

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

Facultad 9



Título

PROPUESTA DE DISEÑO DE UN SISTEMA DE GESTIÓN PARA EL ARCHIVO TÉCNICO DE LA OFICINA NACIONAL DE RECURSOS MINERALES

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

Autores: Macbeth Pérez Díaz

Dáryll Miqueli Simón

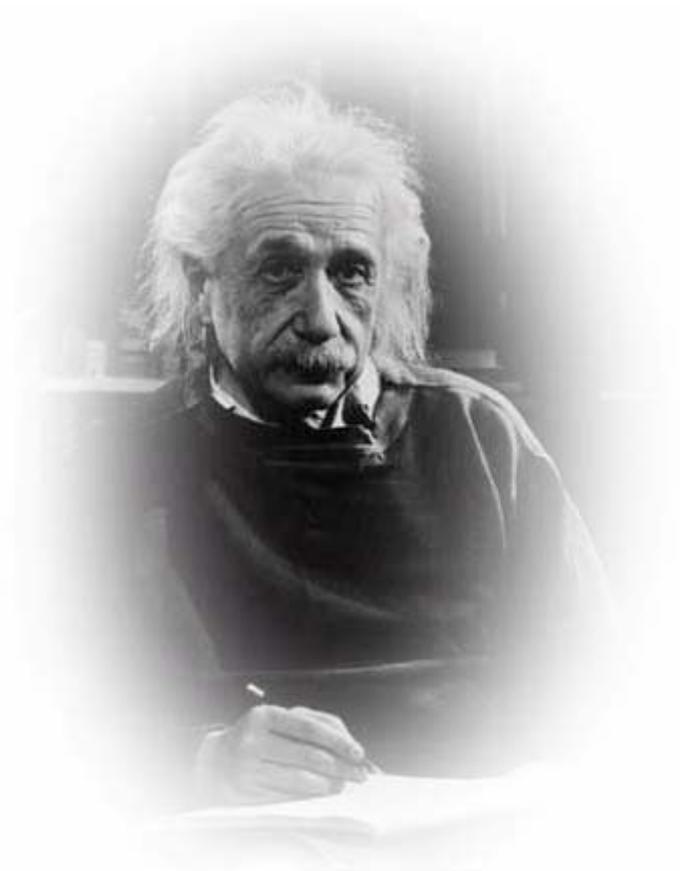
Tutores: Ing. Adrián Vieyto Amador

Consultor: Ing. Ramón Darío Infante Escalona



Ciudad de la Habana, Julio de 2008

“Año 50 de la Revolución”



“En lugar de ser un
hombre de éxito, busca
ser un hombre valioso: lo

demás llegará
naturalmente.”

Albert Einstein

Nosotros, Macbeth Pérez Díaz y Dáryll Miqueli Simón declaramos que somos los únicos autores de este trabajo y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales del mismo.

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

Macbeth Pérez Díaz

Autor

Dáryll Miqueli Simón

Autor

Ing. Adrián Vieyto Amador

Tutor

A los que creen en el optimismo de los hombres.
A mi familia por servirme de bastón en todos estos años.
A los que hoy no están pero van conmigo.
A ti...

Macbeth

Macbeth

A mi madre, por sus consejos y su incondicional ayuda, por hacer de lo imposible lo posible para que llegara hasta aquí, gracias.

A mi padre que aunque ya no se encuentre conmigo por convertirme en lo que soy hoy.

A mi otra madre querida: mi abuela por su infinita preocupación por todo en lo que en mi respecta.

A mi sangre gemela: mi hermano.

A mi tutor Adrian, a Lili y a Isel por su incondicional apoyo.

A la Revolución y a nuestro comandante por permitirme compartir su sueño.

A la Universidad y al proyecto PNICG por haber aportado su grano de arena en mi formación como profesional.

A todos, eternamente agradecido.

Gracias.

A mi familia que es lo más valioso que tengo.
A los que creen que nunca es demasiado tarde para mejorar.
A todos ustedes...

Dáryll

Dáryll

A mi madre por ser la persona que más quiero en este mundo, por su preocupación y ayuda incondicional.

A mi padre por ser mi ejemplo de abnegación y sacrificio, por su confianza depositada en mí.

A mi hermana que la quiero más que a mí mismo, por su cariño y confianza.

A mi tía Bertica por revisarme el trabajo y por sus valiosos consejos.

A mi novia Leanedys por su paciencia, comprensión y cariño.

A mis amigos por soportarme como soy y por compartir conmigo todos estos años.

A mi tutor Adrián y a Ramón Infante por su ayuda muy útil para la realización de este trabajo.

A la universidad por formarme como profesional instruido y capacitado.

A la Revolución por hacer de mí una mejor persona e inculcarme los más hermosos valores que debe tener todo ser humano.

A Fidel por ser mi guía, mi ejemplo.

A todos Gracias

No los defraudaré

RESUMEN

La *Oficina Nacional de Recursos Minerales* (ONRM) pertenece al *Ministerio de Industria Básica de la República de Cuba* (MINBAS) y es la entidad encargada de velar por el aprovechamiento y explotación de los recursos minerales y de hidrocarburos del país. La misma maneja un gran número de información sensible e importante, consecuencia de todas las investigaciones geológicas realizadas a lo largo y ancho del archipiélago cubano durante muchos años.

Este trabajo analiza y diseña una propuesta de aplicación Web para automatizar de manera más eficiente el proceso de gestionar datos referativos de dichos documentos en una base de datos, con el objetivo de optimizar su búsqueda en el *Archivo Técnico* (AT). Además cuenta con el servicio de préstamo y devolución de documentos a personas o entidades interesadas. Una aplicación Web garantiza mayor eficiencia para llevar a cabo estos procesos que la aplicación existente, brinda soporte a estos procesos de manera más dinámica, más amena, más segura, perfeccionando el sistema de gestión de la información.

Este documento expone los resultados de todo el trabajo investigativo realizado. Se identifican y describen los procesos que se van a informatizar; se dan argumentos para demostrar que la situación problemática requiere de un sistema que cumpla con los requisitos propuestos para satisfacer las necesidades identificadas. Se muestran los resultados del Análisis y el Diseño de la propuesta del sistema.

Palabras Claves: MINBAS, ONRM, Aplicación Web SGAT.

ÍNDICE DE FIGURAS

Figura 1: Sectores que integran la Industria Básica-----	14
Figura 2: Proceso de desarrollo de Software -----	77
Figura 3: Fases de RUP-----	77
Figura 4: Modelo de CU del Negocio -----	40
Figura 5: Modelo de Objeto de Negocio-----	43
Figura 6: Diagrama de CU del Sistema -----	47
Figura 7: Diagrama de Clases de Análisis Caso Uso Gestionar Informes -----	54
Figura 8: Diagrama de Clases de Análisis Caso Uso Realizar Préstamos-----	55
Figura 9: Diagrama de Clases de Análisis Caso Uso Devolver Préstamo -----	55
Figura 10: Estructura del patrón MVC según Symfony-----	57
Figura 11: Estructura del Modelo -----	60
Figura 12: Diagrama de Clases del Diseño del CU Realizar Préstamo-----	60
Figura 13: Diagrama de Clases del Diseño del CU Gestionar Informes-----	61
Figura 14: Diagrama de Clases del Diseño del CU Devolver Préstamo-----	62
Figura 15: Flujo de trabajo de Symfony -----	63
Figura 16: Diagrama de Clases Persistentes -----	64
Figura 17: Modelo de Datos -----	66
Figura 18: Diagrama de Componentes-----	67
Figura 19: Modelo de despliegue -----	68
Figura 20: Diagrama de Secuencia Insertar Informe-----	78
Figura 21: Diagrama de Secuencia Modificar Informe-----	79
Figura 22: Diagrama de Secuencia Eliminar Informe-----	79
Figura 23: Diagrama de Secuencia Realizar Préstamos-----	80
Figura 24: Diagrama de Secuencia Devolver Préstamos-----	81
Figura 25: Diagrama de Actividades del CU Devolver Préstamos -----	82

Figura 26: Diagrama de Actividades del CU Realizar Préstamos -----	83
Figura 27: Diagrama de Actividades del CU Entregar Informes -----	84
Figura 28: Clases del Modelo del CU Gestionar Informe -----	85
Figura 29: Clases del Modelo del CU Realizar Préstamo -----	86
Figura 30: Clases del Modelo del CU Devolver Préstamo-----	86
Figura 31: Paquete Success -----	87
Figura 32: Paquete Actions-----	87
Figura 33: Paquete Acceso a Datos-----	88

ÍNDICE DE TABLAS

Tabla 1: Actor del Negocio.....	39
Tabla 2: Trabajador del Negocio.....	40
Tabla 3: Descripción del CU Entregar_Informes	41
Tabla 4: Descripción del CU Realizar_Préstamo.....	42
Tabla 5: Descripción del CU Realizar_Préstamo.....	43
Tabla 6: Actor del Sistema.....	46
Tabla 7: Descripción del Caso de Uso del Sistema Gestionar Informes	48
Tabla 8: Descripción del Caso de Uso del Sistema Realizar Préstamo	50
Tabla 9: Descripción del Caso de Uso del Sistema Devolver Préstamo	51
Tabla 10: Clases del Modelo del framework.....	59

INTRODUCCIÓN	4
CAPÍTULO1 FUNDAMENTACIÓN TEÓRICA	9
1.1 Conceptos Generales.	9
1.1.1 Sistema	9
1.1.2 Sistema Informático	11
1.1.3 Aplicación Informática	11
1.1.4 Aplicación Web	12
1.2 Conceptos asociados al dominio del problema.	12
1.2.1 Documento	12
1.2.2 Base de Datos Referativa (BDR)	13
1.2.3 Archivo Técnico	13
1.2.4 Nomencladores	13
1.2.5 ONRM	13
1.2.6 MINBAS	13
1.2.7 Geología	14
1.3 Objeto de estudio	14
1.3.1 Descripción General	14
1.3.2 Descripción actual del dominio del problema	16
1.4 Situación Problemática	17
1.5 Conclusiones parciales	17
CAPÍTULO2 TENDENCIAS Y TECNOLOGÍAS	18
2.1 Las Tecnologías de la Informática y las Comunicaciones (TIC) como pilar para el desarrollo	18
2.1.1 Definición	18
2.2 Aplicación Web como vía de solución	19
2.2.1 Servicios Web como herramienta de interconectividad	20
2.3 Tecnología Cliente Servidor	20
2.3.1 Características principales de la tecnología cliente-servidor	21
2.4 Lenguajes del lado del Servidor	21
2.4.1 ASP	22
2.4.2 PERL	22
2.4.3 PHP	23
2.4.4 Selección del lenguaje a utilizar	25
2.5 Entorno de desarrollo propuesto para una futura implementación de la aplicación.	25

2.6 Frameworks de desarrollo	26
2.6.1 ZEND Framework	26
2.6.2 Framework de desarrollo que se propone utilizar	27
2.7 Lenguajes del lado del cliente	28
2.7.1 JavaScript	29
2.7.2 HTML	29
2.8. Gestor de Base de Datos a utilizar	29
2.9 Servidores Web	31
2.9.1 Apache HTTP Server Project	31
2.9.2 JBoss	32
2.9.3 Microsoft Internet information Server (MIIS)	32
2.10 El Lenguaje Unificado de Modelado (UML) como herramienta para el modelado de la aplicación Web.	33
2.11 Metodologías de Desarrollo de Software	34
2.11.1 Proceso Unificado de Desarrollo de Software (RUP) como metodología a seguir.	35
2.12 Herramientas CASE (Computer-Aided Software Engineering)	36
2.12.2 Visual Paradigm como herramienta CASE	36
2.13 Conclusiones parciales	37
CAPÍTULO3	38
PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA	38
3.1 Modelo de Negocio	38
3.1.1 Reglas del Negocio	39
3.1.2 Actores y trabajadores del Negocio	39
3.1.3 Casos de Uso del Negocio	40
3.1.4 Diagrama de Casos de Uso del Negocio	40
3.1.5 Descripción textual de los Procesos del Negocio	41
3.1.6 Modelo de Objeto del Negocio	43
3.2 Modelo de Sistema	43
3.2.1 Requerimientos Funcionales	44
3.2.2 Requerimientos No Funcionales	45
3.2.3 Descripción de los actores	46
3.2.4 Casos de Uso del Sistema	46
3.2.5 Diagrama de Casos de Uso del Sistema	47
3.2.6 Descripción textual de los Casos de Uso del Sistema	47
3.3 Conclusiones parciales	52
CAPÍTULO4	53
CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA	53

4.1 Modelo de Análisis	53
4.1.1 Diagrama de Clases de Análisis	54
4.2 Diagramas de interacción	55
4.3 Modelo de Diseño	56
4.3.1 Patrones que utiliza el framework	57
4.3.2 Diagramas de Clases del Diseño	58
4.3.2 Diagrama de clases persistentes	63
4.4 Principios del Diseño	64
4.4.1 Estándares de la interfaz de la aplicación	65
4.6 Concepción general de la ayuda	65
4.7 Diseño de la Base de Datos	66
4.8 Modelo de Implementación	67
4.9 Modelo de Despliegue	67
4.10 Conclusiones parciales	68
CONCLUSIONES	69
RECOMENDACIONES	70
GLOSARIO DE TÉRMINOS	71
REFERENCIA BIBLIOGRÁFICA	74
BIBLIOGRAFÍA	75
ANEXOS	77

INTRODUCCIÓN

En vista del creciente desarrollo, tanto económico como científico, alcanzado por el país se hace necesario y beneficioso el control y gestión de la información y las comunicaciones.

Actualmente el uso de las *Tecnologías y las Comunicaciones* (TIC) es considerado como uno de los factores básicos de desarrollo en los países del primer mundo. No consiste únicamente en la incorporación de tecnologías, sino ha de ir más allá, debe ayudar a prever las necesidades del mercado y a detectar los nuevos productos, procesos y servicios de mayor calidad, generando nuevas prestaciones con el menor coste posible. Este uso también hace necesario la reacción ante los cambios que impone el mercado globalizado.

Para poder alcanzar una mayor productividad a través del desarrollo de las TIC se ha producido un cambio radical en los modelos económicos y de gestión de las empresas, anteriormente basados en la mano de obra y el capital, que han pasado, fundamentalmente en la segunda mitad de los años noventa, a establecer como factores críticos: la tecnología, el conocimiento, la formación y el capital intelectual, marcando la economía basada en el conocimiento.

Más aún, la economía basada en el conocimiento ya está pasando a ser la economía dirigida por el conocimiento, de forma que sea esta la herramienta fundamental de gestión del conjunto de los procesos de la empresa. Ello implica gestionar de forma eficiente una cantidad de información como jamás anteriormente había conocido la sociedad. Es precisamente el hecho de tener información adecuada, en el momento adecuado y a disposición de las personas adecuadas lo que diferencia y da ventajas a las empresas.

La industria cubana no está ajena a estos cambios tecnológicos y en aras de informatizar un campo tan importante como la Industria Básica se decide aplicar el uso de las Tecnologías y las Comunicaciones en dicho sector de la economía.

El Ministerio de la Industria Básica es el organismo encargado de dirigir, ejecutar y controlar la política del Estado y el Gobierno en cuanto a las actividades de:

- Generación, transmisión, distribución y comercialización de la energía eléctrica.
- Producción de la industria del combustible y los lubricantes, y su distribución.

- Búsqueda, exploración y extracción de petróleo y gas.
- Búsqueda, exploración, extracción y beneficio de minerales sólidos.
- Producción de fertilizantes, fibras químicas, gases industriales, artículos de plásticos, plaguicidas, pinturas, colorantes, barnices y otros productos químicos.
- Producción de neumáticos, cámaras y productos del caucho.
- Producción y conversión de papel, cartón y pulpa celulosa.
- Producción de vidrio y sus artículos.

Por tanto tiene bajo su responsabilidad tres importantes sectores de la economía cubana:

- Energía
- Geología y Minería
- Química Básica

El sector de la Geología y la Minería es atendido por dos organizaciones empresariales:

- Unión Geólogo Minera
- Unión del Níquel

Unión Geólogo Minera es la entidad estatal cubana encargada de la prospección, exploración y explotación de minerales metálicos y no metálicos, excepto el níquel, operando en la modalidad a riesgo con firmas extranjeras, enmarcándose áreas muy perspectivas fundamentalmente en oro y plata.

En los últimos 35 años ha ejecutado un intenso programa de mapeo geológico, investigaciones geoquímicas y geofísicas, con los siguientes resultados:

- Levantamiento geológico 1: 50 000 y 1: 10 000 del 50 % del territorio, donde se hallan las zonas más perspectivas.
- 100 % del territorio volado y con levantamientos aeromagnéticos¹, aerogravimétricos² y aerogamma-espectrométricos³.

¹ método muy confiable y preciso para detectar anomalías magnéticas intra-sedimentarias.

² método muy confiable y preciso para ubicar depósitos sedimentarios.

³ método para medir la concentración superficial de los tres elementos radioactivos más abundantes: uranio, thorio y potasio.

Se han establecido 37 regiones de mejores perspectivas para el descubrimiento de minerales metálicos, equivalente a 40 mil kilómetros que representa el 35% del territorio nacional.

En los no metálicos el país cuenta con grandes yacimientos de zeolita en el orden de 500 millones de toneladas de reservas, cuenta además con yacimientos de carbonato de calcio de alta pureza, idóneo para su producción en forma micronizada, con destino a las industrias del papel, plástico y pinturas.

La Oficina Nacional de Recursos Minerales (ONRM), está adscrita al Ministerio de la Industria Básica pero su esfera de influencia y acción se extiende a todos los órganos y organismos de la administración central del estado. La misma es la entidad que vela por el aprovechamiento racional de los recursos minerales del país y constituye el órgano que controla el proceso concesionario, ordenando y fiscalizando la actividad geológica, minera y petrolera de la República de Cuba.

Inicialmente la ONRM se encargaba de:

1. Archivar y conservar todos los documentos geológicos del país.
2. Presentar la información necesaria a los organismos que realizaban actividades geológicas, mineras y en general de la utilización de los recursos minerales.
3. Elaborar análisis críticos, informaciones y otros estudios, sobre trabajos geológicos efectuados hasta el momento en Cuba.
4. Hacer balances de todas las reservas de minerales del país.

En el 2001 el Comité Ejecutivo del Consejo de Ministros de Cuba aprobó las funciones de la Oficina Nacional de Recursos Minerales (acuerdo 3985), adscrita al Ministerio de la Industria Básica y son:

- Fiscalizar y controlar la actividad minera y el uso racional de los recursos minerales, según lo dispuesto en la legislación vigente.
- Fiscalizar y controlar las actividades de exploración-producción de hidrocarburos líquidos y gaseosos, según las disposiciones legales vigentes.
- Asesorar al Ministerio de la Industria Básica y demás organismos de la administración central del estado sobre las actividades mineras de exploración producción de hidrocarburos, sin perjuicio de sus debidas competencias.
- Responder por el registro minero y el registro petrolero y mantener actualizadas las anotaciones sobre las concesiones mineras, áreas mineras reservadas, los contratos de exploración-producción de petróleo, yacimientos, manifestaciones minerales y de hidrocarburos, áreas en investigación y minas y pozos de hidrocarburos en explotación o abandonados.

- Aprobar los cálculos de reservas minerales sólidos, líquidos y gaseosos y registrar y mantener actualizado el inventario de dichas reservas, así como certificar el grado de aprobación de dichas reservas para su asimilación industrial u otros fines.
- Ser el depositario de la información geológica, minera y petrolera de la nación, recibir, organizar y conservar la información, así como brindar servicios de información técnica.
- Proponer y en su caso dictar las disposiciones de carácter técnico que regulen la investigación, la explotación, el procesamiento y el uso de los recursos minerales.
- Mantener actualizadas las estadísticas mineras y de exploración-producción de petróleo y gas del país.
- Proponer y elaborar la documentación para la declaración de las áreas mineras reservadas.
- Rectorar la organización y división del territorio nacional y la zona económica exclusiva de Cuba en bloques o parcelas para la realización de trabajos de exploración-producción de petróleo y gas, certificando sus dimensiones y límites a los efectos de considerar los mismos como el objeto de los contratos petroleros que se negocien con personas extranjeras.
- Procesar y generalizar la información especializada conservada en su Archivo Técnico.

Para ello realiza un proceso lento y muy trabajoso a la hora de manejar y archivar los documentos que se generan de dichas investigaciones geológicas, ya que este, a parte de archivarse en la aplicación existente, se hace a mano sobre un libro, es decir, la información completa queda archivada física y digitalmente pero solo datos referativos de la misma, lo cual se hace difícil si el cliente es del exterior del país ya que debe presentarse en la oficina que radica en Ciudad de la Habana.

Con este trabajo se pretende analizar y realizar un diseño como propuesta de solución al sistema informático el cual optimice el trabajo con los datos geológicos que ingresan en la oficina y que cubra los servicios necesarios, así como la seguridad y confiabilidad de la información. Mejorando la aplicación existente en la oficina que no es la más indicada para estos tipos de funcionalidades.

De ahí, la importancia de este trabajo que ahorraría grandes sumas al país en cuestiones de dinero y tiempo, además de facilitar el intercambio científico entre homólogos de otros países que favorecerían el desarrollo científico. Además la ONRM contará con el diseño para la implementación de una poderosa herramienta de interconectividad entre cualquiera de sus módulos, permitiendo el flujo de información necesaria y vital para el trabajo conjunto dentro de la oficina. Esto permitirá tener la información actualizada rápidamente en el momento adecuado.

A partir de la descripción anterior, se arriba a la siguiente **situación problemática**:

La oficina cuenta con una aplicación y una base de datos deficiente ya que contiene datos inconsistentes. También debido al gran volumen de información esta no permite que funcione de manera óptima, además no cumple con uno de los principales servicios que pretende la ONRM siendo la interconectividad entre los módulos. Esto genera una pérdida de tiempo para el trabajo de la oficina, además de las gestiones de otros países en cuanto a inversiones se refiriere.

Esta situación conduce a formular el siguiente **problema**:

¿Cómo contribuir a automatizar de manera más eficiente la gestión de la información geológica de la Base de Datos Referativa?

La **idea a defender** que se plantea es que, con el Análisis y Diseño de una solución informática que contribuya al desarrollo de una aplicación Web, se podrá automatizar de manera más eficiente la gestión de la información geológica de la Base de Datos Referativa, a partir de la informatización del Archivo Técnico de la ONRM.

El **objeto de investigación** es el Análisis y Diseño de la gestión de la información geológica de la Base de Datos Referativa.

El **campo de acción** lo constituye la automatización de manera más eficiente de la gestión de la información geológica de la Base de Datos Referativa.

Del problema planteado anteriormente se puede devenir el siguiente **objetivo general**:

Analizar y diseñar una aplicación Web para automatizar de manera más eficiente la gestión de la información geológica de la Base de Datos Referativa.

Para dar cumplimiento a los objetivos antes trazados se llevarán a cabo las siguientes tareas:

- Investigar el funcionamiento del flujo de la información geológica almacenada en el Archivo Técnico de la ONRM, así como las características de dicha información.
- Analizar aspectos teóricos, conceptuales relacionados con la base de datos y aplicaciones Web.
- Estudiar las distintas metodologías de desarrollo de aplicaciones Web.
- Estudiar los lenguajes seleccionados tanto del lado del cliente como del lado del servidor.
- Rediseñar y optimizar la base de datos.
- Diseñar la aplicación Web para la gestión de datos sobre el Archivo Técnico.

CAPÍTULO 1

FUNDAMENTACIÓN TEÓRICA

En este primer capítulo se exponen las palabras claves y la clasificación de los conceptos necesarios para un mayor acercamiento al tema, enfatizando en las palabras técnicas y definiciones asociados al problema en cuestión. También se identifican los principales problemas que motivaron la investigación. Es en este capítulo donde se profundiza en el objetivo primario del trabajo, se esclarece con precisión la descripción actual del dominio del problema. Se analiza la solución existente en la Oficina Nacional de Recursos Minerales, sobre la cual está dirigido este trabajo en aras de mejorar la misma.

1.1 Conceptos Generales.

1.1.1 Sistema

Un sistema es un conjunto de "elementos" relacionados entre sí, de forma tal que un cambio en un elemento afecta al conjunto de todos ellos. Los sistemas reciben (entrada) datos, energía o materia del ambiente y proveen (salida) información, energía o materia.

Un sistema puede ser físico o concreto (una computadora, un televisor, un humano) o puede ser abstracto o conceptual (un software). Cada sistema existe dentro de otro más grande, por lo tanto un sistema puede estar formado por subsistemas y partes, y a la vez puede ser parte de un supersistema. Los sistemas tienen límites o fronteras, que los diferencian del ambiente. Ese límite puede ser físico (el gabinete de una computadora) o conceptual.[1]

Sistemas Conceptuales

Un *sistema conceptual* o *sistema ideal* es un conjunto organizado de definiciones, nombres, símbolos y otros instrumentos de pensamiento o comunicación. Ejemplos de sistemas conceptuales son las

Matemáticas, la Lógica formal, la Nomenclatura binomial ⁴ o la notación musical. O a una lengua, que es un sistema conceptual complejo en cuya aparición y evolución participan la biología y la cultura.

Sistemas Reales

Un *sistema real* es una entidad material formada por partes organizadas (o sus "componentes") que interactúan entre sí de manera que las propiedades del conjunto, sin contradecirlas, no pueden deducirse por completo de las propiedades de las partes. Tales propiedades se denominan propiedades emergentes.

Los sistemas reales intercambian con su entorno energía, información y, en la mayor parte de los casos, también materia. Una célula, un ser vivo, la Biosfera o la Tierra entera son ejemplos de sistemas naturales. El concepto se aplica también a sistemas humanos o sociales, como una sociedad entera, la administración de un estado, un ejército o una empresa.[2]

El esfuerzo por encontrar leyes generales del comportamiento de los sistemas reales es el que funda la Teoría de Sistemas y, más en general, aquella tendencia de la investigación a la que se alude como pensamiento sistémico o sistémica, en cuyo marco se encuentran disciplinas y teorías como la Cibernética⁵, la Teoría de la Información, la Teoría de Juegos, la Teoría del Caos y otras.

Después del análisis de estos diferentes criterios se llegó a la conclusión siguiente:

Un *sistema* es:

- Parte del universo (con una extensión limitada en espacio y tiempo).
- Existen mayor cantidad de correlaciones y correlaciones más fuertes entre una parte del sistema y otra; que entre esta parte del sistema y partes fuera del sistema. Se define parte como el resultado de dividir mentalmente al universo. Esta división se puede hacer en varios niveles. Una parte también resulta cuando se divide mentalmente a una parte.

⁴ convenio estándar utilizado en la biología para denominar las diferentes especies de organismos.

⁵ estudio del control y comunicación en los Sistemas Complejos: organismos vivos, máquinas y organizaciones.

1.1.2 Sistema Informático

Es un conjunto formado por componentes de cómputo, dígame elementos de hardware o software, con el fin de procesar datos que han sido almacenados.

Un sistema informático utiliza dispositivos programables por medio de computadoras, siendo una síntesis de hardware y software.

Como ejemplo de sistema informático se puede citar el caso de un individuo manejando una computadora personal a través de sus periféricos. También se clasifica como sistema informático a una computadora en sí misma, dado que los componentes de software y hardware deben operar sistémicamente.

Cuando varios sistemas informáticos se conectan entre sí, lógicamente se crea un sistema informático de mayor tamaño el cual comparte recursos de variado tipo, como por ejemplo impresoras por red, o conexión a Internet.

Los sistemas informáticos de importantes instituciones conectadas a Internet son generalmente fuertemente protegidos por expertos en seguridad informática, siendo atacados por crackers⁶ cotidianamente ya sea por intereses meramente didácticos, por intereses políticos o de otra clase. [3]

Después de analizar lo anteriormente planteado se define sistema informático como la estrecha relación que existe entre programas o aplicaciones y elementos físicos, dígame tanto materiales computacionales como el personal que los utiliza, de manera que satisfaga las peticiones para el cual fue creado.

1.1.3 Aplicación Informática

Son los programas con los cuales el usuario final interactúa, es decir, son aquellos programas que permiten la interacción entre el usuario y la computadora. Esta comunicación se lleva a cabo cuando el usuario elige entre las diferentes opciones o realiza actividades que le ofrece el programa.

Las aplicaciones forman parte del software de la computadora ya que son una serie de programas que han sido desarrollados para facilitarle al usuario la utilización de la máquina para una acción o fin determinados. Existen aplicaciones de distintos tipos, que dependen de la finalidad que tenga.

⁶ alguien que viola la seguridad de un sistema informático con fines de beneficio personal o para hacer daño.

De esta manera, se encuentran softwares de aplicaciones de negocios, de utilería, personales o de entretenimiento. Algunos ejemplos de aplicaciones son los procesadores de texto, las planillas de cálculo, los editores de gráficos, los administradores de bases de datos, entre otros. Además, las aplicaciones tienen que ser independientes del hardware de manera tal que puedan ser utilizadas por cualquier computadora. [4]

Según lo analizado se puede arribar a la conclusión de que aplicaciones informáticas son los programas que permiten la comunicación entre los usuarios y la computadora con el fin de facilitarle su uso.

1.1.4 Aplicación Web

Es un sitio que se encuentra en la red, al cual los usuarios acceden con el fin de beneficiarse con el uso de la información que posee. Aborda un determinado tema. Se dice que los usuarios se benefician con la información que estos les brindan porque una vez que acceden, pueden interactuar con él, actualizarlo o modificarlo según los privilegios de seguridad de la aplicación que estos tengan.

Una aplicación Web es un sistema informático que los usuarios utilizan accediendo a un servidor Web a través de Internet o de una intranet. Las aplicaciones Web son populares debido a la practicidad del navegador Web como cliente ligero. La facilidad para actualizar y mantener aplicaciones Web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su popularidad.

Gracias a la popularidad de Internet en el mundo concreto, se han abierto infinidad de posibilidades en cuanto al acceso a la información desde cualquier lugar o sitio Web. Esto implica un desafío a los desarrolladores de aplicaciones, ya que los avances en tecnología demandan cada vez aplicaciones más rápidas, ligeras y robustas que permitan utilizar la Web de una manera funcional e intuitiva.

1.2 Conceptos asociados al dominio del problema.

1.2.1 Documento

Se refiere al cualquier libro o informe existente en la oficina, los cuales son prestados, con el fin de ser consultados por personas o instituciones que lo requieran.

1.2.2 Base de Datos Referativa (BDR)

Es la herramienta para optimizar la búsqueda en el Archivo Técnico que posee algunos datos importantes de los documentos y la referencia a donde está guardado el documento físico.

1.2.3 Archivo Técnico

Archivo donde se conserva el patrimonio documental geológico, minero y petrolero del país y que está en función de dar servicio a toda la comunidad de las geociencias. La documentación del archivo representa la inversión efectuada por el Estado Cubano y contiene valiosa información de valor científico, histórico, docente, económico y estratégico en la esfera de la industria geólogo-minera y petrolera.

Los servicios que se ofrecen en el Archivo Técnico de la Oficina Nacional de Recursos Minerales están encaminados a todas las entidades o personas que tengan interés en informaciones históricas y actuales de la geología, minería y el petróleo de la República de Cuba.

1.2.4 Nomencladores

Terminología estándar que se accede a ella a través de un Servicio Web utilizada para brindar información a nuestra investigación.

1.2.5 ONRM

La Oficina Nacional de Recursos Minerales se creó en 1995 con la promulgación de la Ley de Minas la cual constituye la autoridad minera de Cuba. Hereda y acrecienta las funciones del Centro Nacional del Fondo Geológico, su precursor, el cual realizaba esta tarea desde su fundación en 1960. [5]

1.2.6 MINBAS

El Ministerio de la Industria Básica integra sectores estratégicos de la economía cubana y sus principales producciones y servicios se orientan tanto al mercado nacional como internacional.

Su misión es garantizar el desarrollo de las ramas que integra la Industria Básica para hacer sostenible el resto de las ramas del país y lograr el bienestar del pueblo cubano. [5]

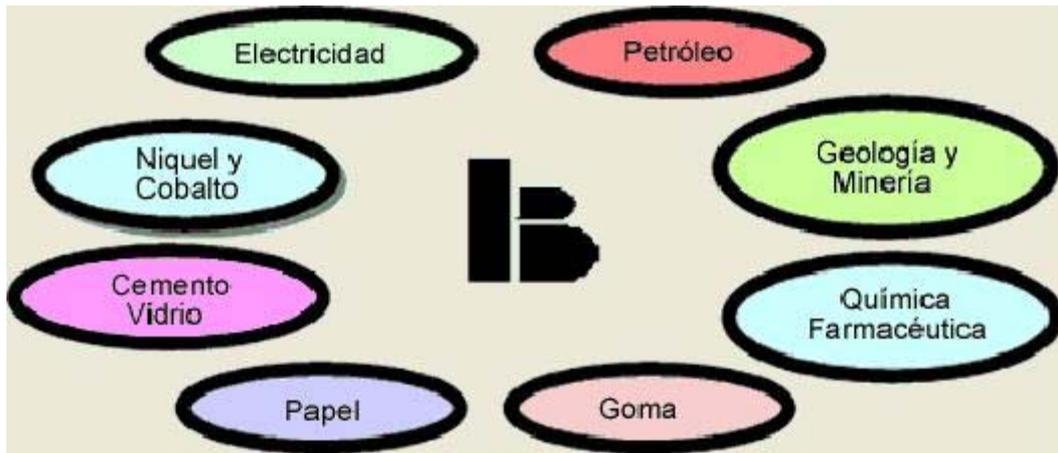


Figura 1: Sectores que integran la Industria Básica

1.2.7 Geología

Ciencia que estudia la constitución y origen de la Tierra y de los materiales que la componen interior y exteriormente.[6]

1.3 Objeto de estudio

Esta investigación está encaminada a mejorar el proceso de la gestión de la información geológica de la Base de Datos Referativa.

1.3.1 Descripción General

La Oficina Nacional de Recursos Minerales se creó en 1995 con la promulgación de la Ley de Minas. Está adscrita al MINBAS pero su esfera de influencia y acción se extiende a todos los órganos y organismos de la administración central del estado cubano.

La misma es la entidad que vela por el aprovechamiento racional de los recursos minerales del país y constituye el órgano que controla el proceso concesionario, ordenando y fiscalizando la actividad geológica, minera y petrolera de la República de Cuba.

A principio de 1997 se realiza la introducción masiva de los medios de computación lo que ocasionó un vuelco en la agilidad y efectividad del trabajo del Centro, así ya en este año se elaboró la base de datos digital del Balance y de las Concesiones Mineras.

Específicamente la BD de las Concesiones Mineras conjuntamente con una aplicación en Microsoft Access fue creada por un trabajador de la oficina, el Ingeniero Ramón Darío Infante Escalona. Desde su inicio esta herramienta sufrió varias versiones hasta la actualidad.

Mejóro el proceso de gestión de la información geológica ya que permitió guardar en una base de datos en Microsoft Access la información referativa de los documentos e investigaciones organizando así el trabajo para con dichos documentos generados por Cuba en los estudios y demás investigaciones que se realizan.

Con la realización de este trabajo se pretende proveer el diseño de una aplicación Web para la administración, comunicación, intercambio y publicación de información relacionada con el Sistema Concesionario de la República de Cuba, Política Minera, Archivo Técnico Geológico, Estadística Minera, Medio Ambiente, y Seguridad Minera, a la vez que ofrece vínculos a los principales proveedores de datos geológicos, mineros y de datos espaciales del país, utilizando para esto la tecnología Web.

Para lograr lo antes mencionado es necesario realizar un análisis profundo y posteriormente el diseño de una aplicación informática que conste con una base de datos, que referencie los documentos geológicos facilitando su búsqueda, que constituirá la entrada principal al conocimiento geológico que esta institución atesora por mandato del Estado, mientras no se constituya el Banco de Datos Geológicos de la República de Cuba como una entidad independiente. La misma deberá permitir la gestión de la información geológica en general dentro de la ONRM, comunidad de usuarios de la información geológica, organizaciones científicas, universitarias y de dirección del país para la toma de decisiones. Esta información se obtendrá a partir de los diversos proyectos y programas en ejecución por diferentes organizaciones.

El Diseño de dicha aplicación se encargará también de insertar, eliminar y modificar los datos referativos de los documentos e investigaciones geológicas que serán almacenados en el AT. Además posibilitará la comunicación y retroalimentación con todos los departamentos de la Oficina.

¿Cuál será la audiencia o usuarios del producto?

- Oficina Nacional de Recursos Minerales y empresas vinculadas al quehacer geológico, minero y petrolero del MINBAS.
- Usuarios relacionados con el servicio geológico nacional ajenos al MINBAS.
- Organizaciones universitarias y científicas.

- Organizaciones relacionadas con la Administración Central del Estado de la República de Cuba.

¿Cuáles son los usuarios más importantes?

Toda la comunidad geológica, minera y petrolera de Cuba. Trabajadores de la ONRM. Geólogos, mineros y otros especialistas de las empresas geológicas en general, incluyendo la actividad del petróleo. Institutos y centros de investigación vinculados a la actividad, profesores y estudiantes universitarios.

¿Qué imagen de la institución quiere proyectar en sus usuarios?

Estabilidad, eficiencia, confiabilidad, madurez técnica y científica, eficacia en la gestión, comprometimiento, seriedad e imparcialidad.

¿Qué motivará a estos usuarios a utilizar el producto Web que se le va a ofertar?

La inmediatez para el conocimiento de la información, la confiabilidad de sus contenidos, el ahorro de recursos y la posibilidad de obtener la información sin necesidad de trasladarse a la sede de la ONRM. Téngase presente que los usuarios de los servicios de la ONRM radican en todo el país.

1.3.2 Descripción actual del dominio del problema

En la ONRM, existe un Archivo Técnico donde se encuentra registrada toda la información correspondiente a las investigaciones geológicas hechas en Cuba desde su inicio como institución.

Esta información se recoge en el Libro de Registros, y dichos documentos pueden tener anexados: mapas, libretas, discos compactos y discos magnéticos. A las personas o entidades interesadas en la búsqueda de información, la entrega de algún documento o la solicitud de un préstamo, se les denominan clientes, los cuales para realizar estas gestiones deben presentarse personalmente en la Oficina. Estos procesos se realizan de la siguiente manera.

En el caso de que se genere un documento debido a una investigación realizada en la isla de Cuba, este se presenta en la oficina y se procede a ser almacenado en la aplicación, tomando del documento los datos principales, o sea, una referencia del mismo las cuales se guardan en la base de datos. Posteriormente el informe original se archiva en el almacén y se registra en el Libro de Registro. De esta manera se culmina el proceso.

Cuando se presenta el cliente para solicitar el préstamo de algún documento almacenado, se verifica si el cliente posee documentos en su poder, de ser afirmativo se le notifica que debe entregarlos y no se lleva a cabo el préstamo, de no poseer ninguno se busca en la Base de Datos Referativa dicho documento y si está disponible se efectúa el préstamo y se registran los datos del interesado. Cuando se devuelve se recoge el documento y se archiva.

En la manipulación de los documentos es inevitable el desgaste y deterioro de los mismos, enfatizando que existen archivos muy antiguos los cuales por su estado pueden ser irremplazables. Estos archivos contienen información de todas las investigaciones geológicas que datan desde las primeras realizadas de su tipo en Cuba, estos son consultados por personal técnico de todo el país y del mundo para realizar investigaciones; también pueden ser estudiantes o investigadores de otras ramas interesados en información.

Debido al creciente volumen de información almacenada y los problemas que existían en la ONRM para gestionarlos, surgió la necesidad de automatizar dicho proceso y para esto se creó una aplicación en Microsoft Access.

1.4 Situación Problemática

Actualmente esta aplicación en Microsoft Access que se creó contiene datos inconsistentes, debido a que contiene campos duplicados en gran medida por la falta de validación en el momento de su entrada. También posee datos innecesarios, así como tablas en la base de datos que no son imprescindibles. En la ONRM se requiere interactividad entre los módulos que presenta y con esta aplicación no se logra. Además no posee una interfaz de usuario amigable ya que este software (Access) no brinda al usuario todas las comodidades y funcionalidades que podría brindar una herramienta más poderosa y moderna como la que se propone como solución, también debido al gran volumen de información no permite que funcione de manera óptima porque la misma no soporta todo tipo de archivos a ser almacenados.

1.5 Conclusiones parciales

El estudio de los conceptos y la fundamentación teórica del trabajo permitió desarrollar el marco conceptual, así como, lograr un mayor entendimiento del dominio el problema. Después del análisis de la situación problemática se concluye que la aplicación existente no satisface todas las necesidades y servicios que demanda el trabajo en la Oficina Nacional de Recursos Minerales. La optimización de dicha aplicación sería beneficiosa para la gestión de la información geológica de Cuba.

CAPÍTULO 2

TENDENCIAS Y TECNOLOGÍAS ACTUALES A CONSIDERAR

En la nueva era de "la sociedad del conocimiento", la información y las comunicaciones, son factores extremadamente claves en los procesos de producción y creación de riqueza. Las TIC cumplen un papel determinante en esta nueva sociedad, por lo que los ciudadanos en general, especialmente en aquellas naciones que van a la vanguardia del desarrollo, han comprendido la convergencia de elementos que comienzan a desplegarse tratando por ende de organizar el contexto tecnológico para incrementar su productividad y bienestar.

Aparejado al desarrollo de las TIC las Aplicaciones Web han surgido en Internet para mejorar el intercambio de información y facilitar los servicios en las esferas económicas, socio-culturales, entre otras. La prestación de Servicios online ha revolucionado la manera en que el hombre se relacionaba entre si, incluso en la manera de pensar. Las herramientas para la creación de aplicaciones informáticas cada vez son más numerosas y complejas.

El desarrollo de los lenguajes de programación y su alto grado de complejidad han hecho que el hombre muy pronto sea capaz de modelar su entorno de manera digital y adaptarlo para su beneficio. Las metodologías de desarrollo han logrado que el trabajo sea mucho más simple y organizado. En el capítulo en cuestión se hace un análisis de las tecnologías y tendencias que existen en la actualidad a nivel mundial y que pudieran ser útiles en el desarrollo de la propuesta de solución. Además se tiene en cuenta la metodología utilizada para el desarrollo de la misma.

2.1 Las Tecnologías de la Informática y las Comunicaciones (TIC) como pilar para el desarrollo

2.1.1 Definición

Para lograr un mejor dominio de lo mencionado anteriormente respecto a las TIC es necesario definir qué significa este término y qué papel juega cuando se refiere al mismo.

Las TIC se encargan del estudio, diseño, desarrollo, fomento, mantenimiento y administración de la información por medio de sistemas informáticos, esto incluye todos los sistemas informáticos no solamente las computadoras, éstas son sólo un medio más, el más versátil, pero no el único; también las redes de telecomunicaciones, los teléfonos celulares, la televisión, la radio y los periódicos digitales. [7]

Las Tecnologías de la Informática y las Comunicaciones optimizan el manejo de la información y el desarrollo de la comunicación. Permiten actuar sobre la información y generar mayor conocimiento e inteligencia. Abarcan todos los ámbitos de la experiencia humana. Están en todas partes y modifican los ámbitos de la experiencia cotidiana: el trabajo, las formas de estudiar, las modalidades para comprar y vender, los trámites, el aprendizaje y el acceso a la salud, entre otros.

El comercio electrónico, que tiene que ver con el intercambio de bienes y servicios realizado gracias a un soporte de protocolos y plataformas digitales estandarizadas, permite llegar a acuerdos sin tener que estar presente, lo que constituye un gran ahorro en dinero y tiempo. Este es solo un ejemplo de cómo se manifiestan las TIC en la sociedad y como se pueden utilizar estos logros para el beneficio económico.

2.2 Aplicación Web como vía de solución

Para el desarrollo de la aplicación informática que se diseña en este trabajo es considerable utilizar una aplicación de tipo Web debido a lo siguiente:

Con el creciente desarrollo alcanzado por Internet las aplicaciones Web se han convertido en la mayor herramienta de intercambio de información que se utiliza actualmente en empresas y entidades, así como, en ramas tan importantes como el comercio internacional.

El acceso y manejo de la información en forma de aplicación Web asegura una manera rápida, sencilla y concisa de realizarse, donde el usuario no necesariamente debe conocer de procedimientos ni protocolos para ello. Algo muy necesario debido a la cantidad de información que debe ser registrada de los informes y documentos geológicos.

Además con una aplicación Web en la ONRM se garantizan algunos servicios importantes como: muestreo de inventarios, un acceso a la información adecuada en el momento necesario desde cualquier lugar, ya que con una aplicación de este tipo solo es necesario garantizar una conexión a Internet y un navegador Web, conexión con todos los módulos que operan en su interior garantizando el flujo de información rápido y simple.

Se pueden modificar los permisos de acceso de los usuarios, garantizando un fuerte control de la seguridad e integridad de la información en la ONRM, donde todos los trabajadores no tienen los mismos privilegios.

Permite actualizar la aplicación y la base de datos constantemente permitiendo que la información disponible sea siempre la óptima debido a que las mismas son muy fáciles de actualizar y darles mantenimiento. Estas actualizaciones son importantes pues muchas instituciones utilizan estos datos en su proceso de toma de decisiones.

2.2.1 Servicios Web como herramienta de interconectividad

Existen múltiples definiciones sobre lo que son los Servicios Web, lo que muestra su complejidad a la hora de dar una adecuada definición que englobe todo lo que son e implican. Una posible sería hablar de ellos como:

Un conjunto de aplicaciones o de tecnologías con capacidad para interoperar en la Web. Estas aplicaciones o tecnologías intercambian datos entre sí con el objetivo de ofrecer unos servicios. Los proveedores ofrecen sus servicios como procedimientos remotos y los usuarios solicitan un servicio llamando a estos procedimientos a través de la Web. [8]

¿Qué se logra con dichos servicios en la ONRM?

Estos servicios proporcionan mecanismos de comunicación estándares entre los diferentes módulos dentro de la ONRM, que interactúan entre sí para presentar información dinámica al usuario. Para proporcionar interoperabilidad y extensibilidad entre estos módulos, y que al mismo tiempo sea posible su combinación para realizar operaciones complejas.

2.3 Tecnología Cliente Servidor

Se hace necesario en el uso de aplicaciones Web tecnologías para el acceso y distribución de los servicios, tecnologías que permitan ejecutar y responder de forma segura y óptima las peticiones de los usuarios por muy complejas que estas puedan ser.

Debido al gran volumen de información que se pretende almacenar en la BDR y a los múltiples servicios que brinda la aplicación Web no solo internamente sino también para el servicio al público se utiliza la Tecnología Cliente Servidor.

Pero qué es en definitiva dicha tecnología.

La tecnología denominada Cliente-Servidor es utilizada por todas las aplicaciones de Internet/Intranet. La misma permite al usuario desde una máquina, llamada cliente, hacer un pedido de servicio a una máquina a la que está unida, llamada servidor, mediante una red ya sea local o externa. Estos servicios pueden ser peticiones de datos a una base de datos, de información contenida en archivos o los archivos en sí mismos o peticiones de imprimir datos en una impresora asociada.[9]

Un solo servidor puede atender a varios clientes y brindarles la información o pedidos requeridos por los mismos de manera rápida y simultánea, así se ahorra el problema de tener la información instalada y almacenada localmente.

2.3.1 Características principales de la tecnología cliente-servidor

La tecnología cliente – servidor cuenta con varias características dentro de las cuales cabe señalar las más importantes, que son:

- El Cliente y el Servidor pueden actuar como una sola entidad y también pueden actuar como entidades separadas, realizando actividades o tareas independientes.
- Las funciones del Cliente y el Servidor pueden estar en plataformas separadas, o en la misma plataforma.
- Un servidor da servicio a múltiples clientes en forma concurrente.
- Cada plataforma puede ser escalable independientemente. Los cambios realizados en las plataformas de los Clientes o de los Servidores, ya sean por actualización o por reemplazo tecnológico, se realizan de una manera transparente para el usuario final.

2.4 Lenguajes del lado del Servidor

Existe una multitud de lenguajes concebidos o no para Internet. Cada uno de ellos explota más a fondo ciertas características que lo hacen más o menos útiles para desarrollar distintas aplicaciones.

La versatilidad de un lenguaje está íntimamente relacionada con su complejidad. Un lenguaje complicado en su aprendizaje permite en general realizar un espectro de tareas más amplio y más profundamente. En el dominio de la red, los lenguajes de lado servidor más ampliamente utilizados para el desarrollo de páginas dinámicas son el ASP, PHP y PERL. [10]

A continuación se explican brevemente estos lenguajes anteriormente mencionados, haciendo énfasis en el lenguaje propuesto para el futuro desarrollo de la aplicación.

2.4.1 ASP

Microsoft introdujo este lenguaje llamado Active Server Pages en diciembre de 1996, por lo que no es nada nuevo. Es parte del Internet Information Server (IIS) desde la versión 3.0 y permite el uso de diferentes scripts y componentes en conjunto con el tradicional HTML para mostrar páginas generadas dinámicamente, traduciendo la definición de Microsoft: "Las Active Server Pages son un ambiente de aplicación abierto y gratuito en el que se puede combinar código HTML, scripts y componentes ActiveX del servidor para crear soluciones dinámicas y poderosas para el Web".

El principio de ASP es el VBScript, pero existe otra diversidad de lenguajes de programación que pueden ser utilizados como lo es Perl, JScript, entre otros.

Entre sus funciones principales están el acceso a base de datos, envío de correo electrónico, creación dinámica de gráficos y otros.

Para el desarrollo de las Active Server Pages es necesario un servidor con Windows NT ⁷4.x o mayor y el Internet Information Server. En caso del uso de un servidor Linux, Chilisoft⁸ ha desarrollado el Chilisoft ASP que también permite el uso del mismo. Otra opción es InstantAsp también para correr ASP en servidores que no sean el IIS. [11]

2.4.2 PERL

Perl es un acrónimo de Practical Extracting and Reporting Language, que viene a indicar que se trata de un lenguaje de programación muy práctico para extraer información de archivos de texto y generar informes a partir del contenido de los ficheros.

Es un lenguaje gratuito y anteriormente estaba muy asociado a la plataforma Unix, pero en la actualidad está disponible en otros sistemas operativos como Windows.

Perl es un lenguaje de programación interpretado, al igual que muchos otros lenguajes de Internet como JavaScript o ASP. Esto quiere decir que el código de los scripts en Perl no se compila sino que

⁷ familia de sistemas operativos producidos por Microsoft.

⁸ Chilisoft es una compañía que ofrece la tecnología ASP a través de Chilisoft ASP.

cada vez que se quiere ejecutar se lee el código y se pone en marcha interpretando lo que hay escrito. Además es extensible, ya que desde Perl se puede hacer llamadas a subprogramas escritos en otros lenguajes. También desde otros lenguajes se permite ejecutar código Perl. [10]

Perl está inspirado a partir de lenguajes como C, sh⁹ y awk¹⁰ (algunos provenientes de los sistemas Unix), pero está enfocado a ser más práctico y fácil que estos últimos.

Es por ello que un programador que haya trabajado con el lenguaje C y los otros, tendrá menos problemas en entenderlo y utilizarlo rápidamente. Una diferencia fundamental de Perl con respecto a los otros lenguajes es que no limita el tamaño de los datos con los que trabaja, el límite lo pone la memoria que en ese momento se encuentre disponible.

2.4.3 PHP

PHP es un lenguaje de programación interpretado usado normalmente para la creación de páginas Web dinámicas. PHP es un acrónimo recursivo que significa "PHP Hypertext Pre-Processor" (inicialmente PHP Tools, o, *Personal Home Page Tools*). [12]

Visión General

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones.

¿Cómo funciona?

Cuando el cliente hace una petición al servidor para que le envíe una página Web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. Mediante extensiones es también posible la generación de archivos PDF¹¹, Flash, así como imágenes en diferentes formatos.

⁹ lenguaje de programación de bajo nivel en UNIX (Shell).

¹⁰ lenguaje de programación diseñado para procesar datos basados en texto, ya sean ficheros o flujos de datos.

¹¹ formato de documento portable.

¿Qué ventajas presenta PHP?

- Es un lenguaje multiplataforma. PHP corre en casi cualquier plataforma utilizando el mismo código fuente, pudiendo ser compilado y ejecutado en algo así como 25 plataformas, incluyendo diferentes versiones de Unix, Windows (95,98,NT,ME,2000,XP) y Macs. Como en todos los sistemas se utiliza el mismo código base, los scripts pueden ser ejecutados de manera independiente al OS¹².
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad. Puede interactuar con muchos motores de bases de datos tales como MySQL, MS SQL, Oracle, DB2, PostgreSQL, y otros muchos. Siempre se puede disponer de ODBC para situaciones que lo requieran.
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos. PHP es Open Source, lo cual significa que el usuario no depende de una compañía específica para arreglar cosas que no funcionan, además no es necesario pagar actualizaciones anuales para tener una versión que funcione.
- Permite las técnicas de Programación Orientada a Objetos. Rapidez. PHP generalmente es utilizado como módulo de Apache, lo que lo hace extremadamente veloz. Está completamente escrito en C, así que se ejecuta rápidamente utilizando poca memoria.
- Biblioteca nativa de funciones sumamente amplia e incluida. Una gran variedad de módulos cuando un programador PHP necesite una interface para una librería en particular, fácilmente podrá crear una API para esta. Algunas de las que ya vienen implementadas permiten manejo de gráficos, archivos PDF, Flash, Cybercash¹³, calendarios, XML, IMAP, entre otros.
- No requiere definición de tipos de variables.
- Tiene manejo de excepciones.

Desventajas

- No posee una abstracción de base de datos estándar, sino bibliotecas especializadas para cada motor (a veces más de una para el mismo motor).
- No posee adecuado manejo de internacionalización, Unicode.
- Por su diseño dinámico no puede ser compilado y es muy difícil de optimizar.

¹² sistema operativo.

¹³ mecanismo de pago para procesar los pagos con tarjeta de crédito a través de Internet

- Por sus características promueve la creación de código desordenado y complejo de mantener.
- Está diseñado especialmente para un modo de hacer aplicaciones Web que es ampliamente considerado problemático y obsoleto (mezclar el código con la creación de la página Web).

2.4.4 Selección del lenguaje a utilizar

Se selecciona, como lenguaje a utilizar para la creación de la aplicación Web, PHP como la opción más eficaz debido a que se pueden lograr grandes funcionalidades con pocas líneas de código, lo que hace que merezca la pena aprenderlo. Además viene acompañado por una excelente biblioteca de funciones que permite realizar cualquier labor (acceso a base de datos, encriptación, envío de correo, gestión de un e-commerce¹⁴, entre otras).

Está siendo utilizado con éxito en varios millones de sitios Web. Es multiplataforma, por lo que no requiere de un OS específico para trabajar con él. Es libre y es precisamente el objetivo de Cuba emigrar a Software de este tipo. Se puede obtener en la Web y su código está disponible bajo la licencia GPL. Lo que garantizaría un ahorro para la economía cubana en caso de actualizaciones del producto.

2.5 Entorno de desarrollo propuesto para una futura implementación de la aplicación.

Para lograr una estabilidad en el trabajo, el programador utiliza un programa compuesto por un conjunto de herramientas conocido como entorno de desarrollo, el cual permite compilar y depurar el código brindando confiabilidad y comodidad. Para el desarrollo futuro de la aplicación se propone utilizar la herramienta Eclipse PDT.

Eclipse PDT es principalmente una plataforma de programación usada para crear entornos integrados de desarrollo (del Inglés IDE). Fue desarrollado originalmente por IBM como el sucesor de su familia de herramientas para VisualAge, ahora es desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto.[13]

Es gratuito y puede funcionar con varios lenguajes de programación como son Java, C, C++, PHP. El mismo cuenta con características destacables:

- Ayudas para escribir en ficheros HTML y CSS¹⁵.

¹⁴ comercio electrónico a través de Internet.

¹⁵ lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML.

- Soporte para Debugging.
- Opción a ver qué tipo de objetos o variables devuelve un método.
- Poder indicar si el código de un proyecto es PHP4 o PHP5.
- Posibilidad de crear templates generadores de código asociados con sus alias.
- Editor sensible al contexto, el cual provee resaltamiento de código, asistente de código y autocompletado de código.
- Soporte para el debugg incremental del código de PHP.

2.6 Frameworks de desarrollo

Actualmente este concepto es muy manejado y utilizado como herramienta de apoyo en la construcción de aplicaciones no solo Web.

Primeramente no es un software completo, es una estructura compuesta de componentes que puede ser configurables en la construcción de una aplicación.

Se recomienda la utilización de un framework como herramienta de apoyo debido a que soluciona algunas desventajas que posee el lenguaje que se propone. Además un framework simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. También proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener. Por último, un framework facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas.

2.6.1 ZEND Framework

Es un framework de aplicación Web Open Source y orientado a objeto, implementado en PHP 5 y licenciado bajo licencia nueva del BSD¹⁶. El mismo, más conocido como ZF se desarrolla con la meta de simplificar el desarrollo de la Web mientras que promueve las mejores prácticas en la comunidad del revelador de PHP. Permite que los reveladores reutilicen componentes cuando y donde tengan sentido en sus usos sin requerir otros componentes de ZF más allá de dependencias mínimas. No hay por lo tanto paradigma o patrón del desarrollo que todos los usuarios del framework de Zend deban seguir, aunque ZF proporciona los componentes para el MVC (Modelo Vista Controlador) y los patrones del diseño de la entrada de la tabla que se utilizan en la mayoría de los usos de ZF.

¹⁶ licencia de software perteneciente al grupo de licencias de software Libre.

También proporciona los componentes individuales para muchos otros requisitos comunes en el desarrollo del uso de la Web, incluyendo la autenticación y la autorización en las listas del control de acceso (ACL), configuración del uso, datos que depositan, filtrado y validación de usuario, la integridad de la seguridad y los datos, internacionalización, interfaces a la funcionalidad de AJAX, composición y entrega de e-mail, ajuste a formato, la indexación de direcciones de la búsqueda, y todos los datos APIs de Google junto con muchos otros servicios populares de la Web.[14]

- Cuenta con módulos para manejar archivos PDF, canales RSS, Web Services (Amazon, Flickr, Yahoo).
- También incluye objetos de las diferentes bases de datos, por lo que es extremadamente simple para consultar su base de datos, sin tener que escribir ninguna consulta SQL.
- Una solución para el acceso a base de datos que balancea el ORM con eficiencia y simplicidad.
- Un poderoso framework MVC.
- Completa documentación y tests de alta calidad.
- Soporte avanzado para i18n¹⁷.
- Un buscador compatible con Lucene.
- Robustas clases para autenticación y filtrado de entrada.
- Clientes para servicios Web, incluidos Google Data APIs y Strikelron.
- Muchas otras clases útiles para hacerlo tan productivo como sea posible.

2.6.2 Framework de desarrollo que se propone utilizar

Se utiliza Symfony por ser un completo framework diseñado para optimizar el desarrollo de las aplicaciones Web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación. 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. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación Web. [15]

Symfony está desarrollado completamente con PHP 5. Ha sido probado en numerosos proyectos reales y se utiliza en sitios Web de comercio electrónico de primer nivel. Es compatible con la

¹⁷ terminología utilizada para la internacionalización y localización de aplicaciones.

mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. Se puede ejecutar tanto en plataformas Unix, Linux, como en plataformas Windows.

Otras características fundamentales son:

Symfony se diseñó para que se ajustara a los siguientes requisitos:

- Fácil de instalar y configurar en la mayoría de las plataformas (y con la garantía de que funciona correctamente en los sistemas Windows y Unix estándares).
- Independiente del sistema gestor de bases de datos.
- Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.
- Basado en la premisa de “*convenir en vez de configurar*”, en la que el desarrollador solo debe configurar aquello que no es convencional.
- Sigue la mayoría de *mejores prácticas* y patrones de diseño para la Web.
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.
- La capa de presentación utiliza plantillas y *layouts*¹⁸ que pueden ser creados por diseñadores HTML sin ningún tipo de conocimiento del framework.
- Los *helpers* incluidos permiten minimizar el código utilizado en la presentación, ya que encapsulan grandes bloques de código en llamadas simples a funciones.
- Los formularios incluyen validación automatizada y relleno automático de datos (“*repopulation*”), lo que asegura la obtención de datos correctos y mejora la experiencia de usuario.
- La autenticación y la gestión de credenciales simplifican la creación de secciones restringidas y la gestión de la seguridad de usuario.

2.7 Lenguajes del lado del cliente

Los lenguajes de lado cliente (entre los cuales no sólo se encuentra el HTML sino también el Flash y el JavaScript los cuales son simplemente incluidos en el código HTML) son aquellos que pueden ser directamente “digeridos” por el navegador y no necesitan un pretratamiento.

Un lenguaje de lado cliente es totalmente independiente del servidor, lo cual permite que la página pueda ser albergada en cualquier sitio sin necesidad de pagar más ya que, por regla general, los

¹⁸ bocetos o maquetas bien acabados que sirven para presentar una aplicación Web

servidores que aceptan páginas con scripts de lado servidor son en su mayoría de pago o sus prestaciones son muy limitadas. Como ejemplos de lenguajes del lado del cliente se cuenta con:

JavaScript, HTML, entre otros.

2.7.1 JavaScript

Para describir mejor este lenguaje se puede decir que es utilizado para crear pequeños programas encargados de realizar acciones dentro del ámbito de una página Web. Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento. Su uso se basa fundamentalmente en la creación de efectos especiales en las páginas y la definición de interactividades con el usuario.

2.7.2 HTML

Este lenguaje es muy utilizado por los realizadores Web. Las siglas HTML vienen de (HyperText Markup Language - Lenguaje de Marcas de Hipertexto), indica al navegador donde colocar cada texto, cada imagen o cada video y la forma que tendrán estos al ser colocados en la página.

El lenguaje consta de etiquetas que tienen esta forma o <P>. Cada etiqueta significa una cosa, por ejemplo significa que se escriba en negrita (bold) o <P> significa un párrafo, <A> es un enlace, entre otras. Casi todas las etiquetas tienen su correspondiente etiqueta de cierre, que indica que a partir de ese punto no debe de afectar la etiqueta.

Por ejemplo se utiliza para indicar que se deje de escribir en negrita. Así que el HTML no es más que una serie de etiquetas que se utilizan para definir la forma o estilo que se quiera aplicar al documento.

2.8. Gestor de Base de Datos a utilizar

Los gestores de bases de datos son programas que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

El gestor de base de datos que se utiliza es PostgreSQL por la variedad de funciones que este brinda y que cubren las necesidades de almacenamiento para con los datos de los informes propiamente a almacenar. Ejemplo provee soporte para:

- Números de precisión arbitraria.
- Texto de largo ilimitado.
- Figuras geométricas (con una variedad de funciones asociadas).
- Direcciones IP (IPv4 e IPv6).
- Bloques de direcciones estilo CIDR.
- Direcciones MAC¹⁹.
- Arrays (arreglos).

Otro aspecto por el que se utiliza es que es relacional orientado a objetos. También como muchos otros proyectos Open Source²⁰, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group). Esto permite un mínimo costo de las actualizaciones.[16]

Otras dentro de las principales características que este posee están:

- Claves ajenas también denominadas Llaves Ajenas o Llaves Foráneas (foreign keys).
- Disparadores (triggers).
- Vistas.
- Herencia de tablas. [17]
- Alta concurrencia ya que permite que mientras un proceso escriba en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos.

Adicionalmente los usuarios pueden crear sus propios tipos de datos, los que pueden ser por completo indexables gracias a la infraestructura GiST de PostgreSQL. Algunos ejemplos son los tipos de datos GIS creados por el proyecto PostGIS.

¹⁹ dirección MAC (Media Access Control address o dirección de control de acceso al medio) es un identificador de 48 bits que corresponde de forma única a una tarjeta o interfaz de red.

²⁰ código abierto, término con el que se conoce al software distribuido y desarrollado libremente

Funciones

Bloques de código que se ejecutan en el servidor. Pueden ser escritos en varios lenguajes, con la potencia que cada uno de ellos da, desde las operaciones básicas de programación, tales como bifurcaciones y bucles, hasta las complejidades de la programación orientación a objetos o la programación funcional. Los disparadores son funciones enlazadas a operaciones sobre los datos.

PostgreSQL soporta funciones que retornan "filas", donde la salida puede tratarse como un conjunto de valores que pueden ser tratados igual a una fila retornada por una consulta (query en inglés).

2.9 Servidores Web

Un servidor Web es un programa que implementa el protocolo HTTP (Hypertext transfer Protocol). Este protocolo está diseñado para transferir lo que se conoce por hipertextos, páginas Web o páginas HTML (Hypertext markup language): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música.

Un servidor Web se encarga de mantenerse a la espera de peticiones HTTP llevada a cabo por un cliente HTTP que se conoce como navegador. El navegador realiza una petición al servidor y éste le responde con el contenido que el cliente solicita. El proyecto está apoyado por una red mundial de colaboradores.

2.9.1 Apache HTTP Server Project

Se utiliza el servidor HTTP Apache por ser software libre, de código abierto para plataformas Unix (BSD, GNU/Linux), Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1. [18].

Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: en 2005, Apache fue el servidor HTTP más usado, siendo el servidor empleado en el 48% de los sitios Web en el mundo.

Sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años. La arquitectura del servidor Apache es muy modular. El servidor consta de una sección core y diversos módulos que aportan mucha de la funcionalidad que podría considerarse básica para un servidor Web.

2.9.2 JBoss

JBoss es un servidor de aplicaciones J2EE de código abierto implementado en Java puro. Al estar basado en Java, JBoss puede ser utilizado en cualquier sistema operativo que lo soporte. Combinando una arquitectura orientada a servicios revolucionaria con una licencia de código abierto, JBoss puede ser descargado, utilizado, incrustado, y distribuido sin restricciones por la licencia. Por este motivo es la plataforma más popular de middleware²¹ para desarrolladores, vendedores independientes de software y, también, para grandes empresas. Tiene una calidad y rendimientos realmente buenos. [19]

Las características destacadas de JBoss incluyen:

- Producto de licencia de código abierto sin coste adicional.
- Cumple los estándares.
- Confiable a nivel de empresa.
- Incrustable, orientado a arquitectura de servicios.
- Flexibilidad consistente.
- Servicios del middleware para cualquier objeto de Java.
- Ayuda profesional de la fuente.
- Soporte completo para JMX²².

2.9.3 Microsoft Internet information Server (MIIS)

Microsoft Internet Information Services, MIIS, es una serie de servicios para los ordenadores que funcionan con Windows. Los servicios que ofrece son: FTP, SMTP, NNTP y HTTP/HTTPS. Este servicio convierte a un ordenador en un servidor de Internet o Intranet es decir que en las computadoras que tienen este servicio instalado se pueden publicar páginas Web tanto local como remotamente.[20]

²¹ software de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas.

²² tecnología que define una arquitectura de gestión de aplicaciones basadas en Java.

2.10 El Lenguaje Unificado de Modelado (UML) como herramienta para el modelado de la aplicación Web.

Los modelos son muy importantes en todas las disciplinas de la ingeniería, así como, en la solución de problemas cotidianos ya que describen el aspecto y la conducta de un elemento, dígase elemento como una unidad o como un conjunto. Los problemas más complicados y abstractos pueden ser modelados para su fácil entendimiento de manera exitosa.

"UML" son las siglas de Unified Modeling Language (Lenguaje Unificado de Construcción de Modelos), con que se construyen sistemas por medio de conceptos orientados a objetos. UML es un lenguaje para modelar, que es el procedimiento que emplean los ingenieros para el diseño de software antes de pasar a su construcción.[21]

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. Incluye aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. UML ayuda al usuario a entender la realidad de la tecnología y la posibilidad de que reflexione antes de invertir y gastar grandes cantidades en proyectos que no estén seguros en su desarrollo, reduciendo el coste y el tiempo empleado en la construcción de las piezas que constituirán el modelo.

Desde el punto de vista puramente tecnológico, UML tiene una gran cantidad de propiedades que han sido las que, realmente, han contribuido a hacer de UML el estándar de facto de la industria que es en realidad. Algunas de las propiedades de UML como lenguaje de modelado estándar son:

- Concurrencia, es un lenguaje distribuido y adecuado a las necesidades de conectividad actuales y futuras.
- Ampliamente utilizado por la industria desde su adopción por OMG²³.
- Reemplaza a decenas de notaciones empleadas con otros lenguajes.
- Modela estructuras complejas.
- Las estructuras más importantes que soportan tienen su fundamento en las tecnologías orientadas a objetos, tales como objetos, clase, componentes y nodos.
- Emplea operaciones abstractas como guía para variaciones futuras, añadiendo variables si es necesario.

²³ consorcio dedicado al cuidado y el establecimiento de diversos estándares de tecnologías orientadas a objetos.

Es importante resaltar que UML es un "lenguaje" para especificar y no para describir métodos o procesos. Se utiliza para definir un sistema de software, para detallar los artefactos en el Sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo. Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional), pero no especifica en sí mismo qué metodología o proceso usar.

2.11 Metodologías de Desarrollo de Software

Para la realización de cualquier proceso ya sea productivo, de aprendizaje, entre otros., el hombre se apoya en las metodologías para lograr una mejor planificación de su tiempo, del costo, de su trabajo en general, además de aumentar la calidad del resultado final de dicho proceso, requisitos muy requeridos hoy en día.

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. Se van indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además qué personas deben participar en el desarrollo de las actividades y qué papel deben de tener. Además detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

Existen varias metodologías a emplear para modelar la solución del problema seleccionado. Se puede mencionar como ejemplo la Metodología XP (Programación Extrema).

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo.

La misma se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios.

Esta metodología se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre.[22].

2.11.1 Proceso Unificado de Desarrollo de Software (RUP) como metodología a seguir.

Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software (véase Anexo 1), es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto.

El Proceso Unificado Racional (Rational Unified Process en inglés, habitualmente resumido como RUP) es un proceso de desarrollo de software que utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema software. Constituye la metodología estándar más utilizada para el Análisis, Implementación y documentación de sistemas orientados a objetos.[23]

RUP define claramente quien, cómo, cuándo y qué debe hacerse en el proyecto. Posee 3 características esenciales:

- Está dirigido por los Casos de Uso: que orientan el proyecto a la importancia para el usuario y lo que este quiere.
- Está centrado en la arquitectura: que relaciona la toma de decisiones que indican cómo tiene que ser construido el sistema y en qué orden.
- Es iterativo e incremental: donde divide el proyecto en miniproyectos donde los Casos de Uso y la arquitectura cumplen sus objetivos de manera más depurada.[24]

El ciclo de vida de RUP

RUP divide el proceso en 4 fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades. En las iteraciones de cada fase se hacen diferentes esfuerzos en diferentes actividades (véase Anexo 2)

- Inicio: Se hace un plan de fases, se identifican los principales Casos de Uso y se identifican los riesgos. Se define el alcance del proyecto.
- Elaboración: se hace un plan de proyecto, se completan los Casos de Uso y se eliminan los riesgos.
- Construcción: se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario.

- Transición: se instala el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados. [24]

2.12 Herramientas CASE (Computer-Aided Software Engineering)

El uso de las herramientas CASE (Ingeniería de Software Asistida por Ordenador) puede mejorar la productividad en el desarrollo de una aplicación de bases de datos. Y por productividad se entiende tanto la eficiencia en el desarrollo, como la efectividad del sistema desarrollado. La eficiencia se refiere al coste, tanto en tiempo como en dinero, de desarrollar la aplicación. La efectividad se refiere al grado en que el sistema satisface las necesidades de los usuarios. Para obtener una buena productividad, subir el nivel de efectividad puede ser más importante que aumentar la eficiencia. [25]

Rational Rose por ejemplo es una herramienta muy poderosa de modelado visual para el Análisis y Diseño de sistemas basados en objetos. Se utiliza para modelar un sistema antes de proceder a construirlo. Cubre todo el ciclo de vida de un proyecto:

- Concepción y formalización del modelo.
- Construcción de los componentes.
- Transición a los usuarios.
- Certificación de las distintas fases.

2.12.2 Visual Paradigm como herramienta CASE

La herramienta CASE a utilizar es Visual Paradigm ya que es muy profesional, soporta el ciclo de vida completo del desarrollo de software: Análisis y Diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML.

Entre sus características fundamentales se encuentran:

- Soporte de UML versión 2.1.
- Ingeniería inversa Java, C++, Esquemas XML, XML.NET exe/dll, CORBA IDL.
- Generación de código.

- Entorno todo-en-uno para la especificación de los detalles de los Casos de Uso, incluyendo la especificación del modelo general y de las descripciones de los Casos de Uso.
- Generación de bases de datos.
- Transformación de diagramas de Entidad-Relación en tablas de base de datos Ingeniería inversa de bases de datos desde Sistemas Gestores de Bases de Datos (DBMS) existentes a diagramas de Entidad-Relación.
- Generador de informes para generación de documentación.
- Distribución automática de diagramas.
- Reorganización de las figuras y conectores de los diagramas UML.
- Importación y exportación de ficheros XML.
- Integración con Visio.

Además, la herramienta es colaborativa, es decir, soporta múltiples usuarios trabajando sobre el mismo proyecto; genera la documentación del proyecto automáticamente en varios formatos como Web o .pdf, y permite control de versiones. Cabe destacar igualmente su robustez, usabilidad y portabilidad. En definitiva, Visual Paradigm es una herramienta muy a tener en cuenta a la hora de realizar un proyecto importante.

2.13 Conclusiones parciales

En este capítulo se puede llegar a la conclusión que las TIC desempeñan un papel muy importante dentro del desarrollo y el progreso del país. También se puede arribar a la conclusión de que el lenguaje PHP utilizado cubre las necesidades para la realización de la aplicación Web SGAT (Sistema de Gestión del Archivo Técnico). Además, la herramienta gestora de base de datos (PostgreSQL) es ideal para satisfacer los requisitos necesarios en cuanto a almacenamiento de los archivos geológicos se refiere y para la organización y planificación de la aplicación, la metodología de desarrollo RUP es la opción más factible. También se considera utilizar un framework de desarrollo (Symfony) para la simplificación del trabajo.

CAPÍTULO 3

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

El primer paso en el proceso de desarrollo de software es alcanzar cierto nivel de conocimientos sobre el problema en cuestión.

En la fase de inicio se define la viabilidad del sistema, porque es aquí donde se realiza un análisis para decidir si se sigue adelante con el proyecto (Análisis del Negocio). Lograr una comunicación efectiva entre los usuarios y el equipo del proyecto, así como también entre los miembros de este último, con el objetivo de llegar a un entendimiento de lo que hay que hacer, es la clave del éxito en la producción del software.

En el presente Capítulo se describe la propuesta de solución, se realiza un análisis de los procesos que se emplean en la ONRM. También, se describen los procesos del Negocio que tienen que ver con el objeto de estudio, primeramente se modela el Negocio, se identifican los actores, trabajadores, los Casos de Uso correspondientes y la descripción de los mismos.

También se enumeran los requisitos; tanto *funcionales* como *no funcionales* que debe cumplir el sistema, derivando la identificación de las relaciones de los actores que interactúan con el sistema y las secuencias de acciones con las que interactúan, mediante un Diagrama de Casos de Uso.

3.1 Modelo de Negocio

La puesta en práctica de las técnicas de modelado que propone UML permiten un mejor entendimiento de cómo funciona el proceso de gestión de la información geológica en la Oficina Nacional de Recursos Minerales. El paso más importante dentro de dicho modelado es la identificación de los procesos del Negocio o Casos de Uso del Negocio. La obtención de los mismos es sumamente importante ya que establece las metas y límites de modelado del Negocio.

3.1.1 Reglas del Negocio

Las Reglas de Negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del Negocio.

Las reglas que se definen en el SGAT son:

1. El administrador es la persona encargada de gestionar los usuarios que operan con la aplicación y la base de datos.
2. El administrador es el encargado de introducir los datos de los informes a archivar.
3. El administrador gestiona los documentos ha ser prestados.
4. El usuario no puede realizar un pedido si este debe algún documento.
5. El informe a archivar debe ser previamente certificado por alguna institución.

3.1.2 Actores y trabajadores del Negocio

Un actor del Negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el Negocio interactúa.

A continuación se define el actor del Negocio en la siguiente tabla:

Actor	Descripción
Cliente	Este rol lo juega cualquier persona que tenga interés de consultar información geológica contenida en el Archivo Técnico, también puede ser un representante de una entidad concesionaria con un documento para ser registrado en la BD.

Tabla 1: Actor del Negocio

Un trabajador del Negocio es la persona o sistema que, vinculada a uno o más procesos del Negocio, participa en ellos interactuando directamente con la aplicación. El trabajador que se define en el Modelo de Negocio propuesto es el siguiente:

Trabajador	Descripción
Administrador	Es quien gestiona la información en la BDR.

Tabla 2: Trabajador del Negocio

3.1.3 Casos de Uso del Negocio

Los Casos de Uso que surgen después de la definición del Negocio lo constituyen la entrega de los informes que surgen a partir de investigaciones geológicas, la realización de los préstamos de documentos geológicos, así como, la devolución de los mismos.

3.1.4 Diagrama de Casos de Uso del Negocio

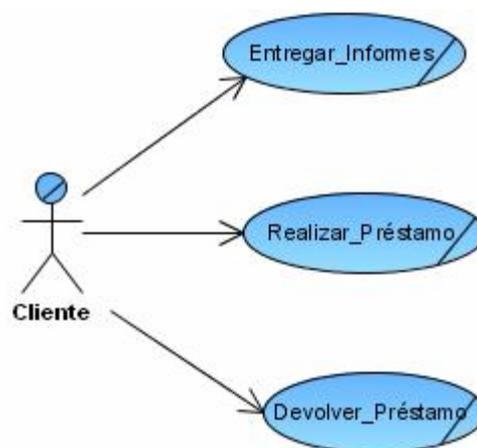


Figura 2: Modelo de CU del Negocio

3.1.5 Descripción textual de los Procesos del Negocio

Descripción del CU Entregar_Informes:

Caso de uso del Negocio	CU Entregar_Informes
Actores	Cliente
Resumen	El CU se inicia cuando el cliente trae el documento de un estudio realizado, para que este sea archivado. El administrador verifica el documento y procede a archivarlo.
Casos de Uso asociados	
Acción del actor	Respuesta del proceso de Negocio
1.-El cliente se presenta en la ONRM y entrega el documento de un estudio realizado.	1.1- El administrador procede a archivar dicho documento en la base de datos correspondiente.
	1.2- El administrador hace un registro en el libro de la información entrada.
Otras secciones	
Mejoras propuestas	

Tabla 3: Descripción del CU Entregar_Informes

Descripción del CU Realizar_Préstamo:

Caso de uso del Negocio	CU Realizar_Préstamo
Actores	Cliente
Resumen	El CU se inicia cuando el cliente solicita consultar un documento. Si el préstamo es interno se le entrega el documento, si no, se verifica que no debe documentos y se procede al préstamo.

Casos de Uso asociados	
Acción del actor	Respuesta del proceso de Negocio
1.-El cliente se presenta en la ONRM y solicita un documento.	1.1- El administrador pide la carta de autorizo.
	1.2- El administrador le pide identificación.
	1.3- El administrador procede a buscar el documento para ver si existe.
	1.4- El administrador pregunta si es un préstamo interno o externo.
Sección “Realizar Préstamo Interno”	
	1.5- El administrador entrega el documento.
	1.6- El administrador modifica el libro de préstamos.
Sección “Realizar Préstamo Externo”	
	1.7- El administrador verifica si debe algún documento.
	1.8- Si debe documento el administrador le notifica que debe y no puede prestar.
	1.9- Si no debe se procede a realizar el préstamo y se modifica el libro de préstamos.
Mejoras propuestas	

Tabla 4: Descripción del CU Realizar_Préstamo

Descripción del CU Devolver_Préstamo:

Caso de uso del Negocio	CU Devolver_Préstamo
Actores	Cliente
Resumen	El CU se inicia cuando el cliente trae el

	documento para que este sea devuelto. El administrador procede a archivarlo nuevamente.
Casos de Uso asociados	
Acción del actor	Respuesta del proceso de Negocio
1.-El cliente se presenta en la ONRM y entrega el documento.	1.1- El administrador le pide el CI del cliente.
2.- El cliente entrega el CI.	2.1- El administrador procede a archivar el documento nuevamente.
Mejoras propuestas	

Tabla 5: Descripción del CU Realizar_Préstamo

3.1.6 Modelo de Objeto del Negocio

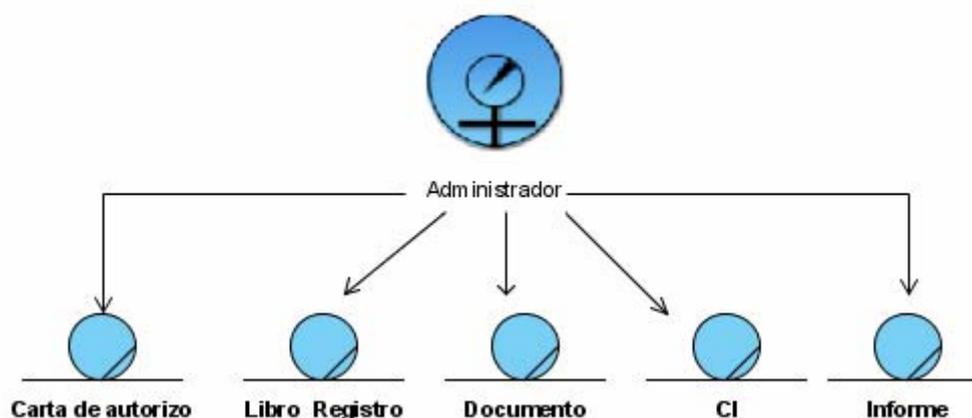


Figura 3: Modelo de Objeto de Negocio

3.2 Modelo de Sistema

La captura de requisitos y el modelamiento del Sistema son las actividades fundamentales que se desarrollan en el *Flujo de Requerimientos* durante la Fase de Inicio del desarrollo de un software. Una vez definido el Negocio se comienza a modelar el Sistema que se va construir, y de paso se

analiza ¿qué condiciones debe cumplir para que se venzan los objetivos planteados al inicio de este trabajo?

3.2.1 Requerimientos Funcionales

Los requisitos funcionales son capacidades o condiciones que el sistema debe cumplir.

El sistema debe permitir:

R1. Registrar Informe.

Con este requisito se garantiza que el usuario con derecho de administración en la base de datos pueda introducir los datos más significativos de un documento geológico.

R2. Actualizar Informe.

Con este requisito se garantiza que el usuario pueda realizar cualquier cambio o modificación en la información de un documento existente en la BD.

R3. Eliminar Informe.

Con este requisito se garantiza que el usuario pueda eliminar un informe existente en la BD.

R4. Registrar Préstamo del Informe.

Esta funcionalidad permitirá registrar cualquier préstamo de informes que se realice en la oficina, para guardar los datos de los usuarios que solicitaron documentos. Este registro de préstamos se especifica en caso que sea externo o interno.

R5. Eliminar Préstamo de Informe.

Implementando esta funcionalidad se podrá eliminar los registros correspondientes a préstamos que hayan terminado ocurriendo una devolución del documento.

R6 Modificar Préstamo.

Esta funcionalidad permitirá modificar en caso necesario los registros correspondientes a préstamos.

3.2.2 Requerimientos No Funcionales

Son propiedades o cualidades que el producto debe tener. Las mismas son características que hacen al producto amigable, atractivo, rápido y confiable.

R7. Usabilidad

R7.1. Los usuarios del sistema pueden hacer uso de la aplicación sin complicaciones porque ha sido diseñado para un manejo eficaz.

R7.2. El sistema debe poder usarse desde Internet.

R7.3. Los usuarios pueden acceder al sistema desde cualquier PC con acceso a Internet.

R8. Fiabilidad

R8.1. La confiabilidad y seguridad del software está dada en que el software presenta un sistema que permite al usuario registrarse, además hay una diferencia de los permisos de los mismos. Solo el administrador debe introducir y modificar los datos.

R8.2 El sistema debe tener las características de no ser vulnerable a partir de ataques como inyección de código u otras que puedan poner en riesgo la base de datos.

R9. Hardware

Para poder usar el sistema se debe contar con un ordenador que tenga al menos 256 Mb de memoria RAM, un nivel de procesamiento normal que puede ser incluso bajo, un monitor que soporte resolución 800 X 600, disponer de al menos una conexión a Internet o a la red local donde se instale el sistema con una velocidad no menos de los 56Kb/s.

R10. Software

Es necesario tener un sistema operativo que pueda tener instalados navegadores como Internet Explorer, o Mozilla Firefox.

R11. Soporte

R11.1. El software podrá operar en cualquier sistema operativo debido a que se desarrollará en herramientas de software libre como es el caso de PostgreSQL y PHP.

El ambiente de desarrollo donde se implementará dicho software será:

R11.2. Gestor de Base de Datos: PostgreSQL.

R11.3. Lenguaje de Programación: Java Script, PHP.

R11.4. Servidor de PHP: Apache.

R11.5. IDE de Desarrollo: EclipsePDT.

R11.6. Framework: Symfony.

R12. Interfaces de usuario

La interfaz de usuario de la aplicación debe ser de fácil comprensión y utilización, además amigable y que pueda ser utilizada por cualquier tipo de usuario.

3.2.3 Descripción de los actores

Actor	Descripción
Administrador	Persona autorizada para gestionar los informes y los préstamos que se manejan en la BD.

Tabla 6: Actor del Sistema

3.2.4 Casos de Uso del Sistema

Después de agrupar y analizar los requisitos funcionales se obtienen los Casos de Usos del Sistema: gestionar informes que agrupa registrar un nuevo informe, actualizar y eliminar un informe existente; el caso de uso realizar préstamo que agrupa registrar, modificar y eliminar un préstamo; por último el caso de uso devolver préstamo.

3.2.5 Diagrama de Casos de Uso del Sistema

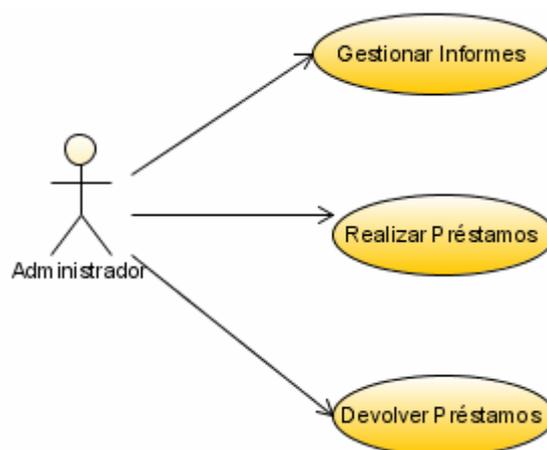


Figura 4: Diagrama de CU del Sistema

3.2.6 Descripción textual de los Casos de Uso del Sistema

Descripción del Caso de Uso Gestionar Informes:

Caso de Uso:	Gestionar Informes
Actores:	Administrador
Resumen:	El CU se inicia cuando el administrador ingresa, elimina o modifica un informe.
Precondiciones:	Que exista un informe que registrar, modificar o eliminar.
Referencias	R1, R2, R3 (listado de requisitos funcionales)
Prioridad	CU crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del sistema
1.- El administrador accede a la interfaz para agregar, eliminar o modificar un informe.	1.1.- El sistema muestra todos los informes. a) Si se desea agregar un nuevo informe, ver Sección1: "Ingresar Informe". b) Si se desea modificar un informe, ver

	Sección 2: “Modificar Informe”. c) Si se desea eliminar un informe, ver Sección 3: “Eliminar Informe”.
Sección 1: “Ingresar Informe”	
1.- El administrador selecciona introducir un nuevo informe. 2.- El administrador inserta un nuevo informe.	1.1.- El sistema muestra el formulario para ingresar un nuevo informe. 2.1.- El sistema lo adiciona en la base de datos y muestra el listado actualizado.
Sección 2: “Modificar Informe”	
1.- El administrador selecciona un informe para ser modificado. 2.- El administrador modifica el informe y hace clic en actualizar.	1.1.- El sistema muestra el formulario para modificar el informe. 1.2.- El sistema actualiza el informe en la base de datos y muestra un listado actualizado de los informes.
Sección 2: “Eliminar Informe”	
1.- El administrador selecciona el informe a eliminar y ejecuta dicha operación.	1.1.- El sistema elimina de la BD dicho informe. 1.2.- El sistema muestra un listado actualizado de los informes.
Flujos Alternos	
Acción del Actor	Respuesta del sistema
1- El administrador introduce datos incorrectos.	1.1.- El sistema muestra un mensaje de error correspondiente.
Prototipo de Interfaz	
Postcondiciones	

Tabla 7: Descripción del Caso de Uso del Sistema Gestionar Informes

Descripción del Caso de Uso Realizar Préstamo.

Caso de Uso:	Realizar Préstamo
Actores:	Administrador
Resumen:	El CU se inicia cuando el administrador ingresa, modifica o elimina un préstamo.
Precondiciones:	Que exista un préstamo que registrar, modificar o eliminar.
Referencias	R5, R6, R7 (listado de requisitos funcionales)
Prioridad	CU crítico

Flujo Normal de Eventos

Acción del Actor	Respuesta del sistema
1.- El administrador accede a la interfaz para agregar, modificar o eliminar un préstamo.	<p>1.1.- El sistema muestra un listado de los préstamos registrados.</p> <p>a) Si se desea agregar un nuevo préstamo, ver Sección1: Ingresar Préstamo.</p> <p>b) Si se desea modificar un préstamo, ver Sección 2: Modificar Préstamo.</p> <p>c) Si se desea eliminar un préstamo, ver Sección 3: Eliminar Préstamo.</p>

Sección 1: “Ingresar Préstamo”

1.- El administrador solicita la carta de autorización y el carnet de identidad.	
2.- El administrador solicita los datos del documento y los introduce en el sistema para comprobar si no está prestado.	2.1.- El sistema muestra todos los documentos filtrados por los datos que se introdujeron con su estado correspondiente (Prestado o Devuelto).
3.- Si el documento está disponible el administrador solicita los documentos personales y se los introduce al sistema, sino	3.1.- El sistema muestra un listado de los préstamos asociados al usuario

<p>está disponible no puede proceder a prestar.</p> <p>4.- El administrador pregunta si el préstamo es interno o externo, en caso de ser interno, accede al formulario ingresar préstamo e introduce los datos correspondientes.</p> <p>5.- En caso de ser externo el administrador verifica que el usuario no debe ningún documento previo y accede al formulario insertar préstamo e introduce los datos correspondientes, en caso de deber algún documento se le notifica al usuario y no se lleva a cabo el préstamo.</p>	<p>introducido.</p> <p>4.1.- El sistema agrega un nuevo préstamo en la BD y actualiza la lista de préstamos.</p> <p>5.1.- El sistema agrega un nuevo préstamo en la BD y actualiza la lista de préstamos.</p>
Sección 2: “Modificar Préstamo”	
<p>1.- El administrador selecciona un préstamo a modificar.</p> <p>2.- El administrador modifica el préstamo y hace clic en actualizar.</p>	<p>1.1.- El sistema muestra el formulario para modificar el préstamo.</p> <p>1.2.- El sistema actualiza el préstamo en la base de datos y muestra un listado actualizado de los préstamos.</p>
Sección 3: “Eliminar Préstamo”	
<p>1.- El administrador selecciona el préstamo a eliminar y ejecuta dicha operación.</p>	<p>1.1.- El sistema elimina de la BD dicho préstamo.</p> <p>1.2.- El sistema muestra un listado actualizado de los préstamos.</p>
Flujos Alternos	
Acción del Actor	Respuesta del sistema
<p>2- Si el administrador introduce datos incorrectos.</p>	<p>2.1.- El sistema muestra un error correspondiente.</p>
Prototipo de Interfaz	
Postcondiciones	

Tabla 8: Descripción del Caso de Uso del Sistema Realizar Préstamo

Descripción del Caso de Uso Devolver Préstamo.

Caso de Uso:	Devolver Préstamo	
Actores:	Administrador	
Resumen:	El CU se inicia cuando el administrador concluye un préstamo.	
Precondiciones:	Que exista un préstamo.	
Referencias	R6 (listado de requisitos funcionales)	
Prioridad	CU crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del sistema	
<p>1.- El administrador accede a la interfaz para concluir un préstamo.</p> <p>2.- El administrador solicita los documentos personales (Carnet de Identidad) y se los introduce al sistema.</p> <p>3.- El administrador selecciona el documento que se entrega y se cambia el estado del mismo de prestado a devuelto.</p>	<p>1.1.- El sistema muestra dicha interfaz.</p> <p>2.1.- El sistema muestra si existen préstamos asociado al usuario.</p> <p>3.1.- El sistema actualiza en la BD.</p>	
Flujos Alternos		
Acción del Actor	Respuesta del sistema	
1. Si el Administrador no introduce algún dato.	1.1- El sistema muestra un mensaje donde informe que se deben llenar correctamente los campos.	
Prototipo de Interfaz		
Postcondiciones		

Tabla 9: Descripción del Caso de Uso del Sistema Devolver Préstamo

3.3 Conclusiones parciales

En el presente capítulo se ha logrado describir a través de los diferentes conceptos, la información que manejará el sistema. Se logra una descripción detallada de los procesos del Negocio y del Sistema que se identifican. En el mismo se obtiene una primera visión de cómo quedaría el Diseño de la aplicación. Por todo lo expuesto anteriormente se concluye que los Modelos de Negocio y de Sistema que se obtuvieron en el presente capítulo satisfacen los objetivos por los que fueron creados y sirven como pilar para las siguientes fases de desarrollo.

CAPÍTULO 4

CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

El Análisis es una etapa muy importante en el desarrollo de la solución. Es aquí donde se realiza una distinción y separación completa de las partes del todo hasta llegar a conocer sus principios o elementos. Se examina detalladamente los hechos para conocer sus elementos constitutivos, sus características representativas, así como sus interrelaciones y la relación de cada elemento con él. Se analizan los requisitos que se describieron en la captura de requisitos, refinándolos y estructurándolos.[24]

Se define el diseño, como un proceso o labor a proyectar, coordinar, seleccionar y organizar un conjunto de elementos para producir y crear objetos.

En el Diseño se modela el sistema y encuentra su forma (incluida la arquitectura) para que soporte todos los requisitos (incluyendo los no funcionales y otras restricciones). El Diseño es el centro de atención al final de la Fase de Elaboración. Esto contribuye a una arquitectura estable y sólida y a crear un plano del Modelo de Implementación. [24]

En este capítulo se realiza el Análisis y Diseño de la propuesta de solución, a través de los Flujos de Trabajo de Análisis y Diseño e Implementación. Se presenta el Diagrama de Clases del Análisis y del Diseño de los diferentes Casos de Usos definidos en el capítulo anterior. Se muestra el modelo lógico y físico de datos, se especifican los Principios de Diseño así como una concepción general de la ayuda. Finalmente se realiza el Modelo de Despliegue donde se representan los nodos en los que se distribuye la aplicación y el Diagrama de Componentes para una mejor descripción de la solución propuesta.

4.1 Modelo de Análisis

El Modelo de Análisis ayuda a refinar y estructurar los requisitos centrado en el mantenimiento, en aspectos tales como la flexibilidad ante los cambios y la reutilización. Permite razonar sobre los

aspectos internos del sistema, incluidos sus recursos compartidos internos. Además ofrece un mayor poder expresivo y una mayor formalización.

Por lo tanto el Modelo de Análisis es importante por los siguientes motivos:

- Ofrece una especificación más precisa de los requisitos que la que se obtuvo del levantamiento de los mismos.
- El Modelo de Análisis se describe utilizando el lenguaje de los desarrolladores y puede por tanto introducir un mayor formalismo y ser utilizado para razonar sobre los funcionamientos internos del sistema.
- Estructura los requisitos de un modo que facilita su comprensión, su preparación, su modificación y su mantenimiento.
- Se considera como una primera aproximación al Modelo de Diseño y es por tanto una entrada fundamental para los procesos que le siguen (el Diseño y la Implementación).

4.1.1 Diagrama de Clases de Análisis

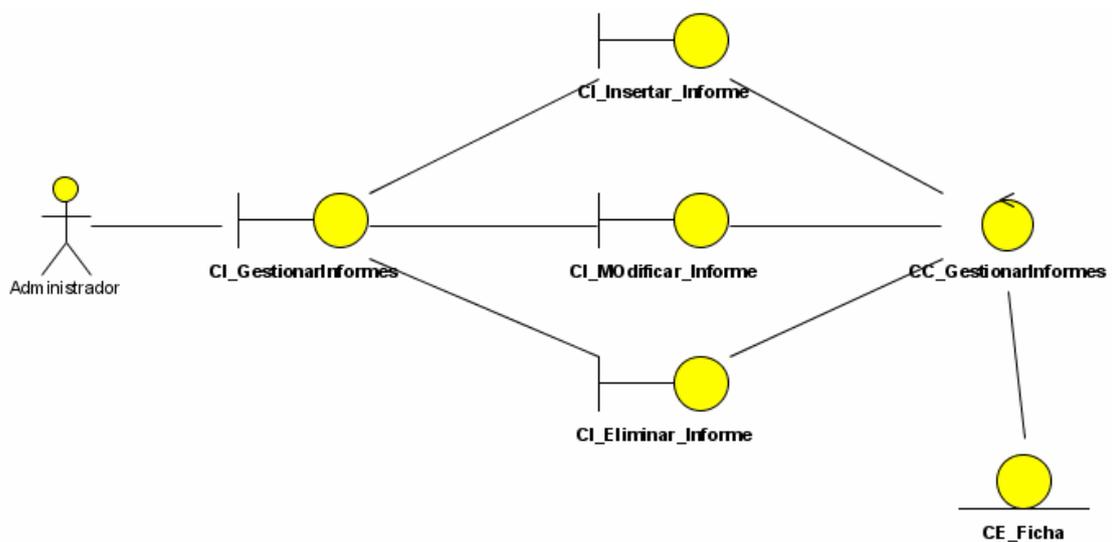


Figura 5: Diagrama de Clases de Análisis Caso Uso Gestionar Informes

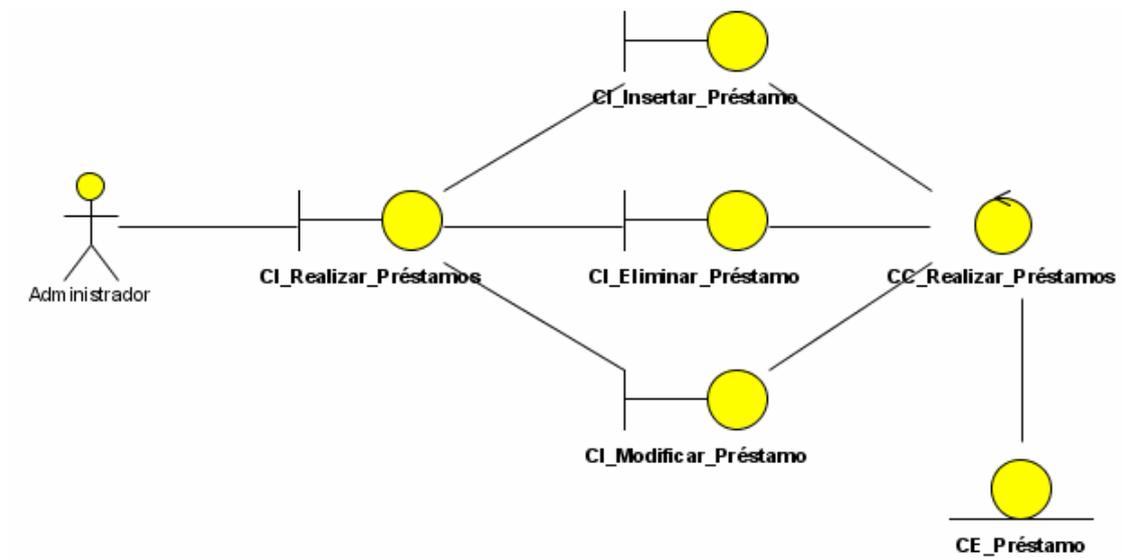


Figura 6: Diagrama de Clases de Análisis Caso Uso Realizar Préstamos

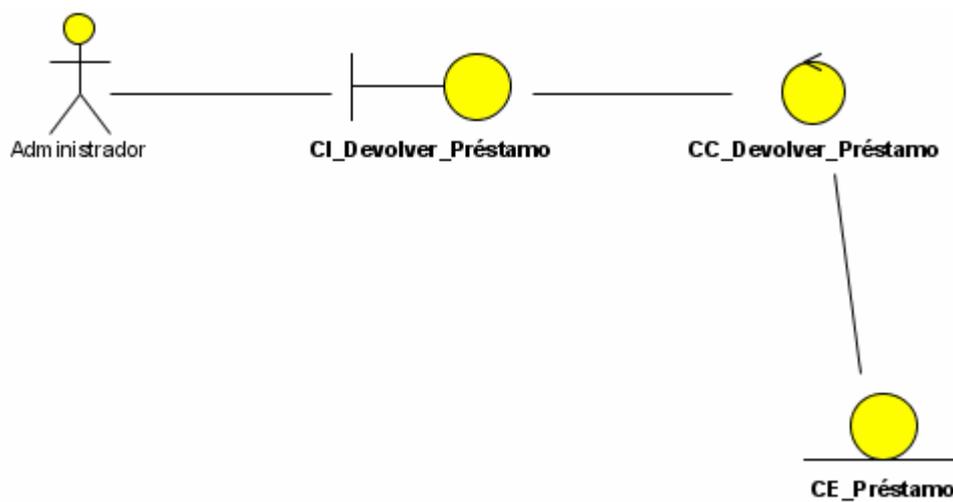


Figura 7: Diagrama de Clases de Análisis Caso Uso Devolver Préstamo

4.2 Diagramas de interacción

Los Diagramas de Interacción están constituidos por dos tipos: los Diagramas de Colaboración y los Diagramas de Secuencia. Ambos expresan información similar, pero en una forma diferente.

Los Diagramas de Secuencia muestran la interacción entre varios objetos y los enlaces que existen entre ellos. Representan las interacciones entre objetos organizadas alrededor de los objetos y sus vinculaciones. Ver Anexo 3

Los Diagramas de Colaboración muestran las interacciones entre un conjunto de objetos, ordenadas según el tiempo en que tienen lugar. Representan una forma de indicar el período durante el que un objeto está desarrollando una acción directamente o a través de un procedimiento.

4.3 Modelo de Diseño

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 tienen impacto en el Sistema a considerar. Además sirve de abstracción de la Implementación y es utilizado como una entrada fundamental de las actividades de la Implementación. Aquí los Casos de Usos son realizados por las Clases del Diseño y sus objetos.

Symfony está basado en un patrón arquitectónico llamado Modelo – Vista – Controlador (MVC). Este proporciona grandes ventajas, como la organización del código, la reutilización, la flexibilidad y una programación mucho más eficiente. El mismo está organizado de la siguiente manera:

Modelo: Se divide en una capa de acceso y otra de abstracción de datos. La abstracción indica qué quiere de la base de datos, y la capa de acceso hace las consultas necesarias para obtener esa información. Así, si cambia la base de datos, tan solo cambiando la capa de acceso, la capa de abstracción podrá seguir haciendo las mismas operaciones.

Vista: En esta capa se producen las páginas que se muestran como resultado de las acciones, trabaja con plantillas llamadas Success que cambian en dependencia de la acción que se realice y un archivo denominado Layout, conocido también como plantilla global, este almacena el código HTML que es común a todas las páginas de la aplicación.

Controlador: Aquí se encuentran las clases sfActions que se encargan de obtener datos, realizar comprobaciones, manejo de las peticiones del usuario, el manejo de la seguridad, cargar la configuración de la aplicación y crear o inicializar las variables necesarias para que se presenten o se utilicen en la plantilla. Además se encuentra el controlador frontal (fronted.php), este se encarga de manejar todas las peticiones Web utilizando el sistema de enrutamiento para asociar el nombre de una acción y el nombre de un módulo con la URL escrita o accedida por el usuario. Además es el único punto de entrada a la aplicación. [15]

Propel: Es una librería integrada en el framework que se encarga de la generación automática, a través de la utilidad Phing, de las clases del Modelo en función de la estructura de datos de la aplicación. La abstracción de la base de datos es completamente invisible al programador, ya que la realiza otro componente llamado Creole, perteneciente también a la librería Propel. De esta manera, si se cambia el sistema gestor de base de datos, no se debe reescribir ni una línea de código, ya que tan solo es necesario modificar un parámetro en un archivo de configuración.

Componentes de Symfony: Representan el comportamiento interno del framework, así como el resto de los componentes que se utilizan para diversos propósitos pero que no son relevantes explicarlos en el Modelo de Diseño.

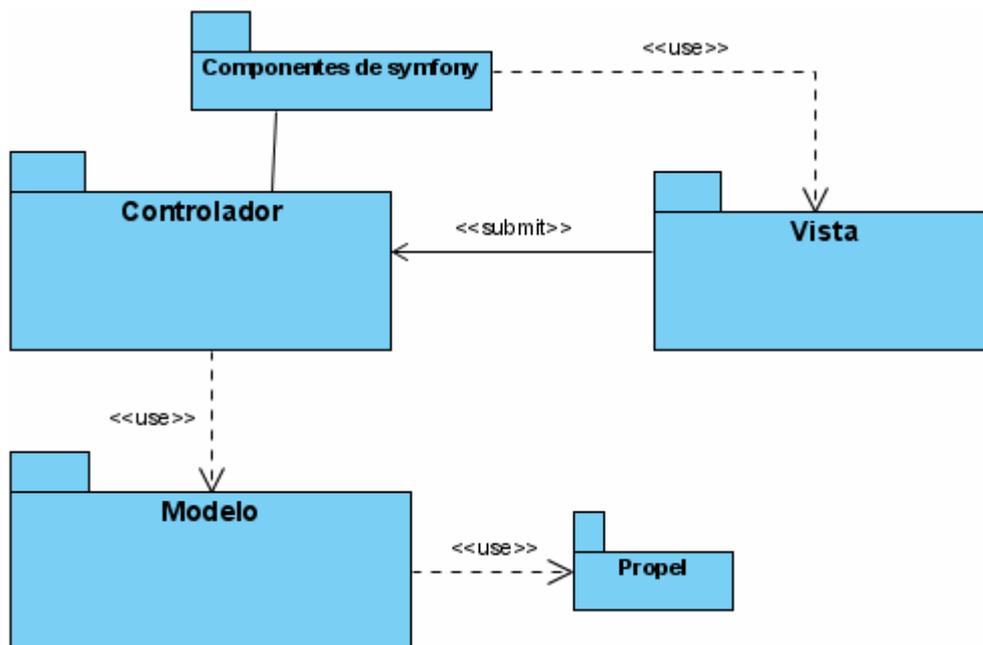


Figura 8: Estructura del patrón MVC según Symfony

4.3.1 Patrones que utiliza el framework

Un patrón es una solución probada que se puede aplicar con éxito a un determinado tipo de problemas que aparecen repetidamente en algún campo (arquitectura, informática, entre otros). Son un esqueleto básico que cada diseñador adapta a las peculiaridades de su aplicación.

Existen distintos tipos de patrones de diseño como son los GRASP y los GOF utilizados como ayuda para diseñar sistemas, independientes del lenguaje que se vayan a implementar. También son utilizados con el fin de establecer un vocabulario de diseño común entre los desarrolladores, así como proveer la reutilización de este diseño en sistemas posteriores.

Symfony no está ajeno ante las facilidades que brindan estos patrones por lo que implementa algunos de ellos. Dentro de los GRASP implementa por ejemplo, el Alta Cohesión para asignar responsabilidades con una alta cohesión, por ejemplo la clase `sfActions` tiene la responsabilidad de definir las acciones para las plantillas y colabora con otras para realizar diferentes operaciones, instanciar objetos y acceder a las propiedades, es decir, está formada por diferentes funcionalidades que se encuentran estrechamente relacionadas proporcionando que el software sea flexible frente a grandes cambios. El Experto, ejemplo de ello lo es la utilización de Propel para realizar su capa de abstracción en el Modelo a través de Creole y generar automáticamente con Phing las clases con todas las funcionalidades comunes de las entidades.

También implementa algunos de los patrones GOF como el Singleton (Instancia única) para garantizar que con la ejecución de un solo método en el controlador frontal, se realicen llamadas en otras clases localizadas en los componentes propios del framework. Este proceso llamado enrutamiento permite determinar la acción correspondiente, a la vez que se define la clase `Success` y su decorado (Layout) para en su conjunto dar respuesta a una determinada petición del usuario. El Layout almacena el código HTML que es común a todas las páginas de la aplicación y como se explica anteriormente utiliza la `Success` que posee la información a mostrar para conformar la página cliente. Esta unión es posible con la ayuda del patrón Decorator.

4.3.2 Diagramas de Clases del Diseño

Debido al alto grado de complejidad y a la abundante información a modelar en los Diagramas de Clases del Diseño, se decide representar únicamente las clases y componentes que por su importancia y su papel representativo, determinan la construcción de la solución, sin dejar de incluir los aspectos esenciales del framework.

En el Modelo se realiza el mapeo de los objetos relacionales (ORM) y por cada tabla de la base de datos, el framework genera cuatro clases con el objetivo de optimizar la lógica del negocio.

Nombre de la Clase	Descripción
BaseClase	Se generan directamente a partir del esquema. No se le añaden métodos personalizados ya que se borrarían cada vez que se vuelven a generar esas clases. Contiene todos los atributos definidos en la tabla y un conjunto de métodos ya implementados que gestionan el acceso a los datos y aceleran y simplifican el trabajo.
BaseClasePeer	Se utilizan para manejar varios objetos de tipo BaseClase.
ClasePeer	Clases que poseen métodos estáticos para trabajar con las tablas de la BD. Proporcionan los medios necesarios para obtener los registros de las tablas. Sus métodos devuelven un objeto o una colección de objetos de la clase relacionada.
Clase	Es la clase objeto que representa un registro de la base de datos. Clase que hereda de BaseClase, a la que se añaden métodos propios.

Tabla 10: Clases del Modelo del framework

Para no sobrecargar el Modelo con dichas clases y lograr una mejor explicación del mismo se decide incluir solamente las BaseClases pues es a través de ellas que se relacionan las demás. Es de vital importancia destacar que el acceso a estas clases desde el controlador se realiza a través de las ClasesPeer. El Modelo quedaría representado de la siguiente manera:

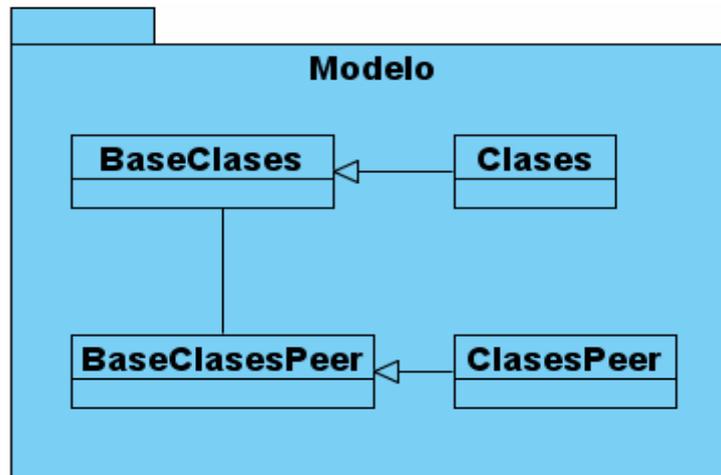


Figura 9: Estructura del Modelo

Los Diagramas de Clases del Diseño se representan por CU, teniendo en cuenta las restricciones explicadas anteriormente para evitar abrumar el mismo. Además las clases del Modelo propias de cada CU se muestran en el ANEXO 5.

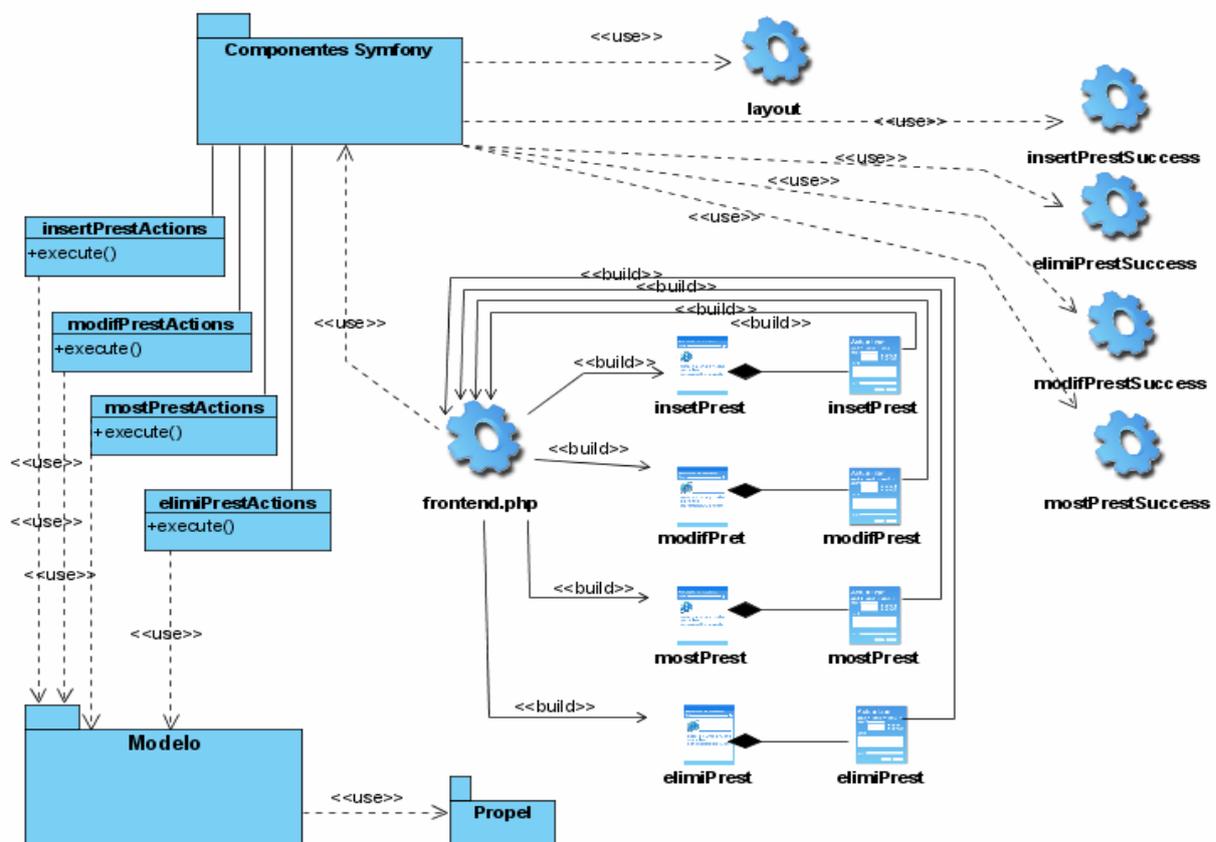


Figura 10: Diagrama de Clases del Diseño del CU Realizar Préstamo

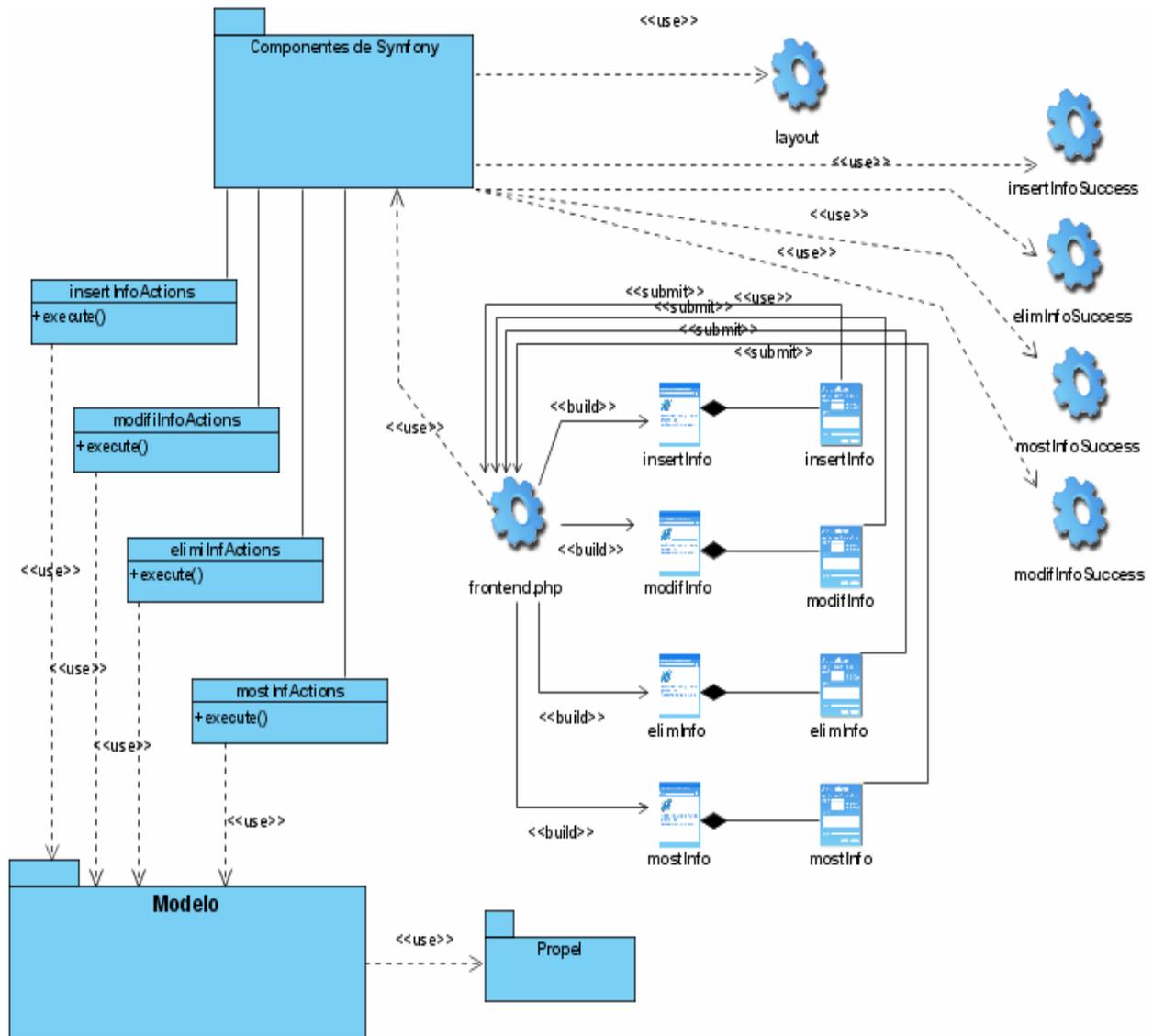


Figura 11: Diagrama de Clases del Diseño del CU Gestionar Informes

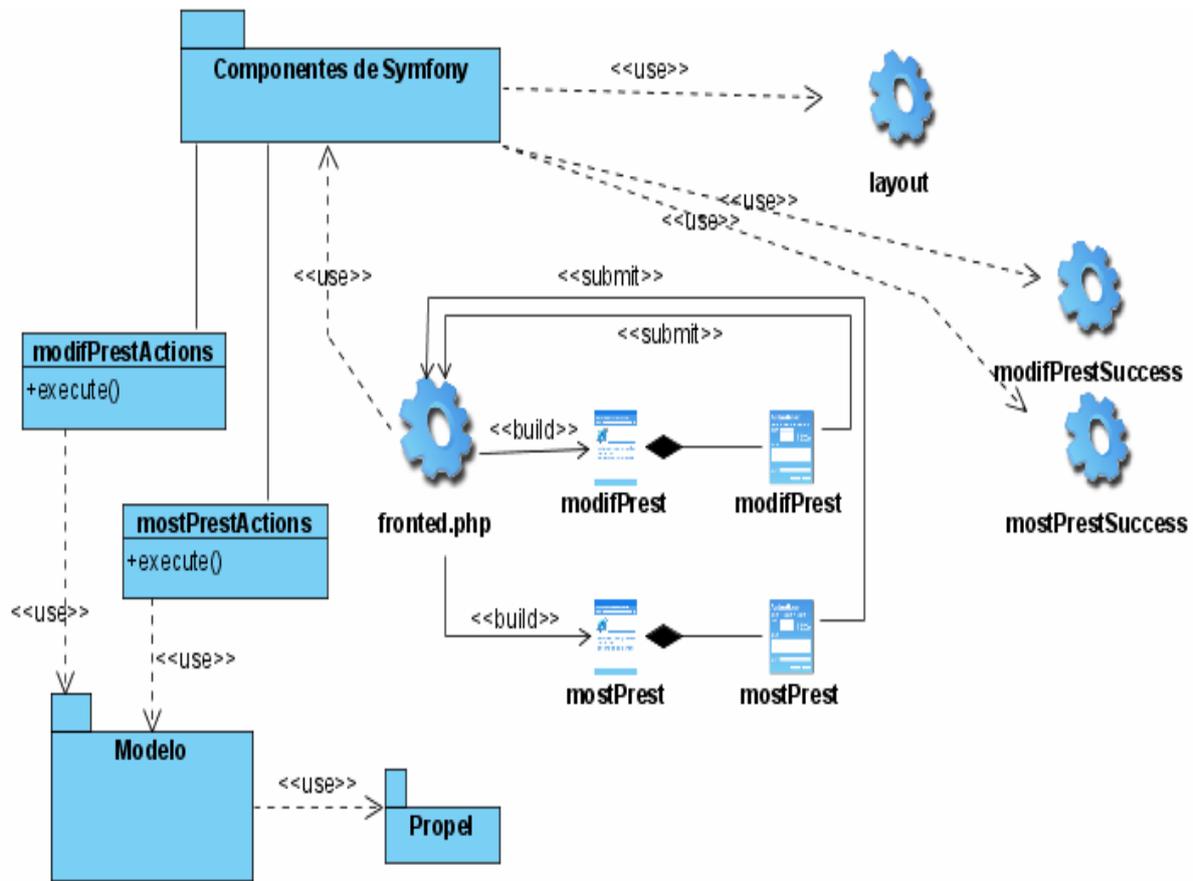


Figura 12: Diagrama de Clases del Diseño del CU Devolver Préstamo

Para abordar más sobre el flujo de información que brindan estos diagramas y lograr una mejor comprensión de las acciones que gestionan la entrada y salida de datos, se explica la sección insertar del CU Gestionar Informe.

La petición del usuario de insertar un informe, arriba al controlador frontal (fronted.php). Inmediatamente el mismo usa los componentes de Symfony para acceder a las clases Actions. Estas últimas se encargan de acceder a las clases del Modelo las cuales ejecutan la acción requerida en la base de datos (en este caso introducir los datos de un nuevo informe). Una vez culminado el proceso de inserción, las clases Actions envían la respuesta a las clases responsables de ello en el núcleo del framework. Seguido a esto estas clases del núcleo que forman parte de los componentes del framework, definen que clase Success utilizar para mostrar en la plantilla llamada Layout. Es entonces el controlador frontal quien ejecuta esta acción. Esta unión de las clases Success con la plantilla Layout constituiría la página cliente que vería el usuario. La respuesta que se muestra no es más que el listado actualizado de los informes registrados, en caso de ser satisfactoria la inserción

del mismo. Si la inserción no es válida, el propio framework tiene implementado por defecto donde mostrar un error.

De manera general el flujo de trabajo que gestiona Symfony sigue siempre los mismos pasos que se muestran en la figura siguiente:

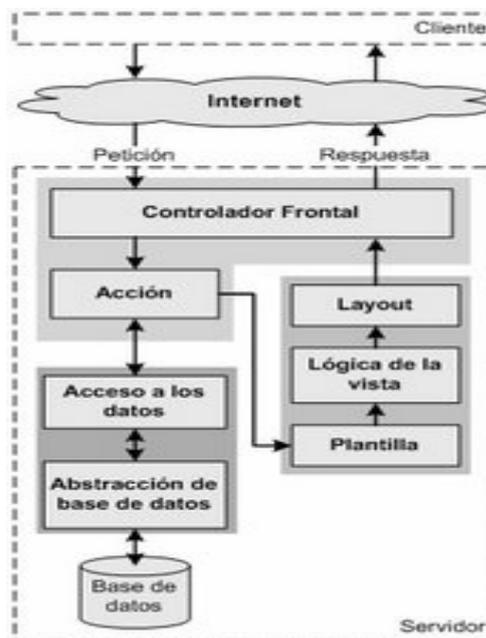


Figura 13: Flujo de trabajo de Symfony

4.3.2 Diagrama de clases persistentes

A continuación se muestra el diagrama de clases persistentes del sistema, donde aparecen todas las entidades que se manejan en él.

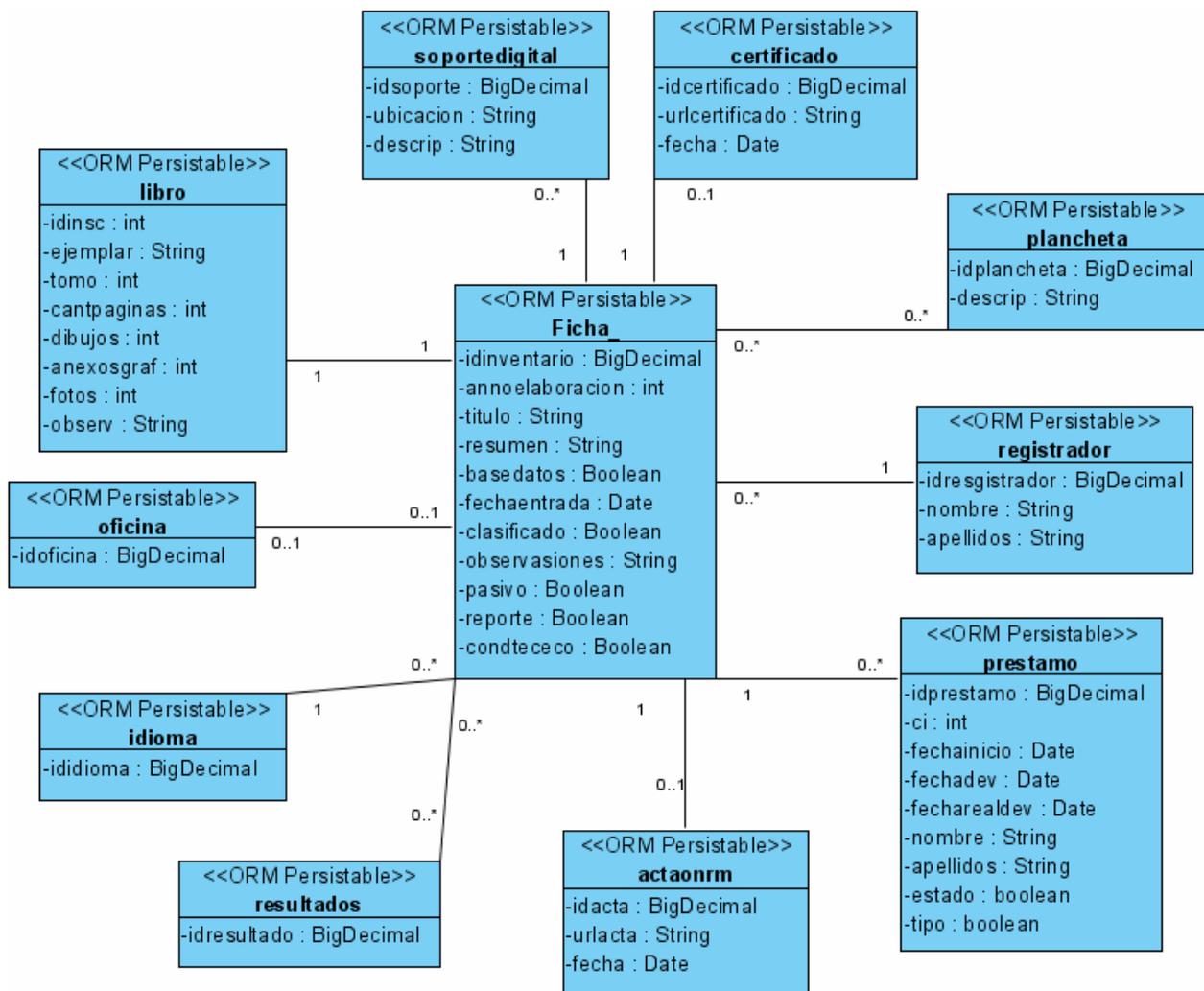


Figura 14: Diagrama de Clases Persistentes

4.4 Principios del Diseño

El diseño, no importa cual sea la esencia del mismo, se basa en el usuario donde la gran mayoría de ellos no poseen una buena preparación en informática. El diseño Web no se limita a la apariencia estética, la combinación de colores, o a un logo más o menos acertado. De él depende que la información sea útil, que los servicios se puedan usar, es decir, el diseño convierte a una aplicación Web en algo atractivo para el usuario, por su estética y su utilidad.

4.4.1 Estándares de la interfaz de la aplicación

La interfaz gráfica del usuario es el medio por el cual este interactúa con el sistema, por lo que esta debe ser lo más amigable posible y lograr que se sienta identificado con la misma.

Una aplicación con una interfaz bien diseñada debe tener, además de un buen diseño gráfico, una buena navegabilidad, usabilidad y distribución de los contenidos. Para ello, este sistema utiliza ciertos estándares que garantizan la usabilidad en los diseños para aplicaciones Web.

- 1) Requerir de un mínimo transcurso de instrucción, permitiendo su uso desde el primer momento, por cualquier persona que posea poco dominio de la computación.
- 2) Permitir que la privacidad, garantía y seguridad estén igualmente disponibles para todos los usuarios, y que el diseño sea atractivo para todos en general.
- 3) Garantizar la legibilidad, el color de los textos debe contrastar con el del fondo, y el tamaño de fuente debe ser suficientemente grande.
- 4) Evitar elementos invisibles de navegación que han de ser inferidos por los usuarios, menús desplegados, indicaciones ocultas, entre otras.
- 5) Requerir de los usuarios un mínimo esfuerzo para alcanzar sus objetivos.
- 6) Evitar las caídas inesperadas de la aplicación y los enlaces rotos.
- 7) Limitar el número de acciones que puede realizar el usuario sobre la aplicación, mostrando sugerencias (opciones) para cada posible acción, evitando así al máximo los errores de usuario.
- 8) Mostrar al usuario solamente aquellas opciones a las que, dado su rol en el Negocio, tiene derecho a acceder.
- 9) Mostrar al usuario, siempre que vaya a realizar una acción relevante sobre el sistema, un mensaje de confirmación que le permita asegurarse de que es correcta la opción seleccionada.
- 10) Mostrar la mayor cantidad de información acerca de las opciones brindadas en un momento dado, de modo que el usuario siempre sepa cuáles son las operaciones a las que puede acceder y en qué consiste exactamente cada una.

4.6 Concepción general de la ayuda

La aplicación va dirigida a usuarios con escasa apreciación en el tema de la informática y con ritmo de trabajo muy agitado. Por este motivo se consideró que una ayuda formal, sería poco funcional. La ayuda está accesible en todas las páginas de la aplicación en dependencia de donde se encuentre el usuario, con el fin de que él vea como relacionarse y trabajar con lo que necesite en el momento, cada página muestra como realizar solo aquellas operaciones que se estén realizando en el momento.

4.7 Diseño de la Base de Datos

La descripción de la estructura de una base de datos es el Modelo de Datos, una colección de herramientas conceptuales para describir datos, relación de datos, semánticas y restricciones.

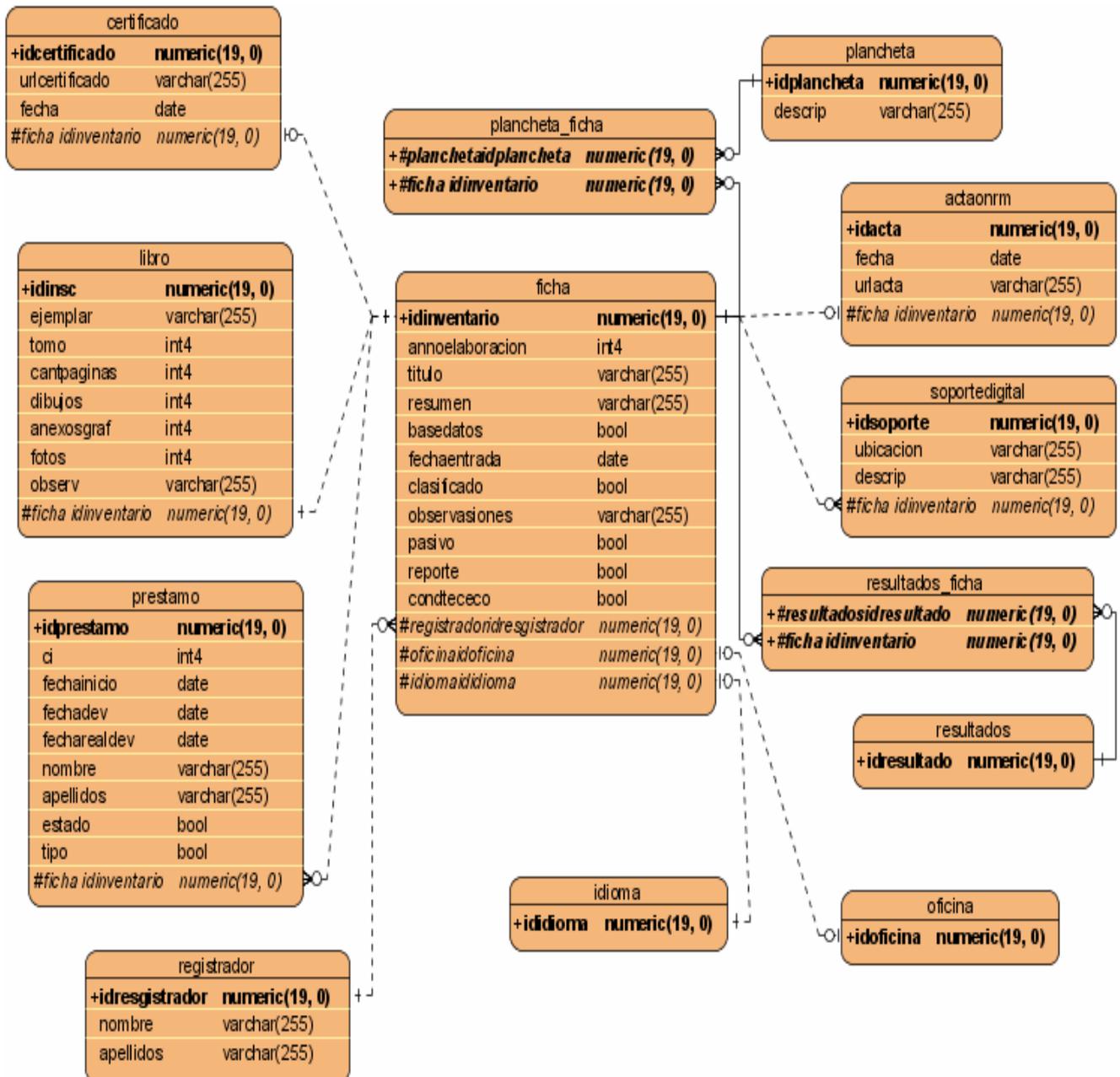


Figura 15: Modelo de Datos

4.8 Modelo de Implementación

El Modelo de Implementación describe como los elementos del Modelo de Diseño se implementan en términos de componentes, como ficheros de código fuente, ejecutables, entre otros. También describe como se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de Implementación y los lenguajes de programación utilizados. Define una jerarquía. Para representar el Diagrama de Componentes, debido a la cantidad de información que brinda, se decide modelarlo por paquetes para no recargarlo y lograr un mejor entendimiento del mismo. La explicación del contenido de dichos paquetes se muestra en el ANEXO 6.

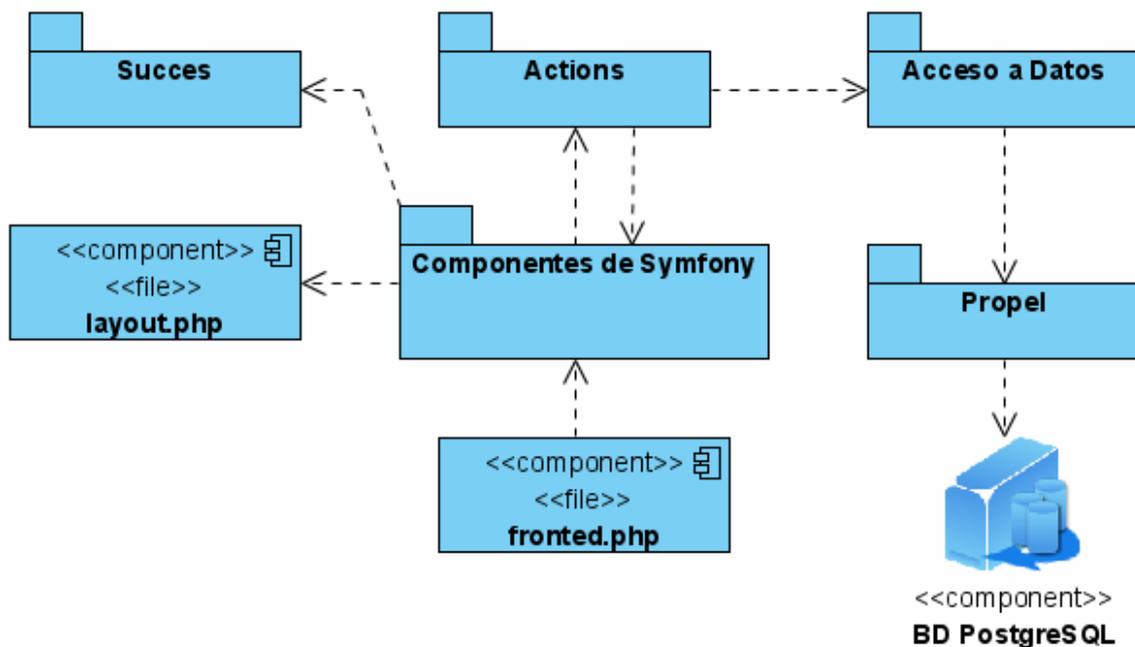


Figura 16: Diagrama de Componentes

4.9 Modelo de Despliegue

El Modelo de Despliegue describe la distribución física del sistema, muestra como están distribuidos los componentes de software entre los distintos nodos de cómputo. Permite comprender la correspondencia entre la arquitectura software y la arquitectura hardware.

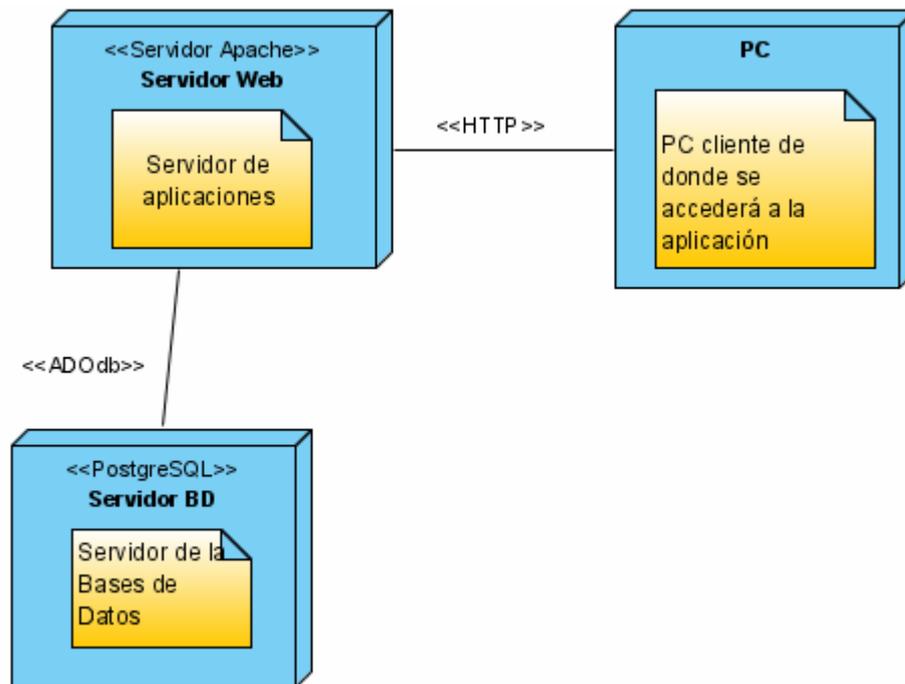


Figura 17: Modelo de Despliegue

4.10 Conclusiones parciales

En el presente capítulo se logra una aproximación al Diseño de la aplicación gracias al eficiente modelado del Análisis. Se realizó el diseño de la aplicación quedando plasmadas las ventajas que brinda el framework Symfony. Además se logró rediseñar la base de datos de forma tal que cubriera las necesidades para la gestión del Archivo Técnico, así como los diferentes servicios requeridos por la ONRM. Por lo antes mencionado se arriba a la conclusión que en esta etapa recién concluida, quedan diseñadas las principales funcionalidades que se definieron para la futura implementación de la aplicación.

CONCLUSIONES

Con la realización de este trabajo, una vez concluida cada fase de la metodología que se utilizó y por ende haber modelado los procesos que se definieron como claves en la construcción de la solución, se lograron los objetivos trazados en función de las tareas programadas para llevar a cabo el proceso investigativo.

Se pudo optimizar la base de datos existente en la ONRM. También se logró diseñar una aplicación Web que cubriera las necesidades de la oficina y además que fuera capaz de mejorar los procesos de gestión de los informes generados de las investigaciones geológicas en Cuba, añadiéndole nuevas funcionalidades. Además se logró gestionar los préstamos de dichos documentos en la ONRM. La propuesta de Diseño elaborada en este trabajo proporciona la seguridad de que la Fase de Implementación de dicha aplicación se lleve a cabo satisfactoriamente.

RECOMENDACIONES

Esta aplicación está diseñada como un módulo o subsistema de un proyecto general de aplicaciones Web, por lo que la seguridad se gestiona a nivel superior. Lo que significa que la autenticación y validación de usuario se realiza fuera del módulo SGAT. Se recomienda en una futura versión del trabajo gestionar el tema de seguridad haciendo chequeos constantes de autenticación del usuario para evitar accesos no autorizados.

Se recomienda gestionar una nueva funcionalidad de inserción de datos a partir de la interpretación gráfica de mapas o bosquejos geológicos, ya que esta información es importante comprenderla y utilizarla en la oficina y hasta el momento, la misma, no cuenta con herramientas para dicho propósito.

Se exhorta mejorar la aplicación diseñada desde el punto de vista de convertirlo de software a la medida (software requerido por una empresa específica) a software producto (software que puede ser utilizado por cualquier empresa).

GLOSARIO DE TÉRMINOS

ActiveX: Tecnología de Microsoft para el desarrollo de páginas dinámicas. Tiene presencia en la programación del lado del servidor y del lado del cliente, aunque existan diferencias en el uso en cada uno de esos dos casos.

AJAX: (Asynchronous JavaScript And XML), es una técnica de desarrollo Web para crear aplicaciones interactivas.

API: (Application Programming Interface - Interfaz de Programación de Aplicaciones), es el conjunto de funciones y procedimientos (o métodos si se refiere a programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

C: Lenguaje de programación creado en 1972 por Ken Thompson y Dennis M. Ritchie en los Laboratorios Bell como evolución del anterior lenguaje B.

C++: Lenguaje de programación, diseñado a mediados de los años 1980, por Bjarne Stroustrup, como extensión del lenguaje de programación C.

CGI: Interfaz de entrada común (en inglés Common Gateway Interface) es una importante tecnología de la World Wide Web que permite a un cliente (explorador web) solicitar datos de un programa ejecutado en un servidor web.

CIDR: Classless Inter-Domain Routing (Encaminamiento Inter-Dominios sin Clases), se introdujo en 1993 y representa la última mejora en el modo como se interpretan las direcciones IP. Su introducción permitió una mayor flexibilidad al dividir rangos de direcciones IP en redes separadas.

CORBA: (Common Object Request Broker Architecture — arquitectura común de intermediarios en peticiones a objetos), es un estándar que establece una plataforma de desarrollo de sistemas distribuidos facilitando la invocación de métodos remotos bajo un paradigma orientado a objetos.

DB2: Motor de base de datos relacional que integra XML de manera nativa.

Debugging: Depuración, corrección de errores en la programación empleando un debugger.

ECMAScript: Define un lenguaje de tipos dinámicos ligeramente inspirado en Java y otros lenguajes del estilo de C. Soporta algunas características de la programación orientada a objetos mediante objetos basados en prototipos y pseudoclases.

Flash: Flash es la tecnología más comúnmente utilizada en el Web que permite la creación de animaciones vectoriales.

FTP: Protocolo de Transferencia de Archivos FTP son las siglas de File Transfer Protocol. Es un sistema que permite enviar y recibir ficheros entre computadores a través de la red Internet.

GPL: (General Public License), es una licencia de distribución de software gratuito que permite copiarlo, modificarlo y redistribuirlo.

IBM: (International Business Machines), es la compañía de computadoras más grande del mundo.

IMAP: (Internet Message Access Protocol) es un protocolo de red de acceso a mensajes electrónicos almacenados en un servidor.

Java: Lenguaje que se inspira en la sintaxis de C++, pero su funcionamiento es más similar al de Smalltalk que a éste. Incorpora sincronización y manejo de tareas en el lenguaje mismo (similar a Ada) e incorpora interfaces como un mecanismo alternativo a la herencia múltiple de C++.

Java Servlets: Pequeños programas escritos en Java que se ejecutan en el contexto de un navegador Web. Por contraposición, un Servlet es un programa que se ejecuta en un servidor.

JScript: Lenguaje de programación que permite a los desarrolladores crear acciones en sus páginas Web. Lenguaje de secuencias de comandos moderno con una gran variedad de aplicaciones, orientado a objetos. Es la implementación de Microsoft de ECMAScript.

J2EE: Java 2 Platform Enterprise Edition. Es un ambiente de Sun enfocado en Java, independiente de la plataforma, para el desarrollo, construcción e implementación de aplicaciones online para empresas basadas en Web.

Linux: Sistema operativo tipo Unix (también conocido como GNU/Linux) que se distribuye como software libre. Su nombre proviene del Núcleo de Linux, desarrollado en 1991 por Linus Torvalds.

Mac OS X: es un sistema operativo basado en UNIX, donde el gestor de ventanas X11 es característico de estos sistemas.

MySQL: Sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones.

NNTP: Protocolo para la Transferencia de Noticias en Red

Oracle: Potente herramienta cliente/servidor para la gestión de Bases de Datos.

RSS: Sublenguaje surgido de la aplicación del metalenguaje XML.

SMTP: Simple Mail Transfer Protocol. Es el protocolo estándar más usado en Internet para enviar e-mail.

UNIX: Unix es el sistema operativo por excelencia en aquellos entornos que necesitan fiabilidad y robustez para funcionar las 24 horas del día a pleno rendimiento. Es un sistema operativo portable, multitarea y multiusuario.

URL: son las siglas de Localizador de Recurso Uniforme (en inglés Uniform Resource Locator), la dirección global de documentos y de otros recursos en la World Wide Web.

VBScript: (Visual Basic Script Edition), es un lenguaje interpretado por el Windows Scripting Host de Microsoft.

Weblog: es un sitio Web en el que se publican anotaciones (historias, artículos, posts) mediante un sistema de publicación sencillo.

XML: Extensible Markup Language (lenguaje de marcas extensible), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C).

REFERENCIA BIBLIOGRÁFICA

1. García, J.M., *Teoría y ejercicios prácticos de Dinámica de Sistemas*. Segunda Edición ed. 2003.
2. Jaramillo, O. (2007) *El concepto de Sistema*. **Volume**,
3. Guglielmetti, M. (2004) *Definición de Sistema informático*. **Volume**,
4. Lanzillotta, A. (2004) *Definición de Aplicación*. **Volume**,
5. TELEBAS, D. *Datos Generales MINBAS*. 2001 [cited 2008 4/4/2008]; Available from: <http://www.minbas.cu/>.
6. Espasa-Calpe. *Geología*. 2005 [cited 2008 3/2/2008]; Available from: <http://www.wordreference.com/definicion/geolog%EDa>.
7. Pimienta, D. *¿Qué son las TIC?* 2005 [cited 2008 17/3/2008]; Available from: <http://www.etic.bo/Default/default.htm>.
8. W3C. *Guía Breve de Servicios Web*. 2007 [cited 2008 12/2/2008]; Available from: <http://www.w3c.es/divulgacion/guiasbreves/ServiciosWeb>.
9. CASTILLO, G. *Herramientas Web para la enseñanza de Protocolos de Comunicación*. 2002 [cited 2008 15/3/2008]; Available from: <http://neo.lcc.uma.es/evirtual/cdd/tutorial/aplicacion/cliente-servidor.html>.
10. Alvarez, M.A. *Lenguajes de lado servidor*. 2007 [cited 2008 23/2/2008]; Available from: <http://www.desarrolloweb.com/articulos/243.php>.
11. Henst, C.V.D. (2001) *¿Qué es el ASP?* **Volume**,
12. Alvarez, R. *Manual de PHP*. 2008 [cited 2008 4/4/2008]; Available from: <http://www.desarrolloweb.com>.
13. Arkaitz Arteaga. *¿Qué es Eclipse?* 2007 [cited 2008 13/3/2008]; Available from: <http://plataformaclipse.com/>.
14. Technologies, Z. *Zend Framework*. 2008 [cited 2008 14/3/2008]; Available from: <http://framework.zend.com/manual/en/>.
15. Fabien Potencier, F.Z., *The Definitive Guide to Symfony* Apress ed. 2007. 520.
16. Group, P.G.D. *PostgreSQL*. 2008 [cited 2008 4/4/2008]; Available from: <http://www.postgresql.org/about/press/presskit82.html.es>.
17. tinysofa. 2008 [cited 2008 4/4/2008]; Available from: <http://www.postgresql.org/about/press/presskit82.html.es>.
18. Foundation, T.A.S. *Versión 2.2 de la documentación del Servidor de HTTP Apache*. 2006 [cited 2008 3/3/2008]; Available from: <http://www.apache2.com/>.
19. Ubuntu, C. *JBoss*. 2007 [cited 2008 16/3/2008]; Available from: <http://www.guia-ubuntu.org/index.php?title=JBoss>.
20. Corporation, M. *Internet Information Services*. 2008 [cited 2008 21/2/2008]; Available from: <http://www.microsoft.com/windowsserver2003/iis/default.msp>.
21. Larman, C., *UML Y PATRONES*. Prentice Hall Hispanoamericana ed. 1999, Mexico. 536.
22. Patricio Letelier, M.C.P. (2006) *Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*. **Volume**,
23. GALLEGO, J.P.G. (2007) *FUNDAMENTOS DE LA METODOLOGIA RUP*. **Volume**,
24. Ivar Jacobson, G.B., James Rumbaugh, *El Proceso Unificado de Software*. Pearson Education ed. 2000, Madrid.
25. Mercedes, M. (2001) *Herramientas CASE*. **Volume**,

BIBLIOGRAFÍA

- ALVAREZ, M. A. *Lenguajes de lado servidor* 2008.
- ALVAREZ, R. *Manual de PHP* 2008
- ARKAITZ ARTEAGA. *¿Qué es Eclipse?* 2008
- CASTILLO, G. *Herramientas Web para la enseñanza de Protocolos de Comunicación* 2008.
- CORPORATION, M. *Internet Information Services* 2008
- ESPASA-CALPE. *Geología* 2008
- FABIEN POTENCIER, F. Z. *The Definitive Guide to Symfony Apress* 2007
- FOUNDATION, T. A. S. *Versión 2.2 de la documentación del Servidor de HTTP Apache* 2008
- GALLEGO, J. P. G. *FUNDAMENTOS DE LA METODOLOGIA RUP.* 2007
- GARCÍA, J. M. *Teoría y ejercicios prácticos de Dinámica de Sistemas.* 2003.
- GROUP, P. G. D. *PostgreSQL* 2008
- GUGLIELMETTI, M. *Definición de Sistema informático.* 2004
- HENST, C. V. D. *¿Qué es el ASP?* 2001
- IVAR JACOBSON, G. B., JAMES RUMBAUGH. *El Lenguaje Unificado de Modelado.* n°
- IVAR JACOBSON. *El Proceso Unificado de Software.* 2000.
- JARAMILLO, O. *El concepto de Sistema.* 2007
- LANZILLOTTA, A. *Definición de Aplicación.* 2004
- LARMAN, C. *UML Y PATRONES.* 1999.
- MERCEDES, M. *Herramientas CASE.* 2001
- MICROSOFT. *Sistema informático* 2008
- PATRICIO LETELIER, M. C. P. *Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP).* 2006
- PIMIENTA, D. *¿Qué son las TIC?* 2008
- TECHNOLOGIES, Z. *Zend Framework* 2008

TELEBAS, D. *Datos Generales MINBAS* 2008

UBUNTU, C. *JBoss* 2008

VIEYRA, G. E. *Definición de computadora*. publicado 2000

W3C. *Guía Breve de Servicios Web* 2008

PRESSMAN, R. S. *Ingeniería del Software. Un enfoque Práctico*. Quinta Edición ed.
Madrid: 2001.

JESÚS GARCÍA MOLINA, M. J. O., BEGOÑA MOROS, JOAQUÍN NICOLÁS,
AMBROSIO TOVAL. *De los Procesos del Negocio a los Casos de Uso*.

ANEXOS

ANEXO 1: Proceso de desarrollo de Software

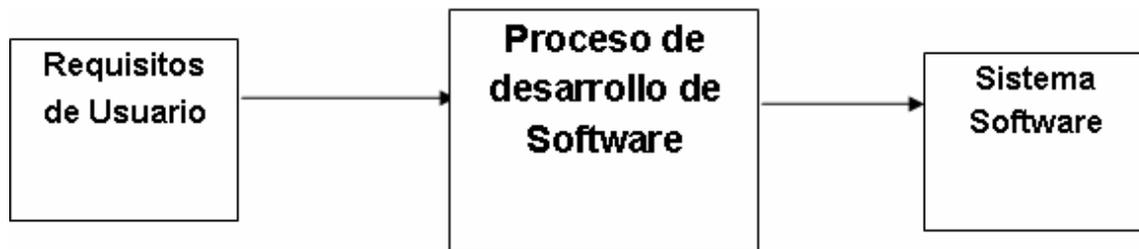


Figura 18: Proceso de desarrollo de Software

ANEXO 2: Ciclo de vida de RUP

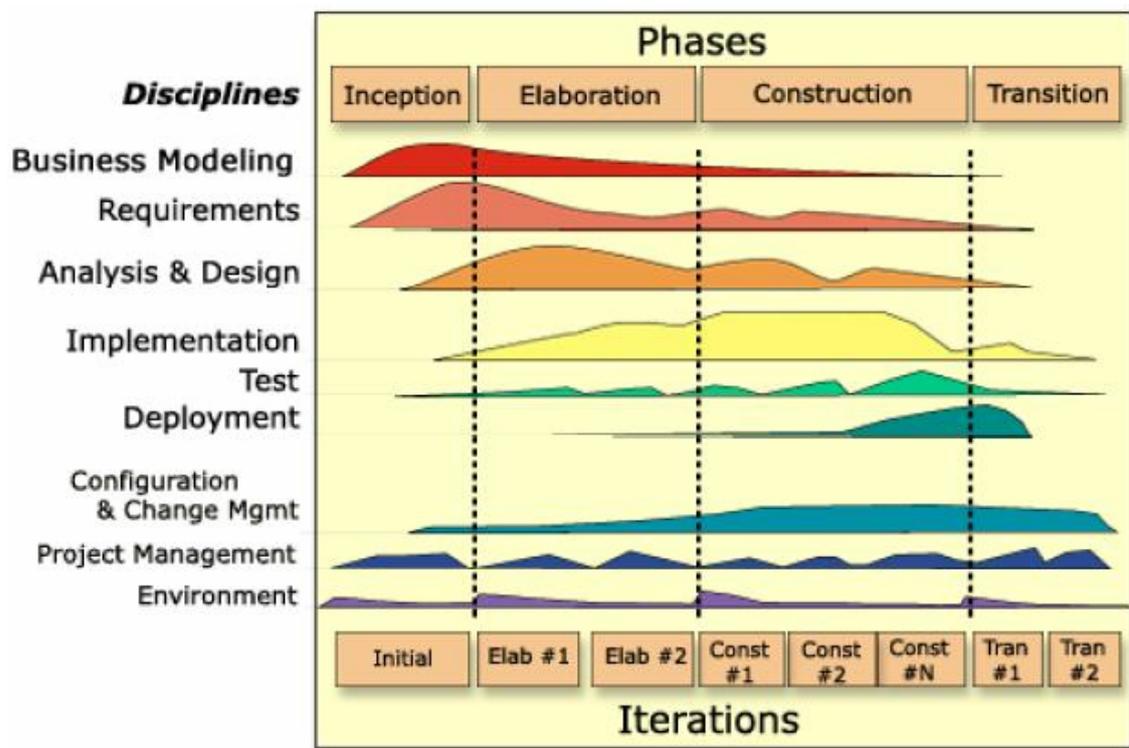


Figura 19: Fases de RUP

ANEXO 3 Diagramas de Secuencia del Análisis

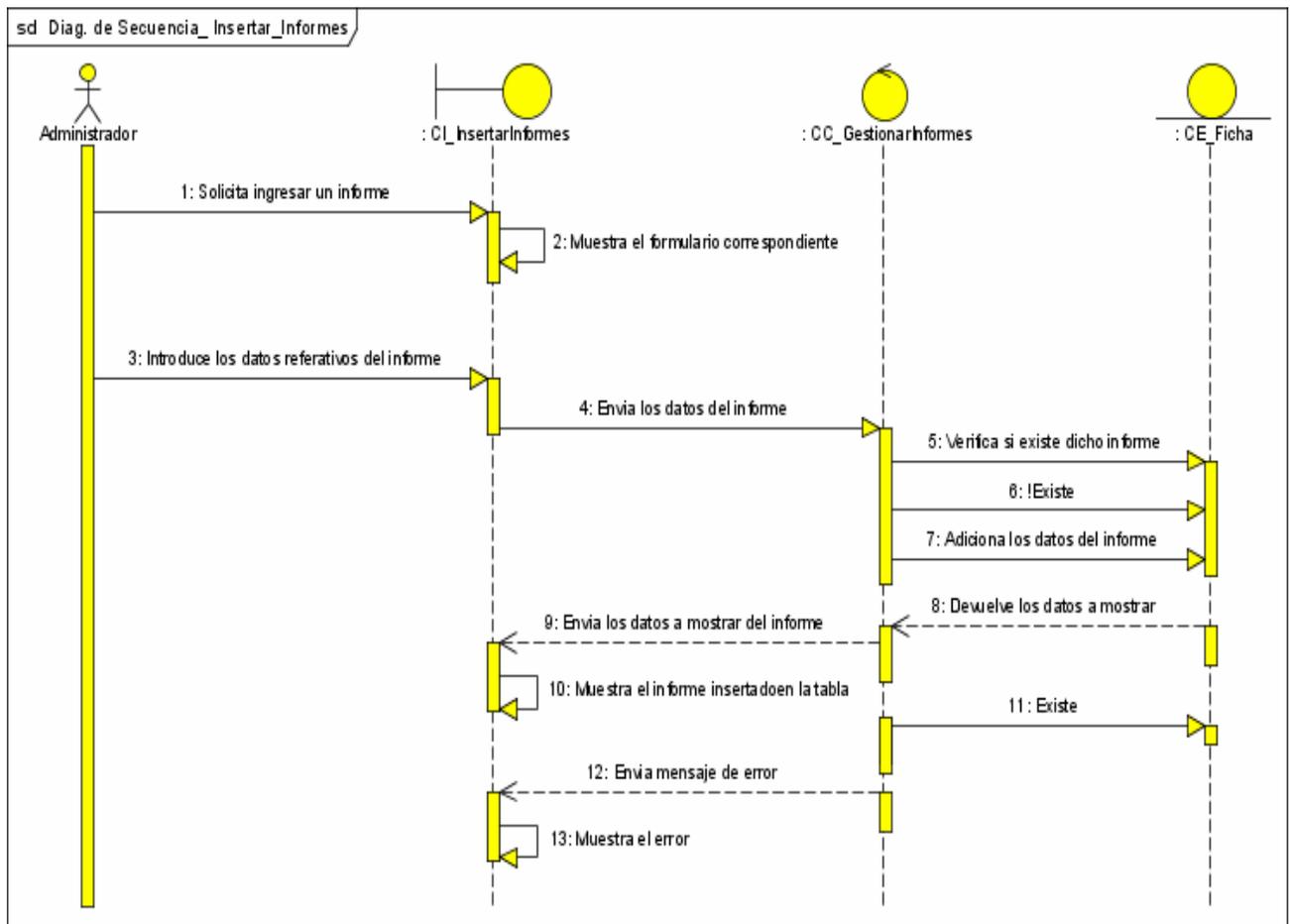


Figura 20: Diagrama de Secuencia Insertar Informe

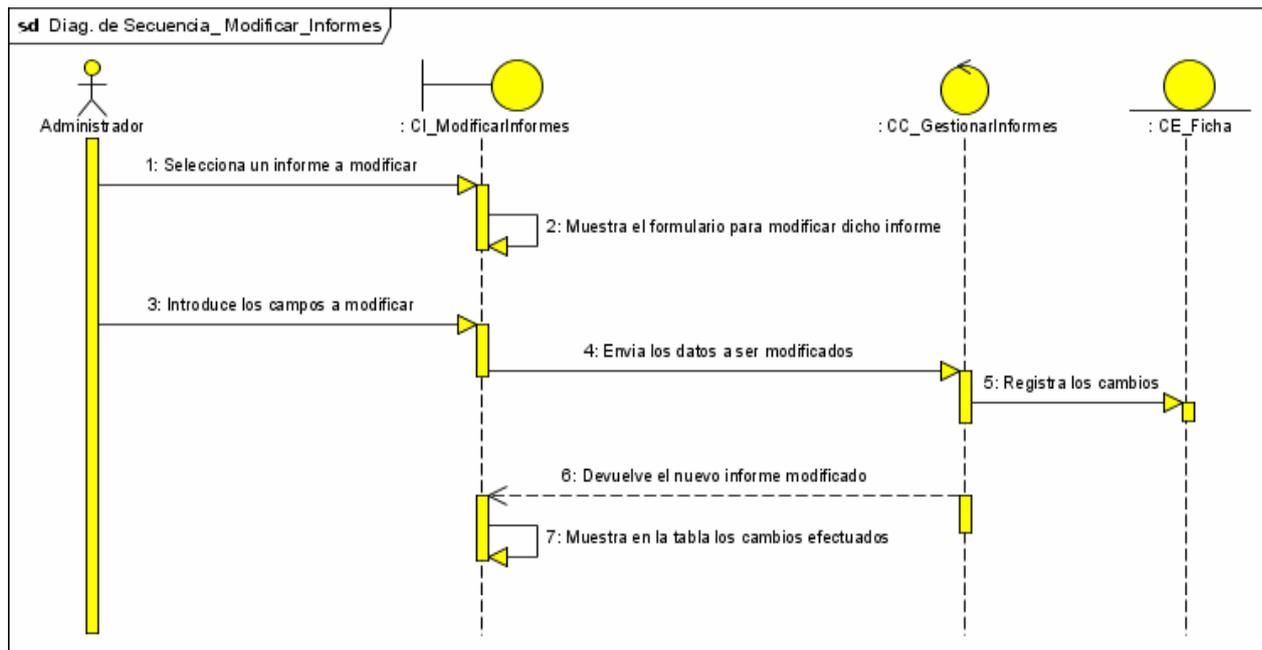


Figura 21: Diagrama de Secuencia Modificar Informe

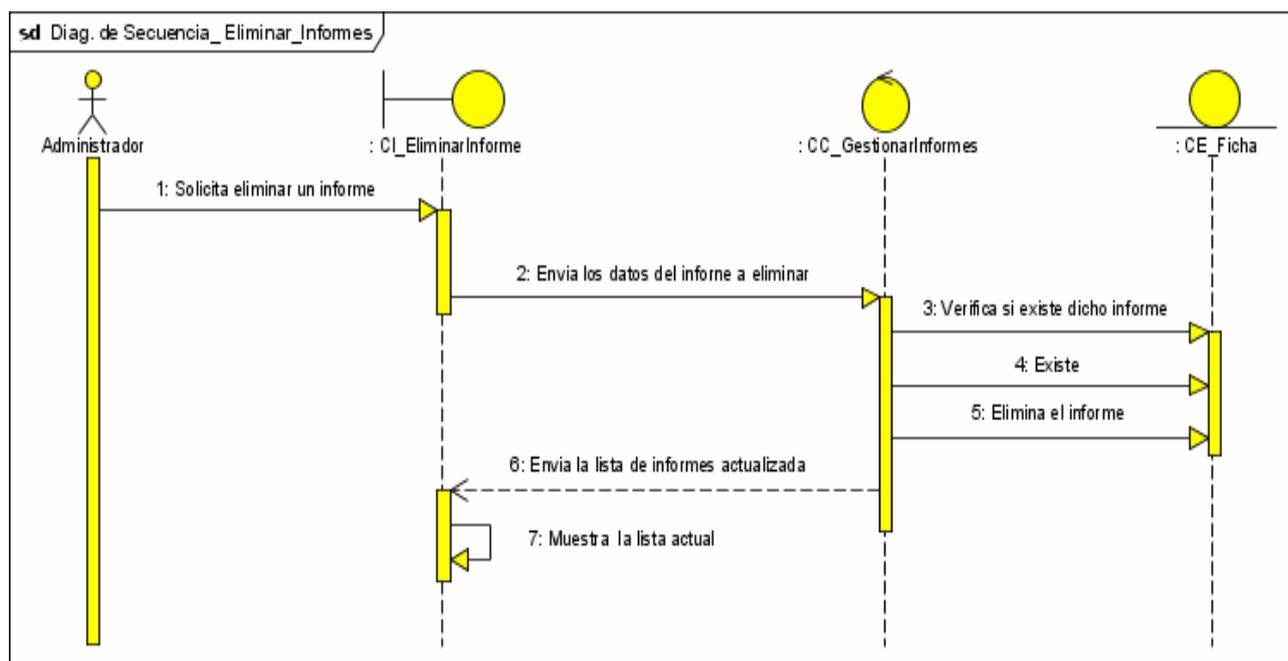


Figura 22: Diagrama de Secuencia Eliminar Informe

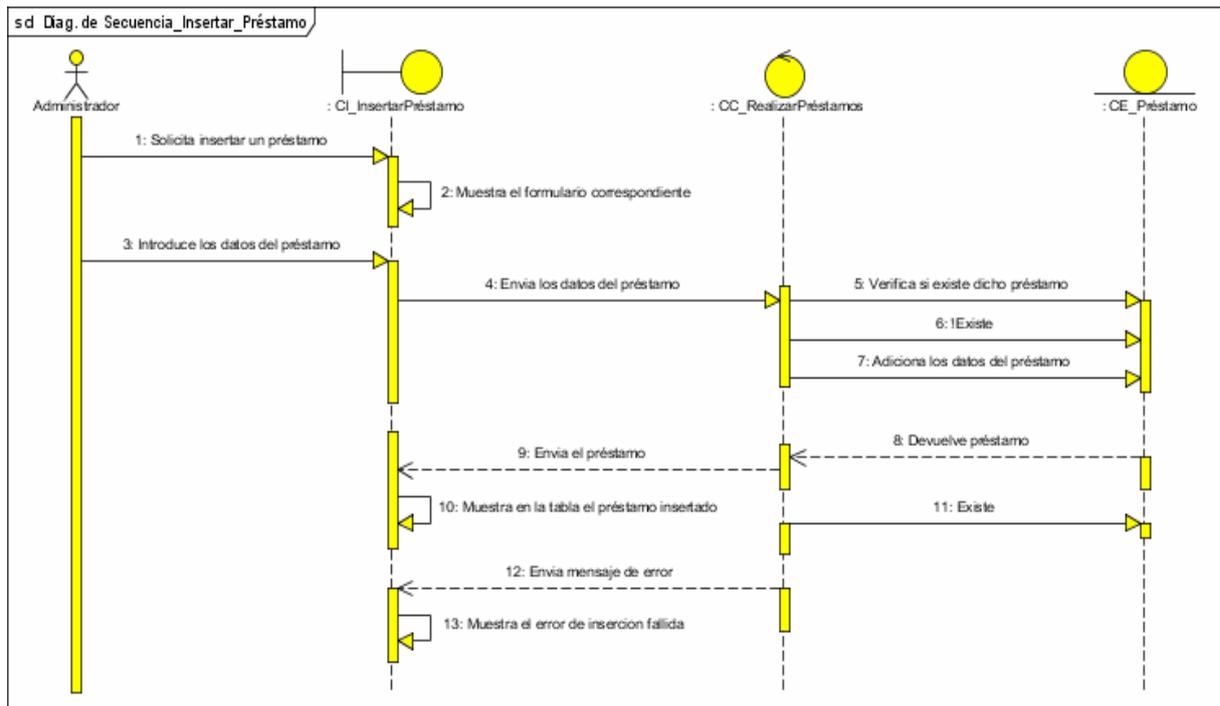


Figura 23: Diagrama de Secuencia Insertar Préstamos

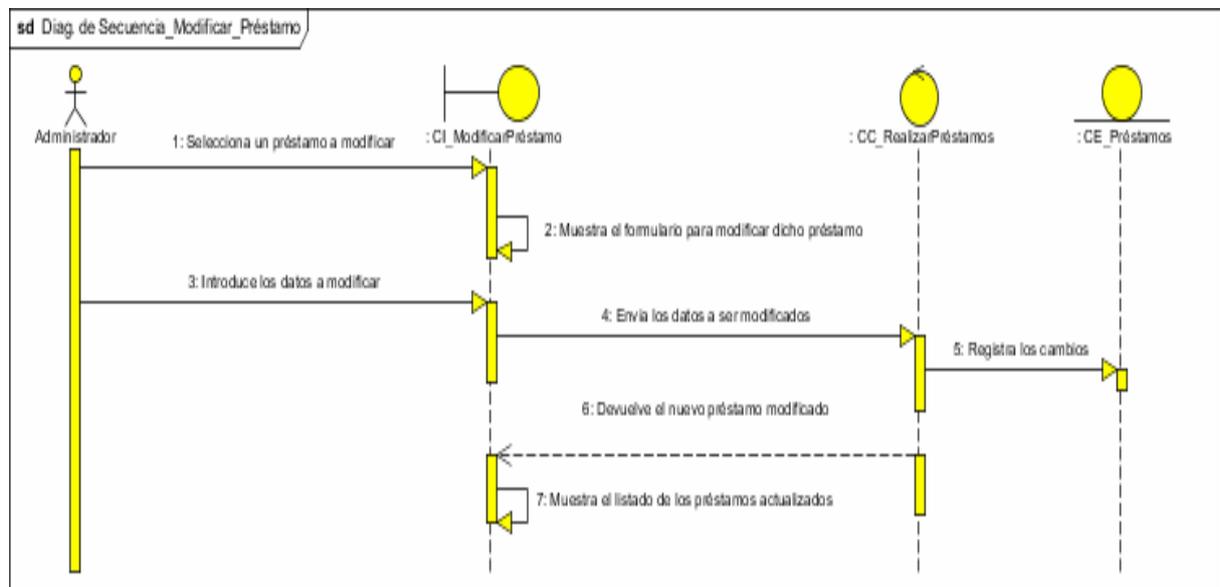


Figura 24: Diagrama de Secuencia Modificar Préstamos

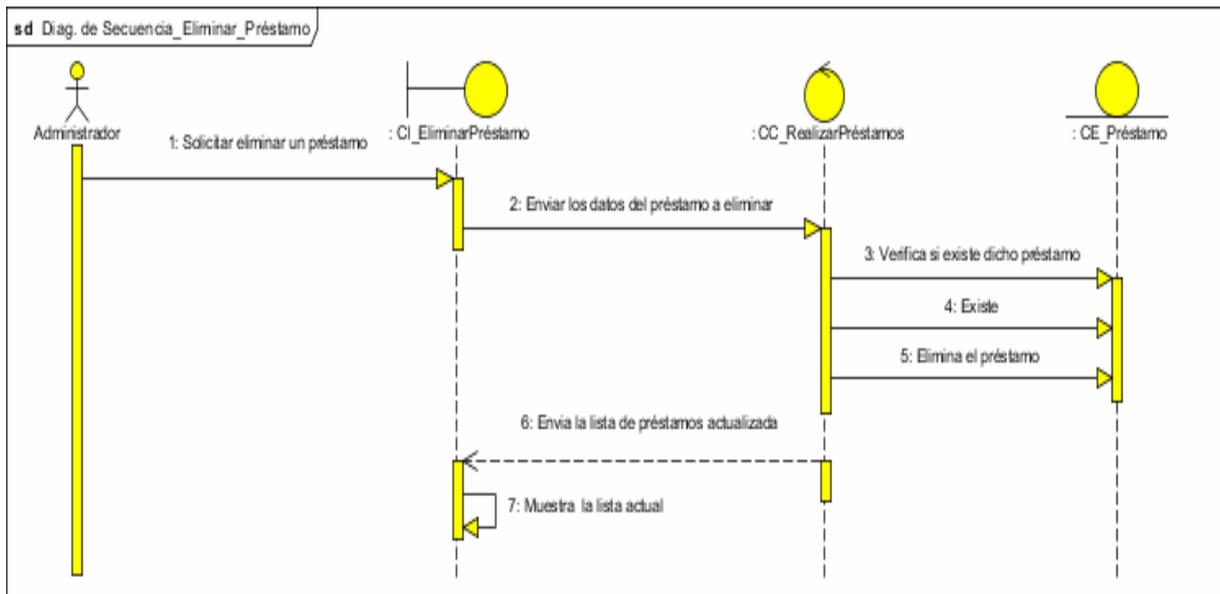


Figura 25: Diagrama de Secuencia Eliminar Préstamos

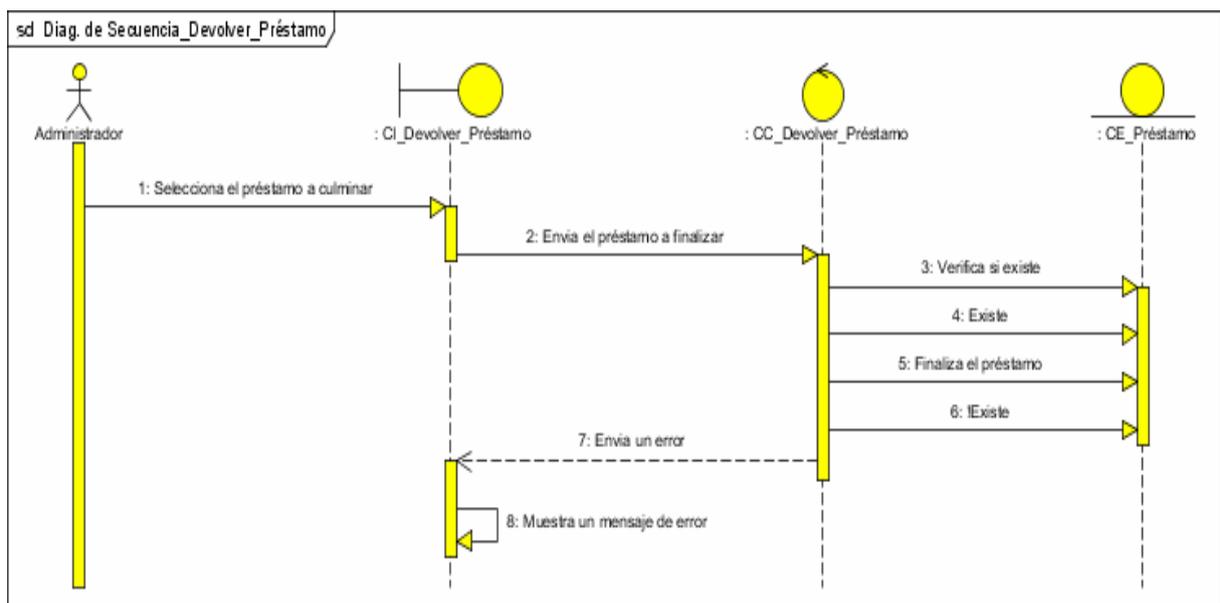


Figura 26: Diagrama de Secuencia Devolver Préstamos

ANEXO 4 Diagramas de Actividades

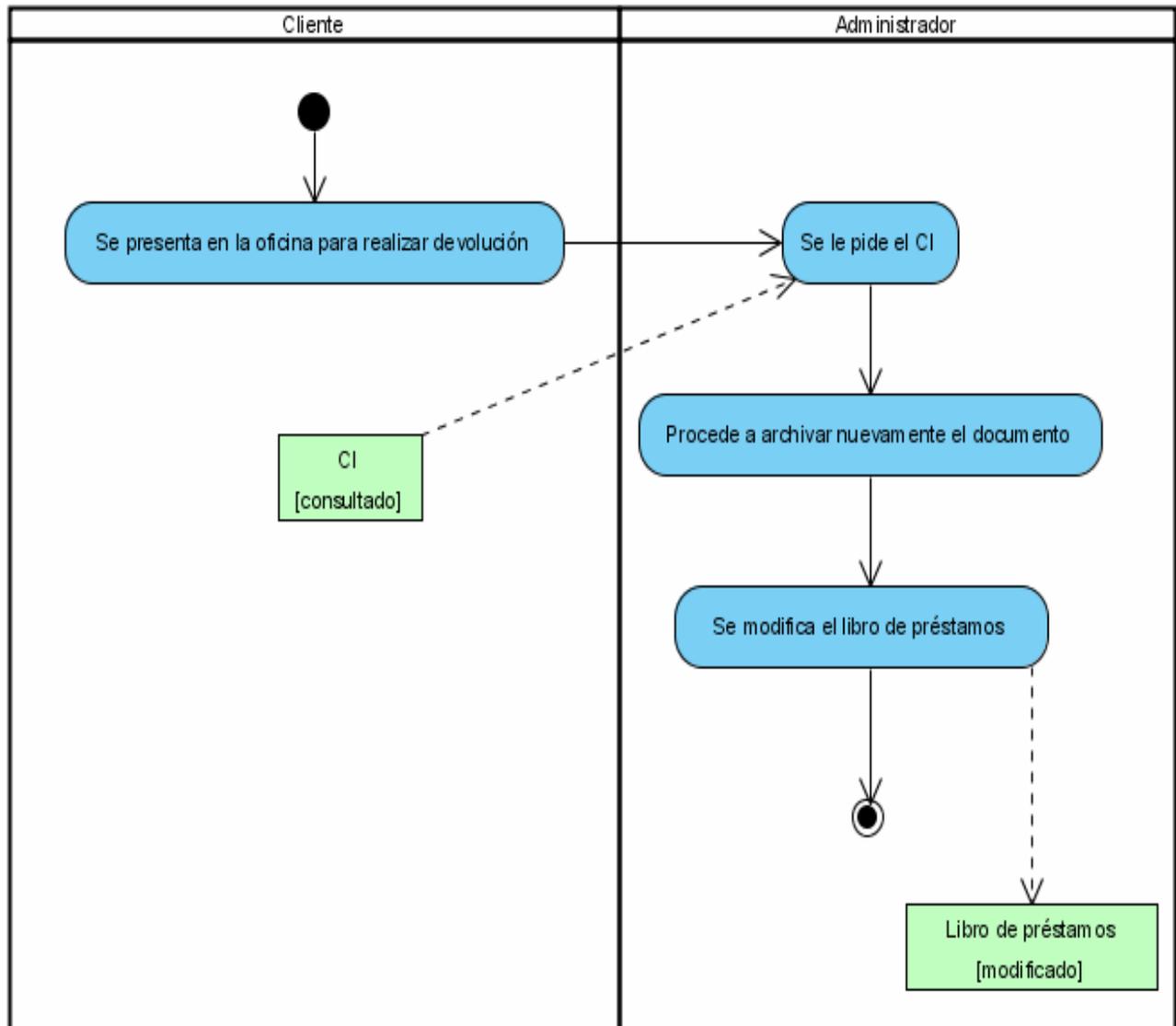


Figura 27: Diagrama de Actividades del CU Devolver Préstamos

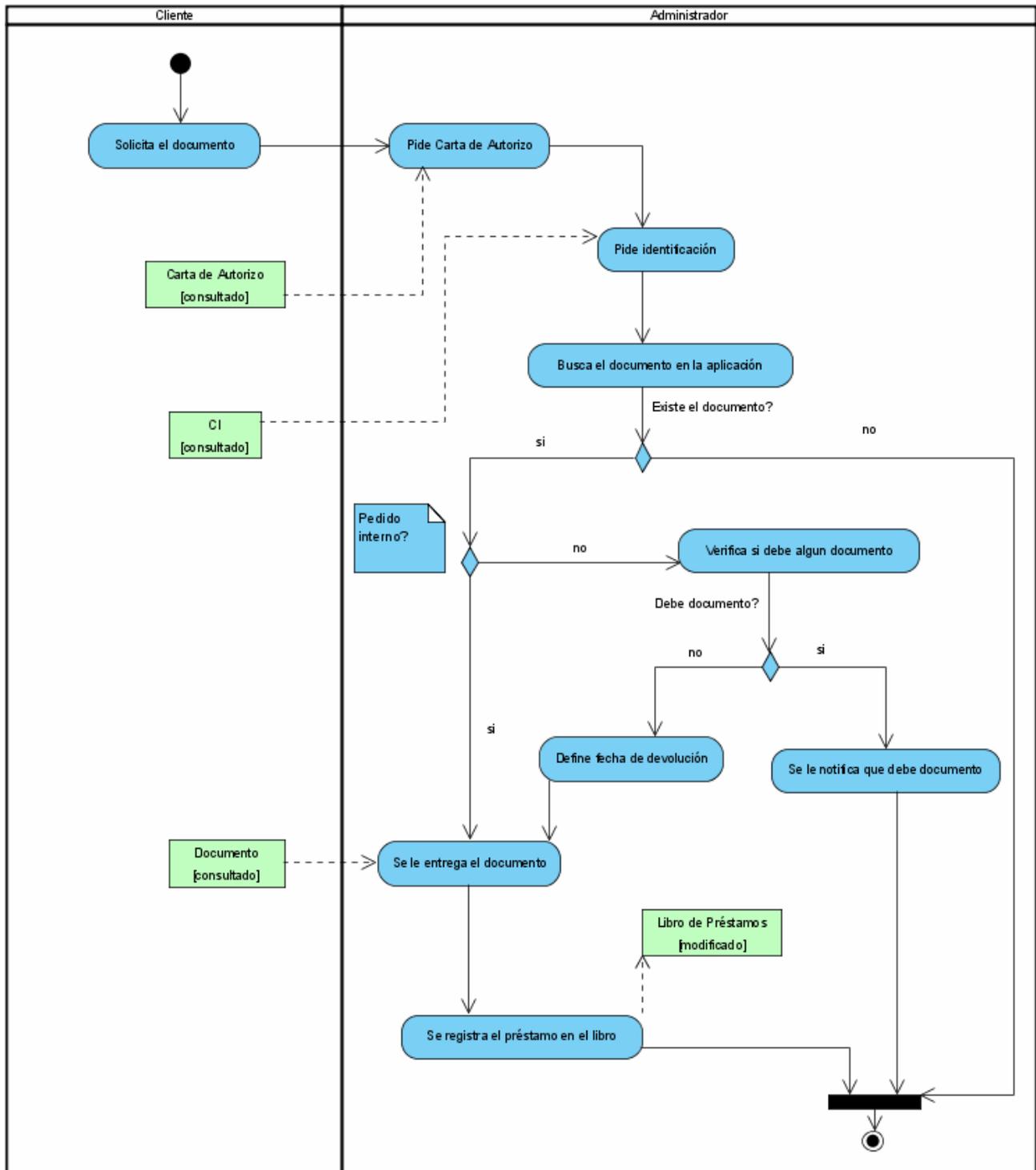


Figura 28: Diagrama de Actividades del CU Realizar Préstamos

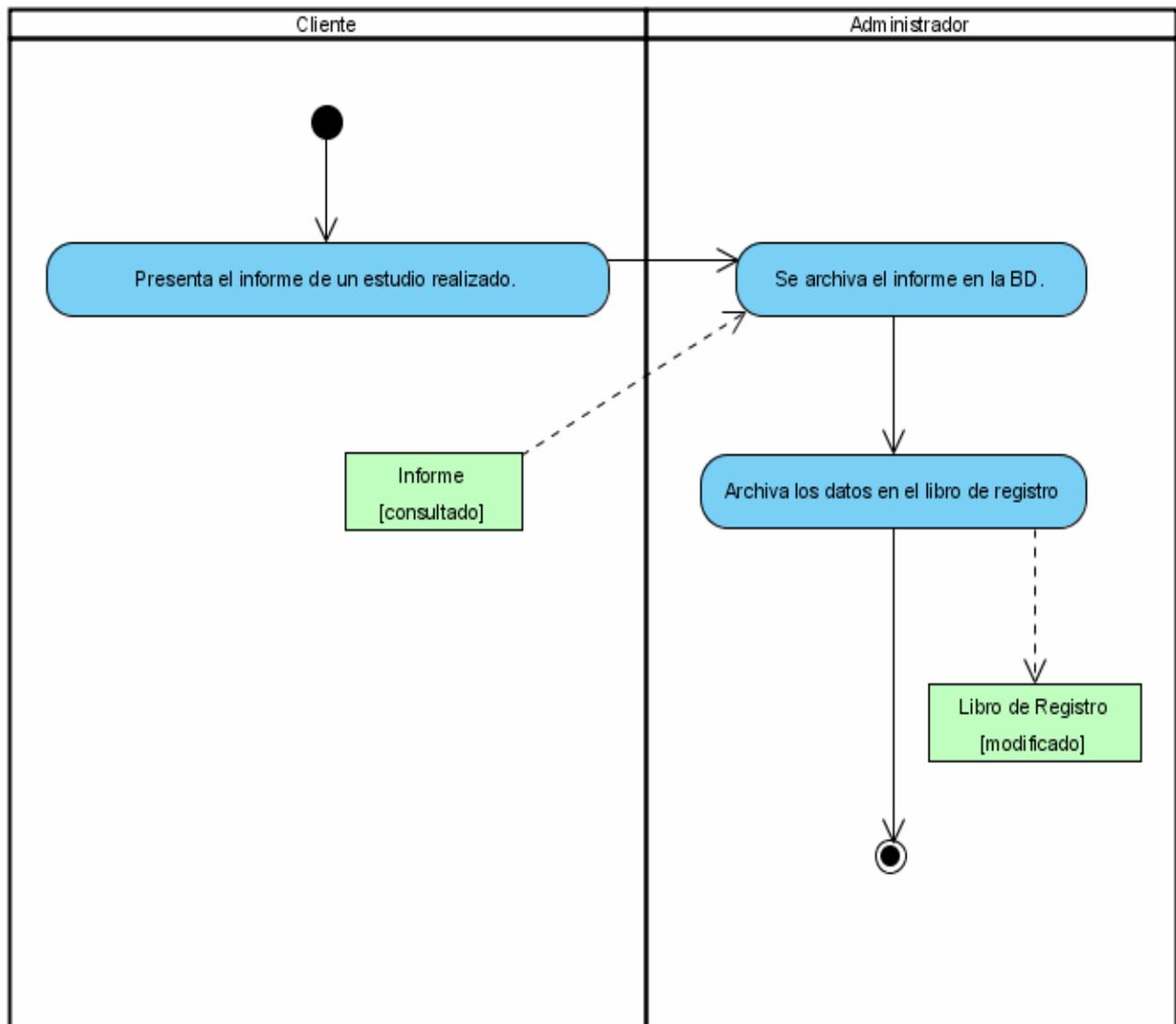


Figura 29: Diagrama de Actividades del CU Entregar Informes

ANEXO 5 Estructura de las clases del Modelo por Casos de Uso

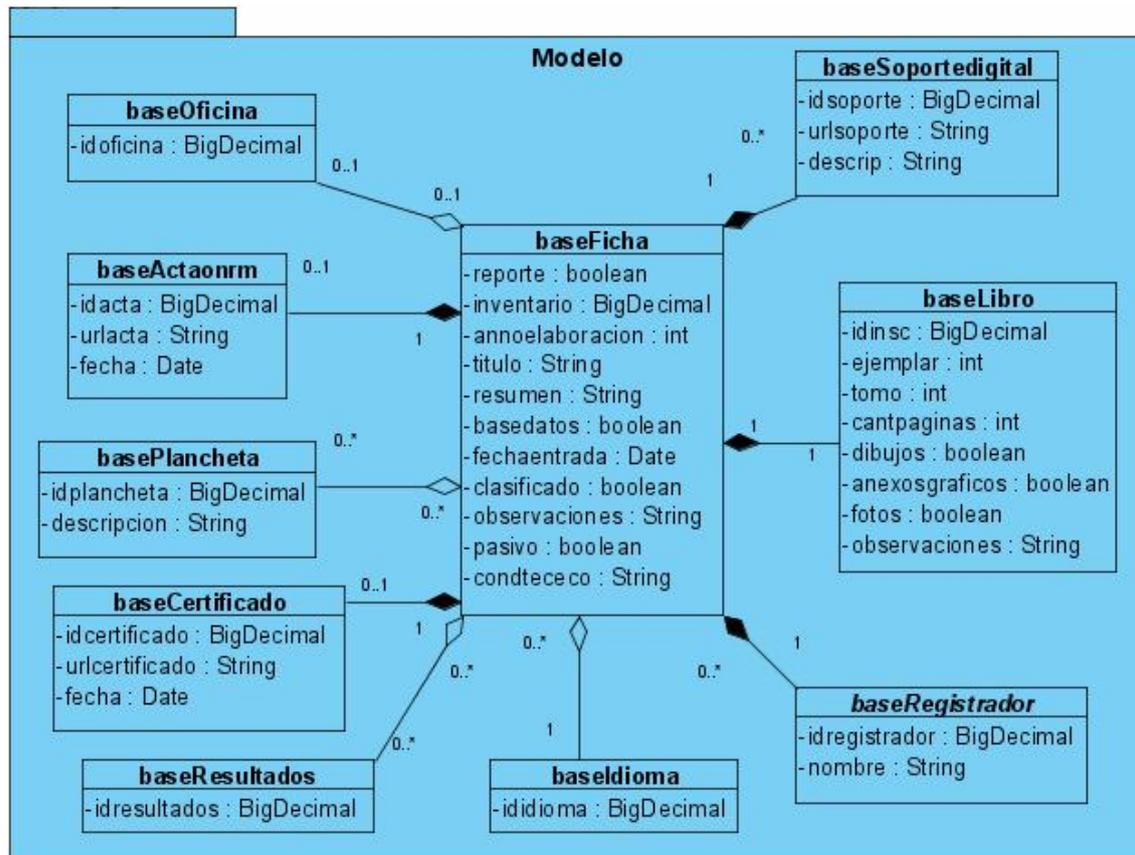


Figura 30: Clases del Modelo del CU Gestionar Informe

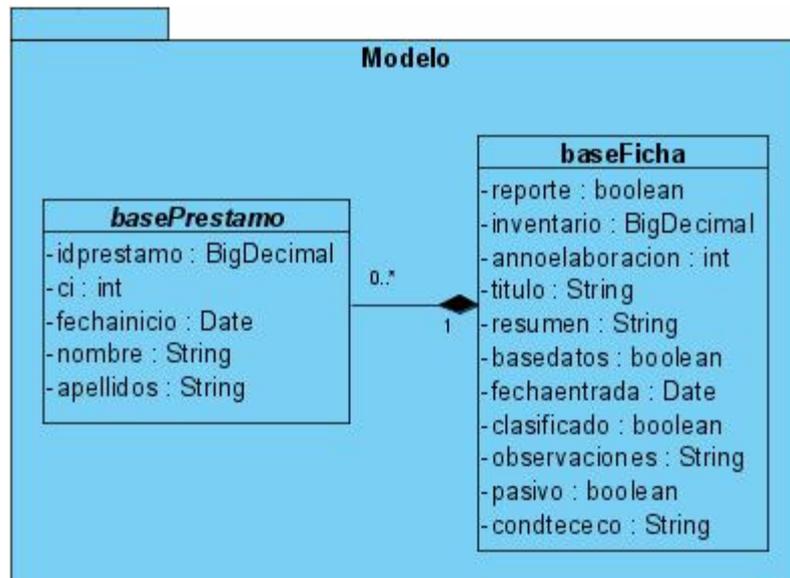


Figura 31: Clases del Modelo del CU Realizar Préstamo

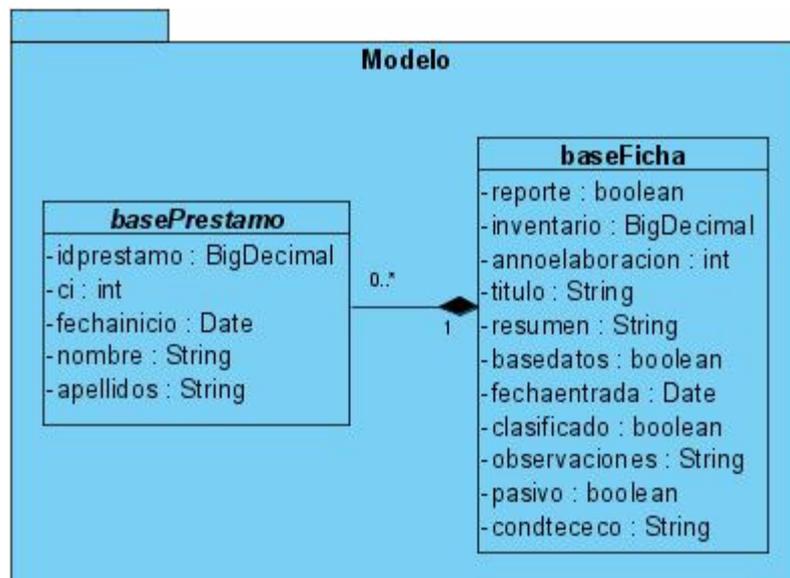


Figura 32: Clases del Modelo del CU Devolver Préstamo

ANEXO 6 Paquetes del Diagrama de Componentes

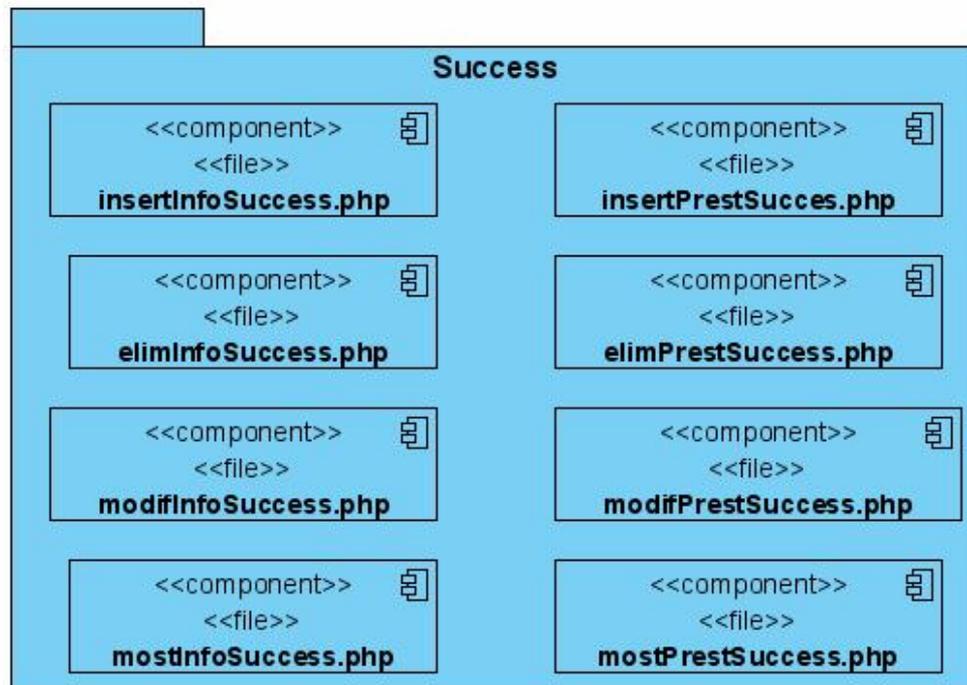


Figura 33: Paquete Success

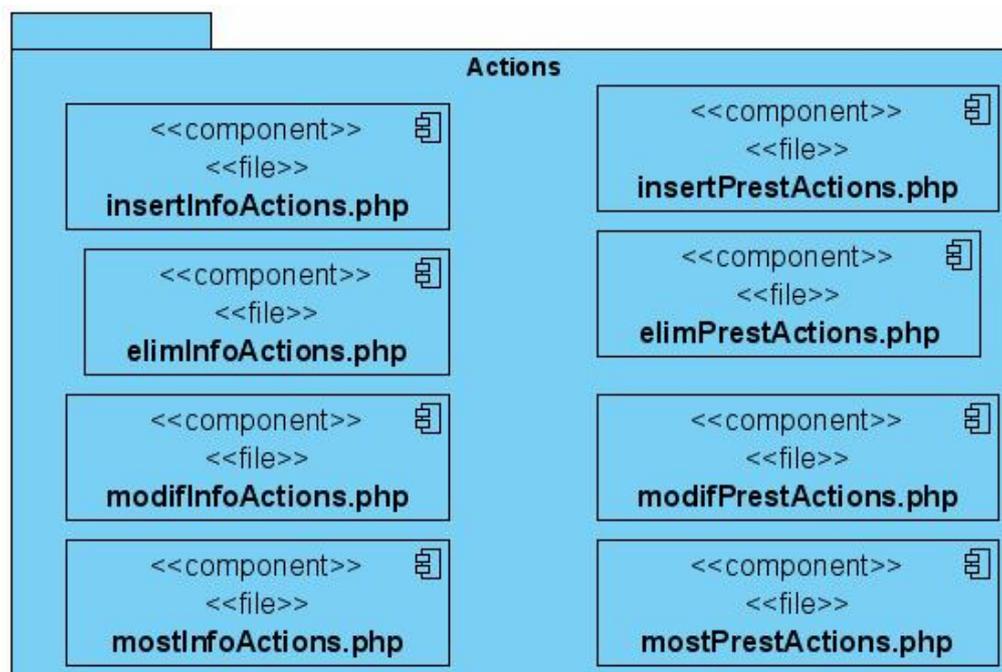


Figura 34: Paquete Actions

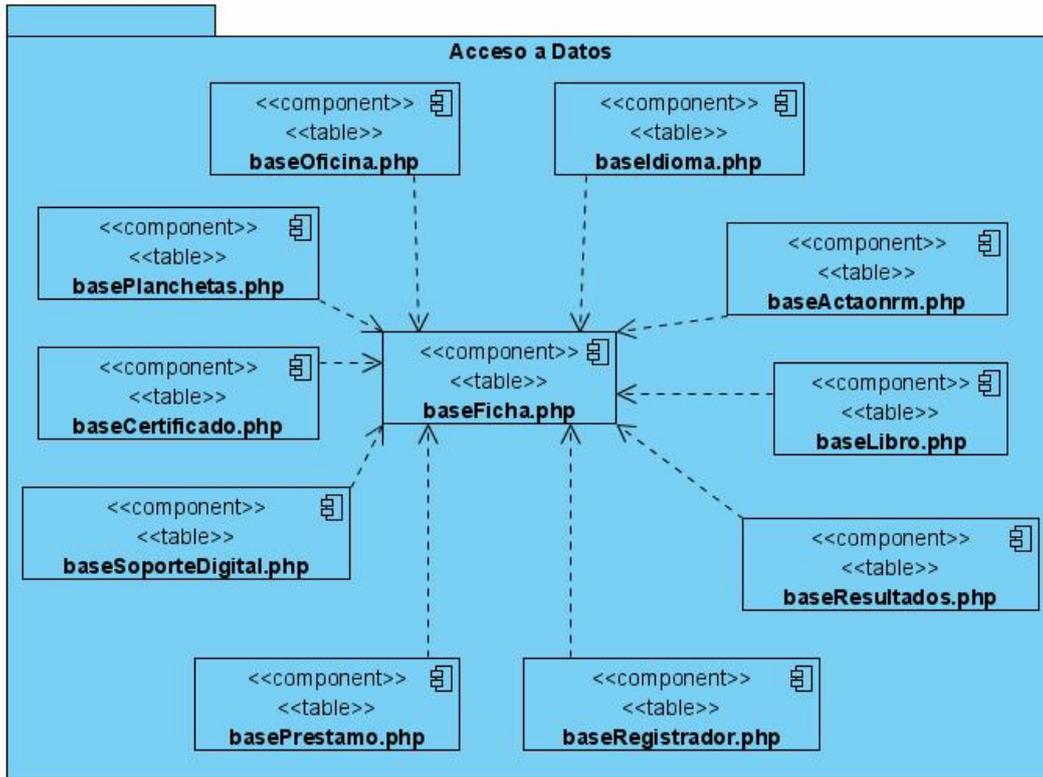


Figura 35: Paquete Acceso a Datos