

Universidad de las Ciencias Informáticas

Facultad 4



**Sistema de gestión de información sobre las tortugas marinas en el proyecto
Archipiélagos del Sur**

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas

Autora: Magdalys Martínez Jiménez

Tutores: Ing. Rosalba Carralero Medina

Ing. Marcel Puentes Rojas

La Habana, junio de 2013

Año 55 de la Revolución

Declaración de autoría

Declaro que soy la única autora de este trabajo y autorizo a la Facultad 4 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

“Magdalys Martínez Jiménez”

“Rosalba Carralero Medina”

“Marcel Puentes Rojas”

Agradecimientos

A mi mamá y a mi papá porque sin ellos no hubiera podido llegar tan lejos, su apoyo incondicional, su dedicación y consejos me han convertido en la profesional que soy hoy. Ellos siempre creyeron en mí y esta tesis así lo evidencia.

A mis hermanos Fidelito y Raidel que siempre me han protegido desde pequeños y siempre se han preocupado por mis estudios.

A mi tía Magday que ha sido mi segunda madre y me apoyó en una etapa muy importante de mi vida.

A mi novio Marcel que en todo lo que he necesitado siempre me ha ayudado. Por darme tantas horas de su tiempo. Por su apoyo, cariño y amor incondicional.

A mi mejor amiga Yarelys que aunque esté muy lejos siempre se ha preocupado por todo lo que me acontece.

A mis amistades que me ayudaron a pasar mejor los momentos en la UCI Nellis, Kathy, Sandra, Ole, Aimet, Mily, Dayanis, Odelvis, Yary, Leannys, Pedro, Luis, Osmel, los dos Leandro, Adrian, Pepe, Heriberto y Adriel. A mis compañeros de grupo en especial a Cuni, Lanneis, Ramel, Amado, Susi, Yoe, Basulto, Anay, Randy, Adia, Ivis, Hector y Ortiz.

A mi gran familia, cariñosamente apodadas las locas Lily, Yudi, Kare, Mary, Aliu, Yisi, Yasi. A Arcadio y Mara por compartir conmigo tres cursos en la universidad.

A mis tutores Rosalba por dedicar parte de su tiempo en las dudas que tenía. A Marcel por darme el tema de tesis cuando más lo necesitaba y por atenderme en todo momento cuando lo necesité.

A Wilfredo que en cada duda que tuve siempre me atendió, es una persona especial.

Al equipo de Softball por todas las victorias que obtuvimos.

A todos los profesores que contribuyeron a mi desarrollo como profesional.

Dedicatoria

A mi mamá porque desde pequeña siempre conversaba mucho conmigo y no solo hacía papel de madre sino de profesora. Nunca se cansó de enseñarme y prepararme para cada paso que diera no fuera en falso. Gracias por hacerme participar en cada concurso de dibujo, disfraz y baile me enseñaron mucho a desenvolverme como persona.

A mi papá porque gracias a todo lo que me ha enseñado logré ser una de las deportistas destacadas de la facultad; una persona de grandes sentimientos, que muchas veces pone los intereses de otros por encima de los suyos y por confiar en mí.

A mi hermano Fidelito que también ha sido mi amigo incondicional y me ha escuchado siempre que lo he necesitado.

A mi hermano Raidel que en los primeros años de estudio sirvió de ejemplo para esforzarme y ser mejor estudiante.

A mi tía Magday por tantas cosas que sacrificó para cuidarnos a mis hermanos y a mí.

A mi novio Marcel, mi amigo, mi tutor, mi amor; siempre apoyándome en los momentos difíciles.

A mi familia que siempre ha estado pendiente de cómo han ido mis estudios.

Resumen

Los sistemas de gestión de información permiten a las organizaciones mejorar el procesamiento de la información que se obtiene a partir de los procesos que realizan. El Sistema Nacional de Áreas Protegidas desarrolla un programa para el estudio de tortugas en el marco del proyecto Archipiélagos del Sur. Este genera gran volumen de datos, que en la actualidad son procesados manualmente, lo cual dificulta su almacenamiento y mantenimiento, así como la obtención de estadísticas relacionadas al estado real de las tortugas marinas en Cuba. El presente trabajo describe el desarrollo de un sistema para la gestión de información, sobre el estado de conservación de estas especies, que permite mejorar el control y manejo de la información que se obtiene como parte de los estudios. El mismo fue desarrollado utilizando la metodología RUP, basándose en el framework Symfony y el lenguaje PHP. Se realizaron los flujos Requerimientos, Análisis y diseño, Implementación y Prueba; resultando un sistema que responde a los requerimientos funcionales y no funcionales definidos. La aplicación desarrollada permite evitar errores en el análisis de datos que se introducen cuando este no se realiza de forma automatizada; unifica en una sola base de datos la información de todo el país, facilitando el acceso a la misma de forma rápida y sencilla.

Palabras claves: sistema de gestión de información, tortugas marinas

Índice de contenidos

Introducción	1
Capítulo 1. Fundamentación teórica	6
Introducción	6
1.1 Sistemas de gestión de información	6
1.2 Sistemas similares sobre las tortugas marinas	7
1.2.1 Fondo Mundial para la Naturaleza	7
1.2.2 Convención Interamericana para la Protección y Conservación de las Tortugas Marinas	8
1.2.3 Red colombiana para la conservación de tortugas marinas	9
1.3 Sistemas nacionales	10
1.4 Valoración sobre las soluciones anteriores	10
1.5 Herramientas y tecnologías a utilizar	11
1.5.1 Metodología de desarrollo de software	11
1.5.2 Lenguaje de modelado	14
1.5.3 Herramienta CASE para el modelado UML	15
1.5.4 Tecnologías del lado del cliente	16
1.5.5 Capa de acceso a datos	19
1.5.6 Sistemas gestores de bases de datos	19
1.5.7 Servidor web Apache 2.2.21	21
1.5.8 Framework	21
1.5.9 Entorno de desarrollo integrado	23
1.5.10 Bootstrap 2.3	25
Conclusiones del capítulo	26
Capítulo 2: Propuesta de solución	27
Introducción	27
2.1 Descripción del negocio	27
2.2 Modelo de dominio	28
2.2.1 Conceptos del dominio	28
2.2.2 Diagrama del Modelo de dominio	29
2.3 Descripción del sistema propuesto	30
2.4 Requerimientos funcionales	31
2.5 Requerimientos no funcionales	33
2.6 Patrones de caso de uso	35

2.7 Modelo de casos de uso del sistema _____	35
2.7.1 Descripción de los actores del sistema _____	36
2.7.2 Diagrama de casos de uso del sistema _____	36
2.7.3 Descripción de los casos de uso del sistema _____	37
Conclusiones del capítulo _____	43
Capítulo 3: Análisis y diseño _____	44
Introducción _____	44
3.1 Patrón arquitectónico Modelo- Vista- Controlador en Symfony2 _____	44
3.2 Patrones de diseño _____	45
3.3 Modelo de diseño _____	47
3.3.1 Diagrama de clases del diseño _____	47
3.3.2 Diagrama de despliegue _____	48
3.3.3 Diagramas de secuencia _____	49
3.4 Diseño de la base de datos _____	51
3.4.1 Descripción de las tablas _____	52
Conclusiones del capítulo _____	54
Capítulo 4: Implementación y prueba _____	55
Introducción _____	55
4.1 Modelo de implementación _____	55
4.1.1 Diagrama de componentes _____	55
4.2 Pruebas de software _____	57
4.2.1 Estrategia de prueba _____	58
4.2.2 Niveles de prueba _____	58
4.2.3 Métodos de prueba _____	59
4.2.4 Partición equivalente _____	60
4.2.5 Diseño de casos de prueba _____	60
4.2.6 Resultados obtenidos en las pruebas _____	68
Conclusiones del capítulo _____	68
Conclusiones _____	69
Recomendaciones _____	70
Referencias bibliográficas _____	71
Bibliografía _____	74
Glosario de términos _____	75

Introducción

El desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC) ha tenido un impacto positivo en casi todos los sectores de la sociedad. Diversos sistemas o herramientas informáticas permiten gestionar los extensos volúmenes de información generados. Ante la necesidad de manejar gran cantidad de datos, procesarlos y obtener de ellos información pertinente como apoyo a la toma de decisiones surgen los sistemas para la gestión de la información. Los mismos están formados por un conjunto de elementos que interactúan entre sí con el objetivo de apoyar las actividades de una organización. Estas herramientas en sentido general realizan cuatro actividades básicas: entrada, almacenamiento, procesamiento y salida de información. Lo anterior permite evitar errores en el análisis de datos que se introducen cuando este se realiza de forma manual; mejorando la eficiencia en los procesos involucrados.

En Cuba muchas entidades o empresas carecen de herramientas de este tipo para mejorar sus procesos. Tal es el caso del Sistema Nacional de Áreas Protegidas (SNAP) que con el apoyo de varias organizaciones no gubernamentales y otras instituciones cubanas desarrolla el proyecto Archipiélagos del Sur. Este contempla nueve programas para el estudio de especies de la flora y la fauna que se encuentran amenazadas o en peligro de extinción.

Uno de estos programas está relacionado con las tortugas marinas. Las mismas marinas juegan un papel clave en dos ecosistemas fundamentales para su desarrollo: los océanos y las dunas de las playas. En este programa se observan un grupo de insuficiencias relacionadas a la captura y procesamiento de la información que imposibilitan obtener mejores resultados. Como son:

- La existencia de un total de veintiún coordinadores de los programas de monitorización, pertenecientes a organismos diferentes, lo cual implica que la información que se procesa no siga un mismo patrón y a menudo sea incompleta o errónea.
- El número de áreas protegidas a estudiar asciende a veintiocho, ubicadas a lo largo de la geografía cubana. Este valor puede continuar ascendiendo si se detectan otros ecosistemas en peligro o con afectaciones de interés, lo cual exige una constante comunicación e intercambio de información entre las áreas y el nivel nacional.

- La información recopilada es muy extensa: teniendo en cuenta que el estudio se desarrolla en todo el territorio nacional y en él se registra el comportamiento de una gran cantidad de variables que permiten caracterizar el estado actual de cada área de anidación estudiada.
- Cada coordinador científico de los programas de monitorización realiza el análisis de los datos crudos utilizando hojas de cálculo en Excel y el software propietario Statistica. Los mismos limitan el procesamiento que se puede hacer de la información recopilada en los estudios realizados. Además se dificulta el mantenimiento de la misma a nivel nacional.
- No se garantiza la seguridad de la información debido a que cualquier persona con acceso al lugar donde se almacenan estos datos podría realizar cambios relevantes en los mismos, corriéndose el riesgo de su destrucción al no estar respaldados en otros soportes.
- No existe una base de datos que agrupe en un mismo lugar la información de los estudios realizados en las diferentes áreas del país.

Es contradictorio de una parte, que exista el conocimiento y las tecnologías necesarias para mejorar el procesamiento que en la actualidad se realiza sobre estos datos para el control y conservación de una especie en extinción en la fauna cubana y de otra parte, no existan herramientas informáticas que permitan un mejor trabajo en el estudio de las diferentes especies de tortugas marinas.

Teniendo en cuenta la situación anterior, se define el **problema de investigación** en la siguiente interrogante: ¿Cómo contribuir a mejorar el control y manejo de la información que se obtiene en el programa de tortugas marinas del proyecto Archipiélagos del Sur?

Para resolver el problema anterior se establece como **objetivo general**, desarrollar un sistema de gestión de información sobre el estado de conservación de las tortugas marinas, que permita mejorar el control y manejo de la información que se obtiene en los estudios.

Para dar cumplimiento al objetivo general se derivan los siguientes **objetivos específicos**:

- Elaborar el marco teórico de la investigación.
- Analizar las herramientas y tecnologías para el desarrollo de la solución.

- Definir las funcionalidades y características que debe cumplir el sistema.
- Implementar un sistema de gestión de información que cumpla con los requerimientos definidos.
- Aplicar pruebas de calidad al sistema de gestión de información resultante.

El **objeto de estudio** se enmarca en los sistemas para la gestión de información sobre tortugas marinas, teniendo como **campo de acción** el sistema de gestión de información sobre tortugas marinas para el proyecto Archipiélagos del Sur.

Se tiene como **idea a defender** que si se desarrolla un sistema de gestión de información será posible mejorar el control y manejo de la información que se obtiene en el programa de tortugas marinas del proyecto Archipiélagos del Sur.

Con el propósito de dar cumplimiento a lo anteriormente planteado, se elaboraron las siguientes **tareas de investigación**:

- Revisión bibliográfica para generar el marco teórico conceptual en lo referente al desarrollo de sistemas de gestión de información.
- Análisis de los principales fallos y deficiencias de la forma actual de procesar la información sobre el estado de conservación de las tortugas marinas.
- Investigación de las herramientas existentes para realizar la implementación del sistema.
- Identificación de los requerimientos funcionales y no funcionales del sistema de gestión de información sobre el estado de conservación de las tortugas marinas.
- Desarrollo de los artefactos necesarios de acuerdo con la metodología seleccionada.
- Implementación de las funcionalidades propuestas, teniendo en cuenta los requisitos definidos anteriormente.
- Realización de pruebas funcionales al sistema desarrollado.

En la investigación se utilizan los siguientes métodos investigativos:

Métodos teóricos:

- *Histórico - Lógico*: para el estudio del desarrollo y evolución de los diferentes sistemas de gestión de información similares nacionales e internacionales, así como las herramientas y tecnologías para el desarrollo del software, entre ellos los lenguajes de programación, framework, metodologías y herramientas CASE.
- *Analítico - Sintético*: para la realización del análisis de la información empleada para la investigación, identificando así, conceptos, definiciones y avances acerca de los sistemas de gestión de información existentes.
- *Inductivo - Deductivo*: permite el arribo a conclusiones a partir del estudio, según corresponda, de las características generales y/o particulares de los elementos, procesos o fenómenos de interés para la investigación.

Se espera como **posibles resultados** de esta investigación que el desarrollo del sistema permita:

- Gestionar la información recopilada durante los estudios.
- Centralizar en una base de datos los registros históricos nacionales de los elementos en estudio.

El presente trabajo de diploma cuenta con la siguiente estructura: introducción, tres capítulos, conclusiones, recomendaciones y por último referencias bibliográficas. A continuación un resumen de las diferentes temáticas que se abordan en los capítulos.

Capítulo 1: “Fundamentación teórica”

En este apartado se encuentra el marco teórico conceptual que permite develar la parte de la ciencia que está siendo estudiada, además, se realiza el estado del arte de los sistemas de gestión de información nacional e internacional de los estudios a las tortugas marinas, así como la descripción de las principales herramientas y tecnologías a utilizar para el desarrollo de un sistema de gestión de información sobre el estado de conservación de las tortugas marinas.

Capítulo 2: “Propuesta de solución”

En este capítulo se realiza una propuesta de solución basada en el estudio del sistema de gestión de información sobre el estado de conservación de las tortugas marinas. Se

especifican los requisitos definidos y se describen utilizando los diagramas de UML. Se generan otros artefactos y también se describe cómo está realizado el sistema.

Capítulo 3: “Análisis y diseño”

En este apartado se realiza el diseño de las funcionalidades mediante los diagramas de clases del diseño, diagramas de secuencia, el modelo de datos, descripción de las tablas y el diagrama de despliegue los cuales brindan una visión clara del producto.

Capítulo 4: “Implementación y prueba”

En este último capítulo se realiza una descripción de los módulos utilizados en la solución. Se implementan todas las funcionalidades, obteniendo un sistema con los requerimientos solicitados por el cliente. Se describen las pruebas realizadas al sistema una vez terminado, asegurándose la calidad del mismo.

Capítulo 1. Fundamentación teórica

Introducción

En la actualidad los programas informáticos han devenido en sistemas muy versátiles: entre ellos se encuentran los sistemas de gestión de la información, programas que facilitan el procesamiento de los datos en disímiles contextos. En el plano ambiental se han desarrollado diversas herramientas para evaluar impactos, medir variables, crear bases de conocimientos, entre otros.

Numerosas organizaciones se han enfocado a evaluar el estado de la diversidad biológica. El estudio de las tortugas marinas desempeña un papel importante para la conservación de los ecosistemas costeros. Por sus hábitos migratorios, la conservación de estos quelonios debe tratarse a nivel regional pues cubren enormes distancias que abarcan aguas territoriales de varios países.

1.1 Sistemas de gestión de información

Según la bibliografía consultada, varios autores establecen que los sistemas de gestión de información han sido ampliamente implementados en los entornos empresariales como un enfoque sistemático para la recopilación y presentación de la información necesaria para la toma de decisiones. (1)

Otra definición refleja que es un proceso mediante el cual los datos (entrada) se registran, almacenan, recuperan y procesan para la toma de decisiones (salida). (2)

Según el concepto que establece la Universidad de California un Sistema de Gestión de Información (IMS por sus siglas en inglés) es la aplicación de la tecnología de la información para apoyar las principales funciones y actividades de un negocio ya sea del sector privado o institución del sector público. En el pasado, las organizaciones reconocían la importancia de la gestión de los recursos, como el trabajo, el capital y las materias primas. Hoy día, está ampliamente aceptado que la gestión del recurso de información es muy a menudo igualmente importante. Un IMS apoya el proceso de recolección, manipulación, almacenamiento, distribución y utilización de los recursos de información de una organización. (3)

Los gerentes utilizan un IMS para crear informes que les proporcionen una visión completa de toda la información que necesitan para tomar decisiones que van desde pequeños detalles diarios a nivel superior de estrategia. Los sistemas actuales de gestión de la información se basan en gran medida en la tecnología para recopilar y presentar datos, pero el concepto es más antiguo que las tecnologías informáticas modernas. (4)

A partir de lo planteado se establece para la presente investigación que un sistema de gestión de información recopila información de una variedad de fuentes, la compila y presenta en un formato legible, ayudando a los usuarios en la prestación de los servicios, la recogida y búsqueda de datos, análisis y extensión de información

1.2 Sistemas similares sobre las tortugas marinas

Luego del estudio realizado es posible comentar que internacionalmente existen sitios que se centran en la labor de informar a sus usuarios sobre temas relacionados a las investigaciones referentes a las tortugas marinas, sin embargo no se encuentran sitios dedicados a la gestión de los datos que se manejan en los estudios realizados a las mismas. A continuación se describen algunos sitios internacionales sobre tortugas marinas:

1.2.1 Fondo Mundial para la Naturaleza

El Fondo Mundial para la Naturaleza (WWF¹ por sus siglas en inglés) tiene como objetivo la preservación de la naturaleza su sitio (disponible en: http://www.wwfca.org/about/somos222/programa_tortugas/) se centra en la muestra de información sobre todo el trabajo que realizan. Incluyen investigaciones sobre la conservación de las tortugas marinas en las playas de Centro América, hacen mención de los planes estratégicos de la red de tortugas en la zona y a los programas que desarrollan para la conservación de las diferentes especies. Su trabajo se extiende no solo al estudio de especies también a los bosques, agua dulce, costas y mares.

¹ World Wildlife Fund



Central America

Home » WWF Centroamérica » Quiénes somos » Programa de Tortugas Marinas (LAC)

Programa de Tortugas Marinas para América Latina y el Caribe

© WWF/ Gabriel FRANCIA

Por sus hábitos migratorios, la conservación de las tortugas marinas debe tratarse a nivel regional pues cubren enormes distancias que abarcan aguas territoriales de varios países.

Figura 1: Sitio WWF

1.2.2 Convención Interamericana para la Protección y Conservación de las Tortugas Marinas

La Convención Interamericana (CIT) es un tratado intergubernamental que provee el marco legal para que los países del Continente Americano tomen acciones en favor de estas especies. (5)

El objetivo de este sitio (disponible en: <http://www.iacseaturtle.org/default.htm>) es la muestra de información acerca de las acciones que realiza la CIT, hacen alusión a las conferencias que realizan, informes anuales y las publicaciones que se desarrollan.

En él se promueve la protección, conservación y recuperación de las poblaciones de tortugas marinas y de los hábitats de los cuales dependen, sobre la base de los datos más fidedignos disponibles y considerando las características ambientales, socioeconómicas y culturales de las partes. Estas acciones deben ser tomadas tanto en las playas de anidamiento como lo correspondiente a los mares territoriales de los países. (5)



Figura 2: Sitio de la CIT

1.2.3 Red colombiana para la conservación de tortugas marinas

Esta red se encarga mediante su sitio (disponible en: <http://redtortugasmarinascolombia.ning.com/>) de exhortar a las comunidades locales como actores esenciales en el proceso de conservación de las tortugas marinas. También se refleja la realización de eventos, foros, fotos, videos y una biblioteca referente al trabajo con las diferentes especies de tortugas marinas en Colombia.

En este sitio solo se puede encontrar información referida a las tortugas marinas, no se efectúa ningún tipo de gestión de datos sobre el estudio de estas exóticas especies.



Figura 3: Sitio Red colombiana para la conservación de tortugas marinas

1.3 Sistemas nacionales

Actualmente en Cuba no existen sistemas que gestionen la información sobre el estudio que se realiza a las tortugas marinas en las áreas protegidas. Solo se encuentran sitios como los mencionados anteriormente, que se encargan de brindar a los usuarios información sobre las acciones y mecanismos que utilizan para la protección de esas especies, tal es el caso del portal del Medioambiente² en Cuba y el SNAP³.

El mecanismo que utilizan las organizaciones cubanas para la gestión y almacenamiento de los datos de estos estudios consiste en el registro en una hoja de cálculo de Excel de todas las acciones que se conciben, la cual suele ser muy extensa lo que trae consigo que al efectuar una búsqueda se dificulte arribar a ciertos resultados.

1.4 Valoración sobre las soluciones anteriores

El estudio realizado evidencia que ninguno de los sistemas anteriores posee las características que normalmente están presentes en una herramienta de gestión de información, en este caso relacionada a la conservación de las tortugas marinas. Sin embargo

² Disponible en el sitio www.medioambiente.cu

³ Disponible en el sitio www.snap.cu

se tienen en cuenta las características de otros sistemas para el desarrollo de la investigación, entre ellas:

- Una estructura en la cual está presente un módulo de administración encargado de la gestión de usuarios, roles, permisos y otras entidades que engloban procesos importantes de la información del propio sistema.
- Uno o varios módulos relacionados con la gestión de la información que se maneja en cada sistema para permitir su creación, modificación y visualización de forma individual y general.
- La posibilidad de obtener un conjunto de reportes para visualizar los datos relevantes.

1.5 Herramientas y tecnologías a utilizar

Para el desarrollo de la solución se hace necesario el estudio de varias herramientas y tecnologías, a continuación la presentación y valoración de las mismas.

1.5.1 Metodología de desarrollo de software

El éxito de un proyecto es muy importante para la empresa que lo desarrolla, pues los beneficios serían cuantiosos. El uso de las metodologías de software propicia que en un proyecto se gestionen, administren, ejecuten y cumplan todas las tareas que se trazan a lo largo del desarrollo del producto.

Una definición estándar de metodología puede ser el conjunto de métodos que se utilizan en una determinada actividad con el fin de formalizarla y optimizarla. Determina los pasos a seguir y cómo realizarlos para finalizar una tarea. Si esto se aplica a la ingeniería del software, se destaca que una metodología: (6)

- Optimiza el proceso y el producto software.
- Contiene métodos que guían en la planificación y en el desarrollo del software.
- Define qué hacer, cómo y cuándo durante todo el desarrollo y mantenimiento de un proyecto.

Solo se estudiarán las metodologías XP y RUP debido al poco tiempo que resta para el desarrollo de la solución. Lo anterior se sustenta además en el hecho de que “XP es una

metodología que resalta por contar con la mayor cantidad de información disponible y es con diferencia la más popular” (7) y en que RUP capta muchas de las mejores prácticas en el desarrollo de software moderno en una forma que es adecuada para una amplia gama de proyectos y organizaciones (8). Otro aspecto valorado es que son dos metodologías de las cuales la autora de la investigación tiene dominio.

Programación extrema

La programación extrema (XP por sus siglas en inglés) es una metodología ligera de desarrollo de software que se basa en la simplicidad, la comunicación y la realimentación o reutilización del código desarrollado.

XP se caracteriza por ser una metodología ágil, la cual ofrece al cliente el software que necesita y cuando lo necesita además de combinarse en ella las mejoras prácticas para el desarrollo de software. Es uno de los procesos ágiles más destacados, diferenciándose de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. Tiene como meta la disminución de costes y su objetivo principal es satisfacer a los clientes y potenciar al máximo el trabajo en grupo, además los jefes de proyecto, los desarrolladores y el cliente son parte del equipo de desarrollo. En el caso de los desarrolladores deben conocer que durante el ciclo de desarrollo de un proyecto de software los cambios aparecen, cambian los requisitos, las reglas del negocio, el personal y la tecnología, por lo tanto este no es el principal problema ya que va a suceder, en lo que reside el principal problema es en la capacidad que tengan para adaptarse a esos cambios. (9)

Promueve el trabajo en equipo, preocupándose en todo momento del aprendizaje de los desarrolladores y estableciendo un buen clima de trabajo. Este tipo de método se basa en una realimentación continuada entre el cliente y el equipo de desarrollo con una comunicación fluida entre todos los participantes, también busca simplificar las soluciones implementadas y coraje para los múltiples cambios. (10)

El Proceso Unificado de Rational

El Proceso Unificado Rational (RUP por sus siglas en inglés) es un marco de trabajo del proceso de desarrollo de software. RUP no es un proceso preceptivo concreto individual, sino un marco de trabajo de proceso adaptable, con la idea de ser adaptado por las organizaciones

de desarrollo y los equipos de proyecto de software que seleccionarán los elementos del proceso que sean apropiados para sus necesidades. Incluye una base de conocimiento con artefactos de ejemplo y descripciones detalladas para muchos tipos diferentes de actividades.
(6)

RUP se basa en un conjunto de módulos o elementos de contenido, que describen qué se va a producir, las habilidades necesarias requeridas y la explicación paso a paso describiendo cómo se consiguen los objetivos de desarrollo.

Los módulos principales, o elementos de contenido engloban las Fases del ciclo de vida del proyecto. RUP determina que el ciclo de vida del proyecto consiste en cuatro fases que permiten que el proceso sea presentado a alto nivel.

A continuación se presentan las fases en las que se divide el desarrollo del software y los objetivos que se trazan:

Fases	Objetivos
Inicio	Determinar la visión del proyecto.
Elaboración	Determinar la arquitectura óptima del proyecto.
Construcción	Obtener la capacidad operacional inicial.
Transición	Obtener el producto acabado y definido.

Además, se divide en nueve flujos de trabajo: Modelado del negocio, Requisitos, Análisis y diseño, Implementación, Pruebas, Despliegue, Gestión de cambio y configuraciones, Gestión del proyecto y Entorno, donde los seis primeros son básicos y los demás son de soporte. En la siguiente figura se observan las fases y el comportamiento de los flujos en las mismas:

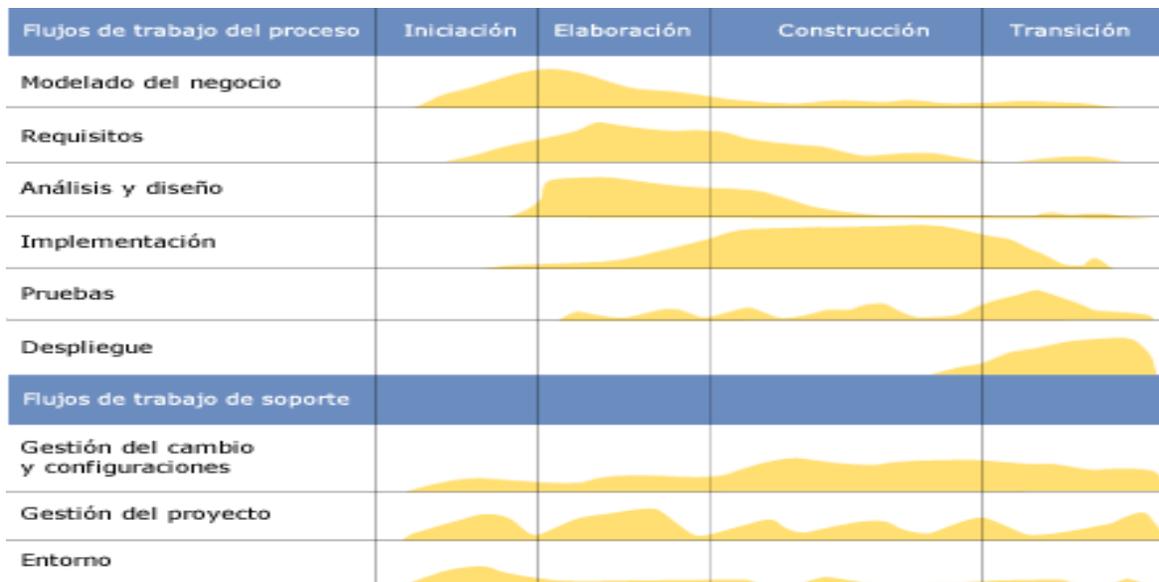


Figura 4: Fases y flujos de RUP

Selección de la metodología a utilizar

Una vez realizada la investigación de las metodologías, se puede apreciar que aun cuando estas se centran en la producción de software, poseen sus diferencias. XP tiene como ventaja que agiliza los procesos para el acabado de un proyecto a corto plazo, la misma no es de gran utilidad, pues se necesita una metodología que tenga criterios detallados y un mayor nivel de descripción de los procesos, además, la metodología XP propone por la dinámica del desarrollo un especialista del negocio trabajando junto con el cliente y en la presente solución el especialista realiza las entrevistas con el cliente a largo plazo. Se selecciona RUP como metodología para emplear en la investigación porque mantiene al equipo enfocado en producir producto funcional a tiempo, con las características y calidad requerida. Para la presente investigación permite obtener una amplia documentación del sistema a desarrollar a partir de un conjunto de actividades orientadas a visualizar, especificar, construir y documentar los artefactos necesarios para la producción del software con la calidad que el cliente necesita guiando así el proceso de desarrollo del producto. Teniendo en cuenta las características anteriores y además la experiencia de la autora en el uso de RUP se elige la misma.

1.5.2 Lenguaje de modelado

Los lenguajes de modelado establecen una guía importante para la representación de la estructura de los sistemas informáticos. Consiste en un conjunto de vistas, diagramas,

símbolos y reglas que indican cómo emplear los elementos.

El modelado sirve no solamente para los grandes sistemas, aun en aplicaciones de pequeño tamaño se obtienen beneficios de modelado, sin embargo es un hecho que entre más grande y más complejo es el sistema, más importante es el papel que juega el modelado por una simple razón: "El hombre hace modelos de sistemas complejos porque no puede entenderlos en su totalidad". (11)

UML 2.1

El Lenguaje Unificado de Modelado (UML⁴, por sus siglas en inglés) es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. (6)

UML está compuesto por diversos elementos gráficos que se combinan para conformar diagramas; es un lenguaje por lo que cuenta con reglas para combinar tales elementos. Es importante recalcar que UML no es una guía para realizar el análisis y diseño orientado a objetos, es decir, no es un proceso. UML es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos. (11)

Prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación. UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. (6)

Por las características antes explicadas se decide utilizar UML como soporte de la modelación de la solución propuesta dado que los diagramas que provee permiten documentar de manera completa el software a realizar.

1.5.3 Herramienta CASE para el modelado UML

Las herramientas de Ingeniería de Software Asistida por Computadora (en lo adelante CASE, por sus siglas en inglés, Computer Aided Software Engineering) son un conjunto de

⁴ *Unified Modeling Language*

programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un software. Como es sabido, los estados en el ciclo de vida de desarrollo de un software son: Investigación preliminar, Análisis, Diseño, Implementación e Instalación. (12)

Visual Paradigm 8.0

Visual Paradigm es una herramienta CASE que utiliza como lenguaje de modelado UML. Es compatible con una amplia gestión de casos de uso, diagramas de requerimiento y diseño de base de datos y proporciona medidas más eficaces en el análisis y diseño de sistemas. Entre los diagramas que genera se encuentran los de casos de usos, actividades, despliegue, de secuencia entre otros. Permite la generación automática de reportes en formato pdf y html, el reconocimiento de artefactos de ingeniería a partir de reconocimiento de textos formales o informales, implementa una actualización automática del modelo de diseño y código permitiendo mantener la documentación de ambos modelos actualizadas con los cambios que ocurran en ambos sentidos, optimizando la descripción textual de elementos de código a partir de la descripción visual. Es capaz de importar y exportar elementos de otras herramientas CASE como Rational Rose y presenta una interfaz de uso intuitiva y con increíbles facilidades a la hora de modelar los diagramas que soportan la Ingeniería de Requerimientos. (13)

El uso de Visual Paradigm es importante para la investigación dado que se destaca entre sus características la capacidad para incluir nuevos estereotipos que se pueden utilizar para la representación de los diagramas de clases utilizando estereotipos web como formularios html, páginas servidoras y páginas clientes, además en la versión 8.0 brinda la posibilidad de generar código php, integrarse con el entorno de desarrollo NetBeans así como el soporte a la versión 2.1 de UML. Otro factor de peso para la selección de esta herramienta lo constituye el dominio de la autora a partir de su utilización en investigaciones anteriores.

1.5.4 Tecnologías del lado del cliente

Ajax

Significa Asynchronous JavaScript and XML, que se puede traducir como “JavaScript asíncrono y XML”. En el artículo publicado por Jesse James Garrett en el 2005 realiza la definición de Ajax como: “Ajax no es una tecnología por si sola. En realidad, se trata de varias

tecnologías independientes que se unen de formas nuevas y sorprendentes”. Las tecnologías que forman Ajax son XHTML y CSS para crear una presentación basada en estándares, DOM para la interacción y manipulación dinámica de la presentación. XML, XSLT y JSON para el intercambio y la manipulación de información, XMLHttpRequest para el intercambio asíncrono de la información y JavaScript para unir todas las demás tecnologías. (14)

Lenguajes del lado del cliente

HTML⁵: Es un estándar para HTML. Navegadores como Safari, Chrome, Firefox, Opera, Internet Explorer son compatibles con muchos de los nuevos elementos de HTML5.

Algunas de las nuevas características son:

- Nuevos elementos de contenido específico, como article, footer, header, nav y section.
- El elemento canvas para el dibujo 2D.
- Soporte para almacenamiento local.
- Nuevos controles de formulario, como el calendario, la fecha, la hora, email, URL y búsqueda.
- Se reduce la necesidad de plugins externos.
- Mejora el manejo de errores.

CSS³: Creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación, y es imprescindible para crear páginas web complejas. Mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes. (15)

CSS3 se divide en varios documentos separados llamados módulos. Cada módulo añade nuevas capacidades o amplía funciones definidas en CSS2, sobre la preservación de compatibilidad con versiones anteriores. El trabajo en CSS nivel 3 comenzó alrededor del momento de la publicación del original de la recomendación de CSS2. Debido a la estructura en módulos, se tiene diferente estabilidad y estados. (16)

⁵ *eXtensible HyperText Markup Language en español lenguaje extensible de marcado de hipertexto*

⁶ *Cascading Style Sheets en español hojas de estilo en cascada*

JavaScript 1.8.5: Es un lenguaje de programación del lado del cliente que se utiliza principalmente para aportar dinamismo a la interfaz. Una interfaz dinámica es aquella que incorpora efectos, como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario. Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. JavaScript es una marca registrada de la empresa Sun Microsystems. (17)

Las tecnologías del lado del cliente ayudan a mejorar considerablemente la interacción entre el usuario y el sistema dado que permite: visualizar la información y los componentes de entrada de datos con XHTML y XML, mejorar mediante CSS los estilos de las plantillas visibles para el usuario y realizar funcionalidades dinámicas con el uso de JavaScript y Ajax.

Lenguajes del lado del servidor

PHP ⁷5.3.8: Es un lenguaje interpretado de alto nivel embebido en páginas HTML. Su interpretación y ejecución ocurren en el servidor web, en el cual se encuentra almacenado el script, el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente. (18)

Brinda soporte a una gran cantidad de bases de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, Sybase mSQL, Informix, entre otras y se integra con varias librerías externas, lo que extiende aún más su capacidad. (19) Esta versión corrige los problemas identificados en la versión anterior.

Twig: Es un motor y lenguaje de plantillas para PHP muy rápido y eficiente. Symfony2 recomienda utilizar Twig para crear todas las plantillas de la aplicación. La sintaxis de Twig se ha diseñado para que las plantillas sean concisas y muy fáciles de leer y de escribir. (20)

Entre sus características se encuentran:

⁷ *Hypertext Pre-Processor*

- Rápido: compila plantillas a código PHP simple y optimizado. La sobrecarga en comparación con el habitual código PHP se ha reducido a la mínima expresión.
- Debido a su arquitectura abierta, se pueden implementar sus propias construcciones del lenguaje (etiquetas, filtros, funciones e incluso los operadores).
- Ante un problema de sintaxis en una plantilla, Twig emite un mensaje de ayuda con el nombre de archivo y el número de línea donde se produjo el problema. Facilita mucho la depuración. (21)

1.5.5 Capa de acceso a datos

Doctrine2

Doctrine2 es una herramienta para el mapeo objeto relacional (ORM por su siglas en inglés) para PHP 5.3.0+ que proporciona persistencia transparente de objetos PHP. Se sitúa en la parte superior de una poderosa capa de abstracción de base de datos (DBAL⁸ por sus siglas en inglés). La principal tarea de los ORM es la traducción transparente entre objetos (PHP) y las filas relacionales de la base de datos.

Una de las características clave de Doctrine es la opción de escribir las consultas de base de datos en un dialecto SQL propio orientado a objetos llamado Lenguaje de Consulta Doctrine (DQL⁹ por sus siglas en inglés), inspirado en Hibernate HQL. Además DQL difiere ligeramente de SQL en que abstrae considerablemente la asignación entre las filas de la base de datos y objetos, permitiendo a los desarrolladores escribir poderosas consultas de una manera sencilla y flexible. (22)

1.5.6 Sistemas gestores de bases de datos

Para la investigación se hace necesaria la selección de un sistema gestor de base de datos y por ello se realiza el estudio de MySQL en su versión 5.5 y PostgreSQL en su versión 9.1 para luego definir cuál es el mejor para el desarrollo de la solución.

⁸ *DataBase Abstraction Layer*

⁹ *Doctrine Query Language*

MySQL 5.5

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL (del inglés General Public License) de la GNU¹⁰. Su diseño multihilo le permite soportar una gran carga de forma eficiente. Este gestor de bases de datos es uno de los más usados en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación se debe a que existen infinidad de librerías y otras herramientas que permiten su uso a través de un gran número de lenguajes de programación, además de su fácil instalación y configuración.

(23)

PostgreSQL 9.1

Es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD (del inglés Berkeley Software Distribution) y con su código fuente disponible libremente.

PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.

Su desarrollo comenzó hace más de 16 años, y durante este tiempo, estabilidad, potencia, robustez, facilidad de administración e implementación de estándares han sido las características que más se han tenido en cuenta durante su desarrollo. PostgreSQL funciona muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema. (24)

Entre sus características se encuentran también:

- El soporte de distintos tipos de datos: datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP), cadenas de bits entre otros. También permite la creación de tipos propios.
- La gestión de diferentes usuarios y los roles que juegan, así como los permisos que se les asignan.
- Incorpora una estructura de datos de arreglos.
- Es soportado por el lenguaje PHP.

¹⁰ GNU es un acrónimo recursivo para "Gnu No es Unix"

Selección del gestor a utilizar

A partir del estudio realizado sobre los sistemas gestores de bases de datos se decide utilizar PostgreSQL 9.1 porque es de código abierto y gratis además está distribuido bajo la licencia BSD la cual brinda ciertas ventajas, entre ellas hacer software propietario, lo que no permite MySQL 5.5. También cuenta con una activa comunidad de usuarios que están en constante intercambio de conocimientos lo cual es ventajoso si ocurre algún tipo de error. Es un gestor que cuenta con más de 15 años de desarrollo lo que permite que sea confiable.

1.5.7 Servidor web Apache 2.2.21

Apache es el servidor web más popular y utilizado a nivel mundial, multiplataforma, extensible y de código abierto, puede soportar un sin número de sistemas operativos. Sus últimas versiones le permiten ejecutarse en modo híbrido multiproceso - multihebra, lo cual mejora la escalabilidad para la mayoría de las configuraciones, posee la infraestructura necesaria para servir distintos protocolos, es muy rápido y estable en sistemas que no son tipo Unix, tales como BeOS, OS/2 y Windows. (25)

A continuación se observan algunas características específicas de la versión a utilizar:

- Los módulos de autenticación y autorización empaquetados han sido rediseñados.
- Los módulos de memoria caché han sufrido muchos cambios, y ahora son considerados con producción calidad.
- El diseño de la configuración por defecto se ha simplificado y estructurado en módulos. Fragmentos de configuración que se pueden utilizar para activar las funciones de uso común están incluidos con Apache, y pueden ser fácilmente añadidos a la configuración del servidor principal. (26)

1.5.8 Framework

A continuación se realiza el estudio de los framework Zend y Symfony donde se efectúa la selección del indicado para desarrollar la solución.

Zend

Es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP5. La estructura de los componentes de Zend es única; cada uno de ellos está construido con

una baja dependencia de los otros. Esta arquitectura débilmente acoplada permite a los desarrolladores utilizar los componentes por separado. A menudo se refiere a este tipo de diseño como uso a voluntad (use-at-will).

Aunque se pueden utilizar de forma individual, los componentes de la biblioteca estándar de Zend conforman un potente y extensible framework de aplicaciones web al combinarse. Ofrece también un gran rendimiento y una robusta implementación Modelo Vista Controlador, y una abstracción de base de datos fácil de usar. (27)

Zend cuenta con una comunidad que está disponible en correos y foros, todo el tiempo a disposición de los usuarios.

Symfony2

Es un completo framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. (28)

Está desarrollado completamente con PHP 5. Es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. Se puede ejecutar tanto en plataformas *nix (Unix, Linux) como en plataformas Windows.

Es fácil de instalar, configurar y de usar en la mayoría de los casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos. (29)

Symfony2 ha sido ideado para exprimir al límite todas las nuevas características de PHP 5.3 y por eso es uno de los frameworks PHP con mejor rendimiento. Su arquitectura interna está completamente desacoplada, lo que permite reemplazar o eliminar fácilmente aquellas partes que no encajan en un proyecto. (20)

Selección del framework a utilizar

Luego de realizado el estudio de estos dos framework para el desarrollo de aplicaciones web, se propone Symfony2 para realizar la implementación de la propuesta de solución, debido a

los siguientes elementos:

Documentación y aprendizaje: Symfony dispone de guías y manuales en abundancia, además cuenta con numerosos foros en varios idiomas donde la comunidad va resolviendo dudas. Entre los logros alcanzados por esta comunidad se encuentran el desarrollo de muchos plugins o extensiones que facilitan el trabajo con tareas comunes. En el caso de Zend, a pesar de ser de la empresa que está detrás de PHP, la comunidad es algo escasa, por tanto su documentación también.

Algunos usuarios de internet en foros consultados refieren que Symfony es mejor comparado con Zend para programadores de PHP con poca o ninguna experiencia en la arquitectura Modelo Vista Controlador.

Bases de datos: El uso de Doctrine como ORM listo para usar al instalar Symfony facilita el trabajo al desarrollador.

Facilidades de NetBeans: Como se explicará más adelante NetBeans a partir de su versión 7.2 permite ejecutar comandos de Symfony de forma sencilla y rápida.

1.5.9 Entorno de desarrollo integrado

Eclipse 4.2

Es un software de código abierto multiplataforma para desarrollar aplicaciones. Tiene como principales características:

- La disponibilidad de un editor de texto con resaltado de sintaxis.
- Las compilaciones las realiza en tiempo real.
- Las pruebas unitarias son con JUnit.
- Integración con sistemas de control de versiones CVS ¹¹ a través de los plugins añade con Subversión.
- Integración con Ant¹². (30)

¹¹ *Concurrent Versions System*

¹² *Apache Ant: es una herramienta usada para la realización de tareas mecánicas y repetitivas, durante la fase de compilación y construcción.*

Para el uso con Symfony2 en Eclipse es necesario descargar el plugin creado por Robert Gründler. Las ventajas que brinda este son:

- Asistente de código para elementos específicos de Symfony2, como los servicios, rutas, entidades, traducciones y bloques de Twig.
- Soporte para anotaciones.
- Soporte para Twig. (31)

NetBeans 7.2

Es un entorno de desarrollo integrado (IDE por sus siglas en inglés), herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Es multiplataforma, libre, gratuito y sin restricciones de uso. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el NetBeans IDE. (32)

Principales características:

- Incluye un editor de código fuente.
- Admite la integración con JUnit.
- Admite la integración con sistemas de control de versiones.
- Permite la colaboración entre varios usuarios.
- Permite el resaltado de sintaxis.
- Sus funcionalidades se pueden ampliar mediante la instalación de paquetes.

Para el uso con Symfony2 dispone de una serie de comandos que se pueden ejecutar desde el mismo IDE sin necesidad de instalar extensiones adicionales.

Selección del IDE a utilizar

Se decide utilizar NetBeans 7.2 para el desarrollo del sistema, ya que es libre, multiplataforma y soporta los lenguajes a utilizar para la implementación de la solución. Este IDE tiene la ventaja de permitir la ejecución de comandos de Symfony2 mientras que en Eclipse es necesario instalar un plugin. Además la experiencia de la autora en el uso de esta herramienta

fue otro factor relevante para la selección de la misma.

1.5.10 Bootstrap 2.3

Bootstrap es un conjunto de herramientas front-end¹³ para el rápido desarrollo de aplicaciones web. Se trata de una colección de las convenciones de estilo CSS y HTML. Utiliza algunas de las técnicas más avanzadas del navegador para ofrecerle tipografía elegante, formas, botones, cuadros, redes, navegación y todo lo que necesita en un recurso pequeño.

En su esencia, Bootstrap es sólo CSS, pero está construido con LESS, un pre-procesador flexible que ofrece mucha más potencia y flexibilidad que CSS regular. Con LESS, gana una serie de características como las declaraciones anidadas, variables y funciones de color. Además, como Bootstrap es puramente CSS cuando se compila a través de LESS, se obtienen dos beneficios importantes: (33)

- Sigue siendo fácil de implementar, simplemente colocarlo en el código es suficiente.
- Una vez compilado, Bootstrap solo contiene CSS, lo que significa que no hay imágenes superfluas, Flash o Javascript. Todo lo que queda es CSS simple y de gran alcance para las necesidades de desarrollo web.

Los componentes que utiliza se adaptan y escalan según las resoluciones de pantalla y dispositivos para proporcionar una experiencia consistente, no importa el qué. En principio fue creado solo teniendo los navegadores modernos en mente, pero ha evolucionado para incluir soporte para un mayor número de navegadores (incluso IE7), y, con Bootstrap 2, también para tabletas y smartphones. (34)

Las características anteriores permitirán que el sistema de gestión a implementar trabaje rápido, debido a la buena estructura de clases que este posee. También facilita que en un futuro si se logra un desarrollo en hardware (dígase tabletas) en las instituciones donde se emplee el sistema este estará listo para usarse dado que se podrá adaptar a las dimensiones del dispositivo, permitiendo que los estudios que se realicen con los datos de las tortugas marinas no solo se gestionen bien, al mismo tiempo se realicen rápidos.

¹³ Traducido: de aplicaciones para usuario

Conclusiones del capítulo

El estudio realizado apoyado en los métodos de la investigación científica especificados anteriormente, permitió elaborar el marco teórico conceptual que soporta la investigación, siendo así definidas las características generales de los sistemas de gestión de información de datos medioambientales que se tendrán en cuenta para el desarrollo de la propuesta de solución, la cual se desarrollará utilizando la metodología RUP para guiar el proceso de desarrollo; el lenguaje UML 2.1 para el modelado de la solución mediante Visual Paradigm en su versión 8.0; para la implementación del sistema el uso de las tecnologías del lado del cliente y del lado del servidor previamente definidas, utilizando la herramienta NetBeans 7.2.

Capítulo 2: Propuesta de solución

Introducción

La metodología RUP establece en cada una de sus fases una serie de artefactos que son importantes para la comprensión del sistema a realizar, entre ellos se encuentran: el Modelo de dominio donde se explican los principales conceptos asociados al problema que posteriormente se reflejan en un diagrama; el Modelo de casos de uso en el cual se realizan los diagramas y especificación de casos de uso y el Modelo de análisis donde se realizan los diagramas de interacción: colaboración y secuencia, y los prototipos de interfaz de usuario. Todos estos artefactos facilitan la comprensión de lo que se quiere concebir para lograr así una correcta solución.

2.1 Descripción del negocio

Las áreas protegidas del archipiélago cubano abarcan diferentes playas, las cuales constituyen el lugar de anidamiento de algunas especies de tortugas marinas entre ellas la *Eretmochelys imbricata* (Carey), *Chelonia mydas* (Tortuga Verde), *Caretta caretta* (Caguama), *Dermochelys coriacea* (Tinglado) y la *Lepidochelys olivacea* (Golfina). Los recorridos que se realizan en las playas tienen como objetivo:

- Registrar las tortugas de las cuales obtienen datos como: El nombre de especie, hora de descenso, la acción que realiza, si tiene Fibropapiloma, el código de la marca, sus medidas: largo curvo, ancho curvo, ancho recto, largo recto y una imagen de la tortuga.
- Reconocer los rastros que dejan las mismas y registran: su ancho, tipo, condición, dirección, localización, número y a que especie pertenece.
- Examinar los nidos, de los cuales guardan datos como: su número, la fecha de puesta, los intentos de la tortuga en hacer nidos, la distancia de la línea de marea, su profundidad, cantidad de huevos, el porcentaje de banda opaca, cuántos huevos rotos, el tiempo de incubación, la zona donde está situado, la especie, la condición, su localización y si fue depredado, por qué animal.

- Realizar el análisis de los huevos que se encuentren en los nidos y registrar datos como: número del nido al que pertenece, total de cascarones, total de huevos con y sin desarrollo embrionario, total de huevos, periodo de incubación, total de embriones con anomalías, el total de huevos depredados por hormigas, cangrejos, larvas y hongos.
- Realizar el análisis de los neonatos y registrar datos como: el número de neonatos y de estos el total de atrasados, muertos, normales, con restos de vitelo y con larvas, el ancho recto promedio, el largo recto promedio, profundidad y las observaciones.

Es importante esclarecer que estos registros son realizados por especialistas.

2.2 Modelo de dominio

El modelo de dominio está en un nivel alto de abstracción y no debe confundirse con los modelos de base de datos o esquemas, que estarán a un nivel mayor de detalle. El modelo de dominio no se refiere a la forma en que los datos y la información son físicamente guardados o procesados. Puede incluir conceptos y datos que actualmente no están informatizados o almacenados electrónicamente. Se incluyen otros datos que pueden residir en una variedad de bases de datos físicas o sistemas de archivos bajo el control de diversos sistemas de aplicación, posiblemente distribuidos a través de varios entornos técnicos o ubicaciones geográficas. Puede incluir conceptos que no están actualmente de forma explícita, por ejemplo, datos acerca de las condiciones del mercado y competidores. (35)

El modelo de dominio tiene como objetivo ayudar a comprender los conceptos que manejan los actores, así como los procesos que se desean entender y desarrollar, por ello se hace necesaria su confección.

2.2.1 Conceptos del dominio

Áreas protegidas: Zonas para la conservación de la biodiversidad así como el mantenimiento de los procesos ecológicos necesarios para su preservación.

Playas: Ribera del mar o de un río grande, formada de arenales en superficie casi plana. Lugar donde anidan las tortugas y depositan sus huevos.

Especialista: Persona que se encarga de realizar los estudios, la toma de datos y análisis de la información.

Administrador: Persona que se encarga de registrar las áreas protegidas, playas y los usuarios.

Recorrido: Observación que se realiza por un grupo de especialistas en un espacio determinado.

Tortugas marinas: Reptil marino del orden de los Quelonios, que llega a tener hasta dos metros y medio de largo y uno de ancho.

Rastro: Huella que dejan las tortugas al desplazarse sobre la arena de la playa.

Nido: Lecho en forma de cavidad o agujero donde las tortugas depositan sus huevos.

Análisis de huevos: Estudio que se realiza a los huevos de tortuga en el nido para determinar las condiciones en las que se encuentran estos.

Análisis de neonatos: Estudio que se realiza a los neonatos de tortuga para determinar las condiciones con que los recién nacidos emergen.

2.2.2 Diagrama del Modelo de dominio

El siguiente diagrama muestra las relaciones existentes entre los conceptos establecidos anteriormente:

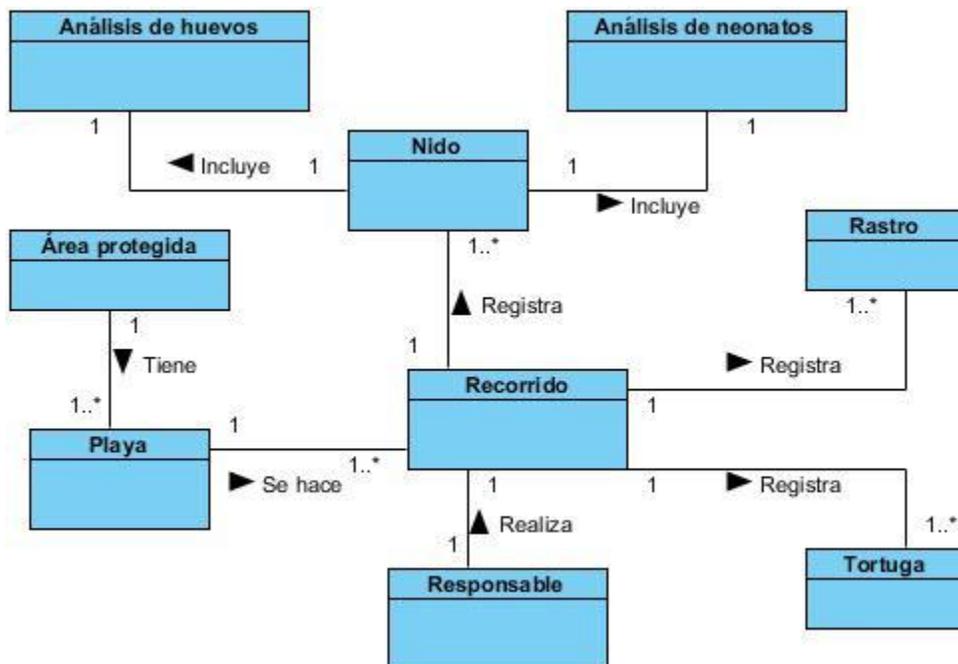


Figura 5: Modelo de dominio

2.3 Descripción del sistema propuesto

El sistema propuesto tiene como objetivo registrar y organizar toda la información referente a los estudios que se realizan en las playas de anidamiento de tortugas. Este incluirá tres módulos los cuales tendrán los nombres Administración, Campaña y Reportes a continuación se observan las funcionalidades propuestas:

Usuarios: En esta sección se mostrará toda la información referente a los usuarios registrados en el sistema. Entre las funcionalidades a realizar se encuentran la visualización de todos los usuarios, la consulta, la creación y actualización de la información de los mismos.

Áreas protegidas: En esta sección aparecerá la relación de todas las áreas protegidas existentes en el sistema. Se podrán realizar consultas así como la gestión de la información que contiene cada una.

Playas: En esta se representarán las playas con una descripción y a qué área protegida pertenece cada una. Se podrán realizar consultas así como la gestión de la información que contiene cada una.

Recorridos: En los recorridos se mostrará la fecha en que se realizan, su responsable, la playa en que se efectúan entre otros datos, los mismos se podrán crear, modificar, visualizar o eliminar en cualquier momento. De igual forma pero con atributos diferentes se gestionan los datos de las secciones Rastros y Tortugas.

Nidos de tortugas: En esta sección aparecerá la relación de todos los nidos encontrados en los recorridos que se realicen, a cada nido se le podrá realizar un análisis de huevos y un análisis de neonatos, estos datos se mostrarán, se podrán modificar y solo serán eliminados si es eliminado el nido de tortugas.

Reportes: En los reportes se mostrarán estadísticas utilizando:

- Un gráfico de pastel donde se representa el porcentaje de la cantidad de tortugas por especie.
- Un gráfico de barras donde se observa la cantidad de tortugas que existen por especies y dentro de cada barra se mostrará otro gráfico indicando el promedio de las dimensiones de las tortugas de esa especie.

- Un gráfico de pastel donde se refleja el porcentaje de ejemplares enfermos de Fibropapiloma.
- Un resumen de la cantidad de áreas protegidas, playa, recorridos, tortugas, nidos, rastros y usuarios creados en el sistema. Este muestra además, la cantidad de casos por cada anomalía y los datos de los últimos recorridos creados.

2.4 Requerimientos funcionales

Los requerimientos funcionales siguientes definen las capacidades o condiciones que el sistema debe cumplir:

RF1 Gestionar usuarios:

RF1.1 Crear nuevo usuario

RF1.2 Ver datos de un usuario

RF1.3 Modificar los datos del usuario

RF1.4 Eliminar usuario

RF2 Consultar datos de los usuarios

RF3 Autenticar usuario

RF4 Gestionar área protegida:

RF4.1 Crear área protegida

RF4.2 Ver datos del área protegida

RF4.3 Modificar el área protegida

RF4.4 Eliminar área protegida

RF5 Buscar los datos de un área protegida siguiendo un criterio de búsqueda

RF6 Gestionar información de la playa:

RF6.1 Crear una nueva playa

RF6.2 Ver datos de la playa

RF6.3 Modificar los datos de la playa

RF6.4 Eliminar la playa

RF7 Realizar búsquedas de las playas para la consulta de sus datos

RF8 Gestionar recorrido:

RF8.1 Crear recorrido

RF8.2 Ver datos del recorrido

RF8.3 Modificar recorrido

RF8.4 Eliminar recorrido

RF9 Gestionar tortuga:

RF9.1 Crear tortuga

RF9.2 Ver datos de la tortuga

RF9.3 Modificar tortuga

RF9.4 Eliminar tortuga

RF10 Buscar tortugas siguiendo un criterio de búsqueda

RF11 Gestionar nido:

RF11.1 Crear nido

RF11.2 Ver datos del nido

RF11.3 Modificar nido

RF11.4 Eliminar nido

RF12 Gestionar análisis de neonatos:

RF12.1 Crear nuevo análisis de neonatos

RF12.2 Ver análisis de neonatos

RF12.3 Modificar análisis de neonatos

RF12.4 Eliminar análisis de neonatos

RF13 Gestionar análisis de huevos:

RF13.1 Crear análisis de huevos

RF13.2 Ver análisis de huevos

RF13.3 Modificar análisis de huevos

RF13.4 Eliminar análisis de huevos

RF14 Gestionar rastro:

RF14.1 Crear nuevo rastro

RF14.2 Ver datos del rastro

RF14.3 Modificar rastro

RF14.4 Eliminar rastro

RF15 Mostrar reportes

RF16 Cambiar contraseña

2.5 Requerimientos no funcionales

Los requerimientos no funcionales siguientes definen las propiedades o cualidades que complementan el funcionamiento del sistema:

Apariencia o Interfaz Externa

- La información aparecerá correctamente organizada de forma, que el usuario pueda navegar sin ningún problema.
- Debe notificar al usuario ante la presencia de un error.
- Los mensajes que se muestren deben seguir un mismo formato.
- El menú debe estar en todas las páginas.
- El diseño y los colores deben estar acorde al tema de la investigación, colores claros y diseño sencillo.
- El nombre del usuario debe reflejarse en algún lugar de la interfaz.

Software

- Las computadoras cliente deben tener instalado el navegador web Internet Explorer 7.0 o superior o Mozilla Firefox 3.6 o superior o Chrome 10 o superior.
- El servidor debe tener instalado PostgreSQL 9.2 y Apache 2.2.21.

Usabilidad

- Debe ser fácil de usar para usuarios con conocimientos básicos en el uso de computadoras, de esta forma se garantiza que los mismos especialistas que realizan el estudio puedan digitalizar los datos.

Seguridad

- El acceso a la información deberá estar restringido por usuario y contraseña.
- A los usuarios autorizados se les debe garantizar el acceso a la información.
- Las contraseñas se guardarán encriptadas en la base de datos garantizando mayor seguridad.

Hardware

- Para un buen funcionamiento el servidor donde se ejecute el sistema debe contar con las siguientes características:
 - ✓ Procesador Core i3.
 - ✓ Memoria RAM de 4Gb.
- Para un buen funcionamiento la computadora donde se instale el sistema debe contar con las siguientes características:
 - ✓ Procesador Pentium 4 o superior.
 - ✓ Memoria RAM de 512 MB o superior.

Portabilidad

- El usuario debe acceder a la aplicación desde cualquier sistema operativo.

Rendimiento

- Los tiempos de respuesta no deben tardarse por encima de los 6 segundos.

Accesibilidad

- Debe estar accesible desde cualquier lugar de la red.
- La información actualizada debe estar accesible para la gestión de los datos.

2.6 Patrones de caso de uso

Utilizar patrones de caso de uso permite reflejar con más claridad los requisitos reales de la investigación. A continuación se presentan algunos utilizados en el diagrama de casos de uso del sistema:

- **Concordancia:** Se encarga de extraer una subsecuencia de acciones que aparecen en diferentes partes del flujo de casos de uso y es expresado por separado. Estos también se conocen como casos de usos extendidos o incluidos, un ejemplo lo constituye el caso de uso incluido Gestionar análisis de neonatos.
- **CRUD¹⁴:** Este patrón se basa en la fusión de casos de uso simples para formar una unidad conceptual. Se puede observar el uso de este patrón en el caso de uso Gestionar playa el cual está compuesto por los casos de uso simples Crear, Ver, Modificar y Eliminar playa.
- **Múltiples actores:** Este patrón está dado por dos o más actores que representen el mismo rol sobre un caso de uso. En ese caso se crea otro actor, heredado por los actores que comparten ese rol. Este patrón se puede observar en la Figura 6 donde los actores Invitado y Especialista son especializaciones del actor Usuario.

2.7 Modelo de casos de uso del sistema

El modelo de caso de uso proporciona información detallada acerca de los comportamientos del sistema o aplicación que se está desarrollando. Contiene diagramas de casos de uso y diagramas de actividades que describen cómo los usuarios interactúan con el sistema. El modelo de casos de uso identifica los requisitos del sistema en términos de la funcionalidad que debe existir para lograr los objetivos establecidos por el usuario o para resolver un problema identificado por el usuario. (36)

¹⁴ *Creating, Reading, Updating, Deleting (en español Crear, Ver, Modificar, Eliminar)*

La confección del modelo de casos de uso constituye la guía del proceso de desarrollo de software y proporciona una visión del sistema a desarrollar.

2.7.1 Descripción de los actores del sistema

Tabla 1: Actores del sistema y su descripción

Actor	Descripción
Administrador	Actor que podrá crear otros usuarios y poseerá acceso completo a todos los módulos del sistema, para ver, crear, modificar y eliminar la información que disponga.
Especialista	Actor que podrá acceder y gestionar la información que se encuentra en los módulos de Campaña y Reportes. Además de poder consultar la información en las secciones Áreas protegidas y Playas.
Usuario	Actor que podrá acceder solo al módulo de Reportes.

2.7.2 Diagrama de casos de uso del sistema

El diagrama siguiente muestra la relación que existe entre los casos de uso y el usuario que los inicializa:

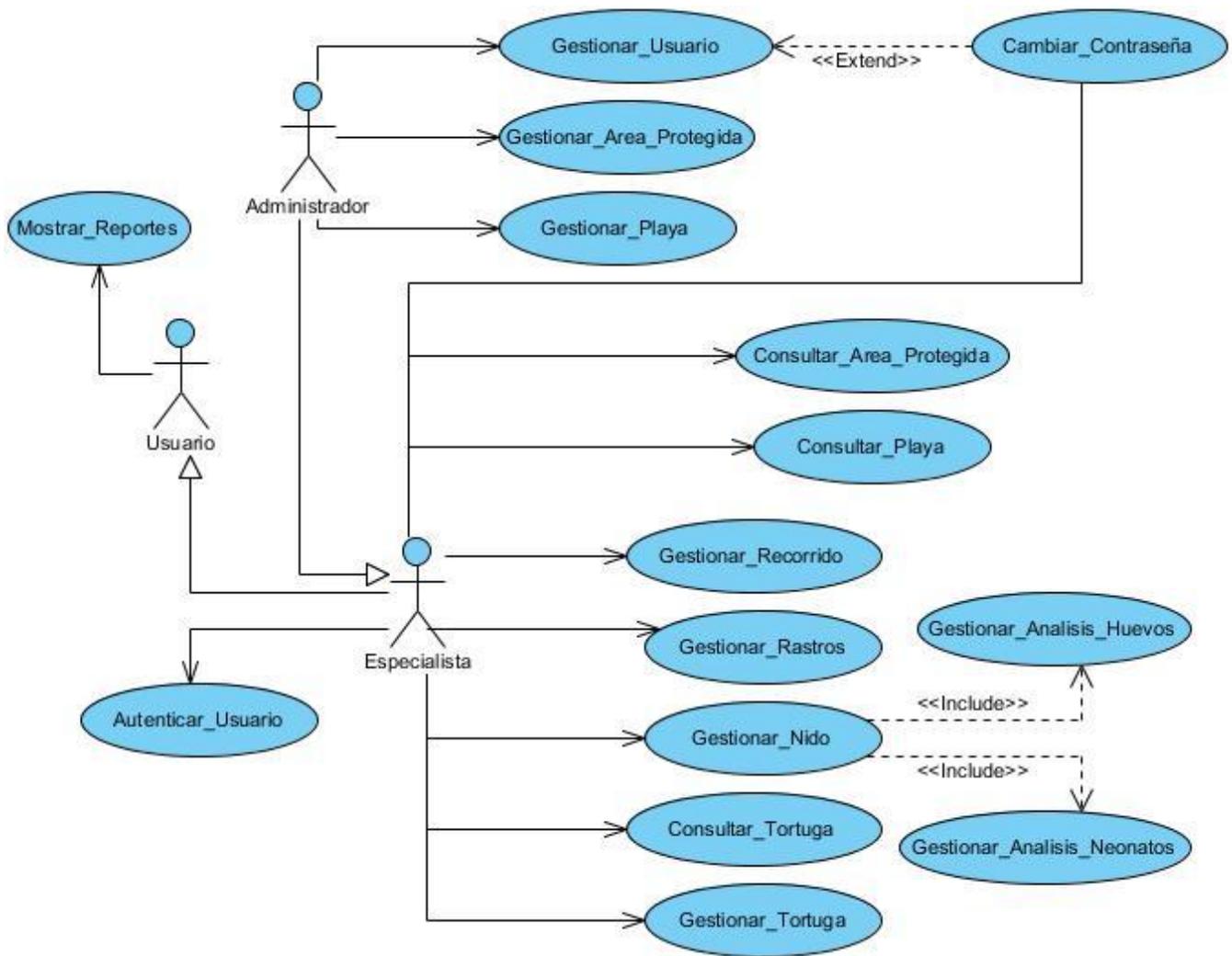


Figura 6: Diagrama de casos de uso del sistema

2.7.3 Descripción de los casos de uso del sistema

A continuación se muestra la descripción del caso de uso Gestionar recorrido, las restantes descripciones se encuentran en el Anexo 1.

Tabla 2. Descripción del CU Gestionar recorrido

Caso de uso	Gestionar recorrido
Actores	Especialista (inicia)
Resumen	El caso de uso inicia cuando el especialista decide realizar una acción para crear, modificar, ver o eliminar los datos sobre un recorrido. Si el especialista selecciona la opción de incluir un recorrido el sistema le

	<p>brindará la posibilidad de llenar los datos necesarios y le permitirá cancelar la operación o crearlo. En caso de que desee modificar los datos del recorrido, el actor deberá seleccionar el elemento a modificar, el sistema brindará la posibilidad de realizar modificaciones a los campos existentes permitiendo guardar los cambios realizados. Para eliminar un recorrido el sistema permitirá seleccionar y eliminará el mismo, terminando el caso de uso.</p>
<p>Precondiciones</p>	<ul style="list-style-type: none"> - El usuario ha autenticado. - Para modificar los datos de un recorrido: el recorrido ha sido seleccionado previamente por el usuario. - Para eliminar un recorrido: el recorrido ha sido mostrado previamente por el usuario.
<p>Flujo de eventos Flujo básico</p>	
<p>1. Accede a la sección de Recorrido.</p>	
	<p>2. Permite realizar varias acciones con un recorrido:</p> <ul style="list-style-type: none"> - Crear un nuevo recorrido. - Modificar los datos de un recorrido. Ver Sección 1: “Modificar recorrido”. - Ver los datos del recorrido. Ver Sección 2: “Ver recorrido”. - Eliminar un recorrido. Ver Sección 3: “Eliminar recorrido”.
<p>3. Selecciona la opción crear un nuevo recorrido.</p>	
	<p>4. Brinda la posibilidad de introducir los datos siguientes:</p> <ul style="list-style-type: none"> • Área protegida • Playa

	<ul style="list-style-type: none"> • Fecha • Hora inicio • Hora fin • Distancia • Participantes • Responsables • Fase lunar • Marea • Tiempo • Mar <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar los datos. • Cancelar la operación
5. Introduce los datos y selecciona la opción de guardar los datos del recorrido.	
	6. Valida los datos.
	7. Registra los datos del recorrido.
	8. El caso de uso termina.
Flujos alternos	
*a El actor selecciona la opción de cancelar	
Actor	Sistema
	*.a.1 Regresa al listado de recorridos.
	*.a.2 Muestra el mensaje de información “La operación se ha cancelado”.
	*.a.3 El caso de uso termina.
6.a Existen datos incompletos	
Actor	Sistema
	7. a.1 Muestra el mensaje de información “No

	puede dejar campos vacíos”.
	7. a.2 Muestra un indicador sobre los campos vacíos.
	7. a.3 Regresa al paso 4 del flujo básico.
6.a Existen datos incorrectos	
Actor	Sistema
	7. a.1 Muestra el mensaje de información “Existen datos incorrectos”.
	7. a.2 Muestra un indicador sobre los campos incorrectos.
	7. a.3 Regresa al paso 4 del flujo básico.
Sección 1: “Modificar recorrido”	
Flujo básico	
Actor	Sistema
1. Selecciona la opción modificar los datos de un recorrido.	
	<p>2. Muestra el recorrido permitiendo modificar los valores del mismo:</p> <ul style="list-style-type: none"> • Área protegida • Playa • Fecha • Hora inicio • Hora fin • Distancia • Participantes • Responsables • Fase lunar • Marea • Tiempo

	<ul style="list-style-type: none"> • Mar <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar los datos • Cancelar la operación
3. Modifica los datos que necesite y selecciona la opción de guardar los mismos.	
	4. Valida los datos.
	5. Actualiza los datos del recorrido.
	6. El caso de uso termina.
Flujos alternos	
*.a El actor selecciona la opción de cancelar	
Actor	Sistema
	*.a.1 Regresa al listado de recorridos.
	*.a.2 Muestra el mensaje de información “La operación se ha cancelado”.
	*.a.3 El caso de uso termina.
4.a Existen datos incompletos	
Actor	Sistema
	5. a.1 Muestra el mensaje de información “No puede dejar campos vacíos”.
	5. a.2 Muestra un indicador sobre los campos vacíos.
	5. a.3 Regresa al paso 4 del flujo básico.
4.a Existen datos incorrectos	
Actor	Sistema
	5. a.1 Muestra el mensaje de información “Existen datos incorrectos”.
	5. a.2 Muestra un indicador sobre los campos

	incorrectos.
	5. a.3 Regresa al paso 4 del flujo básico.
Sección 1: “Ver recorrido”	
Flujo básico	
Actor	Sistema
1. Selecciona la opción ver los datos de un recorrido.	
	<p>2. Muestra los datos del recorrido seleccionado:</p> <ul style="list-style-type: none"> • Área protegida • Playa • Fecha • Hora inicio • Hora fin • Distancia • Participantes • Responsables • Fase lunar • Marea • Tiempo • Mar <p>Y permite:</p> <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar el recorrido
	3. El caso de uso termina
Sección 1: “Eliminar recorrido”	
Flujo básico	
Actor	Sistema
1. Selecciona la opción eliminar un	

recorrido.	
	<p>2. Muestra un mensaje de alerta.</p> <p>Y permite:</p> <ul style="list-style-type: none"> • Aceptar • Cancelar
3. Selecciona la opción de aceptar.	
	4. Elimina el recorrido.
	5. Muestra un mensaje de información.
	6. El caso de uso termina.
Flujos alternos	
2.a El actor selecciona la opción de cancelar	
Actor	Sistema
	2. a.1 Regresa a la vista anterior.
	2. a.2 El caso de uso termina.

Conclusiones del capítulo

El estudio realizado de los conceptos asociados al negocio permitió comenzar a desarrollar la propuesta de solución del sistema. La confección del modelo de dominio tributó a la comprensión del entorno en que se relacionan las funcionalidades. Los requisitos funcionales y no funcionales ayudaron a definir las capacidades y cualidades que debe tener el sistema. Mediante la realización del diagrama de casos de usos del sistema se establecieron las relaciones entre los actores y los casos de uso. Los artefactos generados: el modelo de dominio, la especificación de casos de uso y otros mencionados anteriormente servirán de punto de partida para el posterior análisis y diseño.

Capítulo 3: Análisis y diseño

Introducción

Análisis y diseño es el tercer flujo que propone la metodología de desarrollo RUP. En el caso específico de la presente investigación solo se desarrollarán las actividades del diseño debido a que se tienen claros los requerimientos, generándose los artefactos correspondientes: diagramas de secuencia y de clase, seleccionando de estas las clases persistentes para obtener el modelo de la base de datos. Lo anterior se basa en patrones que ayudan a mejorar el diseño.

3.1 Patrón arquitectónico Modelo- Vista- Controlador en Symfony2

Modelo-Vista-Controlador (MVC) se define como un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones web. (37)

Symfony2 basa su funcionamiento interno en la arquitectura MVC. A continuación se muestra el mismo en un esquema:

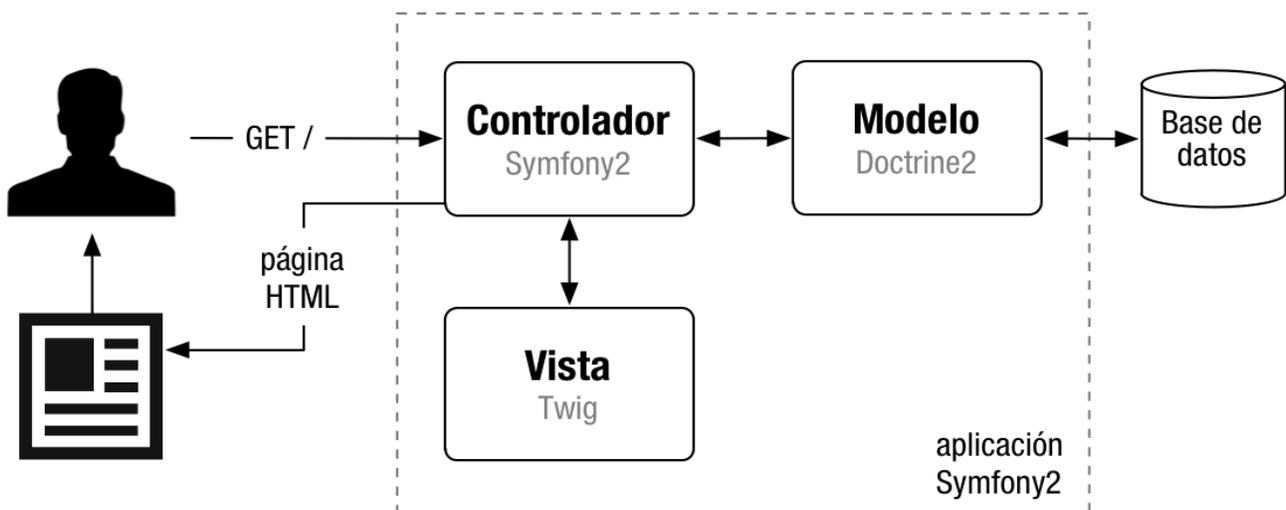


Figura 7: Esquema Modelo-Vista-Controlador

1. El sistema de enrutamiento determina qué Controlador está asociado con la página a la cual se accederá.
2. Symfony2 ejecuta el Controlador asociado a la página que se desea acceder.

3. El Controlador solicita al Modelo los datos. El modelo no es más que una clase PHP especializada en obtener información, normalmente de una base de datos.
4. Con los datos devueltos por el Modelo, el Controlador solicita a la Vista que cree una página mediante una plantilla y que inserte los datos del Modelo.
5. El Controlador entrega al servidor la página creada por la Vista. (20)

Enrutamiento: El sistema de enrutamiento transforma URL en controladores. Más en concreto, determina cuál es el controlador que se debe ejecutar para cada URL solicitada por los usuarios. Esta transformación se realiza mediante la información de las rutas, normalmente configurada en archivos YAML (también se pueden configurar mediante anotaciones y archivos XML).

Controlador: Es la parte de la aplicación que contiene lo que se llama la lógica de negocio, que es una forma elegante de decir que cada controlador se encarga de una funcionalidad completa de la aplicación. Los controladores hacen uso de otros componentes para obtener la información y para generar las páginas, por lo que su código no suele ser muy largo. (20)

En el sistema propuesto un ejemplo de esta estructura se observa en el bundle¹⁵ Tortuga donde operan varios controladores representados, por las clases RecorridoController.php, RastroController.php, TortugaController.php; en el directorio Entity se encuentran las clases entidad que representan el modelo como Tortuga.php, Recorrido.php, Rastros.php y por último en el directorio Resources están contenidos los recursos necesarios para construir las vistas: ya sea las plantillas de Twig, los archivos javascript, css o las imágenes.

3.2 Patrones de diseño

Una de las ventajas de utilizar framework es que se basan en el uso de patrones de diseño, lo que facilita la usabilidad, independientemente del tipo de aplicación que se desee realizar. Symfony hace uso de varios patrones, los cuales incluye por defecto en su arquitectura y están concebidos de forma tal, que obliga al programador a utilizarlos, entre ellos los patrones GRASP¹⁶ y GoF¹⁷.

¹⁵ Paquete que contiene un software y todo lo que necesita para operar.

¹⁶ Por su siglas en inglés General Responsibility Assignment Software Patterns (en español patrones generales de software para asignar responsabilidades).

¹⁷ Por sus siglas en inglés Gang of Four (en español Grupo de 4).

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. (11)

Creador: En las clases controladoras se localizan las acciones definidas en el sistema con el uso de los métodos Actions; donde, en cada una de ellas se pueden crear objetos de las clases que representan las entidades, por lo que se puede afirmar que las clases controladoras son “creadoras” de dichas entidades.

Experto: Según Craig Larman el patrón Experto se encarga de asignar una responsabilidad al experto en información, es decir a la clase que cuenta con la información necesaria para cumplir la responsabilidad. Para el caso del sistema a implementar las clases entidades serían las expertas dado que las mismas son las que poseen toda su información.

Controlador: Se encarga de asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase que represente. (11) En el caso del sistema a realizar un ejemplo de clase controladora sería AreaProtegidaController.php.

Bajo acoplamiento: Este patrón se evidencia dentro del framework Symfony en la capa modelo ya que las clases de acceso a los datos tienen independencia de las clases de abstracción de datos. El bajo acoplamiento se observa en la separación que existe entre entidad, controlador y vista de modo que cada una contiene solo la información necesaria lo cual implica que un cambio en una no afecta o afecta lo menos posible a las restantes.

Alta cohesión: Una de las características principales del framework Symfony es la organización del trabajo en cuanto a la estructura del proyecto, lo cual permite crear y trabajar con clases con una alta cohesión. Se observa en el ejemplo de los controladores donde cada uno contiene únicamente las operaciones e información relacionada a la entidad correspondiente.

Patrones GoF

Decorador: Este patrón se encarga de crear una plantilla de diseño base que incorpora todos los componentes (cabecera, pie de página, menú, secciones entre otros) comunes del sistema

y luego puede ser llamada para su uso en otras plantillas. Para el caso específico de la investigación la plantilla decorada es la clase `page.html.twig`.

Estrategia: Permite crear formularios personalizados que se envían al FormBuilder para transformar el formulario concreto en vistas. La inserción de estas vistas en la plantilla HTML también se realiza utilizando patrón Estrategia, pues el HTML no tiene por qué saber nada sobre el formulario concreto aunque en la práctica el HTML es consciente de la clase de forma que se debe administrar.

Singleton¹⁸: Permite la creación de objetos pertenecientes a una clase, facilitando de esa forma la realización de instancias únicas.

3.3 Modelo de diseño

Según los autores del libro “El Proceso Unificado de Desarrollo de Software”, el modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso centrándose en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación tienen impacto en el sistema a considerar. Además, el modelo de diseño sirve como abstracción de la implementación del sistema y es de ese modo utilizado como una entrada fundamental de las actividades del mismo. (6)

3.3.1 Diagrama de clases del diseño

A continuación se presenta el diagrama de clases del diseño del caso de uso Gestionar recorrido donde se observan los paquetes, las clases participantes y las relaciones entre ellas. Los restantes diagramas de clases se encuentran en el Anexo 14.

¹⁸ *Única instancia*

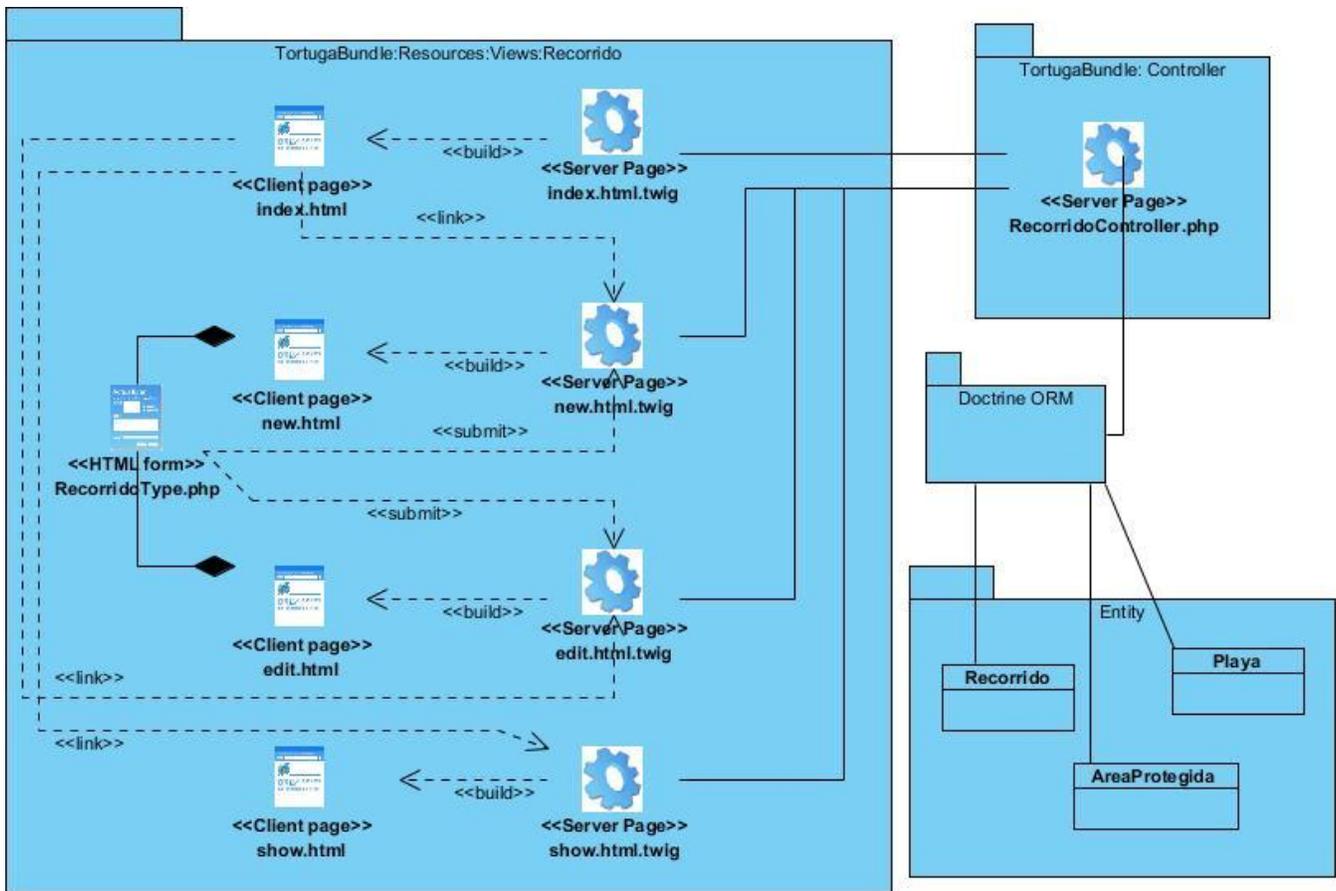


Figura 8: Diagrama de clases del diseño del caso de uso Gestionar recorrido

3.3.2 Diagrama de despliegue

El diagrama de despliegue que se muestra a continuación representa la distribución física del sistema. El mismo está compuesto por una computadora que utilizará el usuario para acceder al sistema y deberá tener instalado un navegador web. Este ordenador se comunica con el servidor de aplicación a través del protocolo HTTP; el cual a su vez utilizará una conexión TCP-IP para la comunicación con el servidor de base de datos. Desde la PC cliente se puede además conectar una impresora mediante la interfaz USB.

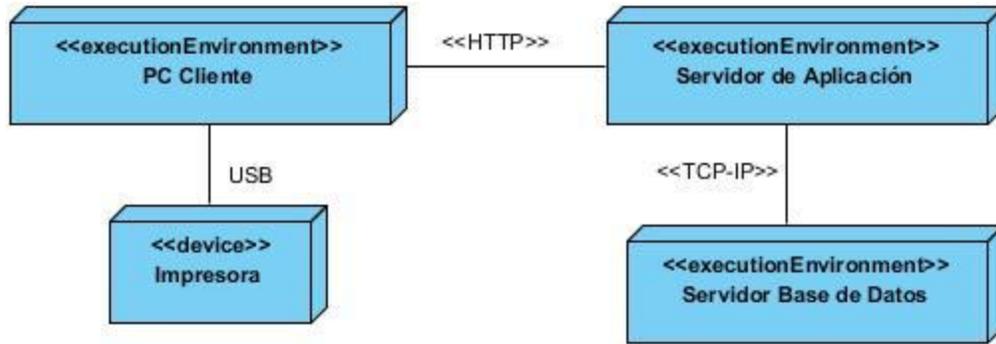


Figura 9: Diagrama de despliegue

3.3.3 Diagramas de secuencia

Cuando un actor interactúa con una aplicación Web, muchas de estas interacciones tienen lugar, entre los distintos elementos que la componen; representarlos en un diagrama de secuencia ayuda a entender mejor su funcionamiento, colaborando con el objetivo de documentar la solución. (38)

A continuación se muestran los diagramas de secuencia del caso de uso Gestionar recorrido, donde se observan las interacciones entre los objetos mediante transferencias de mensajes. Los diagramas de los restantes casos de uso se pueden encontrar en el Anexo 22.

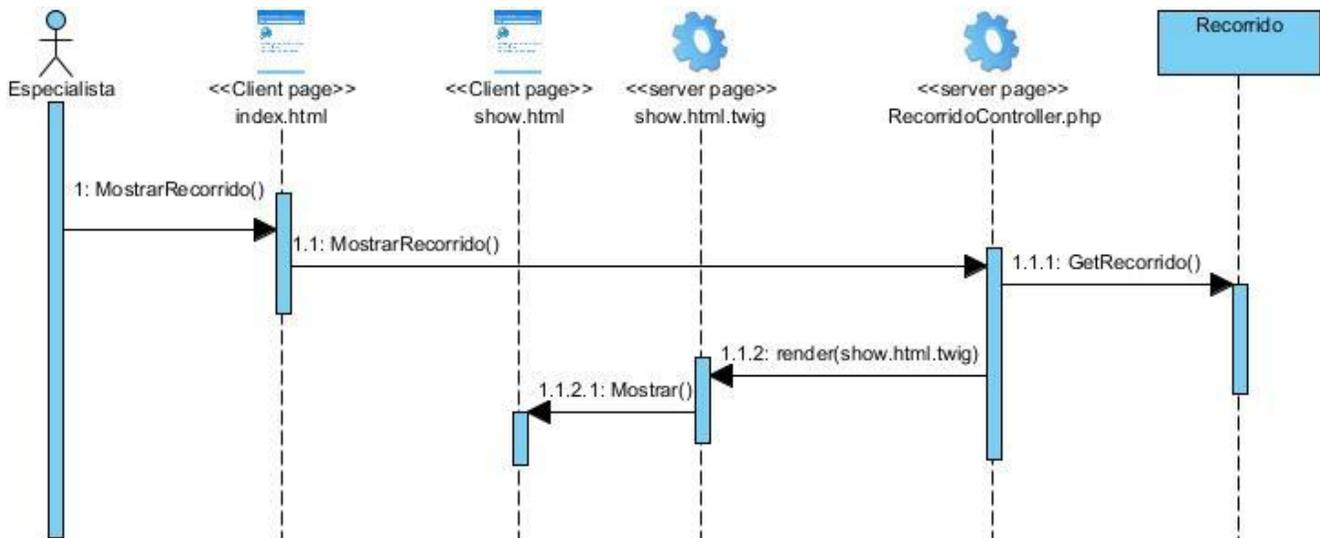


Figura 10: Diagrama de secuencia de la sección Mostrar recorrido

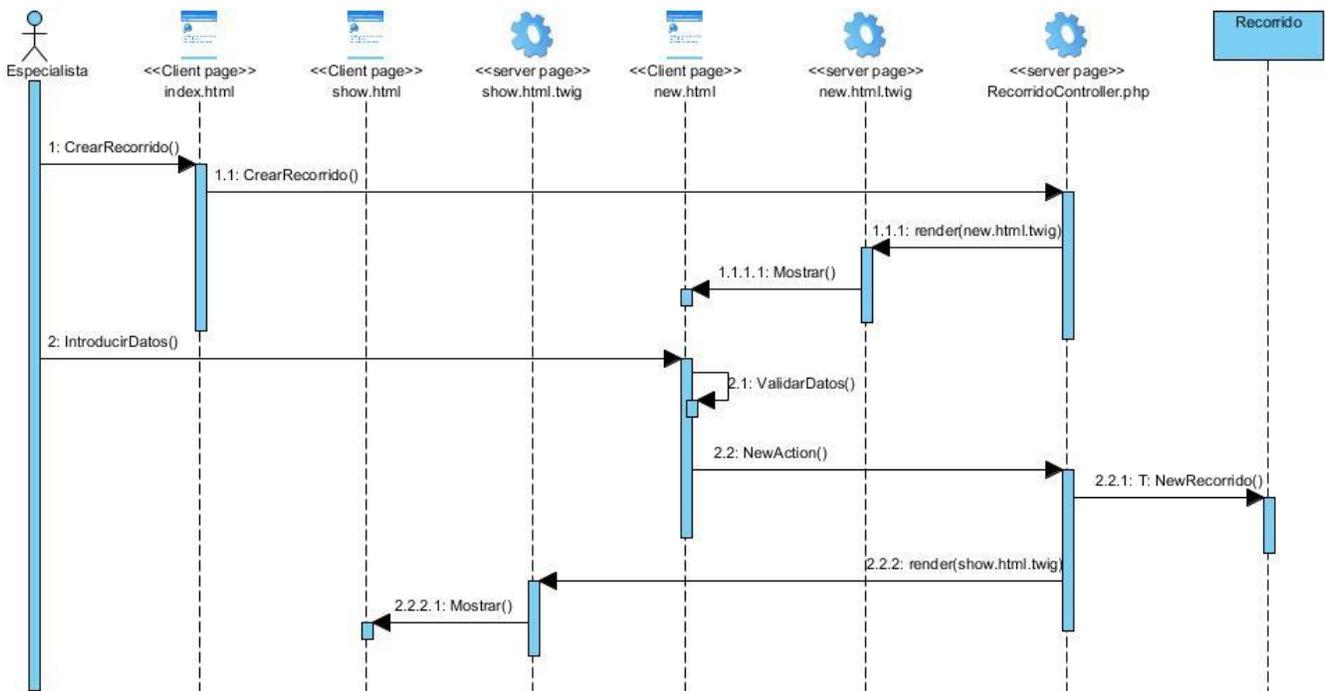


Figura 11: Diagrama de secuencia de la sección Incluir recorrido

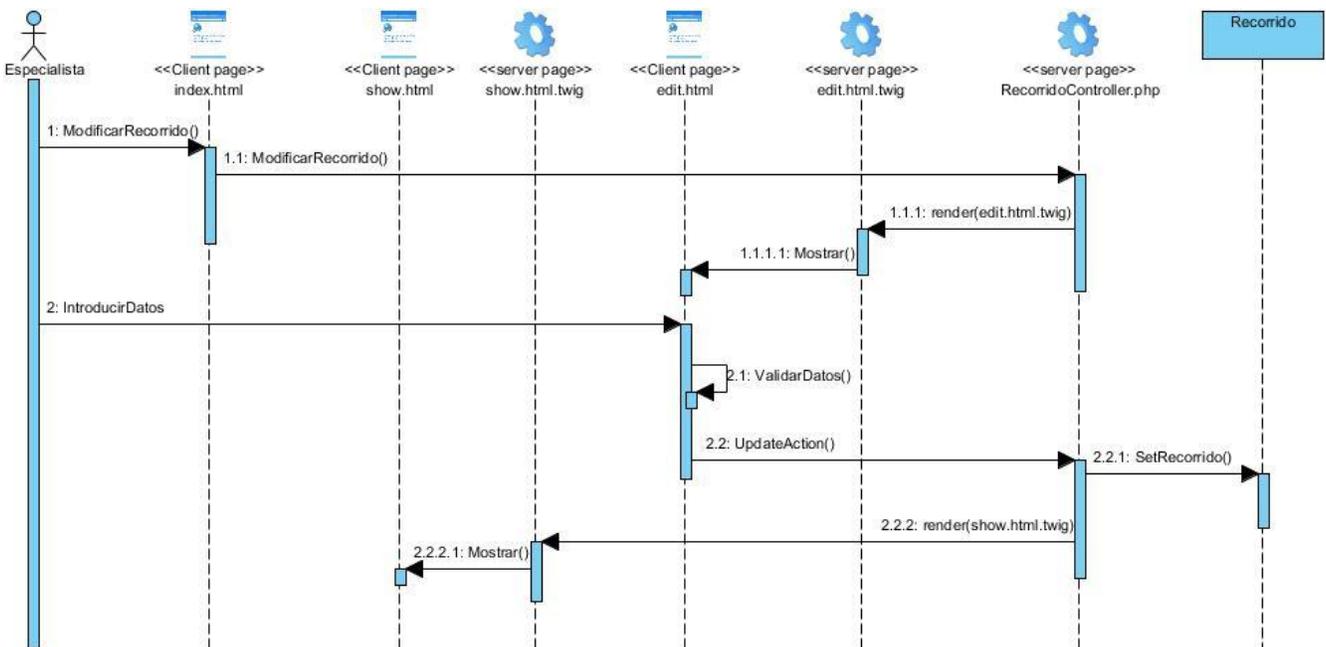


Figura 12: Diagrama de secuencia de la sección Modificar recorrido

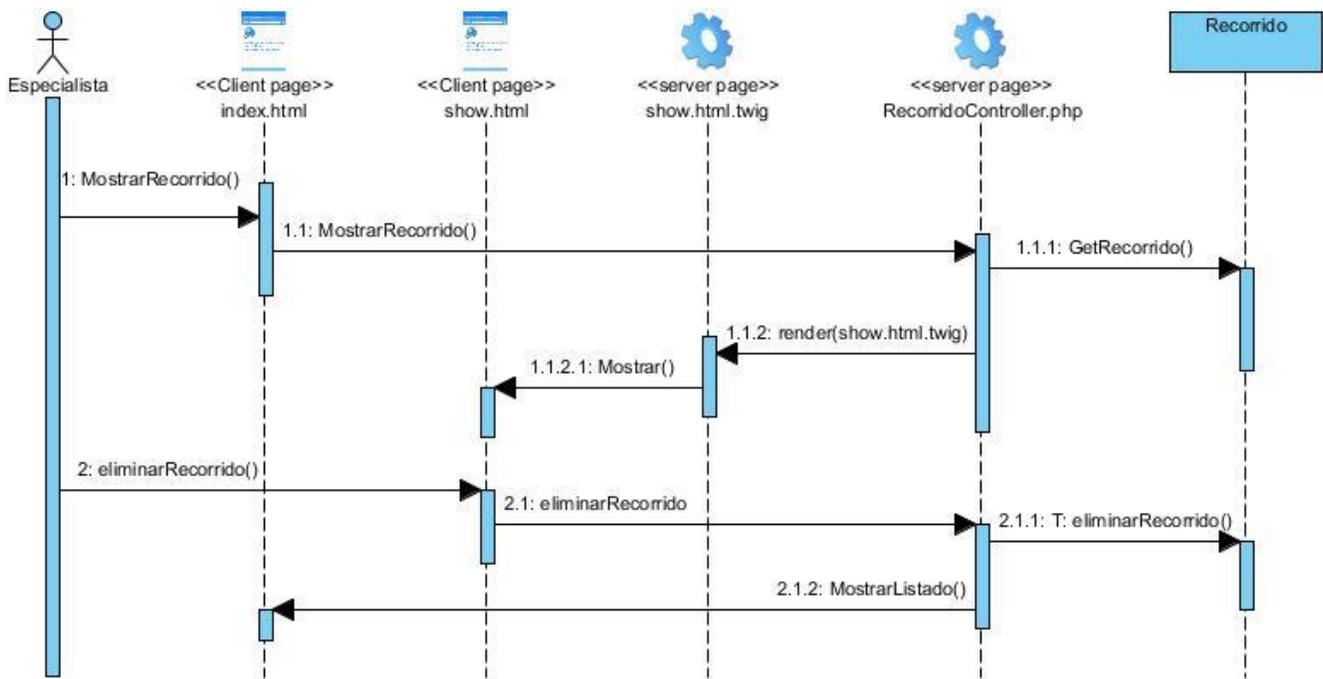


Figura 13: Diagrama de secuencia de la sección Eliminar recorrido

3.4 Diseño de la base de datos

En esta sección se muestra el modelo de datos que está compuesto por entidades que se crean en Symfony y luego pasan a ser tablas de la base de datos, posteriormente son empleadas por las funcionalidades a desarrollar.

Tabla recorrido		
Atributo	Tipo	Descripción
id	integer	Identificador de cada recorrido en la tabla.
responsable_id	integer	Identificador de la tabla que guarda el nombre del responsable.
distanciarecorrida	numeric	Número de la distancia recorrida.
numeroparticipantes	integer	Cantidad de participantes en el recorrido.
fecha	timestamp	Fecha en que se realiza.
horainicio	timestamp	Hora en que empieza el recorrido.
horafin	timestamp	Hora que concluye el recorrido.
faselunar_id	integer	Identificador de la tabla que guarda el estado de la fase lunar.
estadomarea_id	integer	Identificador de la tabla que guarda el estado de la marea.
estadotiempo_id	integer	Identificador de la tabla que guarda el estado del tiempo.
estadomar_id	integer	Identificador de la tabla que guarda el estado del mar.
playa	integer	Identificador de la tabla que guarda la playa.
areaprotegida	integer	Identificador de la tabla que guarda el área protegida.

Tabla playa		
Atributo	Tipo	Descripción
id	integer	Identificador de cada playa en la tabla.
areaprotegida	integer	Identificador de la tabla que guarda el área protegida.
nombre	character varying(255)	Nombre de la playa.
descripcion	text	Describe características de la playa.

Tabla areaprotegida		
Atributo	Tipo	Descripción
id	integer	Identificador de cada área protegida en la tabla.
nombre	character varying(255)	Nombre del área protegida.
descripcion	text	Describe características del área protegida.

Conclusiones del capítulo

Los diagramas de clases y de secuencia realizados, fueron enfocados al cumplimiento de los requisitos funcionales. Estos lograron que se obtuvieran de forma concreta y detallada las relaciones que existen entre las clases, sentando bases para el desarrollo de la propuesta. El diagrama de despliegue facilitó que se representara la distribución física del sistema. La realización del modelo de datos permitió que quedaran modeladas todas las tablas y sus relaciones, necesarias para el progreso de la investigación.

Capítulo 4: Implementación y prueba

Introducción

Los artefactos generados en el diseño constituyen la entrada esencial para el flujo de trabajo de Implementación, el cual propone mostrar cómo se implementan las funcionalidades en términos de componentes y cómo lograr la organización de los mismos. Luego de realizar este flujo se elabora el de Pruebas donde se efectúan las pruebas necesarias a las funcionalidades implementadas para verificar que estén correctamente desarrolladas. No obstante las pruebas realizadas al sistema permiten comprobar su adecuado funcionamiento antes de ser usado por los usuarios finales.

4.1 Modelo de implementación

El modelo de implementación describe cómo los elementos del diseño se implementan en términos de componentes; representa ficheros de código abierto, ficheros de código binario, scripts, ejecutables entre otros.

El modelo de implementación es la entrada principal de las etapas de prueba que siguen a la implementación. Más concretamente, durante la etapa de prueba cada construcción generada durante la implementación es sometida a pruebas de integración, y posiblemente también a pruebas de sistema. (6)

Como parte del modelo de implementación se obtiene el diagrama de componentes que a continuación se presentan.

4.1.1 Diagrama de componentes

Este diagrama muestra cómo el sistema está dividido en componentes y las dependencias entre ellos. Estos pueden ser usados para modelar y documentar cualquier arquitectura del sistema.

Los componentes son implementados por varias clases u objetos del sistema, se considera una unidad autónoma que provee una o más interfaces las cuales representan un contrato de servicios que el componente ofrece. Los componentes pueden ser archivos, códigos fuentes,

ejecutables y paquetes. El diagrama de componentes provee una vista arquitectónica de alto nivel del sistema. Ayuda a los desarrolladores a visualizar el camino de la implementación. Permite tomar decisiones respecto a las tareas de implementación. (6)

A continuación se muestra el diagrama de componentes del caso de uso Gestionar recorrido y la descripción de los principales elementos que lo conforman.

- Paquete Vista: Agrupa los componentes relacionados con las vistas de las funcionalidades del sistema, como son: las plantillas, las direcciones y los formularios.
- Paquete Controlador: Agrupa las clases controladoras encargadas de gestionar todas las acciones para las plantillas.
- Paquete Modelo: Agrupa las clases entidad generadas por Doctrine que proporciona persistencia para los objetos y un lenguaje de consultas orientado a objetos.
- Componente routing: Contiene todos los archivos yml utilizados para definir las rutas de acceso.
- Componente RecorridoType.php: Es el formulario del caso de uso encargado de estructurar cada elemento de la vista para editar o crear recorridos.
- Componente Recorrido: Contiene todas las plantillas con extensión html.twig que son las encargadas de mostrar la información del sistema.
- Componente RecorridoController.php: Es el componente que contiene todos los métodos action los cuales son llamados en las plantillas.
- Componente symfony: Contiene todos los elementos del framework que actúan o intervienen en la realización del caso de uso.
- Componente Recorrido.php: Es la clase entidad donde se originan todas las relaciones que va a tener esa clase entidad con las asociadas a ella, también se establecen algunas validaciones.
- Componente Base de Datos: Encapsula todos los datos del sistema.

- Componente css: Contiene los archivos que garantizan la correcta presentación de la información en las páginas web.
- Componente jquery: Almacena los archivos con código javascript que aportan dinamismo necesario para las páginas web.
- Componente lib: Se refiere a las clases y librerías externas, se suele guardar todo el código común a todas las aplicaciones del proyecto.

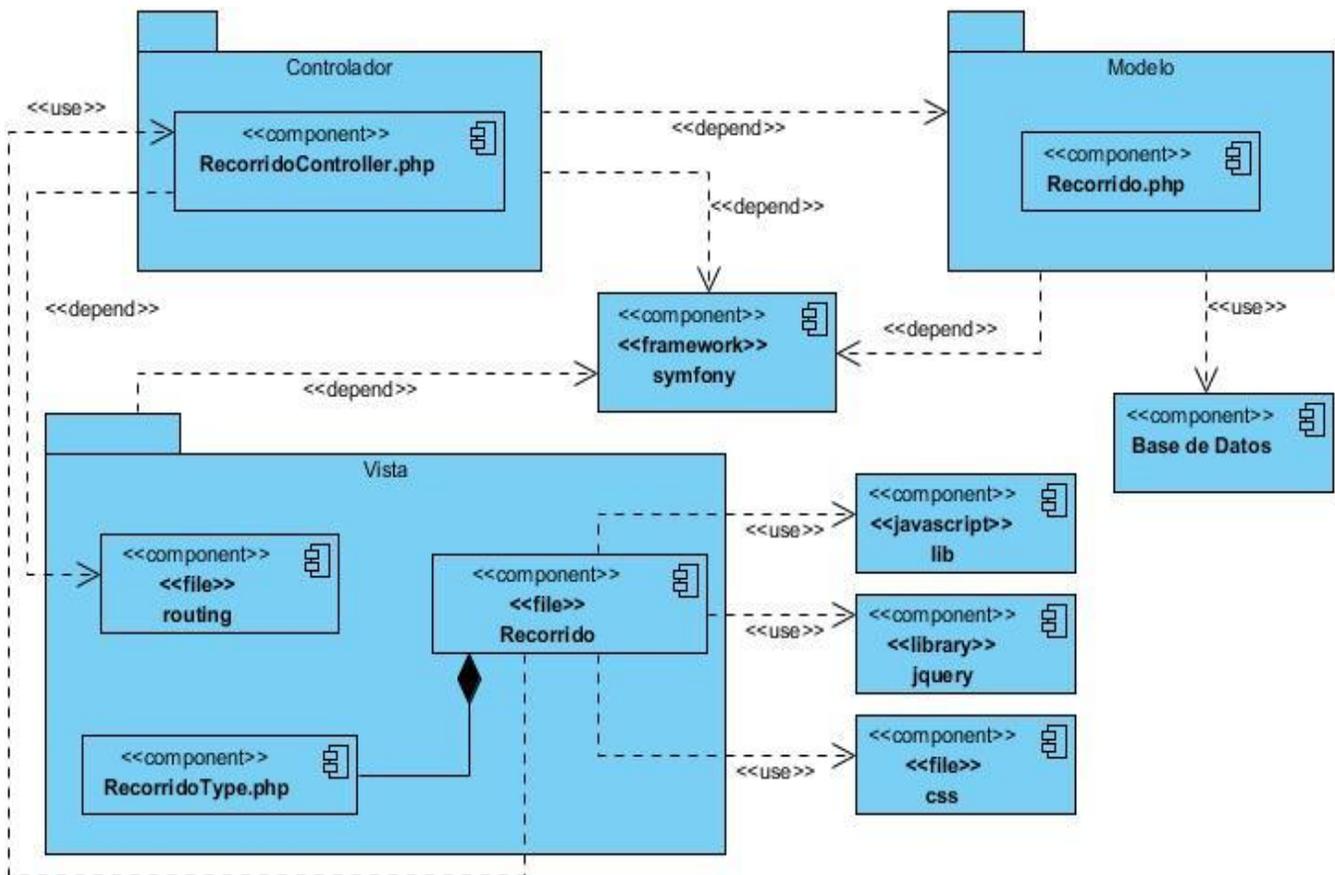


Figura 15: Diagrama de Componentes del caso de uso Gestionar recorrido

4.2 Pruebas de software

A continuación se presentan las pruebas realizadas al sistema. Éstas se utilizaron con el objetivo de identificar y corregir los fallos de la implementación para así verificar el buen funcionamiento de lo desarrollado. Para la realización de las mismas fue trazada una estrategia de prueba.

4.2.1 Estrategia de prueba

Se traza una estrategia de prueba para definir el enfoque y los objetivos generales de las actividades de las pruebas.

Una vez generado el código fuente, es necesario probar el software para descubrir y corregir la mayor cantidad de errores posible antes de entregarlo al cliente. Para lograr contribuir con la calidad del software es recomendable que el producto sea evaluado en la medida en que se va construyendo. Posteriormente, se hace necesario llevar a cabo, paralelo al proceso de desarrollo, un proceso de evaluación y comprobación de los distintos productos o modelos que se van generando, en el que participarán desarrolladores y clientes. Las pruebas de software constituyen un pilar indispensable para evaluar y determinar la calidad de un software. (6)

A continuación se muestra la estrategia diseñada en la investigación:

Nivel	Sistema
Método	Caja negra
Técnica	Partición de equivalencia

4.2.2 Niveles de prueba

El proceso de prueba se realiza en varios niveles, en un determinado momento del ciclo de desarrollo de las funcionalidades. Seguidamente se muestran algunos tipos de niveles de pruebas para valorar cuales se van a utilizar:

Prueba de sistema: Son las pruebas que se hacen cuando el software está funcionando como un todo. Es la actividad de prueba dirigida a verificar el programa final, después que todos los componentes de software y hardware han sido integrados. En un ciclo iterativo estas pruebas ocurren más temprano, tan pronto como subconjuntos de comportamiento de caso de uso son implementados. (39)

Prueba de unidad: Es la prueba enfocada a los elementos testeables más pequeños del software. Es aplicable a componentes representados en el modelo de implementación para verificar que los flujos de control y de datos están cubiertos, y que ellos funcionen como se espera. La prueba de unidad siempre está orientada a caja blanca. Antes de iniciar cualquier otra prueba es preciso probar el flujo de datos de la interfaz del componente. Si los datos no entran correctamente, todas las demás pruebas no tienen sentido. El diseño de casos de prueba de una unidad comienza una vez que se ha desarrollado, revisado y verificado en su sintaxis el código a nivel fuente. (38)

Prueba de integración: Pruebas integrales o pruebas de integración son aquellas que se realizan en el ámbito del desarrollo de software una vez que se han aprobado las pruebas unitarias. Únicamente se refieren a la prueba o pruebas de todos los elementos unitarios que componen un proceso, hecha en conjunto, de una sola vez. Consiste en realizar pruebas para verificar que un gran conjunto de partes del software funcionan juntos. (6)

Para la presente investigación, no se ejecutan pruebas de integración dado que al sistema no se le incorpora ningún módulo, componente o sección que se haya desarrollado ajena a la solución. En el caso de las pruebas de unidad no se realizan mediante el uso de una herramienta informática solo se efectuaron revisiones del código por parte de la desarrolladora. Son las pruebas de sistema las que permitirán comprobar el cumplimiento de los requisitos funcionales y no funcionales por estar enfocadas a verificar cómo trabaja el software una vez concluido su desarrollo.

4.2.3 Métodos de prueba

RUP propone dos métodos fundamentales: caja blanca y caja negra. A continuación se describen ambos métodos:

Las pruebas de caja blanca del software denominadas a veces pruebas de caja de cristal comprueban los caminos lógicos y algoritmos del software proponiendo casos de pruebas que ejerciten conjuntos específicos de condiciones y/o bucles. Pueden examinar el estado del programa en varios puntos para determinar si el estado real coincide con el esperado o mencionado.

Las pruebas de caja negra, también denominadas pruebas funcionales o de comportamiento, se centran en los requisitos funcionales del software. La prueba de caja negra permite al ingeniero del software obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. (40) Éstas se refieren a las pruebas que se llevan a cabo sobre la interfaz del software. Los casos de pruebas pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene. (39)

4.2.4 Partición equivalente

La partición equivalente es una técnica de prueba de caja negra que divide el dominio de entrada de un programa en clases de datos a partir de las cuales pueden derivarse casos de prueba. Un caso de prueba ideal de manejo simple descubre una clase de errores; como, el procesamiento incorrecto de todos los datos de caracteres; que, de otra forma, requeriría la ejecución de muchos casos antes de que se observe el error general. La partición equivalente se esfuerza por definir un caso de prueba que descubra ciertas clases de errores, reduciendo así el número total de casos de prueba que deben desarrollarse.

El diseño de casos de prueba para partición equivalente se basa en una evaluación de las clases de equivalencia para una condición de entrada. Una clase de equivalencia representa un conjunto de estados válidos y no válidos para las condiciones de entrada. Por lo general, una condición de entrada es un valor numérico específico, un rango de valores, un conjunto de valores relacionados o una condición booleana. (6)

4.2.5 Diseño de casos de prueba

Un caso de prueba especifica una forma de probar el sistema, incluyendo la entrada o resultado con la que se ha de probar y las condiciones bajo las que ha de probarse. También puede derivarse de, y por tanto puede seguir la traza de, un caso de uso en el modelo de caso de uso o de una realización de caso de uso en el modelo de diseño.

Los casos de prueba de caja negra demuestran que: (6)

- Las funciones del software son operativas.

- La entrada se acepta de forma adecuada.
- Se produce una salida correcta.
- La integridad de la información externa se mantiene.

A continuación se presenta el caso de prueba del caso de uso Gestionar recorrido. Los restantes casos de pruebas se encuentran en el Anexo 43.

CP Gestionar recorrido

Descripción general

El caso de uso inicia cuando el especialista decide realizar una acción sobre un recorrido, esta puede ser para crear, modificar, ver o eliminar los datos del mismo. Si el especialista selecciona la opción de incluir un recorrido el sistema le brindará la posibilidad de llenar los datos necesarios y le permitirá cancelar la operación o crearlo. En caso de que desee modificar los datos del recorrido, el actor deberá seleccionar el elemento a modificar, el sistema brindará la posibilidad de realizar modificaciones a los campos existentes permitiendo guardar los cambios realizados. Para eliminar un recorrido el actor selecciona el mismo y lo elimina, terminando el caso de uso.

Condiciones de ejecución

El usuario ha sido validado.

Para modificar los datos de un recorrido: el recorrido ha sido seleccionado previamente por el actor.

Para eliminar un recorrido: el recorrido ha sido mostrado previamente por el actor.

SC 1 Incluir recorrido

Escenario	Descripción	Área protegida	Plaza	Fechas	Horario	Horario Fin	Distancias	Participantes	Responsables	Fases	Material	Tiempos	Materia	Respuesta del sistema	Flujo central

										r				
EC 1.1 Entrar a la sección de recorrido	Realiza una acción sobre un recorrido.												Permite realizar varias acciones con un recorrido: - Crear un nuevo recorrido. - Modificar los datos de un recorrido. Ver SC 3 1: "Modificar recorrido". - Ver los datos del recorrido. Ver SC 2: "Ver recorrido". - Eliminar un recorrido. Ver SC 4: "Eliminar recorrido".	Campaña/Recorridos
EC 1.2 Incluir recorrido	Selecciona la opción de crear un recorrido.												Brinda la posibilidad de introducir los datos siguientes: • Área protegida. • Playa • Fecha • Hora inicio • Hora fin • Distancia • Participantes • Responsables • Fase lunar • Marea • Tiempo	Campaña/Recorridos /Crear recorrido

														<ul style="list-style-type: none"> • Mar <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar los datos. • Cancelar la operación. 	
EC 1.3	Intr odu cir dat os del rec orri do	Introduce los datos y selecciona la opción de guardar los datos.	V	V	V	V	V	V	V	V	V	V	V	<ul style="list-style-type: none"> • Valida los datos. • Registra los datos del recorrido. • Muestra un mensaje de información. 	Campaña/Recorridos /Crear recorrido/Aceptar
EC 1.4	Can cela r cre aci ón del rec orri do	Selecciona la opción de Cancelar.												<ul style="list-style-type: none"> • Elimina los datos creados. • Muestra un mensaje de información. • Regresa al listado de recorridos. 	Campaña/Recorridos /Crear recorrido/Cancelar
EC 1.5	Dat os inc om plet os	Existen datos incompletos.	I	V	V	V	V	V	V	V	V	V	V	<ul style="list-style-type: none"> • Muestra encima del campo vacío un mensaje de información. • Muestra un indicador sobre los campos vacíos. 	Campaña/Recorridos /Crear recorrido/Aceptar
			V	I	V	V	V	V	V	V	V	V	V		
			V	V	I	V	V	V	V	V	V	V	V		
			V	V	V	I	V	V	V	V	V	V	V		
			V	V	V	V	I	V	V	V	V	V	V		
			V	V	V	V	V	I	V	V	V	V	V		
			V	V	V	V	V	V	I	V	V	V	V		
			V	V	V	V	V	V	V	I	V	V	V		

		V	V	V	V	V	V	V	V	V	V	I	V	V	<u>Regresa al EC 1.3.</u>	
		V	V	V	V	V	V	V	V	V	V	V	I	V		
		V	V	V	V	V	V	V	V	V	V	V	V	I		
EC 1.6 Datos incorrectos	Existen datos incorrectos.	I	V	V	V	V	V	V	V	V	V	V	V	V	Muestra encima del campo incorrecto un mensaje de información. Muestra un indicador sobre los campos incorrectos. <u>Regresa al EC 1.3.</u>	Campaña/Recorridos/Crear recorrido/Aceptar
		V	I	V	V	V	V	V	V	V	V	V	V	V		
		V	V	I	V	V	V	V	V	V	V	V	V	V		
		V	V	V	I	V	V	V	V	V	V	V	V	V		
		V	V	V	V	I	V	V	V	V	V	V	V	V		
		V	V	V	V	V	I	V	V	V	V	V	V	V		
		V	V	V	V	V	V	V	I	V	V	V	V	V		
		V	V	V	V	V	V	V	V	I	V	V	V	V		
		V	V	V	V	V	V	V	V	V	I	V	V	V		
		V	V	V	V	V	V	V	V	V	V	I	V	V		
		V	V	V	V	V	V	V	V	V	V	V	I	V		
		V	V	V	V	V	V	V	V	V	V	V	V	I		

SC 2 Ver datos del recorrido

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 2.1 Ver datos de un recorrido	Selecciona la opción de ver los datos de un recorrido.	Muestra los datos del recorrido seleccionado: <ul style="list-style-type: none"> • Área protegida. • Playa • Fecha • Hora inicio • Hora fin • Distancia • Participantes • Responsables • Fase lunar • Marea • Tiempo • Mar Y permite: <ul style="list-style-type: none"> • Volver a la vista anterior 	Campaña/Recorridos/Mostrar

		<ul style="list-style-type: none"> • Editar los datos • Eliminar el recorrido 	
EC 2.2 Salir de ver datos del recorrido	Selecciona la opción de volver a la vista anterior.	Regresa al listado de recorridos.	Campaña/Recorridos/Mostrar/Atrás

SC 3 Modificar recorrido

Escenario	Descripción	Área protegida	Playa	Fecha	Hora inicio	Hora Fin	Distancia	Participantes	Responsable	Fase lunar	Marea	Tiempo	Mare	Respuesta del sistema	Flujo central
EC 3.1 Opción de modificar los datos del recorrido	Selecciona la opción de modificar los datos del recorrido.													Muestra los datos del recorrido seleccionado: <ul style="list-style-type: none"> • Área protegida. • Playa • Fecha • Hora inicio • Hora fin • Distancia • Participantes • Responsables • Fase lunar • Marea • Tiempo • Mar Y permite: <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar el recorrido 	Campaña/Recorridos/Editar

EC 3.2 Modifica los datos que necesite y selecciona la opción de aceptar para actualizar	Modifica los datos que necesite y selecciona la opción de aceptar para actualizar.	V	V	V	V	V	V	V	V	V	V	V	V	Valida los datos. Actualiza los datos del recorrido. Muestra un mensaje de información. Muestra los datos del recorrido. Ver Sección 2: "Ver datos del recorrido"	Campana/ Recorridos/ Editar/Aceptar
EC 3.3 Opción de cancelar.	Selecciona la opción de Cancelar la operación.													Regresa al listado de recorridos. Muestra un mensaje de información	Campana/ Recorridos/ Editar/Cancelar
EC 3.5 Datos incorrectos	Existen datos incorrectos.	I	V	V	V	V	V	V	V	V	V	V	V	Muestra un mensaje de información. Muestra un indicador sobre los campos vacíos. <u>Regresa al EC 3.2.</u>	Campana/ Recorridos/ Editar/Aceptar
		V	I	V	V	V	V	V	V	V	V	V	V		
		V	V	I	V	V	V	V	V	V	V	V	V		
		V	V	V	I	V	V	V	V	V	V	V	V		
		V	V	V	V	I	V	V	V	V	V	V	V		
		V	V	V	V	V	I	V	V	V	V	V	V		

		V	V	V	V	V	V	V	I	V	V	V	V		
		V	V	V	V	V	V	V	V	I	V	V	V		
		V	V	V	V	V	V	V	V	V	I	V	V		
		V	V	V	V	I	V	V	V	V	V	I	V		
		V	V	V	V	V	V	V	V	V	V	V	I		
EC 3.6 Datos incompletos	Existen datos incompletos.	I	V	V	V	V	V	V	V	V	V	V	V	Muestra un mensaje de información. Muestra un indicador sobre los campos vacíos. Regresa al EC 3.2.	Campana/Recorridos/Editar/Aceptar
		V	I	V	V	V	V	V	V	V	V	V	V		
		V	V	I	V	V	V	V	V	V	V	V	V		
		V	V	V	I	V	V	V	V	V	V	V	V		
		V	V	V	V	I	V	V	V	V	V	V	V		
		V	V	V	V	V	I	V	V	V	V	V	V		
		V	V	V	V	V	V	V	I	V	V	V	V		
		V	V	V	V	V	V	V	V	I	V	V	V		
		V	V	V	V	V	V	V	V	V	I	V	V		
		V	V	V	V	V	V	V	V	V	V	I	V		
		V	V	V	V	V	V	V	V	V	V	V	I		

SC 4 Eliminar recorrido

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 4.1 Opción de eliminar el recorrido	Selecciona la opción de eliminar el recorrido.	Muestra un mensaje de alerta. Y permite: • Aceptar. • Cancelar.	Campana/Recorridos/Mostrar/Aceptar
EC 4.2 Acepta eliminar el elemento seleccionado	Selecciona la opción Aceptar.	Elimina el recorrido. Muestra un mensaje de información	Campana/Recorridos/Mostrar/Aceptar
EC 4.3 Cancela la operación de eliminar el elemento	Selecciona la opción de Cancelar.	Regresa a la vista anterior.	Campana/Recorridos/Mostrar/Cancelar

4.2.6 Resultados obtenidos en las pruebas

En las pruebas realizadas de sistema se utilizó el método de caja negra para validar la capa de presentación del sistema, con la elaboración de las mismas se comprobó el cumplimiento de los requisitos funcionales y no funcionales del software, a continuación los resultados de las mismas:

Iteraciones	Cantidad de casos de prueba	No conformidades detectadas			
		Alta	Media	Baja	Total
Primera	16	4	12	24	40
Segunda	16	2	4	6	12
Tercera	16	1	1	2	4

Anteriormente se refleja cómo en la medida que se fueron realizando las iteraciones se le da solución a las no conformidades detectadas, donde al final fueron corregidos en su totalidad los errores detectados, dejando el sistema listo para su posterior uso.

Conclusiones del capítulo

Los diagramas de componentes obtenidos en el flujo de trabajo de implementación permitieron concebir el modelo de implementación, de forma tal que describe cómo los elementos del modelo de diseño se implementan en términos de componentes.

Las pruebas realizadas al sistema validaron que las funcionalidades desarrolladas satisfacen los requerimientos especificados, dejando listo el software para su posterior implantación.

Conclusiones

Al finalizar la investigación se arribaron a las siguientes conclusiones:

- Se implementó un sistema de gestión de información con los datos obtenidos en los estudios a las tortugas marinas.
- La solución informática propuesta en esta investigación es la primera existente en Cuba.
- Se hicieron pruebas de sistema y las no conformidades detectadas fueron resueltas en tres iteraciones.
- El cliente quedó satisfecho con el sistema desarrollado, permitiendo solucionar un grupo de problemas existentes en el SNAP.

Recomendaciones

Teniendo en cuenta los resultados durante el desarrollo de la investigación se proponen las siguientes recomendaciones.

- Elaborar un Manual de usuario interactivo para la aplicación.
- Elevar el número de reportes con otros tipos de estadísticas.
- Incluir un Sistema de Información Geográfica (SIG por sus siglas) para la ubicación de los nidos de tortugas.
- Integrar la gestión de la información de los ocho programas restantes.

Referencias bibliográficas

1. *Development of an on-line management information system for community mental health centers.* **HANSEN, KENNETH E, JOHNSON, JAMES H and WILLIAMS, THOMAS A.** 2, s.l. : Behavior Research Methods & Instrumentation, Vol. IX.
2. *Evaluation of computerized health management information system for primary health care in rural India.* **Krishnan, Anand, et al., et al.** 2010.
3. **University of California, Santa Cruz.** Technology & Information Management (TIM) . *What is Information Systems Management (ISM)?* [Online] 2013. URL: <http://tim.soe.ucsc.edu/undergraduates>.
4. **Ingram, David.** Small Bussiness. The Houston Chronicle. *What Is a Management Information System?* [Online] 2013. <http://smallbusiness.chron.com/management-information-system-2104.html>.
5. Sea turtle. [Online] <http://www.iacseaturtle.org/default.htm>.
6. **Jacobson, Ivar, Booch, Grady and Rumbaugh G, James.** *El proceso unificado de desarrollo de software.* Madrid : s.n., 2000.
7. *Métodologías Ágiles en el Desarrollo de Software.* **Canós, José H, Letelier, Patricio and Penadés, M^a Carmen.**
8. **Rational Software** . Rational Unified Process Best Practices for Software Development Teams. [Online] 07 23, 2005. http://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf.
9. **H, José, Letelier, Patricio and Penadés, María del Carmen.** *Metodologías ágiles para el desarrollo de software.* España: Universidad Politécnica de Valencia : s.n., 2006.
10. **ExtremeProgramming.** ExtremeProgramming. [Online] 2009. <http://www.extremeprogramming.org/>.
11. **Craig, Larman.** *UML y patrones.* La Habana : Félix Varela, 2004.
12. *Herramientas CASE.* **Instituto Superior de Estadísticas e Informática.** 1, 1990.
13. **Visual Paradigm.** Visual Paradigm. [Online] [Cited: enero 22, 2013.] <http://www.visual-paradigm.com/aboutus/>.
14. **Pérez, Javier Eguíluz.** *Introducción a Ajax.* [Online] 2008. <http://www.librosweb.es>.

15. **Pérez, Javier Eguíluz.** Introducción al CSS. [Online] 2009. <http://librosweb.es>.
16. **Bos, Bert.** CSS current work. World Wide Web Consortium. [Online] febrero 26, 2011.
17. Libros web - Introducción a JavaScript. [Online] 2009. <http://librosweb.es>.
18. Programación en castellano. [Online] 2008. <http://www.programacion.net/php>.
19. Maestros de la web. [Online] 2005. <http://www.maestrosdelweb.com/>.
20. **Pérez, Javier Eguíluz.** *Desarrollo web ágil con symfony2*. 2012.
21. Twig - The flexible, fast, and secure template engine for PHP. [Online] 2010 - 2012. <http://twig.sensiolabs.org/>.
22. **Traducido por Pacheco, Nacho.** *Doctrine 2 ORM Documentation, Release 2.1: Doctrine Project Team*. 2011.
23. **Pecos, Daniel.** PostgreSQL vs. MySQL. [Online] 2008. [Cited: 02 27, 2013.] http://danielpecos.com/docs/mysql_postgres/x15.html.
24. **PostgreSQL.** Sobre PostgreSQL. [Online] http://www.postgresql.org/es/sobre_postgresql.
25. **Apache.** Servidor HTTP Apache. [Online] 2011. http://httpd.apache.org/docs/2.0/new_features_2_0.html..
26. —. Overview of new features in Apache 2.2. [Online] http://httpd.apache.org/docs/2.2/new_features_2_2.html.
27. Manuel de Zend Framework. [Online] 11 16, 2011. <http://manual.zfdes.com/es/introduction.overview.html>.
28. Libros Web. *Symfony en pocas palabras*. [Online] noviembre 2012. http://www.librosweb.es/symfony_1_0/capitulo1/symfony_en_pocas_palabras.html.
29. Libros Web - Symfony en pocas palabras. [Online] noviembre 2012. http://www.librosweb.es/symfony_1_0/capitulo1/symfony_en_pocas_palabras.html.
30. **Eclipse.** Eclipse- About us. [Online] [Cited: 02 27, 2013.] www.eclipse.org.
31. Convertir Eclipse en un buen IDE para programar con Symfony2. [Online] 07 24, 2012. <http://jonsegador.com/2012/09/convertir-eclipse-buen-ide-programar-symfony2/>.
32. **NetBeans.** [Online] [Cited: 01 20, 2013.] <http://netbeans.org/>.
33. **Otto, Mark.** [Online] 08 19, 2011. <https://dev.twitter.com/blog/bootstrap-twitter>.
34. Bootstrap, desde Twitter. [Online] <http://www.anidocs.es/bootstrap/docs/scaffolding.php>.
35. **McLeod, Graham.** *Brief Introduction to Domain Modeling*. s.l. : Inspired 2009, 2009.
36. **IBM.** Designing a software application using models. [Online] 2005.

<http://publib.boulder.ibm.com/infocenter/rsdvhhelp/v6r0m1/index.jsp?topic=%2Fcom.ibm.rsd.nav.doc%2Ftopics%2Fcusecasemodel.html>.

37. **Alvarez, Miguel Angel**. Desarrollo web. [Online] <http://desarrolloweb.com>.

38. **Navarro, Ing. Jose Angel Franco**. *UML en acción. Modelando Aplicaciones Web*. La Habana : s.n.

39. *Conferencia 7 de Ingeniería de Software II*. **Departamento de Ingeniería**. Curso 2009-2010. 2009-2010.

40. **Pressman, Mc Graw Hill**. *Ingeniería de Software. Un enfoque práctico*. 2005.

Bibliografía

Doctrine. Doctrine. [Online] 02 28, 2013. <http://www.doctrine-project.org/>.

Hernández León, Rolando Alfredo and Coello González, Sayda. *El proceso de investigación científica*. La Habana : Universitaria cubana, 2011. ISBN 978-959-16-1307-3.

INTECO, Laboratorio Nacional de la Calidad de Software de. *Ingeniería de Software: Metodologías y ciclos de vida*. Marzo del 2009. .

Introducción a JavaScript. [Online] 2009. <http://www.librosweb.es>.

Red colombiana para la conservación de tortugas marinas. [Online] <http://redtortugasmarinascolombia.ning.com/>.

Sea turtle. Sea turtle. [Online] <http://www.iacseaturtle.org/default.htm>.

Sitio oficial de NetBeans. [Online] http://netbeans.org/index_es.html.

Vojacek, Alex. Labs Tecnogaming. [Online] abril 24, 2012. [Cited: enero 11, 2013.] <http://labs.tecnogaming.com/2012/04/web-servers-nginx-vs-apache-vs-lighttpd/>.

WWF. Programa de Tortugas Marinas para América Latina y el Caribe. [Online] 2012. http://www.wwfca.org/about/somos222/programa_tortugas.

Glosario de términos

CSS: Hoja de Estilo en Cascada: contienen un conjunto de etiquetas que definen el formato que se aplicará al contenido de las páginas de una Web. Se llama cascada porque una hoja puede heredar los formatos definidos en otra hoja de forma de que no hace falta que vuelva a definirlos. Estas hojas permiten la separación entre el contenido y la presentación de un Sitio Web.

Framework: Es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problema particular que sirve como referencia, para enfrentar y resolver nuevos conflictos de índole similar. Es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, que puede servir de base para la organización y desarrollo de software.

HTTP: Es el protocolo usado para intercambiar archivos (texto, gráfica, imágenes, sonido, video y otros archivos multimedia) en la World Wide Web.

Javascript: Lenguaje de programación interpretado, utilizado principalmente en páginas Web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C. Actualmente todos los navegadores modernos interpretan el código Javascript integrado dentro de las páginas Web.

PHP: PHP es un lenguaje multiplataforma, multiparadigma, script (no se compila para conseguir códigos máquina sino que existe un intérprete que lee el código y se encarga de ejecutar las instrucciones que contiene éste código) para el desarrollo de páginas web dinámicas del lado del servidor, cuyos fragmentos de código se intercalan fácilmente en páginas HTML, debido a esto y a que es de código.

Sistema de gestión de información: Proceso mediante el cual los datos entrada se registran, almacenan, recuperan y procesan para la toma de decisiones salida.

XHTML: Es el lenguaje pensado para sustituir a HTML como estándar para las páginas web.

Anexos

Anexo 1: Descripción del caso uso Gestionar usuario

Objetivo	Crear, modificar, ver o eliminar un usuario	
Actores	Administrador (Inicia): Crea, modifica, ve y valida los datos de un usuario en el sistema.	
Resumen	El caso de uso inicia cuando el administrador decide realizar una acción crear, modificar, ver o eliminar los datos de un usuario. Si el administrador selecciona la opción de incluir un usuario el sistema le brindará la posibilidad de llenar los datos necesarios y le permitirá cancelar la operación o crearlo. En caso de que desee modificar los datos de un usuario, el actor deberá seleccionar el elemento a modificar, el sistema brindará la posibilidad de realizar modificaciones a los campos existentes permitiendo guardar los cambios realizados. Para eliminar un usuario el actor selecciona el mismo y lo elimina. Terminando el caso de uso.	
Precondicion es	Para modificar los datos de un usuario: el usuario ha sido seleccionado previamente por el actor. Para eliminar un usuario: el usuario ha sido mostrado previamente por el actor.	
Postcondicio nes	Se incluyó, modificó, mostró o eliminó un usuario por el actor	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
1.	Realiza una acción sobre un usuario.	
2.		Permite realizar varias acciones con un usuario: <ul style="list-style-type: none"> - Crear un nuevo usuario. - Modificar los datos de un usuario. Ver Sección 1: “Modificar usuario”. - Ver los datos del usuario. Ver Sección 2: “Ver usuario”. - Eliminar un usuario. Ver Sección 3: “Eliminar usuario”.
3.	Selecciona la opción de crear un nuevo usuario.	
4.		Brinda la posibilidad de introducir los datos siguientes: <ul style="list-style-type: none"> • Nombre.

		<ul style="list-style-type: none"> • Apellidos • Carnet de identidad • Correo electrónico • Contraseña • Repetir contraseña • Tipo de usuario <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar los datos. • Cancelar la operación.
5.	Introduce los datos y selecciona la opción de guardar los datos del usuario	
6.		Valida los datos
7.		Registra los datos del usuario
8.		El caso de uso termina
Flujos alternos		
*.a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*.a.1 Regresa al listado de usuarios
		*.a.2 Muestra un mensaje de información.
		*.a.3 El caso de uso termina
6.a Existen datos incompletos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos vacíos.
		7. a.3 Regresa al paso 5 del flujo básico.
6.a Existen datos incorrectos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos incorrectos.
		7. a.3 Regresa al paso 5 del flujo básico.
Sección 1: “Modificar usuario”		
Flujo básico		
	Actor	Sistema
1.	Selecciona la opción de modificar los datos de un usuario	
2.		Muestra los datos de un usuario

		<p>permitiendo modificar los valores del mismo:</p> <ul style="list-style-type: none"> • Nombre. • Apellidos • Carnet de identidad • Correo electrónico • Contraseña • Repetir contraseña • Tipo de usuario • Especie <p>Y permite:</p> <ul style="list-style-type: none"> • Editar los datos • Cancelar la operación
3.	Modifica los datos que necesite y selecciona la opción de editar los mismos.	
4.		Valida los datos
5.		Actualiza los datos del usuario.
6.		El caso de uso termina.
Flujos alternos		
*a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*a.1 Regresa al listado de usuarios
		*a.2 Muestra un mensaje de información
		*a.3 El caso de uso termina
4.a Existen datos incompletos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos vacíos.
		5. a.3 Regresa al paso 3 del flujo básico.
4.a Existen datos incorrectos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos incorrectos.
		5. a.3 Regresa al paso 3 del flujo básico.

Sección 2: “Ver usuario”		
Flujo básico		
	Actor	Sistema
1.	Selecciona la opción de ver los datos de un usuario	
2.		<p>Muestra los datos del usuario seleccionado:</p> <ul style="list-style-type: none"> • Nombre. • Apellidos • Carnet de identidad • Correo electrónico • Tipo de usuario <p>Y permite:</p> <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar el usuario
3.		El caso de uso termina
Sección 3: “Eliminar usuario”		
Flujo básico		
	Actor	Sistema
1.	Selecciona la opción de eliminar un usuario	
2.		<p>Muestra un mensaje de alerta.</p> <p>Y permite:</p> <ul style="list-style-type: none"> • Aceptar. • Cancelar.
3.	Selecciona la opción de aceptar.	
4.		Elimina el usuario.
5.		Muestra un mensaje de información
6.		El caso de uso termina.
Flujos alternos		
2.a El actor selecciona la opción de cancelar		
	Actor	Sistema
1.		2. a.1 Regresa a la vista anterior.
2.		2. a.2 El caso de uso termina.
Requisitos no funcionales	<p>La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido.</p> <p>Debe notificar al usuario ante la presencia de un error.</p>	

	A los usuarios autorizados se les debe garantizar el acceso a la información.
--	---

Anexo 2: Descripción del caso uso Consultar datos de los usuarios

Objetivo	Consultar datos del usuario.	
Actores	Administrador (Inicia): Consulta datos del usuario.	
Resumen	El caso de uso se inicia cuando el Administrador desea consultar los datos de un usuario. El actor introduce los datos de búsqueda y selecciona la opción buscar. El sistema muestra el listado de coincidencias. El actor consulta los datos. El caso de uso termina.	
Precondiciones	Debe haberse generado el escritorio de trabajo del usuario autenticado	
Postcondiciones	Se consultó los datos del usuario, coincidente con el criterio de búsqueda.	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
9.	Accede a la opción de consultar datos del usuario.	
10.		Brinda la posibilidad de realizar la búsqueda por algunos criterios elementales como: <ul style="list-style-type: none"> • Nombre • Apellidos • Correo Y permite: <ul style="list-style-type: none"> • Buscar
11.	Introduce o selecciona los datos que considere para realizar una búsqueda.	
12.	Selecciona la opción de Buscar.	
13.		Consulta los datos y muestra una lista de posibles coincidencias.
14.	Consulta los datos de los elementos de la lista de coincidencias.	
15.		El caso de uso termina
Flujos alternos		
*.a El sistema no encuentra ninguna coincidencia.		
	Actor	Sistema

		*.a.1 Muestra un mensaje de información. Y permite: <ul style="list-style-type: none"> Realizar una nueva búsqueda.
		*.a.2 Regresa al paso 2 del Flujo Básico.
		*.a.3 El caso de uso termina
6. a El actor selecciona la opción de realizar una nueva búsqueda.		
	Actor	Sistema
		7. a.1 Regresa al paso 2 del Flujo Básico.
		7. a.2 El caso de uso termina.
Requisitos no funcionales	<p>La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido.</p> <p>Debe notificar al usuario ante la presencia de un error.</p> <p>A los usuarios autorizados se les debe garantizar el acceso a la información.</p>	

Anexo 3: Descripción del caso uso Autenticar usuario

Objetivo	Autenticar un usuario	
Actores	Administrador, Especialista (Inicia): Se autentica en el sistema.	
Resumen	El caso de uso inicia cuando el administrador o especialista decide entrar al sistema. El actor selecciona la opción de autenticarse el sistema le brindará la posibilidad de llenar los datos necesarios. Terminando el caso de uso.	
Precondicion es	El escritorio de trabajo debe haberse generado	
Postcondicio nes	Se autenticó un usuario	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
16.	Selecciona la opción de autenticarse	
17.		<p>Brinda la posibilidad de introducir los datos siguientes:</p> <ul style="list-style-type: none"> Usuario Contraseña No cerrar sección. <p>Y permite:</p> <ul style="list-style-type: none"> Autenticarse.

18.	Introduce los datos y selecciona la opción de Autenticarse.	
19.		Valida los datos
20.		Entra a la sección del usuario autenticado.
21.		El caso de uso termina
Flujos alternos		
4.a Existen datos incorrectos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.3 Regresa al paso 5 del flujo básico.
Requisitos no funcionales	<p>La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido.</p> <p>Debe notificar al usuario ante la presencia de un error.</p> <p>A los usuarios autorizados se les debe garantizar el acceso a la información.</p>	

Anexo 4: Descripción del caso uso Gestionar área protegida

Objetivo	Crear, modificar, ver o eliminar un área protegida.
Actores	Especialista (Inicia): Crea, modifica, ve y valida los datos de un área protegida.
Resumen	El caso de uso inicia cuando el especialista decide realizar una acción crear, modificar, ver o eliminar los datos de un área protegida. Si el especialista selecciona la opción de incluir un área protegida el sistema le brindará la posibilidad de llenar los datos necesarios y le permitirá cancelar la operación o crearlo. En caso de que desee modificar los datos del área protegida, el actor deberá seleccionar el elemento a modificar, el sistema brindará la posibilidad de realizar modificaciones a los campos existentes permitiendo guardar los cambios realizados. Para eliminar un área protegida el actor selecciona la misma y la elimina. Terminando el caso de uso.
Precondiciones	<p>El usuario ha sido validado.</p> <p>Para modificar los datos de un área protegida: el área protegida ha sido seleccionada previamente por el actor.</p> <p>Para eliminar un área protegida: el área protegida ha sido mostrada previamente por el actor.</p>
Postcondiciones	Se incluyó, modificó, mostró o eliminó un área protegida por el actor
Flujo de eventos	
Flujo básico	

	Actor	Sistema
22.	Realiza una acción sobre un área protegida.	
23.		<p>Permite realizar varias acciones con un área protegida:</p> <ul style="list-style-type: none"> - Crear una nueva área protegida. - Modificar los datos de un área protegida. Ver Sección 1: “Modificar área protegida”. - Ver los datos del área protegida. Ver Sección 2: “Ver área protegida”. - Eliminar un área protegida. Ver Sección 3: “Eliminar área protegida”.
24.	Selecciona la opción de crear una nueva área protegida	
25.		<p>Brinda la posibilidad de introducir los datos siguientes:</p> <ul style="list-style-type: none"> • Nombre • Descripción <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar los datos. • Cancelar la operación.
26.	Introduce los datos y selecciona la opción de guardar los datos del área protegida	
27.		Valida los datos
28.		Registra los datos del área protegida
29.		El caso de uso termina
Flujos alternos		
*.a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*.a.1 Regresa al listado de áreas protegidas
		*.a.2 Muestra un mensaje de información.
		*.a.3 El caso de uso termina
6.a Existen datos incompletos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos vacíos.

		7. a.3 Regresa al paso 5 del flujo básico.
6.a Existen datos incorrectos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos incorrectos.
		7. a.3 Regresa al paso 5 del flujo básico.
Sección 1: “Modificar área protegida”		
Flujo básico		
	Actor	Sistema
1.	Selecciona la opción de modificar los datos de un área protegida	
2.		Muestra el área protegida permitiendo modificar los valores de la misma: <ul style="list-style-type: none"> • Nombre • Descripción Y permite: <ul style="list-style-type: none"> • Editar los datos • Cancelar la operación
3.	Modifica los datos que necesite y selecciona la opción de editar los mismos.	
4.		Valida los datos
5.		Actualiza los datos del área protegida.
6.		El caso de uso termina.
Flujos alternos		
*.a El actor selecciona la opción de cancelar		
	Actor	Sistema
1.		*a.1 Regresa al listado de áreas protegida
		*a.2 Muestra un mensaje de información
2.		*a.3 El caso de uso termina
4.a Existen datos incompletos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos vacíos.
		5. a.3 Regresa al paso 3 del flujo básico.
4.a Existen datos incorrectos		

	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos incorrectos.
		5. a.3 Regresa al paso 3 del flujo básico.
Sección 2: “Ver área protegida”		
Flujo básico		
	Actor	Sistema
1.	Selecciona la opción de ver los datos de un área protegida	
		Muestra los datos del área protegida seleccionado: <ul style="list-style-type: none"> • Nombre • Descripción Y permite: <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar el área protegida
2.		El caso de uso termina
Sección 3: “Eliminar área protegida”		
Flujo básico		
	Actor	Sistema
7.	Selecciona la opción de eliminar un área protegida	
8.		Muestra un mensaje de alerta. Y permite: <ul style="list-style-type: none"> • Aceptar. • Cancelar.
9.	Selecciona la opción de aceptar.	
10.		Elimina el área protegida.
11.		Muestra un mensaje de información
12.		El caso de uso termina.
Flujos alternos		
2.a El actor selecciona la opción de cancelar		
	Actor	Sistema
1.		2. a.1 Regresa a la vista anterior.
2.		2. a.2 El caso de uso termina.

Requisitos no funcionales	<p>La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido.</p> <p>Debe notificar al usuario ante la presencia de un error.</p> <p>A los usuarios autorizados se les debe garantizar el acceso a la información.</p>
----------------------------------	---

Anexo 5: Descripción del caso uso Consultar datos del área protegida

Objetivo	Consultar datos del área protegida.	
Actores	Administrador (Inicia): Consulta datos del área protegida.	
Resumen	El caso de uso se inicia cuando el Administrador desea consultar los datos de un área protegida. El actor introduce los datos de búsqueda y selecciona la opción buscar. El sistema muestra el listado de coincidencias. El actor consulta los datos. El caso de uso termina.	
Precondiciones	Debe haberse generado el escritorio de trabajo del usuario autenticado	
Postcondiciones	Se consultó los datos del área protegida, coincidente con el criterio de búsqueda.	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
30.	Accede a la opción de consultar datos del área protegida.	
31.		<p>Brinda la posibilidad de realizar la búsqueda por algunos criterios elementales como:</p> <ul style="list-style-type: none"> • Nombre <p>Y permite:</p> <ul style="list-style-type: none"> • Buscar
32.	Introduce o selecciona los datos que considere para realizar una búsqueda.	
33.	Selecciona la opción de Buscar.	
34.		Consulta los datos y muestra una lista de posibles coincidencias.
35.	Consulta los datos de los elementos de la lista de coincidencias.	
36.		El caso de uso termina
Flujos alternos		
*.a El sistema no encuentra ninguna coincidencia.		

	Actor	Sistema
		*.a.1 Muestra un mensaje de información. Y permite: <ul style="list-style-type: none"> Realizar una nueva búsqueda.
		*.a.2 Regresa al paso 2 del Flujo Básico.
		*.a.3 El caso de uso termina
6. a El actor selecciona la opción de realizar una nueva búsqueda.		
	Actor	Sistema
		7. a.1 Regresa al paso 2 del Flujo Básico.
		7. a.2 El caso de uso termina.
Requisitos no funcionales	<p>La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido.</p> <p>Debe notificar al usuario ante la presencia de un error.</p> <p>A los usuarios autorizados se les debe garantizar el acceso a la información.</p>	

Anexo 6: Descripción del caso uso Gestionar playa:

Objetivo	Crear, modificar, ver o eliminar una playa.	
Actores	Especialista (Inicia): Crea, modifica, ve y valida los datos de una playa.	
Resumen	El caso de uso inicia cuando el especialista decide realizar una acción crear, modificar, ver o eliminar los datos de una playa. Si el especialista selecciona la opción de incluir una playa el sistema le brindará la posibilidad de llenar los datos necesarios y le permitirá cancelar la operación o crearla. En caso de que desee modificar los datos de la playa, el actor deberá seleccionar el elemento a modificar, el sistema brindará la posibilidad de realizar modificaciones a los campos existentes permitiendo guardar los cambios realizados. Para eliminar una playa el actor selecciona la misma y la elimina. Terminando el caso de uso.	
Precondiciones	<p>El usuario ha sido validado.</p> <p>Para modificar los datos de una playa: la playa ha sido seleccionada previamente por el usuario.</p> <p>Para eliminar una playa: la playa ha sido mostrada previamente por el usuario.</p>	
Postcondiciones	Se incluyó, modificó, mostró o eliminó una playa por el usuario	
Flujo de eventos		
Flujo básico		
	Actor	Sistema

37.	Realiza una acción sobre una playa.	
38.		<p>Permite realizar varias acciones con una playa:</p> <ul style="list-style-type: none"> - Crear una nueva playa. - Modificar los datos de una playa. Ver Sección 1: “Modificar playa”. - Ver los datos de la playa. Ver Sección 2: “Ver playa”. - Eliminar una playa. Ver Sección 3: “Eliminar playa”.
39.	Selecciona la opción de crear una nueva playa.	
40.		<p>Brinda la posibilidad de introducir los datos siguientes:</p> <ul style="list-style-type: none"> • Nombre • Descripción • Área protegida <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar los datos. • Cancelar la operación.
41.	Introduce los datos y selecciona la opción de guardar los datos de la playa	
42.		Valida los datos
43.		Registra los datos de la playa.
44.		El caso de uso termina
Flujos alternos		
*.a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*.a.1 Regresa al listado playas
		*.a.2 Muestra un mensaje de información.
		*.a.3 El caso de uso termina
6.a Existen datos incompletos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos vacíos.
		7. a.3 Regresa al paso 5 del flujo básico.
6.a Existen datos incorrectos		
	Actor	Sistema

		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos incorrectos.
		7. a.3 Regresa al paso 5 del flujo básico.
Sección 1: “Modificar playa”		
Flujo básico		
	Actor	Sistema
7.	Selecciona la opción de modificar los datos de una playa.	
8.		Muestra la playa permitiendo modificar los valores de la misma: <ul style="list-style-type: none"> • Nombre • Descripción • Área protegida Y permite: <ul style="list-style-type: none"> • Editar los datos • Cancelar la operación
9.	Modifica los datos que necesite y selecciona la opción de editar los mismos.	
10.		Valida los datos
11.		Actualiza los datos de la playa.
12.		El caso de uso termina.
Flujos alternos		
*a El actor selecciona la opción de cancelar		
	Actor	Sistema
1.		*a.1 Regresa al listado de playa.
		*a.2 Muestra un mensaje de información
2.		*a.3 El caso de uso termina
4.a Existen datos incompletos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos vacíos.
		5. a.3 Regresa al paso 3 del flujo básico.
4.a Existen datos incorrectos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.

		5. a.2 Muestra un indicador sobre los campos incorrectos.
		5. a.3 Regresa al paso 3 del flujo básico.
Sección 2: “Ver playa”		
Flujo básico		
	Actor	Sistema
1.	Selecciona la opción de ver los datos de una playa.	
		Muestra los datos de la playa seleccionada: <ul style="list-style-type: none"> • Nombre • Descripción • Área protegida Y permite: <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar la playa
2.		El caso de uso termina
Sección 3: “Eliminar playa”		
Flujo básico		
	Actor	Sistema
13.	Selecciona la opción de eliminar una playa.	
14.		Muestra un mensaje de alerta. Y permite: <ul style="list-style-type: none"> • Aceptar. • Cancelar.
15.	Selecciona la opción de aceptar.	
16.		Elimina la playa.
17.		Muestra un mensaje de información
18.		El caso de uso termina.
Flujos alternos		
2.a El actor selecciona la opción de cancelar		
	Actor	Sistema
1.		2. a.1 Regresa a la vista anterior.
2.		2. a.2 El caso de uso termina.
Requisitos no	La información aparecerá correctamente organizada de forma, que el	

funcionales	<p>usuario no se encuentre perdido.</p> <p>Debe notificar al usuario ante la presencia de un error.</p> <p>A los usuarios autorizados se les debe garantizar el acceso a la información.</p>
--------------------	--

Anexo 7: Descripción del caso uso Consultar datos de la playa

Objetivo	Consultar datos de la playa.	
Actores	Administrador (Inicia): Consulta datos de las playas.	
Resumen	El caso de uso se inicia cuando el Administrador desea consultar los datos de una playa. El actor introduce los datos de búsqueda y selecciona la opción buscar. El sistema muestra el listado de coincidencias. El actor consulta los datos. El caso de uso termina.	
Precondiciones	Debe haberse generado el escritorio de trabajo del usuario autenticado	
Postcondiciones	Se consultó los datos de la playa, coincidente con el criterio de búsqueda.	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
45.	Accede a la opción de consultar datos de la playa.	
46.		<p>Brinda la posibilidad de realizar la búsqueda por algunos criterios elementales como:</p> <ul style="list-style-type: none"> • Playa • Área protegida <p>Y permite:</p> <ul style="list-style-type: none"> • Buscar
47.	Introduce o selecciona los datos que considere para realizar una búsqueda.	
48.	Selecciona la opción de Buscar.	
49.		Consulta los datos y muestra una lista de posibles coincidencias.
50.	Consulta los datos de los elementos de la lista de coincidencias.	
51.		El caso de uso termina
Flujos alternos		
*.a El sistema no encuentra ninguna coincidencia.		

	Actor	Sistema
		*.a.1 Muestra un mensaje de información. Y permite: <ul style="list-style-type: none"> Realizar una nueva búsqueda.
		*.a.2 Regresa al paso 2 del Flujo Básico.
		*.a.3 El caso de uso termina
6. a El actor selecciona la opción de realizar una nueva búsqueda.		
	Actor	Sistema
		7. a.1 Regresa al paso 2 del Flujo Básico.
		7. a.2 El caso de uso termina.
Requisitos no funcionales	<p>La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido.</p> <p>Debe notificar al usuario ante la presencia de un error.</p> <p>A los usuarios autorizados se les debe garantizar el acceso a la información.</p>	

Anexo 8: Descripción del caso uso Gestionar tortuga

Objetivo	Crear, modificar, ver o eliminar un tortuga
Actores	Especialista (Inicia): Crea, modifica, ve y valida los datos de la tortuga en el sistema.
Resumen	El caso de uso inicia cuando el especialista decide realizar una acción crear, modificar, ver o eliminar los datos de una tortuga. Si el especialista selecciona la opción de incluir una tortuga el sistema le brindará la posibilidad de llenar los datos necesarios y le permitirá cancelar la operación o crearla. En caso de que desee modificar los datos de la tortuga, el actor deberá seleccionar el elemento a modificar, el sistema brindará la posibilidad de realizar modificaciones a los campos existentes permitiendo guardar los cambios realizados. Para eliminar una tortuga el actor procede a ver sus datos y luego la elimina terminando el caso de uso.
Precondiciones	<p>El usuario ha sido validado.</p> <p>Para modificar los datos de una tortuga: la tortuga ha sido seleccionada previamente por el usuario.</p> <p>Para eliminar una tortuga: la tortuga ha sido mostrada previamente por el usuario.</p>
Postcondiciones	Se incluyó, modificó, mostró o eliminó una tortuga por el usuario.
Flujo de eventos	

Flujo básico		
	Actor	Sistema
52.	Realiza una acción sobre una tortuga.	
53.		<p>Permite realizar varias acciones con una tortuga:</p> <ul style="list-style-type: none"> - Crear una nueva tortuga. - Modificar los datos de una tortuga. Ver Sección 1: “Modificar tortuga”. - Ver los datos de la tortuga. Ver Sección 2: “Ver tortuga”. - Eliminar una tortuga. Ver Sección 3: “Eliminar tortuga”.
54.	Selecciona la opción de crear una nueva tortuga	
55.		<p>Brinda la posibilidad de introducir los datos siguientes:</p> <ul style="list-style-type: none"> • Área protegida. • Playa • Recorrido • Marca • Fibropapiloma • Ancho recto • Ancho curvo • Largo recto • Largo curvo • Imagen • Hora descenso • Especie • Acción <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar los datos. • Cancelar la operación.
56.	Introduce los datos y selecciona la opción de guardar los datos de la tortuga	
57.		Valida los datos
58.		Registra los datos de la tortuga
59.		El caso de uso termina
Flujos alternos		

*.a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*.a.1 Regresa al listado de tortugas
		*.a.2 Muestra un mensaje de información.
		*.a.3 El caso de uso termina
6.a Existen datos incompletos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos vacíos.
		7. a.3 Regresa al paso 5 del flujo básico.
6.a Existen datos incorrectos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos incorrectos.
		7. a.3 Regresa al paso 5 del flujo básico.
Sección 1: “Modificar tortuga”		
Flujo básico		
	Actor	Sistema
7.	Selecciona la opción de modificar los datos de una tortuga	
8.		<p>Muestra los datos de una tortuga permitiendo modificar los valores de la misma:</p> <ul style="list-style-type: none"> • Área protegida. • Playa • Recorrido • Marca • Fibropapiloma • Ancho recto • Ancho curvo • Largo recto • Largo curvo • Imagen • Hora descenso • Especie • Acción

		Y permite: <ul style="list-style-type: none"> • Editar los datos • Cancelar la operación
9.	Modifica los datos que necesite y selecciona la opción de editar los mismos.	
10.		Valida los datos
11.		Actualiza los datos de la tortuga.
12.		El caso de uso termina.
Flujos alternos		
*a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*a.1 Regresa al listado de tortugas
		*a.2 Muestra un mensaje de información
		*a.3 El caso de uso termina
4.a Existen datos incompletos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos vacíos.
		5. a.3 Regresa al paso 3 del flujo básico.
4.a Existen datos incorrectos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos incorrectos.
		5. a.3 Regresa al paso 3 del flujo básico.
Sección 2: "Ver tortuga"		
Flujo básico		
	Actor	Sistema
1.	Selecciona la opción de ver los datos de una tortuga	
2.		Muestra los datos de la tortuga seleccionada: <ul style="list-style-type: none"> • Imagen • Playa • Marca • Fibropapiloma

		<ul style="list-style-type: none"> • Ancho recto • Ancho curvo • Largo recto • Largo curvo • Imagen • Hora descenso • Especie • Acción <p>Y permite:</p> <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar la tortuga
3.		El caso de uso termina
Sección 3: “Eliminar tortuga”		
Flujo básico		
	Actor	Sistema
19.	Selecciona la opción de eliminar una tortuga	
20.		Muestra un mensaje de alerta. Y permite: <ul style="list-style-type: none"> • Aceptar. • Cancelar.
21.	Selecciona la opción de aceptar.	
22.		Elimina la tortuga.
23.		Muestra un mensaje de información
24.		El caso de uso termina.
Flujos alternos		
2.a El actor selecciona la opción de cancelar		
	Actor	Sistema
1.		2. a.1 Regresa a la vista anterior.
2.		2. a.2 El caso de uso termina.
Requisitos no funcionales	<p>La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido.</p> <p>Debe notificar al usuario ante la presencia de un error.</p> <p>A los usuarios autorizados se les debe garantizar el acceso a la información.</p>	

Anexo 9: Descripción del caso uso Consultar datos de tortuga

Objetivo	Consultar datos de tortuga.	
Actores	Especialista (Inicia): Consulta datos de tortuga.	
Resumen	El caso de uso se inicia cuando el Especialista desea consultar los datos de una tortuga. El actor introduce los datos de búsqueda y selecciona la opción buscar. El sistema muestra el listado de coincidencias. El actor consulta los datos. El caso de uso termina.	
Precondiciones	Debe haberse generado el escritorio de trabajo del usuario autenticado	
Postcondiciones	Se consultó los datos de la tortuga, coincidente con el criterio de búsqueda.	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
60.	Accede a la opción de consultar datos de la tortuga.	
61.		Brinda la posibilidad de realizar la búsqueda por algunos criterios elementales como: <ul style="list-style-type: none"> • Especie • Marca Y permite: <ul style="list-style-type: none"> • Buscar
62.	Introduce o selecciona los datos que considere para realizar una búsqueda.	
63.	Selecciona la opción de Buscar.	
64.		Consulta los datos y muestra una lista de posibles coincidencias.
65.	Consulta los datos de los elementos de la lista de coincidencias.	
66.		El caso de uso termina
Flujos alternos		
*a El sistema no encuentra ninguna coincidencia.		
	Actor	Sistema
		*.a.1 Muestra un mensaje de información. Y permite: <ul style="list-style-type: none"> • Realizar una nueva búsqueda.
		*.a.2 Regresa al paso 2 del Flujo Básico.
		*.a.3 El caso de uso termina
6. a El actor selecciona la opción de realizar una nueva búsqueda.		

	Actor	Sistema
		7. a.1 Regresa al paso 2 del Flujo Básico.
		7. a.2 El caso de uso termina.
Requisitos no funcionales	<p>La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido.</p> <p>Debe notificar al usuario ante la presencia de un error.</p> <p>A los usuarios autorizados se les debe garantizar el acceso a la información.</p>	

Anexo 10: Descripción del caso uso Gestionar nido

Objetivo	Crear, modificar, ver o eliminar un nido	
Actores	Especialista (Inicia): Crea, modifica, ve y valida los datos de un nido en el sistema.	
Resumen	<p>El caso de uso inicia cuando el especialista decide realizar una acción crear, modificar, ver o eliminar sobre los datos de un nido. Si el especialista selecciona la opción de incluir un nido el sistema le brindará la posibilidad de llenar los datos necesarios y le permitirá cancelar la operación o crearlo. En caso de que desee modificar los datos de un nido, el actor deberá seleccionar el elemento a modificar, el sistema brindará la posibilidad de realizar modificaciones a los campos existentes permitiendo guardar los cambios realizados. Para eliminar un nido el actor selecciona el mismo y lo elimina, terminando el caso de uso.</p>	
Precondiciones	<p>Para modificar los datos de un nido: el nido ha sido seleccionado previamente por el usuario.</p> <p>Para eliminar un nido: el nido ha sido mostrado previamente por el usuario.</p>	
Postcondiciones	Se incluyó, modificó, mostró o eliminó un nido por el usuario.	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
67.	Realiza una acción sobre un nido.	
68.		<p>Permite realizar varias acciones con un nido:</p> <ul style="list-style-type: none"> - Crear un nuevo nido. - Modificar los datos de un nido. Ver Sección 1: “Modificar nido”. - Ver los datos del nido. Ver Sección 2: “Ver nido”.

		<ul style="list-style-type: none"> - Eliminar un nido. Ver Sección 3: “Eliminar nido”. - Crear análisis de neonatos. <u>Ver CU Gestionar análisis de neonatos, Sección crear análisis de neonatos.</u> - Crear análisis de huevos. <u>Ver CU Gestionar análisis de huevos, ver Sección crear análisis de huevos.</u>
69.	Selecciona la opción de crear un nuevo nido.	
70.		<p>Brinda la posibilidad de introducir los datos siguientes:</p> <ul style="list-style-type: none"> • Área protegida • Playa • Recorrido • Número • Fecha de la puesta • Intentos • Distancia a línea de marea • Profundidad • Número de huevos • Banda opaca • Huevos rotos • Tiempo de incubación • Zona de la playa • Especie • Condición • Localización <ul style="list-style-type: none"> ✓ Latitud ✓ Longitud ✓ Baliza ✓ Distancia de baliza • Depredador <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar los datos. • Cancelar la operación.
71.	Introduce los datos y selecciona la opción de guardar los datos del nido	

72.		Valida los datos
73.		Registra los datos del nido
74.		El caso de uso termina
Flujos alternos		
*.a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*.a.1 Regresa al listado de nidos
		*.a.2 Muestra un mensaje de información.
		*.a.3 El caso de uso termina
6.a Existen datos incompletos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos vacíos.
		7. a.3 Regresa al paso 5 del flujo básico.
6.a Existen datos incorrectos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos incorrectos.
		7. a.3 Regresa al paso 5 del flujo básico.
Sección 1: “Modificar nido”		
Flujo básico		
	Actor	Sistema
13.	Selecciona la opción de modificar los datos de un nido	
14.		<p>Muestra los datos de un nido permitiendo modificar los valores del mismo:</p> <ul style="list-style-type: none"> • Área protegida • Playa • Recorrido • Número • Fecha de la puesta • Intentos • Distancia a línea de marea • Profundidad • Número de huevos • Banda opaca • Huevos rotos

		<ul style="list-style-type: none"> • Tiempo de incubación • Zona de la playa • Especie • Condición • Localización <ul style="list-style-type: none"> ✓ Latitud ✓ Longitud ✓ Baliza ✓ Distancia de baliza • Depredador <p>Y permite:</p> <ul style="list-style-type: none"> • Editar los datos • Cancelar la operación
15.	Modifica los datos que necesite y selecciona la opción de editar los mismos.	
16.		Valida los datos
17.		Actualiza los datos del nido.
18.		El caso de uso termina.
Flujos alternos		
*a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*a.1 Regresa al listado de nidos
		*a.2 Muestra un mensaje de información
		*a.3 El caso de uso termina
4.a Existen datos incompletos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos vacíos.
		5. a.3 Regresa al paso 3 del flujo básico.
4.a Existen datos incorrectos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos incorrectos.
		5. a.3 Regresa al paso 3 del flujo básico.

Sección 2: “Ver nido”		
Flujo básico		
	Actor	Sistema
1.	Selecciona la opción de ver los datos de un nido	
2.		<p>Muestra los datos del nido seleccionado:</p> <ul style="list-style-type: none"> • Número • Fecha de la puesta • Intentos • Distancia a línea de marea • Profundidad • Número de huevos • Banda opaca • Huevos rotos • Tiempo de incubación • Zona de la playa • Especie • Condición • Localización <ul style="list-style-type: none"> ✓ Latitud ✓ Longitud ✓ Baliza ✓ Distancia de baliza • Depredador <p>Y permite:</p> <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar el nido
3.		El caso de uso termina
Sección 3: “Eliminar nido”		
Flujo básico		
	Actor	Sistema
25.	Selecciona la opción de eliminar un nido	
26.		<p>Muestra un mensaje de alerta.</p> <p>Y permite:</p> <ul style="list-style-type: none"> • Aceptar. • Cancelar.
27.	Selecciona la opción de aceptar.	

28.		Elimina el nido.
29.		Muestra un mensaje de información
30.		El caso de uso termina.
Flujos alternos		
2.a El actor selecciona la opción de cancelar		
	Actor	Sistema
1.		2. a.1 Regresa a la vista anterior.
2.		2. a.2 El caso de uso termina.
Relaciones	CU Incluidos	Análisis de huevos Análisis de neonatos
Requisitos no funcionales	La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido. Debe notificar al usuario ante la presencia de un error. A los usuarios autorizados se les debe garantizar el acceso a la información.	

Anexo 11: Descripción del caso uso Gestionar análisis de neonatos

Objetivo	Crear, modificar, ver o eliminar el análisis de neonatos
Actores	Especialista (Inicia): Crea, modifica, ve y valida los datos de un análisis de neonatos en el sistema.
Resumen	El caso de uso inicia cuando el especialista decide realizar una acción crear, modificar, ver o eliminar los datos del análisis de neonatos. Si el especialista selecciona la opción de incluir un análisis de neonatos el sistema le brindará la posibilidad de llenar los datos necesarios y le permitirá cancelar la operación o crearlo. En caso de que desee modificar los datos de un análisis de neonatos, el actor deberá seleccionar el elemento a modificar, el sistema brindará la posibilidad de realizar modificaciones a los campos existentes permitiendo guardar los cambios realizados. Para eliminar un análisis de neonatos el actor selecciona el mismo y lo elimina, terminando el caso de uso.
Precondiciones	Para modificar los datos de un análisis de neonatos: el análisis de neonatos ha sido seleccionado previamente por el usuario. Para eliminar un análisis de neonatos: el análisis de neonatos ha sido mostrado previamente por el usuario.
Postcondiciones	Se incluyó, modificó, mostró o eliminó un análisis de neonatos por el usuario.
Flujo de eventos	
Flujo básico	

	Actor	Sistema
75.	Realiza una acción sobre un análisis de neonatos.	
76.		<p>Permite realizar varias acciones con un análisis de neonatos:</p> <ul style="list-style-type: none"> - Crear un nuevo análisis de neonatos. - Modificar los datos de un análisis de neonatos. Ver Sección 1: “Modificar análisis de neonatos”. - Ver los datos del nido. Ver Sección 2: “Ver análisis de neonatos”. - Eliminar un análisis de neonatos. Ver Sección 3: “Eliminar análisis de neonatos”.
77.	Selecciona la opción de crear un nuevo análisis de neonatos.	
78.		<p>Brinda la posibilidad de introducir los datos siguientes:</p> <ul style="list-style-type: none"> • Total de neonatos • Total de atrasados • Total de muertos • Total de normales • Total con restos vitelo • Total con larvas • Ancho recto promedio • Largo recto promedio • Profundidad • Observaciones <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar los datos. • Cancelar la operación.
79.	Introduce los datos y selecciona la opción de guardar los datos del análisis de neonatos	
80.		Valida los datos
81.		Registra los datos del análisis de neonatos
82.		El caso de uso termina

Flujos alternos		
*.a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*.a.1 Regresa al listado de análisis de neonatos
		*.a.2 Muestra un mensaje de información.
		*.a.3 El caso de uso termina
6.a Existen datos incompletos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos vacíos.
		7. a.3 Regresa al paso 5 del flujo básico.
6.a Existen datos incorrectos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos incorrectos.
		7. a.3 Regresa al paso 5 del flujo básico.
Sección 1: “Modificar análisis de neonatos”		
Flujo básico		
	Actor	Sistema
19.	Selecciona la opción de modificar los datos de un análisis de neonatos	
20.		<p>Muestra los datos de un análisis de neonatos permitiendo modificar los valores del mismo:</p> <ul style="list-style-type: none"> • Total de neonatos • Total de atrasados • Total de muertos • Total de normales • Total con restos vitelo • Total con larvas • Ancho recto promedio • Largo recto promedio • Profundidad • Observaciones <p>Y permite:</p>

		<ul style="list-style-type: none"> • Editar los datos • Cancelar la operación
21.	Modifica los datos que necesite y selecciona la opción de editar los mismos.	
22.		Valida los datos
23.		Actualiza los datos del análisis de neonatos.
24.		El caso de uso termina.
Flujos alternos		
*a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*a.1 Regresa al listado de análisis de neonatos
		*a.2 Muestra un mensaje de información
		*a.3 El caso de uso termina
4.a Existen datos incompletos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos vacíos.
		5. a.3 Regresa al paso 3 del flujo básico.
4.a Existen datos incorrectos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos incorrectos.
		5. a.3 Regresa al paso 3 del flujo básico.
Sección 2: “Ver análisis de neonatos”		
Flujo básico		
	Actor	Sistema
1.	Selecciona la opción de ver los datos de un análisis de neonatos	
2.		Muestra los datos del análisis de neonatos seleccionado: <ul style="list-style-type: none"> • Total de neonatos • Total de atrasados • Total de muertos

		<ul style="list-style-type: none"> • Total de normales • Total con restos vitelo • Total con larvas • Ancho recto promedio • Largo recto promedio • Profundidad • Observaciones <p>Y permite:</p> <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar el Nido
3.		El caso de uso termina
Sección 3: “Eliminar análisis de neonatos”		
Flujo básico		
	Actor	Sistema
31.	Selecciona la opción de eliminar un análisis de neonatos	
32.		Muestra un mensaje de alerta. Y permite: <ul style="list-style-type: none"> • Aceptar. • Cancelar.
33.	Selecciona la opción de aceptar.	
34.		Elimina el análisis de neonatos.
35.		Muestra un mensaje de información
36.		El caso de uso termina.
Flujos alternos		
2.a El actor selecciona la opción de cancelar		
	Actor	Sistema
1.		2. a.1 Regresa a la vista anterior.
2.		2. a.2 El caso de uso termina.
Requisitos no funcionales	<p>La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido.</p> <p>Debe notificar al usuario ante la presencia de un error.</p> <p>A los usuarios autorizados se les debe garantizar el acceso a la información.</p>	

Anexo 12: Descripción del caso uso Gestionar análisis de huevos

Objetivo	Crear, modificar, ver o eliminar un análisis de huevos	
Actores	Especialista (Inicia): Crea, modifica, ve y valida los datos de un análisis de huevos en el sistema.	
Resumen	El caso de uso inicia cuando el especialista decide realizar una acción crear, modificar, ver o eliminar los datos del análisis de huevos. Si el especialista selecciona la opción de incluir un análisis de huevos el sistema le brindará la posibilidad de llenar los datos necesarios y le permitirá cancelar la operación o crearlo. En caso de que desee modificar los datos de un análisis de huevos, el actor deberá seleccionar el elemento a modificar, el sistema brindará la posibilidad de realizar modificaciones a los campos existentes permitiendo guardar los cambios realizados. Para eliminar un análisis de huevos el actor selecciona el mismo y lo elimina, terminando el caso de uso.	
Precondicion es	Para modificar los datos de un análisis de huevos: el análisis de huevos ha sido seleccionado previamente por el usuario. Para Eliminar un análisis de huevos: el análisis de huevos ha sido mostrado previamente por el usuario.	
Postcondicio nes	Se incluyó, modificó, mostró o eliminó un análisis de huevos por el usuario	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
83.	Realiza una acción sobre un análisis de huevos.	
84.		Permite realizar varias acciones con un análisis de huevos: <ul style="list-style-type: none"> - Crear un nuevo análisis de huevos. - Modificar los datos de un análisis de huevos. Ver Sección 1: “Modificar análisis de huevos”. - Ver los datos del análisis de huevos. Ver Sección 2: “Ver análisis de huevos”. - Eliminar un análisis de huevos. Ver Sección 3: “Eliminar análisis de huevos”.
85.	Selecciona la opción de crear un nuevo análisis de huevos.	
86.		Brinda la posibilidad de introducir los datos siguientes:

		<ul style="list-style-type: none"> • Total de cascarones • Total de huevos con desarrollo embrionario • Total de huevos sin desarrollo embrionario • Total de huevos • Período de incubación • Total de embriones con anomalías • Total de depredados por hormigas • Total de depredados por cangrejos • Total de depredados por larvas • Total de depredados por hongos <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar los datos. • Cancelar la operación.
87.	Introduce los datos y selecciona la opción de guardar los datos del análisis de huevos	
88.		Valida los datos
89.		Registra los datos del análisis de huevos
90.		El caso de uso termina
Flujos alternos		
*.a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*.a.1 Regresa al listado de análisis de huevos
		*.a.2 Muestra un mensaje de información.
		*.a.3 El caso de uso termina
6.a Existen datos incompletos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos vacíos.
		7. a.3 Regresa al paso 5 del flujo básico.
6.a Existen datos incorrectos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los

		campos incorrectos.
		7. a.3 Regresa al paso 5 del flujo básico.
Sección 1: “Modificar análisis de huevos”		
Flujo básico		
	Actor	Sistema
25.	Selecciona la opción de modificar los datos de un análisis de huevos	
26.		<p>Muestra los datos de un análisis de huevos permitiendo modificar los valores del mismo:</p> <ul style="list-style-type: none"> • Total de cascarones • Total de huevos con desarrollo embrionario • Total de huevos sin desarrollo embrionario • Total de huevos • Período de incubación • Total de embriones con anomalías • Total de depredados por hormigas • Total de depredados por cangrejos • Total de depredados por larvas • Total de depredados por hongos <p>Y permite:</p> <ul style="list-style-type: none"> • Editar los datos • Cancelar la operación
27.	Modifica los datos que necesite y selecciona la opción de editar los mismos.	
28.		Valida los datos
29.		Actualiza los datos del análisis de huevos.
30.		El caso de uso termina.
Flujos alternos		
*a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*a.1 Regresa al listado de análisis de huevos
		*a.2 Muestra un mensaje de información
		*a.3 El caso de uso termina

4.a Existen datos incompletos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos vacíos.
		5. a.3 Regresa al paso 3 del flujo básico.
4.a Existen datos incorrectos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos incorrectos.
		5. a.3 Regresa al paso 3 del flujo básico.
Sección 2: “Ver análisis de huevos”		
Flujo básico		
	Actor	Sistema
1.	Selecciona la opción de ver los datos de un análisis de huevos	
2.		<p>Muestra los datos del análisis de huevos seleccionado:</p> <ul style="list-style-type: none"> • Total de cascarones • Total de huevos con desarrollo embrionario • Total de huevos sin desarrollo embrionario • Total de huevos • Período de incubación • Total de embriones con anomalías • Total de depredados por hormigas • Total de depredados por cangrejos • Total de depredados por larvas • Total de depredados por hongos <p>Y permite:</p> <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar el análisis de huevos
3.		El caso de uso termina
Sección 3: “Eliminar análisis de huevos”		

Flujo básico		
	Actor	Sistema
37.	Selecciona la opción de eliminar un análisis de huevos	
38.		Muestra un mensaje de alerta. Y permite: <ul style="list-style-type: none"> • Aceptar. • Cancelar.
39.	Selecciona la opción de aceptar.	
40.		Elimina el análisis de huevos.
41.		Muestra un mensaje de información
42.		El caso de uso termina.
Flujos alternos		
2.a El actor selecciona la opción de cancelar		
	Actor	Sistema
1.		2. a.1 Regresa a la vista anterior.
2.		2. a.2 El caso de uso termina.
Requisitos no funcionales	<p>La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido.</p> <p>Debe notificar al usuario ante la presencia de un error.</p> <p>A los usuarios autorizados se les debe garantizar el acceso a la información.</p>	

Anexo 13: Descripción del caso uso Gestionar rastro

Objetivo	Crear, modificar, ver o eliminar un rastro
Actores	Especialista (Inicia): Crea, modifica, ve y valida los datos de un rastro en el sistema.
Resumen	El caso de uso inicia cuando el especialista decide realizar una acción sobre un rastro, esta puede ser la de crear, modificar, ver o eliminar los datos del mismo. Si el especialista selecciona la opción de incluir un rastro el sistema le brindará la posibilidad de llenar los datos necesarios y le permitirá cancelar la operación o crearlo. En caso de que desee modificar los datos del rastro, el actor deberá seleccionar el elemento a modificar, el sistema brindará la posibilidad de realizar modificaciones a los campos existentes permitiendo guardar los cambios realizados. Para eliminar un rastro el actor selecciona el mismo y lo elimina, terminando el caso de uso.
Precondicion	El usuario ha sido validado.

es	Para modificar los datos de un rastro: el rastro ha sido seleccionado previamente por el actor. Para eliminar un rastro: el rastro ha sido mostrado previamente por el actor.	
Postcondiciones	Se incluyó, modificó, mostró o eliminó un rastro por el actor	
Flujo de eventos		
Flujo básico		
	Actor	Sistema
91.	Realiza una acción sobre un rastro.	
92.		Permite realizar varias acciones con un rastro: <ul style="list-style-type: none"> - Crear un nuevo rastro. - Modificar los datos de un rastro. Ver Sección 1: “Modificar rastro”. - Ver los datos del rastro. Ver Sección 2: “Ver rastro”. - Eliminar un rastro. Ver Sección 3: “Eliminar rastro”.
93.	Selecciona la opción de crear un nuevo rastro	
94.		Brinda la posibilidad de introducir los datos siguientes: <ul style="list-style-type: none"> • Área protegida. • Playa • Recorrido • Número • Ancho • Tipo • Dirección • Especie • Condición • Localización <ul style="list-style-type: none"> • Latitud • Longitud • Baliza • Distancia a la baliza <p>Y permite:</p>

		<ul style="list-style-type: none"> • Guardar los datos. • Cancelar la operación.
95.	Introduce los datos y selecciona la opción de guardar los datos del rastro	
96.		Valida los datos
97.		Registra los datos del rastro
98.		El caso de uso termina
Flujos alternos		
*.a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*.a.1 Regresa al listado de rastro
		*.a.2 Muestra un mensaje de información.
		*.a.3 El caso de uso termina
6.a Existen datos incompletos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos vacíos.
		7. a.3 Regresa al paso 4 del flujo básico.
6.a Existen datos incorrectos		
	Actor	Sistema
		7. a.1 Muestra un mensaje de información.
		7. a.2 Muestra un indicador sobre los campos incorrectos.
		7. a.3 Regresa al paso 4 del flujo básico.
Sección 1: "Modificar rastro"		
Flujo básico		
	Actor	Sistema
31.	Selecciona la opción de modificar los datos de un rastro	
32.		Muestra el Rastro permitiendo modificar los valores del mismo: <ul style="list-style-type: none"> • Área protegida. • Playa • Recorrido • Número • Ancho • Tipo • Dirección

		<ul style="list-style-type: none"> • Especie • Condición • Localización <ul style="list-style-type: none"> • Latitud • Longitud • Baliza • Distancia a la baliza <p>Y permite:</p> <ul style="list-style-type: none"> • Editar los datos • Cancelar la operación
33.	Modifica los datos que necesite y selecciona la opción de editar los mismos.	
34.		Valida los datos
35.		Actualiza los datos del rastro.
36.		El caso de uso termina.
Flujos alternos		
*a El actor selecciona la opción de cancelar		
	Actor	Sistema
		*a.1 Regresa al listado de rastro
		*a.2 Muestra un mensaje de información
		*a.3 El caso de uso termina
4.a Existen datos incompletos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos vacíos.
		5. a.3 Regresa al paso 4 del flujo básico.
4.a Existen datos incorrectos		
	Actor	Sistema
		5. a.1 Muestra un mensaje de información.
		5. a.2 Muestra un indicador sobre los campos incorrectos.
		5. a.3 Regresa al paso 4 del flujo básico.
Sección 2: “Ver rastro”		
Flujo básico		
	Actor	Sistema

1.	Selecciona la opción de ver los datos de un rastro	
2.		<p>Muestra los datos del rastro seleccionado:</p> <ul style="list-style-type: none"> • Área protegida. • Playa • Recorrido • Número • Ancho • Tipo • Dirección • Especie • Condición • Localización <p>Y permite:</p> <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar el rastro
3.		El caso de uso termina

Sección 3: “Eliminar rastro”

Flujo básico

	Actor	Sistema
43.	Selecciona la opción de eliminar un rastro	
44.		<p>Muestra un mensaje de alerta.</p> <p>Y permite:</p> <ul style="list-style-type: none"> • Aceptar. • Cancelar.
45.	Selecciona la opción de aceptar.	
46.		Elimina el rastro.
47.		Muestra un mensaje de información
48.		El caso de uso termina.

Flujos alternos

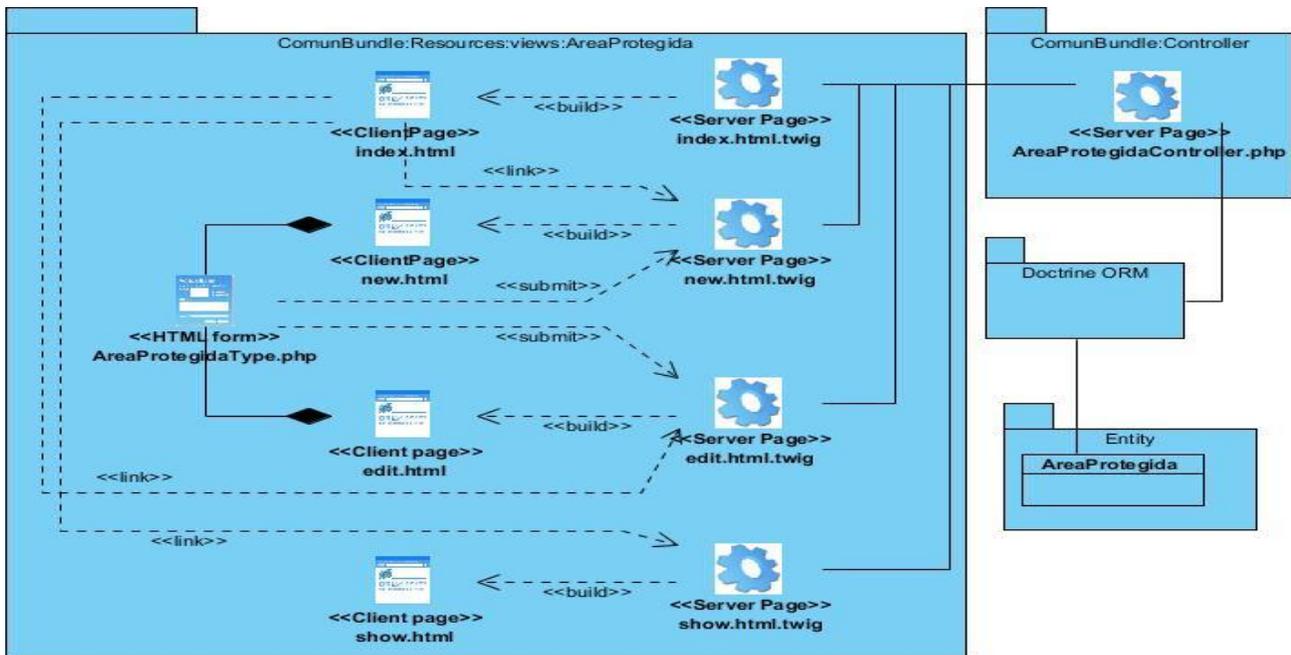
2.a El actor selecciona la opción de cancelar

	Actor	Sistema
1.		2. a.1 Regresa a la vista anterior.
2.		2. a.2 El caso de uso termina.

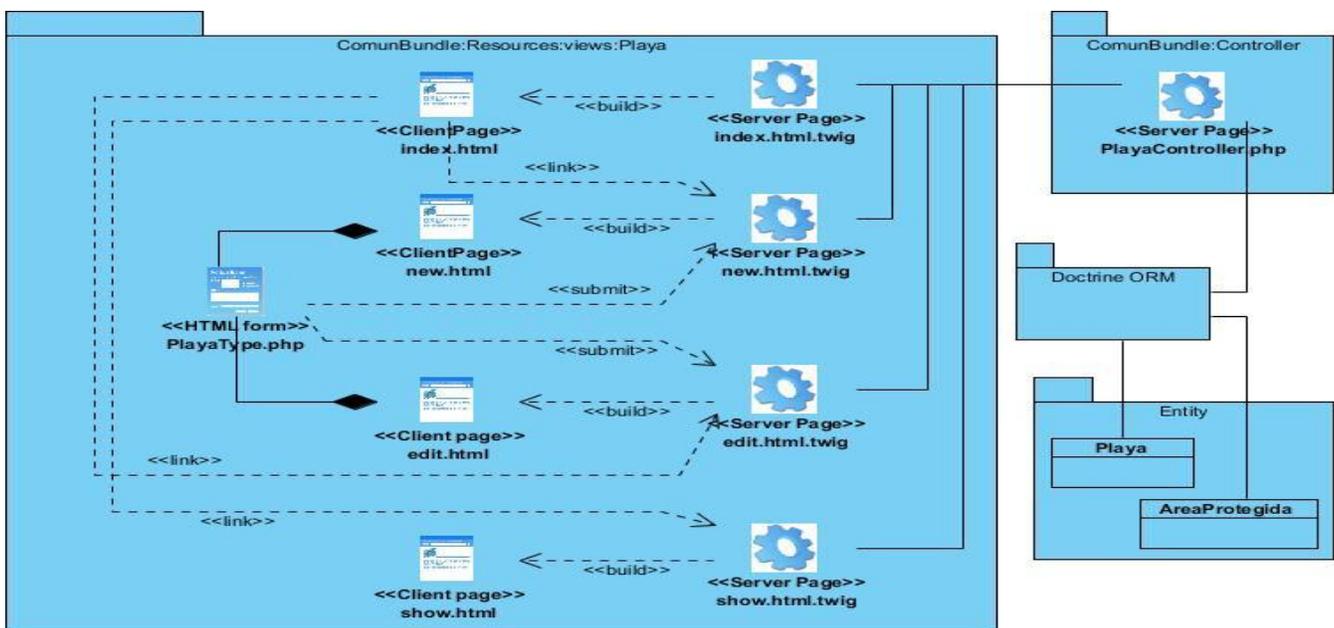
Requisitos no funcionales	La información aparecerá correctamente organizada de forma, que el usuario no se encuentre perdido.
----------------------------------	---

Debe notificar al usuario ante la presencia de un error.
 A los usuarios autorizados se les debe garantizar el acceso a la información.

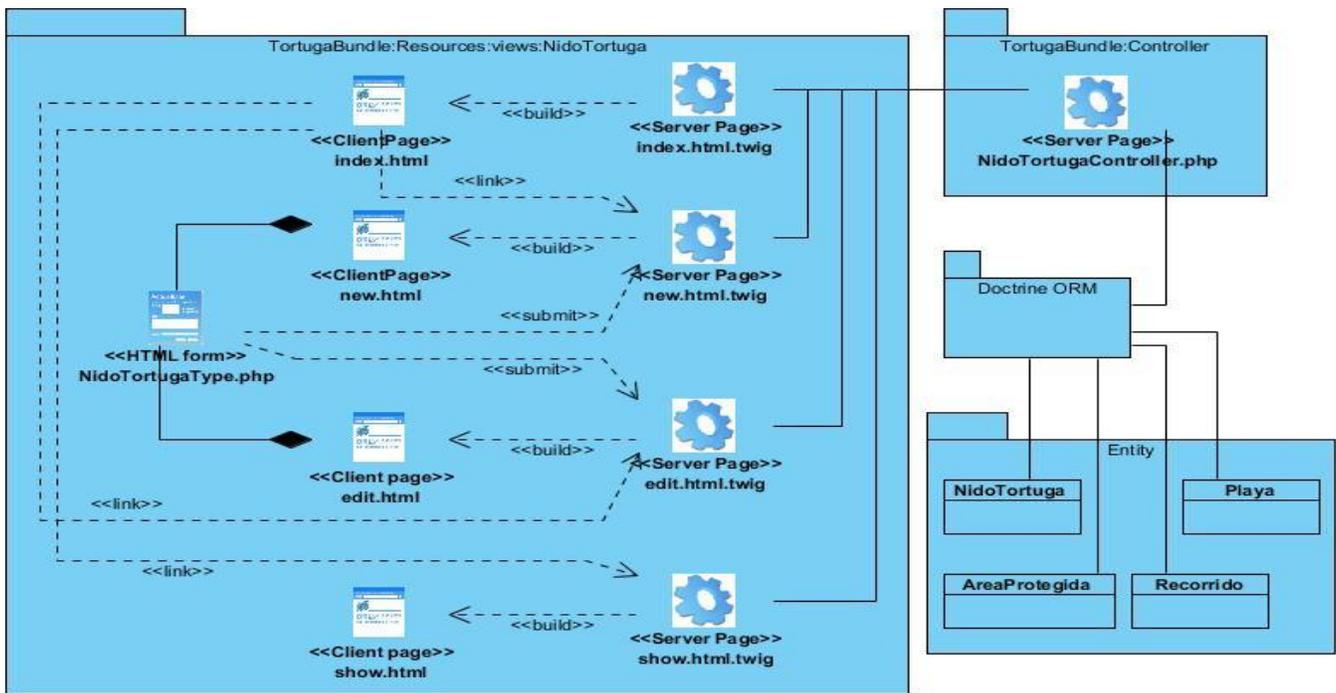
Anexo 14: Diagrama de clases del diseño del caso de uso Gestionar área protegida



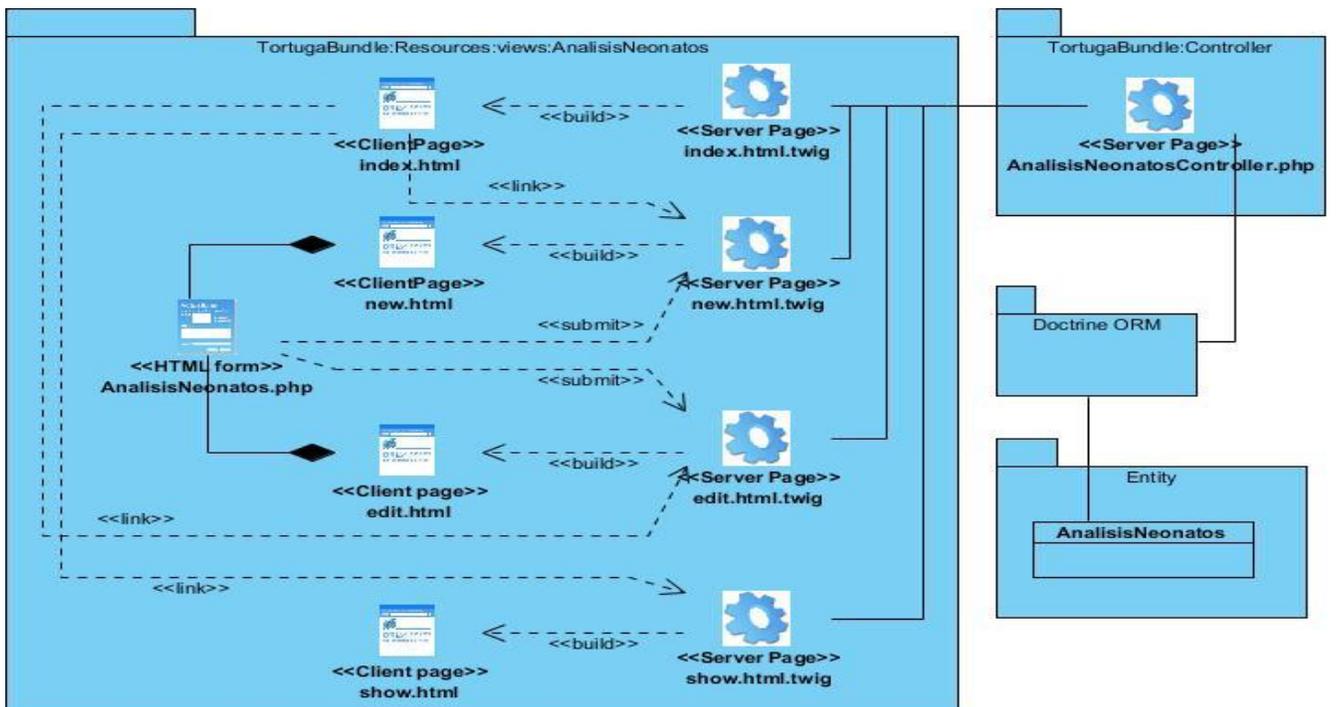
Anexo 15: Diagrama de clases del diseño del caso de uso Gestionar playa



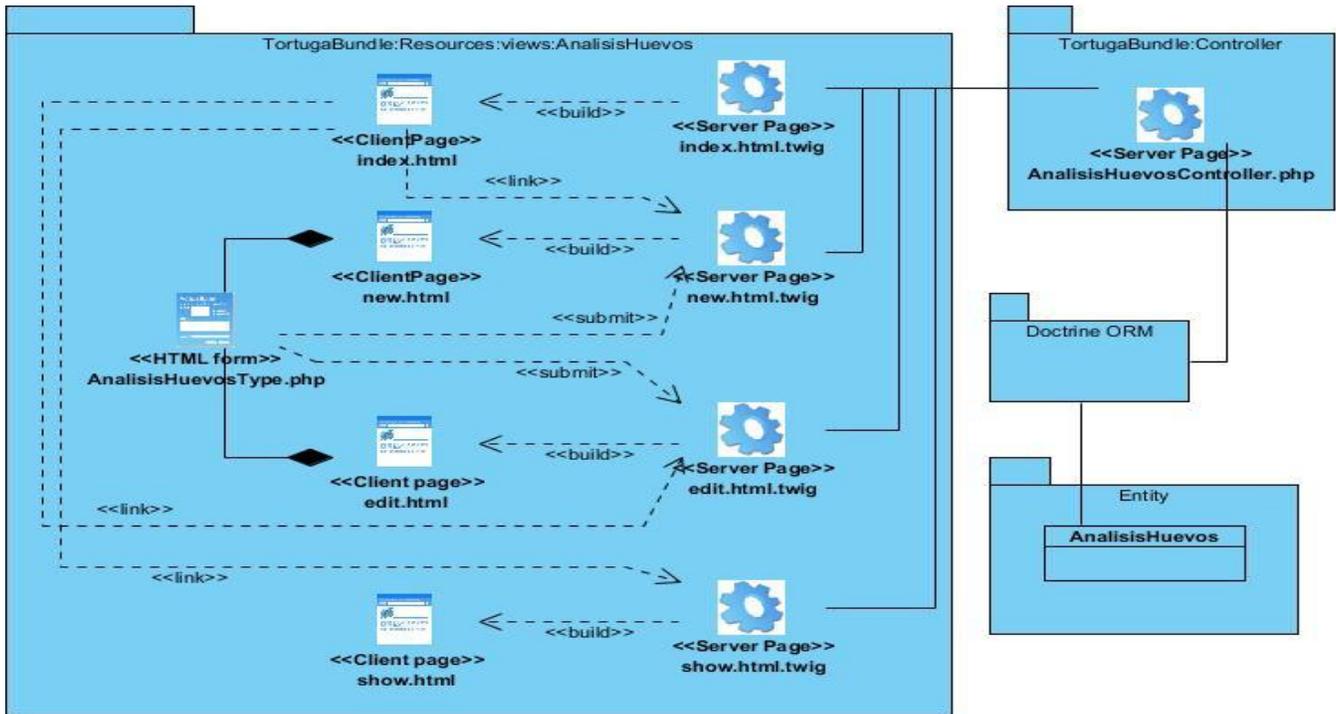
Anexo 16: Diagrama de clases del diseño del caso de uso Gestionar nido



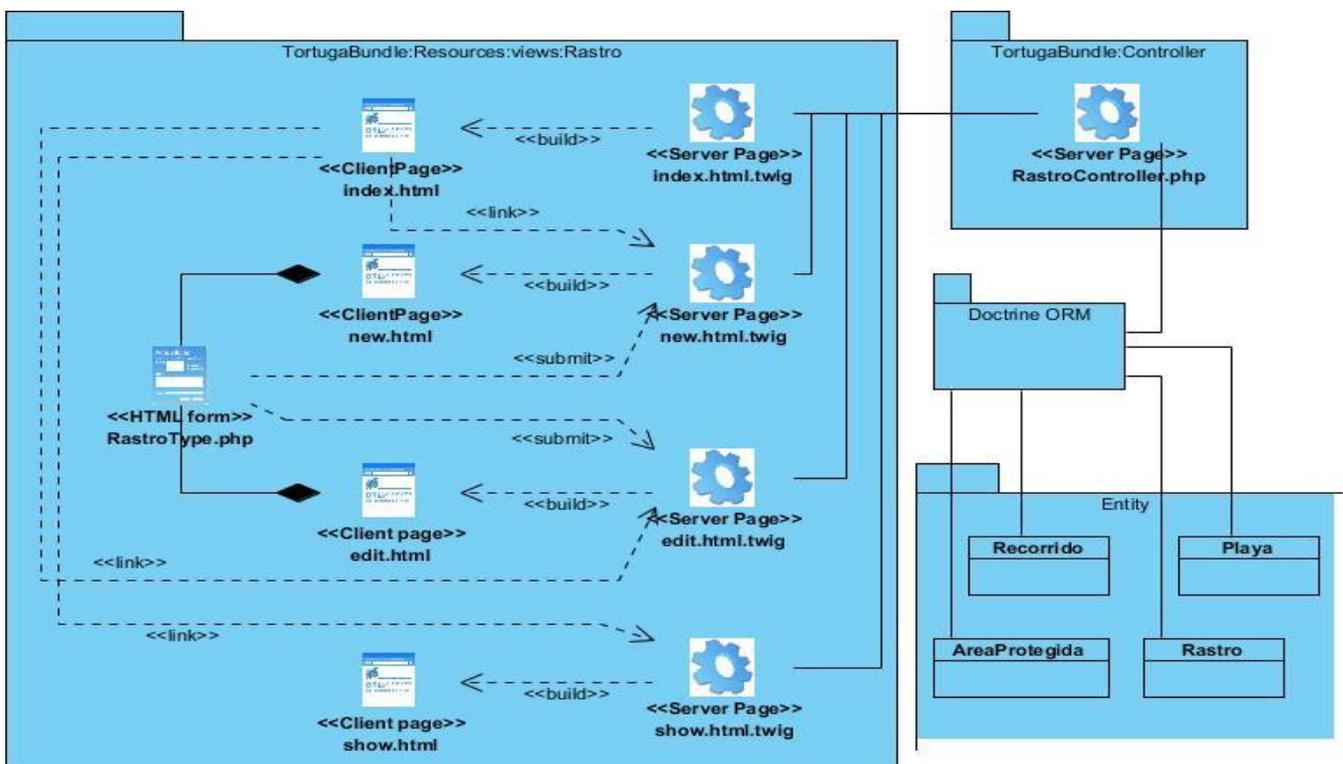
Anexo 17: Diagrama de clases del diseño del caso de uso Gestionar análisis de neonatos



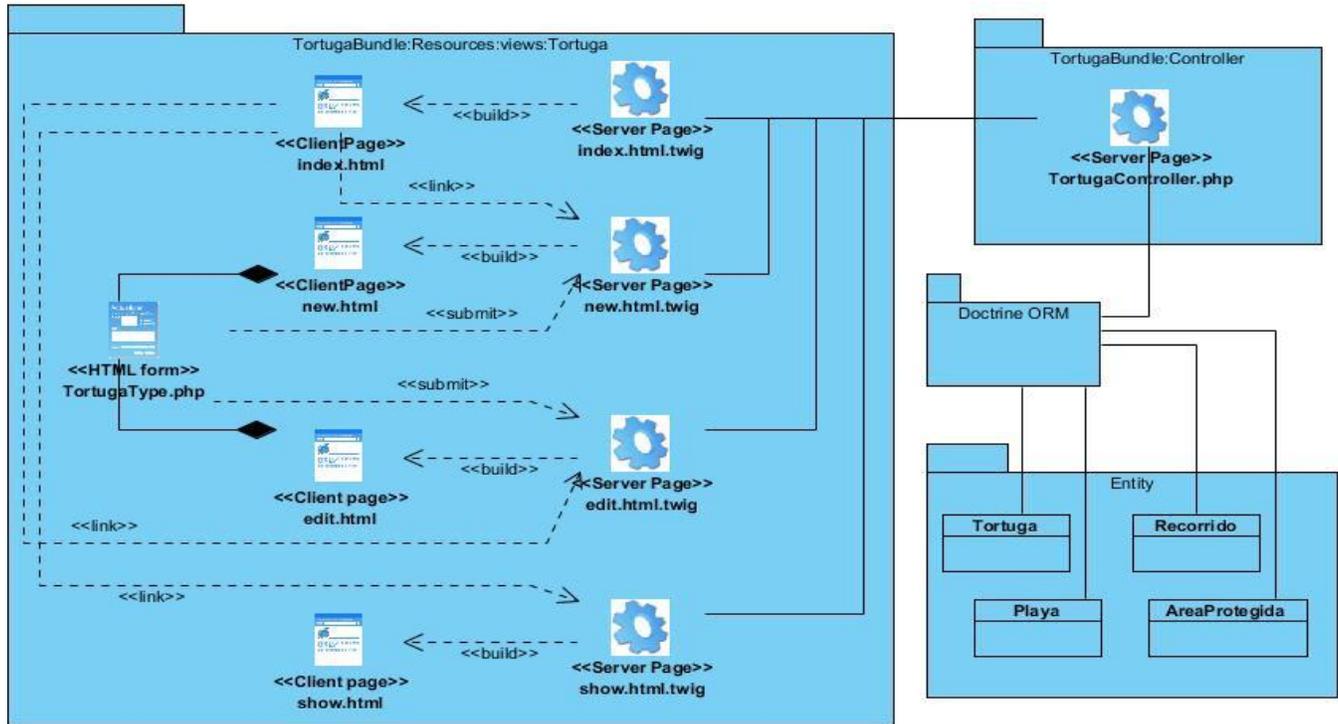
Anexo 18: Diagrama de clases del diseño del caso de uso Gestionar análisis de huevos



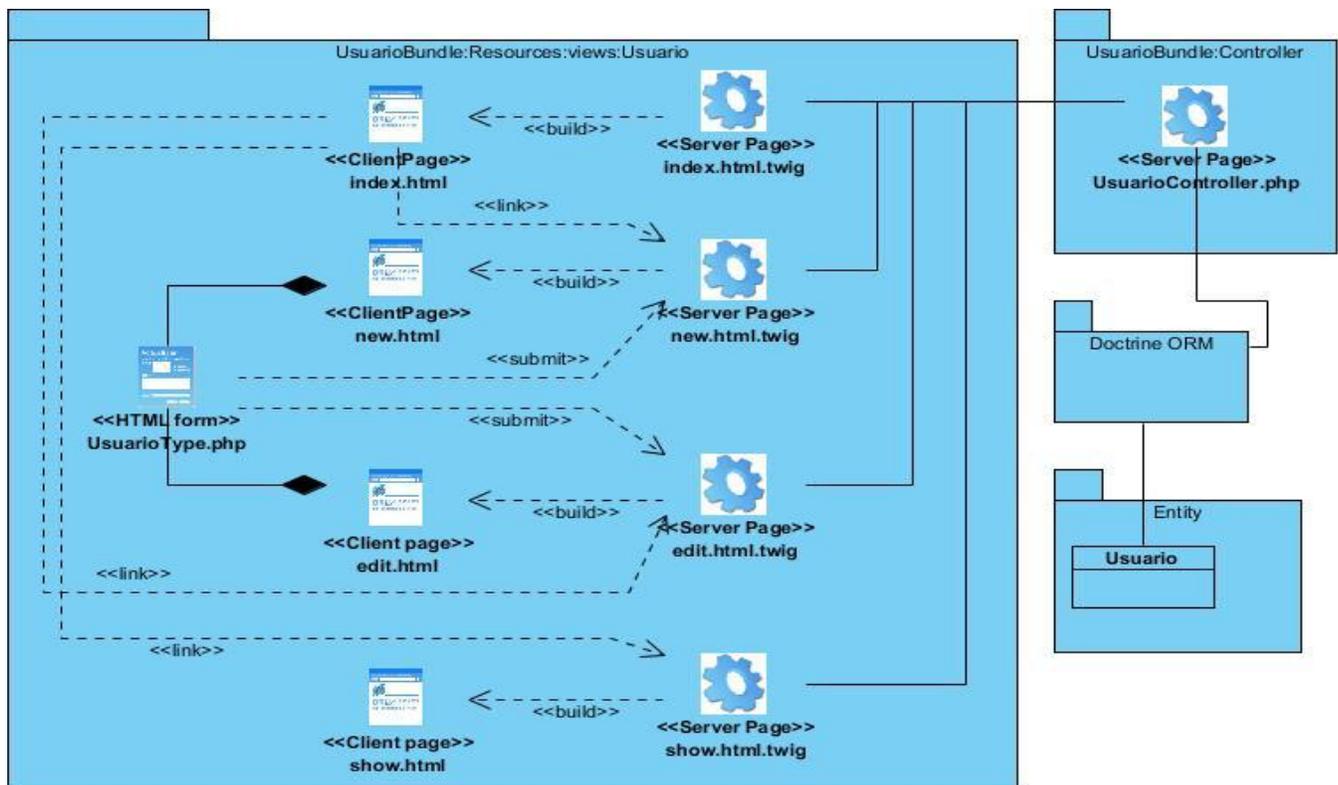
Anexo 19: Diagrama de clases del diseño del caso de uso Gestionar rastro



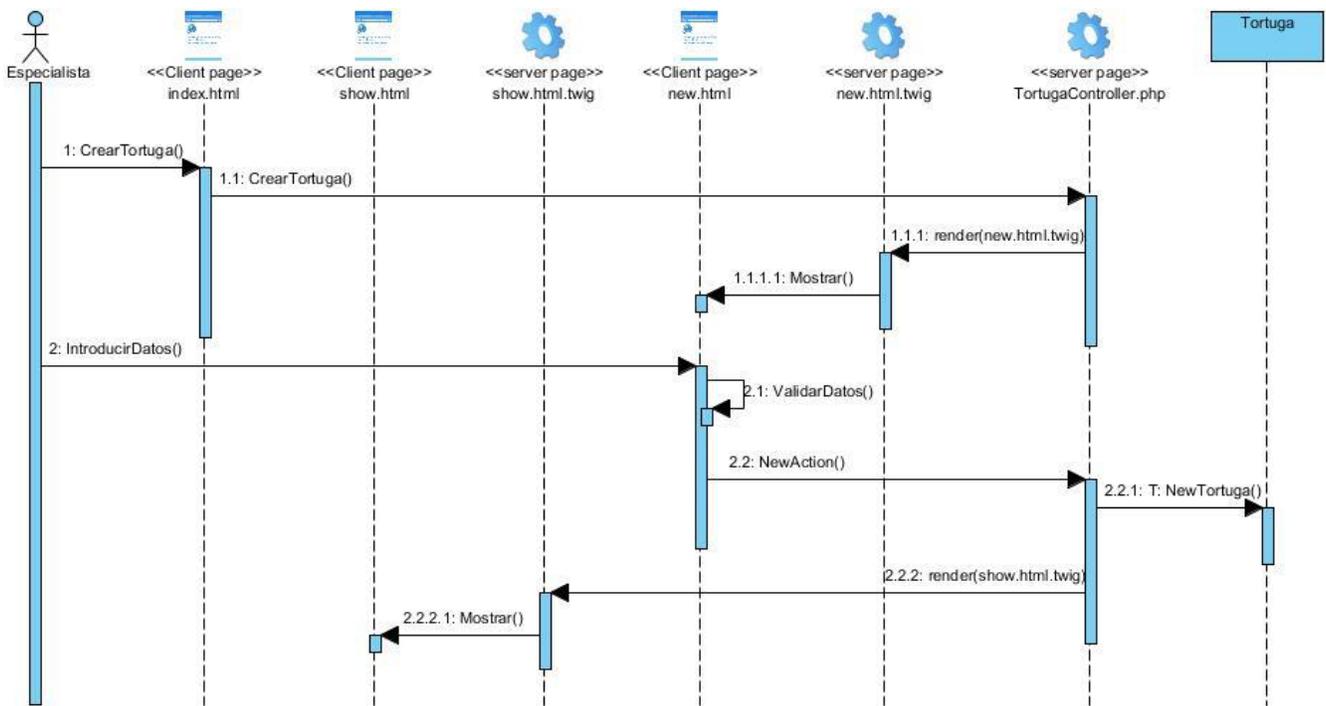
Anexo 20: Diagrama de clases del diseño del caso de uso Gestionar tortuga



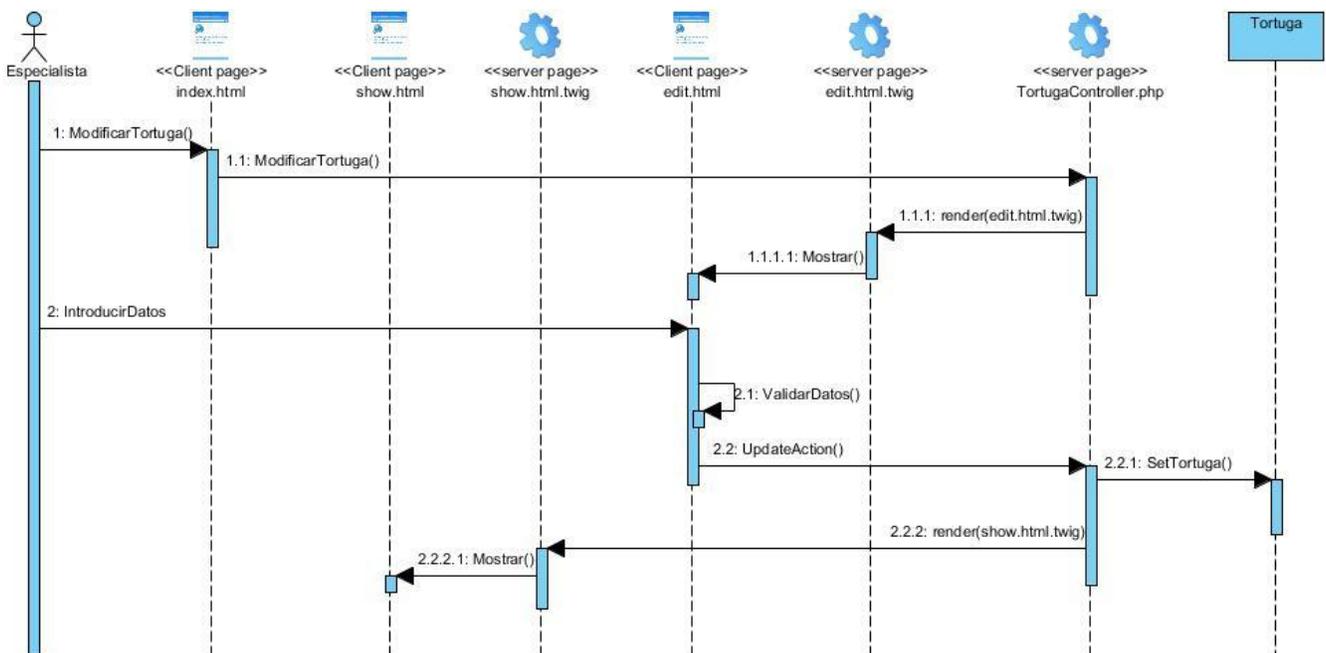
Anexo 21: Diagrama de clases del diseño del caso de uso Gestionar usuario



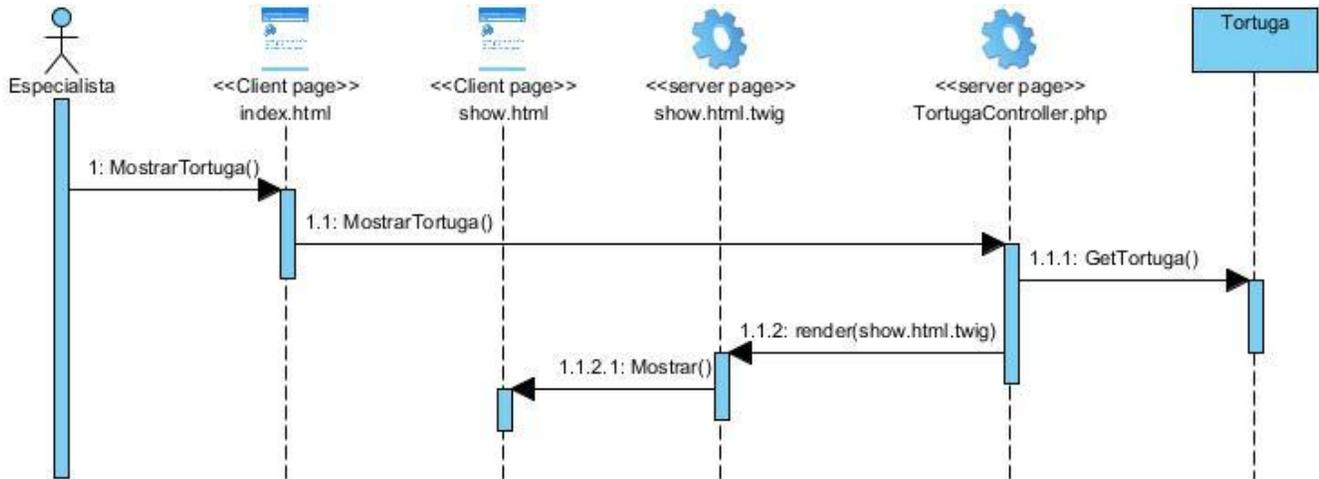
Anexo 22: Diagrama de secuencia del caso de uso Gestionar tortuga, sección Crear tortuga.



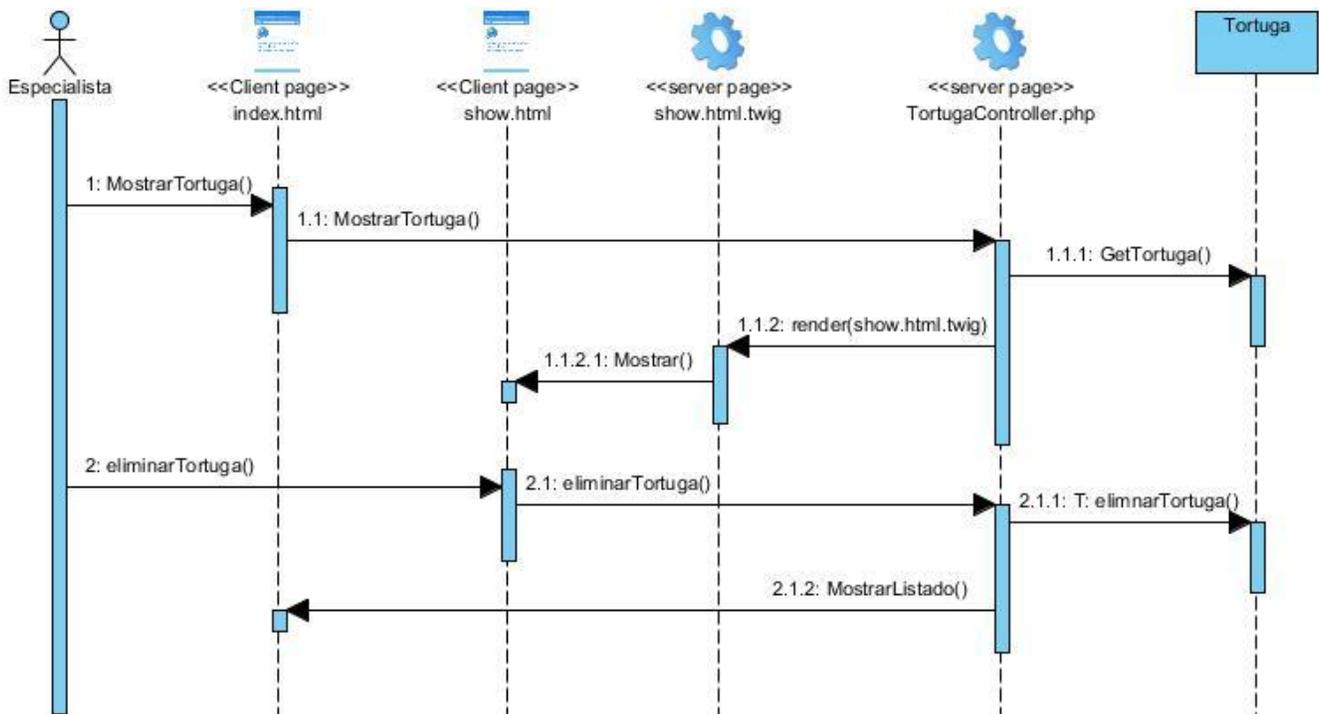
Anexo 23: Diagrama de secuencia del caso de uso Gestionar tortuga, sección Modificar tortuga.



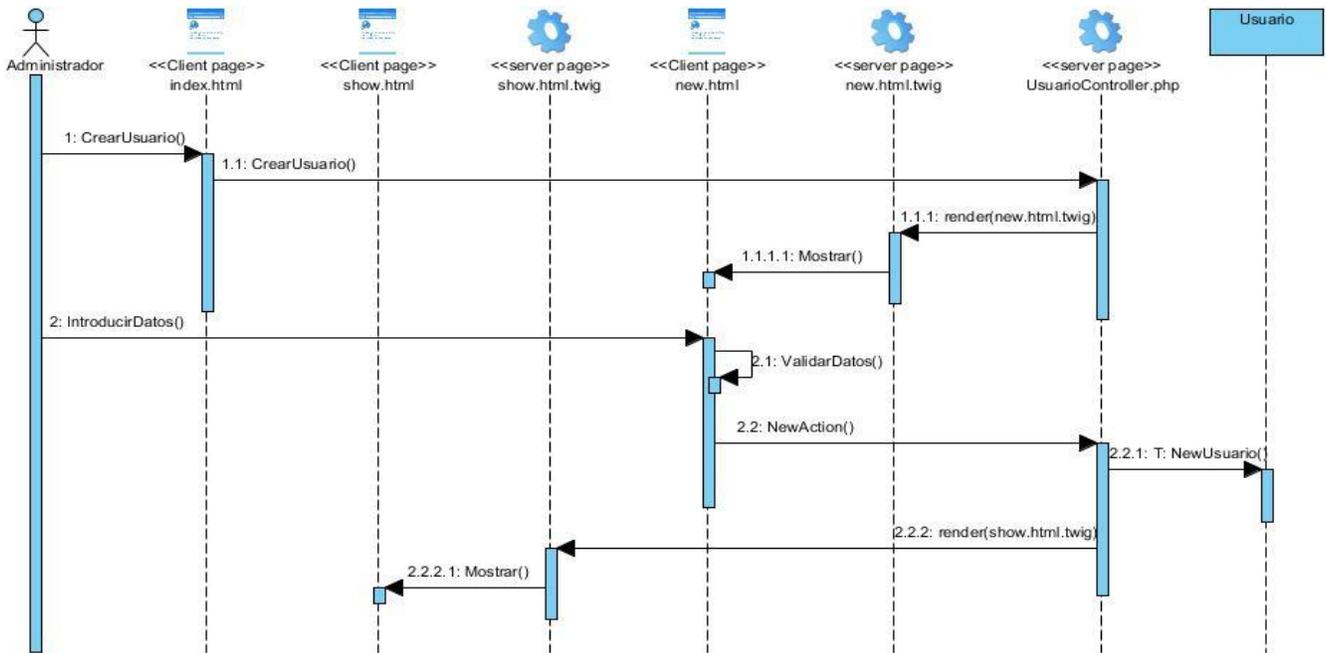
Anexo 24: Diagrama de secuencia del caso de uso Gestionar tortuga, sección Mostrar tortuga.



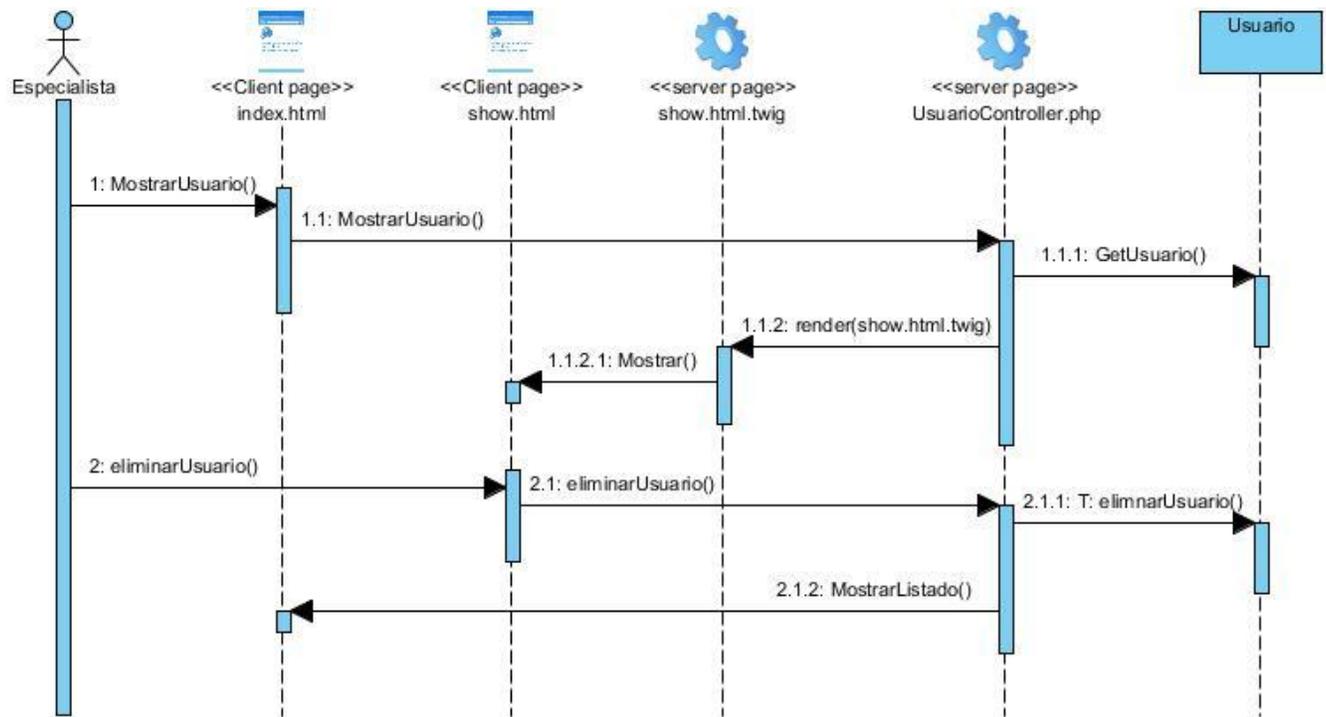
Anexo 25: Diagrama de secuencia del caso de uso Gestionar tortuga, sección Eliminar tortuga.



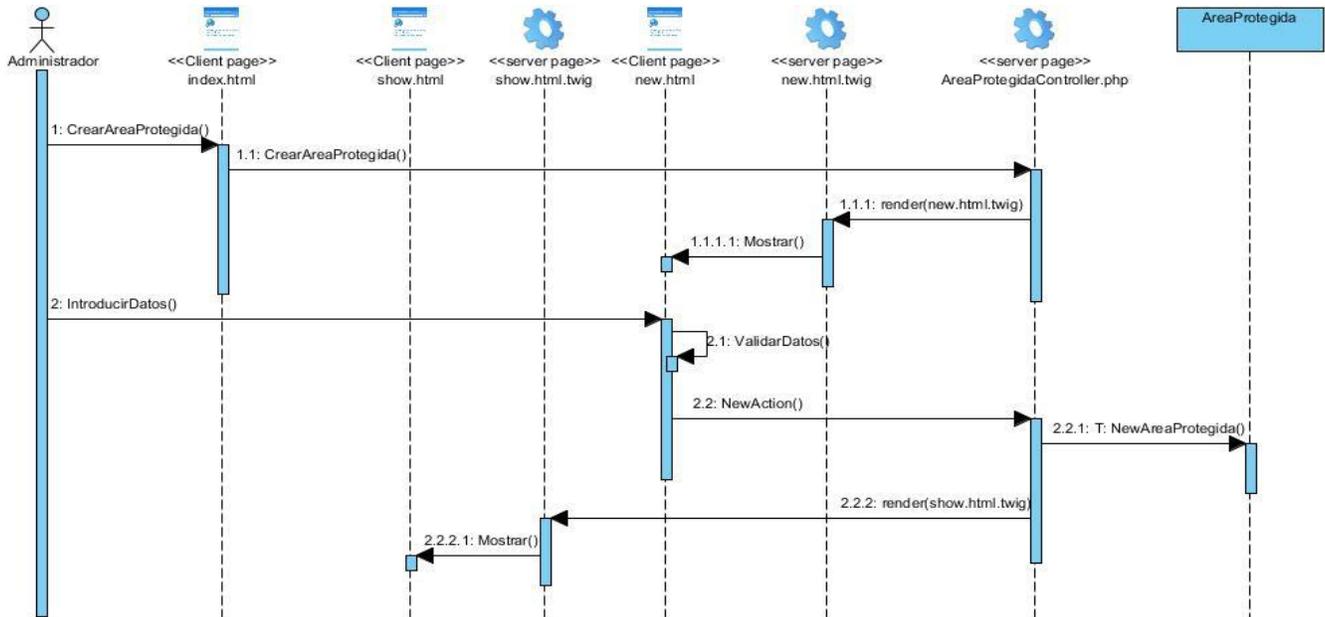
Anexo 26: Diagrama de secuencia del caso de uso Gestionar usuario, sección Crear usuario.



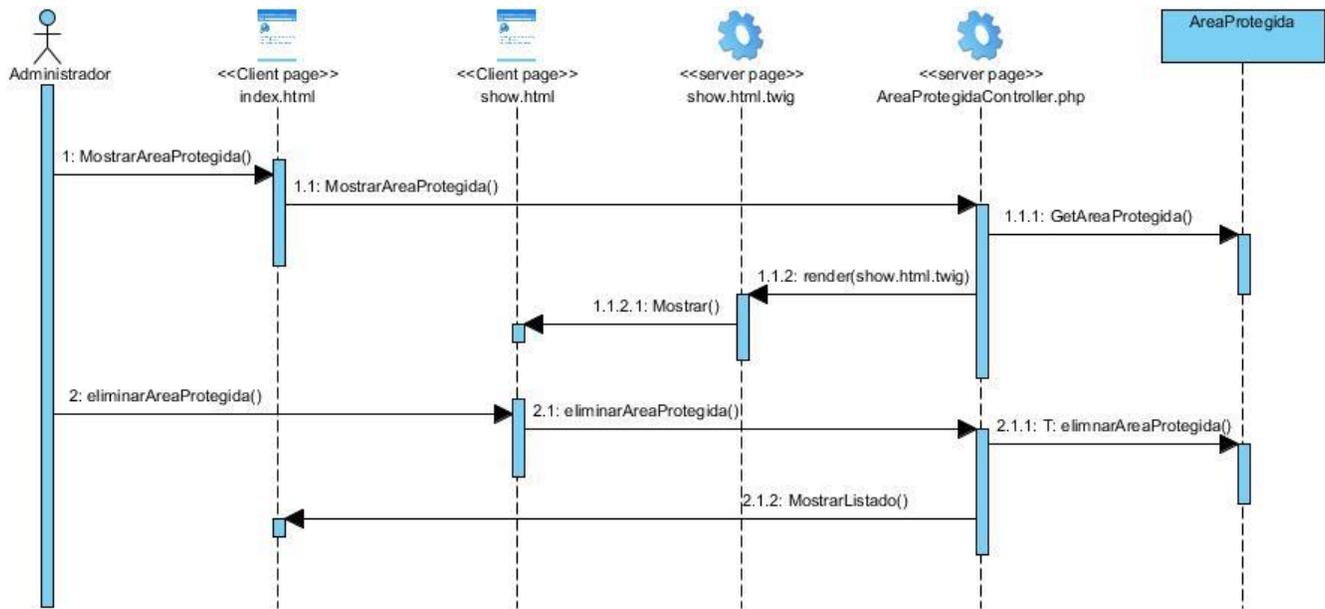
Anexo 27: Diagrama de secuencia del caso de uso Gestionar usuario, sección Eliminar usuario.



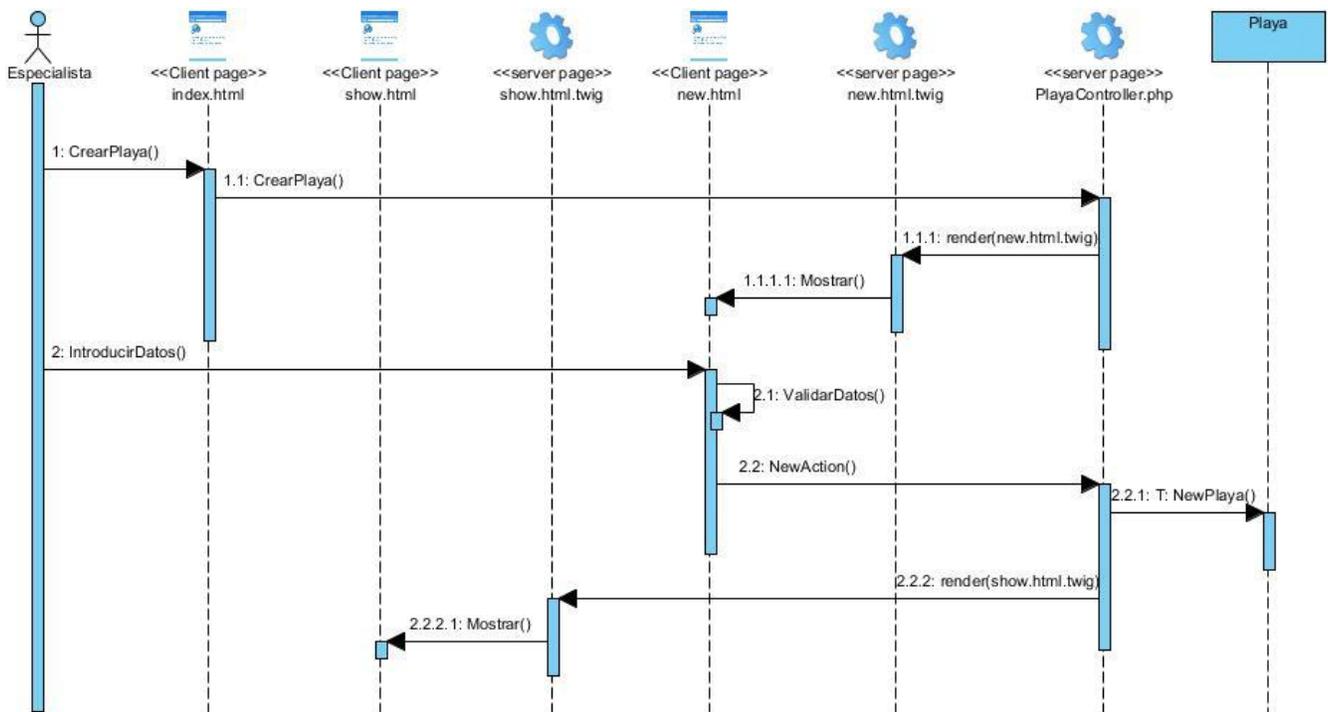
Anexo 28: Diagrama de secuencia del caso de uso Gestionar área protegida, sección Crear área protegida.



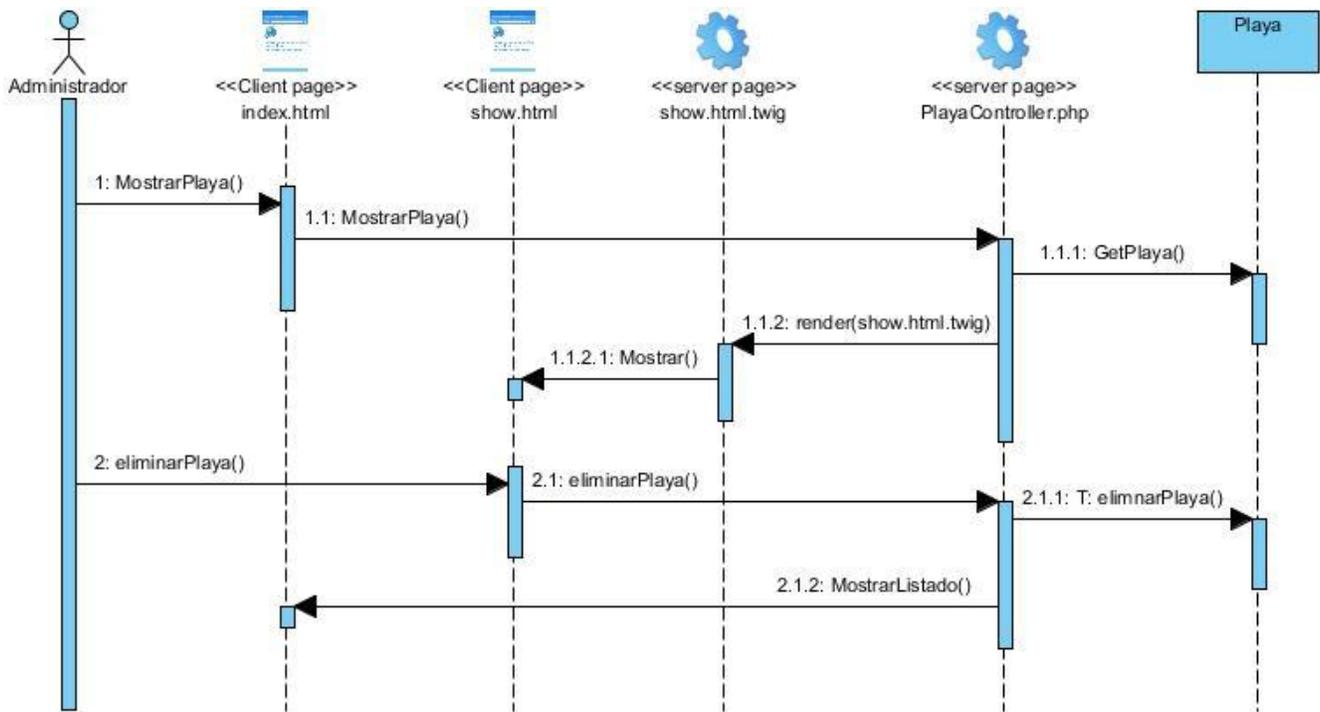
Anexo 29: Diagrama de secuencia del caso de uso Gestionar área protegida, sección Eliminar área protegida.



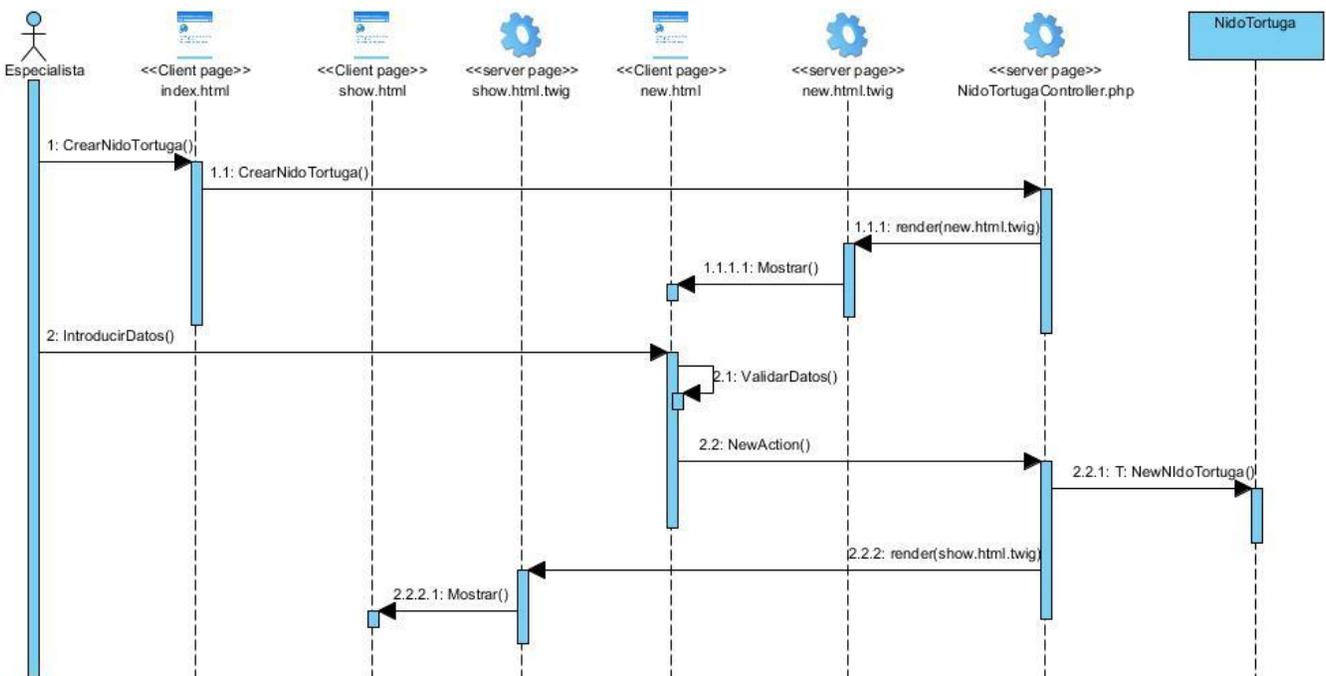
Anexo 30: Diagrama de secuencia del caso de uso Gestionar playa, sección Crear playa.



Anexo 31: Diagrama de secuencia del caso de uso Gestionar playa, sección Eliminar playa.

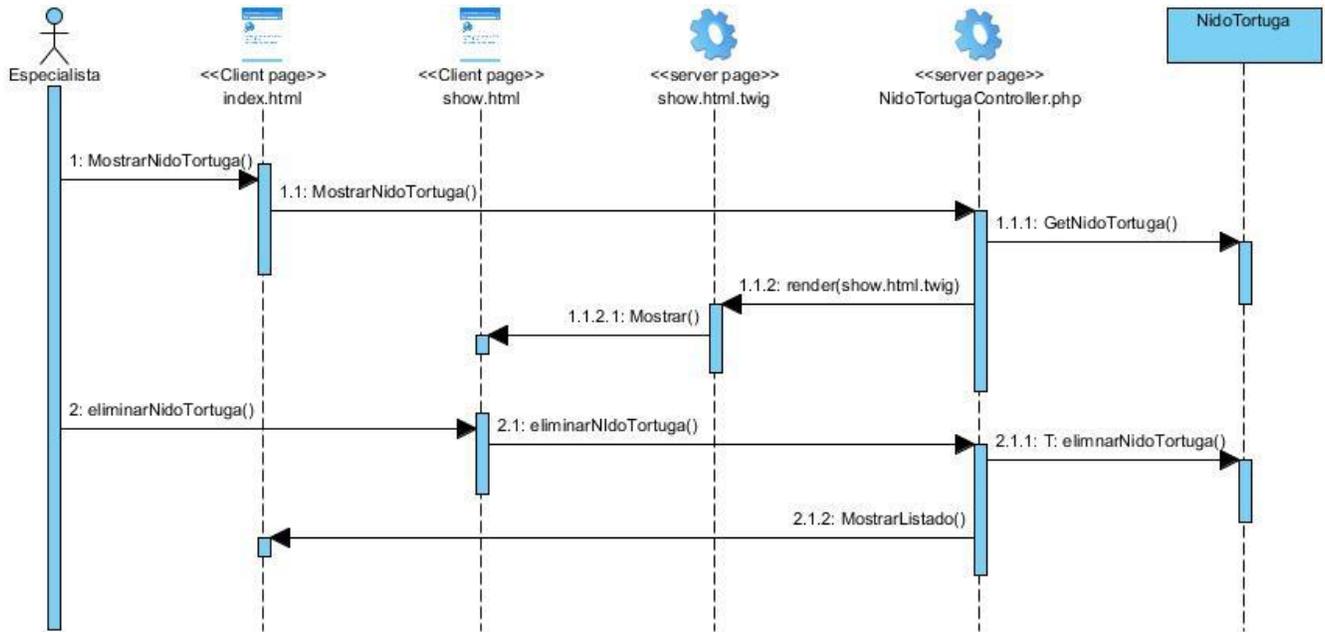


Anexo 32: Diagrama de secuencia del caso de uso Gestionar nido de tortuga, sección Crear nido de tortuga.

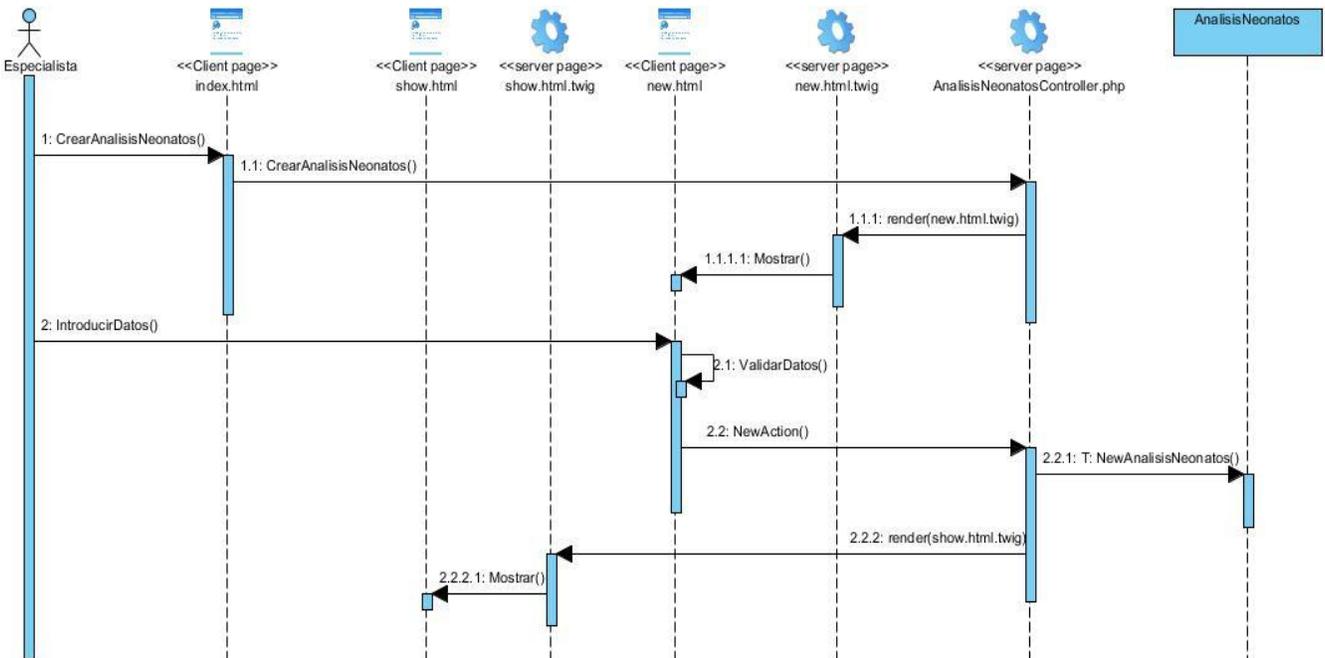


Anexo 33: Diagrama de secuencia del caso de uso Gestionar nido de tortuga, sección

Eliminar nido de tortuga.

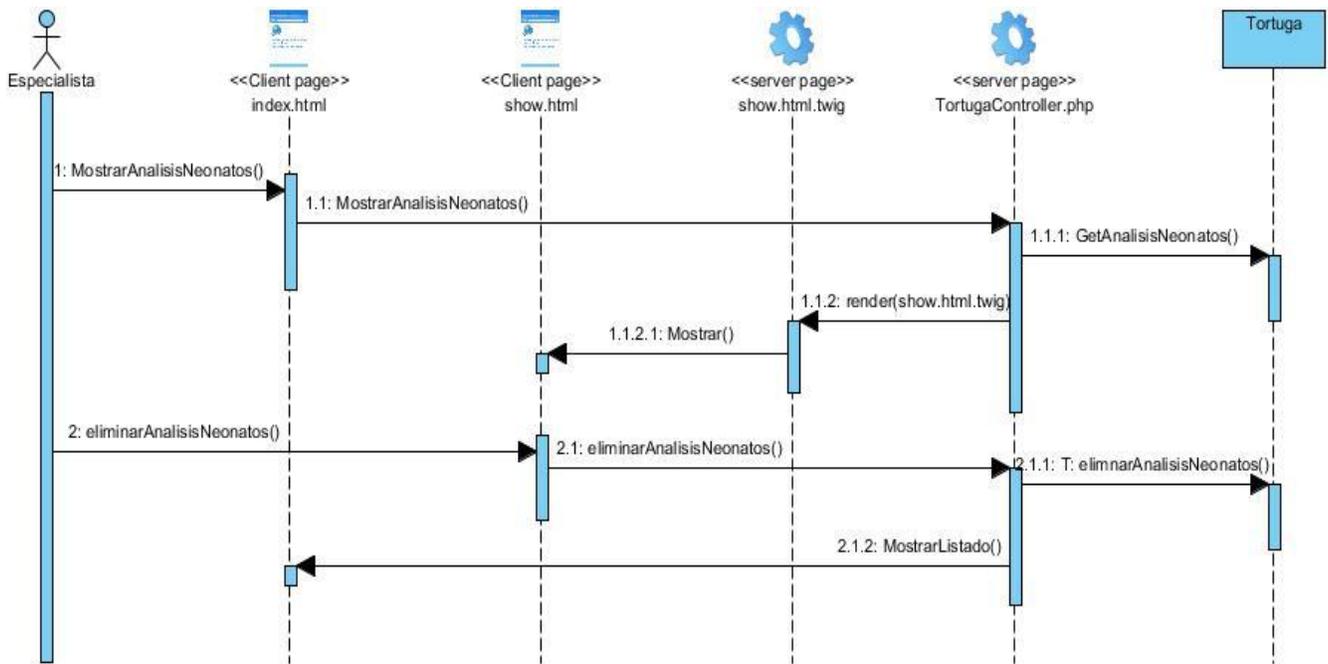


Anexo 34: Diagrama de secuencia del caso de uso Gestionar análisis de neonatos, sección Crear análisis de neonatos.

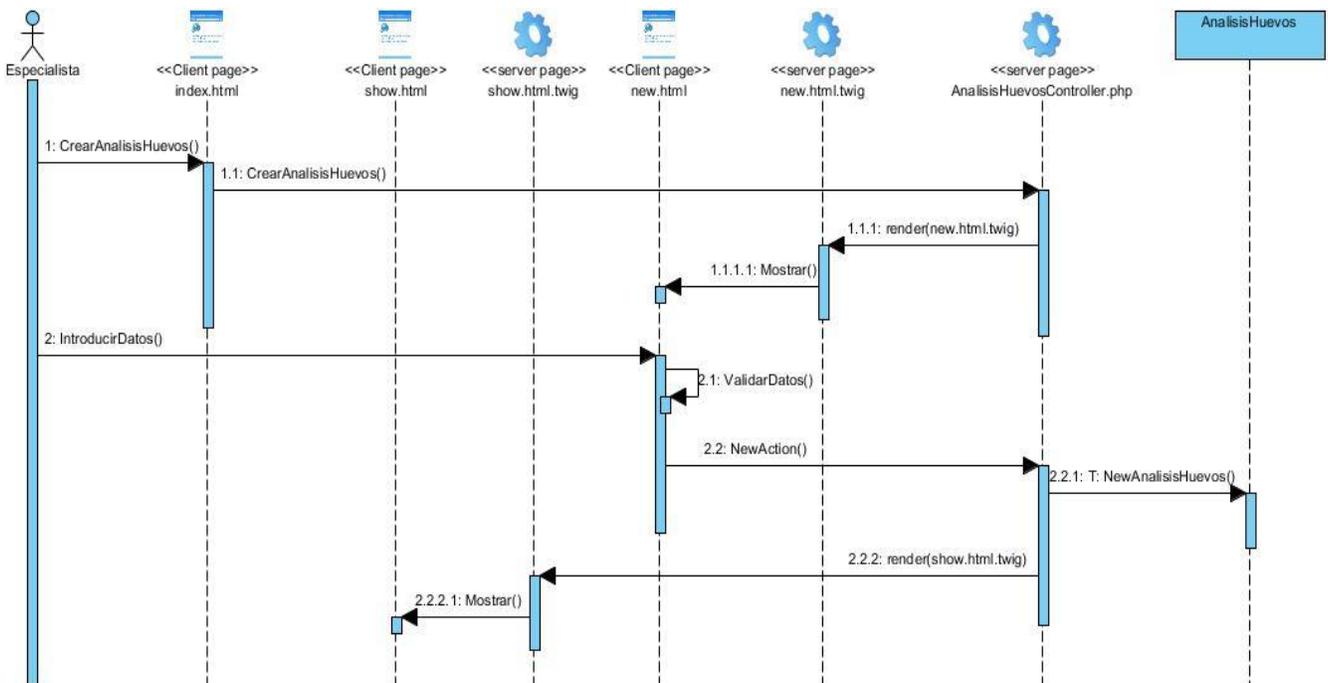


Anexo 35: Diagrama de secuencia del caso de uso Gestionar análisis de neonatos,

sección Eliminar análisis de neonatos.

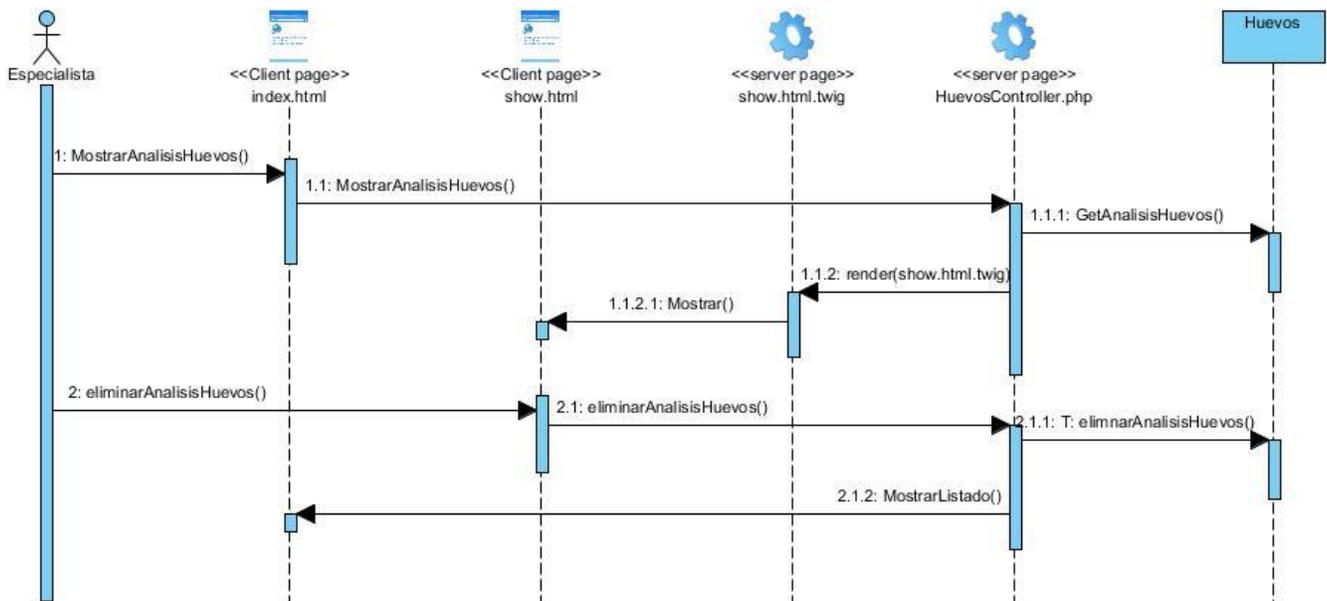


Anexo 36: Diagrama de secuencia del caso de uso Gestionar análisis de huevos, sección Crear análisis de huevos.



Anexo 37: Diagrama de secuencia del caso de uso Gestionar análisis de huevos,

sección Eliminar análisis de huevos.



Anexo 38: Descripción de la tabla rastro.

Tabla rastro		
Atributo	Tipo	Descripción
id	integer	Identificador de cada rastro en la tabla.
tipo_id	integer	Identificador de la tabla que guarda el tipo de rastro.
direccion_id	integer	Identificador de la tabla que guarda la dirección.
especie_id	integer	Identificador de la tabla que guarda el tipo de especie.
condicion_id	integer	Identificador de la tabla que guarda la condición del rastro.
localizacion_id	integer	Identificador de la tabla que guarda la localización del rastro.
numero	character	Número del rastro.
ancho	numeric	Ancho del rastro.
recorrido	integer	Identificador de la tabla que guarda el recorrido.
playa	integer	Identificador de la tabla que guarda la playa.
areaprotegida	integer	Identificador de la tabla que guarda el área protegida.

Anexo 39: Descripción de la tabla nidotortuga.

Tabla nidotortuga		
Atributo	Tipo	Descripción
id	integer	Identificador de cada nido de tortuga en la tabla.
depredador_id	integer	Identificador de la tabla que guarda los depredadores.
numerodeintentos	integer	Cantidad de intentos de poner huevos.
especie_id	integer	Identificador de la tabla que guarda el tipo de especie.
condicion_id	integer	Identificador de la tabla que guarda la condición del nido.
localizacion_id	integer	Identificador de la tabla que guarda la localización del nido.
numero	character	Número del nido.
recorrido	integer	Identificador de la tabla que guarda el recorrido.
playa	integer	Identificador de la tabla que guarda la playa.
areaprotegida	integer	Identificador de la tabla que guarda el área protegida.
fechapuesta	timestamp	Fecha en que pone la tortuga los huevos.
zonadeplaya	integer	Zona de la playa en que está situado el nido.
tiempoincubacion	integer	Cantidad de tiempo de incubación.
totalhuevosrotos	integer	Cantidad de huevos rotos.
porcentajebandaopaca	integer	Porcentaje de la banda opaca.
numerodehuevos	integer	Cantidad de huevos.
profundidad	numeric	Profundidad que tiene el nido.
distancialineamarea	numeric	Distancia que tiene la marea.

Anexo 40: Descripción de la tabla analisishuevos.

Tabla nidotortuga		
Atributo	Tipo	Descripción
id	integer	Identificador del análisis de huevos en la tabla.
totalcascarones	integer	Cantidad de cascarones.

totalhuevoscondesarrolloembrionario	integer	Cantidad de huevos con desarrollo embrionario.
totalhuevossindesarrolloembrionario	integer	Cantidad de huevos sin desarrollo embrionario.
totalhuevos	integer	Cantidad de huevos.
periodoincubacion	integer	Cantidad de tiempo que dura el periodo de incubación.
totalembrionesconanomalias	integer	Cantidad de embriones con anomalías.
totaldepredadoporhormigas	integer	Cantidad de embriones depredados por hormigas.
totaldepredadoporcanchales	integer	Cantidad de embriones depredados por cangrejos.
totalconlarvas	integer	Cantidad de embriones con larvas.
totalconhongos	integer	Cantidad de embriones con hongos.
nido	integer	Identificador de la tabla que guarda el nido.
recorrido	integer	Identificador de la tabla que guarda el recorrido.
playa	integer	Identificador de la tabla que guarda la playa.
areaprotegida	integer	Identificador de la tabla que guarda el área protegida.

Anexo 41: Descripción de la tabla analisisneonatos.

Tabla analisisneonatos		
Atributo	Tipo	Descripción
id	integer	Identificador del análisis de neonatos en la tabla.
totalneonatos	integer	Cantidad de neonatos.
totalmuertos	integer	Cantidad de neonatos muertos.
totalnormales	integer	Cantidad de neonatos normales.
totalconrestosvitelo	integer	Cantidad de neonatos con restos vitelo.
totalconlarvas	integer	Cantidad de neonatos con larvas.

anchorectopromedio	numeric	El ancho recto promedio.
largorectopromedio	numeric	El lago recto promedio.
profundidad	numeric	Profundidad del nido.
observaciones	text	Lo referente a los neonatos que no se refleje en el análisis.
totalneorezagados	integer	Cantidad de neonatos rezagados.
nido	integer	Identificador de la tabla que guarda el nido.
recorrido	integer	Identificador de la tabla que guarda el recorrido.
playa	integer	Identificador de la tabla que guarda la playa.
areaprotegida	integer	Identificador de la tabla que guarda el área protegida.

Anexo 42: Descripción de la tabla usuario.

Tabla usuario		
Atributo	Tipo	Descripción
id	integer	Identificador del usuario en la tabla.
nombre	character	Nombre del usuario.
correo	character	Correo del usuario.
apellidos	character	Apellidos del usuario.
password	character	Contraseña del usuario.
carnetidentidad	character	Carnet de identidad del usuario.
tipousuario	character	Tipo de usuario.

Anexo 43: Diseño de Caso de Prueba (CP)

CP Gestionar área protegida

SC 1 Incluir área protegida

Escenario	Descripción	Nombre	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Entrar a la sección área protegida	Realiza una acción sobre un área protegida.			Permite realizar varias acciones con un área protegida: - Crear una nueva área protegida. - Modificar los datos de un área protegida. Ver SC 3: "Modificar área protegida". - Ver los datos del área protegida. Ver SC 2: "Ver área protegida". - Eliminar un área protegida. Ver SC 4: "Eliminar área protegida".	Administración/Área protegida
EC 1.2 Incluir área protegida	Selecciona la opción de crear una área protegida			Brinda la posibilidad de introducir los datos siguientes: • Nombre • Descripción Y permite: • Guardar los datos. • Cancelar la operación.	Administración/Área protegida/Crear área
EC 1.3 Introducir datos de área protegida	Introduce los datos y selecciona la opción de guardar los datos	V	V	Brinda la posibilidad de introducir los datos siguientes: • Nombre • Descripción Y permite: • Guardar los datos. • Cancelar la operación.	Administración/Área protegida/Crear área/Aceptar
EC 1.4 Cancelar creación del área protegida	Selecciona la opción de cancelar.			Elimina los datos creados. Muestra un mensaje de información. Regresa al listado de área protegida.	Administración/Área protegida/Crear área/Cancelar

EC 1.5 Datos incompletos	Existen datos incompletos.	I	V	Muestra encima del campo vacío un mensaje de información. Muestra un indicador sobre los campos vacíos. <i>Regresa al EC 1.3.</i>	Administración/Área protegida/Crear área/Aceptar
		V	I		
EC 1.6 Datos incorrectos	Existen datos incorrectos.	I	V	Muestra encima del campo incorrecto un mensaje de información. Muestra un indicador sobre los campos incorrectos. <i>Regresa al EC 1.3.</i>	Administración/Área protegida/Crear área/Aceptar
		V	I		

SC 2 Ver área protegida

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 2.1 Ver datos de una área protegida	Selecciona la opción de ver los datos de una área protegida	Muestra los datos del área protegida seleccionado: • Nombre • Descripción Y permite: • Volver a la vista anterior • Editar los datos • Eliminar el área protegida	Administración/Área protegida/Mostrar
EC 2.2 Salir del ver datos del área protegida	Selecciona la opción de volver a la vista anterior.	Regresa al listado de las áreas protegidas.	Administración/Área protegida/Mostrar/Atrás

SC 3 Modificar área protegida

Escenario	Descripción	Nombre	Descripción	Respuesta del sistema	Flujo central
EC 3.1 Opción de modificar los datos del área protegida	Selecciona la opción de modificar los datos			Muestra el área protegida permitiendo modificar los valores de la misma: • Nombre • Descripción Y permite: • Editar los datos	Administración/Área protegida/Editar

				• Cancelar la operación	
EC 3.2 Modifica los datos y selecciona la opción de aceptar para actualizar	Modifica los datos que necesite y selecciona la opción de aceptar para actualizar los datos del área protegida.	V	V	Permite modificar los valores de la misma: • Nombre • Descripción Y permite: • Editar los datos • Cancelar la operación	Administración/Área protegida/Editar/Aceptar
EC 3.3 Opción de cancelar la operación	Selecciona la opción de cancelar.			Elimina los datos creados y regresa al listado de noticias. Muestra un mensaje de información.	Administración/Área protegida/Editar/Cancelar
EC 3.4 Datos incorrectos	Existen datos incorrectos	I	V	Muestra encima del campo incorrecto un mensaje de información. Muestra un indicador sobre los campos incorrectos. <i>Regresa al EC 3.2.</i>	Administración/Área protegida/Editar/Aceptar
		V	I		
EC 3.6 Datos incompletos	Existen datos incompletos.	I	V	Muestra encima del campo vacío un mensaje de información. Muestra un indicador sobre los campos vacíos. <i>Regresa al EC 3.2.</i>	Administración/Área protegida/Editar/Aceptar
		V	I		

SC 4 Eliminar área protegida

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 4.1 Opción de Eliminar una área protegida	Selecciona la opción de eliminar los datos de un área protegida.	Muestra un mensaje de alerta y permite: * Eliminar.	Administración/Área protegida/Mostrar/Eliminar

		* Cancelar.	
EC 4.2 Acepta eliminar el elemento seleccionado	Selecciona la opción Eliminar.	Eliminar los datos del área protegida. Regresa al listado de áreas protegidas. Muestra un mensaje de confirmación.	Administración/Área protegida/Mostrar/Eliminar/Aceptar
EC 4.3 Cancela la operación de eliminar el elemento	Selecciona la opción de cancelar.	Regresa a la vista anterior.	Administración/Área protegida/Mostrar/Eliminar/Cancelar

CP Gestionar playa						
SC 1 Incluir playa						
Escenario	Descripción	Nombre	Descripción	Área protegida	Respuesta del sistema	Flujo central
EC 1.1 Entrar a la sección de playa	Realiza una acción sobre una playa				Permite realizar varias acciones con una playa: - Crear una nueva playa. - Modificar los datos de una playa. Ver SC 3: "Modificar playa". - Ver los datos de la playa. Ver SC 2: "Ver playa".	Administración/P layas

					- Eliminar una playa. Ver SC 4: "Eliminar playa".	
EC 1.2 Incluir playa	Selecciona la opción de crear una playa				Brinda la posibilidad de introducir los datos siguientes: • Nombre • Descripción • Área protegida Y permite: • Guardar los datos. • Cancelar la operación.	Administración/Playas/Crear playa
EC 1.3 Introducir datos de la playa	Introduce los datos y selecciona la opción de guardar los datos	V	V	V	Valida los datos. Registra los datos de la playa. Muestra un mensaje de información	Administración/Playas/Crear playa/Aceptar
EC 1.4 Cancelar creación del playa	Selecciona la opción de cancelar.				Regresa al listado playas. Muestra un mensaje de información.	Administración/Playas/Crear playa/Cancelar
EC 1.5 Datos incompletos	Existen datos incompletos.	I	V	V	Muestra un mensaje de información. Muestra un indicador sobre los campos vacíos. Regresa al EC 1.3.	Administración/Playas/Crear playa/Aceptar
		V	I	V		
		V	V	I		
EC 1.6 Datos incorrectos	Existen datos incorrectos.	I	V	V	Muestra un mensaje de información. Muestra un indicador sobre los campos vacíos. Regresa al EC 1.3.	Administración/Playas/Crear playa/Aceptar
		V	I	V		
		V	V	I		

SC 2 Ver datos del playa

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 2.1 Ver datos de una playa	Selecciona la opción de ver los datos de una playa	Muestra los datos de la playa seleccionada: <ul style="list-style-type: none"> • Nombre • Descripción • Área protegida Y permite: <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar la playa 	Administración/Playas/Mostrar
EC 2.2 Salir de ver datos del playa	Selecciona la opción de volver a la vista anterior.	Regresa al listado de recorridos.	Administración/Playas/Mostrar /Atrás

SC 3 Modificar playa

Escenario	Descripción	Nombre	Descripción	Área protegida	Respuesta del sistema	Flujo central
EC 3.1 Opción de modificar los datos de la playa	Selecciona la opción de modificar los datos de la playa				Muestra la playa permitiendo modificar los valores de la misma: <ul style="list-style-type: none"> • Nombre • Descripción • Área protegida Y permite: <ul style="list-style-type: none"> • Editar los datos • Cancelar la operación 	Administración/Playas/Editar
EC 3.2 Modifica los datos y selecciona la opción de aceptar para actualizar	Modifica los datos que necesite y selecciona la opción de aceptar para actualizar.	V	V	V	Valida los datos. Actualiza los datos de la playa. Muestra un mensaje de información. Muestra los datos de la playa. Ver Sección 2: "Ver datos de la playa"	Administración/Playas/Editar/Aceptar

EC 3.3 Opción de cancelar.	Selecciona la opción de cancelar la operación.				Regresa al listado de playas. Muestra un mensaje de información	Administración/Playas/Editar/Cancelar
EC 3.5 Datos incorrectos	Existen datos incorrectos	I	V	V	Muestra un mensaje de información. Muestra un indicador sobre los campos vacíos. Regresa al EC 3.2.	Administración/Playas/Editar/Aceptar
		V	I	V		
		V	V	I		
EC 3.6 Datos incompletos	Existen datos incompletos.	I	V	V	Muestra un mensaje de información. Muestra un indicador sobre los campos vacíos. <u>Regresa al EC 3.2.</u>	Administración/Playas/Editar/Aceptar
		V	I	V		
		V	V	I		

SC 4 Eliminar playa

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 4.1 Opción de Eliminar la playa	Selecciona la opción de eliminar la playa	Muestra un mensaje de alerta. Y permite: • Aceptar. • Cancelar.	Administración/Playas/Mostrar/Eliminar
EC 4.2 Acepta eliminar el elemento seleccionado	Selecciona la opción Aceptar	Elimina la playa. Muestra un mensaje de información	Administración/Playas/Mostrar/Eliminar/Aceptar
EC 4.3 Cancela la operación de eliminar el elemento	Selecciona la opción de cancelar.	Regresa a la vista anterior.	Administración/Playas/Mostrar/Eliminar/Cancelar

CP Gestionar usuario										
SC 1 Incluir usuario										
Escenario	Descripción	Nombre	Apellidos	Carnet de identidad	Correo electrónico	Contraseña	Repetir contraseña	Tipo de usuario	Respuesta del sistema	Flujo central
EC 1.1 Entrar a la sección de usuario	Realiza una acción sobre un usuario								Permite realizar varias acciones con un usuario: - Crear un nuevo usuario. - Modificar los datos de un usuario. Ver Sección 1: "Modificar usuario". - Ver los datos del usuario. Ver Sección 2: "Ver Usuario". - Eliminar un usuario. Ver Sección 3: "Eliminar usuario".	Administración/Usuario
EC 1.2 Incluir usuario	Selecciona la opción de crear un usuario.								Brinda la posibilidad de introducir los datos siguientes: • Nombre. • Apellidos • Carnet de identidad • Correo electrónico • Contraseña • Repetir contraseña • Tipo de usuario Y permite: • Guardar los	Administración/Usuario/Crear usuario

									datos. • Cancelar la operación.	
EC 1.3 Introducir datos del usuario	Introduce los datos y selecciona la opción de guardar los datos	V	V	V	V	V	V	V	Valida los datos. Registra los datos del usuario. Muestra un mensaje de información.	Administración/Usuario/Crear usuario/Aceptar
EC 1.4 Cancelar creación del usuario	Selecciona la opción de cancelar.								Elimina los datos creados. Muestra un mensaje de información. Regresa al listado de usuarios.	Administración/Usuario/Crear usuario/Cancelar
EC 1.5 Datos incompletos	Existen datos incompletos.	I	V	V	V	V	V	V	Muestra encima del campo vacío un mensaje de información. Muestra un indicador sobre los campos vacíos. <u>Regresa al EC 1.3.</u>	Administración/Usuario/Crear usuario/Aceptar
		V	I	V	V	V	V	V		
		V	V	I	V	V	V	V		
		V	V	V	I	V	V	V		
		V	V	V	V	I	V	V		
		V	V	V	V	V	I	V		
		V	V	V	V	V	V	I		
EC 1.6 Datos incorrectos	Existen datos incorrectos.	I	V	V	V	V	V	V	Muestra encima del campo incorrecto un mensaje de información. Muestra un indicador sobre los campos incorrectos. <u>Regresa al EC 1.3.</u>	Administración/Usuario/Crear usuario/Aceptar
		V	I	V	V	V	V	V		
		V	V	I	V	V	V	V		
		V	V	V	I	V	V	V		
		V	V	V	V	I	V	V		
		V	V	V	V	V	I	V		
		V	V	V	V	V	V	I		

SC 2 Ver datos del usuario

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 2.1 Ver datos de un usuario	Selecciona la opción de ver los datos de un usuario	Muestra los datos del usuario seleccionado: <ul style="list-style-type: none"> • Nombre. • Apellidos • Carnet de identidad • Correo electrónico • Tipo de usuario Y permite: <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar el Usuario 	Administración/Usuario/Mostrar
EC 2.2 Salir de ver datos del usuario	Selecciona la opción de volver a la vista anterior.	Regresa al listado de usuarios.	Administración/Usuario/Mostrar/Atrás

SC 3 Modificar usuario

Escenario	Descripción	Nombre	Apellidos	Carnet de identidad	Correo electrónico	Contraseña	Repetir contraseña	Tipo de usuario	Respuesta del sistema	Flujo central
EC 3.1 Opción de modificar los datos del usuario	Selecciona la opción de modificar los datos del usuario.								Muestra los datos de un usuario permitiendo modificar los valores del mismo: <ul style="list-style-type: none"> • Nombre. • Apellidos • Carnet de identidad • Correo electrónico • Contraseña • Repetir contraseña • Tipo de usuario • Especie Y permite: <ul style="list-style-type: none"> • Editar los datos 	Administración/Usuario/Editar

									• Cancelar la operación	
EC 3.2 Modifica los datos y selecciona la opción de aceptar para actualizar	Modifica los datos que necesite y selecciona la opción de aceptar para actualizar.	V	V	V	V	V	V	V	Valida los datos. Actualiza los datos del usuario. Muestra un mensaje de información. Muestra los datos del usuario Ver Sección 2: "Ver datos del usuario"	Administración/Usuario/Editar/Aceptar
EC 3.3 Opción de cancelar	Selecciona la opción de Cancelar la operación.								Regresa al listado de Usuarios. Muestra un mensaje de información	Administración/Usuario/Editar/Cancelar
EC 3.5 Datos incorrectos	Existen datos incorrectos	I	V	V	V	V	V	V	Muestra un mensaje de información. Muestra un indicador sobre los campos vacíos. <u>Regresa al EC 3.2.</u>	Administración/Usuario/Editar/Aceptar
		V	I	V	V	V	V	V		
		V	V	I	V	V	V	V		
		V	V	V	I	V	V	V		
		V	V	V	V	I	V	V		
		V	V	V	V	V	I	V		
		V	V	V	V	V	V	I		
EC 3.6 Datos incompletos	Existen datos incompletos.	I	V	V	V	V	V	V	Muestra un mensaje de información. Muestra un indicador sobre los campos vacíos. Regresa al EC 3.2.	Administración/Usuario/Editar/Aceptar
		V	I	V	V	V	V	V		
		V	V	I	V	V	V	V		
		V	V	V	I	V	V	V		
		V	V	V	V	I	V	V		
		V	V	V	V	V	I	V		
		V	V	V	V	V	V	I		

SC 4 Eliminar usuario

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 4.1 Opción de eliminar el usuario	Selecciona la opción de eliminar el usuario.	Muestra un mensaje de alerta. Y permite: • Aceptar. • Cancelar.	Administración/Usuario/Mostrar/Eliminar
EC 4.2 Acepta eliminar el elemento seleccionado	Selecciona la opción aceptar	Elimina el usuario. Muestra un mensaje de información	Administración/Usuario/Mostrar/Eliminar/Aceptar
EC 4.3 Cancela la operación de eliminar el elemento	Selecciona la opción de cancelar.	Regresa a la vista anterior.	Administración/Usuario/Mostrar/Eliminar/Cancelar

CP Gestionar tortuga

SC 1 Incluir tortuga

Escenario	Descripción	Área protegida	Plaza	Recolector	Marca	Fibropiloma	Anchorecto	Anchorecto	Largorecto	Largorecto	Imagen	Horadesenso	Espeque	Acción	Respuesta del sistema	Flujo central

<p>EC 1.1 Entrar a la sección de tortuga</p>	<p>Realiza una acción sobre una tortuga</p>														<p>Permite realizar varias acciones con una tortuga: - Crear una nueva tortuga. - Modificar los datos de una tortuga. Ver Sección 1: "Modificar tortuga". - Ver los datos de la tortuga. Ver Sección 2: "Ver tortuga". - Eliminar una tortuga. Ver Sección 3: "Eliminar tortuga".</p>	<p>Campaña/Tortuga</p>
<p>EC 1.2 Incluir tortuga</p>	<p>Selecciona la opción de crear una tortuga</p>													<p>Brinda la posibilidad de introducir los datos siguientes: • Área protegida. • Playa • Recorrido • Marca • Fibropapiloma • Ancho recto • Ancho curvo • Largo recto • Largo curvo • Imagen • Hora descenso • Especie • Acción Y permite: • Guardar los datos. • Cancelar la operación.</p>	<p>Campaña/Tortuga/Crear tortuga</p>	

EC 1.3 Introducir datos del tortuga	Introduce los datos y selecciona la opción de guardar los datos	V	V	V	V	V	V	V	V	V	V	V	V	V	Valida los datos. Registra los datos de la tortuga. Muestra un mensaje de información.	Campaña/Tortuga/Crear tortuga/Aceptar	
EC 1.4 Cancelar creación de la tortuga	Selecciona la opción de cancelar.														Elimina los datos creados. Muestra un mensaje de información. Regresa al listado de tortugas.	Campaña/Tortuga/Crear tortuga/Cancelar	
EC 1.5 Datos incompletos	Existen datos incompletos	I	V	V	V	V	V	V	V	V	V	V	V	V	Muestra encima del campo vacío un mensaje de información. Muestra un indicador sobre los campos vacíos. <u>Regresa al EC 1.3.</u>	Campaña/Tortuga/Crear tortuga/Aceptar	
		V	I	V	V	V	V	V	V	V	V	V	V	V			
		V	V	I	V	V	V	V	V	V	V	V	V	V			V
		V	V	V	I	V	V	V	V	V	V	V	V	V			V
		V	V	V	V	I	V	V	V	V	V	V	V	V			V
		V	V	V	V	V	I	V	V	V	V	V	V	V			V
		V	V	V	V	V	V	I	V	V	V	V	V	V			V
		V	V	V	V	V	V	V	I	V	V	V	V	V			V
		V	V	V	V	V	V	V	V	I	V	V	V	V			V
		V	V	V	V	V	V	V	V	V	I	V	V	V			V
		V	V	V	V	V	V	V	V	V	V	V	V	V			I
		EC 1.6 Datos incorrectos	Existen datos incorrectos	I	V	V	V	V	V	V	V	V	V	V			V
V	I			V	V	V	V	V	V	V	V	V	V	V			
V	V			I	V	V	V	V	V	V	V	V	V	V			
V	V			V	I	V	V	V	V	V	V	V	V	V			

rectos	rectos.	V	V	V	V	I	V	V	V	V	V	V	V	V	información. Muestra un indicador sobre los campos incorrectos. <u>Regresa al EC 1.3.</u>
		V	V	V	V	V	I	V	V	V	V	V	V	V	
		V	V	V	V	V	V	I	V	V	V	V	V	V	
		V	V	V	V	V	V	V	I	V	V	V	V	V	
		V	V	V	V	V	V	V	V	I	V	V	V	V	
		V	V	V	V	V	V	V	V	V	I	V	V	V	
		V	V	V	V	V	V	V	V	V	V	I	V	V	
		V	V	V	V	V	V	V	V	V	V	V	I	V	
		V	V	V	V	V	V	V	V	V	V	V	V	I	

SC 2 Ver datos de tortuga

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 2.1 Ver datos de una tortuga	Selecciona la opción de ver los datos de una tortuga.	<p>Muestra los datos de la tortuga seleccionada:</p> <ul style="list-style-type: none"> • Imagen • Playa • Marca • Fibropapiloma • Ancho recto • Ancho curvo • Largo recto • Largo curvo • Imagen • Hora descenso • Especie • Acción <p>Y permite:</p> <ul style="list-style-type: none"> • Volver a la vista anterior • Editar los datos • Eliminar la tortuga 	Campaña/Tortuga/Mostrar
EC 2.2 Salir de ver datos del tortuga	Selecciona la opción de volver a la vista anterior.	Regresa al listado de Tortugas.	Campaña/Tortuga/Mostrar/Atrás

SC 3 Modificar tortuga

Escenario	Descripción	Área protegida	Playa	Recochado	Marca	Fibropapiloma	Ancho recto	Ancho curvo	Largo recto	Largo curvo	Imagen	Hora descenso	Especie	Acción	Respuesta del sistema	Flujo central
-----------	-------------	----------------	-------	-----------	-------	---------------	-------------	-------------	-------------	-------------	--------	---------------	---------	--------	-----------------------	---------------

												so						
EC 3.1	Selección de opción de modificar los datos de la tortuga																<p>Muestra los datos de una tortuga permitiendo modificar los valores de la misma:</p> <ul style="list-style-type: none"> • Área protegida. • Playa • Recorrido • Marca • Fibropapiloma • Ancho recto • Ancho curvo • Largo recto • Largo curvo • Imagen • Hora descenso • Especie • Acción <p>Y permite:</p> <ul style="list-style-type: none"> • Editar los datos • Cancelar la operación 	<p>Campaña /Tortuga/Editar</p>
EC 3.2	Modificación de los datos y selección de la opción de aceptar para actualizar.	V	V	V	V	V	V	V	V	V	V	V	V	V	V		<p>Valida los datos. Actualiza los datos de la tortuga. Muestra un mensaje de información. Muestra los datos de la tortuga. Ver Sección 2: "Ver datos del Tortuga"</p>	<p>Campaña /Tortuga/Editar/Aceptar</p>

lizar																		
EC 3.3 Opción de cancelar.	Selecciona la opción de Cancelar la operación.																Regresa al listado de tortugas. Muestra un mensaje de información	Campaña /Tortuga/Editar/Cancelar
EC 3.5 Datos incorrectos	Existen datos incorrectos	I	V	V	V	V	V	V	V	V	V	V	V	V	V	V	Muestra un mensaje de información. Muestra un indicador sobre los campos vacíos. <u>Regresa al EC 3.2.</u>	Campaña /Tortuga/Editar/Aceptar
		V	I	V	V	V	V	V	V	V	V	V	V	V	V	V		
		V	V	I	V	V	V	V	V	V	V	V	V	V	V	V		
		V	V	V	I	V	V	V	V	V	V	V	V	V	V	V		
		V	V	V	V	I	V	V	V	V	V	V	V	V	V	V		
		V	V	V	V	V	I	V	V	V	V	V	V	V	V	V		
		V	V	V	V	V	V	I	V	V	V	V	V	V	V	V		
		V	V	V	V	V	V	V	I	V	V	V	V	V	V	V		
		V	V	V	V	V	V	V	V	I	V	V	V	V	V	V		
		V	V	V	V	I	V	V	V	V	V	V	I	V	V	V		
		V	V	V	V	V	V	V	V	V	V	V	V	I	V	V		
		V	V	V	V	V	V	V	V	V	V	V	V	V	I	V		
		EC 3.6 Datos incompletos	Existen datos incompletos.	I	V	V	V	V	V	V	V	V	V	V	V	V		
V	I			V	V	V	V	V	V	V	V	V	V	V	V	V		
V	V			I	V	V	V	V	V	V	V	V	V	V	V	V		
V	V			V	I	V	V	V	V	V	V	V	V	V	V	V		
V	V			V	V	I	V	V	V	V	V	V	V	V	V	V		
V	V			V	V	V	I	V	V	V	V	V	V	V	V	V		
V	V			V	V	V	V	I	V	V	V	V	V	V	V	V		
V	V			V	V	V	V	V	I	V	V	V	V	V	V	V		
V	V			V	V	V	V	V	V	I	V	V	V	V	V	V		
V	V			V	V	V	V	V	V	V	I	V	V	V	V	V		
V	V			V	V	V	V	V	V	V	V	V	I	V	V	V		
V	V			V	V	V	V	V	V	V	V	V	V	I	V	V		
V	V			V	V	V	V	V	V	V	V	V	V	V	I	V		

SC 4 Eliminar tortuga

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 4.1 Opción de Eliminar la tortuga	Selecciona la opción de eliminar la tortuga.	Muestra un mensaje de alerta. Y permite: • Aceptar. • Cancelar.	Campaña/Tortuga/Mostrar/Eliminar
EC 4.2 Acepta eliminar el elemento seleccionado	Selecciona la opción Aceptar	Elimina la tortuga. Muestra un mensaje de información	Campaña/Tortuga/Mostrar/Eliminar/Aceptar
EC 4.3 Cancela la operación de eliminar el elemento	Selecciona la opción de cancelar.	Regresa a la vista anterior.	Campaña/Tortuga/Mostrar/Eliminar/Cancelar