



Universidad Tecnológica de la Habana
"José Antonio Echeverría"

Universidad Tecnológica de La Habana "José Antonio Echeverría"
Facultad de Ingeniería Informática.
Filial de Ciencias Técnicas de 10 de octubre.

**Sistema informático para la gestión de las acciones realizadas para
el control y erradicación de vectores**

Trabajo de diploma para optar por el título de Ingeniería en Informática

Autor(es): Ariel Hernandez Seoane
ariel.seoane@nauta.cu
seoaneirete@gmail.com
54142008

Tutor(es): Ing. Darian Díaz García
ddgarcia537@gmail.com

La Habana, Cuba
Enero, 2019

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor (Declaramos que somos los únicos autores) de este trabajo y autorizo (autorizamos) al Área de salud Mulgoba Campaña Antivectorial del Policlínico Victoria Cuba-Angola y a la Facultad de Ingeniería Informática para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo (firmamos) la presente a los 15 días del mes de
enero del 2019.

Arell Hernandez Deorane

Nombre completo del primer autor

Ing. Darian Diaz Garcia

Nombre completo del primer tutor

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma, titulado **Sistema informático para la gestión de las acciones realizadas para el control y erradicación de vectores**, fue realizado en nuestra entidad Policlínico Victoria Cuba-Angola, Área de salud Mulgoba de la Campaña Antivectorial. Se considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

Totalmente

Parcialmente en un ____ %

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes (cuantificar):

Permite a los usuarios mediante una fuente de información única acceder fácilmente a la información. Facilita el procesamiento de la información de las acciones de prevención y las acciones de control para la realización de los diferentes informes y reportes de los indicadores del área de salud.

Como resultado de la implantación de este trabajo se reporta un efecto económico que asciende a **19677.60 MN y/o 819.90 CUC.**

Y para que así conste, se firma la presente a los 16 días del mes de mayo del año

2019

Lic Miguel Rodríguez Amador

Nombre del representante de la entidad

M.Rodríguez

Firma

J. López Virgiliaca (A.U.)

Cargo



Agradecimientos

A mis padres, por el amor inmenso que siempre me han dado y por todos los consejos y apoyo, que han hecho que me convierta en la persona que soy hoy.

A su jefe de área de Musgooba Lic. Miguel Rodríguez por la ayuda que me dio durante la creación de este trabajo de diploma.

A todas las personas que me han ayudado durante el transcurso de la carrera.

Resumen

El presente trabajo de diploma se realiza en el área de salud Mulgoba perteneciente a la Unidad Nacional de Vigilancia y Lucha Antivectorial. Esta entidad es la encargada de dirigir el monitoreo de los vectores existentes en el país. Esta entidad está dividida territorialmente en Unidades Provinciales, Unidades Municipales y Áreas de Salud, estas últimas son las encargadas de realizar las inspecciones a las viviendas y los centros priorizados y las zonas de riesgo, las revisiones de los centros priorizados, las zonas de riesgo, las larvitrampas ubicadas en estos inmuebles y realizar las encuestas de moscas y cucarachas, estas acciones mencionadas se denominan acciones de prevención, así como también aplicar el tratamiento químico y realizar radiobatidas (acciones de control) a las manzanas donde se hallaron problemas en las acciones de prevención y donde han sido reportados febres. [1]

Este trabajo se enfocará en las áreas de salud y en el municipio, en donde las Áreas de Salud tienen la obligación de registrar toda la información colectada en las diferentes acciones y de entregar a la Unidad Municipal reportes diarios con los datos recopilados en las inspecciones diarias, los tratamientos químicos y las radiobatidas, así como reportes semanales con la información de las acciones realizadas a los centros priorizados, las zonas de riesgo y a las larvitrampas.

Para desarrollar este proceso el método utilizado es obsoleto e ineficiente para el actual flujo de información.

Aprovechando el auge y el desarrollo de la informatización, se propone realizar una aplicación Web, que permitirá darle solución la problemática actual que presenta la entidad, dinamizando los procesos y brindando herramientas que faciliten la toma de decisiones en el futuro.

Palabras Claves: Vectores, inspecciones, radiobatidas, tratamientos químicos, aplicación Web

Abstract

The present diploma work is carried out in the area of health Mulgoba belonging to the National Unit of Surveillance and Antivectorial Fights. This entity is the one in charge of directing the monitored of the existent vectors in the country. This entity is divided territorially in Provincial Units, Municipal Units and Areas of Health, these last are those in charge of carrying out the inspections to the housings and the prioritized centers and the areas of risk, the revisions of the prioritized centers, the areas of risk, the larvitrampas located in these properties and to carry out the surveys of flies and cockroaches, these mentioned actions are denominated actions of prevention, as well as to apply the chemical treatment and to carry out radiobatidas (control actions) to the apples where they were problems in the actions of prevention and where they have been reported feverish. [1]

This work will be focused in the areas of health and in the municipality where the Areas of Health have the obligation of registering all the information collected in the different actions and of surrendering to the Unit Municipal reports newspapers with the data gathered in the daily inspections, the chemical treatments and the radiobatidas, as well as weekly reports with the information of the actions carried out to the prioritized centers, the areas of risk and to the larvitrampas.

To develop this process, him used method is obsolete and inefficient for the current flow of information.

Taking advantage of the peak and the development of the informatization, it intends to be carried out an application Web that will allow to give him/her solution the current problem that presents the entity, energizing the processes and tools that facilitate the taking of decisions in the future toasting.

Key Words: Control actions, actions of prevention, application Web

Índice

<i>INTRODUCCIÓN</i>	1
CAPÍTULO 1 FUNDAMENTOS TEÓRICOS	6
1.1 <i>INTRODUCCIÓN</i>	6
1.2 <i>OBJETIVOS ESTRATÉGICOS DE LA ORGANIZACIÓN</i>	6
1.3 <i>DESCRIPCIÓN DE LOS PROCESOS QUE SE EJECUTAN EN EL CAMPO DE ACCIÓN</i>	7
1.4 <i>ANÁLISIS CRÍTICO DE LA EJECUCIÓN ACTUAL DE LOS PROCESOS</i>	8
1.5 <i>PROCESOS OBJETO DE AUTOMATIZACIÓN</i>	9
1.6 <i>SISTEMAS AUTOMATIZADOS EXISTENTES VINCULADOS AL CAMPO DE ACCIÓN</i>	10
1.7 <i>TENDENCIAS Y TECNOLOGÍAS ACTUALES</i>	11
1.8 <i>ANÁLISIS CRÍTICO DE LAS FUENTES Y BIBLIOGRAFÍAS UTILIZADAS (ESTADO DEL ARTE)</i>	26
1.9 <i>CONCLUSIONES PARCIALES DEL CAPÍTULO</i>	27
CAPÍTULO 2 MODELO DE NEGOCIO	28
2.1 <i>INTRODUCCIÓN</i>	28
2.2 <i>MODELO DEL NEGOCIO ACTUAL</i>	28
2.3 <i>REGLAS DEL NEGOCIO A CONSIDERAR</i>	29
2.4 <i>ACTORES DEL NEGOCIO</i>	30
2.5 <i>DIAGRAMA DE CASOS DE USO DEL NEGOCIO</i>	31
2.6 <i>TRABAJADORES DEL NEGOCIO</i>	32
2.7 <i>CASOS DE USO DEL NEGOCIO</i>	33
2.7.1 <i>Caso de uso Realización de las acciones preventivas en las viviendas, en los centros priorizados y las zonas de riesgo</i>	33
2.7.2 <i>Caso de uso Orientar y realizar acciones de control</i>	34
2.7.3 <i>Caso de uso Solicitar atención de los febriles reportados</i>	35
2.7.4 <i>Caso de uso Recibir los informes de las acciones realizadas</i>	36
2.8 <i>MODELO DE OBJETOS</i>	37
2.8.1 <i>Descripción de las entidades del negocio</i>	38
2.9 <i>CONCLUSIONES PARCIALES DEL CAPÍTULO</i>	39
CAPÍTULO 3 REQUISITOS	41
3.1 <i>INTRODUCCIÓN</i>	41
3.2 <i>DEFINICIÓN DE LOS REQUISITOS FUNCIONALES</i>	41
3.3 <i>ACTORES DEL SISTEMA A AUTOMATIZAR</i>	43
3.4 <i>JERARQUÍA DE ACTORES</i>	45
3.5 <i>PAQUETES Y SUS RELACIONES</i>	45
3.6 <i>DIAGRAMAS DE CASOS DE USO DEL SISTEMA</i>	47
3.6.1 <i>Diagrama de casos de uso del paquete Seguridad</i>	47
3.6.2 <i>Diagrama de casos de uso del paquete Gestión</i>	48
3.6.3 <i>Diagrama de casos de uso del paquete Nomencladores</i>	49
3.6.4 <i>Diagrama de casos de uso del paquete Reportes</i>	50
3.7 <i>DESCRIPCIÓN DE LOS PRINCIPALES CASOS DE USO</i>	50
3.7.1 <i>Paquete de Seguridad</i>	50
3.7.2 <i>Paquete de Gestión</i>	53
3.7.3 <i>Paquete Reportes</i>	58
3.8 <i>DEFINICIÓN DE LOS REQUISITOS NO FUNCIONALES</i>	59
3.9 <i>CONCLUSIONES</i>	61
CAPÍTULO 4 DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA	62
4.1 <i>INTRODUCCIÓN</i>	62
4.2 <i>DIAGRAMA DE CLASES DEL DISEÑO</i>	63
4.2.1 <i>Paquete Seguridad</i>	63
4.2.2 <i>Paquete Gestión</i>	64
4.2.3 <i>Paquete Nomencladores</i>	65
4.2.4 <i>Paquete Reportes</i>	66
4.3 <i>PATRÓN ARQUITECTÓNICO</i>	66

4.4 PRINCIPIOS DE DISEÑO.....	67
4.5 PATRONES DE DISEÑO.....	68
4.6 ESTRUCTURACIÓN EN CAPAS.....	72
4.6.1 Enfoque por reutilización.....	72
4.6.2 Enfoque por responsabilidad.....	73
4.7 MECANISMOS DE ACCESO A DATOS.....	73
4.8 DISEÑO DE LA BASE DE DATOS.....	74
4.8.1 Modelo lógico de datos. [30].....	74
4.8.2 Modelo físico de datos [30]	75
4.9 INTERFAZ DE USUARIO	76
4.9.1 Formato de salida de los reportes.....	77
4.9.2 Tratamiento de errores.....	77
4.10 DIAGRAMA DE DESPLIEGUE	79
4.11 CONCLUSIONES PARCIALES DEL CAPITULO	80
CAPÍTULO 5 VALIDACIÓN Y FACTIBILIDAD DE LA SOLUCIÓN PROPUESTA	81
5.1 INTRODUCCIÓN	81
5.2 PRUEBAS.....	81
5.3 DISEÑO DE LOS CASOS DE PRUEBA	82
5.4 RESULTADOS OBTENIDOS EN LAS PRUEBAS	96
5.5 RESULTADOS DEL ESTUDIO DE FACTIBILIDAD [32]	97
5.6 BENEFICIOS TANGIBLES E INTANGIBLES	97
5.7 ANÁLISIS DE COSTOS Y BENEFICIOS.....	98
5.8 CONCLUSIONES PARCIALES DEL CAPÍTULO	98
CONCLUSIONES GENERALES.....	99
RECOMENDACIONES.....	100
REFERENCIAS	101
GLOSARIO DE SIGLAS Y TÉRMINOS	103
ANEXOS.....	I

Índice de tablas

Tabla 2.1 Descripción de los actores del negocio.....	30
Tabla 3.1 Actores del sistema a automatizar.....	43
Tabla 5.5 Pruebas unitarias de tipo caja negra del caso de prueba Crear Inspección. Técnica de partición de equivalencias.....	82
Tabla 5.6 Pruebas de integración de tipo caja negra.....	83
Tabla 5.7 Pruebas unitarias de tipo caja negra del caso de prueba Crear Foco. Técnica de partición de equivalencias.....	85
Tabla 5.4 Pruebas de integración de tipo caja negra.....	88
Tabla 5.5 Pruebas unitarias de tipo caja negra del caso de prueba Crear Inspección. Técnica de partición de equivalencias.....	90
Tabla 5.6 Pruebas de integración de tipo caja negra.....	92
Tabla 5.7 Pruebas unitarias de tipo caja negra del caso de prueba Crear Foco. Técnica de partición de equivalencias.....	94
Tabla 5.8 Resultados obtenidos.....	96

Índice de imágenes

Imagen 1-1 Lenguajes de programación a nivel de servidor más utilizados en el 2018.....	14
Imagen 2-1 Diagrama de Casos de Uso del Negocio.....	31
Imagen 2-2 Diagramas de actividades del Caso de Uso Realización de las acciones.....	33
Imagen 2-3 Diagramas de actividades del Caso de Uso Orientar y realizar acciones de control.....	34
Imagen 2-4 Diagramas de actividades del Caso de Uso Solicitar atención de los febris reportados.....	35

Imagen 2-5 Diagramas de actividades del Caso de Uso Recibir los informes de las acciones realizadas.....	36
Imagen 2-6 Modelo de Objeto.....	37
Imagen 3-1 Jerarquía de actores del sistema.....	45
Imagen 3-2 Paquetes y sus relaciones.....	45
Imagen 3-3 Diagrama de casos de uso del paquete Seguridad.....	47
Imagen 3-4 Diagrama de casos de uso del paquete Gestión.....	48
Imagen 3-5 Diagrama de casos de uso del paquete Nomencladores.....	49
Imagen 3-6 Diagrama de casos de uso del paquete Reportes.....	50
Imagen 4-1 Diagrama de clases del diseño del Paquete Seguridad.....	63
Imagen 4-2 Diagrama de clases del diseño del Paquete Gestión.....	64
Imagen 4-3 Diagrama de clases del diseño del Paquete Nomencladores.....	65
Imagen 4-4 Paquete Reportes.....	66
Imagen 4-5 Patrón Factory.....	68
Imagen 4-6 Patrón Facade.....	70
Imagen 4-7 Enfoque por reutilización.....	72
Imagen 4-8 Enfoque por responsabilidad.....	73
Imagen 4-9 Modelo lógico de datos.....	74
Imagen 4-10 Modelo físico de datos.....	75
Imagen 4-11 Interfaz de usuario.....	76
Imagen 4-12 Interfaz de Nueva inspección.....	76
Imagen 4-13 Gráfico.....	77
Imagen 4-14 Errores en entrada de datos.....	78
Imagen 4-15 Página de error 404.....	78
Imagen 4-16 Diagrama de despliegue.....	79

Introducción.

Antecedentes

La Unidad Nacional de Vigilancia y Lucha Anti vectorial adscrita al Viceministerio de Higiene y Epidemiología, es el organismo rector de todas las acciones contra los vectores en el país; este dirige metodológicamente las Unidades Provinciales de Vigilancia y Lucha Anti vectorial que existen en cada Provincia. Las Unidades Municipales están divididas en Áreas de Salud que son las ejecutoras de todas las acciones directas en el terreno, estas están conformadas por un Jefe de Área y Lucha Anti vectorial, un biólogo que se encarga de clasificar las muestras o focos, un estadístico, un Jefe de Almacén y una cantidad suficiente de Operarios “A” que forman brigadas, que es dirigida por un Jefe de Brigada. Cada brigada tiene a su cargo varias Zonas del Área y dichas zonas están divididas en manzanas. [1]

El actual Programa de Erradicación de vectores surgió debido a la primera epidemia de Dengue Hemorrágico ocurrida en el año 1981 en el continente americano que se presentó en nuestro país con 344,302 casos notificados, a un costo de 158 muertos, de ellos 101 niños, que fue precedida por una gran epidemia de Dengue Clásico (Serotipo I) en 1977, por lo que se revitalizó la Campaña de Erradicación de vectores que sustentaba el Programa del mismo nombre el cual desde ese momento pasó a ser uno de los más importantes para el Ministerio de Salud Pública y las máximas instancias de nuestro Estado que le han priorizado todos los recursos necesarios para su ejecución.

[3]

Las áreas son las encargadas de realizar las acciones de prevención y las acciones de control de vectores; entre las que se encuentran la inspección de las viviendas y centros laborales de las manzanas, la revisión de las zonas de riesgo y centros priorizados, larvitrampas, la realización de encuestas de moscas y cucarachas, la detección de focos y además de la aplicación de los tratamientos químicos y la realización de las radiobatidas a las manzanas donde se han reportado problemas. Las acciones antes mencionadas son entregadas para su registro por el jefe de cada Brigada a la estadística del Área quien se encarga de realizar los diferentes cálculos y registrar toda la información referente a las mismas. Las muestras o focos recolectados por cada una de las Brigadas son entregadas a la Bióloga del área de salud para su clasificación y registro. Por otra parte, toda esta información se registra en papel, la cual es utilizada por los jefes de área para la toma de decisiones sobre las acciones de control a aplicar y para la elaboración de los reportes semanales que deben ser entregados a las Unidades Municipales. Dichos reportes son registrados por el jefe de área en el documento Excel de reportes y luego enviados al director de la unidad municipal lo que trae como consecuencia la descentralización de la información.

Por lo anteriormente expuesto se deduce la siguiente **situación problemática:**

- El Jefe de Área no tiene la manera de comparar de forma ágil los indicadores históricos de las acciones que se han realizado en los centros priorizados y las zonas de riesgo
- Los reportes diarios y semanales no pueden ser entregados a las Unidades Municipales en tiempo.
- La información no se encuentra centralizada, por lo que resulta engorroso supervisar, buscar y/o actualizar los indicadores de las acciones realizadas.
- Los informes son elaborados en papel, sin la posibilidad de visualizar gráficos comparativos que midan el comportamiento de las acciones de control aplicadas en las áreas.
- Los documentos tienden a deteriorarse rápidamente debido a la constante manipulación por los directivos de las áreas lo que conlleva a la pérdida de información.

Problema a resolver

¿Cómo optimizar la gestión de las acciones realizadas a los centros priorizados, las zonas de riesgo y a las manzanas de las áreas de salud para lograr una mejor organización?

Objeto de estudio

Escenario cliente/usuario se enmarca en:

- Se enmarca en la gestión de las acciones de prevención y control realizadas para la erradicación de vectores.

En el escenario informático se enmarca en:

- En el estudio de tecnologías y herramientas de desarrollo para aplicaciones informáticas.

Campo de acción

Escenario cliente/usuario se enmarca en:

- Análisis de los procesos que se realizan durante la realización de las acciones prevención y control de vectores en los centros priorizados, las zonas de riesgo y manzanas del área de salud.

En el escenario informático:

Marcos de trabajo y las tecnologías para la gestión de datos orientados a entornos web.

Objetivo General

El **objetivo general** que se propone es desarrollar un sistema para la gestión de las acciones de prevención y control realizadas a las áreas de salud de los municipios que facilite el control y la evaluación de la ejecución de dichas acciones.

Para llevar a cabo el objetivo general se proponen los **objetivos específicos** siguientes y las **tareas** correspondientes:

1. Fundamentar el estado del arte de la gestión las acciones de prevención y control de vectores.

- Recopilar información y documentación vinculada a los procedimientos asociados a las acciones de prevención y control. Realización de una descripción detallada de los procesos mediante entrevistas con los Jefes de Área.
- Realizar un estudio tanto a nivel nacional como internacional sobre sistemas de gestión similares al de la solución propuesta.
- Estudiar las tecnologías existentes y seleccionar la más adecuada de acuerdo a los requerimientos.
- Análisis crítico de las fuentes y bibliografías utilizadas.

2. Modelar los procesos que intervienen en la gestión de las acciones de prevención y control.

- Definir los actores y trabajadores del negocio.
- Identificar los casos de uso del negocio que representan los procesos de la gestión de las acciones de prevención y control de vectores.
- Identificar las reglas del negocio.
- Realizar la descripción detallada de los casos de uso del negocio a través de los diagramas de actividades.
- Identificar las entidades del negocio y su relación con los actores del negocio.

3. Diseñar el sistema propuesto con vista a definir la arquitectura que intervendrá en los procesos antes descritos.

- Definir las funcionalidades de la solución propuesta.
- Definir actores del sistema.
- Definir los paquetes y sus relaciones.
- Identificar y describir los casos de uso del sistema.
- Definir los casos de uso del sistema.
- Realizar la descripción detallada de los casos de uso del sistema más importantes.

4. Realizar la descripción de la solución propuesta.

- Definir los diagramas de clases de diseño.
- Modelar la arquitectura de la aplicación.

- Definir los patrones de diseño a utilizar.
- Definir los principios de diseño.
- Describir el modelo lógico y físico de la base de datos.
- Definir el diagrama de despliegue.

5. Realizar la validación y el estudio de factibilidad asociado a la aplicación desarrollada.

- Seleccionar los artefactos a los cuales se les realizaran pruebas.
- Seleccionar los casos de uso para la realización de pruebas.
- Seleccionar los tipos de pruebas que se le realizaran al sistema.
- Documentar las pruebas.
- Estimar el tiempo, el costo y el esfuerzo a tener en cuenta en el desarrollo de la aplicación.
- Realizar análisis de costos y beneficios que se obtienen con el desarrollo de la aplicación propuesta.

Valor práctico esperado del trabajo.

Toda la información de las acciones prevención y control de vectores realizadas estará integrada en un mismo repositorio, lo cual facilitará la realización de consultas y podrá permitir guardar un histórico de indicadores de dichas acciones en los centros priorizados, las zonas de riesgo y manzanas para la realización de análisis de comportamientos para la toma de decisiones. La generación de reportes ayudará a realizar estudios más objetivos y oportunos sobre las acciones de prevención y control de vectores. Al no existir ninguna herramienta anterior en las Áreas de salud; se pretende también darle un punto de partida y crear un estándar para el control de este tipo de trabajo. Una característica que persigue esta aplicación es que pueda ser mantenida y actualizada con el menor esfuerzo posible.

El presente documento está dividido en cinco capítulos que abarcan toda la investigación realizada para este trabajo de diploma. A continuación, se expone una breve descripción de ellos:

En el capítulo 1 “Fundamentos Teóricos” se plasman los estudios y justificaciones que sostienen la presente investigación, se expone el flujo actual de los procesos, donde se realiza un análisis crítico de los mismos y se definen las tendencias y tecnologías actuales para el desarrollo de proyectos informáticos.

En el capítulo 2 “Modelo del Negocio” se presenta la modelación del negocio en la cual se desarrolla la problemática planteada. Se exponen las reglas del negocio a seguir. Se definen las principales entidades y conceptos vinculados al negocio.

En el capítulo 3 “Requisitos” se identifican los actores del sistema, se definen los requisitos funcionales y no funcionales de la aplicación, sus paquetes y los casos de uso asociados.

En el Capítulo 4: “Descripción de la Solución Propuesta” se detalla el diagrama de clases de diseño de cada paquete de la propuesta de solución, los principios de diseño, el tratamiento de errores, el diseño de la base de datos (modelos lógico y físico), el modelo de despliegue.

En el capítulo 5 “Validación y factibilidad de la solución propuesta” se realizan los cálculos relacionados con las estimaciones de costo y tiempo de desarrollo del proyecto, unido a los beneficios que genera el desarrollo de la aplicación, con vistas a avalar su factibilidad. Además, se realiza una documentación de las pruebas realizadas al sistema y los resultados arrojados.

Capítulo 1 Fundamentos teóricos.

1.1 Introducción.

Las actividades de registro y supervisión de las acciones realizadas para el control del mosquito *Aedes Aegypti* y otros Culícidos son realizadas en todas las Áreas con el objetivo de poder garantizar la calidad de las inspecciones diarias, las radiobatidas realizadas, conocer la situación higiénico-sanitaria, el índice de infestación en las áreas de salud además de llevar un control de los febris y viajeros para realizarles las radiobatidas e inspecciones.

El presente capítulo abordará los fundamentos teóricos generales. Se muestran aspectos relacionados con la organización, los objetivos, la misión y los procesos de negocio que la soportan. Se analizan las causas que originan la situación problemática, las consecuencias y se describe de manera general los sistemas informatizados que están vinculados al campo de acción del proyecto. Se pretende brindar un recorrido informativo a través de las tecnologías para el desarrollo del sistema propuesto, se analizan las ventajas y desventajas que las caracterizan y la elección para este sistema.

1.2 Objetivos estratégicos de la organización.

Objetivos estratégicos de la organización:

Los vectores conviven con el hombre desde tiempos remotos de la antigüedad según lo recoge la historia y hoy se encuentran ampliamente dispersos por todos los continentes. Es la especie de mosquitos *Aedes Aegypti* una de las de mayor importancia para el hombre por su amplia distribución, difícil control y azote permanente a la salud. Dada la infestación creciente de esta especie, en países con los que Cuba mantiene estrechas relaciones y en los que los programas de control no cuentan con la voluntad política que se requiere para abordar su enfrentamiento, será permanente en el tiempo la posibilidad de introducciones de este en nuestro país, además del riesgo de aparición de brotes de enfermedades como el dengue, ya que las relaciones de intercambio también crecen en cuanto a recursos humanos; esto hace que todos los esfuerzos y propósitos de las estrategias de control no se depriman, ganen en perfección y sean encaminados a un control integrado que nos permita mantener los índices de infestación por esta especie en niveles que no constituyan riesgos para la salud. En cuanto a otras especies como el *Anopheles*, otros culícidos, hospederos intermediarios, también muy dañinos para el hombre, y cuyo espacio en la transmisión de enfermedades crece, es importante, que las estrategias de vigilancia y lucha estén diseñadas para el control de estas y así disminuir la incidencia de casos por las enfermedades que las mismas propagan.

Misión:

La misión principal del Programa es la erradicación de mosquito *Aedes Aegypti* en todas las áreas donde aún se mantiene, evitando así la introducción y propagación de enfermedades transmitidas por este, así como la eliminación de otros culícidos, hospederos intermediarios que son también muy dañinos para el hombre.

Visión:

Mantener la erradicación de vectores en las áreas negativas perfeccionando el sistema de vigilancia e inspección evitando la reinfestación.

1.3Descripción de los procesos que se ejecutan en el campo de acción.

La Unidad Nacional de Vigilancia y Lucha Antivectorial es la entidad encargada dirigir el control de los vectores existentes en el país. Esta entidad está dividida territorialmente en:

Unidades Provinciales

Unidades Municipales

Áreas de Salud

En las áreas de salud las brigadas realizan las inspecciones a las viviendas y entidades, las revisiones de los centros priorizados y las zonas de riesgo, las larvitrampas que están instaladas en estos inmuebles, las encuestas de moscas y cucarachas, detección de focos, tratando o destruyendo todos los depósitos que puedan ser usados por el mosquito *Aedes Aegypti* como criadero, además se realizan las radiobatidas y los tratamientos químicos a las manzanas donde en las acciones de prevención los indicadores estuvieron fuera de los parámetros normales, se detectaron focos de *Aedes Aegypti* o donde fueron reportados febres.

El flujo comienza cuando el jefe del área orienta que se realicen las acciones de prevención y las acciones de control si se reportaron incidencias. Luego deben realizar el plan diario de trabajo donde se registra la cantidad de viviendas y locales que serán inspeccionados en el día todo en correspondencia a la cantidad de fuerza de trabajo disponible; asimismo realiza la repartición del trabajo e insumos entre las brigadas de su área.

Terminada la reunión de repartición los jefes de brigada en conjunto con los operarios se dirigen a las manzanas revisar las viviendas si el trabajo a realizar es inspección diaria o radiobatidas; de lo contrario si el trabajo a realizar es revisión semanal se dirigen a los centros priorizados y las zonas de riesgo a realizarle las revisiones, las encuestas de vectores a los centros priorizados y la revisión de las larvitrampas. Una vez terminado el trabajo los operarios entregan los partes al jefe de brigada para que este realice cuadre de partes y los entregue al estadístico de su área; además entrega al biólogo si fueron detectados todos los focos encontrados por los operarios.

Por otra parte, el jefe de la brigada de fumigación entrega a sus operarios las manzanas que deben ser tratadas con tratamientos químicos debido la detección de focos, los índices encuestados de moscas y cucarachas están fuera de los parámetros establecidos o debido a que fue reportado un

febril en la manzana. Terminado el trabajo al igual que los demás jefes de brigada debe cuadrar los partes y entregarlos al estadístico para su procesamiento y registro.

El proceso de procesamiento y registros de las acciones de prevención y control comienzan cuando los jefes de brigada le entrega al estadístico del área de salud los documentos con las inspecciones diarias que se realizan en las viviendas o centros de trabajo; las revisiones, las encuestas y revisión de las larvitrampas realizadas en los centros priorizados y las zonas de riesgo semanalmente, los tratamientos químicos aplicados y las radiobatidas realizadas, el estadístico al recibir los documentos de todas las brigadas debe calcular todos los indicadores para luego elaborar el documento con todos los indicadores generales del área para elaborar los diferentes reportes y entregárselo al jefe de área para que este lo registre en el documento Excel donde se encuentran todos los reportes de las acciones de prevención y control realizadas, para luego entregarlos a la Unidad Municipal.

Además, los jefes de brigada entregan al biólogo del área todos los focos detectados durante las inspecciones diarias, las revisiones a los centros priorizados, las zonas de riesgo y larvitrampas, así como los detectados en la radiobatida. El biólogo clasifica todas las muestras por etapa y especie, luego las registra en libro de muestras. Si en el proceso de clasificación existen focos de *Aedes aegypti*, el biólogo realiza el reporte de focos positivos a *Aedes aegypti* donde registra la información de estos y se lo entrega al jefe de área para que estos sean atendidos.

Diariamente el departamento de estadística del policlínico al que pertenece el área entrega a esta los febris sospechosos de dengue reportados en el Área de Salud, para que le sean realizadas las acciones correspondientes (radiobatida y tratamientos químicos) a la manzana donde se reportó el febril.

El jefe de área al día siguiente realiza copias de todos los reportes recibidos y los lleva a la Oficina Municipal de Lucha Antivectorial para entregarlos al jefe del municipio. Una vez entregados los documentos el jefe de municipio los revisa y los guarda, si se realizaron radiobatidas revisa el porcentaje de casas cerradas no sea superior 0.1 y que no existan focos, si todo está correcto certifica la radiobatida.

1.4 Análisis crítico de la ejecución actual de los procesos.

En el proceso de registro y procesamiento de las acciones de prevención, las acciones de control y los focos detectados el estadístico del área de salud debe realizar una gran cantidad de cálculos de forma manual lo cual resulta un poco engorroso y la información presenta errores, la información de estas acciones se registran en papel por parte del estadístico, lo que provoca la demora en el momento de la búsqueda de información y en las comparaciones de registros históricos. Los reportes de las acciones realizadas y de los focos de *Aedes aegypti* se registran en documentos generados utilizando Microsoft Excel sin la posibilidad de visualizar gráficos comparativos que

midan el comportamiento de las acciones de prevención y de las acciones de control aplicadas en las áreas.

Al no estar centralizados en una base de datos se ralentiza la recopilación de la información.

Cuando se va a realizar una revisión de las acciones realizadas los indicadores se obtienen del documento Excel uno por uno y de los documentos de cada una de las brigadas que se encuentran archivados lo que hace que el proceso se tarde y sea poco confiable.

1.5 Procesos objeto de automatización.

Ante la situación existente el acceso a la información se encuentra sensible a ser automatizada con la implementación de un sistema de gestión se encargue de centralizar toda la información en un mismo formato, así de esta manera se encuentran en una misma aplicación la información general del área y la de las diferentes brigadas realizando todos los cálculos de los indicadores automáticamente.

Elaboración y registro de los planes de trabajo de las brigadas:

En este proceso se realiza los planes de trabajo diario de las brigadas donde registra la cantidad de hombres disponibles para trabajar en el día, se seleccionan las manzanas que se le realizaran las acciones, así como los centros priorizados y las zonas de riesgo que se le realizaran las revisiones.

Registro de las acciones de prevención realizadas a las manzanas y a los centros priorizados y las zonas de riesgo:

En este proceso se registrarán las inspecciones diarias, las revisiones a los centros priorizados, las zonas de riesgo y las larvitrampas ubicadas en estos lugares, las encuestas de moscas y cucarachas. Las inspecciones diarias se realizan diariamente a las manzanas y solamente una vez en al ciclo, mientras que, en el caso de las revisiones de los centros priorizados, zonas de riesgo y larvitrampas, así como las encuestas de moscas y cucarachas se realizan una vez por semana.

Lo que permitirá realizar un análisis de las condiciones higiénico-sanitarias de las manzanas, los centros priorizados y zonas de riesgo inspeccionadas y basado en los resultados del trabajo realizado, se realizaran radiobatidas y tratamientos químicos en las manzanas donde se han reportado incidencias.

Registro de los focos detectados: En este proceso se registrarán todos los focos detectados en las acciones de prevención y en las radiobatidas. Esto permitirá tener un registro automatizado de los focos donde los directivos podrán obtener todos los focos detectados de *Aedes Aegypti*, así como toda la información relacionada a estos para que sean tratados.

Registro de los febris: Se registrarán los febris reportados por el departamento de estadística del policlínico donde pertenece el área. Se automatizará la obtención de análisis estadísticos, a través de gráficos y listados.

Registro de las acciones de control realizadas a las manzanas: Se registrarán todas las radiobatidas y todos los tratamientos químicos que se realizan en el área para tratar los focos detectados, febris

reportados y alta infestación de moscas o cucarachas detectadas en los centros priorizados y las zonas de riesgo. Lo que permitirá la obtención de análisis estadísticos.

Registro de las viviendas no inspeccionadas:

Se registrarán todas las viviendas que no fueron inspeccionadas y las viviendas donde el morador se negó a permitir que le realizaran la inspección o el tratamiento químico para de esta forma la cantidad y la información de las viviendas no inspeccionadas que debe recuperar cada brigada del área.

1.6 Sistemas automatizados existentes vinculados al campo de acción.

Nacionales

Software para el Sistema Integrado de Vigilancia de Dengue (SIVD).

Este programa genera una base de datos, la cual se importa el SIG para su posterior análisis con salidas mediante tablas, gráfico, indicadores y mapas temáticos, según intereses. Para el análisis se empleó el método de superposición y herramientas propias del SIG.

El producto del Software para el Sistema Integrado de Vigilancia de Dengue proporcionó información sobre la magnitud de los riesgos ambientales, la frecuencia y distribución del vector, así como los aspectos clínicos y de la vigilancia de laboratorio; de manera que permitió hacer un análisis integrado para encaminar las acciones de prevención y control tanto para el sector salud como para el resto de los organismos y organizaciones involucradas. [4]

Este sistema se basa en la gestión de información básica, no tiene en cuenta quien realiza la inspección, no se recoge los datos de los febris para tener una clara noción de todas las personas que pudieran estar infectadas para el caso en que se detecte un febril infectado. Se desarrolló con el objetivo de realizar un estudio durante un período de tiempo de 2 años en el municipio del Cerro perteneciente a la provincia La Habana, el cual permitió hacer un análisis integrado para encaminar las acciones de prevención y control para el sector salud y organizaciones involucradas; pero no se procedió a desplegar a nivel nacional.

Vigiweb

VigiWeb es un hipertexto multimedia sobre plataforma Web del tema vigilancia en salud; es decir. Está constituido bajo la filosofía Cliente-Servidor de forma tal que las funcionalidades implementadas en el lado del servidor no afecten el contenido de las páginas y de esta manera, aun cuando no se cuente con conexión en red, se podrá interactuar con el sitio en una computadora personal de forma local.

Se diseñó sobre plataforma Web para la educación postgrauada de la vigilancia en salud, el proyecto se comenzó a diseñar en el 2000. Consta de dos componentes: el producto (sitio Web), el cual puede ser adquirido mediante un soporte magnético (disco compacto, disco de 31/2, u otro soporte) o ser visitado en línea y otro que constituye el desarrollo continuo de su código fuente. [5]

Este software fue hecho solamente para la educación postgrada para consultar información, donde en una primera etapa constituyó un cuaderno de estudio de vigilancia que se brindó en soporte de papel y más tarde, en un hipertexto multimedia, con el propósito de brindar una alternativa que fortaleciera la educación postgrada de aquellos que inician su trabajo en vigilancia. Aunque permite el intercambio de estudiantes a través del correo con la autora y la consulta de información sobre la vigilancia epidemiológica, no gestiona información, es decir, sólo se puede consultar y no agregar, eliminar o modificar información.

Internacionales

SUAVE (Sistema Único Automatizado de Vigilancia Epidemiológica.) Sistema de notificación semanal de casos nuevos de enfermedades en México, sintetiza la información de los nuevos casos a través de mecanismos electrónicos a partir de su integración y captura en todas las jurisdicciones sanitarias del país. La información que genera incluye casos probables y confirmados por criterios clínicos y epidemiológicos.

Problemas encontrados

Se decidió implementar un nuevo sistema informático porque los mencionados con anterioridad no ofrecen una solución adecuada a los problemas existentes vinculados al campo de acción. Estos sistemas no presentan todas las funcionalidades necesarias para el manejo de la información requerida, ya que responden a características específicas teniendo en cuenta determinados datos y estructura. No se tiene acceso desde los diferentes niveles, y es imprescindible esta acción, para la consulta de datos, no están desarrollados con las tecnologías de software libre y se realizaron con pequeñas bases de datos que no cubren la capacidad de almacenamiento de las áreas de todo el municipio.

1.7 Tendencias y tecnologías actuales.

Para una selección apropiada de las tecnologías, se tuvo en cuenta criterios como las metodologías o procesos de desarrollo de software. Además, se estableció como política el uso de tecnologías de software libre. Se realizaron análisis detallados de las características, ventajas y desventajas entre plataformas de desarrollo, motores de bases de datos, lenguajes de programación y *frameworks* de desarrollo, con el fin de escoger los más adecuados para desarrollar el sistema informático propuesto.

1.7.1 Tipo de Aplicación

Ante la necesidad de que las Unidades Municipales de Lucha Antivectorial tenga acceso a la información de las diferentes áreas de salud, se decide la creación del sistema como una aplicación web.

1.7.2 Lenguajes de programación



Python fue creado a principios de la década de 1990 por Guido van Rossum en Stichting Mathematisch Centrum, en Holanda como sucesor del lenguaje ABC. Guido sigue siendo el autor principal de Python, aunque incluye muchas contribuciones de los demás. “Este es un lenguaje de scripting independiente de plataforma y orientado a objetos, preparado para realizar cualquier tipo de programa, desde aplicaciones Windows a servidores de red o incluso, páginas web. Es un lenguaje interpretado, lo que significa que no se necesita compilar el código fuente para poder ejecutarlo, lo que ofrece ventajas como la rapidez de desarrollo e inconvenientes como una menor velocidad. es un potente lenguaje de programación fácil de aprender. Cuenta con estructuras de datos de alto nivel, eficientes y con un enfoque simple pero eficaz para programación orientada a objetos. La elegante sintaxis y tipado dinámico, junto con su naturaleza interpretada, lo convierten en un lenguaje ideal para secuencias de comandos y el desarrollo rápido de aplicaciones en muchas áreas, en la mayoría de las plataformas”. [6]

El intérprete y la extensa biblioteca estándar están disponibles gratuitamente en formato fuente o binario para todas las plataformas desde el sitio Web de Python, y pueden ser distribuidos libremente. También puede usarse interactivamente, lo que facilita experimentar con características del lenguaje o probar funciones cuando se hace desarrollo de programas de abajo hacia arriba.

Los programas escritos en Python suelen ser mucho más cortos por varias razones:

- Los tipos de datos de alto nivel permiten expresar operaciones complejas en una sola instrucción.
- El agrupar declaraciones se hace por sangría en lugar de comenzar y terminar con paréntesis.
- Se utiliza además en aplicaciones, interfaces de usuarios, análisis de datos y estadísticas. En estos tres últimos usos, se ha posicionado como herramienta clave.

Ventajas:

- Rápido de desarrollar.
- Sencillez y velocidad.
- Sus bibliotecas hacen gran parte del trabajo.
- Soporta varias bases de datos.

Desventajas:

- Los programas interpretados son más lentos que los compilados.



PHP es un Acrónimo recursivo que significa PHP Hypertext Pre-processor (inicialmente PHP Tools o Personal Home Page Tools). Fue creado originalmente por Rasmus Lerdorf en 1994; sin embargo, la implementación principal de PHP es producida ahora por The PHP Group y sirve como

el estándar de facto para PHP al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre.

PHP es un lenguaje interpretado de propósito general ampliamente usado, diseñado especialmente para desarrollo web y que puede ser incrustado dentro de código HTML. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno.

Ventajas:

Es un lenguaje multiplataforma. Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos. El código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable. Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL. Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones). Posee una amplia documentación en su página oficial (Sitio Oficial), entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda. Es libre, por lo que se presenta como una alternativa de fácil acceso para todos. Permite aplicar técnicas de programación orientada a objetos. Biblioteca nativa de funciones sumamente amplia e incluida. No requiere definición de tipos de variables, aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución. [7]



Ruby fue lanzado públicamente en 1995 por Yukihiro “Matz” Matsumoto. Su autor combinó partes de sus lenguajes favoritos (Perl, Smalltalk, Eiffel, Ada y Lisp) y así formó un nuevo lenguaje de programación funcional con programación imperativa, Ruby. Este es un lenguaje de código abierto, orientado a objetos y multiplataforma. Se encuentra bajo las licencias de Software Libre GPL y Licencia Pública de Ruby. Permite el desarrollo de aplicaciones web y desktop. Se trata de un lenguaje de muy alto nivel interpretado y orientado a objetos no muy conocido. La sintaxis de los comandos Ruby tiene muchas similitudes con los lenguajes Perl, PHP y Python. Permite desarrollar rápidamente páginas web de cierta complejidad. Tiene integrada la interacción con las bases de datos y es fácilmente portable entre plataformas. Ruby es considerado un lenguaje flexible, ya que permite a sus usuarios alterarlo libremente. Las partes esenciales de Ruby pueden ser quitadas o redefinidas a placer. Se puede agregar funcionalidad a partes ya existentes. Ruby intenta no restringir al desarrollador. [8]

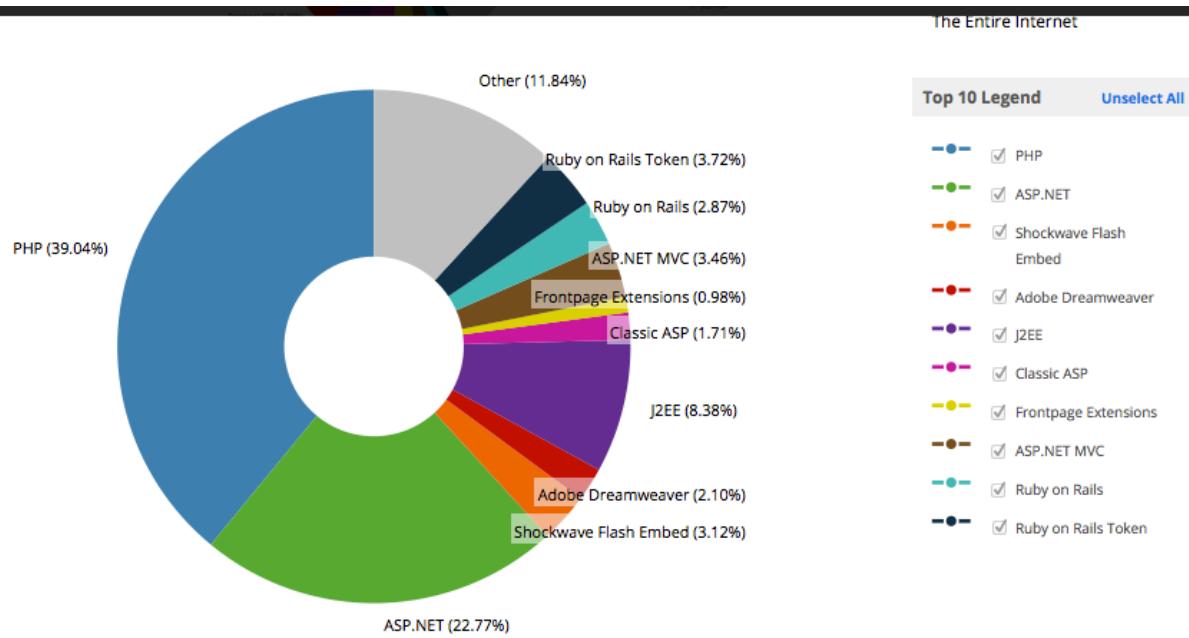


Imagen 1-1 Lenguajes de programación a nivel de servidor más utilizados en el 2018 [2]

Selección

Se ha decidido elegir como lenguaje de programación a PHP, pues posee una extensa documentación, un maduro ambiente de trabajo y es soportado por tecnologías probadas y estándares como Linux y Apache. Además, cuentan con desarrolladas bibliotecas de módulos estándares que cumplen una amplia gama de tarea. Al poseer la mayor diversidad de frameworks reconocidos y de mayor uso para el desarrollo rápido y seguro, lo hace un lenguaje idóneo para el tipo de aplicación que se requiere.

Programación del lado del cliente



jQuery

jQuery es una biblioteca JavaScript rápida, pequeña y rica en funciones. Realiza sus funciones como HTML recorrido, para la manipulación de documentos, manejo de eventos, animación, una interfaz de programación de aplicaciones (API) fácil de usar que funciona a través de múltiples navegadores. Otra de las grandes ventajas de jQuery es que se enfoca en simplificar los scripts y en acceder y modificar el contenido de una página web. [9]

De forma general los beneficios del uso de jQuery:

Utiliza una sintaxis muy parecida a CSS (hojas de estilo en cascada).

Permite manipular series de elementos y modificarlas con una simple línea de código (encadenamiento de enunciados).

Ayuda al desarrollador a concentrarse en el resultado final.

Es muy fácil de expandir, ya que cuenta con gran cantidad de plugin que se pueden utilizar o hasta crear uno propio.

Compatible con todos los navegadores modernos. [10]



Bootstrap 4

Bootstrap es un framework o conjunto de herramientas de software libre para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML (lenguaje de marcas de hipertexto) y CSS, así como, extensiones de JavaScript adicionales.

Características principales: [11]

- Permite crear interfaces que se adapten a los diferentes navegadores, tanto de escritorio como tabletas y móviles a distintas escalas y resoluciones.
- Se integra perfectamente con las principales bibliotecas de JavaScript, por ejemplo, jQuery.
- Ofrece un diseño sólido usando LESS (lenguaje dinámico de hojas de estilo) y estándares como CSS3/HTML5.
- Es un framework ligero que se integra de manera fácil con los proyectos.
- Funciona con todos los navegadores, incluido Internet Explorer usando HTML Shiva para que reconozca las etiquetas HTML5.

1.7.3 Framework de desarrollo

Framework de Php



Symfony

Este framework PHP es útil para acelerar la creación y el mantenimiento de sus aplicaciones web. Proporciona un conjunto de elementos prefabricados que se pueden integrar rápidamente en su aplicación, combinada con una metodología clara para ayudarle a trabajar de forma eficiente y eficaz en las tareas más complejas.

Principales características:

- Fácil de instalar y configurar en la mayoría de las plataformas.
- Puede elegir la versión completa *Stack* (completa) si desea desarrollar una aplicación compleja, o *Brick by Brick*, para construir su propio *framework* de acuerdo a la funcionalidad que necesita.

- Basado en MVC con variantes del clásico como la capa de abstracción de base de datos, el controlador frontal y las acciones.
- Los desarrolladores cuentan con la opción de elegir su propio *ORM*, así como los componentes Symfony se pueden incorporar a proyectos mucho más grandes como Drupal.
- Incorpora herramientas que facilitan la prueba y depuración de aplicaciones: como unidades de generación de código, pruebas del funcionamiento del *framework*, panel de depuración, interfaz por líneas de comandos y configuración en tiempo real.
- Utiliza programación orientada a objetos y características.
- Sigue la mayoría de mejores prácticas y patrones de diseño para la web.
- Ideal para aplicaciones empresariales. Es lo suficientemente estable como para desarrollar aplicaciones a largo plazo. [12]



CakePHP 3

CakePHP es un framework de desarrollo rápido de aplicaciones de código abierto en PHP. Inspirado en Rails, un framework para la construcción de sitios web que utilizan una base de datos como fuente de recursos, posee una infraestructura que tiene como finalidad permitir el desarrollo de aplicaciones web de manera ágil y estructurada, sin perder flexibilidad. Facilita al usuario la interacción con la base de datos mediante el uso de ActiveRecord.

Entre las características más destacables de CakePHP se incluyen:

- Arquitectura basada en el patrón Modelo Vista Controlador (MVC) y orientada a objetos: define clases modelo, vista y controlador con funcionalidades básicas y de las cuales heredan todas las clases que se ajustan a este patrón y que son usadas en la aplicación construida con el framework.
- Una comunidad activa de usuarios y amplia bibliografía.
- Licencia flexible: es distribuido bajo la licencia X11, más conocida entre los desarrolladores de software como MIT License.
- Compatible con PHP4 y PHP5: aunque en PHP4 se requiere especificar algunos parámetros de configuración adicionales en las clases a implementar.
- Operaciones básicas en base de datos (Creación, Obtención, Actualización y Borrado): estas operaciones están integradas para interacción con la base de datos y la simplificación de consultas.

- Despachador de peticiones: permite acceder a la aplicación a través de URLs amigables y configurables.
- de validaciones a lo largo del framework.
- Generación de plantillas de manera rápida y flexible: usando la sintaxis de PHP y con asistentes o helpers. - Incorporación de asistentes de construcción de vistas: para la automatización de la generación de código en AJAX (Asynchronous JavaScript and XML), JavaScript, formularios HTML, entre otros.
- Componentes de seguridad, manejo de sesiones y de peticiones: que reúnen las mejoras prácticas estandarizadas por la industria del software.
- Almacenamiento en caché de las vistas: para acelerar la descarga de las páginas web.
- Trabaja en cualquier subdirectorio de un servidor web: requiere poca o nula configuración del servidor Apache donde se instalará. [13]



Laravel 5.5

Laravel, es un *framework* de código abierto para desarrollar aplicaciones y servicios web con PHP. Intenta aprovechar lo mejor de otros *frameworks* y las características de las últimas versiones de PHP.

Características:

- Modular y extensible: Permite agregar todo lo que necesitas a través de su directorio Packalyst que cuenta con más de 5,500 paquetes donde puedes encontrar lo que necesitas implementado.
- HTTP rounting: Posee un sistema de enrutamiento rápido y eficiente, similar al que se usa en Ruby on Rails. Este nos permite relacionar las partes de nuestra aplicación con las rutas que ingresa el usuario en el navegador.
- Reducción de costos y tiempos en el desarrollo y mantenimiento.
- Curva de aprendizaje relativamente baja en comparación con otros *frameworks* PHP.
- Flexible y adaptable no solo al Modelo-Vista-Controlador Tradicional, sino que para reducir código propone usar “*Routes with closures*”.
- Buena y abundante documentación sobre todo en el sitio oficial.
- Posee una amplia comunidad y foros.
- Hace que el manejo de los datos no sea complejo; mediante *Eloquent*, un ORM basado en el patrón *active record*, la interacción con las bases de datos es totalmente orientada a objetos,

siendo compatible con la gran mayoría de las bases de datos del mercado actual y facilitando la migración de nuestros datos de una forma fácil y segura. Otro punto es que permite la creación de consultas robustas y complejas.

- Facilita el manejo de ruteo de nuestra aplicación como así también la generación de URL amigables y control de enlaces auto–actualizables lo que hace más fácil el mantenimiento de un sitio web.
- El sistema de plantillas *Blade* de Laravel, trae consigo la generación de mejoras en la parte de presentación de la aplicación como la generación de plantillas más simples y limpias en el código y además incluye un sistema de cache que las hace más rápidas, lo que mejora el rendimiento de la aplicación.
- También cuenta con una herramienta de interfaces de líneas de comando llamada Artisan que permite programar tareas como por ejemplo ejecutar migraciones, pruebas, entre otros.
- HTTP Middleware: El uso de Middleware se encarga de analizar y filtrar las llamadas HTTP en tu servidor. Puedes instalarlo para que se encargue de verificar que se trate de un usuario registrado, de evitar problemas de tipo Cross-Site-Scripting (XSS) y otras medidas de seguridad.
- Laravel es considerado el *framework* más popular de PHP según la comunidad de desarrolladores y *Google Trends*, destaca por sus características por encima de otros como Symfony y CakePHP. [14]

Selección

De acuerdo a lo abarcado anteriormente, se ha decidido seleccionar como framework de desarrollo para php a Laravel, porque permite mediante su directorio Packalyst seleccionar los paquetes con las funcionalidades requeridas por el sistema lo que ahorra tiempo en el desarrollo, debido a que solo hay que adaptar las funcionalidades de los paquetes a los requerimientos de la aplicación sin tener que implementarlos completamente, provee un mecanismo para filtrar las solicitudes HTTP mediante el uso de Middleware, lo cual garantiza la seguridad de la aplicación, cuenta con una potente interfaz de administración, posee una amplia documentación, tiene gran cantidad de bibliotecas que contiene, tipos de datos y funciones incorporadas en el propio lenguaje, además de ser el framework para PHP más popular y robusto en la actualidad.

1.7.4 Sistemas gestores de base de datos

PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto. Tiene más de 30 años de desarrollo activo y se ha ganado la reputación de ser confiable y mantener la integridad de los datos. Se ejecuta en la mayoría de los Sistemas Operativos (SO) más utilizados en el mundo incluyendo, Linux, varias versiones de UNIX y por supuesto Windows. Debido a sus características técnicas sobresalientes, PostgreSQL se ha ganado la admiración y el respeto de sus usuarios, así como el reconocimiento de la industria. PostgreSQL está considerado como un SGBD de software libre muy potente. Es un sistema objeto - relacional, pues incluye aspectos del paradigma orientado a objetos, tales como la herencia, tipos de datos, funciones, restricciones, triggers, reglas e integridad transaccional. Presenta llaves primarias y da la posibilidad de que los campos de una tabla sean únicos. En el caso de las llaves foráneas, permite las operaciones de borrar o modificar en cascada a partir de ellas. Este gestor funciona sobre distintas plataformas Unix y Windows.

Como muchos otros proyectos OpenSource, el desarrollo de PostgreSQL no es manejado por una sola compañía, sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo.

Es una buena alternativa a utilizar como gestor fundamentalmente en bases de datos grandes. [14] Las principales características de este gestor de bases de datos:

- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP), cadenas de bits, y otros. También permite la creación de tipos propios.
- Incorpora una estructura de datos array.
- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, y otros.
- Permite la declaración de funciones propias, así como la definición de disparadores.
- Soporta el uso de índices, reglas y vistas.
- Incluye herencia entre tablas (aunque no entre objetos, ya que no existen), por lo que a este gestor de base de datos se le incluye entre los gestores objeto- relacionales.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos. [15]

MySQL es software libre, MySQL AB distribuye una versión comercial de MySQL, que no se diferencia de la versión libre más que en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario ya que, de no ser así, se vulneraría la licencia GPL. Debido a su gran rapidez y facilidad de uso tiene gran aceptación además es fácil de instalar y configurar.

Posee las siguientes características:

- Este se ofrece bajo la licencia GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar una licencia específica que les permita este uso.
- Tiene velocidad para realizar operaciones, lo que hace uno de los gestores de mejor rendimiento.
- Posee gran popularidad ligada a PHP en el desarrollo de aplicaciones Web. En estos casos hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.
- MySQL soporta gran variedad de sistemas operativos.
- Por su conectividad y seguridad es apropiado para acceder a bases de datos en internet.
- Es fácil de instalar y configurar.
- Tiene acceso a las bases de datos de forma simultánea por varios usuarios y/o aplicaciones.
- Es un sistema multiplataforma.
- Posee amplia documentación de consulta.
- Aunque debe señalarse como punto débil, que podría provocar problemas de integridad en entornos de alta concurrencia en la modificación. [16]



Oracle es un sistema de gestión de base de datos relacional fabricado por Oracle Corporation. Es básicamente una herramienta cliente/servidor para la gestión de base de datos, es un producto vendido a nivel mundial, aunque por su gran potencia y su elevado precio hace que solo se vea en empresas grandes y multinacionales.

Surge a finales de los 70 bajo el nombre de Relational Software a partir de un estudio sobre los sistemas gestores de base de datos. En la actualidad se encuentra prácticamente en todas las industrias alrededor del mundo, siendo la primera compañía que desarrolla e implementa software

para empresa, activado por Internet a través de toda su línea de producto: base datos, aplicaciones comerciales y herramientas de desarrollo de aplicaciones y soportes de decisiones.

Características

- Es considerado uno de los sistemas de base de datos más completo y se destaca su soporte de transacciones, su estabilidad, escalabilidad y su soporte multiplataforma. Es un manejador de base de datos que hace uso de los recursos del sistema informático en todas las arquitecturas de hardware, para garantizar su aprovechamiento al máximo en ambientes cargado de información.
- Oracle es utilizado en computadoras personales, microcomputadoras, mainframe y computadoras con procesamiento paralelo masivo.
- Soporta unos 17 idiomas y corre en más de 80 arquitectura de hardware y software distintos, sin tener la necesidad de cambiar una sola línea de código. Esto es posible, ya que el 80% de los códigos internos de Oracle son iguales a los establecidos en todas las plataformas de sistema.
- El mayor problema que presenta es su elevado costo (miles de euros según versiones y licencias). Otro aspecto que ha sido criticado por algunos especialistas es la seguridad de plataforma y la política de suministro de parches de seguridad, modificadas a comienzo del 2005 y que incrementa el nivel de exposición de los usuarios. [17]

Selección

Se selecciono a MYSQL como sistema gestor de base de datos debido a que Oracle es privativo y PostgreSQL no es compatibles con las funcionalidades relacionadas con la base de datos de los paquetes desarrollados para Laravel. Por poseer una **Licencia Pública General de GNU** que garantiza a los usuarios finales (personas, organizaciones, compañías) la libertad de usar, estudiar, compartir (copiar) y modificar el software. Por su estabilidad y confiabilidad en la conservación de los datos y a que tiene gran compatibilidad con el lenguaje de programación y el framework de desarrollo seleccionados.

1.7.5 Metodología y Lenguaje utilizados

Lenguaje de modelado



UML:

UML (Unified Modeling Language, Lenguaje Unificado de Modelado en español), es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado

por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables. Se puede aplicar en el desarrollo de software entregando gran variedad de formas para dar soporte a una metodología de desarrollo de software tal como el Proceso Unificado Racional, pero no especifica en sí mismo qué metodología o proceso usar. [18]

Metodologías y Procesos de desarrollo de software:

Desarrollar software implica muchas cosas, desde su planificación hasta la puesta en marcha se deben de seguir un sinnúmero de pasos o actividades. Hoy en día existen diversas metodologías para hacerlo, en este acápite se exponen las características de los dos grandes enfoques, tanto metodologías tradicionales y metodologías ágiles, las primeras están pensadas para el uso exhaustivo de documentación durante todo el ciclo del proyecto mientras que las segundas ponen vital importancia en la capacidad de respuesta a los cambios, la confianza en las habilidades del equipo y al mantener una buena relación con el cliente. [19]

Diferencias:

Diferencias entre Metodología Tradicionales y Ágiles

Metodologías Tradicionales	Metodologías Agiles
Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo.	Basadas en heurísticas provenientes de prácticas de producción de código
Cierta resistencia a los cambios	Especialmente preparados para cambios durante el proyecto
Impuestas externamente	Impuestas internamente (por el equipo)
Proceso mucho más controlado, con numerosas políticas/normas	Proceso menos controlado, con pocos principios.
El cliente interactúa con el equipo de desarrollo mediante reuniones	El cliente es parte del equipo de desarrollo
Más artefactos	Pocos artefactos

Más roles	Pocos roles
Grupos grandes y posiblemente distribuidos	Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio
La arquitectura del software es esencial y se expresa mediante modelos	Menos énfasis en la arquitectura del software
Existe un contrato prefijado	No existe contrato tradicional o al menos es bastante flexible

Selección

Se decidió utilizar en el desarrollo del proyecto a las metodologías de desarrollo de software tradicionales debido a que permite detallar minuciosamente los procesos que se desarrollan en el negocio y de esta manera se logra una mejor comprensión de los mismos.

1.7.6 Metodologías tradicionales

Microsoft Solutions Framework:

Microsoft Solutions Framework (MSF) es un enfoque personalizable para entregar con éxito soluciones tecnológicas de manera más rápida, con menos recursos humanos y menos riesgos, pero con resultados de más calidad. MSF ayuda a los equipos a enfrentarse directamente a las causas más habituales de fracaso de los proyectos tecnológicos y mejorar así las tasas de éxito, la calidad de las soluciones y el impacto comercial. [20]

MSF se centra en:

- Alinear los objetivos de negocio y de tecnología.
- Establecer de manera clara los objetivos, los roles y las responsabilidades.
- Implementar un proceso iterativo controlado por hitos o puntos de control.
- Gestionar los riesgos de manera proactiva.
- Responder con eficacia ante los cambios.



RUP:

RUP (*Rational Unified Process*, Proceso Unificado Racional en español) es un proceso de desarrollo de software y junto con UML, constituye el proceso estándar más utilizado para el análisis, implementación y documentación de los sistemas actuales. El RUP es un proceso adaptable al

contexto y necesidades de cada organización. El ciclo de vida de RUP es una implementación del desarrollo en espiral, este organiza las tareas en fases e iteraciones y divide el proceso de desarrollo en ciclos, por lo que tiene un producto al final de cada ciclo, pues estos se dividen en fases que finalizan con un hito donde se debe tomar una decisión importante. [21]

RUP tiene tres características fundamentales:

- Dirigido por casos de uso: Los casos de uso representan los requisitos funcionales y el proceso de desarrollo marcha sobre los flujos de trabajo generados por los casos de uso.
- Centrado en la arquitectura: La arquitectura de software contiene los aspectos estáticos y dinámicos más significativos del sistema de software, surge de las necesidades de la empresa y se refleja en los casos de uso. La arquitectura es, una vista del diseño completo con las características más importantes resaltadas y deja los detalles de lado.
- Iterativo-Incremental: Cuando se está desarrollado un software, resulta cómodo dividir todo el trabajo en partes más pequeñas o mini proyectos, sobre los cuales se trabaja con más claridad. Cada mini proyecto es visto como una iteración, y esta hace referencia a pasos en el flujo de trabajo. De cada iteración se obtiene un incremento que produce crecimiento del producto para los clientes.

Win-Win Spiral Model:

El modelo en espiral WIN-WIN es una adaptación del modelo de espiral y define un conjunto de actividades de negociación al principio de cada paso alrededor de la espiral.

Este modelo está basado en la teoría W, que es una teoría de gestión de sistemas, se basa en el principio que el proyecto solo es exitoso si y solo si todos los implicados resultan ganadores. Actividades •Identificación del sistema o subsistemas clave de los directivos.

- Determinación de las condiciones de victoria de los directivos.
- Negociación de las condiciones de victoria de los directivos para reunirlas en un conjunto de condiciones para todos los afectados (incluyendo el equipo del proyecto de software). El modelo “win-win” deriva su nombre del objetivo de estas negociaciones, es decir, de "ganar-ganar". Para lograr este objetivo, el modelo define un conjunto de actividades de egociación al principio de cada paso alrededor de la espiral. El cliente recibe el producto que satisface la mayoría de sus necesidades, y el desarrollador trabaja para alcanzar presupuestos y fechas de entrega. [22]

Selección

A través de un proyecto guiado por RUP, los requerimientos funcionales son expresados en la forma de Casos de Uso, que guían la realización de una arquitectura ejecutable de la aplicación. Además, el proceso focaliza el esfuerzo del equipo en construir los elementos críticos estructuralmente y del comportamiento (llamados Elementos Arquitecturales) antes de construir elementos menos

importantes. Finalmente, RUP partitiona el ciclo de vida en iteraciones que producen versiones increm  ntales de los ejecutables de la aplicaci  n. Adem  s, emplea descripciones m  s detalladas y se adapta perfectamente a las necesidades de la organizaci  n. Su desarrollo iterativo e incremental facilita el control de cambio para necesidades posteriores de la entidad.

1.7.7 Herramientas CASE

En la actualidad existen diversas herramientas CASE para el proceso de desarrollo de software, por lo que seleccionar una se convierte en una d  f  cil tarea. Se puede definir a las herramientas CASE como un conjunto de programas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del Ciclo de Vida de desarrollo de un software.



Rose Rose es una herramienta de dise  o orientada a objetos que da soporte al modelado visual, es decir, permite representar gr  ficamente el sistema, y hace énfasis en los detalles m  s importantes. Se centra en los casos de uso y se enfoca hacia un software de mayor calidad. Emplea un lenguaje est  ndar com  n que facilita la comunicaci  n. Proporciona mecanismos para realizar la ingenier  a inversa, es decir, que a partir del c  digo se puede obtener informaci  n sobre su dise  o; adicionalmente permite generar c  digo en diferentes lenguajes sobre un dise  o en UML. Brinda la posibilidad de que varios analistas trabajen a la vez, as  i cada desarrollador opera en un espacio de trabajo privado que contiene el modelo completo y facilita que se tenga un control exclusivo sobre la propagaci  n de los cambios en ese espacio de trabajo. Soporta los diagramas de UML, excepto los diagramas de implementaci  n. [23]



Visual Paradigm para UML es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software: an  lisis y dise  o orientados a objetos, construcci  n, pruebas y despliegue.

Posee generaci  n de c  digo fuente para alrededor de diez lenguajes de programaci  n de forma autom  tica, reduciendo los tiempos de desarrollo y evitando errores en la codificaci  n del software. Permite adem  s aplicar ingenier  a inversa a aplicaciones desarrolladas en dichos lenguajes. Posibilita la importaci  n de ficheros de otras herramientas CASE, interpretando ficheros con extensi  n XML y MDL. Posee un entorno que permite especificar los detalles de los casos de uso, incluyendo la especificaci  n del modelo general y de las descripciones de los casos de uso. Hace

posible la generación de bases de datos mediante la transformación de diagramas de entidad-relación en tablas de bases de datos. [24]



Enterprise Architect

Tiene probada la habilidad para capturar y controlar la complejidad de los procesos y desempeña un papel fundamental de confianza en la entrega de alta calidad de soluciones. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación. Elegida por miles de empresas, organismos y organizaciones de todo el mundo. La capacidad de Enterprise Architect se demuestra una y otra vez. Tiene herramientas personalizadas e integradas para analizar y visualizar software en ejecución, con simulación avanzada, herramientas de prueba, repositorios de trabajo en equipo, control de versiones y más. Capaz de gestionar los requisitos, los modelos estratégicos y modelos de análisis y procesos. Permite elaborar todos los artefactos en el lenguaje UML de forma fácil. [25]

Selección:

Se ha determinado utilizar la herramienta CASE, ya que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Permite realizar todos los tipos de diagramas, generar código desde diagramas y generar documentación. También proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML.

1.8 Análisis crítico de las fuentes y bibliografías utilizadas (Estado del arte).

Durante la presente investigación fueron consultadas diversas fuentes bibliográficas, entre las que se encuentran: libros, manuales y artículos digitales de sitios confiables en internet relacionados con el tema. Estas son de la autoría de destacados profesores o reconocidos especialistas que a su vez son referenciados en otros artículos demostrando su popularidad en el ámbito en que desarrollan.

Los manuales suministrados por parte de la entidad, Texto Básico para el Curso de Operarios y Técnicos Auxiliares en Vigilancia y Lucha Antivectorial y CAMPAÑA ANTI-AEDES AEGYPTI, Manual para responsables de brigada y visitadores son documentos de hace más de 8 años, pero debido a la no existencia ni actualización de los documentos antes mencionados se encuentran vigentes, además presentaban problemas en la redacción.

Por otra parte, asignaturas de la carrera como Programación Web, Base de Datos, Ingeniería de Software (I, II y III) y Práctica Profesional (I, II y III), representaron un singular apoyo, brindando los elementos teóricos y prácticos necesarios, estos constituyen una de las principales fuentes de

estudio durante todo el período de formación como ingenieros. Hay que aclarar que la bibliografía consultada es actual ya que la mayoría es de no más de 2 años.

1.9 Conclusiones parciales del capítulo.

Al realizar el estudio de los procesos actuales se pudo constatar que existe la necesidad de un sistema informático que proporcione la solución a los problemas existentes. Al no contar con un software que cumpla con las características requeridas por la entidad. Como parte de este estudio se lograron recopilar todos los modelos que se utilizan en dichos procesos.

- Se construyó el marco conceptual que soporta la investigación y se adquirió el conocimiento necesario sobre los métodos de procesos de software, herramientas y lenguajes, seleccionándose los más adecuadas para el cumplimiento del objetivo general propuesto.
- Se identificó como modelo de proceso de desarrollo de software RUP con el lenguaje unificado de modelado UML, puesto provee un marco metodológico integrado de principios, prácticas y técnicas concretas que se ajustan al entorno y/o condiciones del proyecto y tienen el potencial de detallar los procesos que se realizan en la entidad mediante los diferentes artefactos.
- Se decidió realizar una aplicación web, empleando MySQL como sistema gestor de bases de datos, el lenguaje de programación PHP y como marco de trabajo a Laravel.

Capítulo 2 Modelo de negocio

2.1 Introducción

En el presente capítulo se abordará lo referente a la modelación del negocio, para un mejor entendimiento de la estructura y dinámica del mismo. Se hará uso de los principales artefactos que propone RUP en su disciplina Modelación del Negocio. Se presentarán los diagramas de casos de uso del negocio, sus descripciones, diagramas de actividades modelo de objeto, actores y trabajadores del negocio.

2.2 Modelo del negocio actual

En la actualidad el negocio se inicia cuando el jefe de área realiza el plan de trabajo del día, este plan es creado de acuerdo a las incidencias reportadas el día anterior. Si se detectaron focos de *Aedes Aegypti* o se reportaron febris el jefe de área prioriza las acciones de control y si hay disponibilidad de personal se continua con las acciones de prevención.

Como antes mencione si no existieron incidencia o si existe personal disponible orienta realizar las inspecciones diarias de las manzanas, la revisión de las zonas de riesgo, centros priorizados, larvitrampas y la realización de las encuestas de vectores en los centros priorizados. El jefe de área le comunica en el pase de revista a los jefes de brigadas que se tiene que realizar el trabajo antes mencionado, además se les entregan a los operarios los insumos necesarios para realizar su trabajo. Las inspecciones a las viviendas de las diferentes manzanas se realizan diariamente y las manzanas solo pueden ser inspeccionadas una única vez en el ciclo, mientras que todos los centros priorizados y zonas de riesgo serán visitados con una periodicidad semanal.

Por otra parte, si se detectaron incidencias se deben realizar las acciones de control que consisten en realizarle a las manzanas donde se detectaron problemas tratamientos químicos y radiobatidas. Los tratamientos químicos se realizan cuando los índices de las encuestas están fuera de los parámetros normales, cuando se encuentran focos de mosquito *Aedes Aegypti* y también cuando se reportan febris, en cuanto a las radiobatidas se orientan solo cuando son detectados focos y cuando se reportan febris. Estas dos actividades comienzan cuando el jefe de área orienta la realización de las acciones de control por los motivos antes explicados, luego el jefe de brigada reparte el trabajo entre los operarios de la brigada los cuales deben dirigirse hacia la manzana que le fue asignada a realizar la acción.

Si el trabajo que se le asignó al operario es el de realizar una radiobatida se realizan las inspecciones de las manzanas con incidencias. En el caso del tratamiento adultecita el operario realiza una fumigación con la llamada bazuca a todas las habitaciones de la vivienda o locales de los centros estatales.

Al terminar la jornada de trabajo los operarios entregan los partes y los focos con su respectiva etiqueta a los jefes de brigada para que este realice la revisión de estos, para entregarlos al estadístico y al biólogo del área de salud.

Los estadísticos de las áreas de salud realizan cálculos estadísticos con la información suministrada por los jefes de brigada, registra estos datos en formato físico (papel) y realiza el reporte de las inspecciones de las manzanas, las revisiones a las zonas riesgo y centros priorizados, el reporte de las encuestas de moscas y cucarachas y el reporte de las larvitrampas. Las revisiones de los centros priorizados, las zonas de riesgo y las encuestas son registradas en el expediente de cada centro priorizado o zona de riesgo.

La clasificación de los focos es realizada por el biólogo del área de salud cuando los jefes de brigadas le entregan las muestras con todos los datos de estas, luego el biólogo registra la etapa, la especie, la dirección, la manzana, el área de salud, la fecha de la muestra en el libro de focos o modelo 91-12 y realiza el reporte de focos positivos a *Aedes Aegypti* y se lo entrega al jefe de área.

El jefe de área al recibir estos reportes los revisa y los registra en el documento Excel de reportes. Si en la revisión de estos documentos existen parámetros que están fuera de lo establecido como focos de *Aedes Aegypti* o los índices de infestación de las encuestas están por encima lo establecido, el jefe de área debe orientar la realización de las acciones de control (los tratamientos químicos y las radiobatidas) las que deben aplicarse a la manzana donde está ubicada la entidad o la vivienda.

Por otra parte, el departamento de estadística de los policlínicos entrega a los jefes de área la información de los febris que son detectados en el área para que se le realice las radiobatidas y el tratamiento químico correspondiente. El jefe de área registra esta información de los febris en papel.

El jefe de área al día siguiente realiza copias de todos los reportes recibidos y los lleva a la Oficina Municipal de Lucha Antivectorial para entregarlos al jefe del municipio. Una vez entregados los documentos el jefe de municipio los revisa y los guarda, si se realizaron radiobatidas revisa el porcentaje de casas cerradas no sea superior 0.1 y que no existan focos, si todo esta correcto certifica la radiobatida.

2.3 Reglas del negocio a considerar

- A los centros priorizados y las zonas de riesgo es obligatorio realizarles las revisiones semanalmente.
- Las viviendas de las manzanas solo pueden ser inspeccionadas una vez en el ciclo.
- Si es detectado un foco en una larvitrapa esta debe ser retirada inmediatamente y llevada para el área de salud para que le apliquen los procedimientos establecidos.

- Todos los focos que se detecten deben ser capturados, etiquetados correctamente y entregados al biólogo del área de salud para su clasificación y registro en el modelo 91-12.
- Si en la clasificación de los focos detectados se confirma la existencia de focos de *Aedes Aegypti* se tiene que informar rápidamente al jefe de área para que tome las decisiones correspondientes.
- Si en la revisión de un centro priorizado o una zona de riesgo es detectado un foco de *Aedes Aegypti* se debe realizar una radiobatida y el tratamiento químico correspondiente a la manzana donde está ubicada la entidad.
- Cuando se va a realizar una radiobatida se tienen que retirar todas las larvitrampas de la manzana donde se va a realizar la actividad antes mencionada.
- Todas las inspecciones, las radiobatidas, tratamiento adulticidas, los focos, las encuestas y las revisiones emitidos en los partes deben tener asociada la manzana y el área de salud.
- Los biólogos de las áreas de salud son los únicos autorizados a registrar los focos en el modelo 91-12.
- Si los índices de las encuestas de moscas y de cucarachas están por encima de 12 y de 10 respectivamente se debe realizar un tratamiento adulticida a la manzana donde está ubicada la entidad.
- El estadístico del área de salud es el único autorizado realizar los cálculos de los indicadores en todas las acciones registradas, registrar toda la información con respecto a dichas acciones y elaborar los diferentes reportes.
- Cuando un febril es reportado por parte del departamento de estadística de los policlínicos se le tiene realizar una radiobatida y el tratamiento adulticida a la manzana donde vive el febril.
- Diariamente se tienen que realizar los planes de trabajo de cada brigada.
- Las acciones realizadas tienen que ser registradas el mismo día que se hicieron.
- Las acciones realizadas y las incidencias reportadas deben permanecer archivadas por un año.
- Las brigadas tienen zonas fijas a su cargo.
- Las viviendas cerradas en las inspecciones, en las radiobatidas y en los tratamientos químicos no pueden superar el 30%, 8% y 5% de las viviendas existentes respectivamente.

2.4 Actores del negocio

Tabla 2.1 Descripción de los actores del negocio

Actor del negocio	Descripción
Jefe municipal	Directivo recibe los resultados de las acciones de prevención y las acciones de control realizadas para el análisis en la reunión del

	puesto de mando que se realiza semanalmente.
Departamento de estadística del policlínico	Este actor entrega al jefe de área el listado con la información de los febres reportados y le solicita que se le realicen las acciones de control correspondientes.

2.5 Diagrama de casos de uso del negocio

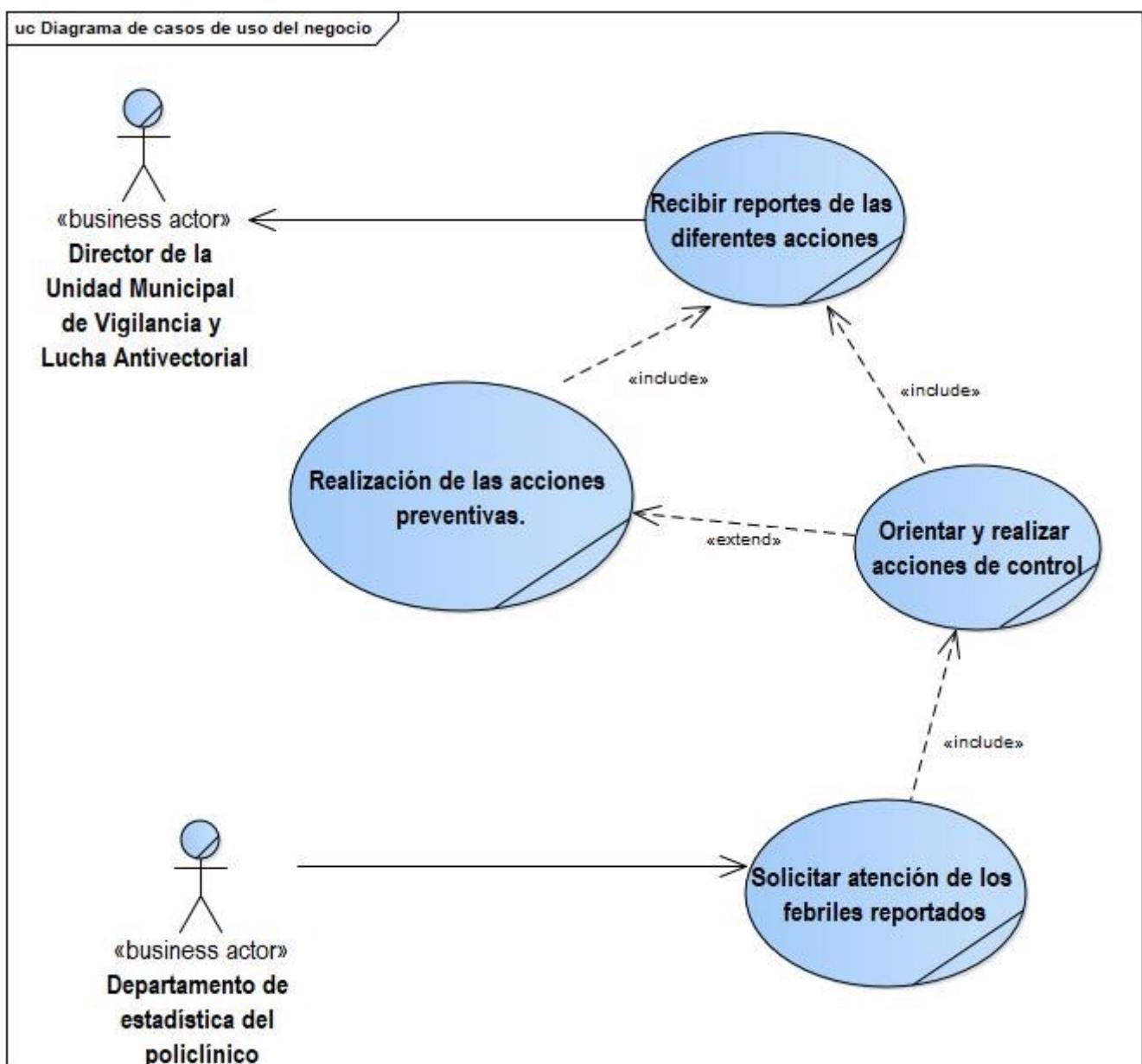


Imagen 2-1 Diagrama de Casos de Uso del Negocio

2.6 Trabajadores del negocio

Tabla 2 Trabajadores del negocio

Trabajador del negocio	Descripción
Estadístico del área de salud	Tiene la responsabilidad de llevar las estadísticas de las acciones de control y de las acciones de prevención, además de elaborar los reportes de las diferentes acciones que se realizan para erradicar vectores.
Biólogo	Tiene la responsabilidad de llevar las estadísticas de los indicadores con respecto a los focos, la clasificación y registro de los mismos.
Jefe de Área	Tiene la responsabilidad orientar las diferentes acciones, de revisar los documentos emitidos por el estadístico del área de salud y el biólogo para el registro de los mismos en el documento Excel de partes, determina si existe algún indicador fuera de los parámetros establecidos para tomar las decisiones correspondientes y entrega los diferentes reportes al Director de la Unidad Municipal de Vigilancia y Lucha Antivectorial, además es el máximo responsable en el área de salud de suministrar a los operarios todos los recursos necesarios para realizar correctamente su trabajo. Además, es el encargado de registrar a todos los febriles reportados.
Operarios	Los operarios son los encargados de realizar las acciones de prevención y las acciones de control a las viviendas de las manzanas, a los centros priorizados y las zonas de riesgo .
Jefe de brigada	Es el encargado de recibir los partes, los focos de los operarios, revisar que todo este correcto y entregárselos al estadístico y al biólogo del área de salud. Y controlar que los operarios estén haciendo correctamente su trabajo.

2.7 Casos de uso del negocio

2.7.1 Caso de uso Realización de las acciones preventivas en las viviendas, en los centros priorizados y las zonas de riesgo

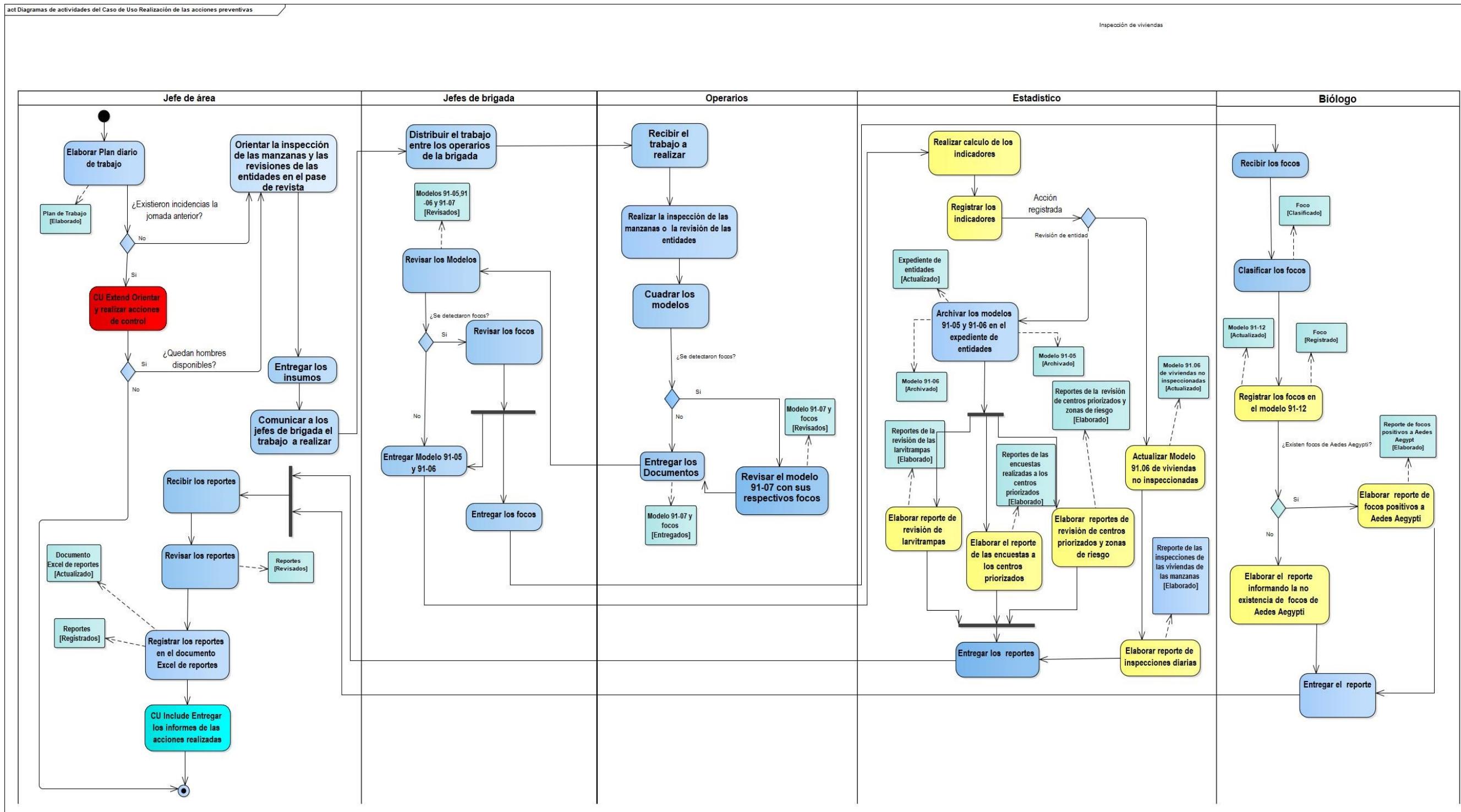


Imagen 2-2 Diagramas de actividades del Caso de Uso Realización de las acciones

2.7.2 Caso de uso Orientar y realizar acciones de control

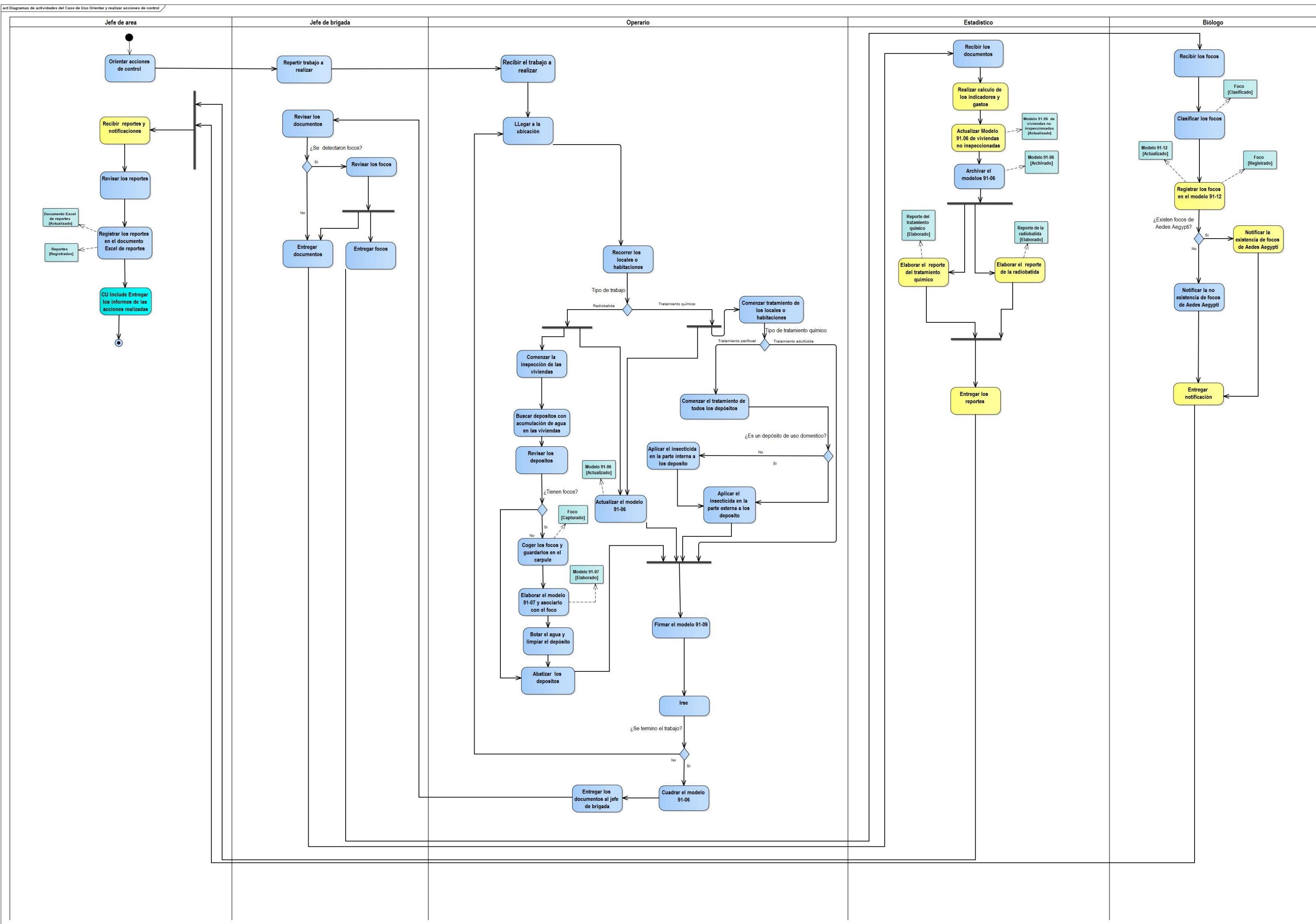


Imagen 2-3 Diagramas de actividades del Caso de Uso Orientar y realizar acciones de control

2.7.3 Caso de uso Solicitar atención de los febriles reportados

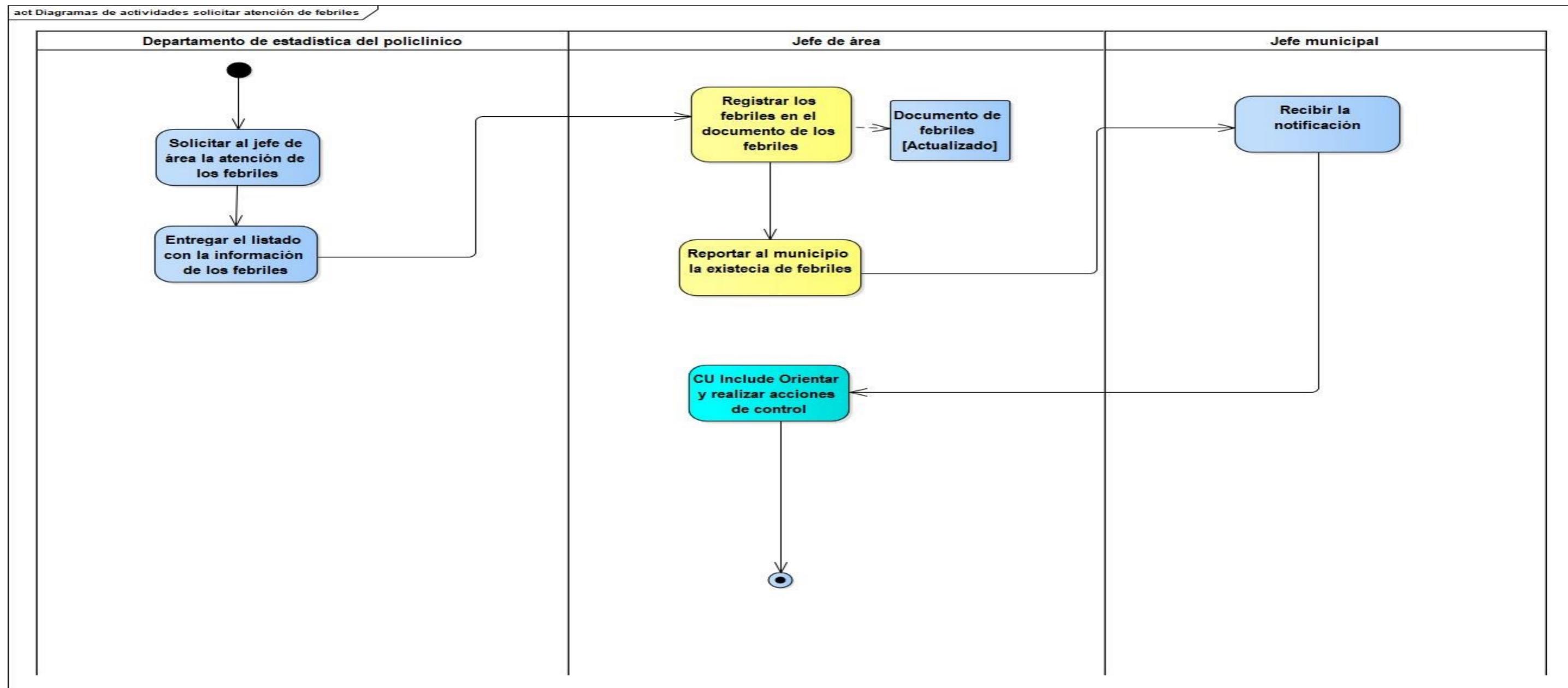


Imagen 2-4 Diagramas de actividades del Caso de Uso Solicitar atención de los febriles reportados

2.7.4 Caso de uso Recibir los informes de las acciones realizadas

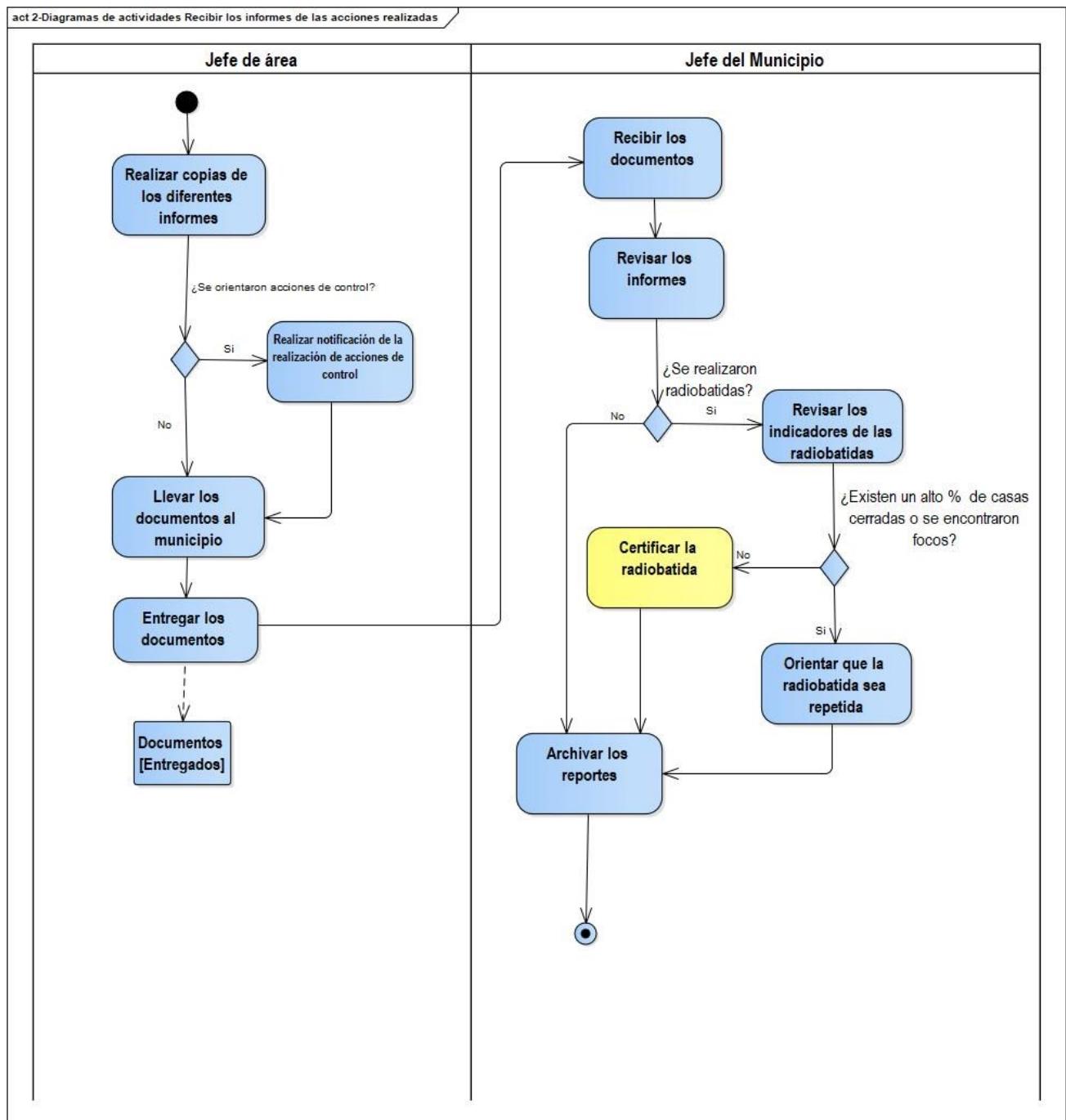


Imagen 2-5 Diagramas de actividades del Caso de Uso Recibir los informes de las acciones realizadas

2.8 Modelo de Objetos

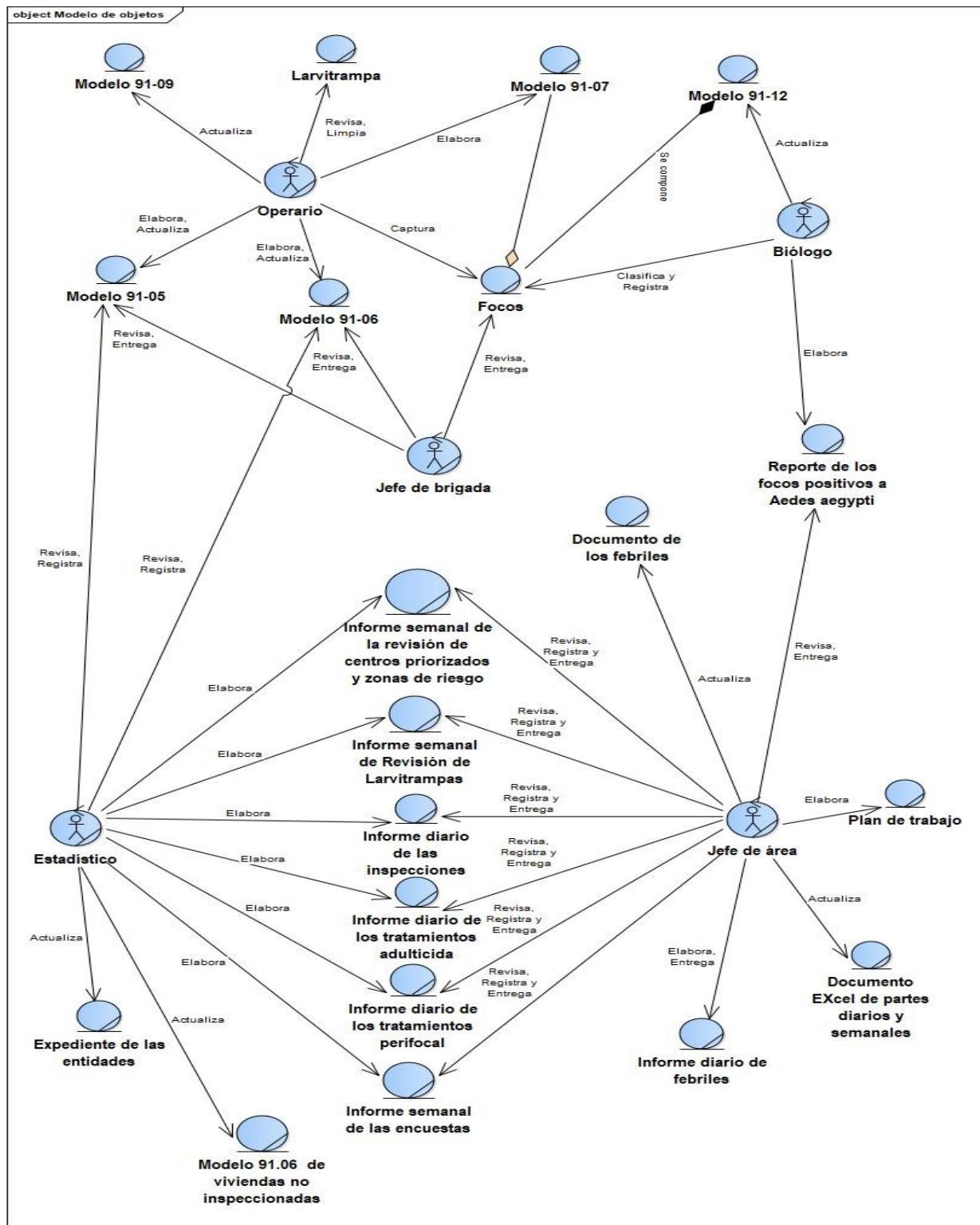


Imagen 2-6 Modelo de Objeto

2.8.1 Descripción de las entidades del negocio

Tabla 5 Descripción de las entidades del negocio

Entidad	Descripción
Plan de trabajo	Documento donde se planifica el trabajo a realizar en el día, donde se registra las brigadas, la cantidad de operarios disponibles y las manzanas donde se realizan las acciones de prevención y control.
Modelo 91-06	Documento donde el operario registra el trabajo realizado (las inspecciones de las manzanas, las revisiones de los centros priorizados y las zonas de riesgo, las revisiones de las larvitrampas).
Modelo 91-05	Documento donde el operario registra las encuestas realizadas a los centros priorizados y las zonas de riesgo.
Informe diario del tratamiento adulticida	Documento que almacena los datos de los tratamientos adulticida realizados en el área de salud y que es elaborado por el Estadístico del área de salud.
Informe diario de las radiobatidas	Documento que almacena los datos de las radiobatidas realizadas en el área de salud y que es elaborado por el Estadístico del área de salud.
Documento Excel de partes	Documento donde el jefe de área de salud registra los partes elaborados por el Estadístico del área de salud y el biólogo.
Informe diario de las inspecciones	Documento que solicita el directivo y que almacena la información de las inspecciones realizadas a las manzanas.
Informe semanal de las revisiones realizadas a los centros priorizados y las zonas de riesgo	Documento que solicita el directivo y que almacena la información de las revisiones realizadas a los centros priorizados y las zonas de riesgo.
Informe semanal de las encuestas	Documento que solicita el directivo y que almacena la información de las encuestas realizadas a todos los centros priorizados.
Foco	Muestra detectada por las brigadas de área de salud la cual tiene que ser clasificada y registrada.

Modelo 91-07	Documento que elabora el operario y registra la información del foco detectado (lugar, dirección, manzana, tipo de depósito). Este documento se asocia al foco. Y que sirve de guía al biólogo para obtener la información del foco.
Modelo 91-12	Documento donde el Biólogo registra los datos de los focos detectados.
Reporte de focos positivos a Aedes Aegypti	Documento que solicita el directivo y que almacena los datos de los focos positivos a <i>Aedes Aegypti</i> y que es elaborado por el Biólogo.
Larvitrampas	Herramienta que es utilizada para detectar la existencia del mosquito <i>Aedes Aegypti</i> en las localidades donde están ubicadas, son revisadas por los operarios y retiradas por estos cuando se detectan focos en ella o si se va a realizar una radiobatida en esa manzana.
Expediente de los centros priorizados y las zonas de riesgo	Documento donde se archivan el modelo 91-06 y el 91-05 que contienen las revisiones y las encuestas que se realizan los centros priorizados y las zonas de riesgo.
Documento de los febriles	Documento donde los jefes de área registran la información de los febriles reportados por parte de los departamentos de estadística de los policlínicos.
Modelo 91.06 de viviendas no inspeccionadas	Documentos donde el estadístico registra viviendas negadas y las viviendas cerradas de las inspecciones y las radiobatidas.

2.9 Conclusiones parciales del capítulo

Luego de realizar un análisis de los principales conceptos que intervienen en el negocio, se logró obtener una propuesta de solución que dará respuesta a la problemática de la investigación. Por ello se concluye que:

- Se identificaron y documentaron los casos de uso del negocio que representan los procesos de la gestión de las acciones de prevención y control de vectores, lo que permitió comprender como se desarrollan los procesos en la entidad, establecer los procesos o actividades que serán objeto de automatización y exponer los estados por los que pasan los objetos en el transcurso de los procesos.

- Especificar los actores y trabajadores del negocio permitió determinar las acciones que ejecutan y aclarar las responsabilidades de cada uno, para de esta forma poder obtener los actores que serán parte del sistema.
- Identificar las reglas del negocio posibilita identificar las restricciones que regirán el funcionamiento del sistema.
- Identificar las entidades del negocio y su relación con los actores del negocio permite obtener las interacciones que tendrán los trabajadores del negocio con las entidades del negocio en el sistema.

Capítulo 3 Requisitos.

3.1 Introducción

Los requisitos son condiciones o capacidades que debe cumplir un sistema. La captura de requisitos tiene dos objetivos fundamentales: encontrar los verdaderos requisitos, aquellos que cuando se implementen añadirán el valor esperado para los usuarios, y representar dichos requisitos de un modo adecuado y comprensible para usuarios, clientes y desarrolladores. [26] Este capítulo está encaminado a desarrollar una estructura funcional del sistema que se pretende construir, este modelo será concebido utilizando los casos de uso, elaborando el modelo de casos de uso del sistema. Además, se identificarán los paquetes que conforman la solución propuesta y sus relaciones, así como los actores, requisitos funcionales y no funcionales y se describirán los casos de uso del sistema para una mayor comprensión de estos.

3.2 Definición de los requisitos funcionales

Requisitos Funcionales.

Paquete de Seguridad

- Autenticar usuario.
- Gestionar usuarios
- Asignar al usuario el área de salud
- Cambiar la contraseña.
- Visualizar el historial del sistema.
- Salva y Restaura de la base de datos.

Paquete de Gestión

- Gestión de los planes de trabajo diario de las brigadas.
- Gestión de las revisiones realizadas a los Centros priorizados y Zonas de riesgo.
- Gestión de encuestas de moscas y cucarachas.
- Gestionar las larvitrampas.
- Gestionar los centros priorizados y las zonas de riesgo.
- Gestionar las inspecciones.
- Gestionar los focos detectados.
- Gestionar los febris reportados.
- Gestionar las viviendas no inspeccionadas
- En dependencia de la detección de focos de *Aedes Aegypti* y/o de febris y si están planificadas las acciones
 - Gestionar las radiobatidas.
 - Gestión de los tratamientos adulticida.

Paquete Nomencladores

- Gestión de Nomencladores
 - Nomencladores de las áreas: El jefe de área podrá insertar, modificar y eliminar todos los nomencladores pertenecientes a su área. Ellos son:
 - ✓ Insumos
 - ✓ Manzanas
 - ✓ Brigadas.
 - ✓ Zonas
 - Nomencladores genéricos: El jefe del municipio podrá insertar, modificar y eliminar los nomencladores genéricos para todas las áreas.
 - ✓ Ciclo
 - ✓ Área de Salud
 - ✓ Especies de mosquitos

Paquete de Reportes

- El sistema permitirá generar los siguientes reportes:

Reportes para el jefe de área

- ✓ Generar gráfico de cantidad y porcentaje de casas cerradas por brigada.
- ✓ Reporte gráfico con los focos agrupados por la especie detectados en el ciclo.
- ✓ Informe con las manzanas inspeccionadas en el ciclo actual.
- ✓ Generar reporte con la comparación entre la revisión más reciente y otra previa, las encuestas de moscas y cucarachas.
- ✓ Generar reporte gráfico con los focos detectados y la cantidad por especie de una revisión seleccionada
- ✓ Generar reporte gráfico con el porcentaje de casas cerradas en una radiobatida seleccionada.
- ✓ Generar listado de viviendas no inspeccionadas de una brigada seleccionada.

Reportes para el jefe municipal

- ✓ Generar reporte gráfico con el índice de infestación a Aedes Aegypti del ciclo actual en el municipio.

Reportes comunes para ambos actores

- ✓ Generar reporte configurable con las acciones realizadas e incidencias detectadas en una fecha seleccionada por el usuario.
- ✓ Generar reporte configurable con las acciones realizadas e incidencias detectadas en el ciclo.

- ✓ Generar reporte gráfico de las manzanas de un área con mayor infestación a Aedes Aegypti en un Ciclo.
 - ✓ Generar reporte configurable con las acciones realizadas e incidencias detectadas en los últimos siete días.
 - ✓ Generar reporte gráfico de la comparación entre Ciclos con respecto al índice de infestación de Aedes Aegypti en un Área.
 - ✓ Generar reporte gráfico de la comparación entre Ciclos con respecto a los Febris.
- Enviar notificaciones o alertas sobre la ocurrencia de eventos:
 Permite al jefe de área y al jefe municipal conocer mediante aviso (alerta) enviados por el sistema sobre la ocurrencia de eventos. Ellos son:
- Si se detecta un foco de *Aedes Aegypti* y no ha sido atendido.
 - Si los índices de las diferentes encuestas que se realizan a los centros priorizados están por encima del valor establecido.
 - Cuando una larvitrapas es retirada.
 - Cuando una radiobatida no puede ser certificada.
 - Cuando se termine cualquiera de los insumos.
 - Cuando es reportado un febral y no ha sido atendido.
 - El sistema notifica al jefe municipal que debe crear un nuevo ciclo cuando termina el anterior y al jefe de área que realice el plan diario.
 - Cuando es creado un nuevo usuario en el municipio y este no tiene área asignada.
- Exportar a PDF los informes: Le brindará al usuario exportar la información de los informes diarios en formato PDF.

3.3 Actores del sistema a automatizar

Tabla 3.1 Actores del sistema a automatizar

Nombre del actor	Descripción del actor
Usuario	Actor genérico que hace uso de funcionalidades como autenticarse, cambiar contraseña, las cuales son comunes a todos los actores, y consecuentemente se crea para que estos hereden de él su funcionalidad.
Estadístico	Se encarga de gestionar las acciones de control (las inspecciones, las revisiones de los centros priorizados y las zonas de riesgo, revisiones de las larvitrapas y las encuestas de moscas y cucarachas), las acciones de

	control (el tratamiento adulticida y las radiobatidas realizadas a las manzanas). A partir de estos datos el sistema calcula automáticamente los indicadores y se elaboran los reportes.
Biólogo	Gestiona lo relacionado a los focos y a partir de estos datos el sistema genera automáticamente los reportes de focos positivos a <i>Aedes Aegypti</i>
Administrador municipal del sistema	Encargado de gestionar los usuarios y de visualizar el historial de acciones.
Administrador provincial del sistema	Es el encargado de gestionar los administradores municipales y las salvas del sistema.
Jefe genérico	Actor que hace uso de los reportes que son comunes para el jefe municipal y para los jefes de área.
Jefe de área	Es el encargado gestionar los nomencladores de su área, así como visualizar los diferentes reportes.
Jefe municipal	Visualiza todos los reportes generados para facilitar la toma de decisiones, se en carga de registrar los nomencladores genéricos del sistema. Además, asignará a los usuarios no administradores de su municipio el área de salud.

3.4 Jerarquía de actores

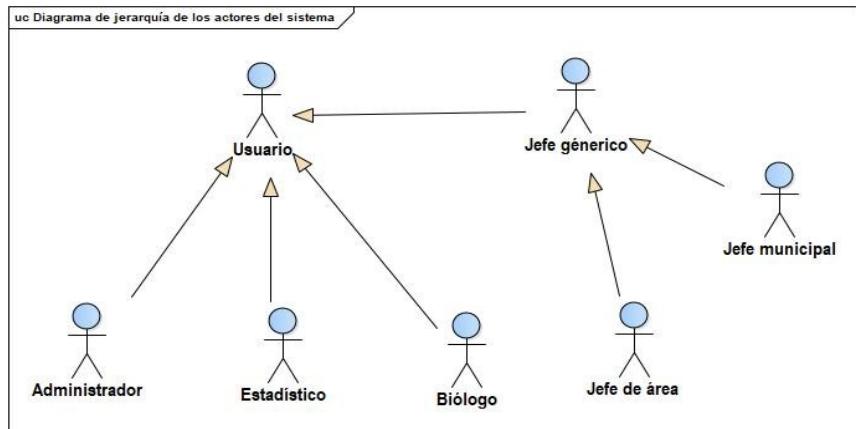


Imagen 3-1 Jerarquía de actores del sistema

3.5 Paquetes y sus relaciones

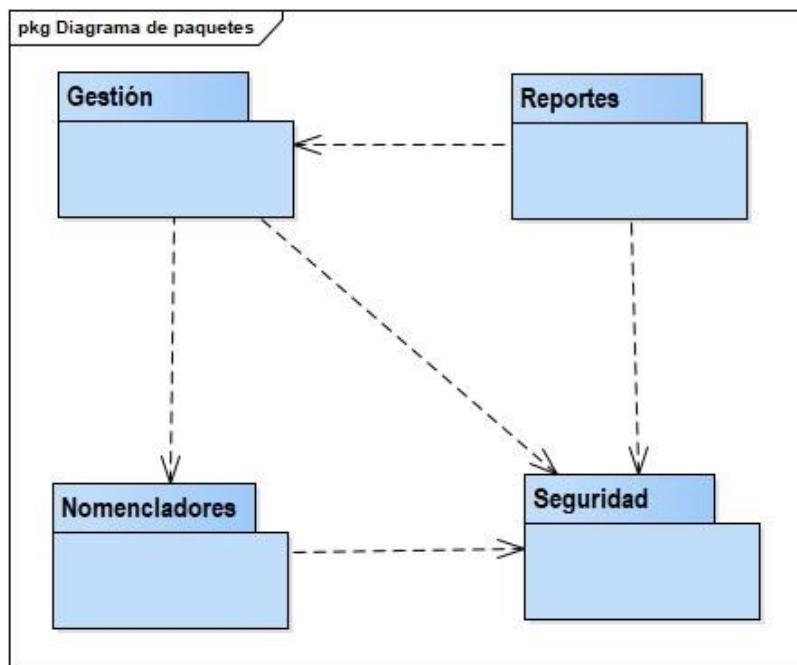


Imagen 3-2 Paquetes y sus relaciones

Paquete Seguridad:

Grupa las funcionalidades que permitan mantener la seguridad de la aplicación, permisos de usuarios según sea el caso, además de la autenticación y registro. Establece los niveles de acceso según los roles especificados. Recoge las salvas y las trazas del sistema. Los administradores municipales son los encargados de gestionar a los usuarios no administradores de su municipio sin asignarle el área de salud, mientras que de esta funcionalidad se encargan los jefes municipales. Por otra parte el administrador provincial es el encargado de gestionar los administradores y realizar las salvas y restauración de la base de datos del sistema. El resto de los usuarios solo podrá

autenticarse en el sistema y realizar el cambio de su contraseña.

Paquete Gestión:

Agrupa las funcionalidades relacionadas con la gestión de las acciones de prevención, las acciones de control, los febris, los focos y las larvitrampas. Donde el jefe de área es el encargado de gestionar los planes de trabajo y los febris, el estadístico es el encargado gestionar las radiobatidas, las inspecciones, las revisiones y todas las acciones de prevención y control en general y por último el biólogo del área es el encargado gestionar los focos detectados.

Paquete Reportes:

En él, se especifican las funcionalidades que permiten la obtención de informes, necesarios para la toma de decisiones de los usuarios del sistema.

Paquete Nomencladores:

Agrupa las herramientas relacionadas con la gestión de los nomencladores del sistema.

3.6 Diagramas de casos de uso del sistema

3.6.1 Diagrama de casos de uso del paquete Seguridad

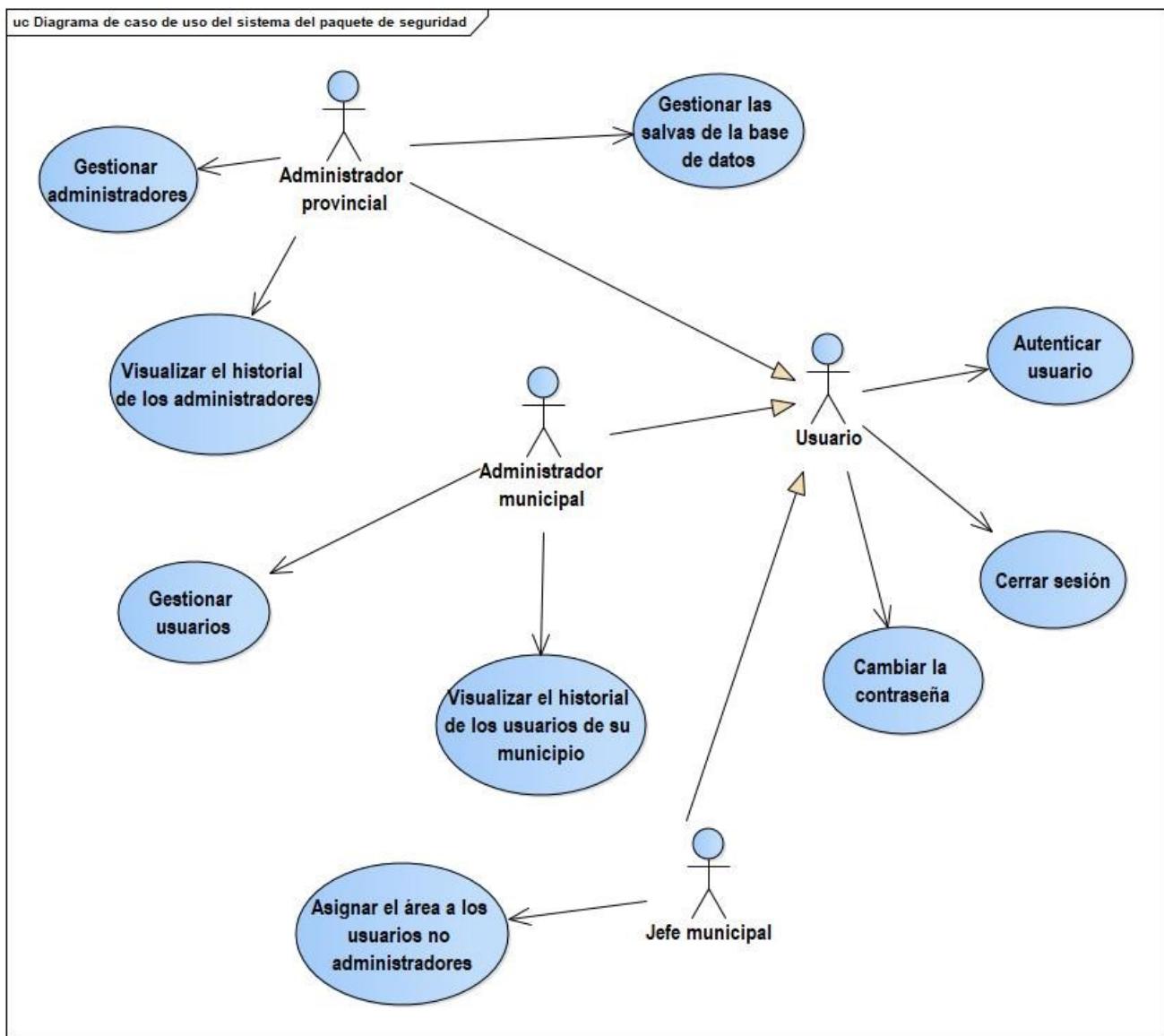


Imagen 3-3 Diagrama de casos de uso del paquete Seguridad

3.6.2 Diagrama de casos de uso del paquete Gestión

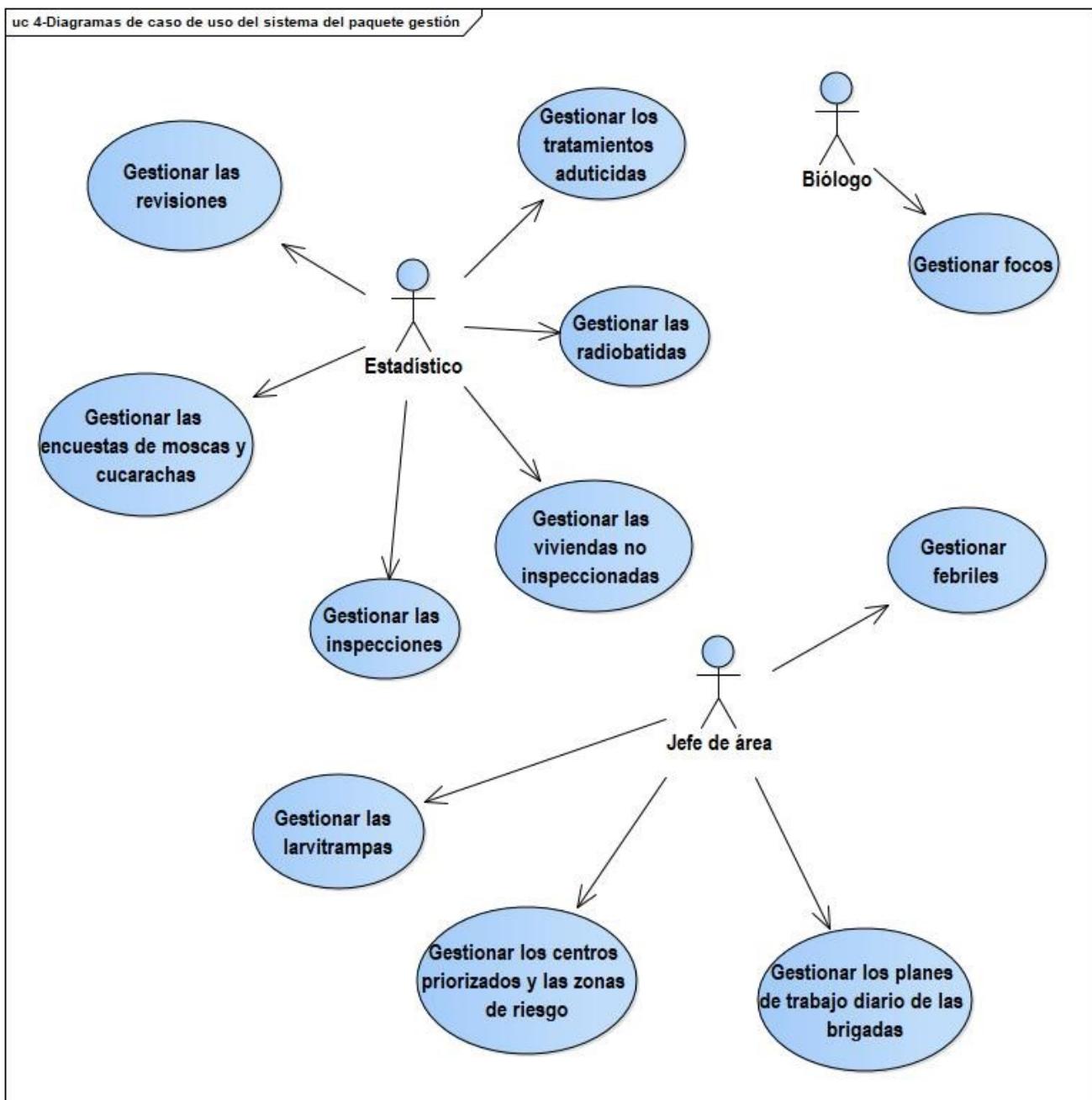


Imagen 3-4 Diagrama de casos de uso del paquete Gestión

3.6.3 Diagrama de casos de uso del paquete Nomencladores

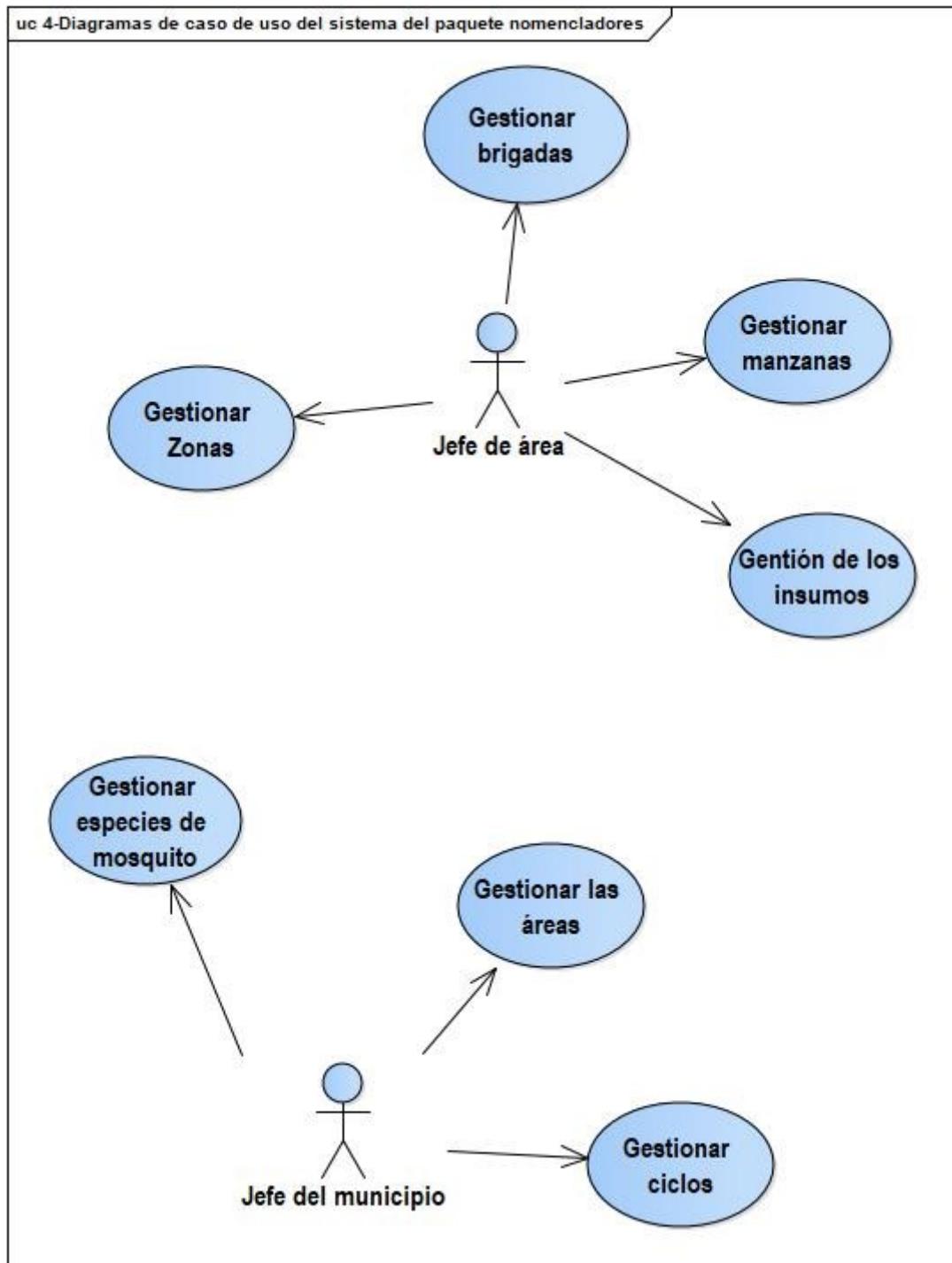


Imagen 3-5 Diagrama de casos de uso del paquete Nomencladores

3.6.4 Diagrama de casos de uso del paquete Reportes

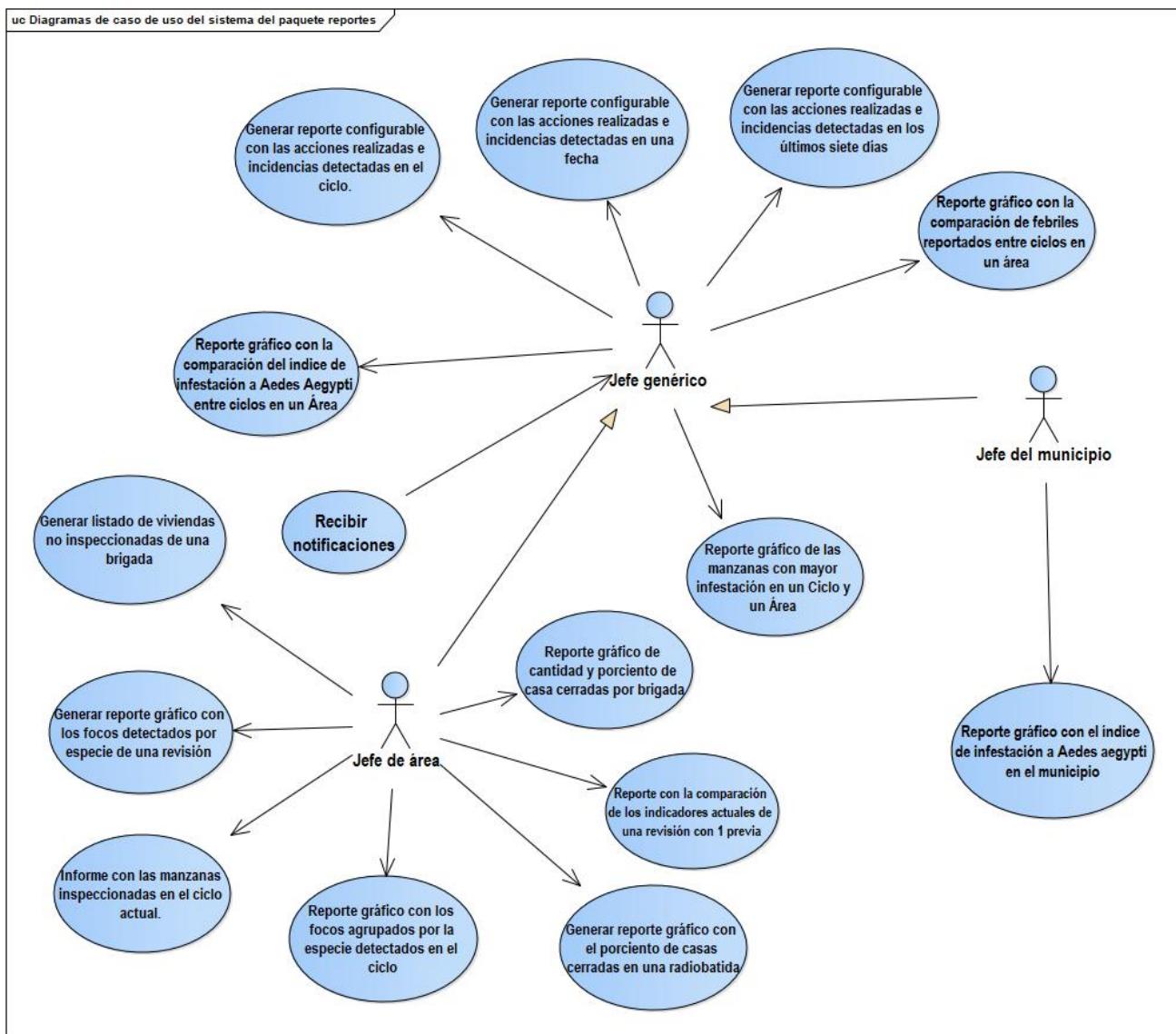


Imagen 3-6 Diagrama de casos de uso del paquete Reportes

3.7 Descripción de los principales casos de uso.

3.7.1 Paquete de Seguridad

Nombre del Caso de uso	Autenticar usuario
Actores	Estadístico, biólogo, administrador municipal del sistema, administrador provincial del sistema, jefe de área y jefe municipal
Resumen	El caso de uso inicia cuando el actor ingresa al sistema su nombre de usuario y contraseña. El sistema verifica la existencia y el caso de uso termina cuando el sistema le muestra la página de inicio correspondiente.

Precondiciones	El usuario debe de estar registrado en la aplicación.
Postcondiciones	El usuario tiene acceso a las funcionalidades del menú principal, generado en correspondencia con el rol asignado.

Nombre del Caso de uso	Gestionar usuario
Actores	Administrador del sistema y Jefe municipal
Resumen	El CU inicia cuando el usuario selecciona la opción “Usuarios” en el menú. El sistema mostrara la lista de usuarios. En este momento el usuario puede escoger las opciones de insertar, modificar o eliminar. El CU finaliza cuando sistema notifica al usuario que los datos se insertaron, modificaron o se desactivo el usuario correctamente.
Precondiciones	Se debe haber insertado un área para insertar un usuario, además el usuario tiene que pertenecer al área insertada. Se debe haber insertado al menos un usuario para modificar y desactivar. El usuario administrador no podrá ser desactivado del sistema, ni modificado su rol si no existe otro administrador del sistema.
Flujo de trabajo	
Acción del usuario	Respuesta del sistema
1- El caso de uso inicia cuando el usuario selecciona la opción “Usuario” en la barra menú	<p>2- El sistema muestra un listado con los usuarios y las diferentes opciones a realizar:</p> <ul style="list-style-type: none"> A- Para insertar un nuevo usuario ir a la sección Insertar usuario. B- Para modificar los datos de un usuario ir a la sección Modificar usuario. C- Para desactivar un usuario ir a la sección Desactivar usuario. D- Para activar un usuario ir a la sección Activar usuario.
Sección A: Insertar usuario	

3- El usuario selecciona la opción de Insertar usuario.	4- El sistema muestra la vista para insertar un usuario. 
5- El usuario inserta los datos necesarios en el formulario y selecciona la opción registrar	6- El sistema pasa a validar los datos. Si los datos están correctos se insertan en la base de datos y el sistema notifica al usuario que los datos se insertaron correctamente.
7- Si el usuario no es un administrador o un jefe de municipio queda registrado en la base de datos, pero no activo hasta que le sea asignada un área de salud. 9- El usuario jefe municipal selecciona la notificación. 11-El usuario jefe municipal selecciona la operación asignar. 13- Selecciona el área a asignar	8- El sistema muestra una notificación para el jefe municipal para que este le asigne un área de salud a los usuarios que no la tienen asignada. 10- El sistema muestra el listado de usuarios del municipio que no tienen área asignada. 12- El sistema muestra el formulario para asignar el área. 13- El sistema asigna el área al usuario y lo activa para que pueda acceder al mismo.
Sección B: Modificar usuario	
7- El usuario selecciona la opción de modificar un usuario en el botón. 	8- El sistema a partir del id del usuario muestra la vista para modificar el usuario con todos los datos asociados a este.
9- El usuario modifica los datos necesarios en la vista y selecciona la opción guardar	10- El sistema pasa a validar los datos. Si los datos están correctos se insertan en la base de datos y el CU finaliza cuando sistema notifica al usuario que los datos se modificaron correctamente.
Sección C: Desactivar usuario	
11- El usuario selecciona la opción desactivar un usuario específico en él botón. 	12- El sistema muestra una pantalla de confirmación y si desea desactivar el usuario seleccionado.
13- El usuario selecciona “Aceptar”.	14- El sistema notifica que el usuario seleccionado se desactivo correctamente.
Sección D: Activar usuario	

11- El usuario selecciona la opción activar un usuario específico en el botón. Activar	12- El sistema notifica que el usuario seleccionado se desactivó correctamente.
Postcondiciones	Quedan actualizados los datos de los usuarios en la BD

Nombre del Caso de uso	Cambiar contraseña
Actores	Estadístico, biólogo, administrador del sistema, jefe de área y jefe municipal
Resumen	El caso de uso inicia cuando el actor desea cambiar su contraseña. Este permite modificar la contraseña previamente definida del usuario autenticado. El caso de uso termina cuando el actor acepta la nueva contraseña.
Precondiciones	El usuario debe de estar registrado en la aplicación.
Postcondiciones	El usuario tiene acceso a las funcionalidades del menú principal, generado en correspondencia con el rol asignado.

3.7.2 Paquete de Gestión

Caso de uso	Gestionar planes de trabajo
Actores	Jefe de área
Resumen	El CU inicia cuando el usuario selecciona la opción planes de trabajo en la barra menú. El sistema mostrara la lista de planes de trabajo. En este momento el usuario puede escoger las opciones de insertar, modificar o eliminar. El CU finaliza cuando sistema notifica al usuario que los datos se insertaron, modificaron o eliminaron correctamente.
Precondiciones	El usuario debe estar autenticado. Se debe haber insertado al menos una manzana y una brigada. Insertar nuevo Plan: Se realizará un nuevo plan solo a las brigadas que no tengan registrado un plan en el día. Se planificarán radiobatidas y tratamientos químicos solo si existen focos o febres.
Flujo de trabajo	
Acción del usuario	Respuesta del sistema

1- El caso de uso inicia cuando el usuario selecciona la opción “Planes de trabajo” en la barra de menú.	2- El sistema muestra un listado con los planes de trabajo de las brigadas del área de salud y las diferentes opciones a realizar: A- Para insertar un nuevo plan de trabajo ir a la sección Insertar plan de trabajo. B- Para modificar los datos de un plan de trabajo ir a la sección Modificar plan de trabajo. C- Para eliminar un plan de trabajo ir a la sección Eliminar plan de trabajo.
--	--

Sección A: Insertar plan de trabajo

3- El usuario selecciona la opción de Insertar Plan.	4- El sistema muestra la vista para insertar un nuevo plan de trabajo, mostrando en el campo donde se selecciona la brigada solo aquellas a las que no se le ha realizado el plan de trabajo de ese día, el sistema mostrara los campos de planificación de radiobatidas y tratamientos químicos solo cuando existen incidencias en al menos una de las manzanas a cargo de la brigada seleccionada, a la vez mostrara los campos para planificar las inspecciones y las revisiones.
--	--

Insertar Plan

5- El usuario inserta los datos necesarios en el formulario y selecciona la opción registrar	6- El sistema pasa a validar los datos. Si los datos están correctos se insertan en la base de datos, si existe alguna incidencia en la realización del plan registra el tipo y las consecuencias y el CU finaliza cuando sistema notifica al usuario que los datos se insertaron correctamente.
--	--

Sección B: Modificar plan de trabajo

7- El usuario selecciona la opción de modificar plan en el botón.	8- El sistema a partir del id del plan muestra la vista para modificar el plan con los datos asociados a esta excluyendo la brigada.
---	--



9- El usuario modifica los datos necesarios en la vista y selecciona la opción guardar	10- El sistema pasa a validar los datos. Si los datos están correctos se insertan en la base de datos y el CU finaliza cuando sistema notifica al usuario que los datos se modificaron correctamente.
--	---

Sección C: Eliminar plan de trabajo

11- El usuario selecciona la opción	12- El sistema a partir del id del plan muestra una pantalla de confirmación y si acepta elimina el plan y notifica al usuario
-------------------------------------	--

de eliminar plan en el botón.	que los datos se eliminaron correctamente finalizando así el CU. 
-------------------------------	---

Caso de uso	Gestionar inspecciones
Actores	Estadístico
Resumen	El CU inicia cuando el usuario selecciona la opción inspección en el menú gestión de acciones. El sistema mostrara la lista de inspecciones. En este momento el usuario puede escoger las opciones de insertar, modificar o eliminar. El CU finaliza cuando sistema notifica al usuario que los datos se insertaron, modificaron o eliminaron correctamente.
Precondiciones	El usuario debe estar autenticado. Se debe haber insertado al menos una manzana, una brigada, un área, un municipio y una provincia para insertar una inspección, además el usuario tiene que pertenecer al área insertada. Se debe haber insertado al menos una inspección para modificar y eliminar.
Flujo de trabajo	
Acción del usuario	Respuesta del sistema
1- El caso de uso inicia cuando el usuario selecciona la opción inspección en el menú gestión de acciones en la barra de menú.	2- El sistema muestra un listado con las inspecciones y las diferentes opciones a realizar: A- Para insertar una nueva inspección ir a la sección Insertar inspección. B- Para modificar los datos de una inspección ir a la sección Modificar inspección. C- Para eliminar una inspección ir a la sección Eliminar inspección.
Sección A: Insertar inspección	
3- El usuario selecciona la opción de Insertar Inspección. 	4- El sistema muestra la vista para insertar una inspección.
5- El usuario inserta los datos necesarios en el	6- El sistema pasa a validar los datos. Si los datos están correctos se insertan en la base de datos y el CU finaliza

formulario y selecciona la opción registrar	cuento sistema notifica al usuario que los datos se insertaron correctamente.
Sección B: Modificar inspección	
7- El usuario selecciona la opción de modificar una inspección en el botón. 	8- El sistema a partir del id de la inspección muestra la vista para modificar la inspección con todos los datos asociados a esta.
9- El usuario modifica los datos necesarios en la vista y selecciona la opción guardar	10- El sistema pasa a validar los datos. Si los datos están correctos se insertan en la base de datos y el CU finaliza cuando sistema notifica al usuario que los datos se modificaron correctamente.

Caso de uso	Gestionar tratamientos adulticidas
Actores	Estadístico
Resumen	El CU inicia cuando el usuario selecciona la opción tratamientos adulticidas en el menú gestión de acciones. El sistema mostrara la lista de tratamientos adulticidas. En este momento el usuario puede escoger las opciones de insertar, modificar o eliminar. El CU finaliza cuando sistema notifica al usuario que los datos se insertaron, modificaron o eliminaron correctamente.
Precondiciones	El usuario debe estar autenticado. Se debe haber insertado al menos una manzana donde se halla detectado un foco o un febril, una brigada, un área, un municipio y una provincia para insertar un tratamiento adulticida, además el usuario tiene que pertenecer al área insertada. Se debe haber insertado al menos un tratamiento adulticida para modificar y eliminar.

Caso de uso	Gestionar viviendas no inspeccionadas
Actores	Estadístico
Resumen	El CU inicia cuando el usuario registra una inspección o una radiobatida y estas últimas presentan viviendas cerradas, viviendas negadas o viviendas recuperadas. Luego el usuario registra los datos de las viviendas. El sistema mostrara el formulario para el registro de estas viviendas. El usuario registra la información. El CU finaliza cuando sistema notifica al usuario que los datos se insertaron correctamente.
Precondiciones	El usuario debe estar autenticado. Se debe haber insertado al menos una inspección o una radiobatida con viviendas cerradas, viviendas negadas o viviendas recuperadas.

Caso de uso	Gestionar focos
Actores	Biólogo
Resumen	El CU inicia cuando el usuario selecciona la opción focos en el menú Gestión Entomológica. El sistema mostrara la lista de focos. En este momento el usuario puede escoger las opciones de insertar, modificar o eliminar. El CU finaliza cuando sistema notifica al usuario que los datos se insertaron, modificaron o eliminaron correctamente.
Precondiciones	El usuario debe estar autenticado. Se debe haber insertado al menos una manzana, una revisión, una inspección, una brigada, un área, un municipio, una provincia y los nomencladores asociados al foco, además el usuario tiene que pertenecer al área insertada. Se debe haber insertado al menos un foco para modificar y eliminar.

Caso de uso	Gestionar febriles
Actores	Jefe de área
Resumen	El CU inicia cuando el usuario selecciona la opción Febriles en el menú Gestión de Acciones. El sistema mostrara la lista de febriles. En este momento el usuario puede escoger las opciones de insertar, modificar o eliminar. El CU finaliza

	cuando sistema notifica al usuario que los datos se insertaron, modificaron o eliminaron correctamente.
Precondiciones	El usuario debe estar autenticado. Se debe haber insertado al menos una manzana, una revisión, una inspección, una brigada, un área, un municipio, una provincia y los nomencladores asociados al foco, además el usuario tiene que pertenecer al área insertada. Se debe haber insertado al menos un foco para modificar y eliminar.

3.7.3 Paquete Reportes

Caso de uso	Generar reporte configurable con las acciones realizadas e incidencias detectadas en una fecha seleccionada
Actores	Jefe de área
Resumen	El CU inicia cuando el usuario selecciona la opción Reporte general por día en la barra menú. El sistema mostrara el formulario donde el usuario seleccionara la fecha, el área de salud y los informes que se mostraran en el reporte. En este momento el usuario escoge la opción generar y el sistema muestra los informes seleccionados.
Precondiciones	El usuario debe estar autenticado.
Flujo de trabajo	
Acción del usuario	Respuesta del sistema
1- El caso de uso inicia cuando el usuario selecciona la opción “Reporte general por día” en el menú “Reportes para el jefe del municipio”.	2- El sistema muestra un formulario con la fecha, el área y los informes a seleccionar.
3- El usuario selecciona la fecha, el área y los informes	4- El sistema genera el reporte con los informes seleccionados en el área y la fecha seleccionada.

que desea generar en el reporte.	
----------------------------------	--

Caso de uso	Generar reporte gráfico de la comparación entre Ciclos con respecto al índice de infestación.
Actores	Jefe de área
Resumen	El CU inicia cuando el usuario selecciona la opción Reporte general por día en el menú “Reportes para el jefe del municipio”. El sistema mostrara el formulario donde el usuario seleccionara el área de salud y el ciclo. En este momento el usuario escoge la opción generar y el sistema muestra el grafico con los índices de infestación del ciclo seleccionado y otro previo.
Precondiciones	El usuario debe estar autenticado.
Flujo de trabajo	
Acción del usuario	Respuesta del sistema
1- El caso de uso inicia cuando el usuario selecciona la opción “Reportes para el jefe del municipio” en el menú “Reportes para el jefe del municipio”.	2- El sistema muestra un formulario con las áreas y los ciclo a seleccionar.
3- El usuario selecciona la el área y el ciclo que desea comparar con uno previo.	4- El sistema genera el grafico con los índices de infestación del ciclo seleccionado y otro previo.

3.8 Definición de los requisitos no funcionales

✓ **Apariencia o interfaz:**

El sistema debe contener una apariencia minimalista, de fácil navegación y rápido entendimiento por parte de los usuarios. El diseño del sistema se basa en las pautas de diseño web establecidas

por entidad, predominando los colores gris y azul, los cuales aportan frescor y legibilidad a la información mostrada al usuario.

✓ **Usabilidad.**

La aplicación debe ser fácil de usar tanto por los usuarios que accedan a él por primera vez como para los habituales, garantizando así que se pueda acceder a la información de manera simple y cómoda.

✓ **Rendimiento:**

El tiempo de respuesta del sistema estará relacionado a la velocidad de las conexiones de red y del servidor donde se aloje el mismo, dado que es una aplicación web. No se requiere el uso de características adicionales de rendimiento, aunque se considera que la navegación sea más eficiente después del primer inicio de sesión dado al uso de cachés en el navegador.

✓ **Soporte:**

El sistema debe brindar facilidad de instalación, facilidad de mantenimiento y facilidad actualización hacia versiones posteriores ya sea del sistema en si como de los paquetes de terceros.

✓ **Seguridad:**

La seguridad del sistema estará dada por los niveles de acceso y por los permisos que tengan los ficheros que conforman el sistema, dentro del servidor. Las contraseñas sólo podrán ser cambiadas por el propio usuario y bajo su responsabilidad, estas deben de ser cifradas con mecanismos criptográficos actualizados y que no estén vulnerados. Se registrarán las trazas para su posterior control.

✓ **Confiabilidad:**

La información que se maneja en la aplicación será protegida de accesos no autorizados y debe poder ser consultada en cualquier momento.

✓ **Software.**

Las necesidades de software para el funcionamiento correcto de la aplicación estarán sustentadas por los siguientes requerimientos:

- **Requerimientos del lado del Cliente:**

- Sistema Operativo: Microsoft Windows 7 o superior, Linux
- Navegador web compatible con HTML5, CSS, JavaScript.

- **Requerimientos del lado servidor:**

- Sistema Operativo: Windows Server 2008 R2 o superior.

- Ubuntu Server 18.04 LTS
- Gestor de Bases de datos MySQL
- Servidor Web Apache v2.4.27

✓ **Hardware.**

- **Requerimientos del lado del Cliente:**
 - Procesador: 2Ghz
 - Memoria: 2GB
 - Espacio en Disco Duro: 20 GB
- **Requerimientos mínimos del lado del Servidor:**
 - Procesador: 1.4 GHz
 - Memoria: 2gb
 - Espacio en Disco Duro: 30GB (Servidor WEB)
 - Espacio en Disco Duro: 1TB (Servidor de Datos)

3.9 Conclusiones

En este capítulo al aplicar el flujo de trabajo de Gestión de Requisitos ha sido posible describir las funcionalidades y los actores del sistema que formarán parte del sistema a desarrollar con el propósito de cumplir con las necesidades concretas del usuario final de la aplicación. Todo se representa a través de diagramas de casos de uso y otras herramientas UML.

- Se enmarca de manera ordenada los requisitos funcionales y no funcionales que estructuran la aplicación facilitando el desarrollo del software y la búsqueda de una solución que sea eficiente en la solución de la situación problemática que se plantea en este trabajo. Los casos de uso del sistema han permitido obtener un grupo de soluciones a los requisitos que se plantean.
- Los diversos actores propuestos permiten crear roles y asignar una serie de permisos a los usuarios para acceder al sistema de una manera más especializada.
- Se define el conjunto de paquetes derivados del análisis para organizar de esta manera el desarrollo del sistema, estos paquetes agrupan un conjunto de funcionalidades.
- El conjunto de requisitos no funcionales permite aportar características que hacen más robusto el software desarrollado y lo convierte en una herramienta muy versátil y adaptable.

Capítulo 4 Descripción de la solución propuesta

4.1 Introducción.

Este capítulo tiene como objetivo describir el diseño de la solución propuesta que da respuesta a los requisitos puntuados en el capítulo anterior. Por tanto, como parte de la documentación de este diseño se definen los diagramas de clases, el modelo lógico y físico de los datos, los principios de diseño que se tuvieron en cuenta para realizar las interfaces de usuarios, el formato de salida de reportes utilizado, el tratamiento dado a los posibles errores y el diagrama de despliegue para representar la distribución de la aplicación.

4.2 Diagrama de clases del diseño.

4.2.1 Paquete Seguridad

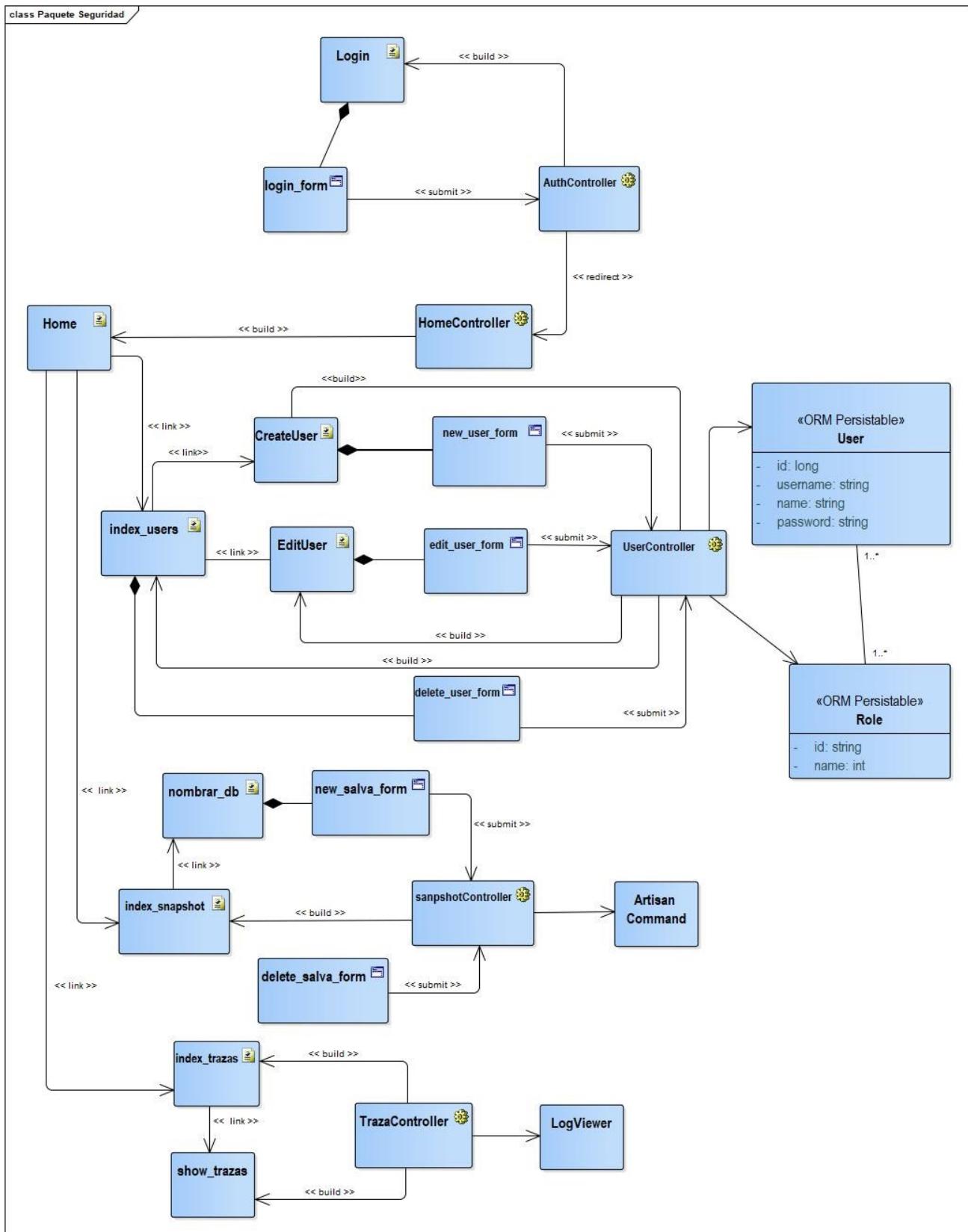


Imagen 4-1 Diagrama de clases del diseño del Paquete Seguridad

4.2.2 Paquete Gestión

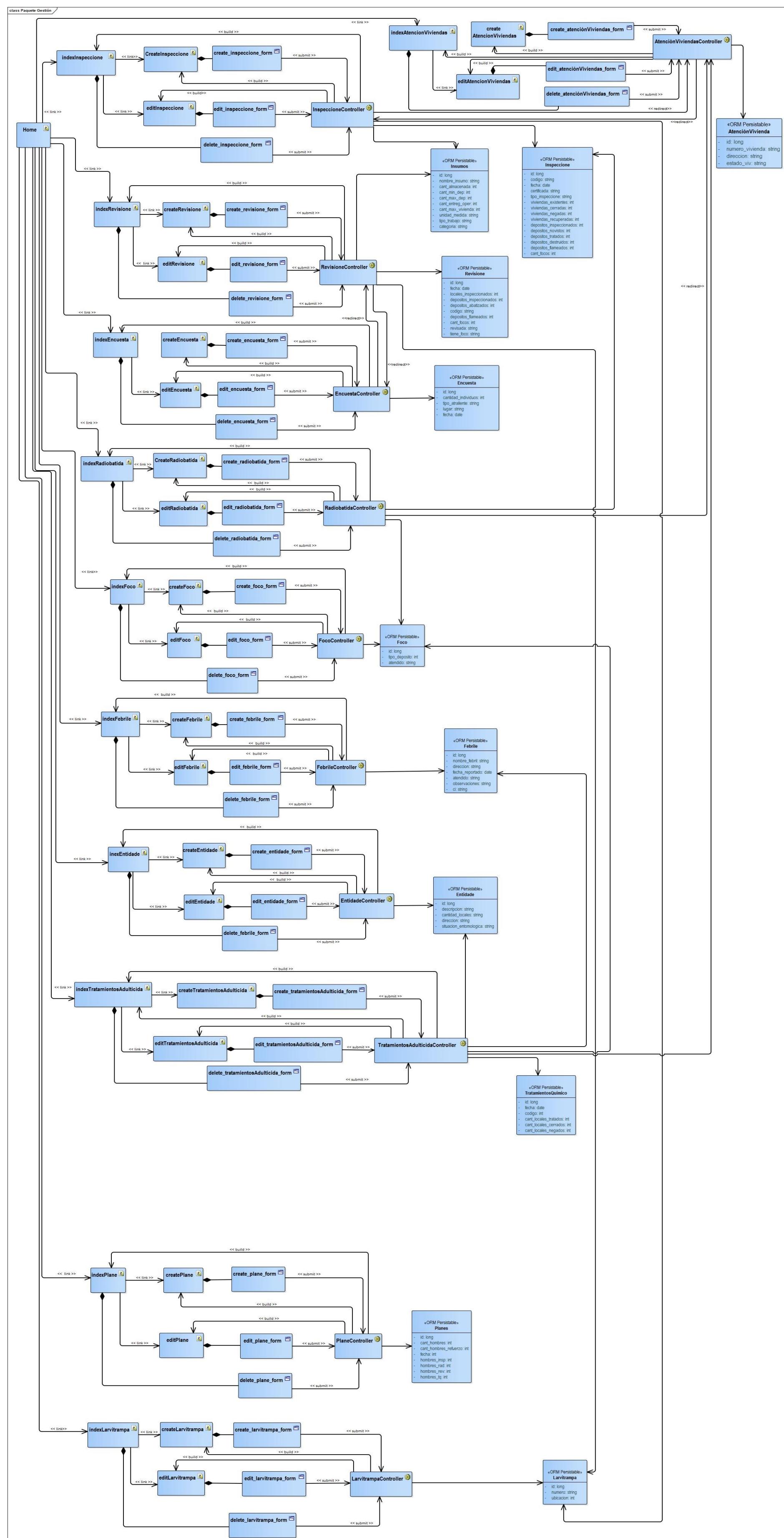


Imagen 4-2 Diagrama de clases del diseño del Paquete Gestión

4.2.3 Paquete Nomencladores

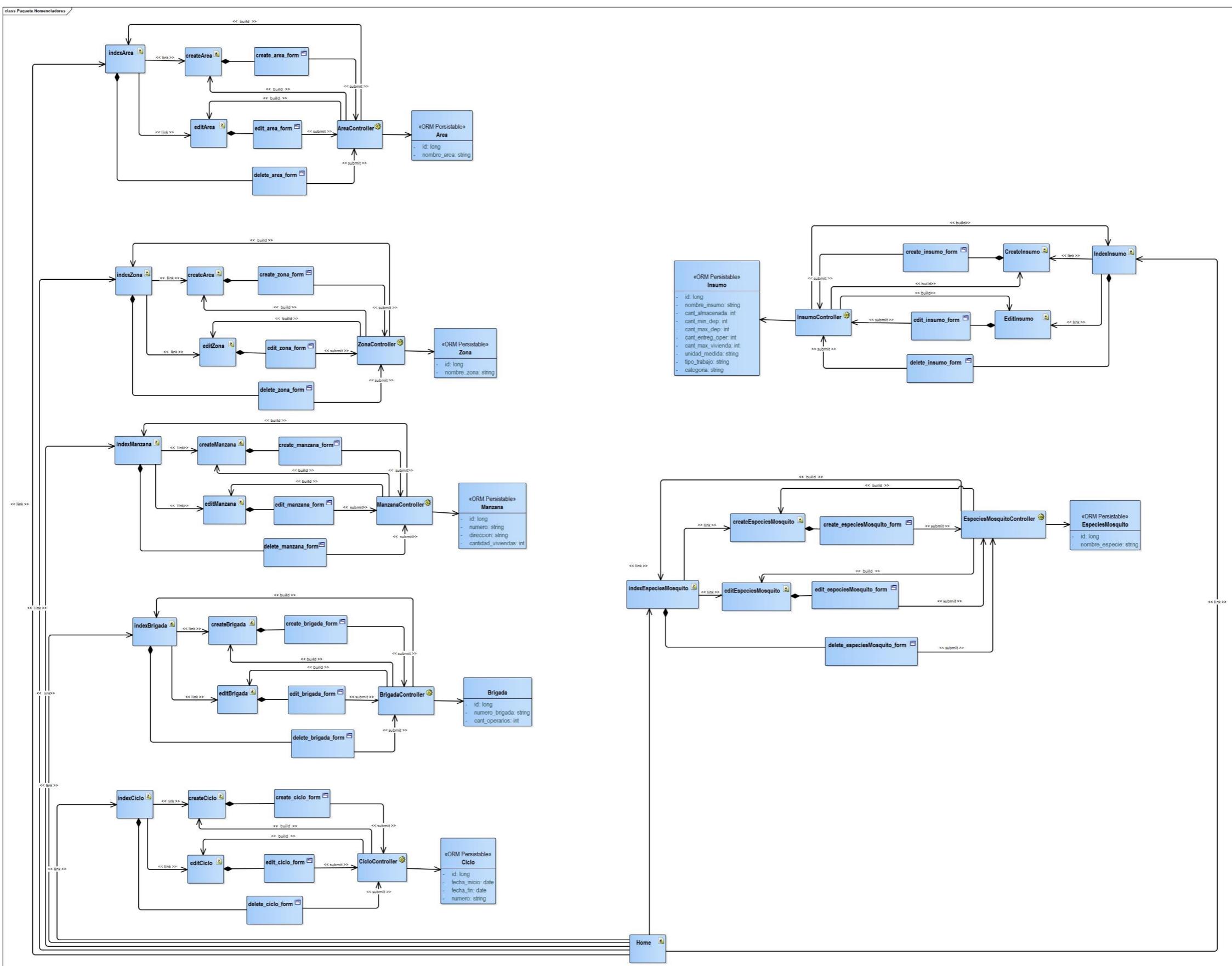


Imagen 4-3 Diagrama de clases del diseño del Paquete Nomencladores

4.2.4 Paquete Reportes

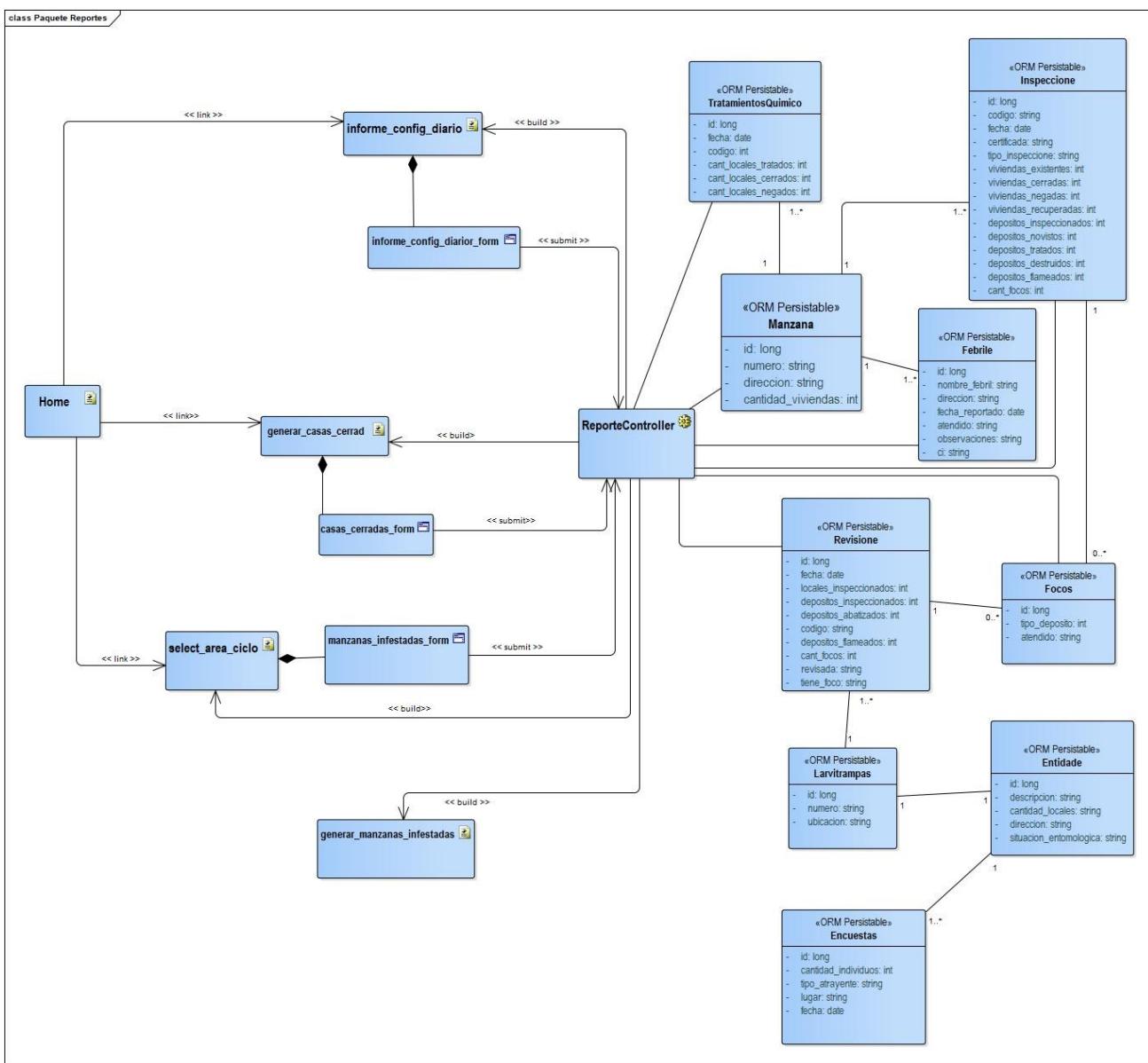


Imagen 4-4 Paquete Reportes

4.3 Patrón arquitectónico

Modelo–vista–controlador es un patrón de arquitectura de software, que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado, define componentes para la representación de la información, y por otro lado para la interacción del usuario.

Modelos

Los modelos son clases encargadas de trabajar con las consultas de la base de datos, es decir que por cada tabla tendremos una clase, cada registro será un objeto y las consultas se llamarán a través de métodos de esas clases. A su vez Laravel trabaja con Eloquent que es un ORM que nos facilitará el trabajo de las consultas a través de métodos ya establecidos, estos nos permitirán realizar las tareas más comunes y que más se repiten en una base de datos.

Vistas

Es el producto final de una petición, el código HTML que se le devuelve al cliente, aquí no debería haber ninguna lógica, sin embargo, puede contener impresiones de variables, condicionales o bucles.

Controladores

Los controladores son clases con métodos, también llamados acciones, estas acciones se comunicarán con los modelos para hacer consultas a la base de datos, y con las vistas para devolver una respuesta al cliente.

Si bien Laravel no utiliza MVC exactamente, se basa en este patrón. En el framework Laravel no es suficiente tres capas para representar su estructura, por ejemplo:

- No tenemos una manera de representar las rutas que se encargan de analizar las URLs y asignarlas a un método o función,
- No tenemos forma de representar un middleware que restringe ciertas áreas de la aplicación basado en el estado del usuario
- Una aplicación moderna requiere mucho más que la representación de datos a través de un modelo

Según el creador de Laravel Taylor Otwell: “No hay manera de encapsular todos los aspectos de aplicaciones web robustas en esas 3 letras.” [27]

4.4 Principios de diseño.

El desarrollo con frameworks como Laravel prioriza el uso de buenas prácticas gracias a un grupo de principios de diseños por los cuales se rigen. Estos principios son conocidos por las siglas SOLID.

S = Single Responsibility Principle

O = Open/Closed Principle

L = Liskov Substitution Principle

I = Interface Segregation Principle

D = Dependency Inversion Principle

Single Responsibility Principle

El Principio de Responsabilidad Única nos guía a separar los comportamientos basándonos en ejes del cambio. Podríamos decir que, y aquí está la clave de todo, cada responsabilidad es un eje del cambio sí y solo si el cambio ocurre. Esto significa que, si tuviéramos una clase con tres

responsabilidades invariantes en su comportamiento, no estaríamos incurriendo en una violación del SRP. Si alguna de esas tres responsabilidades definidas por el comportamiento cambiara en distintos momentos en el tiempo, es decir, se volvieran variantes, entonces sí que deberíamos separar dichas responsabilidades en otras clases ya que tendríamos el síntoma de tener una clase con varios ejes del cambio. [28]

Open-Close Principle El diseño debe quedar abierto para extensiones y cerrado para modificaciones. Sobre todo, aplicable al diseño de piezas reutilizables, que implementan requerimientos generales a todo el sistema. [28]

Interface Segregation Principle

El principio establece que las interfaces deberían ser tan pequeñas como sea posible, de modo que las necesidades queden separadas. Brindando únicamente lo que se necesita.

Dependency Inversion Principle

Bajo acoplamiento y alta cohesión propone trabajar en interfaces o en clases abstractas más que en clases concretas ya que estas serán más flexibles.

Encapsulate What Varies

Solo las cambiar las cosas que varían en la aplicación. Solo debe de centrarse en lo que va a cambiar y encapsular esas partes variables de modo que las que no varían puedan estabilizarse.

4.5 Patrones de diseño

Los patrones de diseño son soluciones efectivas a problemas comunes. Laravel promueve el uso de buenas prácticas y se evidencia en el uso de patrones de diseño, entre los que se encuentran:

Patrón Factory

Define una interfaz para la creación de un objeto, pero permite que las subclases decidan cuál de las clases instanciar. El Método Factory permite que una clase difiera la instantiación a las subclases. La siguiente ilustración muestra lo antes expuesto:

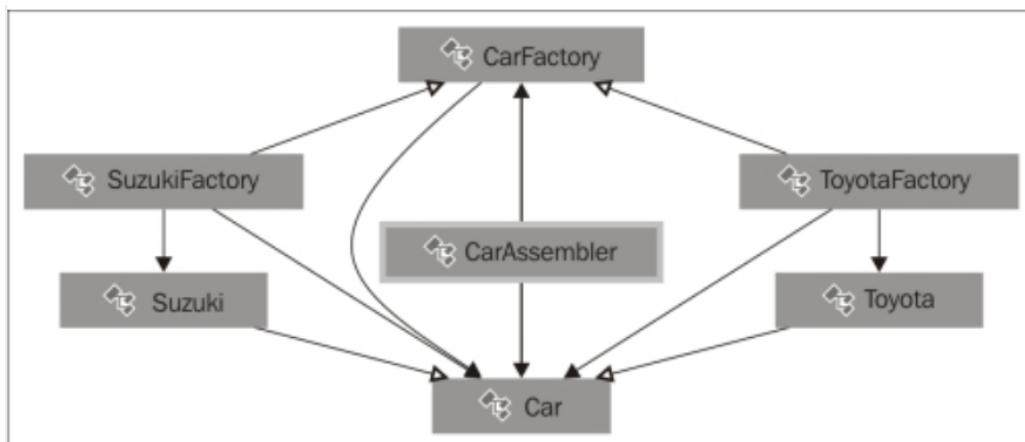


Imagen 4-5 Patrón Factory

En Laravel se evidencia cuando se envían varios tipos de reglas de validación con la clase Validation. Cuando desarrollamos aplicaciones, generalmente necesitamos validar los datos a medida que avanzamos. A veces, necesitamos establecer reglas personalizadas y mensajes de

error personalizados para validar los datos. Para ver como se implementa esta solución se muestra el siguiente segmento de código en [vendor/Illuminate/Validation/Factory.php](#):

```
<?php  
...  
use Illuminate\Container\Container;  
use Symfony\Component\Translation\TranslatorInterface;  
  
class Factory {  
    protected $translator;  
    protected $container;  
    ...  
    protected function addExtensions(Validator $validator)  
    {  
        $validator->addExtensions($this->extensions);  
        $implicit = $this->implicitExtensions;  
        $validator->addImplicitExtensions($implicit);  
        $validator->addReplacers($this->replacers);  
        $validator->setFallbackMessages($this->fallbackMessages);  
    }  
    ...  
}
```

Como podemos ver en el código anterior, la clase Factory de Validation está construida con la clase Translator y un contenedor IoC. La función addExtensions () se establece después de esto. Este método incluye las extensiones definidas por el usuario para una instancia de Validator, lo que nos permite escribir la estructura para crear las extensiones de la clase Validator. Las funciones, que son públicas, nos permiten implementar la clase Translator, y con esto queremos decir que nos permiten escribir reglas y mensajes de validación personalizados.

Patrón Facade

Este permite a un desarrollador unir varias interfaces complicadas en una sola interfaz de clase. Este patrón también le permite ajustar varios métodos de varias clases en una sola estructura. La siguiente ilustración muestra los anteriores expuestos:

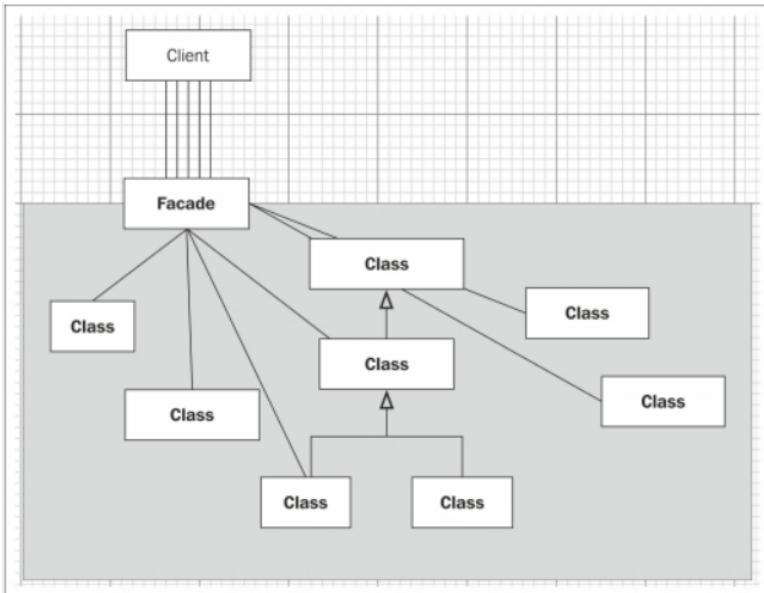


Imagen 4-6 Patrón Facade

Desde Laravel 4 casi todos los métodos parecen ser estáticos, por ejemplo, `Input::has()`, `Config::get()`, `URL::route()`, etc; pero en realidad no lo son. En segundo plano, con la ayuda de IoC Container (una forma de *inyectar dependencias* en una clase), Laravel realmente llama a otra clase a través de una clase Fachada. Para ver un ejemplo del uso de este patrón supongamos que tenemos un método llamado `URL::to('home')`, que es accedido por el alias URL definido en `config/app.php` el cual nos dirige a la clase URL implementada en `/vendor/Illuminate/Support/Facades/URL.php`:

```
class URL extends Facade {
    protected static function getFacadeAccessor() { return 'url'; }
```

Como puede ver, la clase en realidad hereda de la clase Fachada, y no hay un método estático llamado `to()`. En cambio, hay un método llamado `getFacadeAccessor()`, que devuelve la url de cadena. El propósito del método `getFacadeAccessor()` es definir qué inyectar. De esta manera, Laravel entiende que esta clase está buscando `$app['url']`. Esto llama al método `registerUrlGenerator()` en `vendor/Illuminate/Routing/ RouteServiceProvider.php` el cual retorna una instancia de `UrlGenerator` que tiene el método `to()` que estábamos buscando como se aprecia en `vendor/Illuminate/Routing/UrlGenerator.php`:

```
public function to($path, $extra = array(), $secure = null) { ... }
```

Entonces, cada vez que usa un método como este, Laravel primero va y revisa la fachada, luego verifica qué se inyecta y luego se llama al método real a través de la clase inyectada.

Patrón Decorator

Este patrón crea una plantilla que se convierte en padre de todas las vistas que contienen un decorador para permitir agregar funcionalidades dinámicamente. El archivo nombrado `layout.blade.php`, es el que contiene el diseño de la página. Este archivo es conocido como plantilla base, guarda el código HTML que es común para todas las páginas de un módulo del sistema o

puede utilizarse global en el sistema dependiendo del tipo de proyecto, para no tener que repetirlo en cada página el código HTML. Este procedimiento es una implementación del patrón Decorator. [28]

Patrón Active Record

Este patrón de persistencia es quizás uno de los más habituales y es usado por frameworks como Laravel. La idea detrás del patrón es bastante sencilla. Se trata de una clase que se encarga de implementar todas las operaciones de consulta y modificación de una tabla concreta de la base de datos.

De esta forma nuestra aplicación quedará completamente aislada del trabajo con SQL ya que delegará en la capa de componentes ActiveRecord para realizar todas las operaciones. [28]

Patrón Repository

Este patrón nos enseña que debemos crear una capa adicional para extraer o almacenar datos en una fuente externa. En este caso no es lo mismo, es un modelo o entidad que representa un registro dentro del sistema y un repositorio que implementa la lógica para acceder a dicho registro o para editararlo mediante ese modelo.

Al crear una capa de repositorios adicional a la capa del modelo, desvinculamos el controlador de la implementación específica de Eloquent y hacemos posible que más adelante podamos implementar otras formas diferentes de acceder a los datos. Además, evitamos la duplicación de código y nuevamente, damos un paso más para cumplir con los principios SOLID y facilitar la creación de pruebas. [29]

Patrón Observer

El patrón observer trabaja dentro de los estados de una aplicación. Cuando ocurre un cambio en alguna entidad esta notifica al observer y posibilita la ejecución de eventos.

4.6 Estructuración en capas.

4.6.1 Enfoque por reutilización.

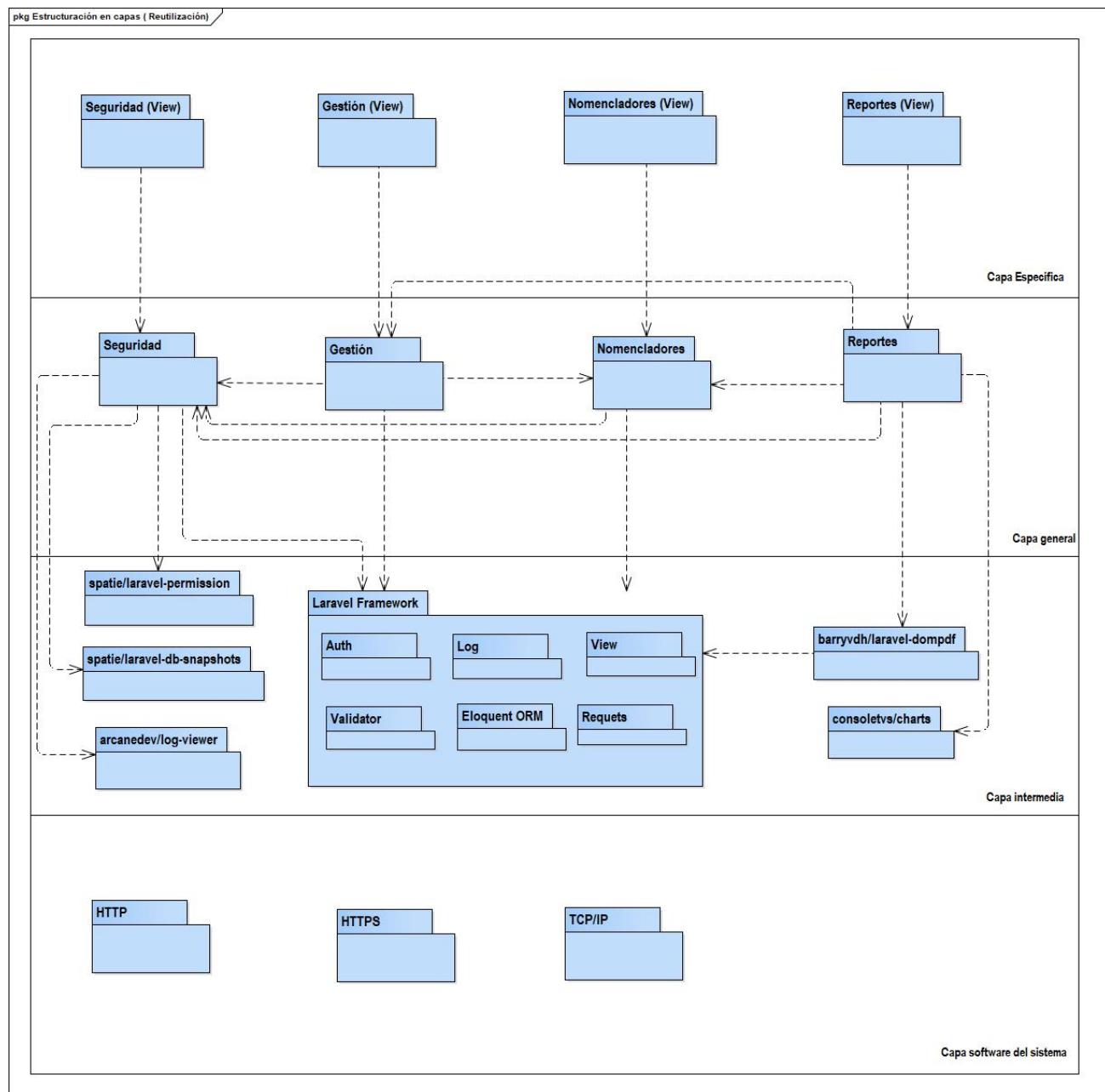


Imagen 4-7 Enfoque por reutilización

4.6.2 Enfoque por responsabilidad.

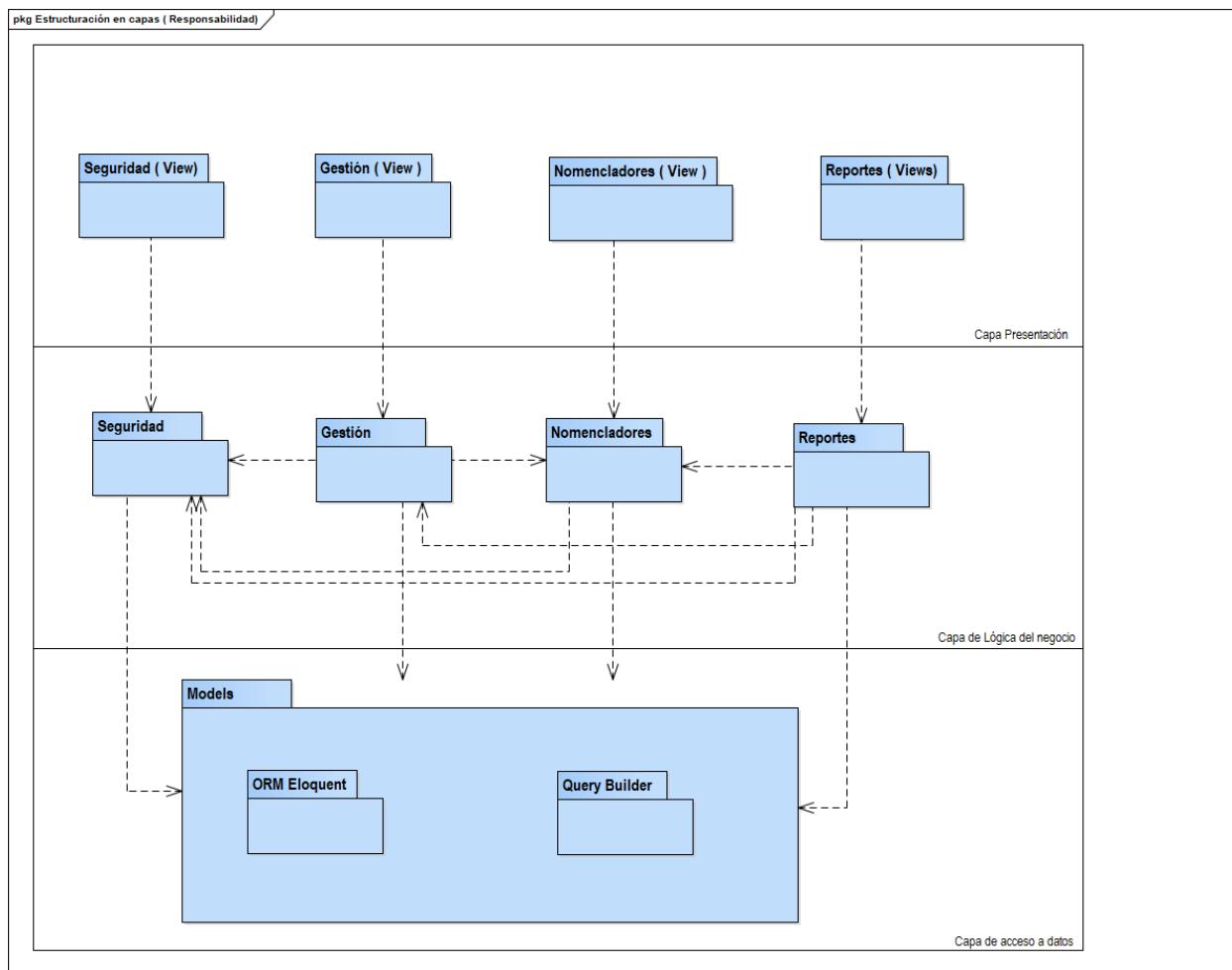


Imagen 4-8 Enfoque por responsabilidad

4.7 Mecanismos de acceso a datos.

Query builder

El constructor de consultas o query builder de Laravel es una interfaz conveniente y fluida para ejecutar consultas de base de datos. Este puede usarse para realizar la mayoría de las operaciones de bases de datos de la aplicación y funciona para todos los sistemas de bases de datos soportados.

El generador de consultas de Laravel utiliza el enlace de parámetros de PDO para proteger la aplicación contra los ataques de inyección de SQL. No es necesario limpiar las cadenas que se pasan como enlaces. [28]

Eloquent ORM

Laravel incluye un sistema de mapeo de datos relacional llamado Eloquent ORM que facilita la creación de modelos. Este ORM se funda en patrón active record y su funcionamiento es muy sencillo. Cada tabla de la base de datos se corresponde con un modelo el cual es usado para interactuar con la misma. [14]

4.8 Diseño de la base de datos.

4.8.1 Modelo lógico de datos. [30]

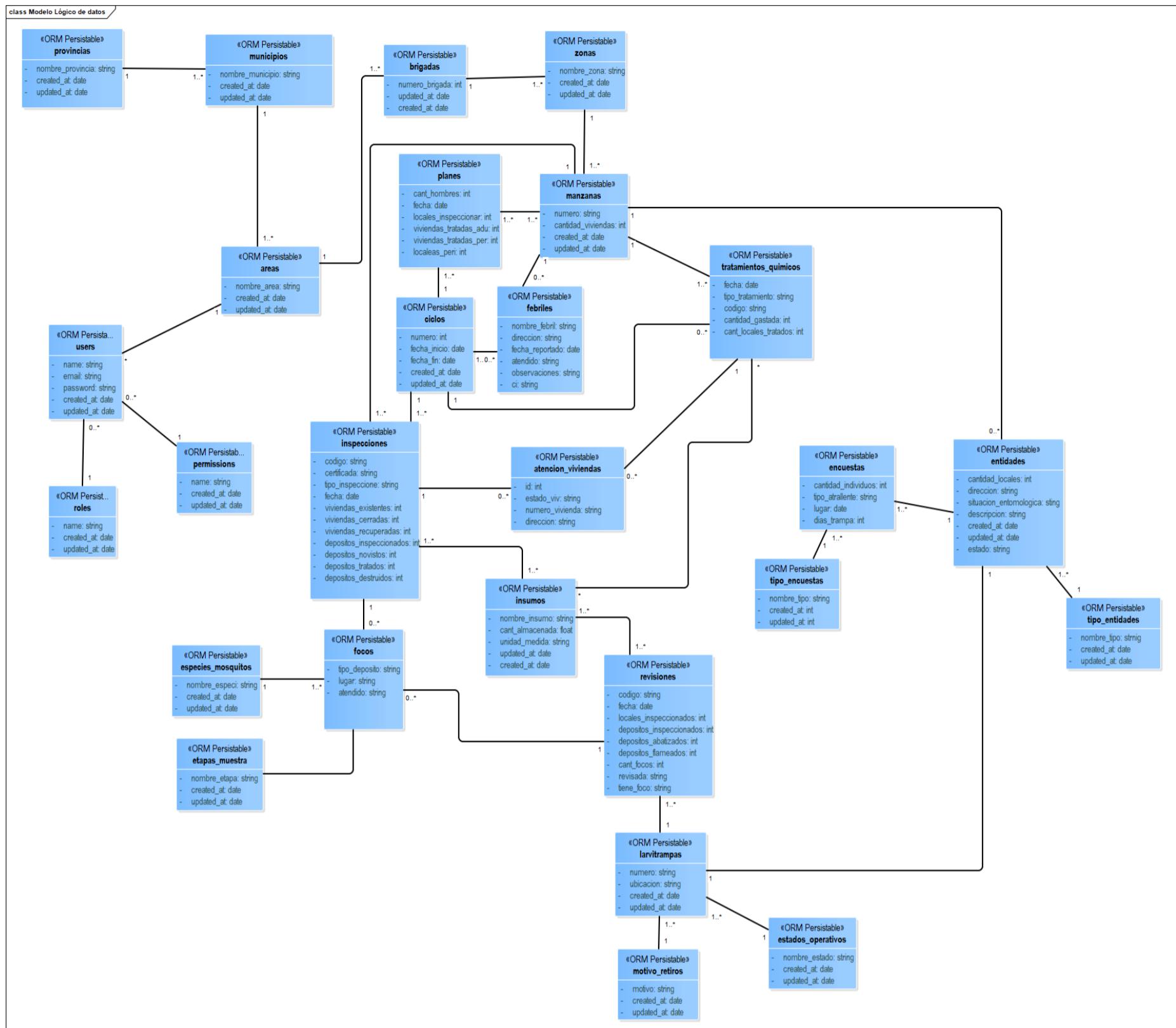


Imagen 4-9 Modelo lógico de datos

4.8.2 Modelo físico de datos [30]



Imagen 4-10 Modelo físico de datos

4.9 Interfaz de usuario

El diseño de la interfaz debe ajustarse a aspectos que garanticen una simple utilización y visualización, al igual que una fácil navegación por el sistema, que permita estar al alcance de usuarios con poca experiencia.

Para el desarrollo de la aplicación se emplearon colores como el azul en varios tonos, blanco y gris, además de otros colores de apoyo para determinados botones y señales. El menú de navegación se encuentra disponible en todo momento y muestra la información según los privilegios asignados. Los mensajes de información o error que se muestran en el sistema son explícitos y se muestran con un mismo formato para que el usuario se sienta rápidamente familiarizado.

The screenshot shows the SIGCAMP application interface. On the left is a vertical menu bar with the title 'SIGCAMP' and 'Anti-Vectorial'. The menu includes 'Página Principal', 'Gestión de Acciones' (selected), 'Entidades', 'Encuestas', 'Tratamientos Químicos', 'Revisiones de las entidades', 'Inspecciones' (selected), 'Lista de inspecciones diarias', 'Lista de radiobatidas', 'Larvitrampas', and 'Viviendas no inspeccionadas'. The main content area is titled 'Inspecciones Diarias' and displays a table of daily inspection data. The table has columns for Fecha (Date), Brigada (Brigade), Manzana (Plot), Viviendas Existentes (Existing Houses), Viviendas Inspeccionadas (Inspected Houses), Viviendas Cerradas (Closed Houses), Viviendas Negadas (Rejected Houses), Viviendas Recuperadas (Recovered Houses), Depósitos Inspeccionados (Inspected Deposits), Depósitos no vistos (Unseen Deposits), Depósitos Abatizados (Abatized Deposits), Focos detectados (Detected Focuses), and Operación (Operation). Below the table, it says 'Mostrando 4 registro(s)' (Showing 4 records) and has navigation buttons for 'Anterior' (Previous), '1', '2', and 'Siguiente' (Next). At the bottom right is the text 'SGCAMP 2019'.

Imagen 4-11 Interfaz de usuario

The screenshot shows the 'Nueva Inspección' (New Inspection) form. The left sidebar has 'Iniciar' and 'Gestión de Acciones'. The main form has a header 'Nueva Inspección' and a note 'Los campos marcados con asterisco (*) son obligatorios'. It contains several input fields: 'Fecha:' with value '26/12/2018', 'Brigada:' with dropdown placeholder 'Seleccione la brigada', 'Manzana:' with dropdown placeholder 'Seleccione la manzana', 'Viviendas existentes:' with placeholder 'Escriba la cantidad', 'Viviendas cerradas:' with placeholder 'Escriba la cantidad', 'Viviendas recuperadas:' with placeholder 'Escriba la cantidad', 'Depósitos inspeccionados:' with placeholder 'Escriba la cantidad', 'Depósitos tratados:' with placeholder 'Escriba la cantidad', 'Depósitos no vistos:' with placeholder 'Escriba la cantidad', 'Depósitos Destruídos:' with placeholder 'Escriba la cantidad', and 'Depósitos Flameados:' with placeholder 'Escriba la cantidad'.

Imagen 4-12 Interfaz de Nueva inspección

4.9.1 Formato de salida de los reportes.

Los reportes que ofrece la aplicación constituyen uno de los elementos más relevantes para el usuario. La mayoría constituyen gráficos de diferentes tipos ofreciendo la oportunidad de realizar comparaciones de la información que se muestra. También se presentan en formato tabular, siempre respetando la organización y estructura de todos los documentos llevados en copia dura antes de la implementación del sistema. También se ofrece la posibilidad de exportación a ficheros PDF con el objetivo de que el usuario pueda almacenar la información para su consulta posterior.

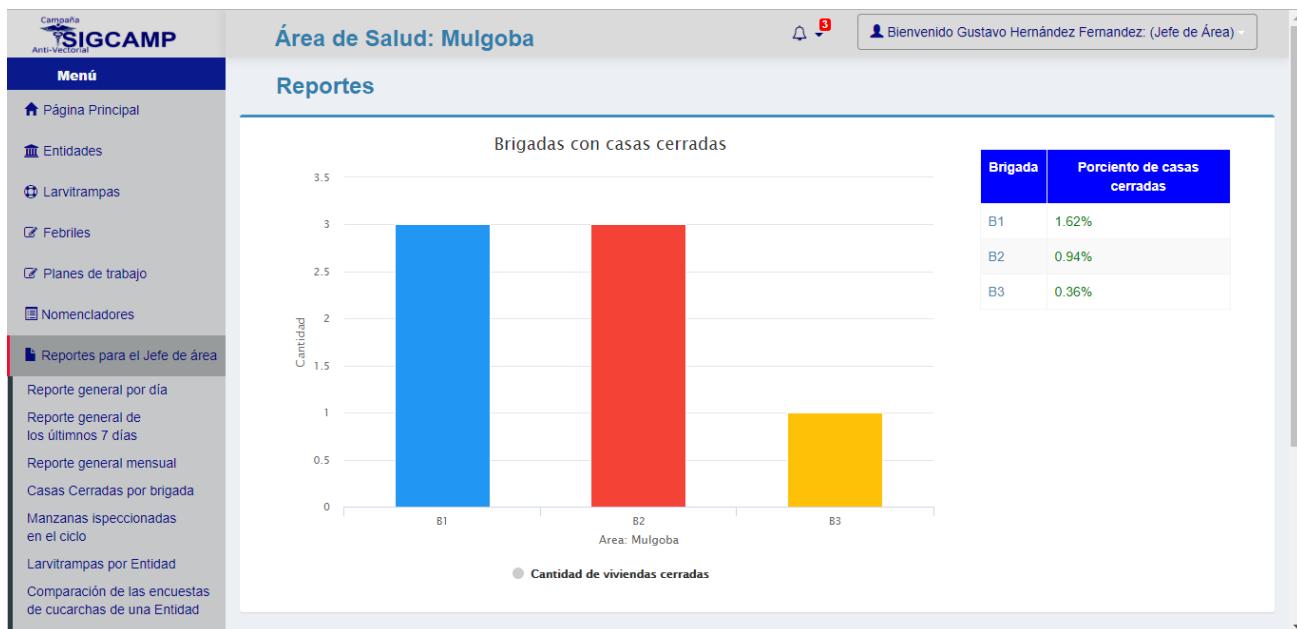


Imagen 4-13 Gráfico

4.9.2 Tratamiento de errores.

Minimizar la ocurrencia de errores dentro del sistema y garantizar que la información gestionada sea al menos coherente en cada momento, es de vital importancia en el logro de una aplicación confiable. Para esto, cada formulario de entrada de datos, tiene validado sus campos y comprueba que aquellos de carácter obligatorio no queden en blanco antes de iniciar cualquier acción, impidiendo así la entrada de información incorrecta, y en caso de producirse algún error, el sistema notifica al usuario y no realiza ninguna acción; a nivel de los modelos se definen dichas validaciones para garantizar que los campos tomen los valores acordes a su tipo y función.

Los formularios hacen usos de criterios de validación ofrecidos por el plugin jQuery Validation [31], que permite la validación de cada campo en función del tipo de dato que pueda contener. Se utilizó el (*) para representar los campos que son obligatorios llenar.

En el lado del servidor el framework Laravel incluye métodos de validación a través de la clase `Illuminate\Support\Facades\Validator` [14], ambos muestran mensajes de error personalizados.

Imagen 4-14 Errores en entrada de datos

Todas las excepciones son manejadas por la clase `App\Exceptions\Handler`, la cual posee dos métodos que son `report` y `render`. El método `report` se utiliza para registrar las excepciones o enviarlas a un servicio externo como Bugsnag. Por otro lado, en método `render` es el responsable de convertir una excepción dada en una respuesta HTTP que debe ser enviada al usuario. Laravel además de esto permite la personalización de las páginas de error como por ejemplo la página de error HTTP 404:

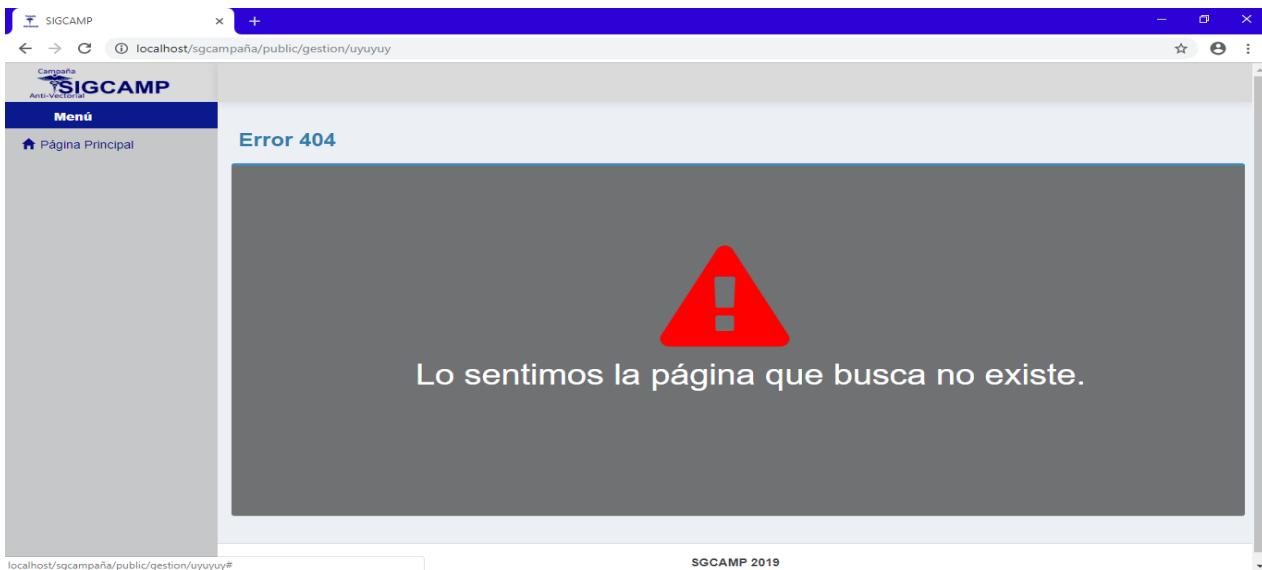


Imagen 4-15 Página de error 404

4.10 Diagrama de despliegue

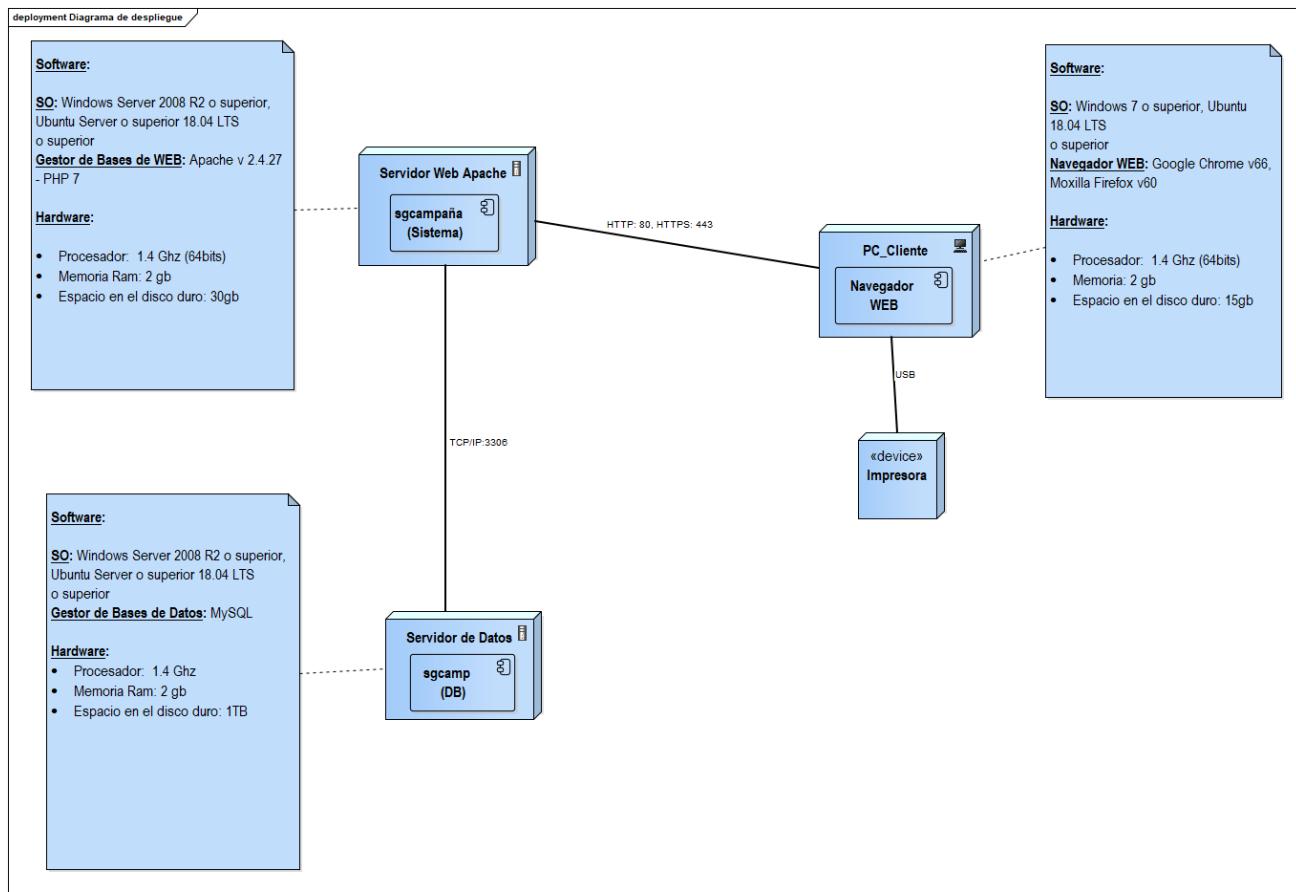


Imagen 4-16 Diagrama de despliegue

El diagrama de despliegue presenta la configuración de los elementos de hardware (nodos procesadores, dispositivos y protocolos), que se identificaron como necesarios para que el sistema tenga un buen funcionamiento, lo que significa, que el diagrama muestra un esquema de objetos que describe la distribución física del sistema, en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo.

El sistema a desarrollar es una aplicación web que se ejecutará sobre un nodo servidor web con Apache y se conectará al nodo servidor de base de datos, con MySQL como SGBD, donde quedará almacenada la información necesaria para el correcto funcionamiento del sistema. A esta información podrán estar accediendo varios nodos, PC cliente, de forma simultánea. Los mismos tendrán conexión con una impresora, por puerto USB o LTP, para imprimir los informes que se generarán del sistema. La comunicación entre los nodos PC _cliente y el nodo

Servidor web será usando el protocolo HTTP y el nodo Servidor web con el nodo Servidor de Base de Datos por TCP/IP.

4.11 Conclusiones parciales del capítulo

En este capítulo se realizó la descripción propuesta obteniéndose los siguientes resultados:

- Se realizó la descripción de los diagramas de clases organizados por paquetes detallando las interacciones que existen entre las clases para un mejor entendimiento y organización en el sistema.
- Se identifico el Modelo Vista Controlador como patrón de arquitectura utilizado por el framework Laravel determinando de esta forma la estructura del sistema.
- Se definieron los principios y patrones de diseño lo que permitió seguir un grupo de buenas prácticas o dogmas para el diseño, así como obtener soluciones efectivas a problemas comunes.
- Se puntuaron las normas de interfaz y el formato de salida de los reportes que se debe tener en cuenta para el desarrollo de la aplicación, así como la proyección del diseño de la base de datos y el tratamiento de los errores, para evitar equivocaciones innecesarias y garantizar la integridad de los datos.
- Se analizó el diagrama de despliegue, a través del cual se obtuvo una visión de los dispositivos e interconexiones necesarias para la implantación del sistema.

Capítulo 5 Validación y factibilidad de la solución propuesta.

5.1 Introducción.

En este capítulo se hace un análisis del esfuerzo, el tiempo y el costo que se invierte para realizar el desarrollo del sistema, así como se estudia los beneficios tangibles e intangibles, con el fin de conocer si es factible llevar a la práctica la ejecución de este sistema. Incluye además el análisis de las pruebas realizadas al sistema.

5.2 Pruebas

Durante la implementación del software es común que se cometan algunos errores los cuales por lo general se detectan en la etapa de prueba. La aplicación de pruebas a un producto de software es la vía más efectiva para determinar el estado de calidad que posee, pues estas tienen como propósitos:

- Verificar la interacción entre objetos.
- Verificar la interacción apropiada de todos los componentes del software.
- Verificar que todos los requisitos se han implementados correctamente.
- Identificar y asegurar que los defectos encontrados se han corregido antes del despliegue del software.

Principios de las pruebas de software.

Las pruebas se rigen por una serie de principios, una buena comprensión de estos facilitará el posterior uso de los métodos en un efectivo diseño de casos de prueba. A continuación, se citan:

La prueba puede ser usada para mostrar la presencia de errores, pero nunca su ausencia. La principal dificultad del proceso de prueba es decidir cuándo parar. Evitar casos de pruebas no planificados, no reusables y triviales a menos que el programa sea verdaderamente sencillo. Una parte necesaria de un caso de prueba es la definición del resultado esperado. Los casos de pruebas tienen que ser escritos no solo para condiciones de entrada válidas y esperadas sino también para condiciones no válidas e inesperadas. El número de errores sin descubrir es directamente proporcional al número de errores descubiertos.

Estas leyes que definen básicamente la aplicación de las pruebas de software ayudan a refinar el producto de software a través de las etapas involucradas.

Etapas involucradas en las pruebas de software.

Seleccionar qué es lo que debe medir la prueba, es decir, cuál es su objetivo, para qué exactamente se hace la prueba. Decidir cómo se va a realizar la prueba, es decir, qué clase de prueba se va a utilizar para medir la calidad y qué clase de elementos de prueba se deben usar. Desarrollar los casos de prueba. Un caso de prueba es un conjunto de datos o situaciones de prueba que se

utilizarán para ejecutar la unidad que se prueba o para revelar algo sobre el atributo de calidad que se está midiendo. Determinar cuáles deberían ser los resultados esperados de los casos de prueba y crear el documento que los contenga.

El tipo de prueba que se ejecutó para verificar el funcionamiento del sistema fue el de caja negra o funcionales.

Este tipo de pruebas evalúan el comportamiento observable del sistema. Consisten en verificar el resultado de la interacción entre los actores y el sistema, el cumplimiento de las precondiciones y postcondiciones y la secuencia de acciones especificadas. Para desarrollar estas pruebas existen varias técnicas:

- Técnica de partición de equivalencia: Permite examinar los valores válidos y no válidos de las entradas existentes en el software.
- Técnica de análisis de valores límites: Prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.
- Técnica de Grafos de causa-efecto: Permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

5.3 Diseño de los casos de prueba.

Caso de prueba #1: Registrar Usuario

Descripción de la prueba.

1. Verificar que todas las etiquetas de datos estén correctas.
2. Verificar que los campos del formulario cumplen con los parámetros establecidos
 - a. Roles: Comprobar que se despliegue un listado con los roles.
 - b. Área: Comprobar que se despliegue un listado con las áreas de salud existentes cuando el rol seleccionado no es jefe municipal ni administrador del sistema.
3. Verificar que los campos obligatorios (*) no queden vacíos.
4. Comprobar que al dar click en el botón Registrar no se envíe el formulario y muestre las notificaciones de error si los campos no están correctamente.

Tabla 5.1 Pruebas unitarias de tipo caja negra del caso de prueba Registrar Usuario. Técnica de partición de equivalencias

Condición de entrada	Dominio	Clases válidas	Clases no válidas	Reglas
Nombre Completo	Alfanumérico	Longitud <=50	Vacío, Longitud > 50	Único
Nombre de usuario	Alfanumérico	Longitud>6 Longitud <=50	Vacío, Longitud < 6, Longitud > 50	Único

Contraseña	Alfanumérico	Longitud>6 Longitud <=50	Vacio, Longitud < 6, Longitud > 50	Valores admitidos
Confirmar contraseña	Alfanumérico	Mismos valores que en el campo contraseña	Valores diferentes al campo contraseña	Mismos valores que en el campo contraseña
Roles	Conjunto de valores de entrada	Rol de usuario seleccionado 1) Jefe de área	No seleccionar Rol	Valores Admitidos que pertenecen al conjunto válido
Áreas	Conjunto de valores de entrada	Área seleccionada 1) Mulgoba	No seleccionar Área	Valores Admitidos que pertenecen al conjunto válido

Tabla 5.2 Pruebas de integración de tipo caja negra

Escenarios	Datos de prueba	Resultados obtenidos	Mensajes de salida
1. Registrar nuevo usuario	Nombre Completo: José García Pérez Nombre de Usuario: Jose1967 Contraseña: pepe1967 Confirmar contraseña: pepe1967 Roles: Jefe de Área Área: Mulgoba	Usuario registrado correctamente	Inspección registrada correctamente
1. Campos vacíos	Nombre Completo: Nombre de Usuario: Jose1967 Contraseña: Confirmar contraseña: Roles: Jefe de Área Área: Mulgoba	No se registró el usuario	Campo requerido Campo requerido Campo requerido

2. Los campos de contraseñas no coinciden	Nombre Completo: José García Pérez Nombre de Usuario: Jose1967 Contraseña: pepe1967 Confirmar contraseña: pepe1967 Roles: Jefe de Área Área: Mulgoba	No se registró la inspección	Por favor, escribe el mismo valor de nuevo.
--	---	------------------------------	--

Caso de prueba #2: Crear Plan de trabajo

Descripción de la prueba.

1. Verificar que todas las etiquetas de datos estén correctas.
2. Verificar que los campos del formulario cumplen con los parámetros establecidos.
 - a. Se muestre en el listado de brigadas a seleccionar las que no se le ha planeado plan en el día.
 - b. Validar los operarios disponibles no sea mayor a la cantidad de operarios que tiene la brigada seleccionada por plantilla.
 - c. Mostrar los campos para planificar las radiobatidas, las revisiones y las inspecciones si la brigada seleccionada no es la brigada de fumigación, por el contrario, si la brigada seleccionada es la de fumigación m los campos para planificar los tratamientos químicos.
Si la brigada seleccionada no es la de fumigación.
 - d. Validar la cantidad operarios de refuerzo no sea mayor a 10.
 - e. Mostrar el listado de selección múltiple con las manzanas en radiobatidas y el campo de texto operarios en radiobatidas solo cuando existan radiobatidas por la detección de focos o el reporte de febriles.
 - f. Mostrar en el listado de manzanas en la selección múltiple solo aquellas donde se han detectado focos o febriles.
 - g. Validar que la cantidad de operarios para las radiobatidas sea la cantidad total de viviendas de las manzanas en radiobatidas entre 25 y si es menor se muestre una notificación que se van realizar las radiobatidas con insuficientes hombres
 - h. Mostrar el campo de texto operarios para las revisiones si no se han realizado las revisiones de los centros priorizados y las zonas de riesgo en la semana y si hay operarios disponibles. Validar que la cantidad de operarios sea igual a la cantidad de total de los locales entre 25 y si es menor se muestre una notificación que se van realizar las revisiones con insuficientes hombres.

- i. Mostrar el listado de selección múltiple con las manzanas a inspeccionar y el campo de texto operarios para las inspecciones si no se han inspeccionado todas las manzanas en el ciclo y si hay operarios disponibles. Validar que la cantidad de viviendas de la manzana seleccionada no sea superior a la cantidad de hombres disponibles por 50 y que la cantidad de operarios sea igual a la cantidad de total de viviendas de las manzanas seleccionadas entre 50.
- Si la brigada seleccionada es la de fumigación.
- j. Mostrar el listado de selección múltiple con las manzanas en tratamientos químicos y el campo de texto operarios para tratamientos químicos solo cuando existan radiobatidas por la detección de focos o el reporte de febris. Validar que la cantidad de operarios sea igual a la cantidad de total de viviendas de las manzanas seleccionadas entre 100.
3. Verificar que los campos obligatorios (*) no queden vacíos.

Tabla 5.3 Pruebas unitarias de tipo caja negra del caso de prueba Crear plan de trabajo. Técnica de partición de equivalencias

Condición de entrada	Dominio	Clases válidas	Clases no válidas	Reglas
Brigada	Conjunto de valores de entrada	Brigada seleccionada (1) B1	No seleccionar una Brigada	Brigadas a las que no se le haya realizado plan de trabajo en el día.
Operarios disponibles	Numérico	Valor > 0 Valor > Cantidad de operarios en plantilla de la brigada seleccionada	No seleccionar una Etapa	Valor > Cantidad de operarios en plantilla de la brigada seleccionada
Operarios de refuerzo	Numérico	Valor > 0 Valor <=10 Valor en blanco	Valor <= 0 Valor > 10	Valores Admitidos que pertenecen al conjunto válido

Mostrar las Manzanas en radiobatidas	Tabla donde se muestran las manzanas en radiobatidas de la brigada seleccionada	Muestra las manzanas en radiobatida (1) 3890(40) (2) 3993(34)	No mostrar las manzanas en radiobatida	
Operarios para las radiobatidas	Numérico	Valor = Cantidad total de viviendas de las manzanas en radiobatida entre 25 y si es menor mostrar notificación que se realizaran las radiobatidas con insuficientes hombres	Valor > Cantidad total de viviendas de las manzanas en radiobatida entre 25 Escribir Letras	Valor = Cantidad total de viviendas de las manzanas en radiobatida entre 25 y si es menor mostrar notificación que se realizaran las radiobatidas con insuficientes hombres
Operarios para las revisiones	Numérico	Valor = Cantidad total de locales de los centros priorizados y las zonas de riesgo entre 25 y si es menor mostrar notificación que se realizaran las revisiones con insuficientes hombres	Valor > Cantidad total de locales de los centros priorizados y las zonas de riesgo entre 25 Escribir Letras	Valores Admitidos
Selección de las Manzanas a inspeccionar	Conjunto de valores de entrada	Cantidad total de las viviendas de las manzanas seleccionadas <=	Cantidad total de las viviendas de las manzanas seleccionadas >	Valores Admitidos que pertenecen al

		Cantidad de hombres disponibles por 40	Cantidad de hombres disponibles por 40.	conjunto válido
Operarios para las inspecciones	Numérico	Valor = Cantidad total de viviendas de las manzanas seleccionada a inspeccionar entre 50	Valor > Cantidad total de viviendas de las manzanas seleccionada a inspeccionar entre 50 Valor < Cantidad total de viviendas de las manzanas seleccionada a inspeccionar entre 50 Letras	Valores Admitidos que pertenecen al conjunto válido
Mostrar las Manzanas que se le realizaran Tratamientos químicos	Tabla donde se muestran las manzanas que se le realizaran Tratamientos químicos	Todas las Manzanas que se le realizaran Tratamientos químicos	No mostrar las Manzanas que se le realizaran Tratamientos químicos	
Operarios para los tratamientos químicos	Numérico	Valor = Cantidad total de viviendas de las manzanas a realizarle TQ entre 100	Valor > Cantidad total de viviendas de las manzanas a realizarle TQ entre 100 Escribir Letras	Valor = Cantidad total de viviendas de las manzanas a realizarle TQ entre 100

Tabla 5.4 Pruebas de integración de tipo caja negra

Escenarios	Datos de prueba	Resultados obtenidos	Mensajes de salida
1. Registrar nuevo plan de trabajo todos los hombres disponibles. Donde la brigada no es la de fumigación	Brigada: B1 Operarios disponibles: 12 Operarios de refuerzo: Manzanas en radiobatidas: 3678(50 viviendas) 3698(60 viviendas) 3648(40 viviendas) Operarios para las radiobatidas: 6 Operarios para las revisiones: 4 Manzanas a inspeccionar: (Seleccionadas) 3674(50 viviendas) 3648(50 viviendas) Operarios para las inspecciones: 2	Plan registrado correctamente	Plan registrado correctamente
2. Registrar nuevo plan de trabajo todos los hombres disponibles. Donde la brigada es la de fumigación	Brigada: B1 Operarios disponibles: 12 Operarios de refuerzo: Manzanas para Tratamientos químicos: 3678(50 viviendas) 3698(60 viviendas) 3648(40 viviendas) Operarios para Tratamientos químicos: 2	Plan registrado correctamente	Plan registrado correctamente

3. Campos vacíos	<p>Brigada:</p> <p>Operarios disponibles: Campo bloqueado</p> <p>Operarios de refuerzo:</p> <p>Manzanas en radiobatidas:</p> <p>No seleccionadas</p> <p>Operarios para las radiobatidas:</p> <p>Operarios para las revisiones: 4</p> <p>Manzanas a inspeccionar:</p> <p>No seleccionadas</p> <p>Operarios para las inspecciones:</p>	<p>No se registró el plan</p>	<p>Campo requerido</p> <p>Campo requerido</p> <p>Campo requerido</p> <p>Campo requerido</p> <p>Campo requerido</p> <p>Campo requerido</p>

Caso de prueba #3: Crear Inspección

Descripción de la prueba.

1. Verificar que todas las etiquetas de datos estén correctas.
2. Verificar que los campos del formulario cumplen con los parámetros establecidos.
 - a. Manzana: Comprobar que se despliegue un listado con las manzanas existentes de acuerdo al área del usuario y que no hayan sido inspeccionadas en el ciclo.
 - b. Viviendas existentes: Verificar que no esté disponible la inscripción de las viviendas existentes mientras no haya sido seleccionada una manzana.
 - c. Comprobar que la cantidad de viviendas existentes no sea superior a la cantidad de viviendas de la manzana seleccionada y que solo se permitan números enteros.
 - d. Viviendas cerradas: Verificar que no esté disponible la inscripción de las viviendas cerradas mientras el usuario no haya reflejado la cantidad de viviendas existentes.
 - e. Viviendas recuperadas: Comprobar que la cantidad de viviendas recuperadas no sea superior a la cantidad de viviendas cerradas que tiene la brigada en el ciclo.
 - f. Depósitos tratados: Verificar que no esté disponible la inscripción de los depósitos tratados mientras el usuario no haya reflejado la cantidad depósitos inspeccionados. Comprobar que la cantidad de depósitos tratados se encuentre entre el 20% y el 60% de los depósitos inspeccionados.
 - g. Depósitos no vistos: Comprobar que la cantidad de depósitos no vistos no sea superior a 10.

- h. Depósitos destruidos: Verificar que no esté disponible la inscripción de los depósitos destruidos mientras el usuario no haya reflejado la cantidad depósitos inspeccionados. Comprobar que la cantidad de depósitos destruidos se encuentre entre el 15% y el 40% de los depósitos inspeccionados.
 - i. Depósitos flameados: Verificar que no esté disponible la inscripción de los depósitos flameados mientras el usuario no haya reflejado la cantidad depósitos inspeccionados. Comprobar que la cantidad de depósitos destruidos se encuentre entre el 5% y el 30% de los depósitos inspeccionados.
 - j. Las cantidades gastadas de los insumos serán inferiores a las cantidades de los insumos correspondientes disponible en el almacén.
 - k. Comprobar que en los campos de entrada de datos los bordes cambien a color rojo siempre que la escritura este incorrecta de acuerdo al tipo de dato que puede contener.
3. Verificar que los campos obligatorios (*) no queden vacíos.

Tabla 5.5 Pruebas unitarias de tipo caja negra del caso de prueba Crear Inspección. Técnica de partición de equivalencias

Condición de entrada	Dominio	Clases válidas	Clases no válidas	Reglas
manzana	Conjunto de valores de entrada	Manzana no inspeccionada Seleccionada (1) 3678 (2) 3867	No seleccionar manzana o seleccionar una manzana que ha sido inspeccionada	Valores Admitidos que pertenecen al conjunto válido
Viviendas existentes	entero	cantidad <= viviendas de la manzana seleccionada longitud<=3	Cantidad > viviendas de la manzana seleccionada, Vació, Longitud >4	Valores Admitidos que pertenecen al conjunto válido
Viviendas cerradas	entero	cantidad <= 25% de las Viviendas existentes Viviendas cerradas longitud<=2	cantidad > 25% Viviendas existentes, Vació, seleccionada longitud>2	Valores Admitidos que pertenecen al conjunto válido

Viviendas negadas	entero	cantidad <= Viviendas cerradas longitud<=2	cantidad > Viviendas cerradas	Valores Admitidos que pertenecen al conjunto válido
Viviendas recuperadas	entero	cantidad <= 10 longitud<=2	cantidad > 10, Vació, longitud>2	Valores Admitidos que pertenecen al conjunto válido
Depósitos inspeccionados	entero	2 x viviendas inspeccionadas <= cantidad <= 5 x viviendas inspeccionadas longitud<=4	2 x viviendas inspeccionadas > cantidad > 5 x viviendas inspeccionadas Longitud>4, Vació	Valores Admitidos que pertenecen al conjunto válido
Depósitos tratados	entero	20% Depósitos inspeccionados <= cantidad <= 60% Depósitos inspeccionados longitud<=4	20% Depósitos inspeccionados > cantidad > 40% Depósitos inspeccionados Longitud>4, Vació	Valores Admitidos que pertenecen al conjunto válido
Depósitos no vistos	entero	cantidad < 10	cantidad > 10, Vació	Valores Admitidos que pertenecen al conjunto válido
Depósitos destruidos	entero	15% Depósitos inspeccionados <= cantidad <= 40% Depósitos inspeccionados longitud<=4	15% Depósitos inspeccionados > cantidad > 40% Depósitos inspeccionados Longitud>4, Vació	Valores Admitidos que pertenecen al conjunto válido
Depósitos flameados	entero	5% Depósitos inspeccionados <= cantidad <=	5% Depósitos inspeccionados > cantidad >	Valores Admitidos que pertenecen al conjunto válido

		30% Depósitos inspeccionados longitud<=4	30% Depósitos inspeccionados Longitud>4, Vació	
cantidades gastadas de los insumos	entero	Cantidad <= cantidad almacenada	Cantidad <= cantidad almacenada, Vació	Valores Admitidos que pertenecen al conjunto válido

Tabla 5.6 Pruebas de integración de tipo caja negra

Escenarios	Datos de prueba	Resultados obtenidos	Mensajes de salida
1. Registrar nueva inspección	Fecha: día actual Brigada: B1 Manzana: 3678(50 viviendas) Viviendas existentes: 50 Viviendas cerradas: 4 Viviendas recuperadas:1 Depósitos inspeccionados: 200 Depósitos tratados: 90 Depósitos no vistos: 1 Depósitos destruidos:60 Depósitos flameados:40 Cantidad Gastada (abate): 3000mg (almacenado:15000mg) Cantidad Gastada (alcohol): 1000mg (almacenado:9 000mg)	Inspección registrada correctamente	Inspección registrada correctamente
2. Campos vacíos	Fecha: día actual Brigada: B1 Manzana: 3678(50 viviendas) Viviendas existentes: 50 Viviendas cerradas: vacío Viviendas recuperadas:1 Depósitos inspeccionados: 200 Depósitos tratados: 90	No se registró la inspección	Campo requerido

	Depósitos no vistos: 1 Depósitos destruidos:60 Depósitos flameados: vacío Cantidad Gastada (abate): 3000mg (almacenado:15000mg) Cantidad Gastada (alcohol): 1000mg (almacenado:9 000mg)		Campo requerido
3. Campos numéricos fuera de los parámetros	Fecha: día actual Brigada: B1 Manzana: 3678(50 viviendas) Viviendas existentes: 60 Viviendas cerradas: 4 Viviendas recuperadas:1 Depósitos inspeccionados: 200 Depósitos tratados: 90 Depósitos no vistos: 20 Depósitos destruidos:60 Depósitos flameados:40 Cantidad Gastada (abate): 3000mg (almacenado:15000mg) Cantidad Gastada (alcohol): 1000mg (almacenado:9 000mg)	No se registró la inspección	<p>El número tiene que ser igual o menor a 50.</p> <p>El número tiene que ser igual o menor a 10.</p>

Caso de prueba #4: Crear Foco

Descripción de la prueba.

1. Verificar que todas las etiquetas de datos estén correctas.
2. Verificar que los campos del formulario cumplen con los parámetros establecidos.
 - a. Tipo de depósito: Comprobar que el valor no tenga una longitud superior a 50 y que no contiene números.
 - b. Etapa de la muestra: Comprobar que se despliegue un listado con las etapas existente.
 - c. Especie de la muestra: Comprobar que se despliegue un listado con las especies existentes.

d. Lugar donde se encontró el foco: Verificar que esté visible la inscripción del lugar donde se encontró el foco mientras el campo tipo de depósito sea distinto a larvitrampa y si es el caso que el campo se oculte y se vacíe si tiene algún lugar insertado.

Comprobar que la longitud no sea superior a 50 y que no contiene números.

e. Tipo de trabajo: Verificar que esté visible la selección del tipo de trabajo mientras el campo tipo de depósito sea distinto a larvitrampa y si es el caso que el campo se oculte y se deseccione si tiene alguno seleccionado.

e. Seleccionar la acción donde se detectó el foco: De acuerdo al tipo de trabajo seleccionado se desplegará un seleccionador con las acciones de este tipo.

f. Larvitrampa: Verificar que esté visible la selección de la larvitrampa mientras el campo tipo de depósito sea igual a larvitrampa y si es el caso que el campo se oculte y se deseccione si tiene alguno seleccionado. Comprobar que se despliegue un listado con las larvitrampas existentes.

g. Dirección: Comprobar que el valor no tenga una longitud superior a 150.

m. Comprobar que en los campos de entrada de datos los bordes cambien a color rojo siempre que la escritura este incorrecta de acuerdo al tipo de dato que puede contener.

3. Verificar que los campos obligatorios (*) no queden vacíos.

Tabla 5.7 Pruebas unitarias de tipo caja negra del caso de prueba Crear Foco. Técnica de partición de equivalencias

Condición de entrada	Dominio	Clases válidas	Clases no válidas	Reglas
Tipo de depósito	Alfanumérico	Longitud <=50	Vacío, Longitud > 50	Valores admitidos
Etapa de la muestra	Conjunto de valores de entrada	Etapa seleccionada (2) adulto (3) larvaria	No seleccionar una Etapa	Valores Admitidos que pertenecen al conjunto válido
Especie de la muestra	Conjunto de valores de entrada	Especie seleccionada (1) <i>Aedes Aegypti</i> (2) <i>Anopheles</i>	No seleccionar una etapa	Valores Admitidos que pertenecen al conjunto válido
Tipo de trabajo	radio	Marcado un tipo de trabajo.	No marcar un tipo de trabajo.	Valores Admitidos

		<ul style="list-style-type: none"> • Inspección diaria ○ Radiobatida ○ Revisión 		
Selección de la acción del tipo de trabajo marcado	Conjunto de valores de entrada	Seleccionada una acción del tipo de trabajo marcado (1) insd-B1-3782 (2) insd-B2-3456	No se selecciona una acción.	Valores Admitidos que pertenecen al conjunto válido

Tabla 5.2 Pruebas de integración de tipo caja negra

Escenarios	Datos de prueba	Resultados obtenidos	Mensajes de salida
1. Registrar nuevo foco donde el tipo de depósito no es una larvitrapa	Tipo de depósito: lata Etapa de la muestra: larvaria Especie de la muestra: <i>Aedes Aegypti</i> Lugar donde se encontró el foco: Patio Tipo de trabajo: inspección Inspección: insd-B2-3456	Foco registrado correctamente	Foco registrado correctamente
2. Registrar nuevo foco donde el tipo de depósito es una larvitrapa	Tipo de depósito: larvitrapa Etapa de la muestra: larvaria Especie de la muestra: <i>Aedes Aegypti</i> Tipo de trabajo: Revisión Revisión: rev-Heroes de Playa Girón	Foco registrado correctamente	Foco registrado correctamente
3. Campos vacíos	Tipo de depósito: vacío Etapa de la muestra: larvaria Especie de la muestra: <i>Aedes Aegypti</i> Tipo de trabajo: inspección Inspección: no seleccionado	No se registró el foco	Campo requerido Campo requerido

--	--	--	--

5.4 Resultados obtenidos en las pruebas.

Las pruebas que se realizaron en los diferentes escenarios fueron exitosas. Se encontraron diferentes defectos que fueron solucionados en su totalidad.

Tabla 5.8 Resultados obtenidos

Caso de prueba	Escenarios	Defectos interfaz	Defectos validación	Defectos funcionalidad	Total de defectos
Registrar Usuario	Registrar nuevo usuario	2	3	2	7
	Campos vacíos	1	3	2	6
	Registrar un usuario donde el rol a seleccionar es diferente a jefe de área y administrador del sistema	1	1	1	3
Registrar plan de trabajo	Registrar un plan donde la cantidad de operarios disponibles es inferior a la cantidad operarios necesarios para realizar el trabajo	3	3	4	10
	Campos vacíos	2	2	1	5

	Campos numéricos fuera de los parámetros	0	1	1	2
Crear Inspección	Registrar nueva inspección	0	0	1	1
	Campos vacíos	1	2	1	4
	Campos numéricos fuera de los parámetros	0	1	1	2
Crear Foco	Registrar nuevo foco donde el tipo de depósito no es una larvitrapa	0	0	1	1
	Campos vacíos	0	1	1	2

5.5 Resultados del estudio de Factibilidad [32]

Para realizar el estudio de factibilidad del proyecto se utilizó el método de estimación basada en CU. Los cálculos de esta estimación se encuentran plasmados en el [Anexo 2 Cálculo de la estimación del trabajo del proyecto](#), arrojando los siguientes resultados:

Tiempo de desarrollo de proyecto: 7200 horas o 2 años y 6 meses.

Costo del proyecto (CUP): \$ 19677.60

Costo del proyecto (CUC): \$ 819.90

5.6 Beneficios tangibles e intangibles

Los beneficios tangibles son las ventajas económicas cuantificables que obtiene la organización a través del uso del sistema informático y se pueden estimar en pesos, recursos y tiempo ahorrado.

Beneficios tangibles

- Ahorro monetario significativo al no tener que comprar un software a terceros. Este sistema está desarrollado en *software libre*, pues las tecnologías que se emplearon para su desarrollo están bajo licencias gratuitas, lo cual significa que el centro se ahorra costos por compra de *software* y licencias de pago.

- Se evita la pérdida de información ya que el sistema permite almacenar toda la información de forma consistente.
- Aumento del rendimiento y disminución de la carga de trabajo.
- Seguridad de la información y acortamiento en el tiempo de respuestas.
- El sistema contribuirá al aprovechamiento de los recursos tanto materiales como humanos.

Beneficios intangibles

Los beneficios intangibles son aquellos beneficios difíciles de cuantificar que obtiene una organización a través del uso de un sistema de información, pero por ello no dejan de ser menos importantes.

- Fácil acceso a información actualizada sobre las acciones realizadas y sus indicadores.
- Acceso a indicadores gráficos importantes para la toma de decisiones.
- Al ser un sistema que almacena todo en un repositorio central, permite que toda la información sobre las acciones realizadas permanezca en un solo lugar, elevando la seguridad e integridad de los datos.
- Disponibilidad de la información para ser visualizada o consultada cuando se requiera.

5.7 Análisis de costos y beneficios

El costo del sistema depende de diferentes aspectos como: la cantidad de desarrolladores que trabajan en él, el volumen de información que se procesará, el tiempo invertido para su implementación, entre otros. Como resultado del estudio de factibilidad, se puede afirmar que desde el punto de vista económico el trabajo representa numerosos beneficios, ya que responde a un ejercicio académico que no implica coste alguno para ninguna entidad; de lo contrario su desarrollo implicaría un costo de \$19677.60 CUP. Teniendo en cuenta estos elementos, los beneficios planteados con anterioridad y la relevancia que tiene la gestión de la Campaña Antivectorial a nivel nacional se puede alegar que la implementación del sistema es factible.

5.8 Conclusiones parciales del capítulo.

En este capítulo se obtuvo el resultado de las pruebas que permitió que los requisitos funcionales, incluyendo la navegación en el sitio, entrada de datos, procesamiento y obtención de resultados sean

de satisfactorios. Además, se realizó una estimación del tiempo y del costo total que tiene asociado el desarrollo del sistema propuesto. Se llevó a cabo un análisis de los beneficios tangibles e intangibles que se obtienen con la realización del mismo, así como también su costo total, arribando a la conclusión de que resulta factible el desarrollo de la aplicación. Obteniendo así los siguientes resultados:

- Para el caso de las pruebas se presentaron 16 defectos de funcionalidad, en el funcionamiento de las interfaces se detectaron 10 errores y 17 en el caso de las validaciones erróneas lo que permitió identificar los errores y solucionarlos.
- El sistema diseñado, el tipo de los actores del sistema es complejo, porque la persona interactúa con el sistema mediante una interfaz gráfica.
- El esfuerzo total empleado es de 6500 Horas / Hombre.
- El sistema implica un costo total que asciende a \$19677.60 CUP, en un tiempo estimado de 2 años y 6 meses.

Se concluyó que la aplicación propuesta resulta factible y le aporta al centro importantes beneficios tangibles e intangibles.

Conclusiones Generales

Una vez realizado el análisis y la ejecución del proyecto en cuestión, se concluye:

- Se estudiaron las aplicaciones relacionadas al objeto de estudio del sistema y se investigaron sus características, ventajas y desventajas, lo cual ilustró la necesidad de desarrollo del producto descrito.
- Se realizó la modelación del sistema propuesto partiendo de los requisitos pactados.
- El empleo de la metodología RUP resultó de gran utilidad como guía en el proceso de desarrollo, permitiendo dejar documentado en lenguaje UML todo lo requerido para las actividades de mantenimiento y futuras versiones del sistema.
- Las pruebas realizadas permitieron comprobar la efectividad de la aplicación desarrollada.
- Con el análisis de los costos estimados de la aplicación, y los beneficios tanto tangibles como intangibles, se demuestra que el desarrollo del proyecto es factible.

Los aspectos antes señalados permiten afirmar que se han cumplido los objetivos trazados para este trabajo.

Recomendaciones

Luego de finalizada la implementación de la aplicación y cumplidos los objetivos declarados en el inicio, con el objetivo de implementar una futura versión de ésta, se recomienda:

- Incorporar al sistema el uso de mapas que muestre las manzanas e incidencias de cada área de salud.

Referencias

- [1] J. R. Vázquez Canga, «Texto Básico para el Curso de Operarios y Técnicos Auxiliares en Vigilancia y Lucha Antivectorial.» La Habana, 2009.
- [2] «Velneo.» [En línea]. Available: www.velneo.es/lenguajes-de-programacion-mas-demandados/. [Último acceso: 29 Diciembre 2018].
- [3] MINSAP, «CAMPAÑA ANTI-Aedes AEGYPTI, Manual para responsables de brigada y visitadores.» La Habana, 2008.
- [4] A. Álvarez Valdés , . C. Díaz Pantoja y M. García Melian , «Sistema integrado de vigilancia para la prevención de dengue,» *REVISTA CUBANA MEDICINA TROPICAL*, nº 3, pp. 193-201, 2007.
- [5] A. Reiniso y S. León, «Vigiweb: alternativa de educación,» [En línea]. Available: <http://files.sld.cu/boletincnses>. [Último acceso: 24 Noviembre 2018].
- [6] Python, «Sitio Oficial de Python,» [En línea]. Available: <https://www.python.org>. [Último acceso: 27 Diciembre 2018].
- [7] PHP, «Sitio Oficial de PHP,» [En línea]. Available: www.php.net. [Último acceso: 14 Diciembre 2018].
- [8] Ruby, «Sitio Oficial de Ruby,» [En línea]. Available: www.ruby-lang.org. [Último acceso: 13 Diciembre 2018].
- [9] JQuery, «¿Qué es JQuery?.» [En línea]. Available: <http://api.jquery.com/> . [Último acceso: 28 Diciembre 2018].
- [10] «Introducción a jQuery,» [En línea]. Available: <http://es.slideshare.net/continuumslides/introduccion-a-jquery>. [Último acceso: 25 Diciembre 2018].
- [11] Bootstrap, «Sitio Oficial de Bootstrap,» [En línea]. Available: <http://getbootstrap.com> . [Último acceso: 22 Diciembre 2018].
- [12] Symfony, «Sitio Oficial de Symfony,» [En línea]. Available: <http://symfony.com>/. [Último acceso: 26 Diciembre 2018].
- [13] CakePHP, «Sitio Oficial de CakePHP,» [En línea]. Available: <http://cakephp.org>/. [Último acceso: 24 Diciembre 2018].
- [14] Laravel, «Sitio Oficial de Laravel,» [En línea]. Available: <http://www.laravel.com>/. [Último acceso: 23 12 2018].
- [15] PostgreSQL, «Sitio Oficial de PostgreSQL,» [En línea]. Available: <http://www.postgresql.org>. [Último acceso: 23 12 2018].
- [16] O. Corporation, «Sitio Oficial de MySQL,» Oracle Corporation, [En línea]. Available: <http://www.mysql.com>/. [Último acceso: 25 Diciembre 2018].

- [17] O. Corporation, «Sitio Oficial de Oracle,» Oracle Corporation, [En línea]. Available: <http://www.oracle.com/>. [Último acceso: 26 12 2018].
- [18] UML, «Sitio oficial de UML,» UML, [En línea]. Available: <http://www.uml.org/>. [Último acceso: 27 12 2018].
- [19] [En línea]. Available: www.eumed.net/libros-gratis/2018/metodologias-tradicionales-y-metodologias-agiles. [Último acceso: 15 5 2019].
- [20] Microsoft, «Microsoft Solutions Framework (MSF),» [En línea]. Available: [https://msdn.microsoft.com/es-es/library/jj161047\(v=vs.120\).aspx](https://msdn.microsoft.com/es-es/library/jj161047(v=vs.120).aspx). [Último acceso: 15 4 2019].
- [21] I. Jacobson, G. Booch y J. & Rumbaugh, «El Proceso Unificado de Desarrollo de Software,» vol. I, La Habana, Félix Varela, 2012.
- [22] Hanantek, «El modelo espiral WinWin,» 2017. [En línea]. Available: <http://www.hanantek.com/win-win>. [Último acceso: 13 4 2019].
- [23] R. Corporation, «Sitio oficial de Rational Rose Enterprise,» Rational Corporation, [En línea]. Available: <http://www.rational.com.ar/herramientas/roseenterprise.html>. [Último acceso: 27 12 2018].
- [24] V. Paradigm, «Sitio Oficial de Visual Paradigm,» Visual Paradigm, [En línea]. Available: <http://www.visual-paradigm.com/>. [Último acceso: 2018 Diciembre 24].
- [25] S. Systems, «Sitio Oficial de Enterprise Architect,» Sparx Systems, [En línea]. Available: <http://www.sparxsystems.com.ar/EAUserGuide/ea.html>. [Último acceso: 19 12 2018].
- [26] I. Jacobson, «Rational Unified Process,» 2011.
- [27] L. N. E., «LARAVEL NO ES MVC,» [En línea]. Available: <https://styde.net/porque-laravel-no-es-mvc-y-tu-deberias-olvidarte-de-mvc/>. [Último acceso: 24 12 2018].
- [28] K. Dokins, «Design Patterns in PHP and Laravel,» Apress Media, LLC, Arkansas, 2017.
- [29] A. Kılıçdağı y İ. YILMAZ, «Laravel Design Patterns and Best Practices,» Packt Publishing Ltd., Birmingham, 2014.
- [30] CUJAE, «Diseño de la BD-Mapeo objeto relacional,» de IS2 ANP 10 Diseño de la BD-Mapeo objeto relacional, La Habana, 2010.
- [31] j. Validation, «Sitio oficial de jQuery Validation,» jQuery Validation, [En línea]. Available: jqueryvalidation.org/documentation. [Último acceso: 29 Diciembre 2018].
- [32] CUJAE, «Cálculo de estimación y esfuerzo,» de IS1 ANP 28 Cálculo de estimación de un proyecto, La Habana, 2014.
- [33] «Foros del Web >> Programando para internet "Jquery enviar por ajax los valores de un select multiple",» 2017. [En línea]. Available: www.forosdelweb.com/f179/jquery-enviar-por-ajax-valores-select-multiple-831148/. [Último acceso: 26 diciembre 2018].

Glosario de siglas y términos

Ciclo: Es el tiempo en que se demoran en realizar las brigadas el tratamiento focal a toda el área de salud.

Acciones de prevención: Estas acciones son las inspecciones diarias y las revisiones.

Inspecciones diarias o Tratamiento focal: Es la inspección casa por casa buscando focos de mosquito de cualquier especie (entre los que pudieran aparecer de *Aedes Aegypti*), destruyéndolos y evitando que aparezcan en caso de que no se encuentren.

Radiobatida: Es una actividad de control integrado en la manzana donde se detectaron incidencias con el objetivo de detectar y eliminar todos los focos.

Tratamiento Adulticida: Es el tratamiento que se realiza con nebulización térmica (fumigación) y se lleva a cabo en el interior de las viviendas.

Febril: Son las personas que han sido detectadas con síntomas de fiebre alta persistente y que son sospechosos de tener el Dengue o Zika, así como personas llegadas de otro país con riesgo de estar infestado con estas enfermedades.

Indicadores de tratamiento focal: Cifras que son importantes para el seguimiento de las Zonas en que se estén trabajando, y para controlar la calidad del trabajo colectivo, de manera que un Área de Salud siempre tendrá un UNIVERSO que no es más que la cantidad de viviendas, locales, viviendas existentes, inspeccionadas, cerradas, negadas y recuperadas, cantidad de depósitos inspeccionados, destruidos, abatizados y la cantidad de abate gastado, de los indicadores antes mencionados parten todos los cálculos para la planificación y el cumplimiento de las actividades.

Vivienda existente: todo local enclavado, real, enmarcado en nuestra unidad territorial (manzanas, zonas, área de salud, municipios). Son las viviendas existentes las que suman nuestro universo real por inmuebles.

Viviendas inspeccionadas: son las viviendas existentes visitadas y verificadas por el personal de la campaña.

Viviendas cerradas: son aquellas viviendas existentes cuyos moradores están ausentes temporalmente. Las viviendas cerradas son un alto riesgo para la campaña por lo que deben ser censadas y vigiladas para su posible e inmediata inspección.

Vivienda recuperada: viviendas existentes que se presentaron cerradas en jornadas laborales anteriores y al comenzar un nuevo día hábil pueden ser inspeccionadas. No se anotan como viviendas existentes del día ya que fueron enumeradas como tal cuando estaban cerradas; pero si toda vivienda recuperada es una vivienda inspeccionada.

VI = VE - VC + VR.

Siendo:

VE: viviendas existentes

VI: viviendas inspeccionadas

VC: viviendas cerradas

VR: viviendas recuperadas

Índice de infestación: Focos de mosquitos colectados. El indicador se calcula dividiendo Viviendas Inspeccionadas entre muestras colectadas y da una muestra por cantidad de viviendas. Debe estar por debajo de 1 X 70.

Framework: Es una infraestructura digital que cuenta con una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos concretos de software, que puede servir de base para la organización y desarrollo de software. Típicamente, puede incluir soporte de programas, bibliotecas, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Anexos

Anexo 1

Caso de uso	Gestionar encuestas de moscas y cucarachas
Actores	Estadístico
Resumen	El CU inicia cuando el usuario registra la revisión de un centro priorizado. El sistema muestra el formulario de las encuestas. El usuario en este momento registra los datos de la encuesta. El sistema calcula los índices y determina el estado entomológico del centro priorizado y mostrara el formulario de encuestas si falta uno de los tipos de encuesta por registrar en caso contrario mostrara el formulario de revisiones.
Precondiciones	El usuario debe estar autenticado. Se debe haber una revisión de un centro priorizado.

Caso de uso	Gestionar revisiones
Actores	Estadístico
Resumen	El CU inicia cuando el usuario selecciona la opción revisión en el menú gestión de acciones. El sistema mostrara la lista de revisiones. En este momento el usuario puede escoger las opciones de insertar, modificar o eliminar. El CU finaliza cuando sistema notifica al usuario que los datos se insertaron, modificaron o eliminaron correctamente.
Precondiciones	El usuario debe estar autenticado. Se debe haber insertado al menos, una entidad que no haya revisado esa semana, una manzana donde se hayan detectado focos de <i>Aedes aegypti</i> o se hayan reportado febriles, una brigada, un área, un municipio y una provincia para insertar una revisión, además el usuario tiene que pertenecer al área insertada. Se debe haber insertado al menos una revisión para modificar y eliminar.

Flujo de trabajo	
Acción del usuario	Respuesta del sistema
1- El caso de uso inicia cuando el usuario selecciona la opción revisión en el menú gestión de acciones en la barra Menú.	2- El sistema muestra un listado con las revisiones y las diferentes opciones a realizar: A- Para insertar una nueva revisión ir a la sección Insertar revisión. B- Para modificar los datos de una revisión ir a la sección Modificar revisión. C- Para eliminar una revisión ir a la sección Eliminar revisión.
Sección A: Insertar revisión	
3- El usuario selecciona la opción de Insertar revisión. 	4- El sistema muestra la vista para insertar una revisión.
5- El usuario inserta los datos necesarios en el formulario y selecciona la opción registrar.	6- El sistema pasa a validar los datos. Si los datos están correctos se insertan en la base de datos y el CU finaliza cuando sistema notifica al usuario que los datos se insertaron correctamente.
Sección B: Modificar revisión	
3- El usuario selecciona la opción de modificar una revisión en el botón. 	4- El sistema a partir del id de la revisión muestra la vista para modificar la revisión con todos los datos asociados a esta.
5- El usuario modifica los datos necesarios en la vista y selecciona la opción registrar.	6- El sistema pasa a validar los datos. Si los datos están correctos se insertan en la base de datos y el CU finaliza cuando sistema notifica al usuario que los datos se modificaron correctamente.
Sección C: Eliminar revisión	

<p>3- El usuario selecciona la opción de eliminar una revisión específica en el botón.</p> 	<p>4- El sistema a partir del id de la revisión muestra una pantalla de confirmación y si acepta elimina la revisión y notifica al usuario que los datos se eliminaron correctamente finalizando así el CU.</p>
--	---

Caso de uso	Gestionar radiobatidas
Actores	Estadístico
Resumen	El CU inicia cuando el usuario selecciona la opción radiobatidas en el menú gestión de acciones. El sistema mostrara la lista de radiobatidas. En este momento el usuario puede escoger las opciones de insertar, modificar o eliminar. El CU finaliza cuando sistema notifica al usuario que los datos se insertaron, modificaron o eliminaron correctamente.
Precondiciones	El usuario debe estar autenticado. Se debe haber insertado al menos, una manzana, una brigada, un área, un municipio y una provincia para insertar una radiobatida, además el usuario tiene que pertenecer al área insertada. Se debe haber insertado al menos una radiobatida para modificar y eliminar.
Flujo de trabajo	
Acción del usuario	Respuesta del sistema
1- El caso de uso inicia cuando el usuario selecciona la opción radiobatidas en el menú	2- El sistema muestra un listado con las radiobatida y las diferentes opciones a realizar: A- Para insertar una nueva radiobatida ir a la sección Insertar radiobatida.

gestión de acciones en la barra Menú.	B- Para modificar los datos de una radiobatida ir a la sección Modificar radiobatida. C- Para eliminar una radiobatida ir a la sección Eliminar radiobatidas.
Sección A: Insertar radiobatida	
3- El usuario selecciona la opción de Insertar radiobatida.	4- El sistema muestra la vista para insertar una radiobatida. 
5- El usuario inserta los datos necesarios en el formulario y selecciona la opción registrar.	6- El sistema pasa a validar los datos. Si los datos están correctos se insertan en la base de datos y el CU finaliza cuando sistema notifica al usuario que los datos se insertaron correctamente.
Sección B: Modificar radiobatida	
3- El usuario selecciona la opción de modificar una radiobatida en el botón.	4- El sistema a partir del id de la radiobatida muestra la vista para modificar la radiobatida con todos los datos asociados a esta. 
5- El usuario modifica los datos necesarios en la vista y selecciona la opción registrar.	6- El sistema pasa a validar los datos. Si los datos están correctos se insertan en la base de datos y el CU finaliza cuando sistema notifica al usuario que los datos se modificaron correctamente.
Sección C: Eliminar radiobatidas	
3- El usuario selecciona la opción de eliminar una radiobatida específica en el botón.	4- El sistema a partir del id de la radiobatida muestra una pantalla de confirmación y si acepta elimina la revisión y notifica al usuario que los datos se eliminaron correctamente finalizando así el CU. 

Anexo 2 Cálculo de la estimación del trabajo del proyecto

Paso 1. Calcular los puntos de caso de uso sin ajustar

El cálculo de los Puntos de Caso de Uso sin ajustar (PCU) es el primer paso para la estimación y se obtiene a partir de la siguiente ecuación:

PCU = FPA + FPCU donde,

- PCU: Puntos de Caso de Uso sin ajustar
- FPA: Factor de Peso de los Actores sin ajustar
- FPCU: Factor de Peso de los Casos de Uso sin ajustar

Factor de peso de los actores sin ajustar (FPA)

Cantidad de actores de tipo complejo= 5

FPA= 3*5=15

Factor de peso de los casos de uso sin ajustar (FPCU)

Tabla 0.1 Cálculo del factor de peso de los casos de uso sin ajustar

Tipo de CU	Cantidad de CU	Factor de Peso	FPCU	Tipo de CU
Simple	36	5	180	Simple
Medio	2	10	20	Medio
Complejo	0	15	0	Complejo
Total	180			

$$FPCU = \sum [(Factor de peso i) * (Cantidad de CU de tipo i)]$$

Por tanto, FPCU = 180 Con los datos de FPA y FPCU obtenidos se puede determinar que:

$$PCU = FPA + FPCU$$

Por tanto, PCU = 15+ 200

$$PCU = 215$$

Paso 2. Calcular los pesos de casos de usos ajustados

Los puntos de casos de uso sin ajustar deben ser ajustados mediante la siguiente ecuación: **PCUA = PCU * FCT *FA** donde,

- PCUA: Puntos de Caso de Uso ajustados
- PCU: Puntos de Casos de Uso sin ajustar
- FCT: Factor de complejidad técnica
- FA: Factor de ambiente

Factor de complejidad técnica (FCT)

Este se compone de 13 puntos que evalúan la complejidad del sistema que se desarrolla, cada uno de estos factores tienen un peso definido con los cuales se obtendrán puntos ponderados por cada uno de ellos, según la valoración que se le asigne. Cada uno de los factores se cuantifica con un valor de 0 a 5.

Tabla 0.2 Cálculo del factor de peso de los casos de uso sin ajustar

Factor	Descripción	Peso	Valor	Peso * Valor
T1	Sistema distribuido.	2	0	0
T2	Tiempo de respuesta	1	5	5
T3	Eficiencia del resultado final.	1	4	4
T4	Procesamiento interno complejo.	1	4	4
T5	Reusabilidad	1	4	4
T6	Facilidad de instalación.	0.5	3	1.5
T7	Facilidad de uso.	0.5	4	2
T8	Portabilidad.	2	3	8
T9	Facilidad de cambio.	1	3	3
T10	Concurrencia.	1	5	5
T11	Objetivos especiales de seguridad.	1	5	5
T12	Provee acceso directo a terceras partes.	1	0	0
	$\Sigma(\text{Peso}_i * \text{Valor}_i)$			40.5

$$\text{FCT} = 0.6 + 0.01 * \Sigma (\text{Peso}_i * \text{Valor}_i)$$

Por tanto,

$$FCT = 0.6 + 0.01 * 40.5$$

$$FCT = 1.005$$

Cálculo del Factor de Ambiente (FA)

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo, por lo cual constituyen factores a contemplar en el cálculo del Factor de ambiente. En la siguiente tabla se muestra el significado y el peso de cada uno de estos factores:

Tabla 0.3 Cálculo del factor de peso de los casos de uso sin ajustar

Factor	Descripción	Peso (P)	Valor (V)	Peso * Valor
E1	Familiaridad con el modelo de proyecto utilizado	1.5	3	4.5
E2	Experiencia en la aplicación	0.5	2	1
E3	Experiencia en la orientación a objetos	1	4	4
E4	Capacidad del analista líder	0.5	4	2
E5	Motivación	1	5	5
E6	Estabilidad de los requerimientos	2	4	8
E7	Personal a tiempo parcial	-1	0	0
E8	Dificultad del lenguaje de programación	-1	0	0
$\Sigma(Pesos * Valores)$				24.5

$$FA = 1.4 - 0.03 * \Sigma (Peso_i * Valor\ asignado_i)$$

Por tanto,

$$FA = 1.4 - 0.03 * 24.5$$

$$FA = 0.665$$

Con los valores de PCU, FCT y FA obtenidos, se puede determinar que:

$$PCUA = PCU * FCT * FA$$

Por tanto,

$$PCUA = 215 * 1.005 * 0.665$$

$$PCUA = 143.69$$

$$PCUA = 144$$

Paso 3. Cálculo del Esfuerzo de Desarrollo (E)

El esfuerzo total calculado para el desarrollo de esta aplicación será de Horas-Hombres.

El esfuerzo en horas-hombre viene dado por:

$$E = PCUA * FC \text{ donde,}$$

- E: Esfuerzo estimado en horas-hombre.
- PCUA: Puntos de casos de uso ajustados.
- FC: Factor de conversión

Para el cálculo del FC se proponen valores según los siguientes criterios:

- Se contabilizan cuántos factores de los que afectan al FA están por debajo del valor medio (3), para los factores 1 al 6.
- Se contabilizan cuántos factores de los que afectan al FA están por encima del valor medio (3), para los factores 7 y 8.
- Si el total es 2 o menos, se utiliza el factor de conversión 20 horas-hombre/PCU, es decir, un PCU toma 20 horas-hombre.
- Si el total es 3 o 4, se utiliza el factor de conversión 28 horas-hombre/PCU, es decir, un PCU toma 28 horas-hombre.
- Si el total es mayor o igual que 5, se recomienda efectuar cambios en el proyecto, ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

Al seguir estos criterios se obtiene un FC de 20 horas-hombre y se puede calcular la estimación del esfuerzo en horas hombre mediante la siguiente ecuación:

$$E = PCUA * FC$$

Por tanto,

$$E = 144 * 20 \text{ H/H } E = 2880$$

Este esfuerzo es el que se requiere para la implementación. Si se tiene en cuenta que este representa un 40 % del esfuerzo total para desarrollar el software entonces se tiene que el esfuerzo total es: $E (\text{Total}) = E / 0.4$ Por tanto, $E (\text{Total}) = 2880 / 0.4$ **E (Total)** = 7200

Tabla 0.4 Estimación de las horas-hombre para el proyecto

Actividad	Porcentaje	Horas-Hombre
Análisis	10%	720
Diseño	20%	1440
Implementación	40%	2880
Pruebas	15%	1080
Sobrecarga (Otras actividades)	15%	1080
Estimación Total	100%	7200

Estimación del Tiempo de desarrollo del proyecto (TDes)

A partir del esfuerzo en horas hombre, es posible obtener otras estimaciones. Una de ellas es el tiempo de desarrollo aproximado del proyecto que se calcularía de la siguiente manera:

$$\mathbf{TDes = E (total) / CH}$$

Donde: **TDes**: Tiempo de desarrollado del proyecto

E (total): Esfuerzo total (7200H/H)

CH: Cantidad de hombres que desarrollan el proyecto (1 hombre)

Por lo tanto, la estimación del tiempo de desarrollo del proyecto sería:

$$\mathbf{TDes = E (total) / CH}$$

$$\mathbf{TDes = 7200H/H / 1 Hombre = 7200 horas}$$

$$\mathbf{TDes = 7200 horas / 10 horas-día = 720 días}$$

$$\mathbf{TDes = 720 días / 24 días-mes}$$

$$\mathbf{TDes = 30 meses (240 horas/mes) \approx 30 meses \approx 2 años y 6 meses}$$

Estimación del Costo de desarrollo del proyecto

Una vez estimado el tiempo de desarrollo del proyecto y conociendo la cantidad y características de los desarrolladores del mismo se puede llevar a cabo una estimación del costo total del proyecto, el cual viene dado por la expresión:

$$\mathbf{C \text{ (total)} = C \text{ (p)} + OG}$$

Donde: **C (total)**: Costo total del proyecto

C (p): Costo del proyecto

OG: Otros gastos

Para realizar el cálculo del costo del proyecto se tiene la fórmula siguiente:

$$\mathbf{C \text{ (p)} = E \text{ (total)} \times CHH}$$

Donde: **E (total)**: Esfuerzo total (7200 H/H)

CHH: Costo por horas-hombre

Para realizar el cálculo del Costo por horas-hombres, se tiene la siguiente fórmula:

$$\mathbf{CHH = K \times THP}$$

Donde: **K**: Coeficiente que tiene en cuenta los costos indirectos (\$1.50 CUP).

THP: Tarifa Horaria Promedio. El salario promedio de las personas que trabajan en el proyecto dividido entre 192 horas.

$$\mathbf{THP = \$ 350.00 / 192 \text{ horas} = \$1.822}$$

$$\mathbf{C \text{ (p)} = E \text{ (total)} \times CHH}$$

$$\mathbf{C \text{ (p)} = E \text{ (total)} \times K \times THP}$$

$$\mathbf{C \text{ (p)} = 7200 \text{ H/H} \times 1.5 \times 1.822 = 19677.60 \text{ CUP}}$$

$$\mathbf{C \text{ (p)} = 19677.60 / 24 = 819.90 \text{ CUC}}$$

Por lo que se estima un costo de desarrollo del proyecto de 19677.60 CUP o lo equivalente a 819.90 CUC, para una taza de cambio de 1CUC x 24 CUP.