

Universidad de las Ciencias Informáticas

Facultad 2



Título: Análisis y Diseño del Subsistema de Planificación, Análisis y Control. Módulo de **Control del Plan.**

Trabajo de Diploma para optar por el título de
Ingeniero Informático

Autor(es): Lismaylen Murga Fernández.

Ivette Barrientos Núñez.

Tutor: Ing. Temis Betancourt Villavicencio.

Cuidad de La Habana, Julio, 2008

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Firma del Autor
Lismaylen Murga Fernández.

Firma del Autor
Ivette Barrientos Núñez.

Firma del Tutor
Ing. Temis Betancourt Villavicencio

“Dime y lo olvido, enséñame y lo recuerdo, involúcrame y lo aprendo”

Benjamín Franklin

AGRADECIMIENTOS

Lisma:

“A mis padres, por todo el amor que siempre me han brindado, a mi abuela por ser tan especial, a mi hermano porque siempre he pensado en él para seguir adelante y a toda mi familia. Quiero agradecerles a mis amigas Aymara e Ismaila, por haberme aguantado estos años. A mi novio Marcos por apoyarme mucho y confiar en mí. A todas las personas que de una forma u otra contribuyeron a mi formación como profesional. A mi compañera de tesis, por su dedicación y esmero. Muchísimas gracias a todos. ”

Ivette:

A mis padres y mis tíos; y a toda mi familia, por haber creído siempre en mí y por apoyarme incondicionalmente en todo momento.

A mis amigos, que estuvieron siempre conmigo cuando me hizo falta una mano amiga y un hombro para desahogarme. Especialmente a Yanet y Leyanny por todo el amor y la paciencia que me ofrecieron. A mis hermanitas Rita, Daiamna y Yeilin, por enseñarme el significado de la verdadera amistad.

A Víctor, por estar en las buenas y en las malas, por hacerme tener fe en mí misma.

A mi compañera de tesis, por apoyarme y ayudarme en este último trayecto de mi vida estudiantil.

A todos a los que estuvieron, “Muchas Gracias”.

DEDICATORIA

A nuestros padres, por depositar toda su confianza en nosotras, por apoyarnos siempre, y por ser nuestros faros en todo momento.

A nuestros amigos, por ser la fuerza que nos mantuvo a flote cuando la familia estuvo lejos.

RESUMEN

El desarrollo de la Informática en Cuba ha dado pie a la informatización de diferentes sectores que sustentan la economía y el desarrollo social. En el marco de este proceso una de las organizaciones que se ha visto beneficiada es el Ministerio de Auditoría y Control (MAC). El mismo lleva a cabo diferentes tareas con el fin de mantener y regular la política del Estado y del Gobierno. El presente trabajo se centra en la necesidad de brindar una solución eficiente para la planificación y control de acciones de control que se realizan la Dirección de Planificación, Análisis y Control del MAC.

En el marco de este trabajo se analizarán las metodologías y herramientas que se utilizarán para la realización del sistema propuesto, así como los procesos ejecutados en dicha dirección. Se desarrollarán una serie de pasos para obtener un buen diseño que establecerá una base para lograr una correcta implementación del sistema propuesto, que brinde una solución real a los problemas que presenta la Dirección de Planificación, Análisis y Control en cuanto a la planificación y control de las acciones de control.

PALABRAS CLAVE

Auditoría, acción de control, planificación, control del plan, Modelo 001, Ministerio de Auditoría y Control, Dirección de Planificación, Análisis y Control, Parte operativo, Plan Trimestral, deficiencias, presuntos hechos delictivos, calificación.

ÍNDICE:

AGRADECIMIENTOS.....	I
DEDICATORIA.....	II
RESUMEN.....	III
INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1. Introducción.....	5
1.2. Definición de Auditoría.....	5
1.4. Desarrollo histórico de la Auditoría en Cuba.....	7
1.5. Planificación de la Auditoría.....	7
1.5.1. Objetivo.....	8
1.5.2. Plan de Auditoría en Cuba.....	8
1.5.2.1. Control de la Planificación en el MAC.....	9
1.6. Software relacionados.....	10
1.6.1. QAction.....	10
1.6.2. ISOdocument Software.....	10
1.7. Tecnologías, Técnicas, Herramientas y Metodologías Utilizadas.....	11
1.7.1. Sistema Operativo.....	11
1.7.2. Herramienta de modelado.....	12
1.7.2.1. Visual Paradigm.....	12
1.7.2.2. Microsoft Office Visio.....	13
1.7.2.3. IDEF0.....	14
1.7.3. Metodologías Utilizadas.....	14
1.7.3.1. RUP.....	14
1.7.3.2. Lenguaje de Modelado.....	15
1.7.4. Lenguaje(s) de programación.....	15
1.7.4.1. PHP5.....	16
1.7.4.2. JavaScript.....	16
1.7.5. Entorno(s) de Desarrollo Integrado.....	17
1.7.6. Gestor de base de datos.....	18
1.7.7. Servidor de aplicaciones web.....	19
1.7.8. Framework Utilizado.....	19
CONCLUSIONES:.....	21
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	23
2.1. Introducción:.....	23
2.2. Objeto de estudio:.....	23
2.3. Problema y situación problemática:.....	23
2.4. Objeto de automatización.....	24
2.4.1. Plan Nacional de Auditoría:.....	24
2.4.2. Parte operativo:.....	25
2.4.3. Cumplimiento del Plan Trimestral:.....	25
2.5. Información que se maneja.....	25
2.6. Propuesta de sistema.....	26
2.7. Modelo de negocio.....	26
2.7.1. Plan Nacional de Auditorías.....	26

2.7.2.	Elaboración Parte Operativo	27
2.7.3.	Cumplimiento del Plan Trimestral.....	29
2.8.	Especificación de los requisitos de software.....	29
2.8.1.	Requerimientos Funcionales.....	29
2.8.2.	Requisitos No Funcionales:.....	30
2.9.	Definición de los Casos de Uso.	35
2.9.1.	Actores del Sistema:	35
2.9.2.	Listado de Casos de Uso	35
2.10.	Diagramas de Casos de Uso	36
2.10.1.	Definición de los Casos de Uso:	37
2.10.1.1.	Gestionar Plan de Acciones de Control:	37
2.10.1.2.	Conciliar Plan Anual:.....	44
2.2.1.1.	Control de la Planificación	50
CONCLUSIONES:	53
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.....		54
3.1.	Introducción:.....	54
3.2.	Descripción de la Arquitectura y el Framework utilizados:	54
3.2.1.	Descripción del Framework.....	54
3.2.2.	Arquitectura y Patrones:.....	57
3.3.	Clases del Diseño:.....	61
3.3.1.	Gestionar Plan Anual:	62
3.3.2.	Control de la Planificación:.....	63
3.3.3.	Conciliar Plan Anual.....	64
3.4.	Modelo de Datos:	65
3.5.	Diagramas de Secuencia.....	65
3.6.	Estudio de la Factibilidad	66
3.6.1.	Descripción del Método utilizado	66
3.6.2.	Cálculo del esfuerzo.....	67
CONCLUSIONES	70
CONCLUSIONES	71
RECOMENDACIONES	72
REFERENCIAS BIBLIOGRÁFICAS	73
BIBLIOGRAFÍA.....	74
ANEXOS	75

INTRODUCCIÓN

Cuba ha venido experimentando avances en torno a la preparación, cada vez, más actualizada de su sociedad, con vista a darle seguimiento al mundo tan cambiante y dinámico en el que vivimos, en donde la máquina es eslabón imprescindible, ya no solo para hacer mover una industria sino también para su utilización en la educación, la salud, la economía, en fin, en todo el espacio humano.

Desde hace algunos años Cuba se ha visto inmersa en un proceso de informatización para poder convertir los conocimientos y las tecnologías de la información en instrumentos a disposición del avance de nuestro pueblo.

Dentro del marco de este proceso de informatización hemos visto muchos avances en cuanto a la extensión de la Informática, como disciplina científico técnica, a todos los rincones y personas del país. Dicho proceso ha reportado beneficios en sectores sociales como son Educación y Salud. También se está llevando a cabo el proceso de informatizar los diferentes ministerios que sustentan la economía y el desarrollo social.

Entre los primeros ministerios a informatizar íntegramente está el Ministerio de Auditoría y Control (MAC¹) que es el encargado de dirigir, ejecutar y controlar la aplicación de la política del Estado y del Gobierno en materia de Auditoría Gubernamental, Fiscalización y Control Gubernamental; así como para regular organizar, dirigir y controlar, metodológicamente, el Sistema Nacional de Auditoría. Este está constituido por direcciones como estructura básica del mismo, entre las que se encuentran: Dirección de Atención al Sistema Nacional de Auditoría, Dirección de Auditorías y Controles Especiales, Dirección de Atención a la Ciudadanía y Dirección de Planificación Análisis y Control, entre otras.

La Dirección de Planificación, Análisis y Control es la encargada de realizar una planificación anual de todas las auditorías a partir de los resultados del año anterior y teniendo en cuenta las directivas trazadas para el año siguiente. Además esta dirección pone en marcha una serie de procedimientos vinculados a ella como: **Registro de Auditores:** que gestiona toda la información de auditores y la actualización de la dispensa; **elaboración de parte operativo quincenal:** información quincenal de todas aquellas auditorías planificadas y especiales que están en curso; **elaboración de cartas:** informa sobre los resultados de las

¹ Ministerio de Auditoría y Control.

acciones de control que obtienen calificaciones de deficiente y mal en las entidades; **elaboración de las directivas:** metas y perspectivas para la elaboración de la planificación anual y **Elaboración de Informe de las empresas en Perfeccionamiento Empresarial:** informar la marcha de estos procesos a la alta dirección de dichas empresas; además de llevar un sistema de indicadores para la evaluación y calificación del funcionamiento de las Delegaciones del Ministerio de Auditoría y Control.

La mayor parte de los procedimientos que se llevan a cabo en esta dirección se realizan de forma manual y con escasa tecnología, es decir, parte de los procesos se realizan en hojas de Excel para poder almacenar las acciones de control. Los documentos generados son guardados y archivados posteriormente de forma tradicional, en grandes estantes. Por otra parte la comunicación entre las direcciones del MAC y las delegaciones provinciales con la Dirección de Planificación, Análisis y Control es muy trabajosa, ya que se realizan personalmente. Los procesos no se realizan con la adecuada organización, no usan un formato específico para realizar la planificación. No existe una forma de obtener un control total de las acciones de control que están previstas. Esta forma de trabajo no presenta resultados satisfactorios ya que imposibilita la búsqueda de la información y el almacenamiento de la misma con la rapidez requerida. No brinda un dominio total de los procesos de planificación y control de las acciones de control. Estas, y otras dificultades atentan contra el buen funcionamiento de la Dirección de Planificación, Análisis y Control.

Tomando en cuenta la necesidad de dar una solución a la situación antes planteada, surge el siguiente **problema científico:** Inexistencia de un mecanismo para la eficiente elaboración y control de la planificación de las acciones de control por parte de la Dirección de Planificación, Análisis y Control del MAC.

El presente trabajo se propone dar solución al problema existente teniendo como **objeto de estudio** los procesos de la Dirección de Planificación, Análisis y Control del MAC, delimitando el **campo de acción**, a los procesos de planificación y control de acciones de control que se manejan en la Dirección de Planificación, Análisis y Control del MAC.

Para la realización del presente trabajo se propone como **Objetivo General** del mismo:

- Diseñar un sistema que permita planificar y controlar las acciones de control en la Dirección de Planificación, Análisis y Control del MAC.

Para cumplir el objetivo general planteado se llevan a cabo las siguientes **tareas de investigación**:

- Investigar y analizar los procesos que se llevan a cabo en la Dirección de Planificación, Análisis y Control del MAC.
- Realizar un estudio sobre sistemas semejantes que se utilicen para los procesos de planificación y control de acciones de control.
- Estudiar y profundizar las tecnologías y metodologías de trabajo definidas para el desarrollo de las aplicaciones del proyecto.
- Llevar a cabo entrevistas al cliente como una técnica para el levantamiento de requisitos.
- Obtener un modelo del sistema a partir de las necesidades del cliente y de los conocimientos adquiridos.

Resumen por Capítulos:

En el **Capítulo 1** se brindará una panorámica acerca de la auditoría, además de presentarse aspectos tales como su definición, desarrollo histórico internacional y nacional. También se abordarán conceptos relacionados con la auditoría, como la planificación de la misma y cómo se realiza el control de esta planificación además de varios software relacionados con este aspecto.

Se ofrecerá información de las Tecnologías, Técnicas, Herramientas y Metodologías utilizadas, así como una descripción de por qué son las escogidas y sus características más importantes.

En el **Capítulo 2** se mostrarán de forma detallada los procesos relacionados con el negocio. Se explicará cada proceso que se encuentra dentro del objeto de automatización, así como una panorámica de la propuesta del sistema que se quiere obtener. Los procesos del negocio se mostrarán mediante diagramas. Se expondrán los requisitos funcionales y no funcionales del sistema. Por último se definirán los actores del sistema, con sus descripciones; se visualizará el diagrama de casos de uso, así como la descripción de los casos de uso del sistema.

En el **Capítulo 3** y final se brindarán las justificaciones necesarias y la importancia de utilizar un framework para el desarrollo del subsistema, además de una explicación detallada del Symfony que será

el escogido y los patrones de diseño web que utiliza. Se visualizarán las clases del diseño de los casos de uso más significativos, así como el modelo de datos elaborado. Por último se desarrollará el estudio de la factibilidad, donde se expondrán los cálculos obtenidos y la descripción del COCOMO II con salidas de Puntos de Función que será el método utilizado.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1. Introducción

En el desarrollo de este capítulo se brindará una panorámica acerca de la auditoría, mostrando aspectos tales como su definición, desarrollo histórico internacional y nacional. También se abordarán conceptos relacionados con la auditoría, como la planificación de la misma y cómo se realiza el control de esta planificación además de varios software relacionados con este aspecto.

Se ofrecerá información de las Tecnologías, Técnicas, Herramientas y Metodologías utilizadas, así como una descripción de porque son las escogidas y sus características más importantes.

1.2. Definición de Auditoría

“Es la verificación sistemática y técnica de los libros y registros de contabilidad, financieros y legales, con el fin de determinar la exactitud e integridad de la contabilidad y el de mostrar las operaciones y la verdadera situación financiera de la empresa y unidad presupuestada.

La auditoría completa abarca todos los aspectos susceptibles de verificación en una entidad, por ejemplo, cuentas de balance general, del estado de ganancias y pérdidas; comprobación de ingresos y gastos, cobros, pagos, revisión de pases y sumas; chequeos de los libros de entrada original; chequeo de compras; de nóminas de sueldos y jornales, etc.” **(SANTANDER 1987)**

1.3. Historia de la Auditoría

La auditoría como profesión fue reconocida por primera vez bajo la Ley Británica de Sociedades Anónimas de 1862 y el reconocimiento general tuvo lugar durante el período de mandato de esta Ley donde se cita que: "Un sistema metódico y normalizado de contabilidad era deseable para una adecuada información y para la prevención del fraude". También reconocía..."Una aceptación general de la necesidad de efectuar una versión independiente de las cuentas de las pequeñas y grandes empresas". Desde 1862 hasta 1905, la profesión de la auditoría creció y floreció en Inglaterra, y se introdujo en los Estados Unidos hacia 1900. En Inglaterra se siguió haciendo hincapié en cuanto a la detección del fraude como objetivo primordial de la auditoría.

En los que podría llamarse los días en los que se formó la auditoría, a los estudiantes se les enseñaban que los objetivos primordiales de ésta eran:

- La detección y prevención de fraude.
- La detección y prevención de errores.

Sin embargo, en los años siguientes hubo un cambio decisivo en la demanda y el servicio, y los propósitos actuales son:

- El cerciorarse de la condición financiera actual y de las ganancias de una empresa.
- La detección y prevención de fraude, siendo éste un objetivo menor.

Este cambio en el objetivo de la auditoría continuó desarrollándose, no sin oposición, hasta aproximadamente 1940. En este tiempo existía un cierto grado de acuerdo en que el auditor podía y debería no ocuparse primordialmente de la detección de fraude. El objetivo primordial de una auditoría independiente debe ser la revisión de la posición financiera y de los resultados de operación como se indica en los estados financieros de los clientes, de manera que pueda ofrecerse una opinión sobre la adecuación de estas presentaciones a las partes interesadas.

Paralelamente al crecimiento de la auditoría independiente en los Estados Unidos, se desarrollaba la auditoría interna y del Gobierno, lo que entró a formar parte del campo de la auditoría. A medida que los auditores independientes se apercebieron de la importancia de un buen sistema de control interno y su relación con el alcance de las pruebas a efectuar en una auditoría independiente, se mostraron partidarios del crecimiento de los departamentos de auditoría dentro de las organizaciones de los clientes, que se encargaría del desarrollo y mantenimiento de unos buenos procedimientos del control interno, independientemente del departamento de contabilidad general. Progresivamente, las compañías adoptaron la expansión de las actividades del departamento de auditoría interna hacia áreas que están más allá del alcance de los sistemas contables. En nuestros días, los departamentos de auditoría interna son revisiones de todas las fases de las corporaciones, de las que las operaciones financieras forman parte.

La auditoría gubernamental fue oficialmente reconocida en 1921 cuando el Congreso de los Estados Unidos estableció la Oficina General de Contabilidad.

El concepto de Auditoría ha evolucionado a lo largo del tiempo desde su surgimiento como profesión hasta nuestros días. Primero esta se basaba en buscar si se habían cometido fraudes en el negocio que estaban auditando. Más tarde las empresas crecieron, los negocios, al ser más grandes,

empezaron a separar el capital y la Administración y aquí el auditor sin dejar de realizar el trabajo que hacía en un principio empezó a verificar y certificar que la información que esos administradores le pasan a través de la cuenta de resultados, fuera veraz. Luego con la aparición de nuevas tecnologías, ordenadores, etc, las transacciones en las empresas se hacen más voluminosas, esto hace que los auditores tengan que revisar las cuentas y el sistema de control interno de las compañías. Por último, en nuestros días, los auditores, además de realizar todas las acciones mencionadas anteriormente, tienen que emitir un informe con el resultado de la auditoría y presentarlo a la empresa para ver si esta está de acuerdo con la forma en que se ha realizado dicho informe.

1.4. Desarrollo histórico de la Auditoría en Cuba.

La auditoría tiene sus cimientos desde la época colonial donde se utilizaba para identificar los resultados obtenidos y determinar el importe correcto del impuesto a pagar a los españoles. Posteriormente en la época republicana se realizaban auditorías financieras que correspondían a intereses de casa matrices norteamericanas y eran realizadas por firmas de auditores independientes. También se realizaban auditorías fiscales y auditorías Internas. Seguidamente desde el triunfo de la Revolución y hasta 1990 se llevaron a cabo auditorías financieras de carácter estatal e internas, y surgen los inspectores fiscales. Y desde ese año hasta la actualidad tienen lugar las auditorías financieras tanto de carácter estatal como privado, con la participación de firmas cubanas de auditores independientes, las auditorías Fiscales y las Auditorías Internas. El organismo estatal encargado de consolidar el trabajo de fiscalización y dirigir, ejecutar y controlar la aplicación de la política del Estado y Gobierno en materia de auditoría y control gubernamentales, es el Ministerio de Auditoría y Control que fue fundado el **3 de mayo de 2001**. Uno de los procesos mas significativos que lleva a cabo el Ministerio, es la planificación de las auditorías.

1.5. Planificación de la Auditoría.

La planificación a nivel mundial es un proceso continuo. Los objetivos, el alcance y la metodología identificados en esta etapa pueden modificarse dependiendo de las situaciones que surjan cuando se estén realizando las pruebas de las operaciones fiscales que se examinarán. Los cambios que se hagan, tendrán que ser notificados y aprobados.

La planificación de la auditoría comprende el desarrollo de una estrategia global para su administración, al igual que el establecimiento de un enfoque apropiado sobre la naturaleza, oportunidad y alcance de los procedimientos de auditoría que deben aplicarse. El planeamiento también permitirá que el equipo de auditoría pueda hacer uso apropiado del potencial humano disponible.

El proceso de la planificación permite identificar las áreas más importantes y los problemas potenciales del examen, evaluar el nivel de riesgo y programar la obtención de la evidencia necesaria para examinar los distintos componentes de la entidad auditada. El auditor planifica para determinar de manera efectiva y eficiente la forma de obtener los datos necesarios e informar acerca de la gestión de la entidad; la naturaleza y alcance de la planificación puede variar según el tamaño de la entidad, el volumen de sus operaciones, la experiencia del auditor y el nivel organizacional.

1.5.1. Objetivo

El objetivo principal de la planificación, consiste en determinar adecuada y razonablemente los procedimientos de auditoría que correspondan aplicar, cómo y cuándo se ejecutarán, para que se cumpla la actividad en forma eficiente y efectiva.

La planificación permite identificar lo que debe hacerse en una auditoría, por quién y cuándo. Generalmente, la planificación es vista como una secuencia de pasos que conducen a la ejecución de procedimientos sustantivos de auditoría; sin embargo, este proceso debe proseguir en forma continua durante el curso de la auditoría.

1.5.2. Plan de Auditoría en Cuba.

En nuestro país, el MAC y específicamente la Dirección de Planificación, Análisis y Control, es la encargada de realizar la planificación de las acciones de control. Para esto hacen un Plan de Acciones de Control por donde se regirán las demás direcciones para realizar las acciones de control de todo un año.

Para realizar este plan cada nivel del sistema, díganse las Delegaciones y Direcciones del MAC, las UCAI² de los CAP³ y OACES⁴ elaboran su plan teniendo en cuenta, además de las directivas y los

² Unidades Centrales de Auditoría Interna.

aspectos prioritarios particulares que sean de su interés, el balance de la fuerza de trabajo, determinando el fondo de tiempo disponible y el fondo de tiempo de reserva; los datos históricos del cumplimiento de planes anteriores y otros indicadores que contribuyan a una optimización de los recursos disponibles.

Las Direcciones y Delegaciones elaboran su plan, incluyendo las auditorías, comprobaciones especiales, visitas de supervisión e inspecciones gubernamentales que se ejecutan a su nivel y se concilia con la Dirección de Planificación, Análisis y Control conforme al cronograma que se establezca al efecto. Después de esta conciliación, la Dirección de Planificación, Análisis y Control elabora el plan anual a nivel nacional con la información de todas las acciones de control programadas para el año en curso, el cual es aprobado finalmente por la Ministra.

Las UCAI elaboran paralelamente su plan, en el que incluyen las acciones de control que se ejecutan por su nivel y las planificadas por las UAI⁵ y AI⁶ de su sistema desglosado por provincias.

El plan de las UCAI de los OACE se concilia con la Dirección de Planificación, Análisis y Control y el de las UCAI de los CAP con las Delegaciones de acuerdo al calendario que se establezca y después de realizados los ajustes recomendados durante la misma, se aprueba por el Jefe del órgano u organismo al que se subordina la UCAI.

El ejemplar aprobado en el modelo establecido se entrega en la Dirección de Planificación, Análisis y Control, en soporte papel debidamente firmado y acuñado y en formato digital, momento en que se supervisa la correspondencia del plan aprobado con los intereses convenidos.

Después de elaborado el Plan Anual Nacional, es necesario llevar a cabo un control sistemático del cumplimiento del mismo.

1.5.2.1. Control de la Planificación en el MAC.

Cada Dirección y Delegación, así como las UCAI ejercen el control sistemático sobre la ejecución del plan de acuerdo con sus características.

³ Consejos de la Administración de las Asambleas Provinciales del Poder Popular y del municipio especial Isla de la Juventud.

⁴ Organismos de la Administración Central del Estado.

⁵ Unidades de Auditoría Interna.

⁶ Auditor Interno.

El control del cumplimiento del plan, se registra en el propio plan anual y se complementa con otros datos importantes como afectación económica, presuntos hechos delictivos, deficiencias, etc, de forma tal que permita operativamente brindar la información que se solicite.

Los cambios que se necesiten introducir al plan, se autorizan por el nivel de dirección que lo aprobó, previa consulta por escrito y debidamente fundamentada. La Dirección de Planificación, Análisis y Control, que es la encargada de tramitar y actualizar la información correspondiente.

1.6. Software relacionados.

En la actualidad existen software para la Planificación y Control de las Auditorías. A continuación vamos a dar una breve panorámica sobre dos de ellos:

1.6.1. QAction:

QAction es programa destinado a las empresas que están interesadas en llevar un buen control de las auditorías realizadas al sistema de calidad, así como las acciones preventivas y correctivas. Este software permite planificar las auditorías al sistema de calidad, genera de forma automática hojas de verificación, emite cartas predeterminadas de notificación de auditorías. Además de verificar y registrar la implantación y efectividad de las acciones correctivas efectuadas. Controla las quejas de los clientes y entre otras emite un informe enviado automáticamente a la dirección sobre los atrasos en las respuestas de acciones inmediatas, análisis de causa raíz, acciones correctivas permanentes y cierre de acciones correctivas.

Para tener acceso a este software es necesario pagar varias licencias, por ejemplo la licencia principal tiene un costo de 790 EUR y la licencia empresarial se monta en 2990 EUR.

1.6.2. ISOdocument Software

Este software comprende el control y seguimiento de las auditorías tanto internas como externas.

Programa el cronograma de auditorías anuales. En este se indican las fechas y que tipo de auditorías (Externa, Interna) que se llevaran a cabo en un año. Luego de elaborado el programa es enviado a revisión para ser aprobado y es enviada la notificación a los usuarios responsables. A medida que se

aproxima la fecha de la auditoría, el modulo envía notificaciones a los usuarios responsables para que elaboren los planes de auditorías.

En el plan de auditoría se especifica el auditor líder, las referencias a manuales y documentos, el objetivo, alcance, se especifica el esquema de distribución del documento y en este podremos especificar las actividades que se realizaran por área durante la auditoría.

En el sistema se puede localizar los documentos con facilidad, ya que están organizados por plan, por programa, por informe, por actividad y estos a su vez clasificados por su estado. Todos estos documentos quedan registrados con su debida traza de personas que aprobaron los documentos.

La plataforma es Cliente-Servidor; como requerimiento de software tenemos para el Cliente: Lotus Notes Release 4.5 o superior y para el Servidor: Lotus Dominio Server Release 4.5 o superior. El mismo es un software propietario y con un alto precio, para utilizarlo.

1.7. Tecnologías, Técnicas, Herramientas y Metodologías Utilizadas

En el desarrollo de todo sistema informático es de vital importancia la selección de las herramientas a utilizar, paso que garantizará, de realizarse correctamente, un óptimo desempeño del sistema. Para el desarrollo del sistema al cual se refiere este documento, la selección se realizó teniendo en cuenta la infraestructura tecnológica del Ministerio de Auditoría Control y el desarrollo que ha alcanzado el software libre en Cuba y el mundo.

1.7.1. Sistema Operativo

Para el desarrollo de este trabajo con el cliente se ha seleccionado como SO (Sistema Operativo) Ubuntu Gutsy Gibbon (7.10) ya que el mismo es una distribución Linux que ofrece un SO predominantemente enfocado a ordenadores de escritorio. Ubuntu concentra su objetivo en la facilidad de uso, la libertad de uso, los lanzamientos de versiones regulares (cada 6 meses) y la facilidad en la instalación. Presenta varias características de gran importancia como:

- Posee una gran colección de aplicaciones prácticas y sencillas para la configuración de todo el sistema, a través de una interfaz gráfica útil para usuarios que se inician en Linux.

- El sistema incluye funciones avanzadas de seguridad y entre sus políticas se encuentra el no activar, de forma predeterminada, procesos latentes al momento de instalarse. Por eso mismo, no hay un firewall predeterminado, ya que no existen servicios que puedan atentar a la seguridad del sistema.
- Mejora la accesibilidad y la internacionalización, de modo que el software está disponible para tanta gente como sea posible.
- Cuenta con la herramienta GUI, que permite seleccionar fácilmente la tarjeta gráfica y resolución de pantalla.
- Presenta un sistema de impresión, crea una impresora virtual "PDF printer" por defecto, que puede usarse para generar PDF desde cualquier aplicación que no tenga salida nativa a PDF.

Este SO además de tener un entorno fácil de trabajar presenta una amplia gama de software equivalente a los utilizados en Windows. Ubuntu es capaz de actualizar a la vez todas las aplicaciones instaladas en la máquina a través de repositorios, a diferencia de otros sistemas operativos comerciales como Windows y MacOS, donde esto no es posible. Ubuntu no cobra honorarios por la suscripción de mejoras de la "Edición Enterprise". Ubuntu está opcionalmente disponible en DVD, para evitar su dependencia de Internet

Para el servidor de este proyecto se utilizará la versión server de la misma distribución. Ubuntu Server Edition (7.10).

1.7.2. Herramienta de modelado

Las herramientas CASE, son un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación.

1.7.2.1. Visual Paradigm.

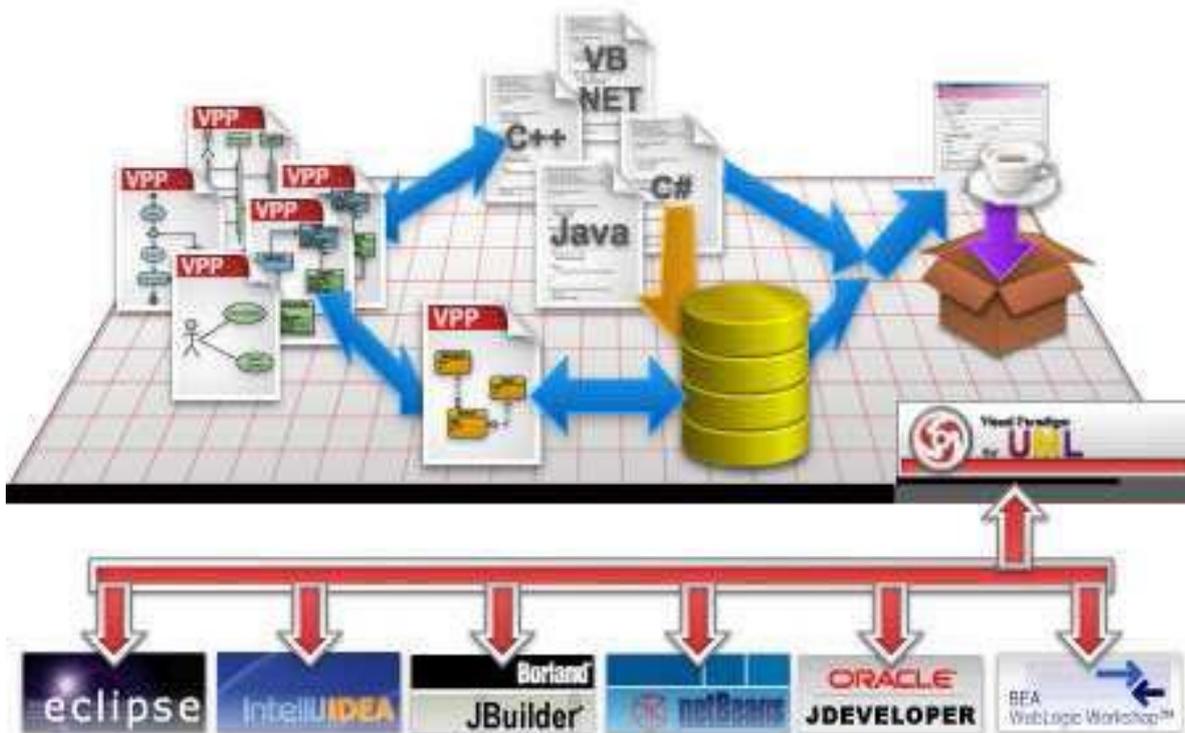
Es la propuesta para el modelado y es la escogida ya que dentro de sus características más importantes tenemos: es multiplataforma, presenta integración con varios IDE's,⁷ como: NetBeans (de

⁷ Entorno de desarrollo integrado o en inglés *Integrated Development Environment* es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica.

Sun Microsystems), JDeveloper (de Oracle) y Eclipse (de IBM) que será el utilizado en el sistema, además permite la generación de código fuente a varios lenguajes como PHP5, la generación del modelo de la base de datos para PostgreSQL, entre otros gestores de base de datos. No se inclina por ninguna metodología específica (se puede modelar cualquier proceso, ya sea RUP que es el utilizado, o cualquier otro como FDD, XP, etc.). También brinda la posibilidad de exportar imágenes jpg, png y svg. Esta herramienta presenta unas características graficas muy cómodas que facilitan la realización de los diagramas de modelado que sigue el estándar de UML que son:

- Diagramas de clase, Casos de Uso, Comunicación, Secuencia, Estado, Actividad, Componentes, etc.

La mayoría usados en el desarrollo del presente trabajo. A continuación se muestra una imagen con las características descritas anteriormente.



1.7.2.2. Microsoft Office Visio

Se utilizará para realizar la modelación del negocio y es el escogido pues es un programa para la creación de diagramas que puede ayudar a crear diagramas empresariales y técnicos en los que se documenten y organicen ideas, procesos y sistemas complejos. Los diagramas creados en Visio permiten clara, concisa y eficazmente visualizar y comunicar información, de unas formas no posibles utilizando exclusivamente texto y números. Visio también automatiza la visualización de los datos al sincronizarse directamente con los orígenes de datos para proporcionar diagramas actualizados, y se puede personalizar para cubrir las necesidades de la organización.

1.7.2.3. IDEF0

IDEF0 es una técnica de modelación concebida para representar de manera estructurada y jerárquica las actividades que conforman un sistema o empresa, y los objetos o datos que soportan la interacción de esas actividades y en este proyecto será la utilizada para modelar los procesos del negocio.

Un modelo IDEF0 se compone de una serie jerárquica de diagramas que permiten mediante niveles de detalle, describir las funciones especificadas en el nivel superior. En las vistas superiores del modelo la interacción entre las actividades representadas permite visualizar los procesos fundamentales que sustentan la organización.

1.7.3. Metodologías Utilizadas.

1.7.3.1. RUP

Este proyecto utiliza la **metodología RUP**, que es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye una de las metodologías estándares más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos de proyectos de largo plazo como el caso del presente sistema. La misma comprende tres frases claves. Dirigido por los casos de uso (orientan el proyecto a la importancia del usuario y a lo que este quiere), centrado en la arquitectura (relaciona la toma de decisiones que indican como tiene que ser construido el sistema y en que orden), iterativo e incremental (divide el proyecto en mini proyectos donde los casos de usos y la arquitectura cumplen sus objetivos de manera más depurada).

El Proceso Unificado es un proceso porque “define quién está haciendo qué, cuándo lo hacer y cómo alcanzar cierto objetivo, en este caso el desarrollo de software”. Los conceptos clave del Proceso Unificado son:

Fase e iteraciones	¿Cuándo se hace?
Flujos de trabajo de procesos (actividades y pasos)	¿Qué se está haciendo?
Artefactos (modelos, reportes, documentos)	¿Qué se produjo?
Trabajador (un arquitecto)	¿Quién lo hace?

Esta metodología ofrece una reducción de riesgos basado en la retroalimentación temprana, pruebas continuas e iterativas que promueven una mejor evaluación del estado del proyecto, además permite a los clientes recibir evidencia concreta del avance del mismo. Y posibilita que los problemas más complejos se ataquen primero. Esta metodología genera una gran documentación, ideal para este proyecto que presenta la característica de ser inestable en cuanto al equipo de trabajo.

1.7.3.2. Lenguaje de Modelado

UML, (por sus siglas en inglés, *Unified Modeling Language*) es el lenguaje de modelado de sistemas de software; está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo 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.

Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como RUP), pero no especifica en sí mismo qué metodología o proceso usar.

1.7.4. Lenguaje(s) de programación.

Según la plataforma LAPP (Linux, Apache, PostgreSQL, PHP) de trabajo y las facilidades de uso se optó por utilizar los siguientes lenguajes de programación en dependencia donde se apliquen:

1.7.4.1. PHP5

PHP es un lenguaje orientado específicamente al desarrollo de aplicaciones web como la que se pretende obtener, ampliamente difundido debido a su facilidad de aprendizaje, el soporte de una importante comunidad de programadores, la abundante documentación y su libertad de uso. En este lenguaje encontramos una multitud de frameworks como Symfony (el utilizado), Cake PHP, CodeIgniter, etc, que no son más que estructuras de software que ayudan a la construcción de aplicaciones, de la cual la mayoría están basados en el patrón Modelo Vista Controlador.

Como ventajas de PHP se puede citar:

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, como: InterBase, MySQL, Oracle, Informix, PostgreSQL, entre otras.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- 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.
- Permite las técnicas de Programación Orientada a Objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables.
- Tiene manejo de excepciones.

PHP es utilizado en aplicaciones Web por algunas de las organizaciones más prominentes tales como Mitsubishi, Redhat, Der Spiegel, MP3-Lycos, Ericsson y NASA. Como ejemplo de aplicaciones hechas en PHP tenemos: Drupal, Mambo Open Source, Joomla, entre otras.

1.7.4.2. JavaScript.

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas, como las que se crearán en el sistema, para mantener una interacción entre el usuario y la aplicación.

JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. Legalmente, JavaScript es una marca registrada de la empresa Sun Microsystems.

1.7.5. Entorno(s) de Desarrollo Integrado.

La definición que da el proyecto Eclipse acerca de su software es: "una especie de herramienta universal - un IDE abierto y extensible para todo y nada en particular".

Eclipse es una plataforma de software de código abierto independiente y que es y ha sido muy utilizada para desarrollar entornos de desarrollo (IDE), pero del mismo modo se puede usar para otros tipos de aplicaciones cliente.

Lo mejor de Eclipse es que tiene una gran comunidad de usuarios extendiendo constantemente las aplicaciones.

Para este proyecto se utilizará el **Eclipse** con el plug-in⁸ **PDT** (plug-in para programar en PHP con el IDE Eclipse). Eclipse PDT es un conjunto de herramientas y frameworks que incrementan el rendimiento de los desarrolladores usando PHP.

Las características más acentuadas en PDT 1.0 incluyen:

- El coloreado de sintaxis.
- El autocompletado de código.
- El explorador de archivos.
- La depuración remota de aplicaciones.
- Integración con Xdebug y una documentación bastante completa.

⁸ Unidad de código que brinda funcionalidad específica para IDE.

1.7.6. Gestor de base de datos

PostgreSQL está considerado como la base de datos de código abierto más avanzada del mundo. PostgreSQL proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales tales como DB2 u Oracle. La siguiente es una breve lista de algunas de esas características, a partir de PostgreSQL 7.1.x.

- **DBMS Objeto-Relacional**

PostgreSQL aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son consultas SQL declarativas, control de concurrencia multi-versión, soporte multi-usuario, transactions, optimización de consultas, herencia, y arrays.

- **Altamente Extensible**

PostgreSQL soporta operadores, funcionales métodos de acceso y tipos de datos definidos por el usuario.

- **Integridad Referencial**

PostgreSQL soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.

- **Lenguajes Procedurales**

PostgreSQL tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL. Este lenguaje es comparable al lenguaje procedural de Oracle, PL/SQL. Otra ventaja de PostgreSQL es su habilidad para usar Perl, Python, o TCL como lenguaje procedural embebido.

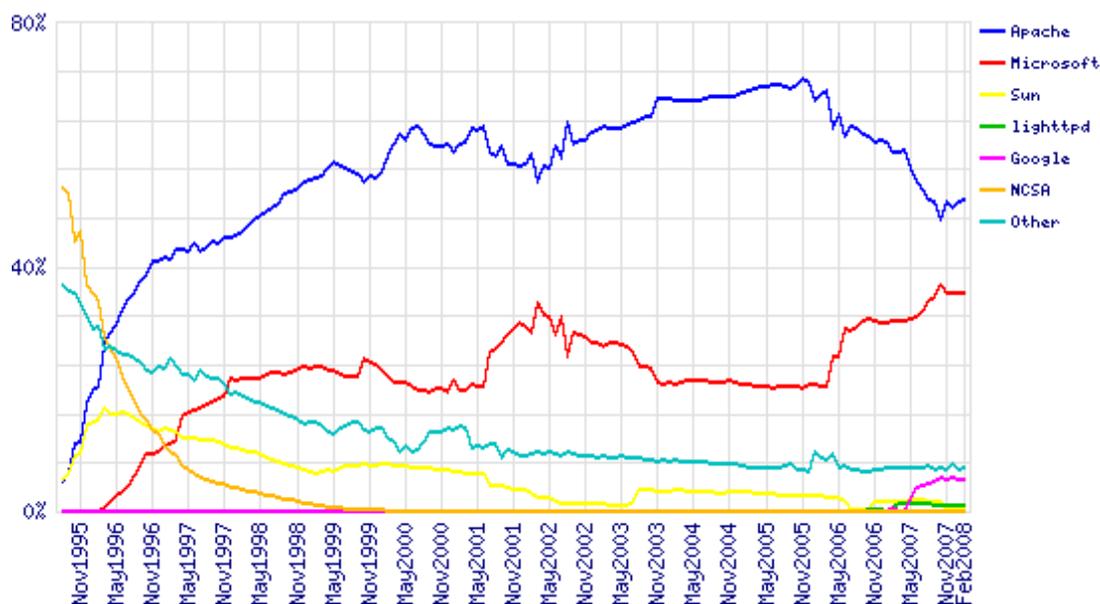
- **Alta concurrencia**

Mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.

1.7.7. Servidor de aplicaciones web

Apache es el servidor web por excelencia, su configurabilidad, robustez y estabilidad hacen que se manipule en este proyecto. Este tiene varias características las cuales lo hacen muy interesante:

- Corre en una multitud de Sistemas Operativos (Linux, MacOS, Windows), lo que lo hace prácticamente universal.
- Es una tecnología gratuita de código fuente abierto.
- Apache es un servidor altamente configurable de diseño modular.
- Apache te permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto.



1.7.8. Framework Utilizado

Symfony está diseñado para optimizar, el desarrollo de las aplicaciones web. Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Automatiza las

tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación.

Entre las características generales del framework podemos citar:

- Fácil de instalar y configurar: ha sido probado con éxito en plataformas Windows y derivadas de Unix.
- Independiente del manejador de base de datos: utiliza Propel, una capa de abstracción que le permite interactuar con varias bases de datos.
- Simple de usar: y al mismo tiempo lo suficientemente flexible para adaptarse a escenarios complejos.
- Cumple con la mayoría de las mejores prácticas en diseño web y patrones de diseño.
- Utilizable en entornos empresariales: puede adaptarse a políticas y arquitecturas ya existentes en tecnologías de información, y es lo suficientemente estable para proyectos de largo plazo.
- Código legible: con comentarios en phpDocumentor para su fácil mantenimiento.
- Fácil de extender: permitiendo la integración con otras librerías.

Adicionalmente, pueden señalarse otro grupo de características que permiten la automatización de ciertas tareas relacionadas con la construcción de aplicaciones web, como las siguientes:

- Uso de plantillas: las cuales pueden ser elaboradas por diseñadores de páginas web que desconocen el resto de los detalles técnicos del framework.
- Validación y regeneración automática de formularios: lo que asegura una buena calidad de los datos en la base de datos y una mejor experiencia de usuario.
- Verificación de la salida enviada por la aplicación: que ofrece una protección frente a ataques por datos corruptos.
- URLs inteligentes: que permiten que las direcciones de las páginas web sean parte de la interfaz y resulten amigables a los motores de búsqueda.

- Las interacciones usando AJAX⁹ son fáciles de implementar gracias a los asistentes de una sola línea de código que encapsulan efectos en JavaScript compatibles con varios navegadores (helpers).

Ha sido probado en numerosos proyectos reales tales como el sitio europeo “**The European Plastics Recyclers**” (<http://www.plasticsrecyclers.eu>) el cual se dedica a promover la utilización de plásticos reciclados y a ayudar a las empresas a iniciar proyectos de reciclaje, otro ejemplo, es el sitio **Zangoa** (<http://www.zangoa.com/es>) que nos es más que un portal que permite encontrar alojamientos de turismo rural a partir de una zona geográfica, así como ver información adicional de su entorno y por último queremos citar el sitio **Barcelona Centre Mèdic** (<http://www.bcm.es/es>) que es único en su tipo en Europa, ya que brinda un servicio de atención médica integral de máxima calidad ofrecido por los 20 centros asistenciales de Barcelona más acreditados y de mayor prestigio internacional.

CONCLUSIONES:

El Capítulo 1 contiene en sus inicios un estudio de los principales aspectos que tienen relación con el objeto de estudio planteado en este trabajo, así como la descripción de los procesos de Planificación y Control de las Auditorías que se desarrollan en el MAC.

En la segunda parte se realizó el análisis de las tecnologías y herramientas a utilizar para el desarrollo del software, exponiendo algunas de las características y ventajas de las mismas. Para el desarrollo de la aplicación Web se tuvo en cuenta el uso de herramientas multiplataforma, además de licencias de utilización libres. Se concilió la metodología RUP para el desarrollo del proyecto por ser un proceso iterativo e incremental, que permite dirigir el proceso por casos de uso y tener una mejor visión de la calidad en cada etapa del proceso, como lenguaje de modelado UML y la herramienta que se usará como entorno de creación de diagramas para UML es el Visual Paradigm. El lenguaje de programación que se pondrá en práctica será PHP, ya que el mismo es un lenguaje orientado específicamente al desarrollo de aplicaciones web. Además de establecer como gestor de base de

⁹ Acrónimo de *Asynchronous JavaScript And XML* (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas. Es una combinación de cuatro tecnologías ya existentes: HTML o XHTML, hojas de estilo (Cascading Style Sheets o css), el DOM (Document Object Model), XML, y el objeto XMLHttpRequest.

datos PostgreSQL, una base de datos de código abierto muy avanzada. Como framework para optimizar las aplicaciones web el Symfony y el servidor de aplicaciones web Apache.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.1. Introducción:

Para desarrollar un sistema con la calidad requerida se debe tener dominio de todos los procesos involucrados en el mismo, además de especificar bien sus características y así desarrollar un software de forma eficiente.

En el presente capítulo se mostrará de forma detallada los procesos relacionados con el negocio. Se explicará cada proceso que se encuentra dentro del objeto de automatización, así como una panorámica de la propuesta del sistema que se quiere obtener. El modelo del Negocio además de ser descrito, se podrá analizar mediante varios diagramas que visualizan el funcionamiento de cada proceso, permitiendo una mejor comprensión del mismo. La captura de los requisitos del software será otro tema a tratar y se especificarán los requerimientos funcionales. Para entrar en detalle sobre el sistema se definirán los actores del sistema, con sus descripciones; se visualizará el diagrama de casos de uso, así como la definición de los casos de uso del sistema.

2.2. Objeto de estudio:

El Ministerio de Auditoría y Control está formado por diferentes direcciones las cuales se encargan de procesos específicos. El objeto de estudio en el que nos vamos a centrar son los procesos de planificación de las acciones de control de la Dirección de Planificación, Análisis y Control del MAC.

2.3. Problema y situación problemática:

El Ministerio de Auditoría y Control está formado por direcciones entre las que se encuentran: Dirección de Atención al Sistema Nacional de Auditoría, Dirección de Auditorías y Controles Especiales, Dirección de Atención a la Ciudadanía y Dirección de Planificación Análisis y Control, entre otras.

La Dirección de Planificación, Análisis y Control es la encargada de realizar la **planificación anual de todas las acciones de control**. Además esta dirección pone en marcha una serie de procedimientos vinculados a ella como: **cumplimiento trimestral del Plan; Registro de Auditores de la República de Cuba; Informe de la situación de los presuntos hechos delictivos**, elaboración de **parte quincenal**, **elaboración de Informe de las empresas en Perfeccionamiento Empresarial**, además

de llevar un sistema de indicadores para la evaluación y calificación del funcionamiento de las Delegaciones del Ministerio de Auditoría y Control, entre otras acciones.

Según investigaciones hechas, el MAC ha transitado por dos formas de trabajo para el desarrollo de los procesos de sus direcciones, primero se utilizaba un software (Raudit) el cual era una aplicación desktop, que permitía entre otras cosas, llevar un control de las auditorías que se realizaban. La primera versión fue elaborada en el año 1999, y presentó los siguientes problemas:

- Interfaz poco amigable.
- Complejidad de manipulación (muchos errores si no se tenía pleno dominio para trabajar con esta).
- Las salvadas se hacían en discos 3 y media. (No existía base de datos).
- Al no existir base de datos, se mantenían los grandes cúmulos de papeles, pues los informes solo se podían ver si eran impresos.

La segunda forma de trabajo, actualmente vigente, es guardando toda la información que se genera de la realización de los Planes y Partes Operativos, en hojas de Excel. Este proceso de almacenamiento es más organizado, ya que permite búsquedas a través de filtros, pero igual no cuenta con las facilidades que un sistema completo puede proporcionar, esta forma de trabajo retrasa la búsqueda de información, así como el almacenamiento de la misma e imposibilita el desarrollo del trabajo con la rapidez y calidad requerida.

2.4. Objeto de automatización.

Los procesos que serán objeto de automatización y que corresponden al módulo en que se trabaja son:

- Plan Nacional de Auditoría.
- Parte Operativo.
- Cumplimiento del plan trimestral.

2.4.1. Plan Nacional de Auditoría:

Para la realización del Plan Nacional de Auditoría las Delegaciones y Direcciones del MAC realizan la presentación de las propuestas del plan de Acciones de Control según el Calendario de Conciliación. En el proceso de presentación de la propuesta las entidades mencionadas anteriormente concilian y analizan con la dirección de Planificación, Análisis y Control, dichas propuestas existentes como Plan de Acciones de Control. Posteriormente el plan conciliado es presentado a los miembros del Directorio y a los Delegados del MAC para su aprobación.

2.4.2. Parte operativo:

Para realizar el Parte Operativo las Direcciones y Delegaciones del MAC que realizan acciones de control confeccionan quincenalmente un informe sobre el estado del cumplimiento del plan, utilizando el Modelo 001(modelo para mostrar el Plan Anual). Luego la Dirección de Planificación, Análisis y Control hace recepción de la información que brindan las direcciones y delegaciones y comprueba que las fechas de inicio y terminación de las acciones de control se encuentren en el tiempo establecido en el Plan Anual, reclamando a los Delegados en caso de incumplimiento o de incongruencia en algún dato, para cerrar el parte.

2.4.3. Cumplimiento del Plan Trimestral:

Para comprobar el cumplimiento del Plan Trimestral las Direcciones y Delegaciones del MAC brindan un reporte con las acciones de control hechas en un período de tiempo (3 meses) y su calificación, así como las que están en proceso. Con esta información se chequea que las acciones de control ejecutadas en el trimestre coincidan con las que están planificadas, de no ser así se contacta con las Direcciones y Delegaciones pidiendo que soliciten formalmente la modificación del plan. Posterior a esto se elabora un resumen por territorios y organismos y tipo de empresa en el que se plasma la situación actual de las acciones de control, el cual es entregado a la Directora de DPAC¹⁰.

2.5. Información que se maneja.

¹⁰ Dirección de Planificación Análisis y Control.

Modelo 001: Este modelo indica el formato y los datos que se toman en cuenta para llevar a cabo la Planificación de las Auditorías, el Control del Plan Trimestral y el Parte Operativo. Datos tales como: la entidad a la cual se le realiza una acción de control, el código REUP asociada a la misma, la fecha de inicio y terminación planificada para la ejecución y otros vinculados al cierre de la acción de control.

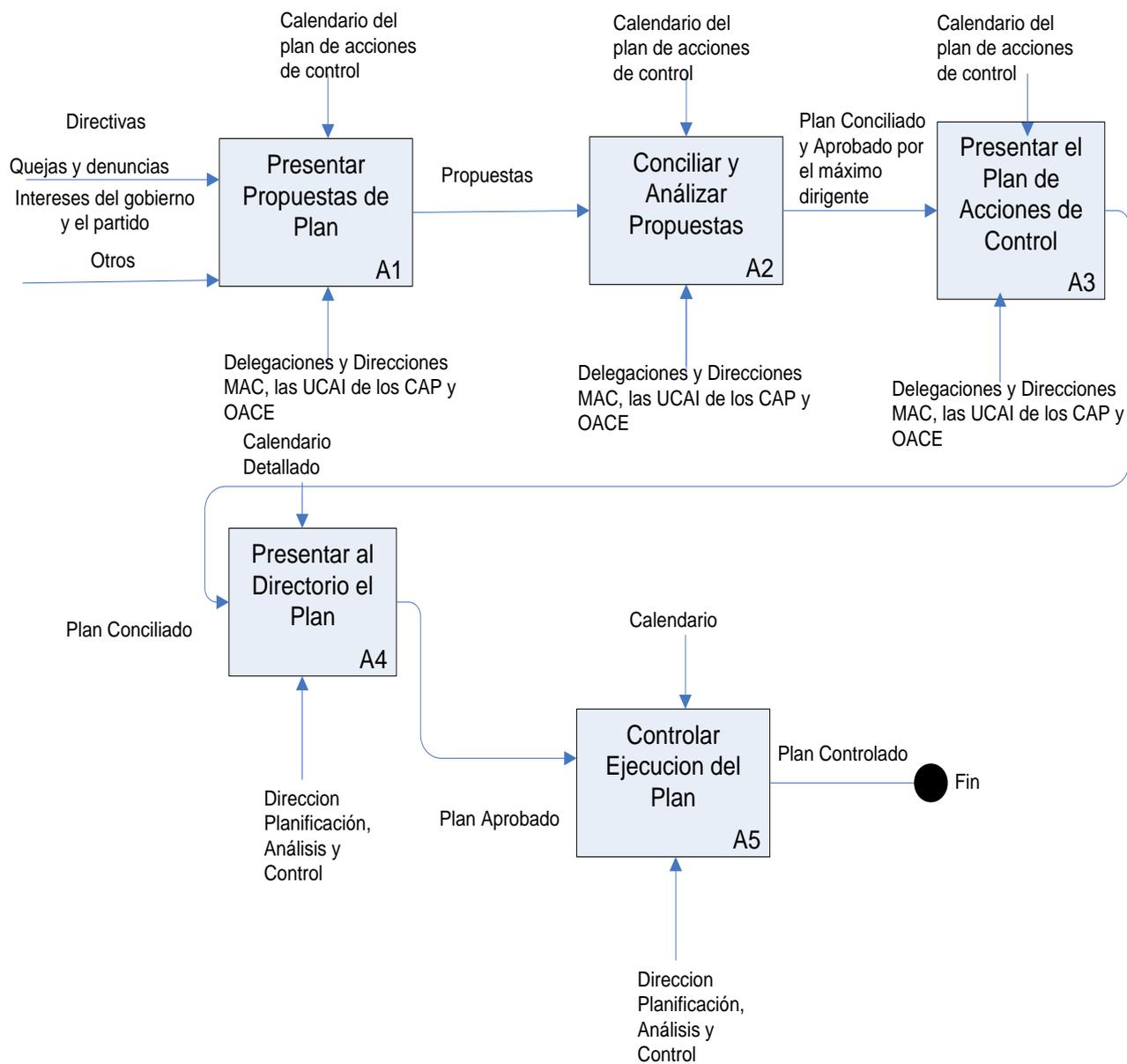
2.6. Propuesta de sistema.

Se ha decidido llevar a cabo el desarrollo de una aplicación web que brinde una solución real al problema existente con la planificación anual nacional de las acciones de control, así como el control de las mismas. La misma ofrecerá la posibilidad de realizar la Planificación de las acciones de control, así como mostrar las ya existentes, además de hacer modificaciones de la información con que se trabaja (insertar, modifica, eliminar acciones de control), dará la opción de realizar varias búsquedas por criterios y mostrar todas las acciones de control a los usuarios autorizados. El sistema será confeccionado de manera tal que se le permita al usuario sólo las opciones autorizadas a las que debe acceder y realizar en las interfaces las modificaciones requeridas según las responsabilidades que los mismos tengan.

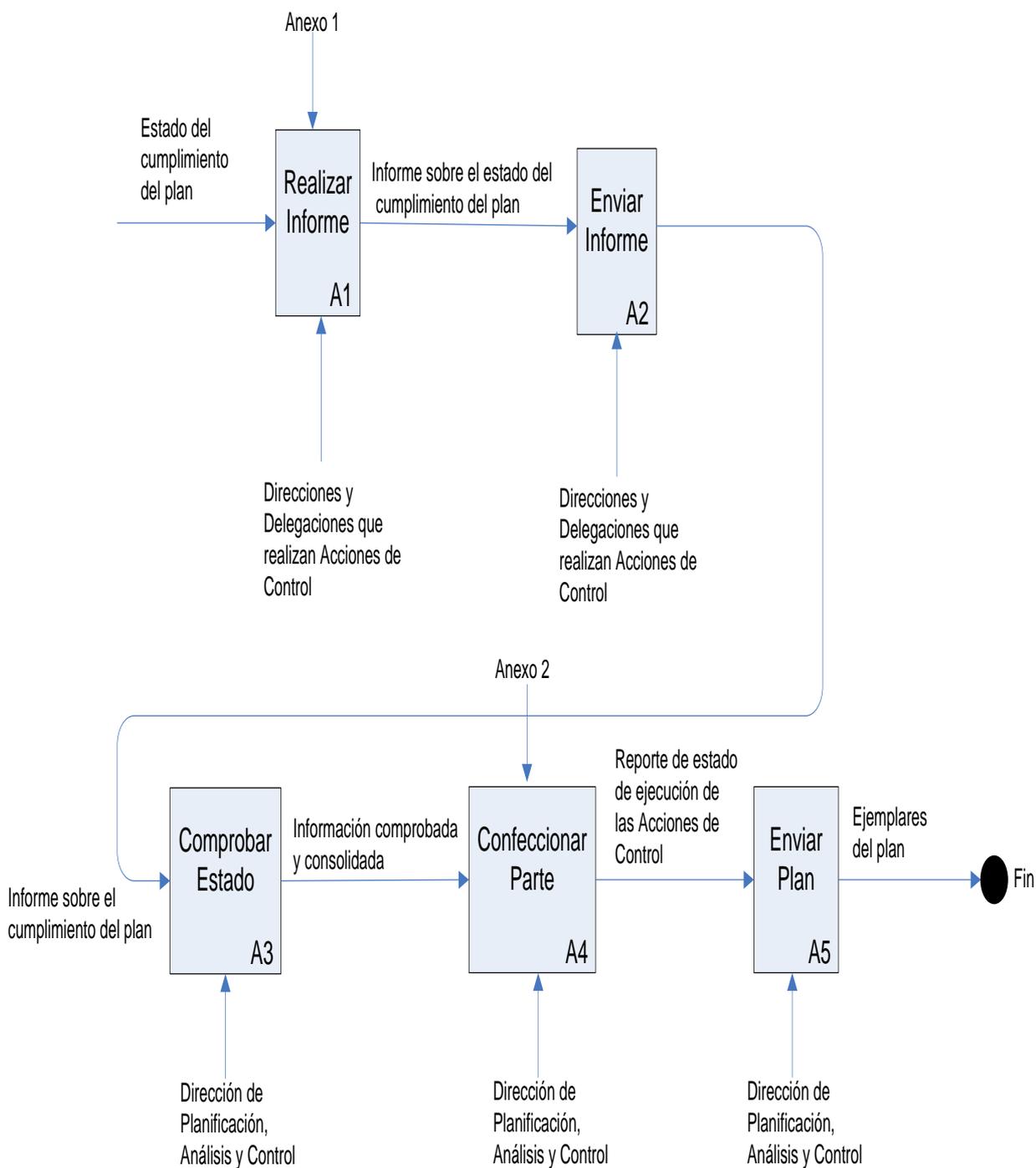
2.7. Modelo de negocio.

A continuación se brindarán diagramas referentes al funcionamiento detallado de los procesos vinculados al negocio, especificando en cada uno de ellos el flujo de actividades que se realizan y la obtención de información que se adquieren de ellas.

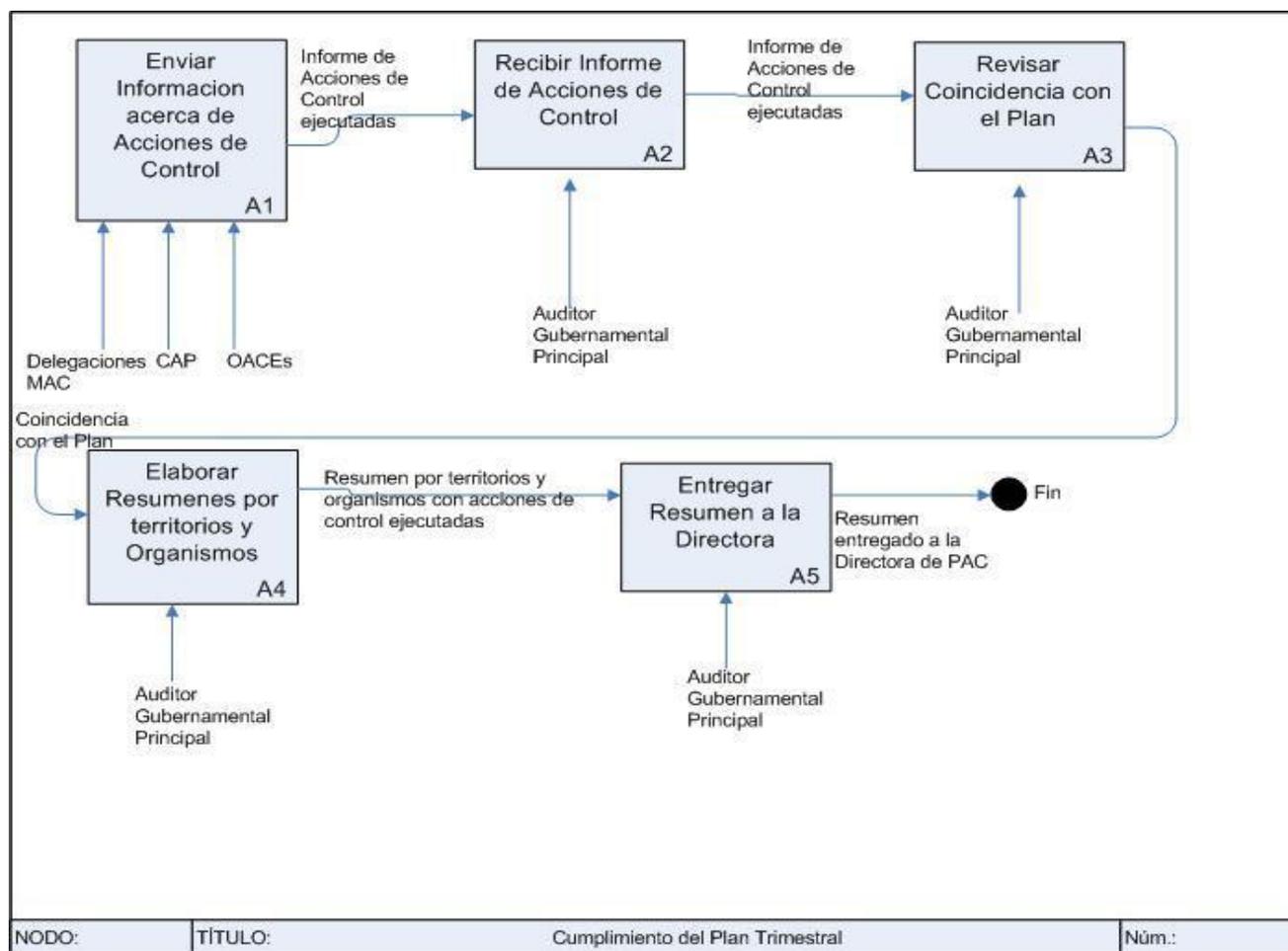
2.7.1. Plan Nacional de Auditorías



2.7.2. Elaboración Parte Operativo



2.7.3. Cumplimiento del Plan Trimestral



2.8. Especificación de los requisitos de software.

La especificación de requisitos es una descripción completa del comportamiento del sistema que se va a desarrollar. Incluye un conjunto de casos de uso que describe todas las interacciones que tendrán los usuarios con el software. Los casos de uso también son conocidos como requisitos funcionales.

2.8.1. Requerimientos Funcionales

RF1 Mostrar acciones de control

- RF2 Insertar nueva acción de control al Plan.
- RF3 Modificar una acción de control del plan.
- RF4 Eliminar una acción de control seleccionada del Plan.
- RF6 Controlar el Plan Anual.
- RF7 Mostrar las acciones de control a conciliar.
- RF8 Mostrar Empresas por criterios.
- RF9 Buscar Acciones de Control hechas a una empresa.
- RF10 Mostrar Modelos de la Orden de Trabajo.

2.8.2. Requisitos No Funcionales:

Especifican las propiedades del sistema que tienen que ver con rendimiento, velocidad, uso de memoria, plataforma, fiabilidad, tiempo de respuesta media, etc. Imponen condiciones a requisitos funcionales.

Una buena definición de requisito no funcional es la dada por Thayer¹¹: “es un requisito software que describe no lo que el software hará, sino como lo hará, como por ejemplo, requisitos de rendimiento”. Los requisitos no funcionales son difíciles de verificar o testear, y por ello son evaluados subjetivamente.

En el presente trabajo se mencionarán los requisitos no funcionales que se pondrán en práctica:

Apariencia o Interfaz Externa:

- La interfaz a implementar debe ser sencilla para disminuir el tiempo de capacitación de los usuarios finales (principalmente aquellas personas que no son expertas en la rama de la informática).
- Por el uso diario y constante que tendrá el software, la interfaz debe ser agradable, que favorezca el estado de ánimo del cliente y que combine correctamente los colores, tipo de letra y tamaño y que los iconos estén en correspondencia con lo que representan.
- La interfaz debe ser intuitiva al usuario, además de contar con una ayuda online.

¹¹ Richard H. Thayer, destacado investigador de la rama de la Ingeniería de Software, autor de los siguientes libros: Software Requirements Engineering, 2nd Edition, The Project Manager's Guide to Software Engineering's Best Practices, Software Engineering, Vol. 2: The Supporting Processes, entre otros.

- Deben utilizarse plantillas con un mismo estilo.

Usabilidad:

- El sistema debe ser de fácil manejo para los usuarios que tengan niveles básicos sobre la computación o hallan trabajado con la Web.
- Debe tener una opción de ayuda sobre las principales operaciones que se realizan y sus iconos respectivos para mejor entendimiento.
- El sistema informatizará lo más fiel posible los procesos que se realizan en el Ministerio para lograr el menor tiempo en cuanto a la comprensión del sistema y del proceso.

Rendimiento:

- La aplicación debe estar concebida para el consumo mínimo de recursos.
- Un total de 350-400 usuarios conectados de forma simultánea al servidor central en cualquier momento de tiempo dado.
- Debe completar las transacciones en un tiempo de 45-60 segundos.
- Los clientes no necesitarán más de 128MB de RAM, lo suficiente para ejecutar un navegador Web.

Portabilidad, Escalabilidad, Reusabilidad:

- El sistema será multiplataforma.
- La aplicación se construirá utilizando patrones (de diseño como los GRAPS y GOF) y estándares internacionales de implementación, documentación y diseño, para facilitar su integración futura, con componentes desarrollados por cualquiera de las partes y garantizar posibilidades de mantenimiento ágil y seguro.
- El sistema deberá poder ser accedido desde cualquier Sistema Operativo.
- El sistema debe hacer un uso racional de los recursos de hardware, sobre todo en estaciones de trabajo.
- Debido a los cambios en las condiciones económicas del país, las empresas cubanas toman decisiones continuas que cambian las condiciones en que se desarrollan los procesos, por lo

que el sistema deberá implementar la forma de adaptarse ante el cambio de dichas condiciones.

Software:

- El sistema se desarrollará con tecnología PHP versión 5.2.5 o superior.
- Se utilizará un servidor con sistema operativo Ubuntu 7.10 o superior.
- Se utilizará tecnología Apache versión 2.0.50 o superior para el servidor Web.
- El sistema utilizará una base datos implementada en PostgreSQL versión 8.2.1 o superior.
- En las computadoras de los clientes se garantizará versiones de Windows 2000 o superior, así como Linux y sus correspondientes distribuciones.
- En las computadoras de los clientes solo se requiere de un navegador (Internet Explorer versión 4.5 o superior, Mozilla Firefox versión 2.0.0.1 o superior).

Hardware.

Para las estaciones de trabajo:

- Se requiere tengan tarjeta de red.
- Se requiere tengan al menos 128 MB de memoria RAM.
- Se requiere al menos 100MB de disco duro.
- Procesador 800 MHz como mínimo.

Para los servidores:

- Se requiere tarjeta de red.
- Se requiere tenga la menos 512MB de RAM.
- Se requiere al menos 40GB de disco duro.
- Procesador 2.0 GHz como mínimo.

Dispositivo: Impresora

- Velocidad de impresión: Calidad casi carta 77 cps (10 cpp).

Requerimientos Políticos y Culturales.

- Emplear en la aplicación idioma español.
- Contar con logotipos e imágenes que se encuentren en correspondencia con el carácter científico y profesional del tema.
- Algún cambio que se quiera realizar debe ser previamente consultado con la Dirección de Control Gubernamental del Ministerio de Auditoría y Control y canalizado por la UCI con la dirección de Producción central y de la Facultad a la que pertenece el proyecto.

Legales.

- El sistema se basa en el manual de normas y principios establecidos por el MAC.
- La mayoría de las herramientas de desarrollo son libres y del resto, las licencias están avaladas.
- El sistema tendrá en cuenta lo establecido por el ‘Reglamento de las funciones de las Direcciones y Delegaciones del Ministerio de Auditoría y Control’, en todo lo referido al desarrollo del sistema.

Seguridad.

- El sistema debe poder comunicarse usando un protocolo seguro (https).
- Los datos que no pueden viajar de forma transparente por la red, deben ser encriptados.
- Chequear si el usuario que está accediendo al sistema está autenticado y brindarle servicio de autenticación.
- Mantener la integridad de la información, es decir que no se pierda durante su almacenamiento o transporte.
- Permitir que cuando se borre cualquier documento o información pueda existir una opción de advertencia antes realizar la acción.
- Realizar auditoría a los principales eventos dentro del sistema, registrando al usuario, el tipo de usuario y los eventos efectuados.
- Se garantizará un fuerte tratamiento de excepciones.
- Un porcentaje de la seguridad corre por parte del lenguaje y Framework propuesto (PHP y Symfony respectivamente) y otra parte por parte de los servidores (Apache, LDAP, PostgreSQL).
- Se implementará un mecanismo de acceso a la base de datos, que está dado por la diferenciación de las acciones que el sistema realiza en cada momento. Es decir, un usuario

para lectura-escritura cuando se requiera modificar y acceder a los datos y otro con los privilegios administrativos, para la realización de copias de seguridad y otras acciones administrativas.

- La encriptación de contraseñas.
- La asignación de usuarios y sus funcionalidades sobre el sistema se definirán desde el modulo de Administración.
- La seguridad se tratará desde las primeras fases de desarrollo del sistema.
- Se utilizará reglas o principios de la programación segura (diseño simple y abierto, separación de privilegios, control de acceso apropiado, validación de datos de entrada y salida, tratamiento de errores, utilización de criptografía, reutilización de código, control del flujo de datos, control de sobrecarga del búfer, control de inyección de código).
- Programación de disparadores, funciones o procedimientos almacenos y reglas en la Base de Datos para no permitir la manipulación de los datos directamente en el SGBD.
- Debe quedar constancia de quién, desde donde, y cuando se realizó una operación determinada en el sistema.

Confiabilidad.

- La información manejada por el sistema está protegida de acceso no autorizado y divulgación.

Integridad.

- La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción de los datos y accesos indebidos.

Fiabilidad.

- Debe garantizarse el resguardo de la información (imágenes, documentos), así como la grabación periódica (backups) de la Base de Datos, de forma tal que se posibilite la reinstalación del sistema y los datos, en caso de fallos en el sistema o en el hardware.

Legales.

- El sistema se basa en el manual de normas y principios establecidos por el MAC.

- La mayoría de las herramientas de desarrollo son libres y del resto, las licencias están avaladas.
- El sistema tendrá en cuenta lo establecido por el ‘Reglamento de las funciones de las Direcciones y Delegaciones del Ministerio de Auditoría y Control’, en todo lo referido al desarrollo del sistema.

2.9. Definición de los Casos de Uso.

2.9.1. Actores del Sistema:

Un actor caracteriza las interacciones que los usuarios exteriores pueden tener con el sistema. Cada actor participa en uno o más casos de uso. Interactúa con el caso de uso (y por lo tanto con el sistema o la clase que posee el caso de uso), intercambiando mensajes.

Se brindará una breve reseña de las actividades que realizan los actores del sistema.

Actor	Descripción
Directora de DPAC	Es la encargada de organizar, controlar, monitorizar y exigir por los procesos que se hacen en la dirección de Planificación, Análisis y Control.
Auditor gubernamental asistente	Es el encargado de los procesos de parte operativo, PHD, cumplimiento del plan trimestral.
Auditor gubernamental principal	Es el encargado de monitorizar algunas actividades de la dirección como Elaboración del plan, las directivas, así como el cumplimiento del mismo, incluido ahí mismo los presuntos hechos delictivos.

2.9.2. Listado de Casos de Uso

A continuación los casos de usos más significativos se mostraran con sus actores, descripción breve y al requisito funcional al que hacen referencia.

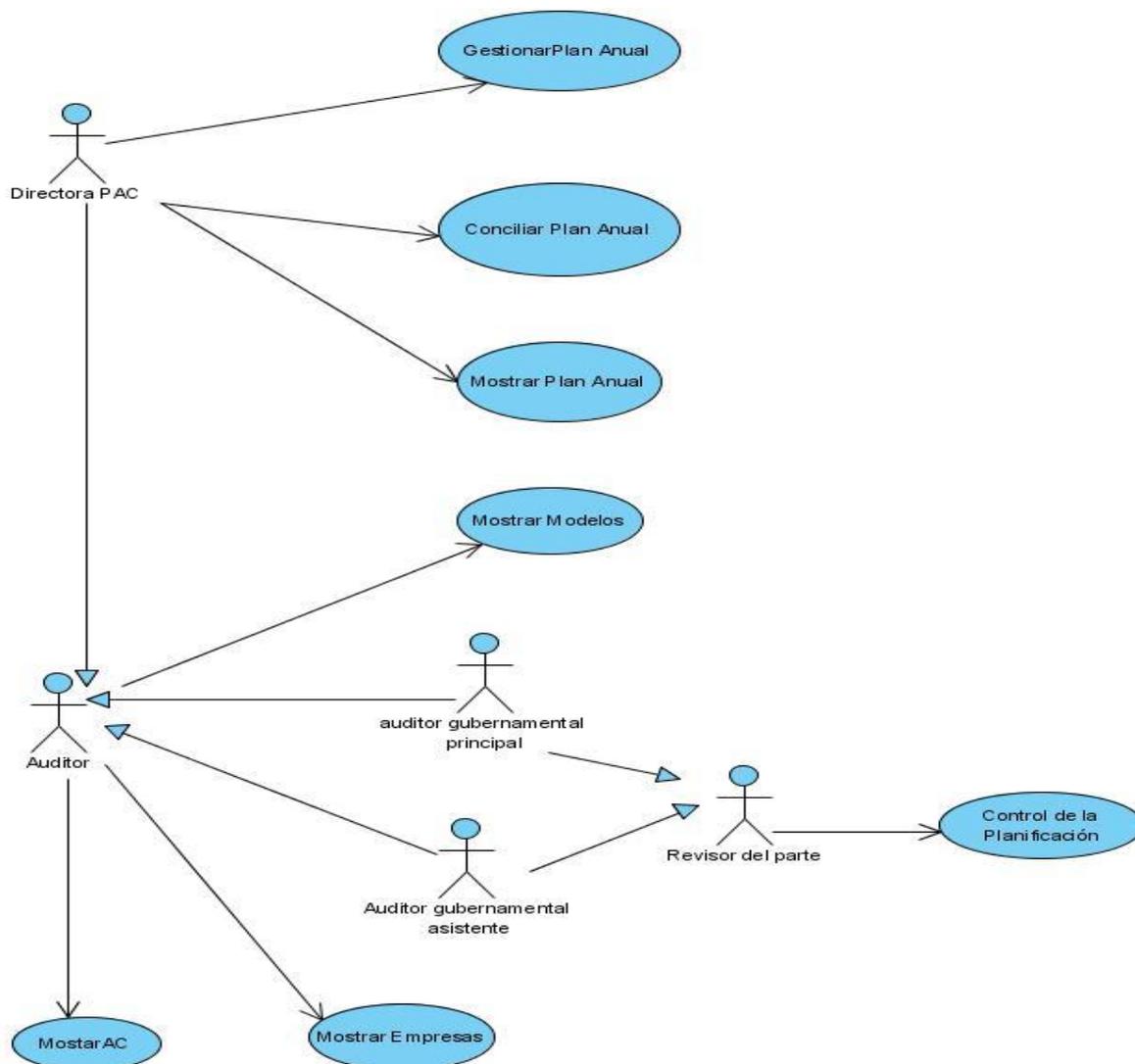
CU1	Gestionar Plan Anual
Actor	Directora de DPAC
Descripción	Mediante la realización de es caso de uso introducen, elimina y modifican acciones de control al plan anual, además de mostrar acciones de control utilizando criterios de búsqueda.
Referencia	RF1, RF2, RF3, RF4

CU2	Conciliar Plan Anual.
Actor	Directora de DPAC
Descripción	Mediante la realización de es caso de uso introducen, elimina y modifican acciones de control planificadas al plan anual, además de mostrar acciones de control utilizando criterios de búsqueda.
Referencia	RF1, RF2, RF3, RF7

CU3	Control de la Planificación.
Actor	Revisor del Parte.
Descripción	Mediante la realización de este caso de uso se lleva a cabo el control en un intervalo de tiempo deseado de las acciones de control que se planifican y se muestran por criterios de búsquedas.
Referencia	RF6

2.10. Diagramas de Casos de Uso

Los diagramas de casos de uso sirven para especificar la funcionalidad y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas.



2.10.1. Definición de los Casos de Uso:

La descripción se enfoca en el valor suministrado por el sistema a entidades externas tales como usuarios humanos u otros sistemas. A continuación los casos de uso críticos del sistema serán descritos de forma detalla.

2.10.1.1. Gestionar Plan de Acciones de Control:

Caso de Uso:	Gestionar Plan Anual.
Actores:	Directora de DPAC (inicia).
Resumen:	El caso de uso se inicia cuando la Directora de DPAC selecciona la opción

	Gestionar Plan Anual y esto le da paso a insertar, modificar o eliminar las acciones de control contenidas en el plan.
Precondiciones:	Acciones de Control registradas en el sistema.
Referencias	RF1, RF2, RF3, RF4
Prioridad	Crítico.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. La Directora del DPAC selecciona en la interfaz “nueva_ac” la opción Gestionar Plan Anual.	2. El sistema muestra los siguientes campos: 2.1. Direcciones: <ul style="list-style-type: none"> • DCG. • DAG. • DACE. • DASNA. 2.2. Delegaciones.
	3. El sistema muestra las siguientes opciones: <ul style="list-style-type: none"> • Adicionar ACC. (Ir a Sección ACC) • Modificar ACC. (Ir a Sección Modificar) • Eliminar ACC. (Ir a Sección Eliminar ACC)
4. La Directora de DPAC selecciona el criterio de búsqueda deseado.	5. El sistema muestra un listado con las acciones de control que respondan al criterio seleccionado.
Sección Adicionar ACC	
Acción del Actor	Respuesta del Sistema
6. La Directora de DPAC selecciona la opción Adicionar ACC.	7. El sistema muestra el formulario “insertar_ac” para introducir los siguientes datos:

	<ul style="list-style-type: none">• Ministerio.• Provincia.• Empresa.• Código REUP.• Tipo de Acción.<ul style="list-style-type: none">• Auditoría Gestión.• Auditoría Financiera.• Auditoría Fiscal.• Auditoría Especial.• Auditoría Recurrente.• Visita de Supervisión y Control.• Inspección Gubernamental.• Comprobaciones Especiales.• Particularidades.<ul style="list-style-type: none">• Perfeccionamiento Empresarial.• Presupuestada.• Inversión Extranjera.• En el Exterior.• Fecha de Inicio Planificada.• Fecha de Terminación Planificada.
--	--

	<ul style="list-style-type: none"> • Fecha de Inicio Real. • Fecha de Terminación Real. • Calificación. • PHD. • PHC. • Afectación Económica. <ul style="list-style-type: none"> • CUP. • CUC. • Procedencia.
8. La Directora del DPAC introduce los datos requeridos para insertar una nueva acción de control.	9. El sistema realiza la validación de los datos entrados.
	10. El sistema muestra un mensaje si la acción de control se insertó satisfactoriamente.

Insertar Acción de Control

Ministerio	<input type="text"/>	Particularidades	Seleccione ▾
Provincia	<input type="text"/>	Tiempo_Inicio_R	2008-06-16 ...
Empresa	<input type="text"/>	Tiempo_Terminación_Real	2008-06-16 ...
Cod.REEUP:	<input type="text"/>	Calificación	Seleccione ▾
Tipo de Acción Control	Seleccione ▾	Afectación Económica	Seleccione ▾
Tiempo_Inicio_P	2008-06-16 ...	Precedencia	<input type="text"/>
Tiempo_Terminación_P	2008-06-16 ...		
<input type="button" value="Insertar"/> <input type="button" value="Cancelar"/>			

Flujos Alternos

Flujos Alterno 9.a “Mensaje de Error”

Acción del Actor	Respuesta del Negocio
	9.a.1. El sistema muestra un mensaje indicando

	el error que se produce.
Prototipo de Interfaz	
Sección Modificar ACC	
Acción del Actor	Respuesta del Sistema
11. La Directora del DPAC selecciona la acción de control que desea modificar.	
12. La Directora del DPAC selecciona la opción Modificar ACC.	<p>13. El sistema muestra el formulario "Modificar ACC", con los siguientes datos con el valor inicial:</p> <ul style="list-style-type: none"> • Ministerio. • Provincia. • Empresa. • Código REUP. • Tipo de Acción. <ul style="list-style-type: none"> • Auditoría Gestión. • Auditoría Financiera. • Auditoría Fiscal. • Auditoría Especial. • Auditoría Recurrente. • Visita de Supervisión y Control. • Inspección Gubernamental. • Comprobaciones Especiales.

	<ul style="list-style-type: none"> • Particularidades. <ul style="list-style-type: none"> • Perfeccionamiento Empresarial. • Presupuestada. • Inversión Extranjera. • En el Exterior. • Fecha de Inicio Planificada. • Fecha de Terminación Planificada. • Fecha de Inicio Real. • Fecha de Terminación Real. • Calificación. • PHD. • PHC. • Afectación Económica. <ul style="list-style-type: none"> • CUP. • CUC. • Procedencia.
<p>14. La Directora del DPAC modifica los datos que desee.</p>	<p>15. El sistema realiza la validación de los datos entrados.</p>
	<p>16. El sistema actualiza la información referente a la acción de control modificada.</p>

Modificar Acción de Control

Ministerio	<input type="text" value="MINTUR"/>	Particularidades	<input type="text" value="Perfeccionami"/>
Provincia	<input type="text" value="Cienfuegos"/>	Tiempo_Inicio_R	<input type="text" value="2008-06-16"/> ...
Empresa	<input type="text" value="Hotel Jagua"/>	Tiempo_Terminación_Real	<input type="text" value="2008-06-16"/> ...
Cod.REEUP:	<input type="text" value="58OP25"/>	Calificación	<input type="text" value="PHD"/>
Tipo de Acción Control	<input type="text" value="Auditoría Gest"/>	Afectación Económica	<input type="text" value="CUP"/>
Tiempo_Inicio_P	<input type="text" value="2008-06-16"/> ...	Precedencia	<input type="text" value="Cienfuegos"/>
Tiempo_Terminación_P	<input type="text" value="2008-06-16"/> ...		

Flujos Alternos

Flujos Alternos 15.a “Mensaje de Error”

Acción del Actor	Respuesta del Negocio
	15.a.1. El sistema muestra un mensaje indicando el error que se produce.

Prototipo de Interfaz

Sección Eliminar ACC

Acción del Actor	Respuesta del Sistema
17. La Directora del DPAC selecciona la acción de control que desea eliminar.	
18. La Directora del DPAC selecciona la opción Eliminar ACC.	19. El sistema muestra el mensaje de comprobación
20. La Directora del DPAC selecciona la acción “Aceptar”	21. El sistema elimina la acción de control seleccionada.
	22. El sistema actualiza los datos de la página.



Flujo Alterno 20.a “Opción Cancelar”

Acción del Actor	Respuesta del Sistema
20.a.1. La Directora del DPAC selecciona la opción Cancelar.	20.a.2. El sistema re direcciona para la interfaz “nueva_ac”
Prototipo de Interfaz	
Poscondiciones	Plan Anual, gestionado.

2.10.1.2. Conciliar Plan Anual:

Caso de Uso:	Conciliar Plan Anual.
Actores:	Directora DPAC (inicia).
Resumen:	El caso de uso se inicia cuando el actor selecciona la opción Mostrar Propuesta del Plan, luego tiene la opción de seleccionar una de las direcciones de MAC o una delegación de una provincia determinada, mostrándose así la propuesta del Plan
Precondiciones:	Existencia de acciones de control planificadas.
Referencias	RF2, RF3, RF4, RF7

Prioridad	Crítico.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. La Directora de DPAC selecciona en la interfaz “conciliar_plan” la opción Conciliar Plan Anual.	2. El sistema muestra los siguientes campos de búsqueda: <ul style="list-style-type: none"> • Direcciones. <ul style="list-style-type: none"> • DAG. • DCG. • DASNA. • Delegaciones. 	
	3. El sistema muestra además las siguientes opciones: <p>3.1. Nueva (Ir a Sección “Adicionar ACP”).</p> <p>3.2. Modificar (Ir a Sección “Modificar ACP”).</p> <p>Eliminar (Ir a ¡Error! No se encuentra el origen de la referencia.).</p>	
4. La Directora de DPAC selecciona la dirección y delegación de las cuales desea ver las acciones de control planificadas.	5. El sistema muestra un listado con las acciones de control planificadas que respondan al criterio de búsqueda seleccionado.	
Sección “Adicionar ACP”		
Acción del Actor	Respuesta del Sistema	
6. La Directora DPAC selecciona la opción	7. El sistema muestra el formulario	

“Adicionar ACP”.

“insertar_acp” para introducir los siguientes datos:

- Ministerio.
- Provincia.
- Empresa.
- Código REUP.
- Tipo de Acción.
 - Auditoría Gestión.
 - Auditoría Financiera.
 - Auditoría Fiscal.
 - Auditoría Especial.
 - Auditoría Recurrente.
 - Visita de Supervisión y Control.
 - Inspección Gubernamental.
 - Comprobaciones Especiales.
- Particularidades.
 - Perfeccionamiento Empresarial.
 - Presupuestada.
 - Inversión Extranjera.
 - En el Exterior.
- Fecha de Inicio Planificada.

	<ul style="list-style-type: none"> • Fecha de Terminación Planificada. • Procedencia.
8. La Directora DPAC inserta los datos necesarios.	9. El sistema valida los datos entrados.
	10. El sistema muestra un mensaje se la acción de control fue insertada satisfactoriamente.
	11. El sistema actualiza la información de la página.

Insertar Acción de Control

Ministerio

Provincia

Empresa

Cod.REEUP:

Tipo de Acción Control

Tiempo_Inicio_P

Tiempo_Fin_P

Flujos alternos

Flujos alternos 8.a “Mensaje de Error”

	8a.1 El sistema muestra un mensaje según el error ocurrido.
--	---

Prototipo de Interfaz

Sección “Modificar ACP”

Acción del Actor	Respuesta del sistema
12. La Directora de DPAC selecciona la acción de control que desea modificar de la	

<p>interfaz “Conciliar AC”.</p>	
<p>13. La Directora de DPAC selecciona la opción “Modificar ACP”.</p>	<p>14. El sistema muestra el formulario “modificar_acp” con los siguientes datos activos:</p> <ul style="list-style-type: none"> • Tipo de Acción. <ul style="list-style-type: none"> • Auditoría Gestión. • Auditoría Financiera. • Auditoría Fiscal. • Auditoría Especial. • Auditoría Recurrente. • Visita de Supervisión y Control. • Inspección Gubernamental. • Comprobaciones Especiales. • Fecha de Inicio Planificada. • Fecha de Terminación Planificada.
<p>15. La Directora de DPAC modifica los datos que desee.</p>	<p>16. El sistema valida los datos nuevos.</p>
	<p>17. El sistema actualiza la información de la acción de control modificada.</p> <p>18. El sistema actualiza la información de la página.</p>

Modificar Acción de Control

Ministerio	<input type="text" value="MINSAP"/>
Provincia	<input type="text" value="Holguín"/>
Empresa	<input type="text" value="Policlínico Sur"/>
Cod.REEUP:	<input type="text" value="45UK45"/>
Tipo de Acción Control	<input type="text" value="Auditoría Gestí"/>
Tiempo_Inicio_P	<input type="text" value="2008-06-16"/> ...
Tiempo_Fin_P	<input type="text" value="2008-06-16"/> ...

Flujos Alternos

Flujos Alterno 16.a “Mensaje de Error”

Acción del Actor	Respuesta del Negocio
	16.a.1. El sistema muestra un mensaje indicando el error que se produce.

Prototipo de Interfaz

Sección Eliminar ACP

Acción del Actor	Acción del Actor
19.La Directora del DPAC selecciona la acción de control que desea eliminar.	
20.La Directora del DPAC selecciona la opción Eliminar ACC.	21.El sistema muestra el mensaje de comprobación
22.La Directora del DPAC selecciona la acción “Aceptar”	23.El sistema elimina la acción de control seleccionada.
	24.El sistema actualiza los datos del sistema.



Flujo Alterno 22.a “Opción Cancelar”

Acción del Actor	Acción del Actor
22.a.1. La Directora del DPAC selecciona la opción Cancelar.	22.a.1. La Directora del DPAC selecciona la opción Cancelar.
Prototipo de Interfaz	
Poscondiciones	Plan Anual, conciliado.

2.2.1.1. Control de la Planificación

Caso de Uso:	Control de la Planificación.
Actores:	Revisor del Parte.
Resumen:	El caso de uso inicia cuando el actor selecciona la opción Control de la Planificación, introduce el intervalo de tiempo a controlar (15 días o e meses) y los criterios de búsquedas por los que quiere que se le muestren las acciones de control.
Precondiciones:	Existencia de acciones de control registradas.
Referencias	RF6
Prioridad	Crítico

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El revisor del parte selecciona en la interfaz “control_plan” la opción “Control de la Planificación”.	2. El sistema muestra el formulario “Período de tiempo” con los siguientes datos: <ul style="list-style-type: none"> • Fecha de Inicio. • Fecha de Fin.
3. El revisor del parte introduce el periodo de tiempo deseado.	4. El sistema muestra todas las acciones de control registradas en ese tiempo que corresponden a las Direcciones y Delegaciones del MAC.
	5. El sistema muestra además muestra los siguientes campos: <ul style="list-style-type: none"> • Estado. • No Iniciadas. • En Proceso. • Terminadas. • Por Provincia. • Por Empresa. • Por Dirección. • Por Tipo de Acción de Control.
6. El revisor del parte selecciona el o los criterios de búsqueda deseados.	7. El sistema muestra un listado con las acciones de control que cumplan con los criterios seleccionados.

Periodo de Tiempo

Fecha de Inicio ...

Fecha de Fin ...

Criterios de Búsqueda

Estado:

- No Iniciadas
- En Proceso
- Terminadas

Provincia:

Direcciones:

Empresa

Tipo de Acción Control

Entidad	Cod REUP	Tipo de AC	Particularidades	F/I Planificada	F/T Planificada	F/I Real	F/T Real	Calificación	Afectación Económica
Provincia:CHabana									
MIHSAP									
Hospital Playa	45F12	AG	X	14/1/2007	21/1/2007	14/1/2007	29/1/2007	Satisfactoria	10 CUP
Hospital Frank Pais	25I12	VSC		3/3/2007	25/3/2007	15/3/2007	30/3/2007	Deficiente	80 CUC
Banco Sangre Lisa	58P54	AE	X	8/5/2007	26/5/2007	8/5/2007	26/5/2007	Deficiente	95 CUC

Flujo Alterno 4a "Mensaje de Error"

Acción del Actor

Respuesta del Sistema

4a.1 El sistema muestra un mensaje con el error correspondiente.

Prototipo de Interfaz

Flujo Alterno 7a "Mensaje de Error"

Acción del Actor	Respuesta del Sistema
	7a.1 El sistema muestra un mensaje con el error correspondiente.
Prototipo de Interfaz	
Poscondiciones	Parte Operativo realizado.

CONCLUSIONES:

En este capítulo se ha recordado el objeto de estudio, así como la situación problemática la cual condujo a la necesidad de la realización del presente trabajo. También se ha descrito detalladamente los procesos que van a ser objeto de automatización. Por otra parte se presentó el Modelo de Negocio, dentro del cual se muestran los modelos IDEF0 de los procesos mencionados anteriormente, luego se transitó por la especificación de los requerimientos funcionales y no funcionales, los cuales servirán de pilar para el desarrollo del módulo propuesto. Además se realizó una descripción textual de los Casos de Uso más significativos del subsistema y dentro de este epígrafe se trató lo referente a los actores del mismo, listado de casos de uso y diagrama de estos.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

3.1. Introducción:

Podemos decir que el diseño de un software se define como el proceso de aplicar ciertas técnicas y principios con el propósito de definir un sistema, con suficientes detalles como para permitir su interpretación y realización física. Transforma elementos estructurales de la arquitectura del programa. La importancia del Diseño del Software se puede definir en una sola palabra: calidad, dentro del diseño es donde se fomenta la calidad del Proyecto. El Diseño es la única manera de materializar con precisión los requerimientos del cliente.

Para el desarrollo de una aplicación completa se utilizan herramientas como esquemas (un esqueleto, un patrón) para el desarrollo y/o la implementación de una aplicación, el mismo se denomina framework. De aquí que en este capítulo se brindarán, las justificaciones necesarias y la importancia de utilizar un framework para el desarrollo del subsistema, además de una explicación detallada del Symfony que será el escogido y los patrones de diseño web que utiliza. Se visualizarán las clases del diseño de los casos de uso más significativos, así como el modelo de datos elaborado. Por último se desarrollará el estudio de la factibilidad, donde se expondrán los cálculos obtenidos y la descripción del COCOMO II con salidas de Puntos de Función que será el método utilizado.

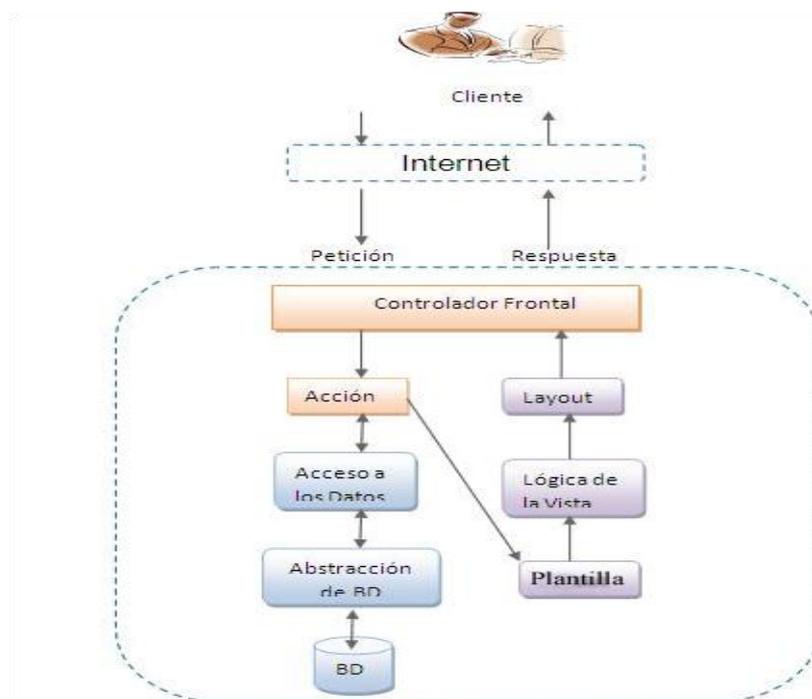
3.2. Descripción de la Arquitectura y el Framework utilizados:

3.2.1. Descripción del Framework

Un framework es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, un framework puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Un framework representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.

Es por ello que para el desarrollo de este trabajo se ha decidido utilizar el Framework Symfony, el cual es un framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas como el Propel, que es un framework que Symfony incorpora; este se encarga de la generación automática de las clases de la capa del modelo, ya que crea el esqueleto o estructura básica de las clases y genera automáticamente el código necesario (ORM). Este posee un componente llamado Creole que se encarga de la abstracción de la base de datos. Estas herramientas están encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Symfony automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación, un ejemplo de esto es que incluye su propio framework que proporciona soporte para las pruebas unitarias (aseguran que un único componente de la aplicación produce una salida correcta para una determinada entrada) llamado Lime, también como en muchas aplicaciones web se crean continuamente acciones y plantillas que realizan las operaciones CRUD (insertar, obtener, modificar y eliminar registro) para una determinada tabla de datos. En Symfony, el modelo contiene la información necesaria para poder generar de forma automática el código de las operaciones CRUD, de forma que se simplifica el desarrollo inicial de la aplicación y las interfaces de la parte de gestión de las aplicaciones. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web. En la imagen que se muestra a continuación se aprecia el flujo de trabajo del Symfony:



Dentro de las características de proyectos web que presenta el Symfony, tenemos que la capa de presentación utiliza plantillas y layouts que no son más que archivos con extensión .php que contienen los elementos que son comunes para un grupo de páginas, que pueden ser creados por diseñadores HTML sin ningún tipo de conocimiento del framework. Las ayudas o helpers incluidos permiten minimizar el código utilizado en la presentación, ya que encapsulan grandes bloques de código en llamadas simples a funciones. Todos estos helpers disponen de múltiples opciones que proporcionan una mayor flexibilidad, encontramos entre estos: Script.aculo.us es una librería de JavaScript (lenguaje de programación) que también está integrada en Symfony y que añade efectos visuales dinámicos que permiten mejorar la interfaz y la experiencia de usuario. JSON (JavaScript Objeto Notación) es un estándar utilizado para que un script¹² de cliente se comunique con un servidor, Prototype es una librería de JavaScript completamente integrada en Symfony y que simplifica el desarrollo de scripts mediante la definición de nuevas funciones y métodos de JavaScript. Los helpers de Ajax permiten al usuario actualizar partes de la página web pinchando sobre un enlace, enviando un formulario o modificando un elemento de formulario. Estos permiten encapsular los efectos Java Script compatibles con todos los navegadores en una única línea de código.

Los formularios incluyen validación automatizada y relleno automático de datos (“repopulation”), este último se pone de manifiesto cuando al ocurrir un error en los datos asociados a un formulario, este se vuelve a cargar con los datos que fueron introducidos por el usuario, para que corrija el o los campos con error y no tenga que escribirlo todo nuevamente; todo lo mencionado anteriormente asegura la obtención de datos correctos y mejora la experiencia de usuario. Los datos incluyen mecanismos de escape que permiten una mejor protección contra los ataques producidos por datos corruptos.

Podemos mencionar que Symfony se integra a varios entornos de desarrollo o IDEs diferentes como Eclipse y Aptana, además crea por defecto 3 entornos: producción (prod, registra los errores), pruebas (test, registra las alertas y los errores en el archivo de log) y desarrollo (dev, se suele desactivar la caché, pero está activa en los entornos de pruebas y de producción). Las herramientas que generan automáticamente código han sido diseñadas para hacer prototipos de aplicaciones y para crear fácilmente la parte de gestión de las aplicaciones. El completo sistema de log (errores) permite a los administradores acceder hasta el último detalle de las actividades que realiza la aplicación.

¹² Guión o conjunto de instrucciones, permiten la automatización de tareas creando pequeñas utilidades.

3.2.2. Arquitectura y Patrones:

La arquitectura del proyecto es en capas e implementa el patrón arquitectónico Modelo Vista Controlador (MVC).

- El modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La vista transforma el modelo en una página web que permite al usuario interactuar con ella.
- El controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

El principio más importante de la arquitectura MVC es la separación del código del programa en tres capas, dependiendo de su naturaleza. La lógica relacionada con los datos se incluye en el modelo, el código de la presentación en la vista y la lógica de la aplicación en el controlador. La programación se puede simplificar si se utilizan otros patrones de diseño. De esta forma, las capas del modelo, la vista y el controlador se pueden subdividir en más capas.

El MVC en el Symfony separa la presentación (vista) de manera tal que el código PHP puro con toda la lógica de negocio se incluye en el script del controlador, el código HTML, que contiene cierto código PHP a modo de plantilla, se almacena en el script de la vista. Esto es una forma de trabajo muy eficiente ya que se puede obtener una vista con la cantidad mínima de código de PHP y además tener la lógica centralizada en el script del controlador y poder reutilizar este en otras presentaciones.

La capa de la vista también puede aprovechar la separación de código. Las páginas web suelen contener elementos que se muestran de forma idéntica a lo largo de toda la aplicación: cabeceras de la página, el layout genérico, el pie de página y la navegación global. Normalmente sólo cambia el interior de la página. Por este motivo, la vista se separa en un layout y en una plantilla. Normalmente, el layout es global en toda la aplicación o al menos en un grupo de páginas. La plantilla sólo se encarga de visualizar las variables definidas en el controlador. El contenido de la plantilla se integra en el layout, o si se mira desde el otro punto de vista, el layout decora la plantilla. El comportamiento descrito anteriormente es una implementación del patrón de diseño llamado “**Decorator**”

Para evitar ciertas dificultades que pueden aparecer al trabajar con la base de datos, todo el código que se encarga de la manipulación de los datos se separa del controlador, y se coloca en otro script denominado modelo (modelo.php) el cual solo se encarga del acceso a datos y puede ser reorganizado a tal efecto.

Todos los parámetros que no dependen de la capa de datos (como por ejemplo los parámetros de la petición del usuario) se deben obtener a través del controlador y por tanto, no se puede acceder a ellos directamente desde el modelo. Las funciones del modelo se pueden reutilizar fácilmente en otros controladores.

La capa del modelo se puede dividir en la capa de acceso a los datos y en la capa de abstracción de la base de datos. De esta forma, las funciones que acceden a los datos no utilizan sentencias ni consultas que dependen de una base de datos, sino que utilizan otras funciones para realizar las consultas. Así, si se cambia de sistema gestor de bases de datos, solamente es necesario actualizar la capa de abstracción de la base de datos.

Las funciones creadas en la capa de abstracción de la base de datos se pueden reutilizar en otras funciones del modelo que necesiten acceder a la base de datos. Dentro de esta capa podemos encontrar los siguientes patrones de diseño:

- **Active Record** : Una de las virtudes del patrón es representar de forma Orientada a Objetos los datos de una Base de Datos Relacional - modelo conocido también como ORM o “Object-Relational Mapping” - , definiendo interfaces sencillas para acceder y manipular esos datos. Active Record es un enfoque al problema de acceder a los datos de una base de datos. Una fila en la tabla de la base de datos (o vista) se envuelve en una clase, de manera que se asocian filas únicas de la base de datos con objetos del lenguaje de programación usado. Cuando se crea uno de estos objetos, se añade una fila a la tabla de la base de datos. Cuando se modifican los atributos del objeto, se actualiza la fila de la base de datos. La clase envoltorio implementa métodos de acceso para cada columna de la tabla o vista.
- **Active Table**: Es un objeto que puede persistir otros objetos en medios no volátiles, como bases de datos. La función de una clase que implementa este patrón es de comunicarse con la base de datos y regresarnos objetos como los tenemos definidos para nuestra aplicación.

- **Facade (Fachada):** Simplifica los accesos a las clases de la capa de acceso a datos proporcionando un objeto que todas las clases de capas superiores utilizarán para acceder a las clases contenidas en la capa del modelo. Define una interface de más alto nivel que permite usar el sistema más fácil. El objetivo de la aplicación de este patrón es reducir la dependencia entre clases. FACADE ofrece un punto de acceso al resto de clases, si estas cambian o se sustituyen por otras solo hay que actualizar la clase Facade sin que el cambio afecte a las aplicaciones cliente. Facade no oculta las clases sino que ofrece una forma más sencilla de acceder a ellas, en los casos en que se requiere se puede acceder directamente a ellas.

En las aplicaciones web el controlador suele tener mucho trabajo. Entre las tareas comunes se encuentran el manejo de las peticiones del usuario, el manejo de la seguridad, cargar la configuración de la aplicación y otras tareas similares. Por este motivo, el controlador normalmente se divide en un controlador frontal, que es único para cada aplicación, y las acciones, que incluyen el código específico del controlador de cada página. Una de las principales ventajas de utilizar un controlador frontal es que ofrece un punto de entrada único para toda la aplicación. Así, en caso de que sea necesario impedir el acceso a la aplicación, solamente es necesario editar el script correspondiente al controlador frontal. Si la aplicación no dispone de controlador frontal, se debería modificar cada uno de los controladores. En esta capa encontramos los patrones:

- **Front Controller:** Está pensado para centralizar las invocaciones de lógica de la aplicación en un único lugar. Todo lo que se pueda hacer o se haga en una aplicación lo podremos ver desde el FrontController. Paralelamente, todas las acciones invocadas (ejecución de commands) están centralizada de forma que si los requerimientos lo demandasen se podría añadir funcionalidad transversal a ellos. Es el componente que recibe los requerimientos, los envía a los elementos encargados de procesar la lógica y luego lo envía nuevamente a la vista (esta vez incluyendo los datos obtenidos).
- **Singleton:** (instancia única) está diseñada para restringir la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto. Su intención consiste en garantizar que una clase sólo tenga una instancia y proporcionar un punto de acceso global a ella. El patrón Singleton se implementa creando en nuestra clase un método que crea una instancia del objeto

sólo si todavía no existe alguna. Para asegurar que la clase no puede ser instanciada nuevamente se regula el alcance del constructor (con atributos como protegido o privado).

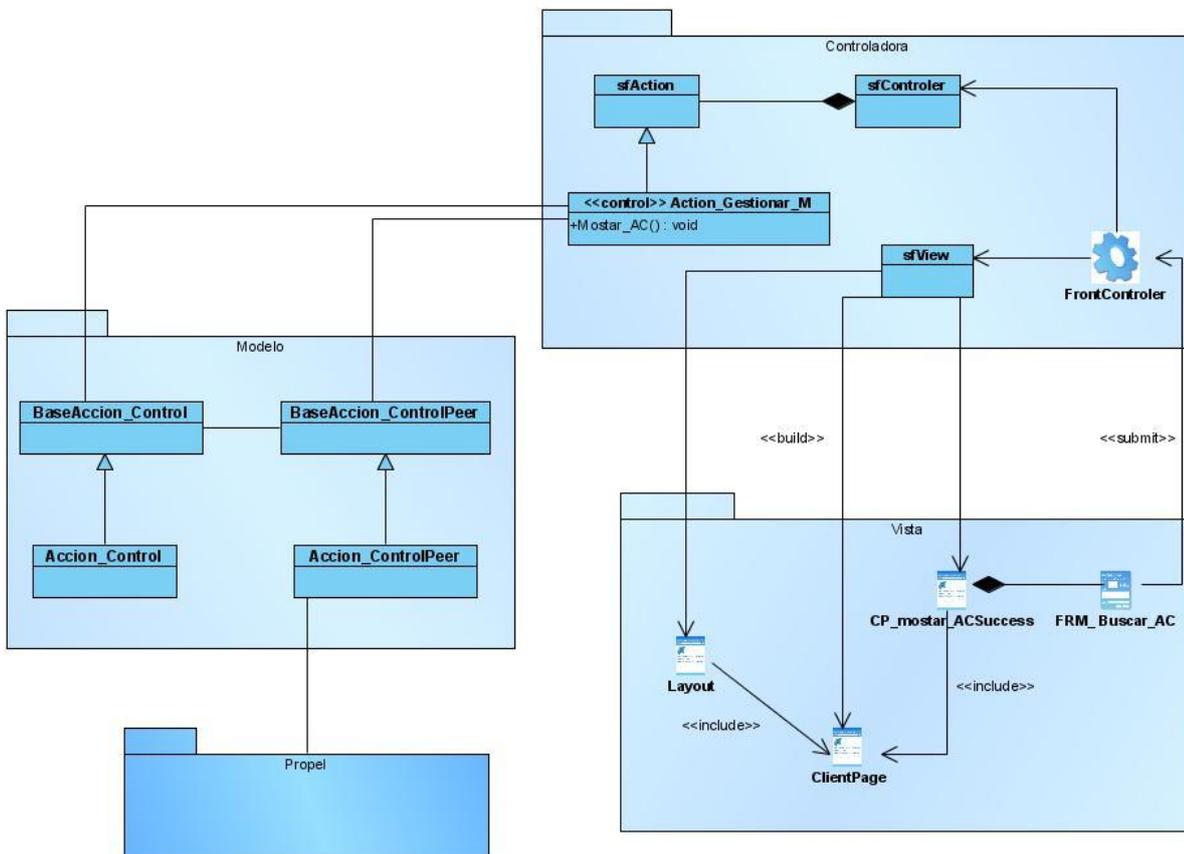
El patrón Singleton provee una única instancia global gracias a que:

- La propia clase es responsable de crear la única instancia.
- Permite el acceso global a dicha instancia mediante un método de clase.
- Declara el constructor de clase como privado para que no sea instanciable directamente.

Los Patrones GRAPS también juegan un papel importante en este trabajo, donde se usaron los siguientes:

- **Creador:** En las clases fachada y fachada_m se encuentran los métodos definidos para la dirección de PAC. En estos métodos se crean los objetos de las clases que representan las entidades, evidenciando de este modo que la clases fachadas son "creadoras" de dichas entidades.
- **Experto:** Este es uno de los más utilizados, puesto que Propel es la librería externa que utiliza Symfony para realizar su capa de abstracción en el modelo, encapsula toda la lógica de los datos y son generadas las clases con todas las funcionalidades comunes de las entidades.
- **Alta Cohesión:** Symfony permite asignar responsabilidades con una alta cohesión, por ejemplo la clase Actions 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 (propiedades), es decir, está formada por diferentes funcionalidades que se encuentran estrechamente relacionadas proporcionando que el software sea flexible frente a grandes cambios.
- **Controlador:** Todas las peticiones Web son manejadas por un solo controlador frontal (FrontController), que es el punto de entrada único de toda la aplicación en un entorno determinado. Cuando el controlador frontal recibe una petición, utiliza el sistema de enrutamiento para asociar el nombre de una acción y el nombre de un módulo con la URL entrada por el usuario.
- **Bajo Acoplamiento:** La clase Action hereda solamente de sfActions para lograr un bajo acoplamiento de clases.

En la siguiente figura se muestra un ejemplo del funcionamiento de Symfony con la aplicación de los patrones anteriormente mencionados.

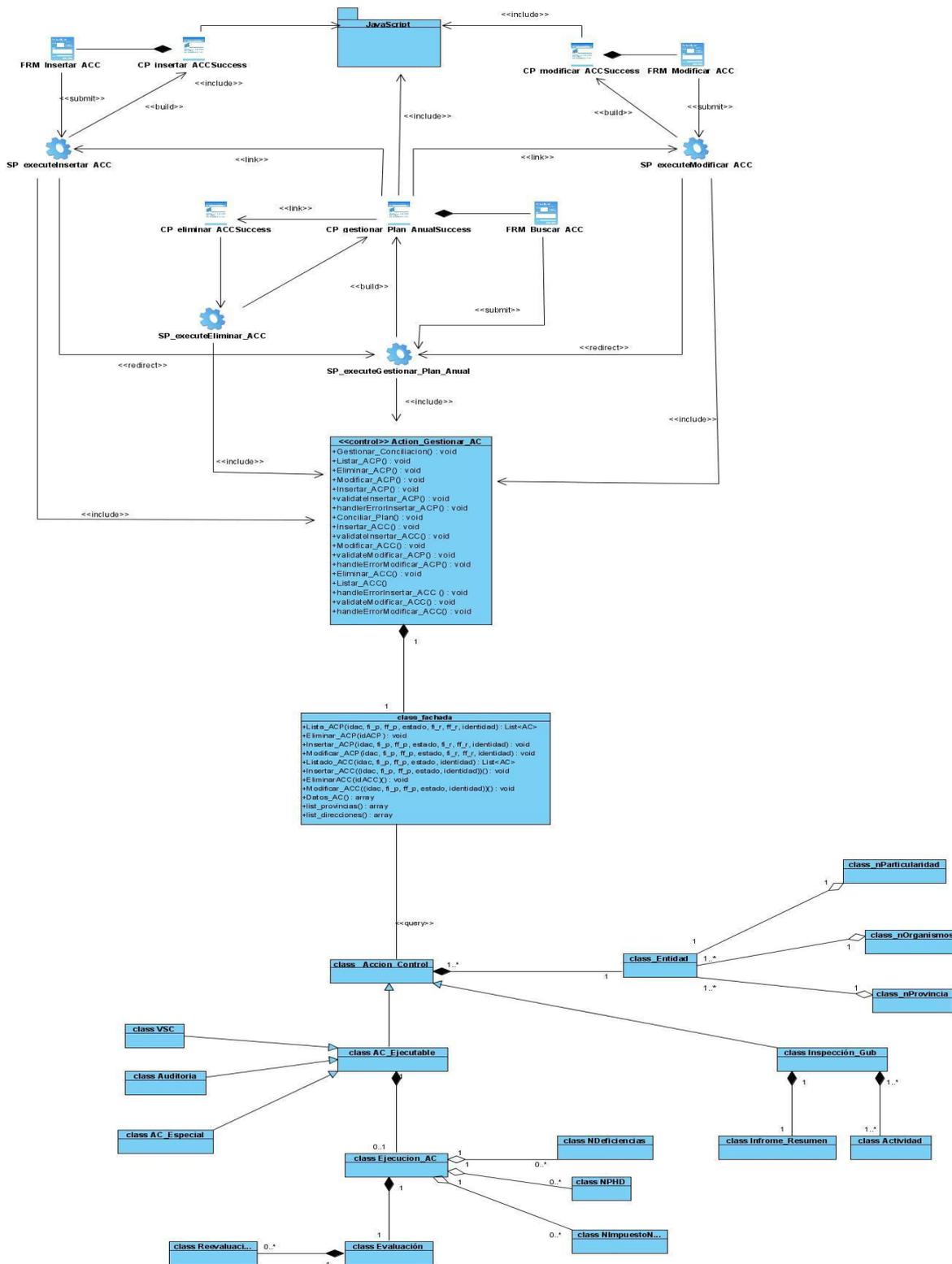


3.3. Clases del Diseño:

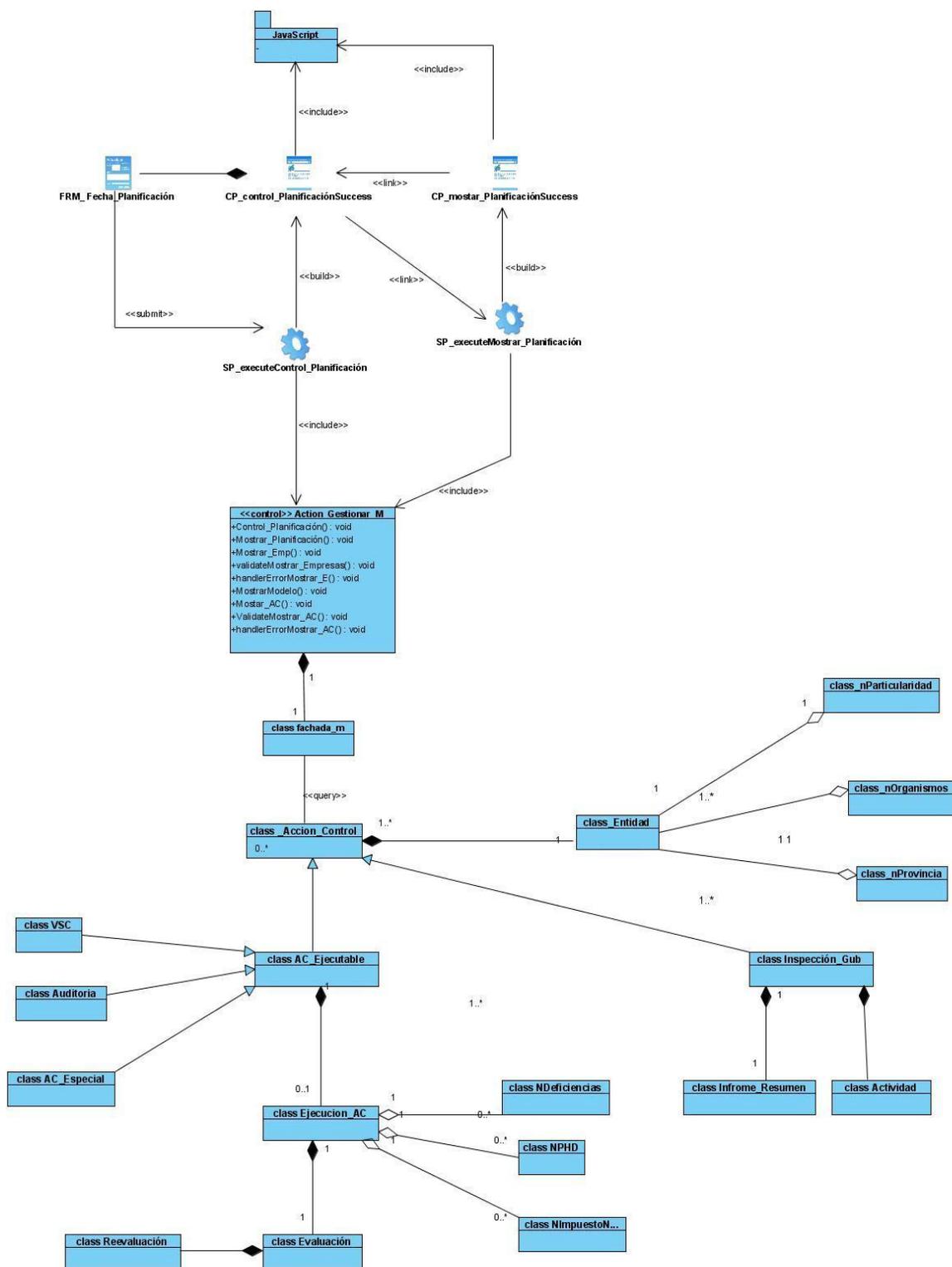
En este epígrafe se describe gráficamente las especificaciones de las clases del software y de las interfaces en una aplicación. Contiene la siguiente información:

- Clases, asociaciones y atributos.
- Interfaces, con sus operaciones y constantes.
- Métodos.
- Navegabilidad.
- Dependencias.

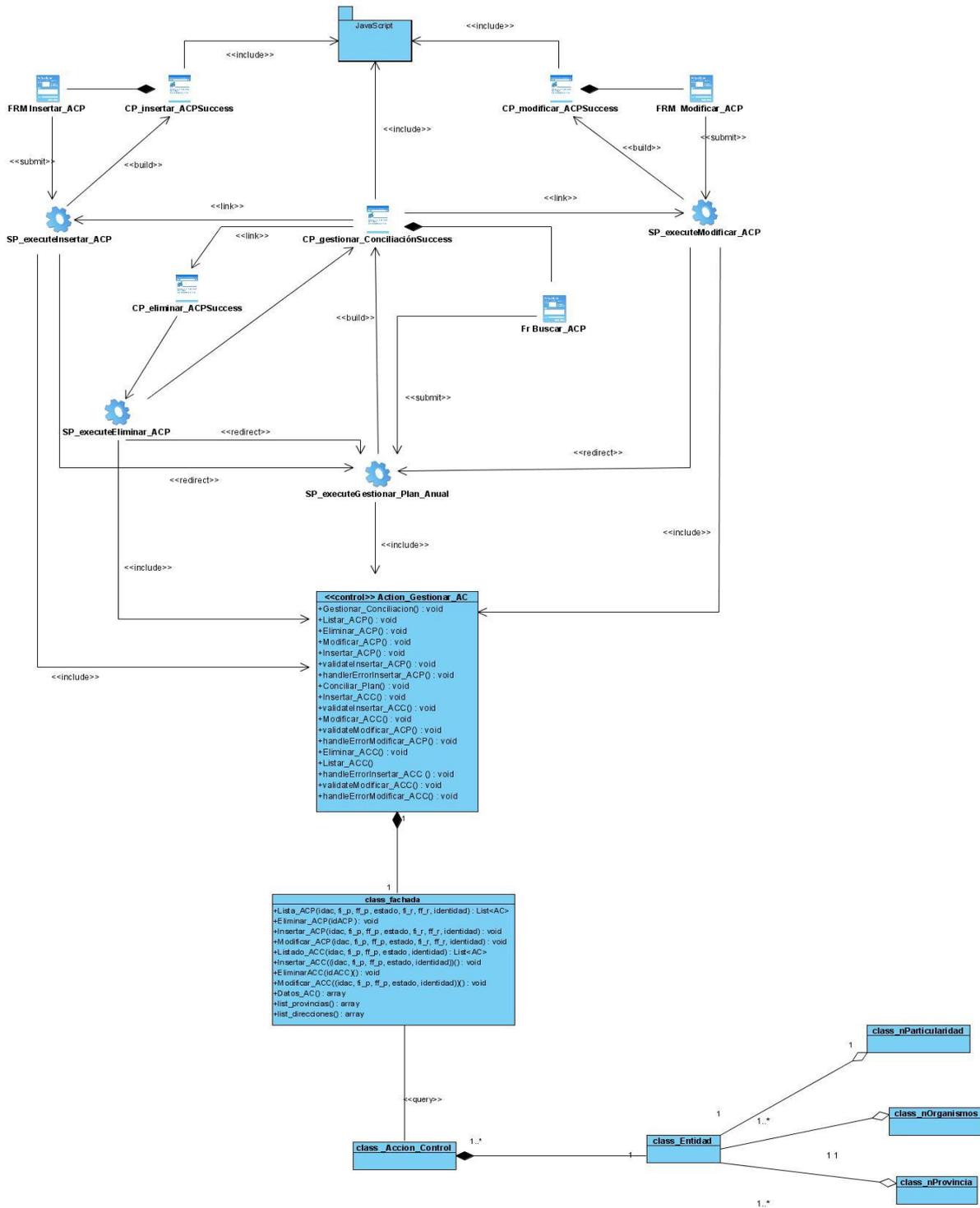
3.3.1. Gestionar Plan Anual:



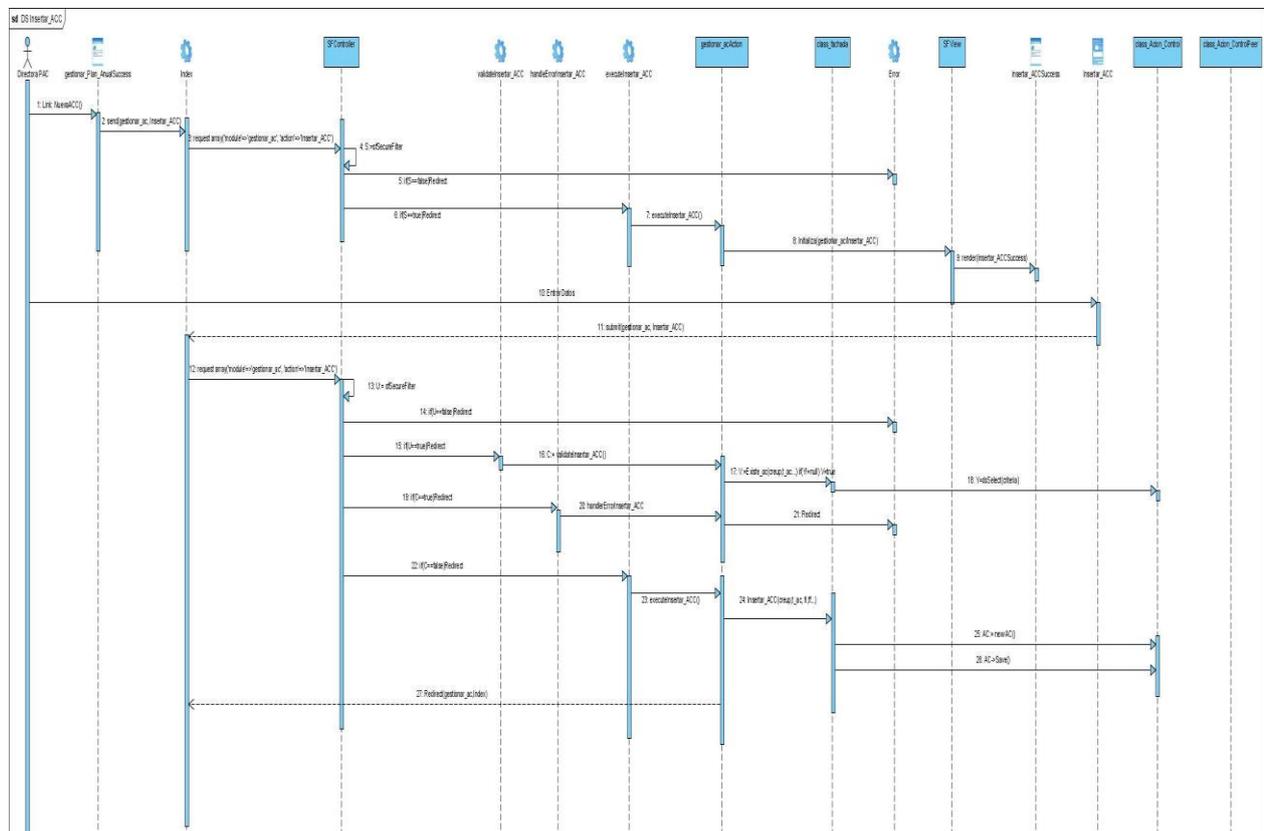
3.3.2. Control de la Planificación:



3.3.3. Conciliar Plan Anual



A continuación se muestra un ejemplo de diagrama de secuencia, que da detalle al escenario Insertar ACC del caso de uso Conciliar Plan Anual.



3.6. Estudio de la Factibilidad.

3.6.1. Descripción del Método utilizado

Para la estimación del tamaño de un sistema a partir de sus requerimientos, existen, entre otras, dos técnicas bastante difundidas, la técnica de Puntos de Función y la de COCOMO II.

La técnica de **Puntos de Función** tiene como propósito medir el software cualificando la funcionalidad que proporciona externamente, basándose en el diseño lógico del sistema. Los objetivos de los puntos de función son:

- Medir lo que el usuario pide y lo que el usuario recibe.
- Medir independientemente la tecnología utilizada en la implantación del sistema.

- Proporcionar una métrica de tamaño que dé soporte al análisis de la calidad y la productividad.
- Proporcionar un medio para la estimación del software.
- Proporcionar un factor de normalización para la comparación de distintos software.

El análisis de los puntos de función se desarrolla considerando cinco parámetros básicos externos del sistema:

1. Entrada (EI, del inglés External Input).
2. Salida (EO, del inglés External Output).
3. Consultas (EQ, del inglés External Query).
4. Ficheros Lógicos Internos (ILF, del inglés Internal Logic File).
5. Ficheros Lógicos Externos (EIF, del inglés External Interface File).

Con estos parámetros, se determinan los puntos de función sin ajustar (PFsA). A este valor, se le aplica un factor de ajuste obtenido en base a unas valoraciones subjetivas sobre la aplicación y su entorno; es decir, las características generales del sistema.

COCOMO II, es uno de los modelos de estimación de costos mejor documentados y utilizados. El modelo permite determinar el esfuerzo y tiempo que se requiere en un proyecto de software a partir de una medida del tamaño del mismo expresada en el número de líneas de código que se estimen generar para la creación del producto software. Este modelo permite estimar el esfuerzo y tiempo de un proyecto de software en dos etapas diferentes: diseño temprano y post-arquitectura.

El método que se usará para la estimación será **COCOMO II** con salida de **Puntos de Función**. COCOMO II permite determinar el esfuerzo y tiempo de un proyecto de software a partir de los Puntos de Función sin ajustar, lo cual supone una gran ventaja, dado que en la mayoría de los casos es difícil determinar el número de líneas de código de que constará un nuevo desarrollo, en especial cuando se tiene poca o ninguna experiencia previa en proyectos de software. Esto hace que ambos modelos, Puntos de Función y COCOMO II, sean perfectamente compatibles y complementarios.

3.6.2. Cálculo del esfuerzo.

$$PM_{nominal} = A \times (Size)^B$$

PM_{nominal}: es el esfuerzo nominal requerido en meses-hombre.

Size: es el tamaño estimado del software en:

- Miles de líneas de código (KSLOC).
- Puntos de Función sin ajustar.

Size = factor de conversión de PHP * **UFP** (Puntos de función sin ajustar)

UFP = $\sum A_{aporte}$.

A: es una constante con un valor de 2.94.

B: Factor escalar, su valor está dado por la resultante de los aspectos positivos sobre los negativos que presenta el proyecto. **B = 1.07**.

	Complejidad			Aporte
	Baja	Media	Alta	
Entradas Externas.	1		6	39
Salidas Externas.	1			4
Consultas Externas.			7	42
Archivos Lógicos Internos.	19			133
Archivos de Interface Externos.				
			Total	218

$$Size = 30 * 218$$

$$Size = 6.54 \text{ KSLOC.}$$

$$PM_{nominal} = 2.94 \times (6.54)^{1.07}$$

$$PM_{nominal} = 2.94 \times 7.46$$

$$PM_{nominal} = 21.9324 \text{ meses/hombre.}$$

Para completar la estimación, hay que ajustar el esfuerzo nominal de acuerdo a las características del proyecto. El ajuste se efectúa aplicando la ecuación:

$$PM_{ajustado} = PM_{nominal} \times \Pi(ME_i)$$

Donde los ME_i (multiplicadores de esfuerzo) varían en función del modelo de estimación seleccionado (Diseño Preliminar o Post arquitectura). En nuestro caso vamos a aplicar el modelo de Diseño preliminar. Entonces, cuantificamos los multiplicadores de esfuerzo para éste modelo:

Multiplicador	Descripción	Ponderación	Valor
PERS	Se tienen analistas y programadores con alta eficiencia y capacidad de trabajo en equipo. Dedicación full-time.	Nominal	1
RCPX	Las exigencias de confiabilidad, documentación y volumen de datos son moderadas, y la complejidad del producto es baja.	Nominal	1
RUSE	No se pretende reutilizar nada	Bajo	0.95
PDIF	No existen restricciones en cuando al tiempo de CPU o al consumo de memoria, la plataforma es muy estable.	Bajo	0.87
PREX	Tanto los analistas como los programadores tienen aproximadamente 6 meses de experiencia en la aplicación, la plataforma, el lenguaje y las herramientas utilizadas.	Muy Bajo	1.33
SCED	Se requiere terminar el proyecto en el tiempo estimado.	Nominal	1
FCIL	Se tienen herramientas CASE simples e infraestructura de comunicaciones básica.	Bajo	1.10

		Total	1.20
--	--	--------------	-------------

PMajustado = 21.9324 x 1.20

PMajustado = 26.3189 meses/hombre.

En la realización de este módulo trabajan dos personas, por tanto el valor del esfuerzo nominal ajustado se divide entre dos, y el resultado se multiplica por 160 para llevar de meses/hombre a horas/hombre y se obtiene como valor final del esfuerzo nominal, **2105.5 horas/hombre**.

CONCLUSIONES

El framework `Symfony` es un enorme conjunto de herramientas y utilidades que simplifican el desarrollo de las aplicaciones web. Este framework emplea el tradicional patrón de diseño MVC (modelo-vista-controlador) para separar las distintas partes que forman una aplicación web. El modelo representa la información con la que trabaja la aplicación y se encarga de acceder a los datos. Gracias a su versatilidad y sus posibilidades de configuración, `Symfony` es un framework adecuado para cualquier proyecto de aplicación web.

La arquitectura MVC proporciona grandes ventajas, como la organización del código, la reutilización, la flexibilidad y una programación mucho más entretenida. Por si fuera poco, crear la aplicación con `Symfony` permite crear páginas XHTML válidas, depurar fácilmente las aplicaciones, crear una configuración sencilla, abstracción de la base de datos utilizada, enrutamiento con URL limpias, varios entornos de desarrollo y muchas otras utilidades para el desarrollo de aplicaciones.

En las clases del diseño se obtuvo un diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro. El modelo de datos consiste en los objetos (entidades que existen y que se manipulan), atributos (características básicas de estos objetos) y las relaciones (forma en que enlazan los distintos objetos entre sí). El COCOMO II con salidas de Puntos de Función es un método muy factible para estimar la duración de un proyecto según su desarrollo y la elaboración del software del presente trabajo cuenta con un valor establecido adecuado.

CONCLUSIONES

Al concluir la investigación realizada para este trabajo se han dado cumplimiento a los objetivos planteados, obteniéndose los siguientes resultados:

- Se realizó un estudio y profundización de las tecnologías y metodologías de trabajo utilizadas para el desarrollo de la aplicación.
- Se desarrolló una investigación y análisis de los procesos que se llevan a cabo en la Dirección de Planificación, Análisis y Control del MAC, centrándose dicha investigación en los procesos de planificación y control realizados en esta dirección.
- Se realizó el diseño de un sistema que permita gestionar la planificación de las acciones de control y el control de dicha planificación para facilitar el trabajo en la Dirección de Planificación, Análisis y Control del MAC, dejando un camino bien claro y organizado para la futura implementación de dicho sistema.
- La primera versión del prototipo no funcional cumple con los requisitos funcionales y no funcionales que fueron planteados en el transcurso del presente trabajo.

Finalmente, luego de haber cumplido con los objetivos y tareas que fueron planteadas al iniciar este trabajo, se realizó un estudio de la factibilidad que justifica la necesidad de desarrollar el sistema, llegando a la conclusión que este es totalmente factible y es necesaria su puesta en funcionamiento.

RECOMENDACIONES

Como recomendaciones de este trabajo se propone:

1. Implementar un sistema con el diseño propuesto que permita gestionar la planificación de las acciones de control y el control de dicha planificación para facilitar el trabajo en la Dirección de Planificación, Análisis y Control del MAC.
2. Aplicar dicho sistema en el MAC para resolver los problemas existentes en este sobre todo los relacionado con la DPAC.
3. Realizar mejoras, como son alertas para informar de la fecha de inicio de una acción de control planificada.

REFERENCIAS BIBLIOGRÁFICAS

SANTANDER, J. L. E. *Diccionario Económico*, Editora Política 1987.

BIBLIOGRAFÍA

Avances en informatización de la sociedad en Cuba *Trabajadores*, 2007.

Visio 2003: Información de Producto. Disponible en:
<http://www.microsoft.com/latam/office/visio/prodinfo/default.aspx>

ARENCIBIA, L. H. *Análisis del proceso de informatización de la sociedad cubana hoy*.

EMMANUELLE GOULEAU, O. M., TRISTAN RIVOALLAN, VINCENT LEMAIRE, XAVIER LACOT. *Frameworks PHP pour l'entreprise. Définition, critères de choix et analyses*, Livre Blanc, 2008.

ESTADO, C. D. *DECRETO-LEY NÚMERO 159*, 1995.

FABIEN POTENCIER, F. Z. *Symfony la guía definitiva*, 2007.

FUENTE, R. D. L. *El carácter multidisciplinario de la Auditoría y el control en TI. Revista Percepciones N° 3*, 2000.

GARCÍA, I. R. P. *Impacto de la Informatización en la Sociedad Cubana*, 2006.

---. *Impacto de la Informatización en la Sociedad Cubana. Ciencia, tecnología y sociedad*, 2005.

GONZÁLEZ, C. D. *Base de Datos PostgreSQL, SQL avanzado y PHP*, 2008.

HENST., C. V. D. *¿Qué es el PHP?*, 2001.

ING. YISEL ALONSO RIVERÓN, I. Y. C. N., ING. YORDANIS TORNES MEDINA. *IDEF Una alternativa para modelamiento de negocio con RUP*, 2008.

JIMENEZ, A. *Historia de la Auditoría*, 2003.

LARMAN, C. *UML y Patrones Introducción al análisis y diseño orientado a objetos*. México, Prentice Hall Hispanoamericana, S.A, 1999. 563 p. 970-17-0261-1

MINISTROS., C. E. D. C. D. *Acuerdo No. 4374 REGLAMENTO MAC*, 2002.

PECOS, D. *PostGreSQL vs. MySQL*. Disponible en:
http://www.netpecos.org/docs/mysql_postgres/index.html

PERALTA, M. *Estimación del Esfuerzo basada en Casos de Uso*.

POW-SANG, J. *Estudio de Técnicas basadas en Puntos de Función para la Estimación del esfuerzo en proyectos de Software*, 2004.

RICARDO. *Herramientas de Gestión de Procesos*, 2007. [Disponible en: <http://ronade-rjr.blogspot.com/>

SOFTWARE, S. Disponible en: <http://www.softexpert.com.mx/>

SWEAT, J. E. *Guide to PHP Design Patterns*, Marco Tabini & Associates, Inc., 2005.

TRENDS, G., 2008.

WOLINSKY, D. J. *AUDITORÍA Y CONTROL INTERNO*, 2002.

ANEXOS

Anexo 1: Requisitos Funcionales

RF1 Mostrar Acciones de Control Planificadas.

1. Buscar empresas mediante:

1.1. Direcciones:

1.1.1. Dirección de Control Gubernamental.

1.1.2. Dirección de Auditoría Gubernamental.

1.1.3. Dirección de Atención al Sistema Nacional de Auditorías

1.2 Delegaciones.

RF2 Insertar nueva acción de control al Plan.

2.1 Introducir los siguientes datos:

2.1.1. Ministerio.

2.1.2. Provincia.

2.1.3. Empresa.

2.1.4. Código REUP.

2.1.5. Tipo de Acción.

2.1.5.1. Auditoría Gestión.

2.1.5.2. Auditoría Financiera.

2.1.5.3. Auditoría Fiscal.

2.1.5.4. Auditoría Especial.

2.1.5.5. Auditoría Recurrente.

2.1.5.6. Visita de Supervisión y Control.

2.1.5.7. Inspección Gubernamental.

2.1.5.8. Comprobaciones Especiales.

2.1.6. Particularidades.

2.1.6.1. Perfeccionamiento Empresarial.

2.1.6.2. Presupuestada.

2.1.6.3. Inversión Extranjera.

2.1.6.4. En el Exterior.

2.1.7. Fecha de Inicio Planificada.

2.1.8. Fecha de Terminación Planificada.

2.1.9. Fecha de Inicio Real.

2.1.10. Fecha de Terminación Real.

2.1.11. Calificación.

2.1.11.1. PHD.

2.1.11.2. PHC.

2.1.12. Afectación Económica.

2.1.12.1. CUP.

2.1.12.2. CUC.

2.1.13. Procedencia.

RF3 Modificar una acción de control del plan.

3.1 Modificar los siguientes datos de la acción de control seleccionada:

3.1.1 Tipo de Acción.

3.1.1.1 Auditoría Gestión.

3.1.1.2 Auditoría Financiera.

3.1.1.3 Auditoría Fiscal.

3.1.1.4 Auditoría Especial.

- 3.1.1.5 Auditoría Recurrente.
- 3.1.1.6 Visita de Supervisión y Control.
- 3.1.1.7 Inspección Gubernamental.
- 3.1.1.8 Comprobaciones Especiales.

3.1.2 Fecha de Inicio Planificada.

3.1.3 Fecha de Terminación Planificada.

3.1.4 Calificación.

3.1.4.1 PHD.

3.1.4.2 PHC.

3.1.5 Afectación Económica.

3.1.5.1.1 CUP.

3.1.5.1.2 CUC.

3.1.6 Procedencia.

RF4 Eliminar una acción de control seleccionada del Plan.

4.1 Buscar Acción de Control a eliminar.

4.2 Eliminar Acción de control seleccionada.

RF6 Controlar el Plan Anual.

6.1 Buscar las acciones de control establecidas en el rango de tiempo introducido.

6.1.1 Fecha Inicial.

6.1.2 Fecha Final.

6.2 Buscar las acciones de control en ese intervalo mediante:

6.2.1.1 Estado.

6.2.1.1.1 No Iniciadas.

6.2.1.1.2 En Proceso

6.2.1.1.3 Terminadas

6.2.1.2 Provincia.

6.2.1.3 Empresa.

6.2.1.4 Dirección.

6.2.1.5 Tipo de Acción de Control

6.3 Mostrar el siguiente resultado de búsqueda que cumpla con lo anterior:

6.3.1 Entidad.

6.3.2 Código REUP.

6.3.3 Tipo de Acción de Control.

6.3.4 Particularidades.

6.3.5 Fecha Inicio Planificada.

6.3.6 Fecha Final Planificada.

6.3.7 Fecha Inicio Real.

6.3.8 Fecha Final Real.

6.3.9 Calificación.

6.3.10 Afectación Económica.

RF7 Mostrar las acciones de control a conciliar por:

7.1. Buscar acciones de control mediante

7.1.1. Direcciones:

7.1.2. Delegaciones.

7.1.2.1.1. Dirección de Control Gubernamental.

7.1.2.1.2. Dirección de Auditoría Gubernamental

7.1.2.1.3. Dirección de Atención al Sistema Nacional de Auditorías

7.2. Mostrar el siguiente resultado:

7.2.1. Tipo de AC

7.2.2. Fecha Inicio Planificada.

7.2.3. Fecha Fin Planificada.

7.2.4. Fecha Inicio Real.

7.2.5. Fecha Fin Real.

RF8 Mostrar Empresas por criterios:

8.1 Buscar Empresas mediante

8.1.1 Empresas con calificaciones de:

8.1.1.1 Satisfactoria.

8.1.1.2 Aceptable.

8.1.1.3 Deficiente.

8.1.1.4 Mal.

8.1.2 Empresas con dos calificaciones Deficientes o Mal.

8.1.3 Empresas en Perfeccionamiento Empresarial.

8.1.4 Empresas Mixtas.

8.1.5 Empresas en el exterior.

8.1.6 Empresas Presupuestada

8.2 Mostar el siguiente resultados de la búsqueda

8.2.1 Nombre de la Empresa.

8.2.2 Provincia.

8.2.3 Ministerio al que pertenece.

RF 9 Buscar Acciones de Control hechas a una empresa.

9.1 Buscar empresa por los siguientes criterios de búsqueda o combinación de los mismos:

9.1.1 Provincia

9.1.2 Código REUP

9.1.3 Nombre de la empresa.

9.2 Mostrar los resultado de la búsqueda:

9.2.1 Mostar las acciones de control realizadas a la empresa deseada con los siguientes datos:

9.2.2 Tipo de Acción de Control.

9.2.3 Fecha de Inicio Planificada.

9.2.4 Fecha de Fin Planificada.

9.2.5 Fecha de Inicio Real.

9.2.6 Fecha de Fin Real.

RF10 Mostrar Modelos de la Orden de Trabajo.

10.1 Buscar modelo según el nombre escogido:

10.1.1 PHD

10.1.2 Deficiencias.

10.2 Mostrar el formato del modelo deseado.

Anexo 2: Descripción textual de los casos de uso del sistema.

Caso de Uso:	Elaborar plan
Actores:	Directora PAC
Resumen:	El caso de uso se inicia cuando la directora del DPAC selecciona la opción Elaborar Plan. Posteriormente se muestra un listado con todas las acciones de control registradas agrupadas por Provincia y Ministerio.
Precondiciones:	Existencia de los planes de las Delegaciones y Direcciones del MAC.
Referencias	RF5
Prioridad	Secundario.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. La directora del DPAC selecciona del menú la opción “Elaborar plan”.	2. El sistema muestra el listado de todas las acciones de control que fueron conciliadas, agrupadas por provincia y organismo, provenientes de todas las direcciones y delegaciones del MAC, con los siguientes datos: <ul style="list-style-type: none"> • Ministerio. • Provincia. • Empresa. • Código REUP. • Tipo de Acción. <ul style="list-style-type: none"> • Auditoría Gestión. • Auditoría Financiera. • Auditoría Fiscal.

- Auditoría Especial.
- Auditoría Recurrente.
- Visita de Supervisión y Control.
- Inspección Gubernamental.
- Comprobaciones Especiales.
- Particularidades.
 - Perfeccionamiento Empresarial.
 - Presupuestada.
 - Inversión Extranjera.
 - En el Exterior.
- Fecha de Inicio Planificada.
- Fecha de Terminación Planificada.
- Fecha de Inicio Real.
- Fecha de Terminación Real.
- Calificación.
- PHD.
- PHC.
- Afectación Económica.
 - CUP.
 - CUC.

Plan Anual



Entidad	Cod REUP	Tipo de AC	Particularidades	F/I Planificada	F/T Planificada	F/I Real	F/T Real	Calificación	Afectación Económica
Provincia:CHabana									
MIISAP									
Hospital Playa	45F12	AG	X	14/1/2007	21/1/2007	14/1/2007	29/1/2007	Satisfactoria	10 CUP
Hospital Frank Pais	25I12	VSC		3/3/2007	25/3/2007	15/3/2007	30/3/2007	Deficiente	80 CUC
Banco Sangre Lisa	58P54	AE	X	8/5/2007	26/5/2007	8/5/2007	26/5/2007	Deficiente	95 CUC

Flujos Alternos

Flujo Alterno 2.a “Mensaje de Error”

Acción del Actor

Respuesta del Sistema

2a.1 El sistema muestra un mensaje según el error que se produce.

Prototipo de Interfaz

Poscondiciones

Plan Anual elaborado.

Caso de Uso:	Mostrar Empresas
Actores:	Auditor.
Resumen:	El caso de uso inicia cuando el actor desea listar empresas por diferentes criterios para complementar el Plan Trimestral. Posteriormente se muestra una lista con las empresas que cumplen con los criterios de búsquedas seleccionados.
Precondiciones:	Existencia de acciones de control registradas en el sistema.
Referencias	RF8
Prioridad	Secundario.

Flujo Normal de Eventos

Acción del Actor

Respuesta del Sistema

1. El auditor selecciona del menú la opción Mostrar AC.

2. El sistema muestra el formulario “mostrar_e” con los siguientes campos:

- Empresas con calificaciones de:

	<ul style="list-style-type: none"> • Satisfactoria • Aceptable • Deficiente • Mal • Empresas con dos calificaciones Deficientes o Mal. • Empresas en Perfeccionamiento Empresarial. • Empresas Mixtas. • Empresas en el exterior. • Empresas Presupuestadas.
<p>3. El auditor selecciona la o las opciones por las desea que s ele muestren la información.</p>	<p>4. El sistema muestra un listado con las empresas que cumplen con los criterios de búsquedas seleccionados, con los siguientes datos:</p> <ul style="list-style-type: none"> • Nombre de la Empresa. • Provincia. • Ministerio al que pertenece.

Criterios de Búsqueda

Calificaciones:

Satisfactoria
 Dos calificaciones Deficientes o Mal
 En el Exterior

Aceptable
 Perfeccionamiento Empresarial
 Presupuestadas

Deficiente
 Mixtas

Mal

Hombre de Entidad	Provincia	Ministerio
Almacenes Universal	Ciudad Habana	CIMEX
TRD Mariposa	Cienfuegos	FAR
Hotel SolMelia	Ciego de Avila	MINTUR

Flujos alternos

Flujos alternos 2.a "Mensaje de Error"

Acción del Actor	Respuesta del Sistema
	2a.1 El sistema muestra un mensaje según el error que se produce.
Prototipo de Interfaz	
Poscondiciones	Empresas buscadas y listadas.

Caso de Uso:	Mostrar AC.
Actores:	Auditor.
Resumen:	El caso de uso inicia cuando el actor selecciona una empresa y el código REUP de la misma y se le muestran las acciones de control realizadas a esta.
Precondiciones:	Existencia de acciones de control realizadas.
Referencias	RF9
Prioridad	Secundario.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El auditor seleccionan del menú la opción	2. El sistema muestra el formulario

<p>Mostrar Acciones de Control.</p>	<p>“buscar_ac” con los siguientes campos:</p> <ul style="list-style-type: none"> • Provincia. • Nombre Empresa. • Código REUP.
<p>3. El auditor selecciona la provincia, el nombre y el código REUP de la empresa de la que se desea saber las acciones de control que le han sido realizadas.</p>	<p>4. El sistema muestra un listado con todas las acciones de control realizadas a la empresa seleccionada, con los siguientes datos:</p> <ul style="list-style-type: none"> • Tipo de Acción de Control. • Fecha Inicio Planificada. • Fecha de Fin Planificada. • Fecha Inicio Real. • Fecha de Fin Real.

Criterio de Búsqueda

Nombre de la Empresa

Código REUP

Provincia:

Tipo de AC	F/I Planificada	F/T Planificada	F/I Real	F/T Real
AG	14/1/2007	21/1/2007	16/1/2007	30/1/2007
VSC	3/3/2007	26/3/2007	3/3/2007	20/3/2007
AE	21/4/2007	9/5/2007	26/4/2007	10/5/2007

Flujos Alternos

Flujo Alterno 4.a “Mensaje de Error”

Acción del Actor	Respuesta del Sistema
	4a.1 El sistema muestra un mensaje según el error correspondiente.

Prototipo de Interfaz

Poscondiciones	Acciones de control realizadas a una empresa, mostradas.
-----------------------	--

Caso de Uso:	Mostrar Modelos.
Actores:	Auditor.
Resumen:	El caso de uso inicia cuando el actor selecciona la opción de Mostrar Modelos y aparecen una serie de opciones para que el actor seleccione el que desea ver.
Precondiciones:	Existencia de la Orden de Trabajo
Referencias	RF10
Prioridad	Secundario.

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El auditor selecciona del la opción Mostrar Modelos.	2. El sistema muestra la opción para escoger los siguientes modelos: <ul style="list-style-type: none"> • “PHD.” • “Deficiencias.”
3. El auditor selecciona la opción que quiera ver.	4. El sistema muestra el modelo con los datos correspondientes a la opción seleccionada.

Deficiencias Detectadas en Acciones de Control



No	Entidad Controlada	Contratación	Inventarios	Combustibles	Activos Fijos	Inversiones	Efectivo	Facturación	Nominas
1	CAP Matanzas	x	x	x		x	x		x
2	UCAI Holguin	x		x	x		x	x	
3	DAG		x	x	x		x	x	
4	CAP Tunas	x		x	x	x	x	x	x

Modelo de Presunto Hecho Delictivo



Unidad:		Nombre:	
Datos de la Acción de Control:			
Organismo: MINTUR	Orden de Trabajo: 45		
Subordinación: Cadena Caribe	Tipo de AC: VSC		
Empresa: Hotel Jagua	Fecha de Inicio: 14/2/2008		
Provincia: Cienfuegos	Fecha de Terminación: 29/2/2008		
Hechos Delictivos:			
Caracterización de los presuntos hechos delictivos: Falta de recursos en el almacén.			
Afectación Económicas: 100cuc		Afectación Económicas:	
Denuncia:			
No Expediente: 8954	Lugar: Cienfuegos	Fecha: 1/3/2008	

Flujos Alternos

Flujo Alternativo 4.a "Mensaje de Error"

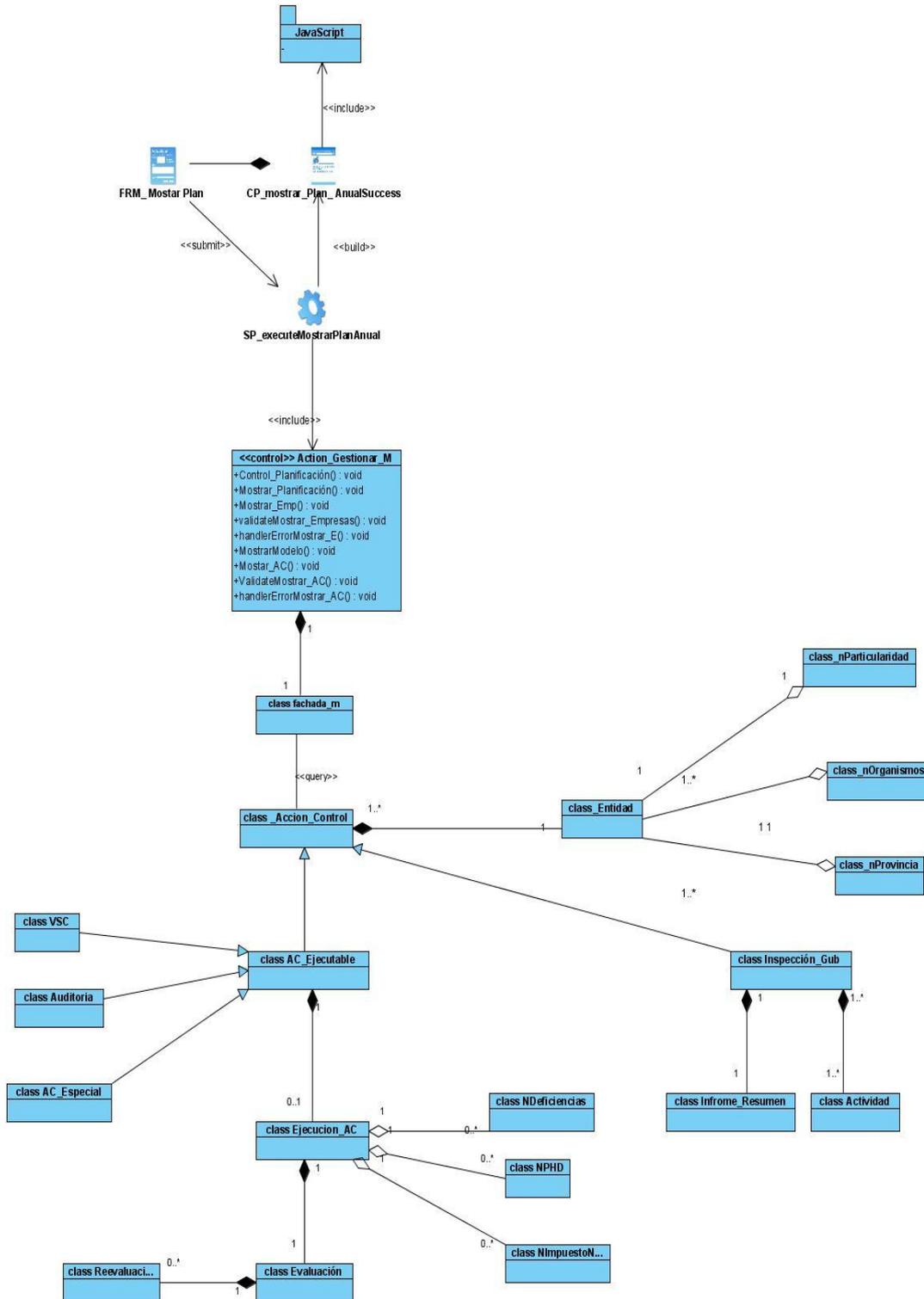
Acción del Actor	Respuesta del Sistema
	4a.1 El sistema muestra un mensaje según el error que corresponda.

Prototipo de Interfaz

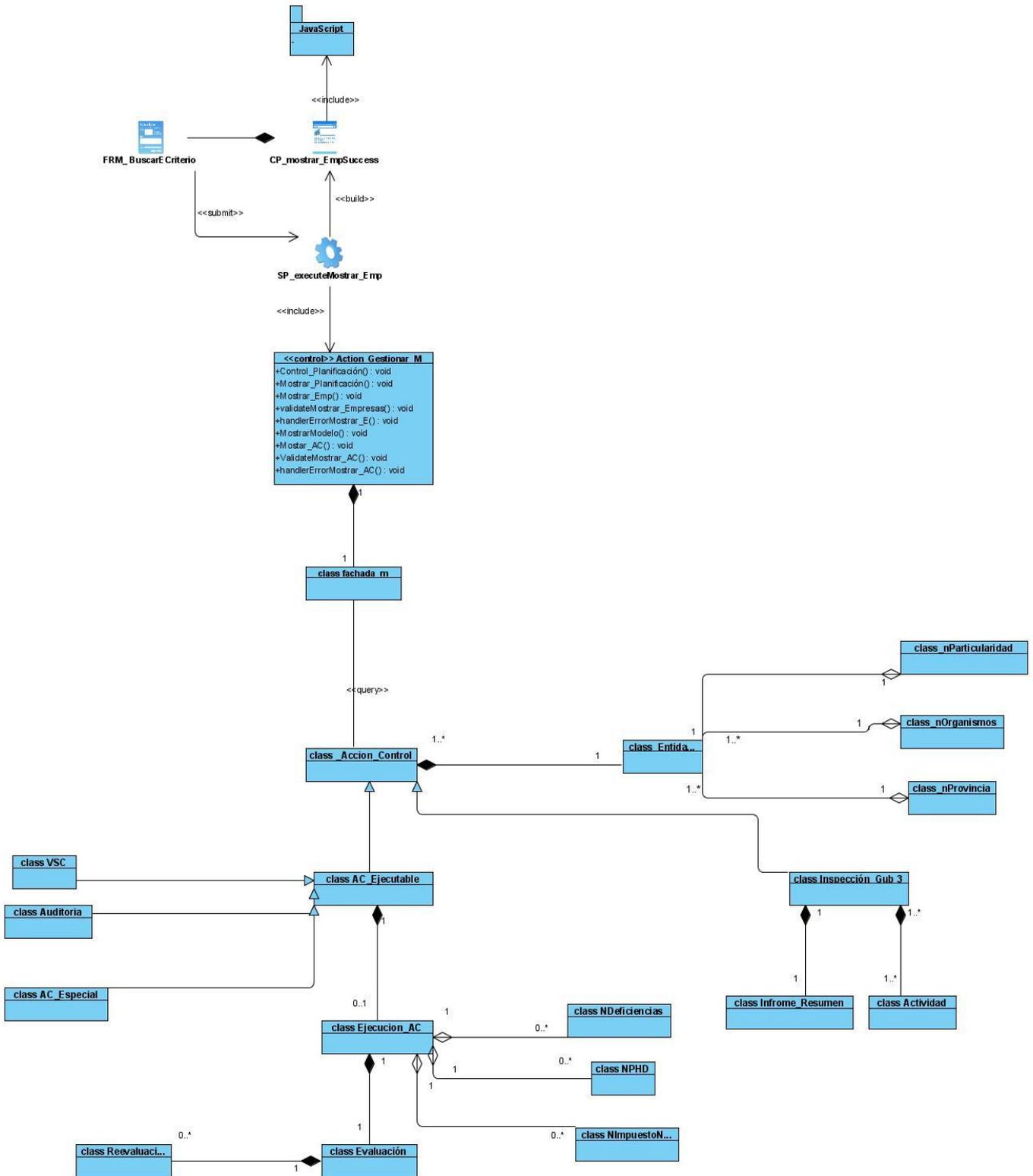
Poscondiciones	Modelos generados por la Orden de Trabajo mostrados.
-----------------------	--

Anexo 3: Diagramas de Diseño.

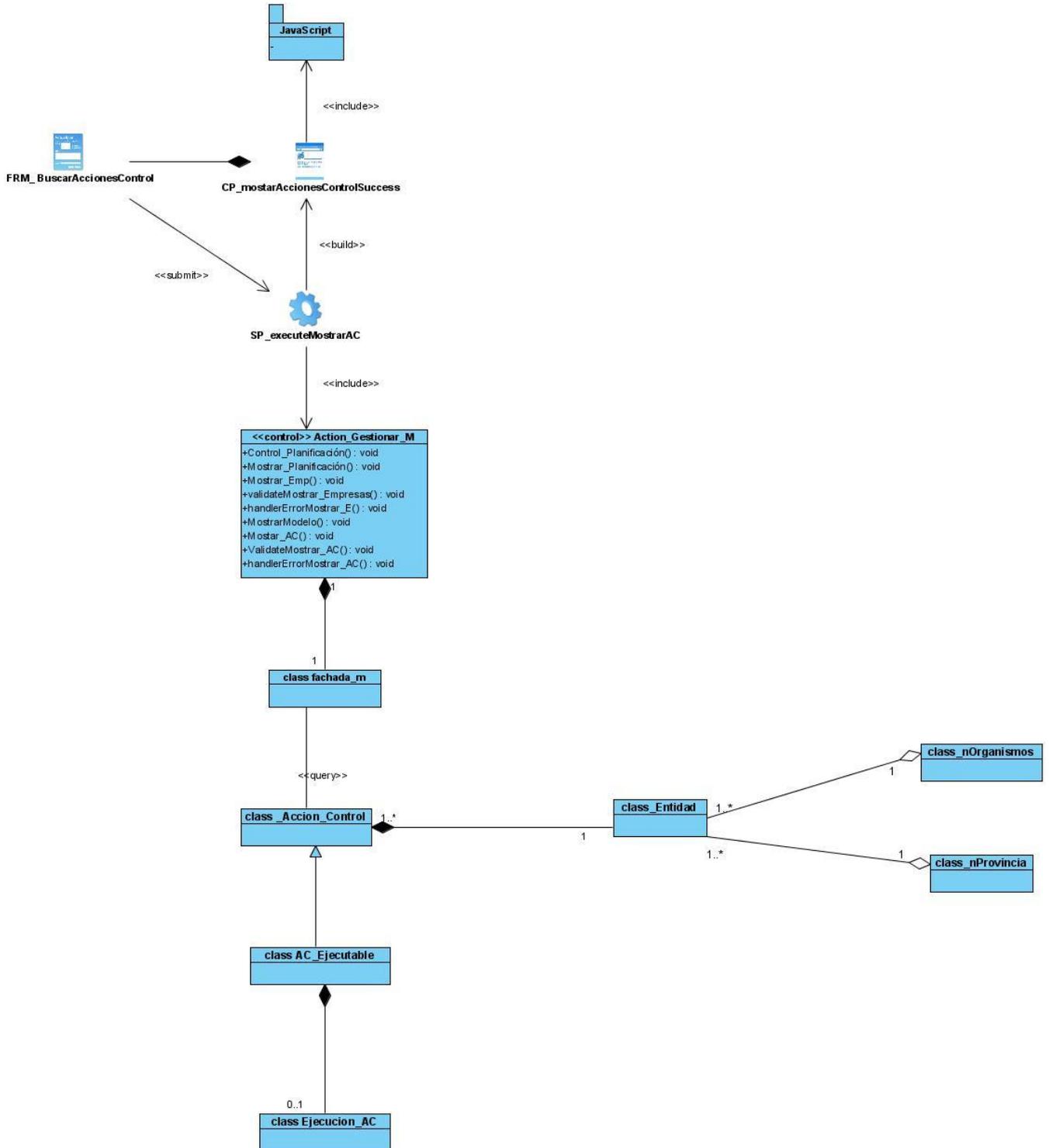
CUS Elaborar Plan Anual.



CUS Mostrar Empresas:



CUS: Mostrar Acción de Control:



CUS Mostrar Modelos:

