware. [En línea] 7 de Mayo de 2005.
<http://www.ingenierosoftware.com/analisisydiseno/uml.php>.

Gomez Gallego, Juan Pablo. 2007. Scribd. [En línea] 16 de Septiembre de 2007.

Gracia, Joaquin. 2005. Patrones de Diseño. [En línea] 27 de Mayo de 2005.
<http://www.ingenierosoftware.com/analisisydiseno/patrones-diseno.php>.

H. Canós, José, Letelie, Patricio y Penadé, Carmen. S/A. Metodologías Ágiles en el Desarrollo de Software. S/A.

Hernández, Enrique. S/A. El Lenguaje Unificado de Modelado(UML). S/A.

Infogdssistemas. S/A. Metodología RUP. [En línea] 29 de Julio de S/A.
<http://infogdssistemas.spaces.live.com/blog/cns!5B9ED81A701B41D2!139.entry>.

Larman, Craig. S/A. UML y Patrones. S/A.

Martínez Zelada, Luz María. 2008. Diario Granma. [En línea] 28 de Mayo de 2008.
<http://granma.co.cu/2008/05/28/nacional/artic22.html>.

Microsoft Solutions Framework. 2006. GPI Consultores. [En línea] 2006.
<http://www.gpicr.com/msf.aspx>.

Ingeniería de Software. 2007. Introducción Ingeniería de Software. [En línea] 16 de Octubre de 2007.

MINSAP. 2008. Informe a la 1ra Conferencia Latinoamericana sobre investigación e innovación en salud. 2008.

Montejano, Germán. 2006. Metodologías de desarrollo de software ágiles. 2006.

Mousqués, Gastón. 2004. Desarrollo Agil de Software. 2004.

Pecos, Daniel. S/A. PostGreSQL vs. MySQL. [En línea] S/A.
http://www.netpecos.org/docs/mysql_postgres/x57.html.

Pompa Sourd, Frank, y otros. 2001. Sistema Informático para la Atención Primaria de Salud. La Habana : s.n., 2001.

Radio Victoria. 2007. Tiempo 21. [En línea] 24 de Diciembre de 2007.
http://www.tiempo21.cu/hipertextos/sistema_integrado_urgencias_medicas_sium_071224.htm.

Revista Cubana de Salud Pública. **Castillo Guzmán, Antonio y Arocha Mariño, Carmen. 2000.** 001, La Habana : s.n., 2000, Vol. 26.

Rojas, Xiomara. 2001. Registros Estadísticos Informáticos del Sistema Integrado de urgencias Médicas. La Habana : s.n., 2001.

Rothirsch Lemberger, Andrés. 2002. Metodologías Ágiles para el desarrollo de sistemas. 2002.

S. Pressman, Roger. S/A. Ingeniería de Software. Un enfoque práctico. S/A.

Scalzone, Patricia. S/A. Microsoft Solution Framework v.4 Agile (MSF). S/A.

Scribd. 2008. Metodologías de desarrollo de software. 2008.

S/A. Sitio de descargas de software. [En línea] S/A.
[http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_\(M%C3%8D\)_1472_0_p/](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_(M%C3%8D)_1472_0_p/).

Solutions Group. S/A. Arquitectura Orientada a Servicios. Bogotá : s.n., S/A.

StartMedia. S/A. Herramientas CASE. [En línea] S/A.

StartNet. S/A. Ingeniería en Desarrollos Informáticos. [En línea] S/A. <http://www.stnt.com.ar/sistema-emergencias-medicas.asp>.

S/A. Unadecodigo. [En línea] S/A. <http://www.unadecodigo.com/2007/05/30/el-paradigma-modelo-vista-controlador-tutorial-ror-ii/>.

Villatoro, Alejandro Martínez. 2005. definiciones Básicas en Medicina de Urgencia. México : s.n., 2005.

WordReference. Diccionario de la Lengua Española. [En línea] [Citado el: 10 de Diciembre de 2007.]
<http://www.wordreference.com/definición/>.

ANEXOS

Anexo 1: Diagrama de Actividades.

Caso de Uso Procesar Demanda.

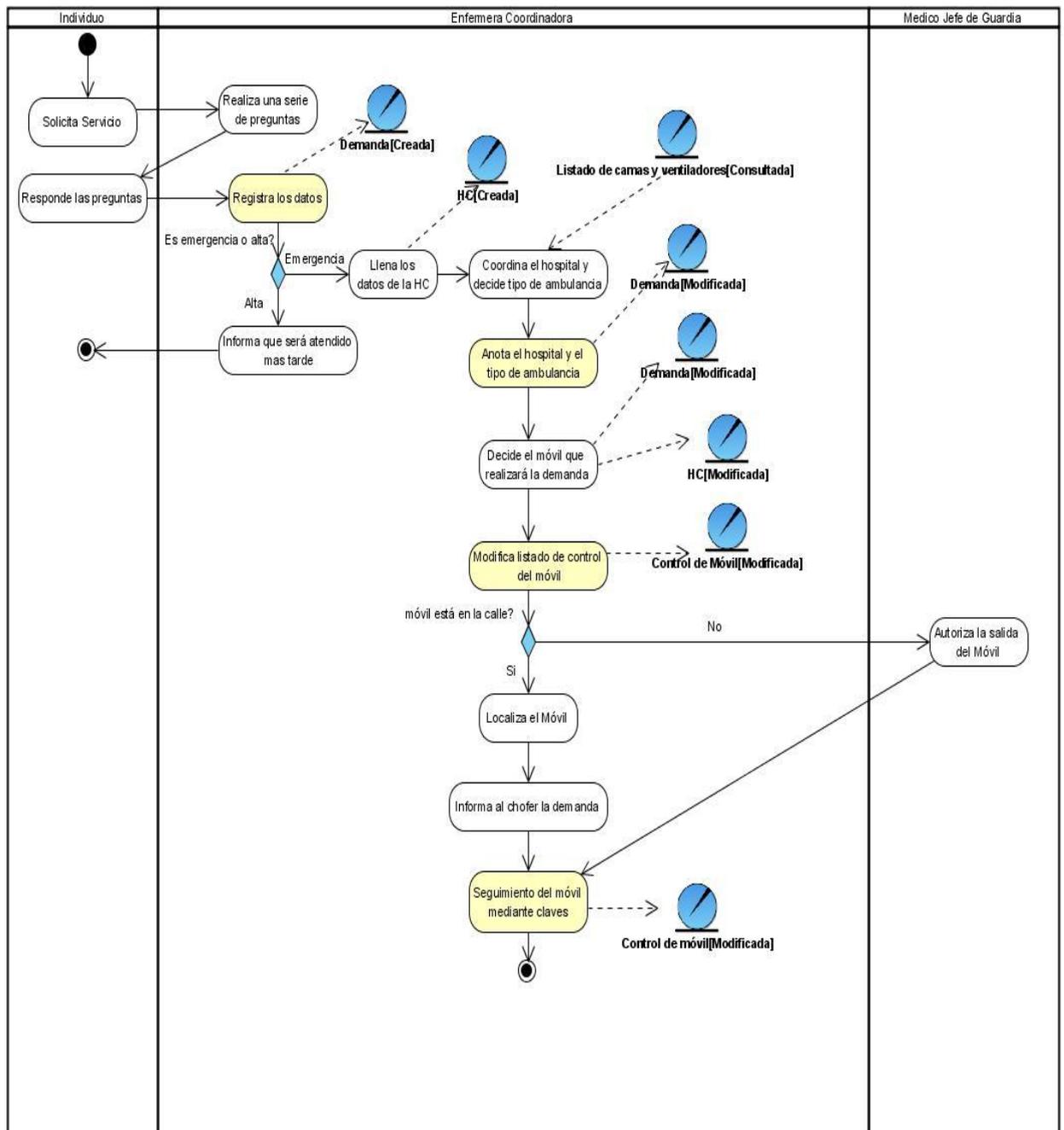


Figura Anexo 1. 1 Diagrama de Actividades Caso de Uso Procesar Demanda.

Caso de Uso Registrar Traslados.

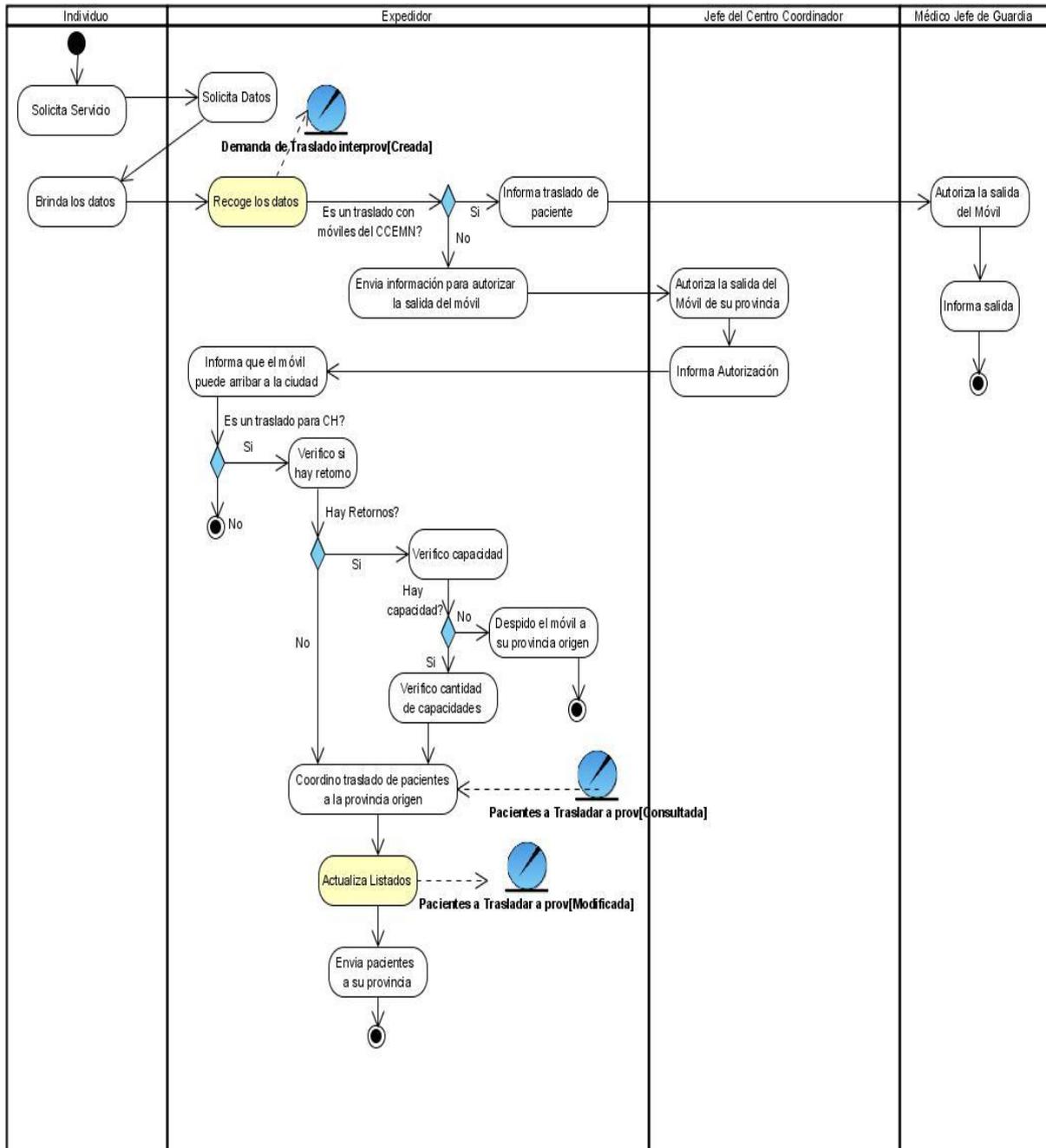


Figura Anexo 1. 2 Diagrama de Actividades Caso de Uso Registrar Traslados.

Caso de Uso Confeccionar Listado No Urgencia.

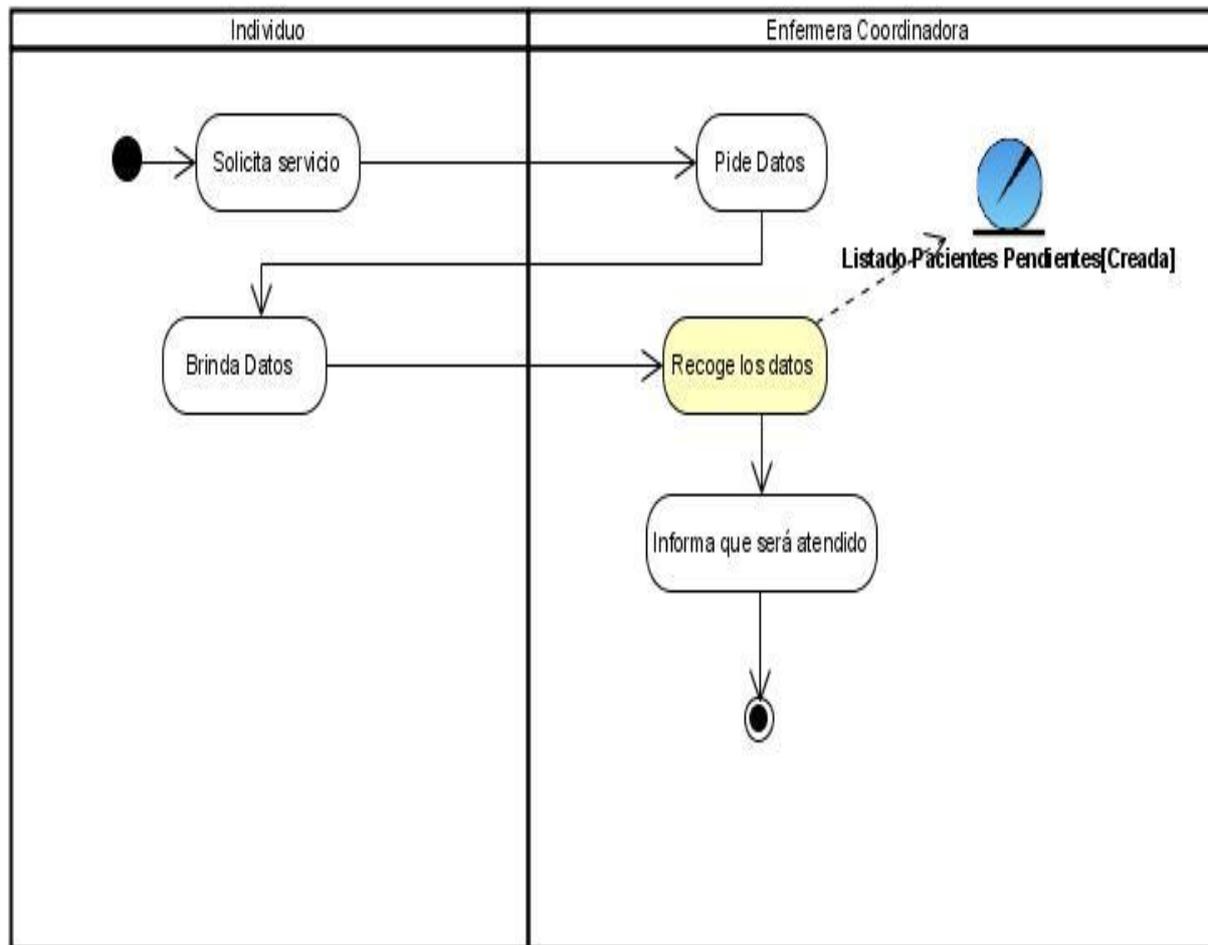


Figura Anexo 1. 3 Diagrama de Actividades Caso de Uso Confeccionar Listado NU.

Caso de Uso Planificar No Urgencias.

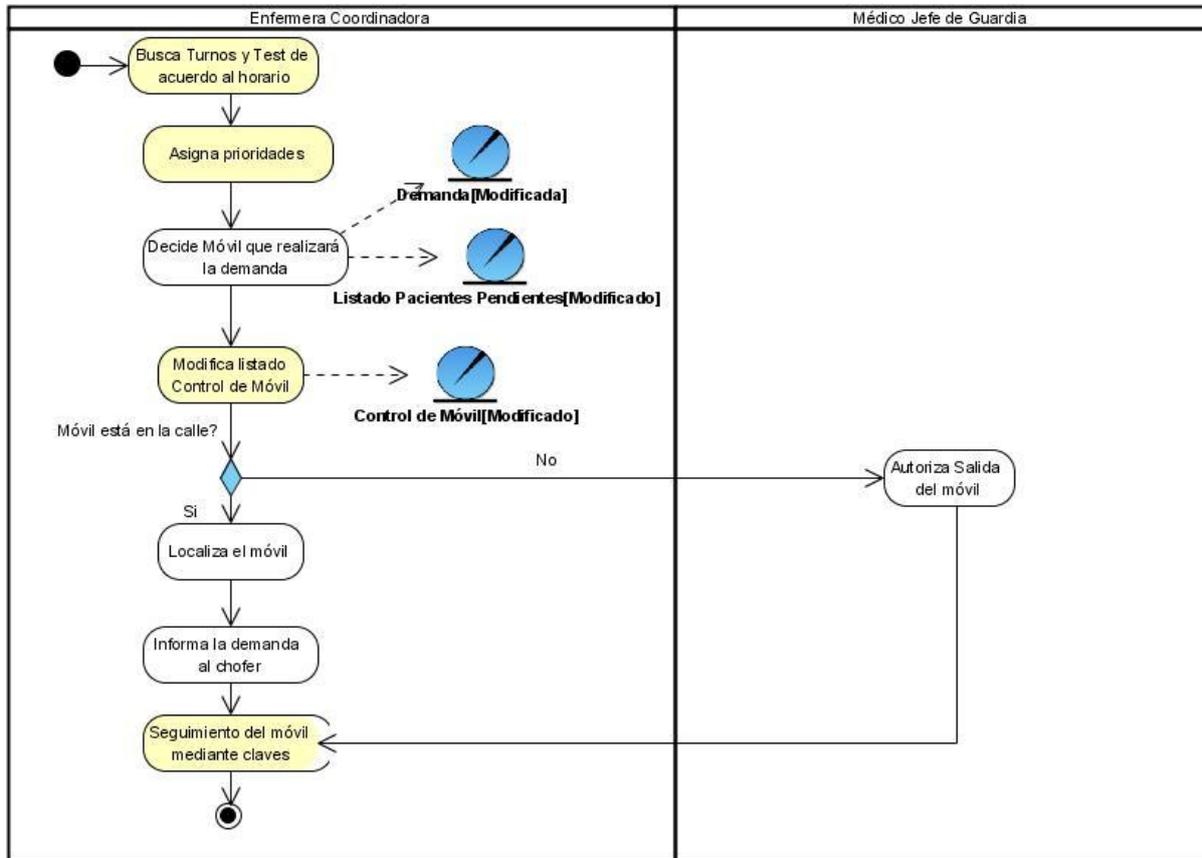


Figura Anexo 1. 4 Diagrama de Actividades Caso de Uso Planificar NU.

Caso de Uso Realizar Altas.

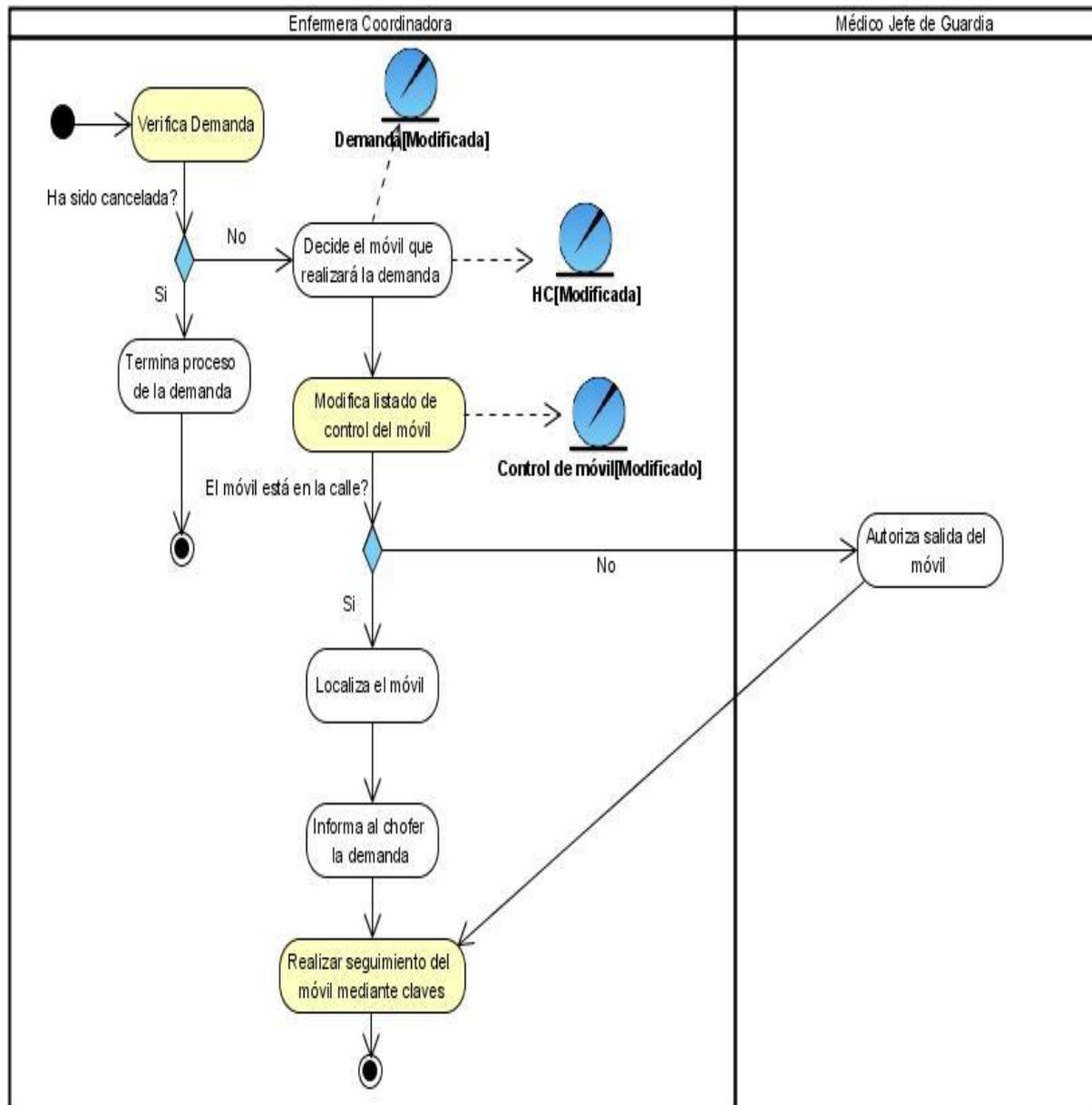


Figura Anexo 1. 5 Diagrama de Actividades Caso de Uso Realizar Altas.

Caso de Uso Controlar Trabajo Diario.

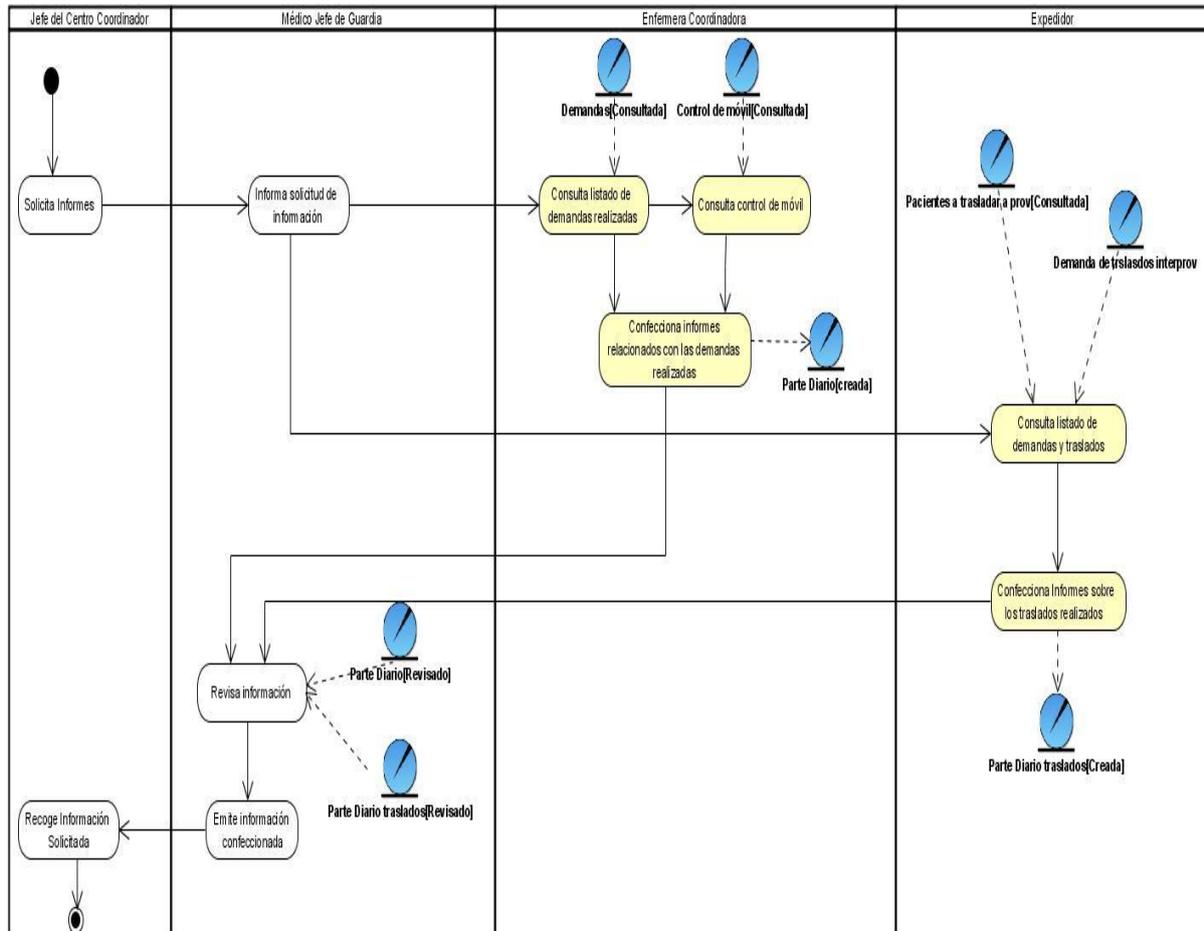


Figura Anexo 1. 6 Diagrama de Actividades Caso de Uso Controlar Trabajo Diario.

Anexo 2: Modelo Pacientes para Traslado a Provincia.

DEMANDA DE PACIENTES PARA TRASLADO A PROVINCIA. _____

Fecha	Activa	Paciente	Edad	Diagnostico	Origen	Destino	C	S	Fecha Salida

ANEXO 4. 1 Modelo Pacientes para Traslado a Provincia.

GLOSARIO DE TÉRMINOS

Arquitectura: Estructura organizativa de un sistema que incluye su descomposición en partes, su conectividad, mecanismos de interacción y principios de guía que proporciona información sobre el diseño del mismo.

Artefactos: Son productos tangibles del proceso, como por ejemplo, el modelo de casos de uso, el código fuente, etc.

Hospital: Unidad de Salud que se responsabiliza de manera íntegra con las unidades de salud de la Atención Primaria (Áreas de Salud) para el enfrentamiento de los problemas de salud de una determinada población.

Historia Clínica de Traslado: Es el documento que utiliza la Enfermera Coordinadora y el Paramédico que viaja en la ambulancia para recoger información general del individuo, antecedentes patológicos, datos del examen físico, además de anotar todas las acciones realizadas en relación con el problema de salud que afecta al individuo.

Hipertexto: Formato que se le aplica a un texto, en el cual se representan palabras claves (en la mayoría de los casos subrayadas o con otros colores) las cuales dan acceso a un información determinado.

Orientado a Objetos: Significa que el software se organiza como una colección de objetos discretos que contiene tanto estructura de datos como también un comportamiento.

Registro de Personal de la Salud (RPS): Registro que brinda información sobre el Personal de Salud existente. Se utiliza en RAS para ubicar al Personal de Salud en los GBT, EBS y Locales de Viviendas.

Registro de Ubicación (RU): Registro que brinda información sobre las Provincias, Municipios, Localidades, Calles y Manzanas. Se utiliza en RAS para ubicar a las Viviendas, Centros Laborales y Locales.

Registro de Unidades de Salud (RUS): Registro que brinda información sobre las Unidades de Salud de Cuba. Se utiliza en RAS para obtener los datos de las Unidades de Salud que son Áreas de Salud y de los hospitales base del área.

Registro de Ciudadanos (RC): Registro que brinda información acerca de todas las personas del país registradas en el sistema.

Sistema de Información para la Salud (SISalud): Es la integración de un conjunto de aplicaciones cuyo propósito fundamental es la informatización del Sistema Nacional de Salud (SNS). Los componentes se distribuyen de acuerdo a su nivel y ponen su información a disposición del resto, permitiendo la interoperabilidad y el intercambio de información entre los mismos.

Software: Al conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema (hardware). Esto incluye aplicaciones informáticas tales como un procesador de textos, que permite al usuario realizar una tarea, y software de sistema como un sistema operativo, que permite al resto de programas funcionar adecuadamente, facilitando la interacción con los componentes físicos y el resto de aplicaciones.

Stakeholders: Personas u organizaciones que están activamente implicadas en el negocio ya sea porque participan en él o porque sus intereses se ven afectados con los resultados del proyecto. Pueden ser los propietarios, la dirección, quienes financian, los clientes, los trabajadores, los proveedores, la competencia, la comunidad local, etc.

Servidor: Una aplicación informática o programa que realiza algunas tareas en beneficio de otras aplicaciones llamadas clientes.

Sistema Operativo: Es un conjunto de instrucciones destinados a permitir la comunicación del usuario con una aplicación y gestionar sus recursos de una forma eficaz.