

Universidad de las Ciencias Informáticas "Facultad 2"



Título: "Sistema de Registro y Control de Llamadas Telefónicas"

Trabajo de Diploma para optar por el título de
Ingeniero Informático

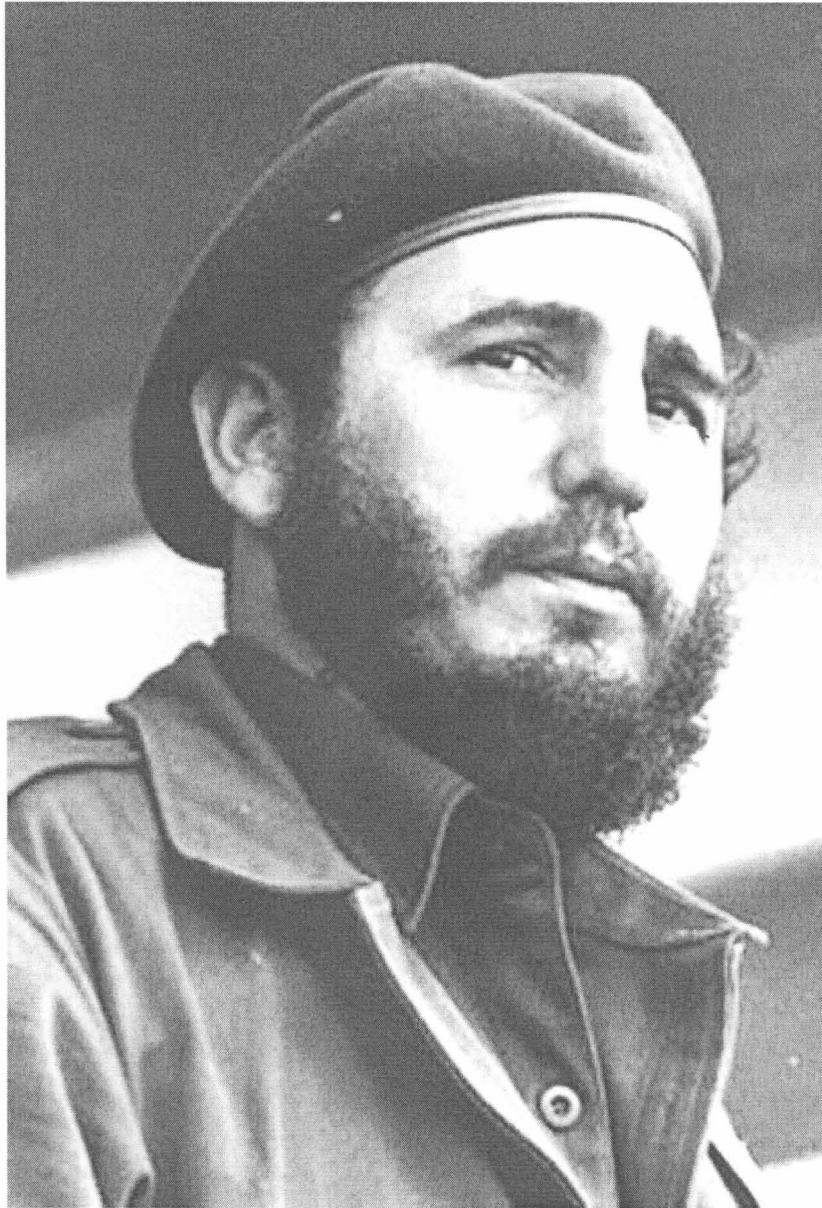
Autor(es): "Nairobis Revé Meléndez"
"Anagel González Martin"

Tutor(es):

Ing. Deivis R. Álvarez Mendoza

Co-tutor: Aliennis Mercedes González Hurtado

"Julio 2008"



*“El futuro de nuestro país tiene que ser necesariamente, un futuro de
hombres de ciencia”.*

Fidel Castro Ruz

AGRADECIMIENTOS

AGRADECIMIENTOS

Nairobi:

A mi madre, mi padre y mi tía por siempre apoyarme en todos los instantes de mi vida.

A mi hermana por ser mi mejor amiga y ser mi confidente.

A Virgen y Caridad por ser lo más especial que me ha pasado en la vida.

A mi prima Janet, porque a pesar de la distancia siempre me ha apoyado mucho.

A Yaraitza Díaz Pérez, que a pesar de todo siempre será para mí como otra más de mis hermanas.

A Liudmila Guzmán González, por el amor y la amistad incondicional que me ha brindado desde los primeros días.

A Anagel González Martín, por todo el apoyo y la fuerza que me ha dado siempre.

A Yudenia de la Rosa Simón, mi otra hermanita dentro de la UCI.

A Raudel José Rosquete Martínez, Reysel Elpidio Arencibia Viamontes, Eleandro Emilio Pérez Acuña, Nayansi Castellón Otero, Ivelis Santos Ochoa; por ser los mejores amigos del mundo, por haberme soportado todos estos años y por hacerme saber que puedo contar con ellos.

Gracias por haber hecho de estos cinco años los más especiales de mi vida.

Anagel:

A mi mamá por todo su amor y cariño y por ser la luz que me ilumina siempre; a mi papá por ser mi ejemplo, guía y por todo su amor y dedicación; a ellos por todo lo que representan en mi vida.

A mis abuelos que tanto quiero y que me han ayudado siempre.

A mis tíos que siempre han estado en mi vida y en especial a mi tía Ana María que siempre me ha querido como su propia hija y que tantos consejos y amor me ha brindado.

A mis primos Luis Fernando, Darien, Arielito y a Rigoberto, Yudiht y Patricia por ser mis amigos siempre y apoyarme en todo.

A mi tío Justo, a Cándida y Norberto por aceptarme y ayudarme.

AGRADECIMIENTOS

A mi mejor amigo Raudel Rosquete, el que siempre me ha brindado su amistad incondicional y al que le voy a agradecer siempre por sus consejos y amor.

A mis amigos y hermanas que han echo de todos estos años los mejores de mi vida y que me han enseñado que amistad es mas que una palabra es estar en lo bueno y en lo malo, es no decir no sino decir te ayudo y estaré ahí para ti siempre, a ellos les agradezco de corazón, Nairobis, Nayansi, Yaraitza, Liudmila, Yamilka, Ivelis, Yudenia, Reysel, Eleandro, Michel, y a Jenely y Ana Maria por estar tantos años en mi vida y aceptarme tal como soy y darme su seguridad y confianza en mi.

Y a todas las personas que de una forma u otra han ayudado en la realización de este trabajo y a mi tutor por su labor todos estos meses

DEDICATORIA

DEDICATORIA

Nairobi:

A mi abuela, porque siempre estará conmigo.

A mi madre, por darme vida y por siempre apoyarme en todo.

A mi padre, por ser mi fuente de inspiración.

A mi tía, por ser mi segunda madre.

Anagel:

A mi mamá, por toda la fuerza y el valor y por guiarme siempre en la vida.

A mi papá, por escucharme y quererme incondicionalmente.

A mi familia, por ayudarme a ser mejor cada día y ser mi seguridad siempre.

A Fidel Castro, por ser mi ejemplo.

RESUMEN

RESUMEN

Debido a la gran cantidad de personal que alberga la Universidad de las Ciencias Informáticas, se ha hecho necesario que exista en cada departamento o local una extensión telefónica, provocando que el flujo telefónico dentro de este centro sea muy grande.

Producto a este inmenso flujo que existe, la Dirección de Gestión Tecnológica tuvo la necesidad de automatizar los procesos de archivar toda la información acerca de las llamadas que se realizan a cualquier teléfono que se encuentre en el interior de este centro de estudios o proveniente de este.

Este trabajo de diploma consiste en la creación de un Sistema para el registro y control de llamadas generales de la planta telefónica de la Universidad de las Ciencias Informáticas.

Los procesos para el registro, control y administración de llamadas generales de la planta telefónica de la UCI eran llevados a cabo por la aplicación Telmonitor; software que brindaba información que previamente era registrada acerca de las llamadas telefónicas que entraban y salían de la universidad. Actualmente la aplicación no está cumpliendo con los objetivos iniciales de su creación, debido a que no maneja la información de la manera debida.

El objetivo principal de este trabajo es crear un sistema fácil de entender y de usar para los usuarios y que muestre una interfaz amigable, además que permita gestionar los usuarios que van a interactuar con la aplicación, así como brindar una serie de reportes referentes a las llamadas telefónicas efectuadas.

PALABRAS CLAVE

Las palabras claves que ayudan a entender este documento son: llamadas telefónicas, tráfico de llamada, control, registro.

TABLA DE CONTENIDOS

TABLA DE CONTENIDOS

AGRADECIMIENTOS	I
DEDICATORIA	III
RESUMEN	IV
INTRODUCCIÓN	1
CAPITULO 1: Fundamentación Teórica	6
1.1 Introducción.	6
1.2 Sistemas de Registro y Control de Llamadas Telefónicas.....	6
1.2.1 Sistemas de Registro y Control de Llamadas en el mundo.	7
1.2.1.1 Sistema de Control y Ahorro Telefónico "SISAT - WIN".	7
1.2.1.2 Comunicaciones en la Universidad Tecnológica de Pereira.....	8
1.2.2 Sistemas de Registro y Control de Llamadas en la UCI.....	8
1.3 Fundamentación Tecnológica.....	8
1.3.1 Aplicación Web.....	8
1.3.2 Arquitectura de software.....	9
1.3.2.1 Cliente-Servidor.....	10
1.3.3 Protocolos estándares.....	11
1.3.4 Lenguajes de programación.	12
1.3.4.1 Lenguaje del lado del servidor.....	12
1.3.4.1.1 PHP.....	12
1.3.4.1.2 ASP.....	13
1.3.4.1.3 ¿Por qué usar PHP?	13
1.3.4.2 Lenguaje del lado del cliente.	14
1.3.4.3 Java Script.....	14
1.3.5 CSS	14
1.3.6 XHTML.	15
1.3.7 Herramientas de desarrollo.	15
1.3.7.1 Zend Studio.....	15
1.3.7.2 Dreamweaver.....	16
1.3.8 Sistemas gestores de Base de datos (SGBD).....	16
1.3.8.1 MySQL.	16
1.3.8.2 PostgreSQL.....	17

TABLA DE CONTENIDOS

1.3.9 Servidor Web.....	17
1.3.10 Apache	18
1.3.11 Metodología de desarrollo. RUP.....	18
1.3.11.1 Extreme Programing (XP).	18
1.3.11.2 Proceso Unificado de desarrollo de software (RUP).	19
1.3.11.3 ¿Por qué usar RUP?	21
1.3.12 Lenguaje de Modelación. UML.....	21
1.3.13 Herramientas.....	22
1.3.13.1 Herramienta CASE. Visual Paradigm.....	22
1.3.13.2 Framework.	22
1.3.13.2.1 Symphony.	22
1.3.13.2.2 CodeIgniter.....	23
1.3.14 Patrón de Diseño. Modelo Vista Controlador (MVC).	23
1.4 Conclusiones.....	24
CAPITULO 2: Características del Sistema.....	25
2.1 Introducción.....	25
2.2 Objeto de estudio.	25
2.2.1 Problema y situación problemática.....	25
2.2.2 Objeto de automatización:	26
2.2.3 Información que se maneja.	27
2.3 Propuesta del sistema.....	27
2.4 Modelo del negocio.	28
2.4.1 Descripción de los actores y trabajadores del negocio.....	29
2.4.2 Diagrama de casos de uso del negocio.....	29
2.4.3 Descripción textual de los casos de uso del negocio.	30
2.4.4 Reglas del negocio:	30
2.4.5 Diagramas de actividad.	30
2.4.6 Modelo de objetos.	30
2.5 Relación de los requerimientos.....	30
2.5.1 Requerimientos funcionales:	30
2.5.2 Requerimientos no funcionales.	35
2.6 Modelo de casos de uso del sistema.	37
2.6.1 Definición de los actores del sistema.	37
2.6.2 Diagrama de casos de uso del sistema.....	38

TABLA DE CONTENIDOS

2.6.3 Descripción de los casos de uso.	38
2.7 Conclusiones:.....	38
CAPITULO 3: Análisis y Diseño del Sistema.....	39
3.1 Introducción.....	39
3.2 Análisis del sistema.....	39
3.2.1 Diagramas de clases del análisis.	39
3.3 Diseño.....	42
3.3.1 Diagramas de interacción.....	42
3.3.2 Diagrama de clases WEB del diseño.	43
3.4 Diseño de la base de datos.....	47
3.4.1 Modelo lógico de datos (Diagrama de clases persistentes).	47
3.4.2 Modelo físico de datos (Modelo de datos).....	47
3.5 Conclusiones.....	48
CAPITULO 4: Implementación y Prueba	49
4.1 Introducción.....	49
4.2 Diagrama de despliegue.	49
4.3 Diagrama de componentes.	50
4.4 Conclusiones.....	52
CAPITULO 5: Estudio de la Factibilidad	53
5.1 Introducción.....	53
5.2 Planificación.....	53
5.2.1 Cálculo de puntos de casos de uso sin ajustar.	53
5.2.1.1 Factor de peso de actores sin ajustar (UAW).....	54
5.2.1.2 Factor de peso de los casos de uso sin ajustar (UUCW).	54
5.2.2 Cálculo de puntos de casos de uso ajustados.	54
5.2.2.1 Factor de complejidad técnica.....	55
5.2.2.2 Factor ambiente.	55
5.2.3 Estimación de esfuerzo.....	56
5.2.3.1 Calculando el factor de conversión.....	56
5.2.4 Calcular el esfuerzo de todo el proyecto	57
5.3 Beneficios tangibles e intangibles.	58
5.4 Análisis de costo.	59
5.5 Conclusiones.....	59
CONCLUSIONES.....	60

TABLA DE CONTENIDOS

RECOMENDACIONES.....	61
REFERENCIAS BIBLIOGRAFICAS	62
BIBLIOGRAFIA.....	64
ANEXOS.....	65
GLOSARIO DE TERMINOS.....	89

INDICE DE FIGURAS

INDICE DE FIGURAS

Figura 1 Logotipo de PHP.....	12
Figura 2 Logotipo de JavaScript.....	14
Figura 3 Logotipo de CCS.....	14
Figura 4 Metodología XP.....	19
Figura 5 Relación entre flujo o disciplinas de trabajo y fases en RUP.....	20
Figura 6 Diagrama de CU del negocio, solicitar información.....	29
Figura 7 Diagrama de CU del Sistema.....	38
Figura 8 Diagrama de Clase del Análisis CU Autenticar Usuario.....	40
Figura 9 Diagrama de Clases del Análisis CU Gestionar Operador.....	40
Figura 10 Diagrama de Clases del Análisis CU Visualizar Reporte de Costo.....	41
Figura 11 Diagrama de Clases del Análisis CU Visualizar Reporte de Tráfico.....	41
Figura 12 Diagrama de Clases del Análisis CU Crear Informe Resumen.....	42
Figura 13 Diagrama de Clases del Diseño CU Autenticar Usuario.....	43
Figura 14 Diagrama de Clases del Diseño CU Gestionar Operador.....	44
Figura 15 Diagrama de Clases del Diseño CU Vizualizar Reporte de Costo.....	44
Figura 16 Diagrama de Clase del Diseño CU Visualizar Reporte de Tráfico.....	45
Figura 17 Diagrama de Clase del Diseño CU Crear Informe Resumen.....	45
Figura 18 Diagrama de Clase del Diseño CU Gestionar Cable.....	46
Figura 19 Diagrama de Clase del Diseño CU Hallar Facilidades de Cables.....	46
Figura 20 Diagrama de Clases Persistentes.....	47
Figura 21 Diagrama Entidad Relación.....	48
Figura 22 Diagrama de Despliegue.....	49
Figura 23 Diagrama de Componentes CU Autenticar.....	50
Figura 24 Diagrama de Componentes CU Gestionar Operador.....	50
Figura 25 Diagrama de Componentes CU Visualizar Reporte de Tráfico.....	51
Figura 26 Diagrama de Componentes CU Visualizar Reporte de Costo.....	51
Figura 27 Diagrama de Componentes CU Crear Informe Resumen.....	51
Figura 28 Diagrama de Componentes CU Gestionar Cables.....	52
Figura 29 Diagrama de Componentes CU Hallar Facilidades de Cables.....	52

INTRODUCCION

INTRODUCCIÓN

Desde la transformación del mono en hombre la necesidad de hablar con los demás se hizo evidente; era preciso expresar sus sentimientos, proyectos, y hasta el propio peligro. Fue entonces cuando tuvo que buscar una forma para comunicarse y expresarse. El hombre primitivo descubrió que era capaz de emitir sonidos inarticulados, y con el tiempo vio que al combinarlos podía expresar palabras e ideas. Descubrió también que a través de los dibujos podía exponer parte de su realidad cotidiana. Estos dos factores combinados, propiciaron el desarrollo comunicativo del ser humano.

La búsqueda constante para extender la comunicación lo llevó a crear instrumentos poderosos y veloces que lo ayudaran a comunicarse con otros seres humanos que habitaban en regiones alejadas. Surge de esta forma la comunicación a través de señales de humo, los métodos de escritura jeroglífica, el alfabeto, el papel y la imprenta, entre otros muchos medios que propiciaron el desarrollo de la comunicación hasta alcanzar los más recientes, entre los que podemos citar el teléfono, el cine, la radio y la televisión, los que se han perfeccionado gracias al desarrollo de las telecomunicaciones y la informática.

Entre los medios de comunicación destaca la telefonía, gracias a la cual las comunicaciones interpersonales se realizan de forma instantánea.

Desde su surgimiento, atravesando todas sus etapas de desarrollo y hasta la actualidad, la informática ha estado estrechamente relacionada con la telefonía, tanto que hoy en día es imposible separar una rama de la otra. En la actualidad el uso de las nuevas tecnologías ha aumentado gradualmente la calidad de vida de las personas y ha hecho que además de la comunicación presencial, la comunicación telefónica se haya impuesto como elemento básico de la vida.

El mundo de hoy se encuentra inmerso en un constante desarrollo de las tecnologías, y nuestro país no ha sido la excepción dentro de esta revolución tecnológica. A pesar de las grandes limitaciones que padece, el gobierno cubano siempre ha apoyado y respaldado las investigaciones que se realicen en esta rama, con lo que se ha logrado que las técnicas y tecnologías de comunicación no se queden obsoletas.

INTRODUCCION

ETECSA, es la empresa que en Cuba se encarga de las comunicaciones; esta institución tiene a su cargo la instalación telefónica en todos los rincones del país y se encuentra en la actualidad enfrascada en la realización de productos que puedan adaptarse fácilmente a los avances tecnológicos que surgen con gran rapidez en el mundo. Con este fin se han creado una serie de programas en los que se preparan profesionales con conocimientos para ayudar a la nación en la investigación en esta rama de la economía.

Uno de esos proyectos es precisamente la Universidad de Ciencias Informáticas (UCI) en la que se preparan profesionales con conocimientos informáticos que pueden ayudar en este tipo de investigaciones.

La UCI alberga a más de diez mil estudiantes y en ella trabajan alrededor de dos mil personas. Debido a este gran flujo de personal y a la gran cantidad de inmuebles con que cuenta esta instalación que se dividen en edificios de residencia y en departamentos en los que se encuentran distribuidas todas las demás áreas del comercio se ha hecho una necesidad que cada local cuente con un teléfono a su disposición, lo que trae consigo que el flujo telefónico en esta institución sea elevado.

Esto hace que surja la necesidad de realizar un control del mismo, el que se realiza en el Subnodo 1. El control se basa en archivar la información de las llamadas que se realizan a través de la pizarra telefónica dentro de la universidad. Para realizar esta operación es necesaria una serie de informaciones acerca de las llamadas telefónicas entre las que se encuentran la duración de la llamada, el costo que importa y el número de destino o el de proveniencia. Este control de las llamadas permite a los trabajadores realizar un estudio de cómo se comporta el flujo telefónico dentro de la universidad para tomar providencias con el fin de disminuir al máximo las congestiones en las líneas y evitar que el servicio telefónico se caiga.

Para facilitar este trabajo fue creado el Telmonitor, software que en un principio resolvió los problemas que se planteaban los trabajadores de este local. En la actualidad este sistema no cumple con los objetivos iniciales de su creación, la información no es almacenada de la mejor manera y han surgido una serie de necesidades de informaciones que es incapaz de brindar, por lo que se plantea la necesidad de crear un nuevo sistema que resuelva los problemas de información que se tienen con el fin de mejorar el trabajo en este sector.

INTRODUCCION

Luego de un análisis del mismo y tomando en cuenta la **situación** actual, surge el siguiente **problema científico**: ¿Cómo facilitar el registro y control de la información de las llamadas generales en la planta telefónica UCI?

El problema que se plantea se encuentra enmarcado en el siguiente **Objeto de Estudio**: Proceso que facilita el registro y control de la información de las llamadas generales de la planta telefónica en la UCI. Delimitando como **Campo de Acción**: La información de las llamadas generales en la planta telefónica de la UCI.

Para darle solución al problema planteado se establece como **Objetivo General**: Desarrollar una aplicación para facilitar el registro y control de las llamadas generales en la planta telefónica en la Universidad de Ciencias Informáticas.

Para facilitar el cumplimiento del objetivo trazado, se establecen los siguientes **Objetivos Específicos**:

- Caracterizar el proceso de la información de llamadas generales de la UCI.
- Definir los procesos vinculados a la información de las llamadas generales en la planta telefónica de la UCI.
- Caracterizar antecedentes del software en cuestión.
- Caracterizar herramientas a usar.
- Realizar el proceso y desarrollo de software.

Para dar cumplimiento a los objetivos planteados se han planificado las siguientes **Tareas de Investigación**:

- Realizar un estudio a nivel mundial, a nivel de país y en la UCI acerca del proceso de información de llamadas.
- Seleccionar el proceso de la información de las llamadas generales de la UCI.
- Estudio de las tecnologías vinculadas a la gestión de la información existentes en la actualidad.
- Seleccionar el proceso de desarrollo de software a utilizar.
- Selección de las herramientas idóneas que se utilizarán en el desarrollo de la aplicación Web.

La **Idea a Defender** que se propone es: Con el desarrollo de una aplicación Web se facilitará el registro y control de las llamadas telefónicas en la planta telefónica en la Universidad de Ciencias Informáticas.

INTRODUCCION

Como parte de la investigación se llevaron a cabo los siguientes **Métodos Científicos**:

Métodos Teóricos:

- Análisis Histórico-Lógico:

A lo largo de la investigación se propone analizar cómo se realizaba el trabajo a través del sistema anterior, cuáles eran las funcionalidades con las que contaba y los principales problemas que presentaba. Con esta tarea pudimos determinar las funcionalidades que la aplicación podría tener y los aspectos relevantes para que el nuevo sistema no corra la misma suerte que el anterior.

Métodos Empíricos:

- Observación:

Mediante la observación del trabajo que realizaban los trabajadores se pudo constatar las necesidades que existían en cuanto a información a la cual necesitaban tener acceso rápido y que les costaba trabajo conseguir. Esto proporcionó una serie de ideas sobre las funcionalidades que debía poseer el sistema a realizar y dio la posibilidad de esbozar como sería el ambiente de la aplicación.

El trabajo cuenta con 5 capítulos, los que se encuentran estructurados de la siguiente manera:

Capítulo 1: Fundamentación teórica.

Se tratan los temas que constituyen la fundamentación teórica de la investigación a realizar. Consta de un estudio de los sistemas que han sido o son usados para la gestión de una planta telefónica tanto en Cuba como fuera del país. El objetivo de este estudio es que se tomen como ejemplo para la realización de la aplicación a desarrollar. Además se hace un análisis de las herramientas que se usarán para el desarrollo de la aplicación, las ventajas que poseen y las razones por las que se escogieron estas y no otras.

Capítulo 2: Características del sistema.

Incluye una breve descripción del sistema. En este capítulo se modelan los procesos del negocio identificando casos de uso, actores, trabajadores y entidades que intervienen en este. También se determinan los requisitos funcionales y no funcionales que establecerán las características que cumplirá el sistema en cuestión.

Capítulo 3: Análisis y diseño del sistema:

INTRODUCCION

Se describen las clases del análisis y del diseño. Se realizan además los diagramas de interacción y se muestra el modelo entidad relación y se realiza una breve descripción de las tablas que conforman la base de datos.

Capítulo 4: Implementación.

Este capítulo aborda las características de la implementación del sistema. A través de él se muestran los diagramas de despliegue y de componentes.

Capítulo 5: Estudio de factibilidad.

Realiza un análisis de los costos y beneficios que trae implicada la realización del proyecto.

CAPITULO 1: FUNDAMENTACION TEORICA

CAPITULO

1

FUNDAMENTACIÓN TEÓRICA

1.1 Introducción.

A lo largo de este capítulo se hace un análisis de los sistemas que han surgido en Cuba y en el mundo en los que se registra y controla la información de llamadas telefónicas brindada por pizarras. También se hace un análisis de las tecnologías que existen alrededor de todo el mundo y que pueden ser utilizadas para realizar este tipo de sistemas; haciéndose énfasis en las propuestas a usar en la elaboración de la aplicación en particular.

1.2 Sistemas de Registro y Control de Llamadas Telefónicas.

La evolución constante de las tecnologías de la informática es uno de los pilares fundamentales para la superación y continuo desarrollo de un país. El teléfono es uno de esos medios que más transformaciones obtiene y debido a la gran demanda existente es necesario tener en cuenta cierta información de este flujo telefónico y específicamente en centro de redes locales. En la actualidad existen softwares que permiten tener el control de las líneas telefónicas, estos softwares son los llamados Sistemas de Registro y Control de Llamadas Telefónicas, estos permiten conocer la facturación de las líneas y lo que ocurre en estas.

Un Sistema para el Registro y Control de Llamadas es una aplicación que se realiza con el objetivo de controlar el flujo de llamadas que se realizan dentro de un área determinada, permitiendo cubrir con éxito toda la información referente a las llamadas telefónicas que se registra en la pizarra telefónica. La aplicación de este sistema posibilita que se evalúe la carga telefónica real del lugar donde se aplica.

Su funcionamiento radica en recibir y procesar toda la información referente a las llamadas que se obtienen de la pizarra telefónica y son almacenadas en una base de datos, con el fin de informar como se comporta el flujo telefónico dentro de esta área y cual es el valor que importan las llamadas que se realizan a través de su pizarra.

CAPITULO 1: FUNDAMENTACION TEORICA

El principal objetivo que se persigue con este tipo de sistema es ver donde ocurre un mayor flujo telefónico, ver la hora en la que ocurre mayor cantidad de llamadas; todo esto para poder tomar medidas y lograr un mejor servicio.

1.2.1 Sistemas de Registro y Control de Llamadas en el mundo.

1.2.1.1 Sistema de Control y Ahorro Telefónico "SISAT - WIN".

El principal objetivo de este software es el control y el ahorro telefónico. Controla las llamadas entrantes y salientes de la empresa controlando el tiempo de duración, destino y costo real de cada llamada. Por medio de este control, el software ha posibilitado la reducción de gastos telefónicos en las empresas usuarias.

Este software se encuentra programado en Visual Basic y está diseñado para recibir y procesar toda la información de las llamadas salientes y entrantes de las centrales telefónicas, las que tarifica y almacena en una base de datos. Con esta información se realiza una serie de reportes.

Está creado bajo una arquitectura cliente-servidor permitiendo el trabajo con diferentes bases de datos.

Se compone de dos módulos:

Administración telefónica: Permite el manejo de múltiples tipos de servicios de telefonía como los celulares. Permite el manejo de tarifas telefónicas por empresas que brindan servicios de telefonía con diferentes horarios, ya sea por horas o por días de la semana.

Registro de llamadas: Permite registrar y procesar la información que brinda la pizarra telefónica a través de un cable serial conectado a la computadora. Este módulo se activa de forma automática cuando recibe un registro telefónico.

Este software presenta diversas ventajas respecto a los demás sistemas con iguales funcionalidades y es que permite la accesibilidad a diferentes Base de Datos como: "SQL – Server 6.5" y "Access", así como a tablas "DBF". Además del control del acceso de la información de cada departamento.

CAPITULO 1: FUNDAMENTACION TEORICA

1.2.1.2 Comunicaciones en la Universidad Tecnológica de Pereira.

Este software sirve de apoyo en el área de comunicaciones de este centro de estudios. Consta de dos módulos:

- Registro de llamadas telefónicas: Este módulo consta de dos submódulos: el primero, Registro de información de la planta telefónica, registra automáticamente todas las llamadas que se realicen a través de la central telefónica y el segundo, Registro de llamadas a través del conmutador, registra manualmente las llamadas solicitadas al conmutador.
- Control de facturas telefónicas: Esta dividido en secciones:
 - Archivo: Permite importar facturas telefónicas descargadas de Internet.
 - Teléfonos Extensiones, Facturas: Adiciona y modifica facturas y conceptos.
 - Reportes: Son cuentas de cobro que compara las llamadas personales realizadas con las facturas que han sido previamente importadas y registradas.
 - Ayuda.

1.2.2 Sistemas de Registro y Control de Llamadas en la UCI.

En la universidad existe un sistema que se encarga de registrar y controlar la información de las llamadas telefónicas que se realizan a través de la pizarra. Este sistema es el Telmonitor y se explica mas adelante.

1.3 Fundamentación Tecnológica.

1.3.1 Aplicación Web.

Desde que surgió el mundo Web, se ha empezado a implementar sitios dinámicos y estáticos que le han dado vida a los diferentes servicios Web, y es precisamente a lo que llamamos aplicaciones Web, a toda aquella programación especial, que tiene a su vez asociada un base de datos y permite de esta manera mayor interacción con el usuario.

CAPITULO 1: FUNDAMENTACION TEORICA

Podemos clasificar como aplicación Web al sistema informático que usan los usuarios para acceder a un servidor web. En la actualidad su uso se ha generalizado pues cuenta con la facilidad de verse en cualquier lugar sin necesidad de instalar ningún software, además de que son independientes del sistema operativo. Y esta es una de sus grandes ventajas, la comodidad de poder acceder desde cualquier parte del mundo.

Las aplicaciones web en su mayoría constan de tres capas: el navegador web que es el que le lleva la información a los usuarios, una base de datos que archiva toda la información que puede ser usada por estos y una capa de gestión de información que es la que conecta las dos anteriores. [1]

En la actualidad para el desarrollo de aplicaciones web se usan una serie de lenguajes entre los que se encuentran: PHP, Java, Perl, HTML y XML.

Hemos decidido crear una aplicación Web para solucionar el problema que se nos planteaba porque de esta manera se puede acceder a ella desde cualquier lugar de la universidad, procurando siempre que solo acceda a ella el personal autorizado a interactuar con esta información.

1.3.2 Arquitectura de software.

Una arquitectura es un entramado de componentes funcionales que aprovechando diferentes estándares, convenciones, reglas y procesos, permite integrar una amplia gama de productos y servicios informáticos, de manera que pueden ser utilizados eficazmente dentro de una organización. [2]

Se puede decir que la Arquitectura de Software no es más que el diseño de más alto nivel de la estructura de un sistema, programa o aplicación y tiene la responsabilidad de:

- Definir los módulos principales
- Definir las responsabilidades que tendrá cada uno de estos módulos
- Definir la interacción que existirá entre dichos módulos:
- Control y flujo de datos
 - Secuenciación de la información
 - Protocolos de interacción y comunicación

CAPITULO 1: FUNDAMENTACION TEORICA

- Ubicación en el hardware. [3]

La arquitectura de software establece fundamentos para que los desarrolladores trabajen siguiendo una línea común con el fin de alcanzar los objetivos que se han propuesto. Su principal objetivo es aportar elementos que ayuden a la toma de decisiones y, al mismo tiempo, proporcionar conceptos y un lenguaje común que permitan la comunicación entre los equipos que participen en un proyecto.

1.3.2.1 Cliente-Servidor

La arquitectura cliente – servidor se basa en separar los datos en una PC que se denominaría servidora a la que tendrán acceso uno o varios usuarios a través de una página cliente o del propio servidor.

Esta arquitectura es una relación de procesos ejecutándose en máquinas separadas, pues el servidor es que va a proveer los servicios que va a consumir el cliente.

En este tipo de arquitectura las aplicaciones se dividen de forma que el servidor contiene la parte que debe ser compartida por varios usuarios, y el cliente es quien gestiona lo particular de cada usuario.

Los clientes realizan generalmente funciones como el manejo de la interfaz de usuario y la captura y validación de los datos de entrada. Por su parte el servidor controla los accesos a bases de datos compartidas y el enlace a otras redes.

Esta manera de trabajar tiene sus ventajas; por ejemplo, al tener toda la información almacenada en un mismo sitio se logra que todos los usuarios trabajen con una misma información, la que al ser modificada en el servidor llegaría de igual manera a todos los clientes. También resulta ventajosa a la hora de establecer la seguridad, pues solo tendrían que establecerse las restricciones de uso en el servidor y ya se aplicaría a cada cliente que use esta información. Al emplear este tipo de arquitectura se puede mejorar el rendimiento de la red, pues se evita el tener que trasladar un gran flujo de información a través de esta, el servidor es quien almacena los datos y procesa la información haciendo llegar a la máquina cliente solo la información que necesita.

Son por todas estas razones que se ha utilizado este tipo de arquitectura para poder obtener mayor rendimiento en la aplicación como también asegurar que una mejor seguridad con cualquier tipo de usuario que se necesite de esta aplicación.

CAPITULO 1: FUNDAMENTACION TEORICA

1.3.3 Protocolos estándares.

Para lograr la interoperabilidad de las aplicaciones de software se usan los protocolos estándares los que realizan esta tarea independientemente de las propiedades que estas tengan y de los programas que se encuentren instalados.

Los protocolos son reglas de comunicación que posibilitan el flujo de información entre distintas computadoras. Estos pueden estar implementados en hardware, software o una combinación de ambos.

Los protocolos determinan el tipo de comprobación de errores que se va a usar, el método a emplear para la comprensión de los datos, también determinan la forma en la que el dispositivo que envía el mensaje indicara que lo acabo de enviar y como el que lo recibe indicara que lo ha recibido. Esto provoca que para que dos computadoras puedan compartir sus datos deben usar el mismo protocolo de comunicación.

Entre los protocolos más usados podemos destacar:

SOAP: Del ingles Simple Object Access Protocol. Es un protocolo liviano basado en XML para el intercambio de información estructurada en un ambiente descentralizado y distribuido. Es un de los protocolos que s emplea en los servicios web.

UDDI: Del inglés Universal Description, Discovery and Integration. Es una iniciativa industrial enmarcada en el contexto de los servicios web. El objetivo de este protocolo es ser accedido por los mensajes SOAP y dar paso a documentos WSDL.

WSDL: Es el lenguaje usado para la descripción de los servicios web. A decir de la W3C, WSDL es “un formato XML para la descripción de servicios de red, como un conjunto de puntos finales operando sobre mensajes que contienen información orientada a documentos u orientada a procesos”.

HTTP: Es el protocolo que se usa en cada transacción de la red. Define la sintaxis y la semántica que utilizan los elementos software de la arquitectura web para comunicarse. Sigue el esquema petición respuesta entre un cliente y un servidor.

CAPITULO 1: FUNDAMENTACION TEORICA

1.3.4 Lenguajes de programación.

Para la realización de cualquier aplicación en el mundo informático es necesario algún tipo de lenguaje de programación; ya sea el que soporta y carga el sistema operativo de la máquina o de un simple software que se esté utilizando.

Un lenguaje de programación es la herramienta que permite crear softwares y programas, esta es representada por símbolos o caracteres que conforman códigos que serán traducidos por la propia computadora. Si se mira el tipo de arquitectura que se ha elegido para la aplicación podemos concluir que tenemos un tipo de lenguaje de programación tanto del lado del cliente como del lado del servidor.

1.3.4.1 Lenguaje del lado del servidor.

Los lenguajes de lado servidor que son aquellos lenguajes que son reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente en un formato comprensible para él. Este tipo de lenguaje es independiente del cliente por lo que es mucho menos rígido respecto al cambio de un navegador a otro o respecto a las versiones del mismo. [4]

1.3.4.1.1 PHP.



Figura 1 Logotipo de PHP.

PHP es un acrónimo que significa Hipertext Pre-processor. Este lenguaje es una de las mejores opciones a la hora de realizar una página dinámica debido a su gran potencia, facilidad de uso, amplia biblioteca de funciones, coste gratuito y velocidad; que hacen que su presencia esté más difundida en la red. [5]

Este es un lenguaje del lado del servidor, lo que tiene acceso a los recursos que este contenga. Esto también posibilita que sea independiente del navegador, por lo que resulta imprescindible para su correcta ejecución que el servidor con el cual se vaya a trabajar lo soporte. [6]

CAPITULO 1: FUNDAMENTACION TEORICA

También nos ofrece una gran variedad de funciones que nos permiten desarrollar múltiples funcionalidades que van desde enviar un e-mail, subir un archivo (upload), crear una imagen en tiempo de ejecución, interactuar con diversos protocolos de comunicación, interactuar con documentos XML, autenticación, creación dinámica de documentos PDF, entre muchas otras cosas.[7]

Es un lenguaje muy simple y versátil puesto que lo mismo brinda la posibilidad de crear aplicaciones web complejas como de crear un simple script para procesar un formulario.

1.3.4.1.2 ASP.

Las páginas servidoras activas (ASP por sus siglas en inglés), es una tecnología desarrollada por Microsoft para realizar páginas web dinámicas. Son una combinación de texto, etiquetas HTML y secuencias de comandos del servidor Este tipo de lenguaje no necesita ser compilado para ejecutarse. Las páginas ASP son ejecutadas en el servidor antes de ser enviadas al cliente.

Una de las cosas que tiene en contra esta tecnología es que no cuenta con información a disposición que pueda ser accedida por quienes quieren aprenderlo, además solo se limita a la plataforma Windows y el servidor con el que funciona es Internet Information Server (IIS).

1.3.4.1.3 ¿Por qué usar PHP?

Se decidió utilizar PHP como lenguaje del lado del servidor por muchas razones, entre las que se pueden citar:

- Es un lenguaje fácil de aprender.
- Cuenta con una amplia comunidad web lo que resulta favorable a la hora de buscar información.
- Es de código abierto.
- Es rápido a la hora de procesar las páginas.
- Los servidores sobre los que trabaja perfectamente son de código abierto.
- Tiene conexión con la mayoría de los gestores de base de datos entre los que se encuentran: MySQL, PostgreSQL y Microsoft SQLServer.

CAPITULO 1: FUNDAMENTACION TEORICA

1.3.4.2 Lenguaje del lado del cliente.

Son los lenguajes que son comprensibles por el navegador y no necesitan ningún tratamiento para que este los comprenda. Este tipo de lenguaje es totalmente independiente del servidor. Los lenguajes de programación del lado del cliente sirven para proporcionar diversos efectos en las páginas, para dar respuestas a las acciones que realice el usuario dentro de la página y para controlar los datos que se insertan en los formularios.

1.3.4.3 Java Script.

JavaScript

Figura 2 Logotipo de JavaScript

Java Script es un lenguaje del lado del cliente debido a que la carga de procesamiento la soporta el navegador que se convierte en el mayor recurso con que cuenta. Gracias a que es compatible con la mayoría de los navegadores se ha convertido en el lenguaje de programación del lado del cliente más utilizado.

Es usado para crear pequeños programitas encargados de realizar acciones dentro del ámbito de una página web. Su uso se basa fundamentalmente en la creación de efectos especiales en las páginas y la definición de interactividades con el usuario. [8]

Se escogió Java Script para desarrollar la programación del lado del cliente debido a que es un lenguaje muy sencillo y fácil de aprender. Permite la programación orientada a objetos y la utilización de estructuras de datos complejas. [9]

1.3.5 CSS



Figura 3 Logotipo de CCS

CSS son las siglas de Cascading Style Sheets, en español Hojas de estilo en Cascada. CSS, es una tecnología que permite crear páginas web de una manera más exacta. Gracias a las CSS somos mucho más dueños de los resultados finales de la página, pudiendo hacer muchas cosas que no se podían haber logrado solamente con HTML (como incluir márgenes, tipos de letra, fondos, colores) de

CAPITULO 1: FUNDAMENTACION TEORICA

una sola vez para aplicarlos a una cantidad indeterminada de páginas. Este tipo de archivos se pueden guardar externamente y luego se incluirlos en las páginas web, si se cambia algo en estos archivos de manera automática se cambia en todas las páginas que lo utilizan. [10]

1.3.6 XHTML.

Lenguaje Extensible de Marcado e Hipertexto; XHTML en sus siglas en inglés. Es un lenguaje creado para sustituir a HTML para páginas web con el objetivo de corregir la incompatibilidad de algunos elementos de este con ciertos navegadores. Sólo se usa para transmitir la información del documento; dejando el estilo a las CSS y el comportamiento de la página a Java Script.

XHTML consta de reglas imprescindibles para su correcto funcionamiento entre las que se encuentran:

- Todos los elementos deben estar debidamente jerarquizados.
- Todo documento debe estar bien formado.
- Los nombres de las etiquetas deben estar en minúsculas.
- Todas las etiquetas deben cerrarse.
- Los nombres de los atributos deben ir en minúsculas.
- Los valores de los atributos deben ir entre comillas.
- El atributo id reemplaza al atributo name.

1.3.7 Herramientas de desarrollo.

Los IDE de desarrollo son programas que constituyen herramientas para los programadores, pueden ser empleados para un lenguaje de programación en particular o para varios lenguajes a la vez. Los IDE contienen un editor de texto, un compilador, un depurador y un creador de interfaz gráfica.

1.3.7.1 Zend Studio.

Zend Studio es un IDE para realizar páginas en código PHP. Consta de dos partes en las que se dividen las funcionalidades de la parte del cliente y de la parte del servidor.

Una de las cosas que destaca esta herramienta es que contiene una ayuda que asiste en todo momento no solo con las funciones del lenguaje, sino también con las que son creadas por el propio desarrollador. Contiene un modulo de Apache que permite depurar una web directamente en el navegador. [11]

CAPITULO 1: FUNDAMENTACION TEORICA

Esta herramienta permite la conexión con base de datos complejos como MySQL y PostgreSQL, es multiplataforma y brinda soporte a servidores como Internet Information Server y Apache que es el servidor que se empleará en el desarrollo de este trabajo.

1.3.7.2 Dreamweaver.

Es una de las herramientas de diseño Web mas avanzada. Es fácil de usar para cualquier persona incluso si no tiene muchos conocimientos en esta área. Dreamweaver brinda la posibilidad de crear desde una página Web sencilla hasta un sitio web muy complicado.

Proporciona una potente combinación de herramientas visuales para la creación de páginas Web y el soporte para la edición de código, gracias a las robustas características para la integración y diseño basado en CSS.

Una de las ventajas que tiene es que permite la creación de páginas a través de su interfaz visual, es decir, para realizar el diseño de una página web no es necesario tener conocimientos de HTML puesto que esta herramienta proporciona todos los elementos para crearla visualmente. [12]

1.3.8 Sistemas gestores de Base de datos (SGBD).

Un sistema gestor de base de datos (SGBD) es un software que facilita el proceso de gestionar, construir y manipular una Base de Datos, la cual no es más que la recopilación de toda la información de un sistema u organización. Este tipo de software sirve como interfaz entre la base de datos y las aplicaciones que la utilizan y garantiza la integridad, confidencialidad y la seguridad de los datos que se encuentran en ella y le posibilita tanto a los usuarios, desarrolladores, analistas y administradores la posibilidad de suministrar los medios básicos para describir, manipular y recuperar los datos guardados en la Base de Datos.

1.3.8.1 MySQL.

MySQL es un sistema gestor de base de datos relacional. Debido a su gran rapidez y facilidad de uso es uno de los gestores más usados e todo el mundo. Esto es provocado por su gran compatibilidad con diferentes lenguajes de programación así como por su fácil instalación y soporte.

CAPITULO 1: FUNDAMENTACION TEORICA

Entre las características que destaca se encuentra su portabilidad, MySQL soporta gran cantidad e tipos de datos para las columnas, soporta hasta 32 índices por tablas y su gestión de usuarios y passwords se realiza manteniendo un alto nivel en la seguridad de los datos.

1.3.8.2 PostgreSQL.

PostgreSQL está considerado como uno de los gestores de base de datos más avanzados de los de código abierto.

Entre las características con que cuenta destacan:

- Es capaz de manejar complejas rutinas y reglas.
- Soporta tipos de datos que sean creados por el usuario.
- Emplea la tecnología MVCC (Control de concurrencia Multi-versión), la que usa con el fin de evitar bloqueos innecesarios dentro de la base de datos.
- Permite la gestión de diferentes usuarios y los permisos otorgados a estos.
- Usa la arquitectura cliente – servidor que es la arquitectura con la que se trabajará.[13]

PostgreSQL muestra una gran cantidad de ventajas con respecto al resto de los gestores de bases de datos que han hecho que haya sido escogido para realizar este trabajo, entre estas se encuentran: su soporte de herencia, es gratuito y de código abierto, es más rápido que muchos de ellos, soporta caídas del sistema y cortes de luz. PostgreSQL es un gestor creado para trabajar con un gran volumen de información y en este trabajo el sistema que se propone maneja una gran cantidad de información debido a que en la universidad se realizan a diario más de mil llamadas telefónicas. [14]

1.3.9 Servidor Web.

Se llama servidor a una máquina física cuyo propósito es proveer datos que después van a ser utilizados por otra máquinas. Estos pueden proveer todo tipo de información, como programas y archivos. El servidor Web es un programa que corre sobre el servidor que escucha las peticiones HTTP que le llegan y las satisface. Dependiendo del tipo de la petición, este buscará una página Web o bien ejecutará un programa en el servidor. Y siempre devolverá algún tipo de resultado HTML al cliente o navegador que realizó la petición.

CAPITULO 1: FUNDAMENTACION TEORICA

El servidor Web va a ser fundamental en el desarrollo de las aplicaciones del lado del servidor, que vayamos a construir, ya que estas se ejecutarán en él.

1.3.10 Apache

El servidor Apache es un software libre y multiplataforma. Es un programa que permite crear un servidor http de una forma rápida y sencilla. Es aplicable para sistemas operativos como NetWare, OS/2, Unix, Windows 95/98 y Windows NT 4.0.

Apache está diseñado para ser un servidor Web potente y flexible que pueda funcionar en la más amplia variedad de plataformas y entornos. Esto hace que a menudo sean necesarias diferentes características o funcionalidades. Apache se ha adaptado siempre a una gran variedad de entornos a través de su diseño modular. Este diseño permite a los administradores de sitios Web elegir qué características van a ser incluidas en el servidor seleccionando qué módulos se van a cargar, ya sea al compilar o al ejecutar el servidor. El servidor puede personalizarse mejor para las necesidades de cada sitio o aplicación Web. [15]

1.3.11 Metodología de desarrollo. RUP.

El "quién" está haciendo "qué", "cuándo" y "cómo" en un proyecto de desarrollo de software lo definen las metodologías de desarrollo. Estas nos muestran los pasos a seguir en un proyecto de software para que éste se realice con la calidad requerida, así como los trabajadores que intervienen y desarrollan las actividades que estas proponen. Actualmente, existen una serie de metodologías que permiten llevar a cabo con éxito el desarrollo de software. Entre las que más destacan se encuentran XP y RUP.

1.3.11.1 Extreme Programming (XP).

Es una de las llamadas metodologías hábiles y es utilizada para proyectos de corto plazo. Esta metodología como su nombre lo indica consiste en una programación extrema y una de sus principales características es que el cliente forma parte del equipo de desarrollo.

CAPITULO 1: FUNDAMENTACION TEORICA

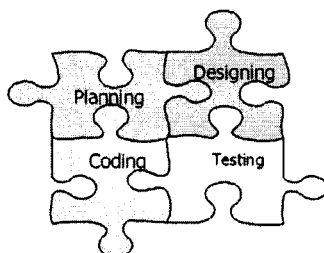


Figura 4 Metodología XP

En esta metodología el cliente; como parte del grupo de trabajo, es quien decide qué es lo que se debe implementar y tiene la posibilidad de agregar o quitar requerimientos en cualquier momento durante el desarrollo. El desarrollador; por su parte, decide la manera de implementar los requerimientos impuestos por el usuario y tiene la oportunidad de cambiarlos a medida que vaya descubriendo cosas.

Lo fundamental que posee este tipo de metodología es:

- Existe una amplia comunicación entre los usuarios y los desarrolladores.

El desarrollo de los módulos del sistema se realiza a través de un proceso muy simple.

1.3.11.2 Proceso Unificado de desarrollo de software (RUP).

Una de las metodologías de desarrollo más usadas en el mundo es el Rational Unified Process (RUP) debido a que es muy fácil de adaptarse a cualquier proceso de desarrollo.

El proceso de desarrollo en esta metodología se encuentra dividido por ciclos, los que se dividen en fases que contienen iteraciones que hacen un recorrido por todas las disciplinas.

La metodología RUP consta de tres características fundamentales: Es dirigida por los casos de uso, centrada en la arquitectura e iterativo e incremental.

Se encuentra dividida en cuatro fases de desarrollo: Inicio; donde se define la visión del proyecto, Elaboración; se realiza un análisis del dominio del problema y se eliminan la mayor cantidad de riesgos, Construcción; se lleva a cabo la implementación de todos los componentes del producto con la

CAPITULO 1: FUNDAMENTACION TEORICA

intención de alcanzar una versión de prueba de éste y por último la fase de Transición; donde al comenzar ya se encuentra listo el producto y se adiestra al cliente en el uso de este.

En RUP se definen 9 disciplinas de trabajo a lo largo del proceso de desarrollo que se integran en cada fase, teniendo en cada una de ellas un mayor o menor esfuerzo. Entre las disciplinas de trabajo podemos citar: Modelo del negocio, Requisitos, Análisis y Diseño, Implementación, Prueba y Despliegue; y tres de apoyo: Gestión de proyectos, Entorno y Gestión de Configuración y Cambios.

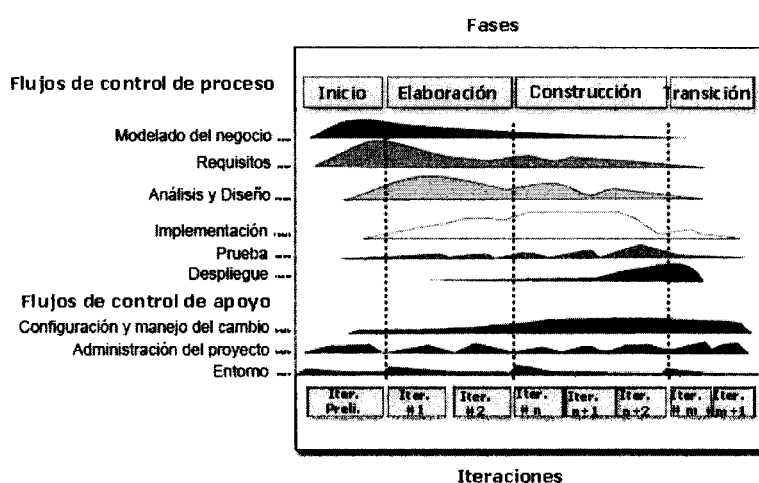


Figura 5 Relación entre flujo o disciplinas de trabajo y fases en RUP

Esta metodología consta de tres elementos fundamentales:

- Trabajadores: Se conoce como trabajadores a las personas o entes involucrados en los procesos.
- Actividades: Es una tarea específica que una persona que se encuentre desempeñando un rol puede realizar. Al terminar estas se va a obtener como resultado un producto.
- Artefactos: Es la información acerca del proyecto que se genera a lo largo del proceso de desarrollo, esta información puede ser producida o modificada a través de este.

Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software. [16]

CAPITULO 1: FUNDAMENTACION TEORICA

1.3.11.3 ¿Por qué usar RUP?

En la realización de este trabajo se usará RUP como Metodología de desarrollo de software debido a que al generar una serie de artefactos a través de su ciclo de desarrollo, proporciona a una persona que no se haya relacionado desde el principio con trabajo la posibilidad de comprender el camino que se ha tomado para desarrollar el software. Esto es una ventaja que tiene RUP sobre XP, además de que esta última está pensada para resolver los problemas inmediatos que tenga el cliente olvidándose de los que podrían surgir más adelante. En RUP se analizan y documentan los requisitos antes de ser aceptados. Todo esto trae consigo que cuando se termine el proceso en RUP se obtenga como resultado un producto más robusto, pues a través del ciclo de desarrollo se van corrigiendo los errores que se encuentren.

1.3.12 Lenguaje de Modelación. UML.

El Lenguaje Unificado de Modelado (UML) se usa para documentar proyectos de software a través de diagramas que sirven de apoyo a las metodologías de desarrollo. En la actualidad es el lenguaje estándar que se utiliza para modelar un sistema de software.

Entre los diagramas que incluye se encuentran: Diagrama de casos de uso, Diagrama de clases, Diagrama de estado, Diagrama de secuencia, Diagrama de clases, Diagrama de colaboración, Diagrama de componentes y Diagrama entidad relación.

En su haber cuenta con un grupo de elementos entre los que se pueden destacar: actores, casos de uso, clases, atributos, operaciones, actividades y entidades.

Este es un lenguaje totalmente independiente del proceso que se utilice para desarrollar el software y permite modelar sistemas totalmente orientados a objetos.

CAPITULO 1: FUNDAMENTACION TEORICA

1.3.13 Herramientas.

1.3.13.1 Herramienta CASE. Visual Paradigm.

Las herramientas CASE han sido creadas para apoyar a los desarrolladores en el proceso de desarrollo de software permitiendo que toda la documentación referente a este quede automatizada a través de ellas.

Visual Paradigm es una herramienta CASE para el modelado que usa UML 2.1. Contiene una interfaz fácil de usar. Entre sus principales características está que tiene integración con diversas IDEs como: NetBean y Eclipse y que contiene ingeniería inversa para Java y .NET entre otras.

Esta herramienta soporta el ciclo de vida completo de desarrollo de software. Brinda ayuda para desarrollar aplicaciones que tengan calidad con menor coste y tiempo. Genera documentación de los diagramas, código a partir de estos y todos los diagramas de clases. Es una herramienta fácil de usar y que asiste en todo momento al desarrollador.

1.3.13.2 Framework.

Se considera framework a una aplicación genérica incompleta y configurable a la que se le añaden las últimas piezas con el fin de construir una aplicación concreta. Existen varios tipos de framework: orientados a la interfaz de usuario, orientados a aplicaciones de publicación de documentos y orientados a la parte de control de eventos.

Los framework han sido diseñados con el único propósito de facilitar el trabajo de los desarrolladores. Estos son fáciles de usar, lo que permite que en un grupo de trabajo se disminuya el tiempo a emplear en desarrollar un sistema funcional. [17]

1.3.13.2.1 Symphony.

Symphony esta compuesto por un conjunto de herramientas que simplifican el desarrollo de una aplicación web. Utiliza el patrón Modelo Vista Controlador. Symphony contiene una serie de elementos de ayuda que permiten incluir formularios y cuenta con un mecanismo de validación automática de los datos de este.

CAPITULO 1: FUNDAMENTACION TEORICA

Este framework solo funciona con PHP5, simplifica al máximo la aplicación de patrones de diseño. Consta de una de las mejores documentaciones en la web, la que se encuentra en una constante actualización. [18]

1.3.13.2 CodeIgniter.

CodeIgniter es un framework que se utiliza para codificadores PHP y es usado por un gran número de personas debido a que es una herramienta de desarrollo fácil para construir aplicaciones Web, tiene una interface simple y un amplio acceso a sus librerías.

Este framework trabaja con el patrón de arquitectura Modelo Vista Controlador (MVC); una de las principales características que posee es que incluye un gran número de librerías que facilitan el trabajo con él. Contiene una gran cantidad de información a la cual se puede acceder y obtener todos los secretos de este, que no requiere instalación para su utilización. Consta de una característica que brinda una forma particular de realizar consultas a la base de datos sin necesidad de tener mucho conocimiento SQL.

1.3.14 Patrón de Diseño. Modelo Vista Controlador (MVC).

El Modelo Vista Controlador (MVC) es un patrón de diseño que separa los datos en una aplicación de las interfaces de los usuarios y la lógica de esta. Esto lo hace básicamente en tres componentes distintos: el Modelo, la Vista y el Controlador.

- **Modelo:** Es el encargado de acceder a los datos con los que se trabaja en la aplicación y se encuentran guardados en una base de datos.
- **Vista:** Representa a la interfaz de usuario, incluye código HTML y CSS que se envían a través del navegador. Es esta la encargada de mostrar al usuario los datos que se encuentran en la Modelo.
- **Controlador:** Es el encargado de unir los otros dos componentes. Responde a los eventos y se encarga de modificar las Vistas y los Modelos.

CAPITULO 1: FUNDAMENTACION TEORICA

Esta separación en componentes brinda la ventaja de que cuando se quiere modificar algo en la aplicación basta con dirigirse a uno de ellos sin verse afectado grandemente ninguno de los otros, lo que posibilita que haya una mayor organización en el trabajo y este cueste menos esfuerzo. [19]

Entre las ventajas que cuenta este patrón se cuentan: que al mismo tiempo varias páginas de la red pueden ver al mismo modelo y de formas diferentes. También se podría agregar otras formas de representar algo y no se vería afectada la modelo, puesto que esta es independiente de las vistas.

1.4 Conclusiones.

En este capítulo se abordó el estudio del estado del arte de los Sistemas de Registro y Control de Llamadas resultado de una investigación realizada acerca de los sistemas que pudieran tener un punto en común con el que se propone desarrollar. Además se realiza un análisis de las tecnologías actuales que pueden servir para el desarrollo de este y como pueden ser empleadas con este objetivo.

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

CAPITULO 

CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción

El desarrollo de un sistema de software se sustenta en la comprensión de los procesos que lo conforman, así como en la identificación de las personas que interactúan con él.

En este capítulo realizaremos un análisis de los procesos que sustentan nuestro negocio; a través de él definiremos actores y trabajadores de este, así como los casos de uso que lo soporten.

Este apartado también nos brindará la oportunidad de determinar los requisitos funcionales y no funcionales con los que contará nuestro sistema. A lo largo de él se definirán los actores y trabajadores del sistema, así como los casos de uso que darán cumplimiento a las características que debe poseer la aplicación a desarrollar.

2.2 Objeto de estudio.

2.2.1 Problema y situación problemática.

El registro y control de las llamadas que se realizan a través de la pizarra telefónica en la universidad es uno de los principales objetivos que se tiene en el SubNodo 1 de la Universidad de las Ciencias Informáticas con el fin de mejorar el servicio telefónico en este centro de estudios.

A través de este proceso se registra la duración de las llamadas, su origen, su destino, el costo que importan y desde, o hacia donde se realizan estas. También se puede conocer como se comporta el flujo de llamadas dentro de la universidad conociendo de esta manera los horarios en los que este se hace mayor, los lugares a los que más se realizan y de los que más se realizan llamadas telefónicas. Con el conocimiento de toda esta información se pueden tomar una serie de medidas para evitar que en determinado momento se congestionen las líneas telefónicas de manera que el tráfico de llamadas se haga imposible dentro de la universidad.

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

En la actualidad el proceso de registro y control de las llamadas telefónicas se realiza a través de un sistema llamado Telmonitor, el cual tiene como función principal ofrecer a los usuarios que acceden a él la información de las llamadas que soliciten.

Teóricamente, este sistema resuelve todo el problema referente a la información relacionada con las llamadas telefónicas, pero en la práctica contiene una serie de problemas que dan al traste con el propósito para el que fue creado.

El principal problema que presenta radica en la forma en que se encuentran almacenados los datos que brinda la pizarra telefónica; estos datos se encuentran en una base de datos modelada en Access que los guarda temporalmente y luego los pasa a otra en MySQL; la primera, cuando se encuentra manejando mucha información no responde a las peticiones que se le realizan y deja de almacenar los datos. Esto provoca que no exista disponibilidad de la información.

Los usuarios del Telmonitor, en ocasiones necesitan conocer otro tipo de información acerca de las llamadas y las extensiones que el sistema no es capaz de brindar lo que provoca que resulten insatisfechos.

Otro de los grandes problemas que brinda actualmente esta aplicación es su inestabilidad pues a veces a los usuarios de este les resulta imposible su acceso a ella.

Con la creación de un sistema que sea capaz de registrar toda esta información y brindarla a los usuarios de la manera que ellos la necesiten se puede agilizar el trabajo de estos, garantizando de esta forma que al encontrarse la información disponible en todo momento para estos, se acelere el proceso de encontrar el origen de un problema y hallar la solución a este.

2.2.2 Objeto de automatización:

En la realización de este trabajo se centrará la atención en 3 procesos:

- Proceso de gestionar los usuarios que tendrán acceso al sistema:

Este es uno de los principales procesos del sistema porque a través de él se le da acceso a la información a una persona determinada.

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

- Proceso de gestionar la información referente a las llamadas telefónicas:

Este es el principal proceso que realiza el sistema; a través de él se le brinda a los usuarios toda la información que se tenga almacenada acerca de las llamadas telefónicas (costo, duración, número marcado, hora de comienzo de la llamada, hora de fin de la llamada). También debe brindar la posibilidad de hallar el máximo, el mínimo, el promedio y el total de cualquier información que se brinde así como de realizar gráficos estadísticos que permitan representar estos datos.

2.2.3 Información que se maneja.

En esta aplicación se manejan los datos referentes a las llamadas telefónicas que se encuentran almacenados en una base de datos. Toda esta información; número marcado, hora de inicio de la llamada, duración de esta, la fecha en que fue realizada, es necesaria para crear los reportes que brinda la aplicación.

2.3 Propuesta del sistema.

Se propone una aplicación Web que proporcione información acerca de las llamadas que se realizan en la Universidad, ya sea desde el propio centro o hacia este. Esta información se proporcionará a partir de reportes que se realizarán en los que se darán a conocer la duración de las llamadas, la hora de su inicio, la hora de su fin, el importe de cada una, el origen o destino que tuvieron, además del tronco por el cual se realizó dicha llamada telefónica. Permite la realización de gráficos estadísticos a través de los cuáles se puede ver el flujo telefónico dentro de la universidad y realizar análisis acerca del comportamiento de este.

El sistema brindará la oportunidad de crear un informe resumen en el que se muestre información acerca de las llamadas pero de manera resumida como por ejemplo: el total de llamadas que se han realizado en un rango de fecha determinado, el costo que importaron estas llamadas, minutos promedio por llamada, importe promedio por minuto, importe promedio por llamada. También muestra las 10 llamadas de mayor duración y las de mayor costo.

Se debe permitir adicionar, modificar y eliminar usuarios con el fin de dar responsabilidades y garantizar que cada uno de ellos pueda desempeñar solo las acciones a las que está asignado. Debe también dar la posibilidad de exportar los reportes que se realicen y de imprimirlos.

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

El Telmonitor, sistema que se usaba con anterioridad, tiene como objetivo la gestión de la información de las llamadas que se realizan en la universidad a través de la pizarra telefónica. Este se encuentra compuesto por una base de datos temporal realizada en Access encargada de almacenar la información sacada de la pizarra y que luego es trasladada a otra base de datos en MySQL y que guardará los datos de forma permanente.

Por su parte la aplicación a realizar contará con una base de datos en PostgreSQL, considerado como uno de los gestores más potentes en la actualidad. La información de la pizarra pasará directamente a la nueva base de datos lo que evitará el traspaso de información evitando así el riesgo de pérdida de esta.

Otra opción que brinda la propuesta y que supera a la anterior es la oportunidad de administrar los usuarios pues con anterioridad el administrador tenía que hacerlo de manera directa en la base de datos y la posibilidad de exportar los reportes que se realicen.

2.4 Modelo del negocio.

Muchas personas en la Universidad de Ciencias Informáticas tienen la necesidad de conocer detalles acerca de las llamadas que se han realizado a determinadas líneas dentro de la universidad. Nuestro negocio comienza cuando esta persona se dirige hacia el SubNodo 1 y solicita un reporte con toda esta información. El operador, persona encargada del Telmonitor; sistema que en la actualidad brinda toda esta información, luego de buscarla en el sistema se la hace llegar al solicitante, que se va satisfecho del lugar.

El Telmonitor gestiona toda la información de las llamadas telefónicas que se encuentra almacenada dentro de una base de datos. Entre la información que brinda este sistema se encuentra la duración de la llamada, el teléfono al que se realizó la llamada así como del que se realizó y la factura que importa esta.

Puede darse el caso de que en un determinado momento la información solicitada no se encuentre archivada, en el cuál esa persona tendría que retirarse insatisfecho sin haber obtenido lo que fue a buscar. Esta; precisamente, es la causa fundamental que originó nuestra investigación, cuyo principal objetivo es garantizar que la información que sea solicitada por una persona determinada se encuentre

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

disponible siempre que sea necesaria. Para lo cual se hace necesario que se archive con una mayor seguridad y de una mejor manera.

2.4.1 Descripción de los actores y trabajadores del negocio.

Actores del negocio	Justificación
Usuario	Representa a la persona que necesita y solicita información referente a llamadas telefónicas que han sido realizadas a través de la pizarra dentro del entorno.

Tabla 1. Descripción del usuario, actor del negocio.

Trabajador del negocio	Justificación
Operador	Representa a la persona que tiene acceso al sistema, por lo cual es la que maneja toda la información referente a las llamadas telefónicas y se la facilita al usuario.

Tabla 2. Descripción del operador, trabajador del negocio.

2.4.2 Diagrama de casos de uso del negocio.

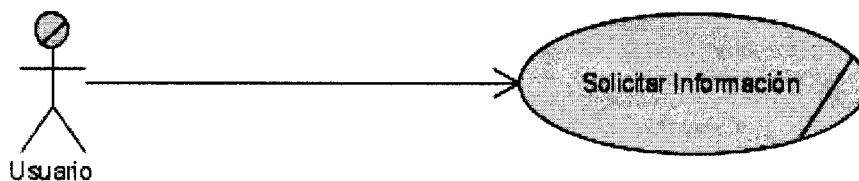


Figura 6 Diagrama de CU del negocio, solicitar información

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

2.4.3 Descripción textual de los casos de uso del negocio.

Mediante la descripción de los casos de uso del negocio se puede evidenciar como interactúan los actores del negocio con este. (Ver Anexo I).

2.4.4 Reglas del negocio:

1. Solo los administradores y trabajadores del SubNodo pueden tener acceso a la información.

2.4.5 Diagramas de actividad.

A través del diagrama de actividades se pueden ver por separado cuáles son las actividades que realiza el actor y cuáles las que realiza el trabajador dentro del negocio. (Ver Anexo II).

2.4.6 Modelo de objetos.

El Modelo de objetos representa la interacción entre los trabajadores del negocio y las entidades de este. (Ver Anexo III).

2.5 Relación de los requerimientos.

Se le llama requerimientos a las condiciones que deben ser cumplidas por un software para ser aceptado por el usuario final. Constan de dos clasificaciones: funcionales y no funcionales.

Requerimientos funcionales: Capacidades o condiciones que el sistema debe cumplir.

Requerimientos no funcionales: Características o propiedades que el sistema debe tener para contar con la aprobación del cliente.

2.5.1 Requerimientos funcionales:

R₁. Adicionar usuario.

Brinda la posibilidad de adicionar un nuevo usuario al sistema mostrando posteriormente el nombre de usuario y el rol que va a jugar dentro del sistema.

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

Dentro del sistema un operador; además de este rol, puede jugar el de administrador que es la persona que tiene la posibilidad de realizar cualquier tipo de acción dentro del sistema. El operador por su parte solo tendrá acceso consultar la información, no a modificarla.

R₂. Modificar usuario.

Brinda la posibilidad de cambiarle los datos a un usuario que se encuentra previamente registrado en el sistema.

R₃. Eliminar usuario.

Posibilita quitarle el acceso a un determinado usuario para entrar en el sistema y acceder a la información que brinda este.

R₄. Autenticar usuario.

Brinda la posibilidad al usuario de poder identificarse antes de entrar al sistema, para lograr esto, el usuario debe de ingresar su nombre de usuario y su contraseña.

R₅. Visualizar tráfico por provincia en un período entre dos fechas.

Permite visualizar el flujo de llamadas existentes en un período determinado entre dos fechas teniendo en cuenta el lugar al que se realizaron.

R₆. Visualizar tráfico por provincia en un período por horas.

Permite visualizar el flujo de llamadas existentes en un período determinado entre dos horas teniendo en cuenta el lugar al que se realizaron.

R₇. Visualizar tráfico por tipo de llamada en un período entre dos fechas.

Permite visualizar el flujo de llamadas que se realizan dependiendo del tipo de llamada en un período entre dos fechas.

Existen tres clasificaciones para las llamadas a tener en cuenta:

Llamadas Locales: Llamadas que se realizan dentro o hacia la misma provincia.

Llamadas Nacionales: Llamadas que se realizan a través de la pizarra telefónica hacia o desde otra provincia del país.

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

Llamadas Internacionales: Llamadas que se realizan a través de la pizarra telefónica desde o hacia otro país.

R₈. Visualizar tráfico por tipo de llamada en un período por horas.

Permite visualizar el flujo de llamadas que se realizan dependiendo del tipo de llamada en un período determinado entre dos horas.

R₉. Visualizar tráfico por tronco en un período entre dos fechas.

Permite visualizar el flujo de llamadas que se realizan dependiendo del tronco de salida o de entrada en un período entre dos fechas.

R₁₀. Visualizar tráfico por tronco en un período por horas.

Permite visualizar el flujo de llamadas que se realizan dependiendo del tronco de salida o de entrada en un período determinado entre dos horas.

R₁₁. Visualizar tráfico por extensión en un período entre dos fechas.

Permite visualizar el flujo de llamadas que se realizan desde o hacia una determinada extensión en un período entre dos fechas.

R₁₂. Visualizar tráfico por extensión en un período por horas.

Permite visualizar el flujo de llamadas que se realizan desde o hacia una determinada extensión en un período determinado entre dos horas.

R₁₃. Visualizar total de tráfico en un período entre dos fechas.

Permite visualizar el flujo total de llamadas que se realizan en el entorno en un período entre dos fechas.

R₁₄. Visualizar total de tráfico en un período por horas.

Permite visualizar el flujo total de llamadas que se realizan en el entorno en un período determinado entre dos horas.

R₁₅. Visualizar costos por tipo de moneda en un período entre dos fechas.

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

Permite visualizar la factura de las llamadas en un período entre dos fechas atendiendo a un determinado tipo de moneda.

R₁₆. Visualizar costos por tipo de moneda en un período por horas.

Permite visualizar la factura de las llamadas en un período determinado entre dos horas atendiendo a un determinado tipo de moneda.

R₁₇. Visualizar costos por tipo de llamada en un período entre dos fechas.

Permite visualizar la factura de las llamadas en un período entre dos fechas atendiendo al tipo de llamada que se realice.

R₁₈. Visualizar costos por tipo de llamada en un período por horas.

Permite visualizar la factura de las llamadas en un período determinado entre dos horas atendiendo al tipo de llamada que se realice.

R₁₉. Visualizar costos por tronco en un período entre dos fechas.

Permite visualizar la factura de las llamadas atendiendo al tronco de salida o de entrada en un período entre dos fechas.

R₂₀. Visualizar costos por tronco en un período por horas.

Permite visualizar la factura de las llamadas atendiendo al tronco de salida o de entrada en un período determinado entre dos horas.

R₂₁. Visualizar costos por extensión en un período entre dos fechas.

Permite visualizar la factura de las llamadas que se realizan desde o hacia una determinada extensión en un período entre dos fechas.

R₂₂. Visualizar costos por extensión en un período por horas.

Permite visualizar la factura de las llamadas que se realizan desde o hacia una determinada extensión en un período determinado entre dos horas.

R₂₃. Visualizar total de costos en un período entre dos fechas.

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

Permite visualizar la factura del total de llamadas que se realizan en el entorno en un período entre dos fechas.

R₂₄. Visualizar total de costos en un período por horas.

Permite visualizar la factura total de las llamadas que se realizan en el entorno en un período determinado entre dos horas.

R₂₅. Filtrar por cálculo de máximos.

Permite obtener el máximo valor de la información que se ha obtenido del sistema a través de reportes.

R₂₆. Filtrar por cálculo de mínimos.

Permite obtener el mínimo valor de la información que se ha obtenido del sistema a través de reportes.

R₂₇. Filtrar por cálculo de totales.

Permite obtener el valor total de la información que ha sido obtenida del sistema a través de reportes.

R₂₈. Filtrar por cálculo de promedios.

Calcula el valor promedio de la información que ha sido obtenida del sistema a través de reportes.

R₂₉. Limpiar filtro.

Brinda la posibilidad de que cuando se cree un filtro y se decida no seguir adelante con él se pueda eliminar toda acción que se haya iniciado con respecto al filtro.

R₃₀. Ordenar indicadores.

Permite el ordenamiento de los indicadores por los cuáles se realiza el reporte en un orden ascendente o descendente.

R₃₁. Visualizar gráficamente los indicadores.

Brinda la posibilidad de ver el reporte realizado en forma de gráfico estadístico lo que proporcionará una comprensión más clara para el usuario del reporte.

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

R₃₂. Enviar por correo.

Brinda la posibilidad al usuario de una vez que tenga el reporte hecho enviarlo por correo electrónico a una determinada persona.

R₃₃. Exportar reporte.

Brinda la posibilidad de sacar el reporte de la aplicación con un formato deseado por el propio usuario.

R₃₄. Imprimir reporte.

R₃₅. Crear informe resumen.

Permite crear un informe resumen a través del cual se brindará un resumen de la información que se tiene almacenada de las llamadas telefónicas en un periodo entre dos fechas determinadas.

2.5.2 Requerimientos no funcionales.

Los requerimientos no funcionales son las propiedades o cualidades con las que debe cumplir el producto para hacerlo más atractivo al cliente y más confiable.

Apariencia o interfaz externa:

- La interfaz debe ser amigable y fácil de entender y de usar para los usuarios del sistema.
- Diseño debe estar orientado a una navegación sencilla.
- Debe comunicarse a través del protocolo http.

Usabilidad:

- El sistema debe ser usado por personas con conocimientos básicos de computación.
- El sistema no contiene documentos de ayuda debido a que su uso es muy sencillo.
- Incorporar la funcionalidad de paginado en las respuestas del sistema que lo requieran.

Rendimiento:

- El sistema debe ser eficiente en la búsqueda de datos.

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

- El tiempo de respuesta no debe excederse de 5 segundos y en caso de que suceda se dará notificación al usuario.
- Debe garantizarse que no ocurra pérdida de datos.

Soporte:

- Se debe garantizar la realización de diferentes pruebas al sistema para garantizar su correcto funcionamiento.
- Se debe garantizar que exista alguna persona que tenga conocimiento acerca de los detalles de la aplicación que sea la encargada de darle mantenimiento.

Portabilidad:

- El sistema debe ser capaz de ejecutarse sin problemas en el Sistema Operativo Windows y en el Sistema Operativo Linux.

Seguridad:

- El sistema debe usar un protocolo de comunicación que sea seguro (https).
- Se debe garantizar que cada usuario se identifique antes de entrar a la aplicación y que una vez dentro solo tenga acceso a la información que tiene permiso.
- Debe garantizarse que se verifiquen acciones irreversibles como las eliminaciones y las modificaciones.
- Se debe garantizar que el sistema se encuentre disponible para el usuario las 24 horas del día de manera que se pueda acceder a ella en cualquier momento.

Hardware:

- Para el correcto funcionamiento del sistema se requiere de un MODEM, una tarjeta de red o de algún dispositivo de red inalámbrica que permitan conectarse a una red local.

Software:

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

- Para el correcto funcionamiento del sistema se necesitan lo siguientes softwares: Sistema Operativo Linux o Windows XP; se debe también contar con la existencia de un navegador; en este caso puede ser Internet Explorer 6 o superior o Mozilla Firefox.

Restricciones en el diseño y la implementación:

- Lenguaje de programación a usar: PHP.
- IDE de desarrollo: Zend Studio.
- Herramienta Casie para modelado UML: Visual Paradigm.
- Gestor de base de datos: PostgreeSQL.
- Servidor Web: Apache en su versión 2.0.

2.6 Modelo de casos de uso del sistema.

2.6.1 Definición de los actores del sistema.

Actores	Justificación
Administrador	Representa a la persona que va a tener el control total en el sistema. Este puede ser el que determine quien puede o no acceder al sistema.
Operador	Representa a la persona que va a interactuar directamente con el sistema y que va a definir todas las pautas de la búsqueda.

Tabla 3. Actores del sistema y su justificación

CAPITULO 2: CARACTERISTICAS DEL SISTEMA

2.6.2 Diagrama de casos de uso del sistema.

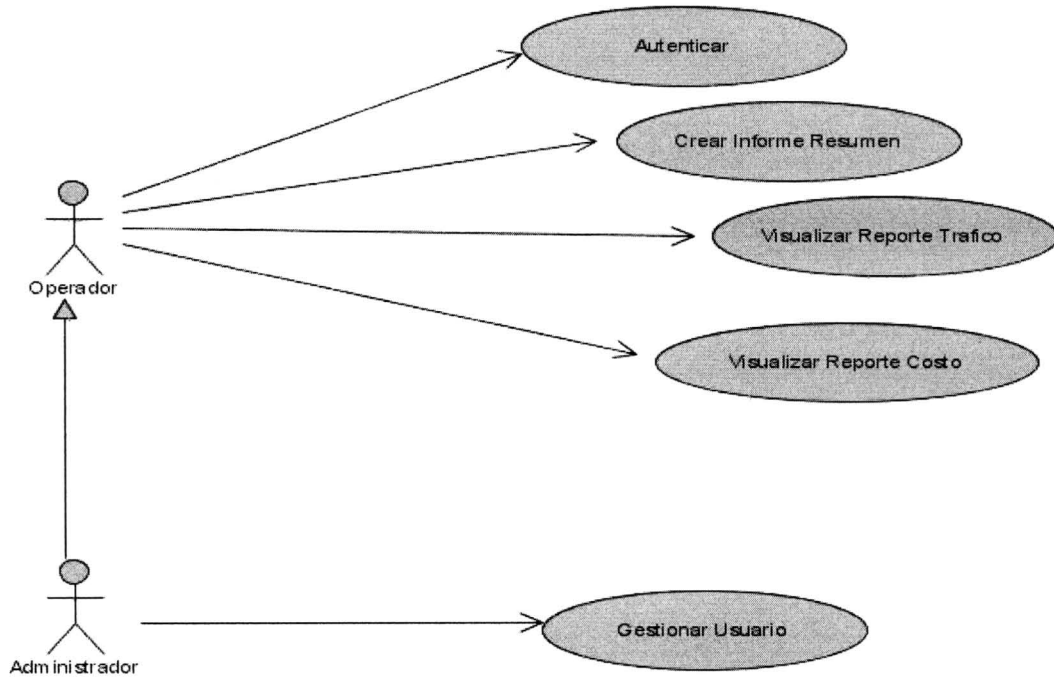


Figura 7 Diagrama de CU del Sistema

2.6.3 Descripción de los casos de uso.

La descripción de los casos de uso del sistema describe como es la interacción actor-sistema. (Ver Anexo IV).

2.7 Conclusiones:

El desarrollo de este capítulo permitió ver cuáles eran las principales características con las que debía cumplir el software. También se mostró el objeto de estudio, situación problemática y se realizó un análisis de todos los procesos que se ejecutan para llevar a cabo las actividades del negocio que luego dan paso al sistema, que también es analizado a través de este apartado.

ANÁLISIS Y DISEÑO DEL SISTEMA

3.1 Introducción.

A través del análisis y el diseño se persigue traducir los requisitos de manera que describa como debe ser implementado el sistema. El análisis solo se encarga de la parte de 'qué' es lo que debe hacer el sistema, por lo que solo se tienen en cuenta los requisitos funcionales. Entre tanto, el diseño se centra en 'como' debe hacerse por lo que también se fija en los requisitos no funcionales.

En este capítulo se hará una descripción del análisis y el diseño de la aplicación a desarrollar, se identificarán las clases del análisis y las del diseño y se modelarán ambas conjuntamente con los diagramas de interacción. Otro aspecto que se verá en esta sección es el modelo lógico de datos y el modelo entidad relación de la base de datos.

3.2 Análisis del sistema.

El objetivo del análisis es comprender con claridad los requisitos de software para determinar que es lo que hará el sistema. A la hora de realizar el análisis no se tiene en cuenta el lenguaje de programación en que se desarrollará la aplicación, ni la plataforma en que será ejecutada esta, pero al desarrollarse en un lenguaje que los desarrolladores entienden sirve para que estos tengan en cuenta algunas de las funcionalidades que puede tener el sistema.

Aquí se definen las clases del análisis; clases interfaz, controladoras y clases entidad con las que se construyen los diagramas de clases del análisis.

3.2.1 Diagramas de clases del análisis.

Los diagramas de clases del análisis contienen las clases de este; su relación con el actor y la relación que tienen entre ellas.

CAPITULO 3: ANALISIS Y DISEÑO DEL SISTEMA

En el análisis se especifican tres tipos de clases: Clases Interfaz (Modelan la interacción que tiene el actor con el sistema), Clases Controladoras (Coordinan las actividades de los objetos que implementan determinadas funcionalidades de los casos de uso) y Clases Entidad (Modelan información que tienen una larga vida y que en ocasiones es persistente, este tipo de clases ayudan a entender de que tipo de clases va a depender el sistema a realizar).

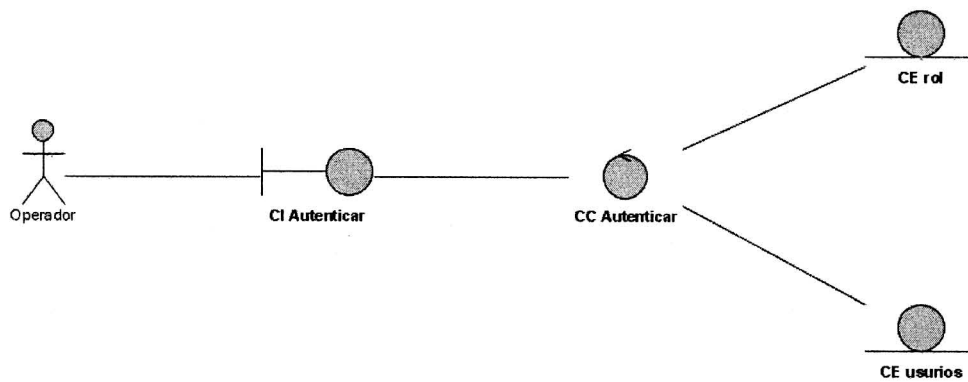


Figura 8 Diagrama de Clase del Análisis CU Autenticar Usuario

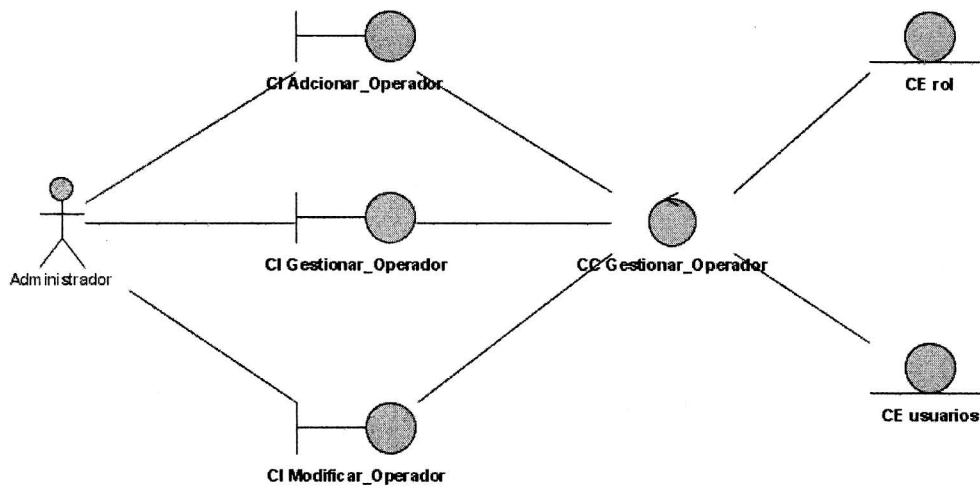


Figura 9 Diagrama de Clases del Análisis CU Gestionar Operador

CAPITULO 3: ANALISIS Y DISEÑO DEL SISTEMA

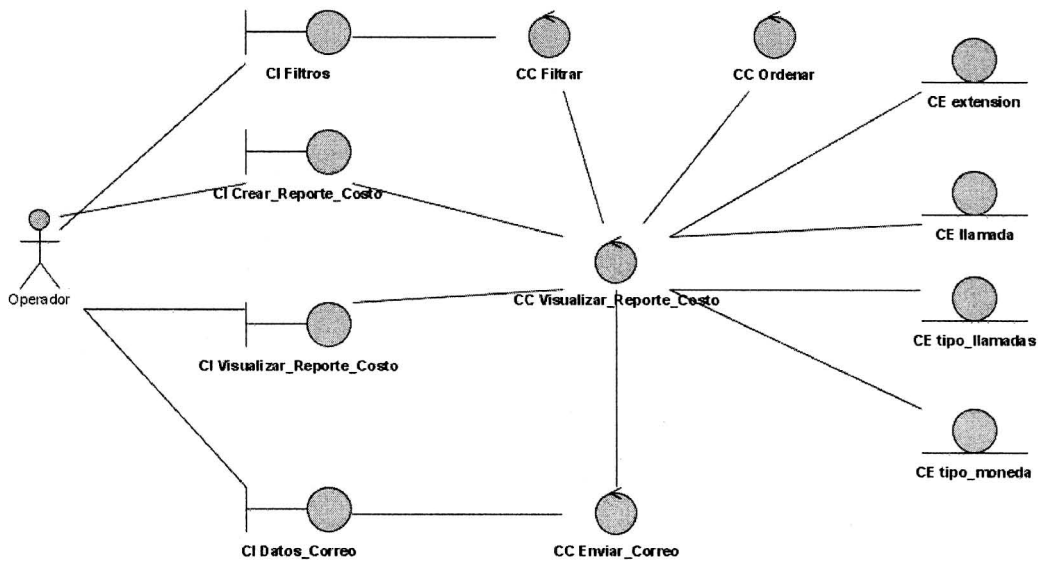


Figura 10 Diagrama de Clases del Análisis CU Visualizar Reporte de Costo

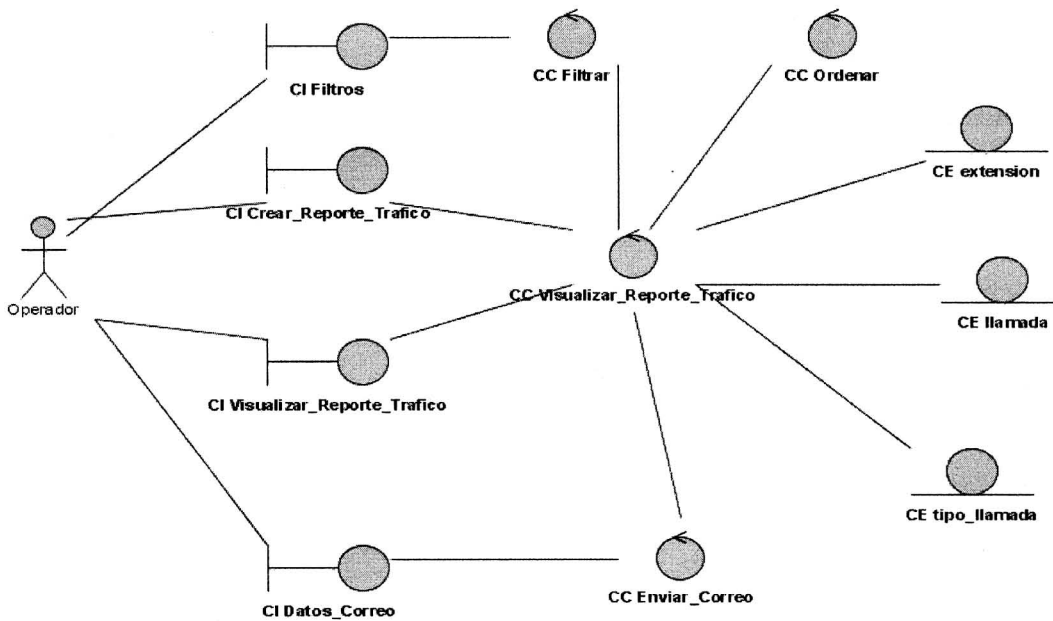


Figura 11 Diagrama de Clases del Análisis CU Visualizar Reporte de Tráfico

CAPITULO 3: ANALISIS Y DISEÑO DEL SISTEMA

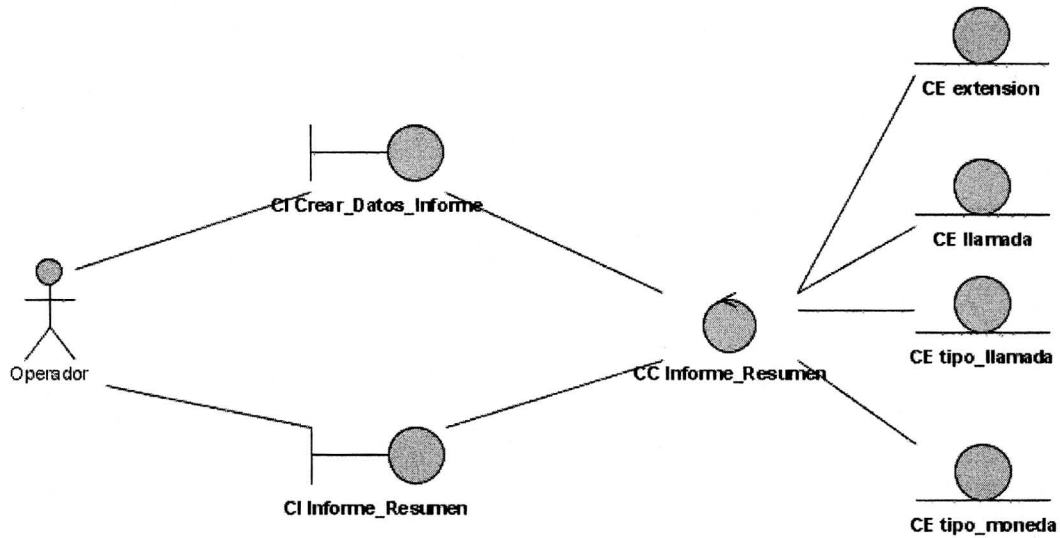


Figura 12 Diagrama de Clases del Análisis CU Crear Informe Resumen

3.3 Diseño.

En el diseño se modela el sistema incluyendo una arquitectura que soporte todos los requisitos de software; aquí se tienen en cuenta todos los requisitos, tanto los funcionales como los no funcionales. El análisis constituye una entrada al diseño, aportándole una estructura del sistema como resultado de un análisis detallado de los requisitos de software. El diseño constituye un punto de partida para la implementación, por lo que debe hacerse de manera que sea suficiente para implementar el software sin ningún tipo de imprecisiones.

3.3.1 Diagramas de interacción.

Un diagrama de interacción muestra una interacción entre un conjunto de objetos; sus relaciones y los mensajes que se mandan entre ellos. Existen dos tipos de diagramas de interacción: Diagramas de colaboración y Diagramas de secuencia, estos dos diagramas se basan en mostrar la misma información pero cada uno la aborda de una manera diferente.

CAPITULO 3: ANALISIS Y DISEÑO DEL SISTEMA

3.3.2 Diagrama de clases WEB del diseño.

Un diagrama de clases representa las clases que serán utilizadas a través de la realización del sistema y la relación existente entre ellas.

El diagrama de clases del diseño constituye una entrada fundamental a la implementación. A través de la realización de este se empleó el patrón de diseño Modelo Vista Controlador.

Caso de uso Autenticar usuario.

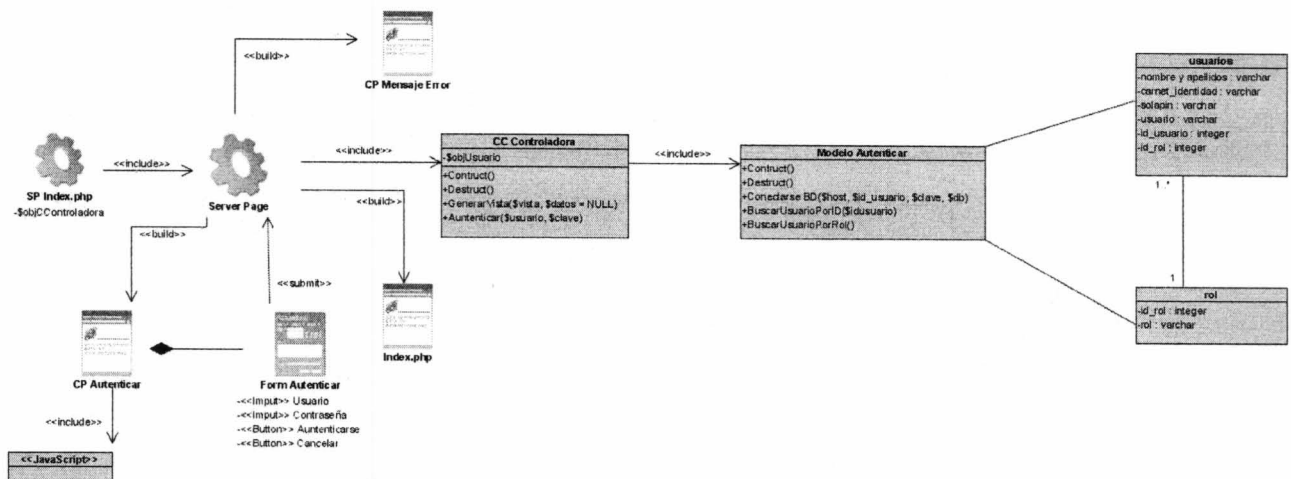


Figura 13 Diagrama de Clases del Diseño CU Autenticar Usuario

CAPITULO 3: ANALISIS Y DISEÑO DEL SISTEMA

Caso de uso Gestionar Operador.

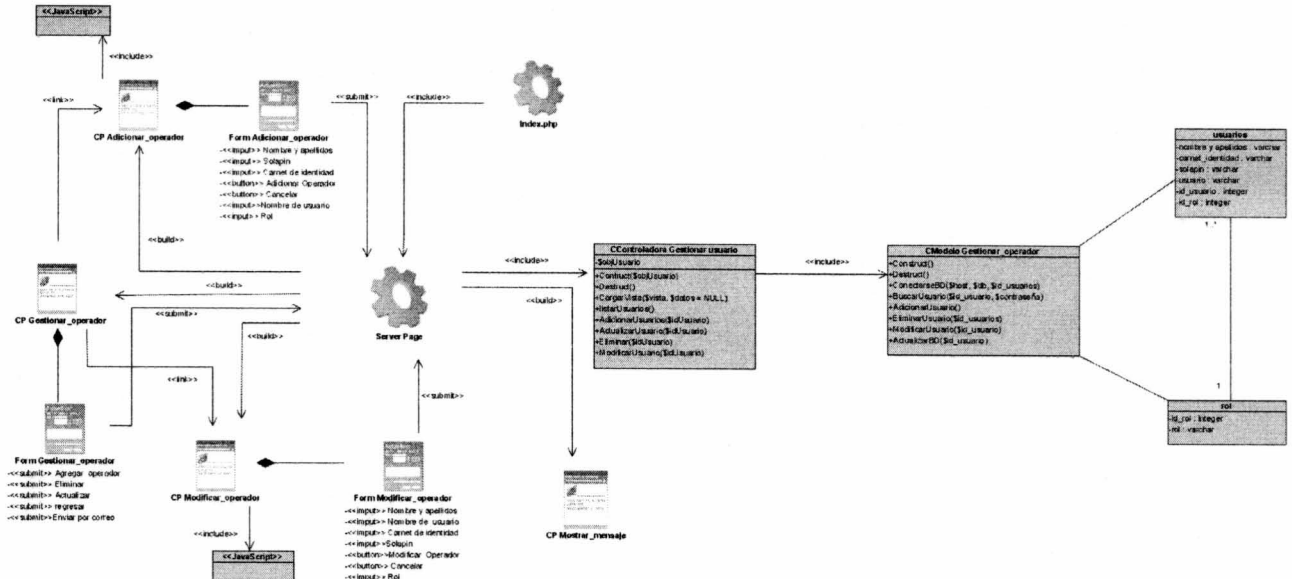


Figura 14 Diagrama de Clases del Diseño CU Gestionar Operador

Caso de uso Visualizar reporte de costo.

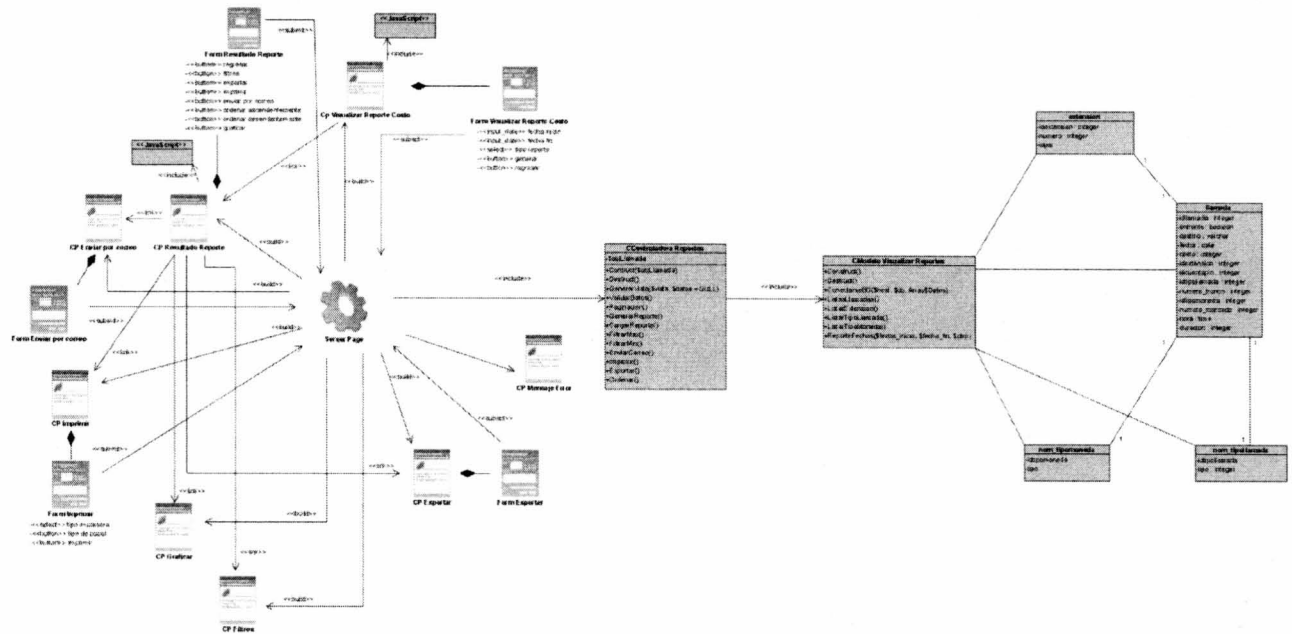


Figura 15 Diagrama de Clases del Diseño CU Vizualizar Reporte de Costo

CAPITULO 3: ANALISIS Y DISEÑO DEL SISTEMA

Caso de uso Gestionar cable.

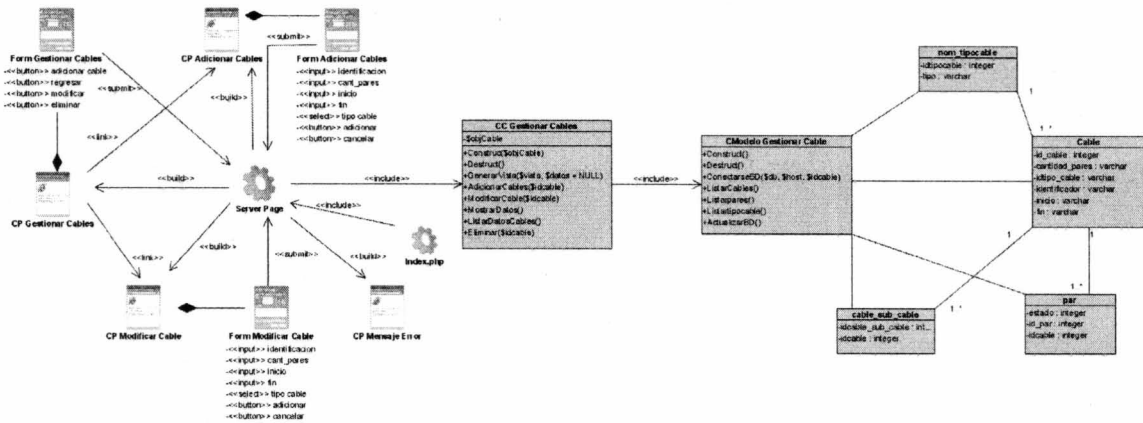


Figura 18 Diagrama de Clase del Diseño CU Gestionar Cable

Caso de uso Hallar facilidades de cables

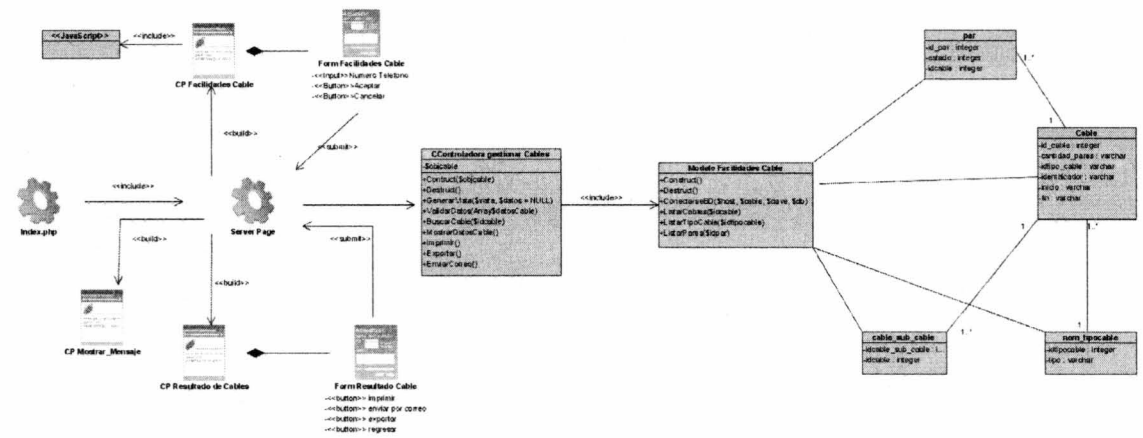


Figura 19 Diagrama de Clase del Diseño CU Hallar Facilidades de Cables

CAPITULO 3: ANALISIS Y DISEÑO DEL SISTEMA

3.4 Diseño de la base de datos.

3.4.1 Modelo lógico de datos (Diagrama de clases persistentes).

El Modelo lógico de datos o Diagrama de clases persistentes lo componen las clases que compondrán la base de datos y que son definidas en el diseño. Su propósito es proveer una vista de las entidades lógicas de datos y sus relaciones independientemente de la plataforma de base de datos que se vaya a usar.

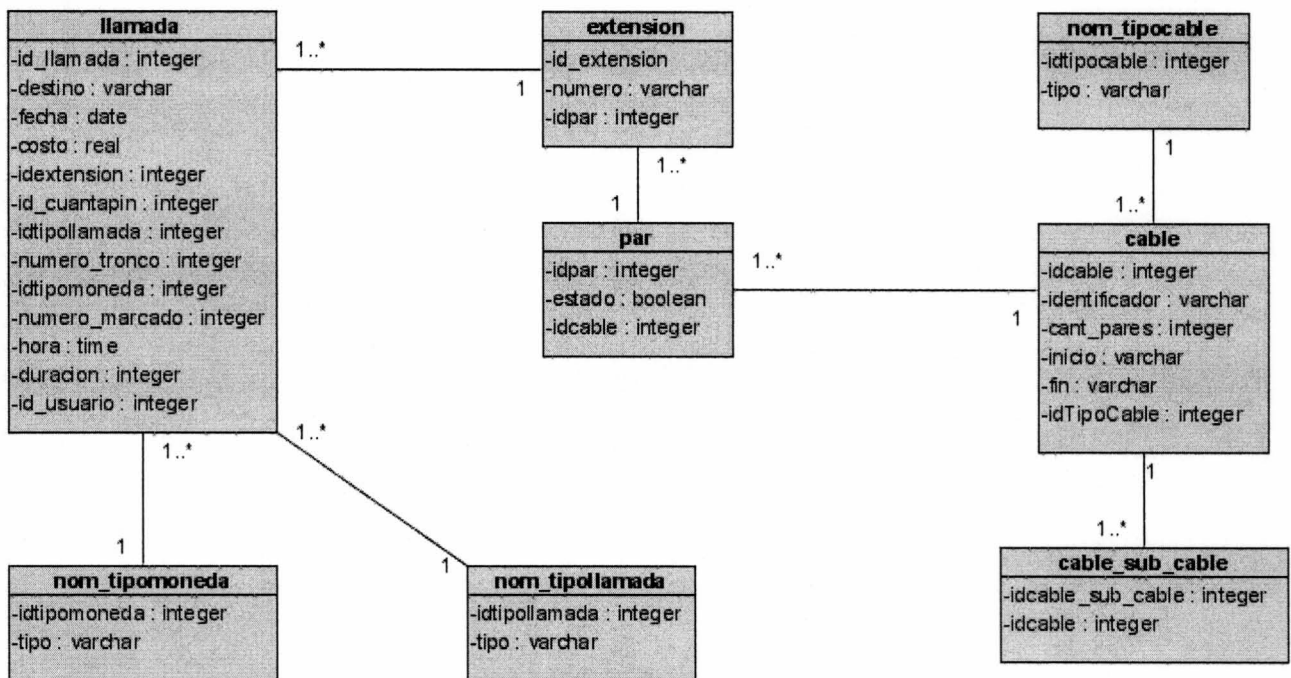


Figura 20 Diagrama de Clases Persistentes

3.4.2 Modelo físico de datos (Modelo de datos).

El modelo físico lo representan las tablas que conforman la base de datos, sus relaciones y sus atributos.

CAPITULO 3: ANALISIS Y DISEÑO DEL SISTEMA

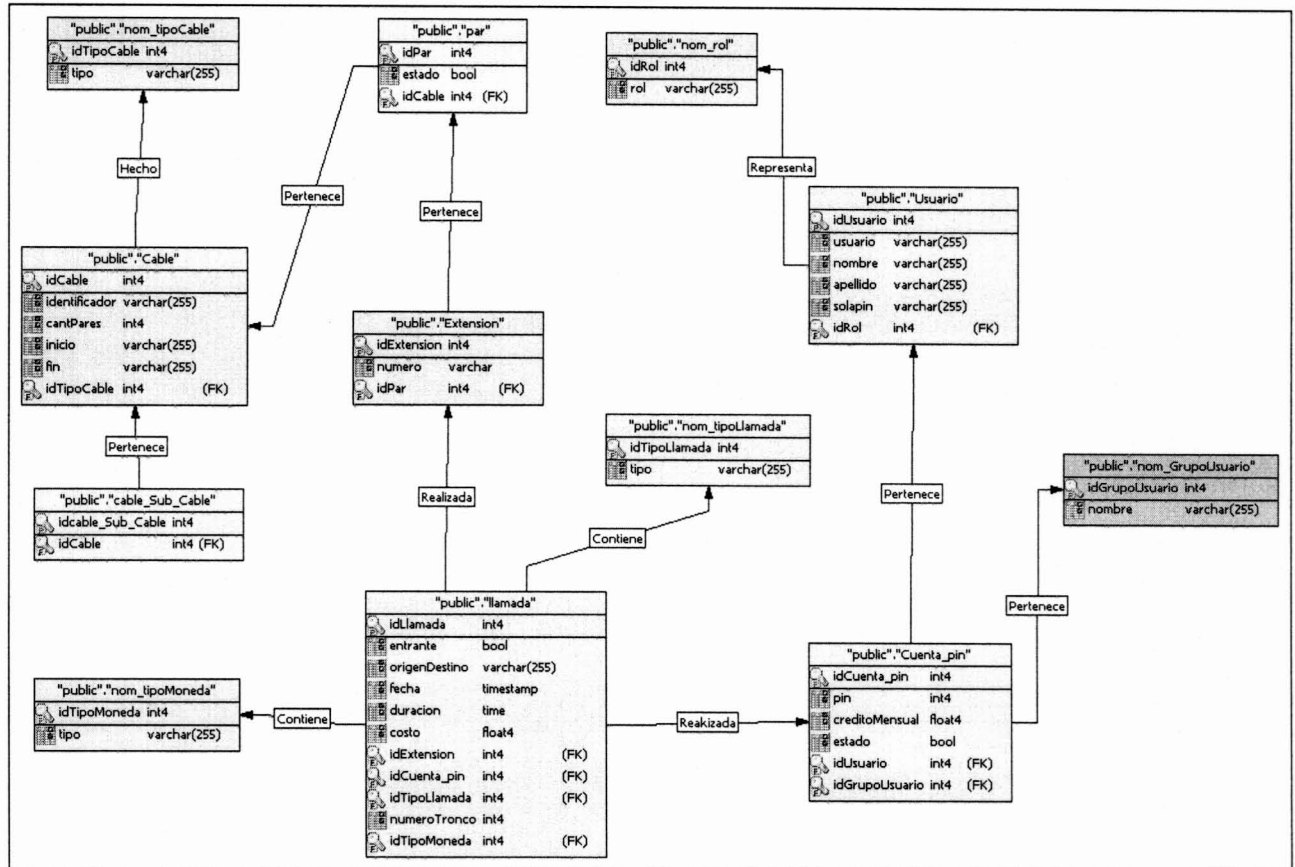


Figura 21 Diagrama Entidad Relación

3.5 Conclusiones.

En este capítulo se modelaron los diagramas de análisis y de diseño quedando conformada de esta manera la arquitectura del sistema. También se muestra el Modelo de datos que refleja las entidades que componen el sistema y la relación existente entre ellas.

IMPLEMENTACIÓN Y PRUEBA

4.1 Introducción.

La implementación es donde se llevan a cabo todas las actividades que se han venido planificando a lo largo del desarrollo del software. A través de la realización de este capítulo se mostrará el diagrama de despliegue y el diagrama de componentes.

4.2 Diagrama de despliegue.

Un diagrama de despliegue representa las relaciones físicas entre los componentes del sistema ya sean de hardware o de software. Este diagrama describe como es la distribución de la funcionalidad entre los nodos de cómputo.

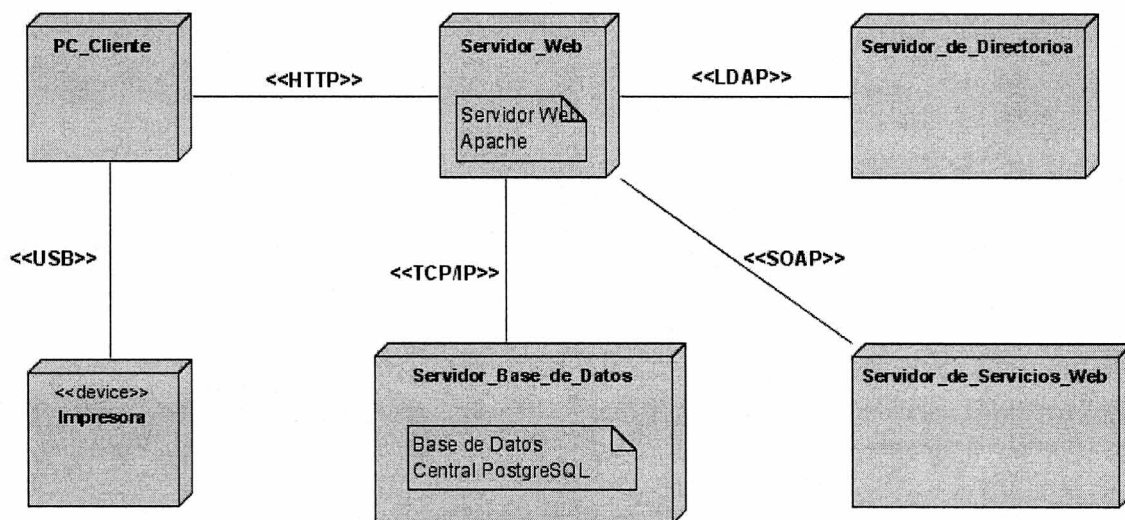


Figura 22 Diagrama de Despliegue

CAPITULO 4: IMPLEMENTACION Y PRUEBA

4.3 Diagrama de componentes.

El diagrama de componentes es un diagrama que se utiliza para mostrar las relaciones existentes entre los modelos de implementación. La importancia de estos radica en que deben especificar los subsistemas de implementación, aclarando sus dependencias a la hora de importar código y organizándolos en capas.

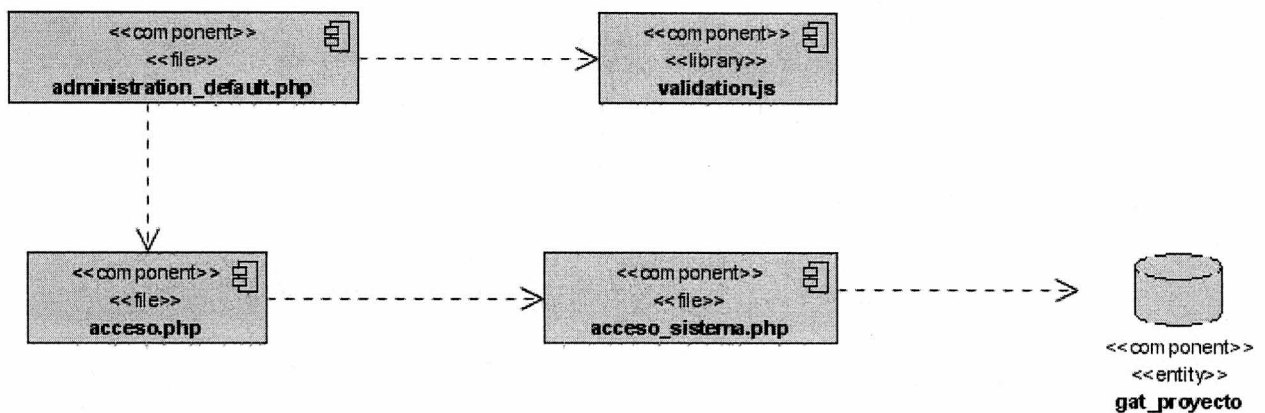


Figura 23 Diagrama de Componentes CU Autenticar

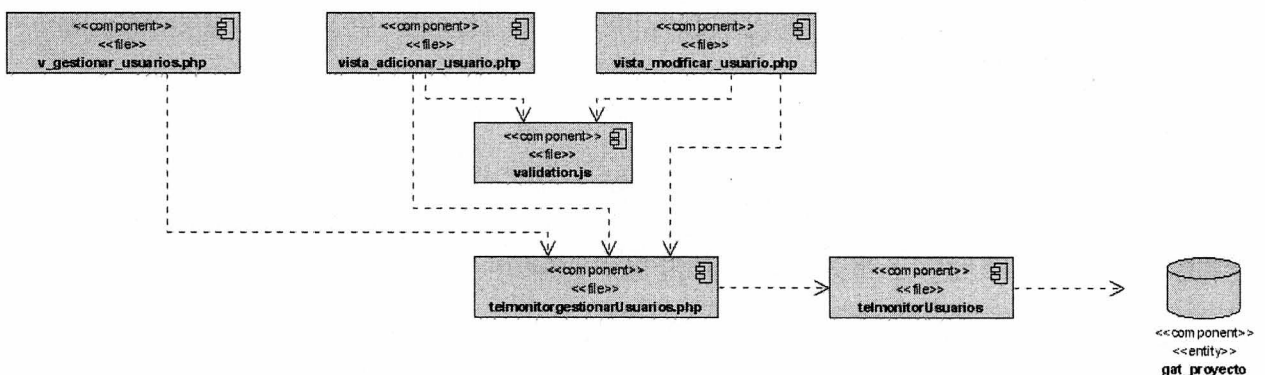


Figura 24 Diagrama de Componentes CU Gestionar Operador

CAPITULO 4: IMPLEMENTACION Y PRUEBA

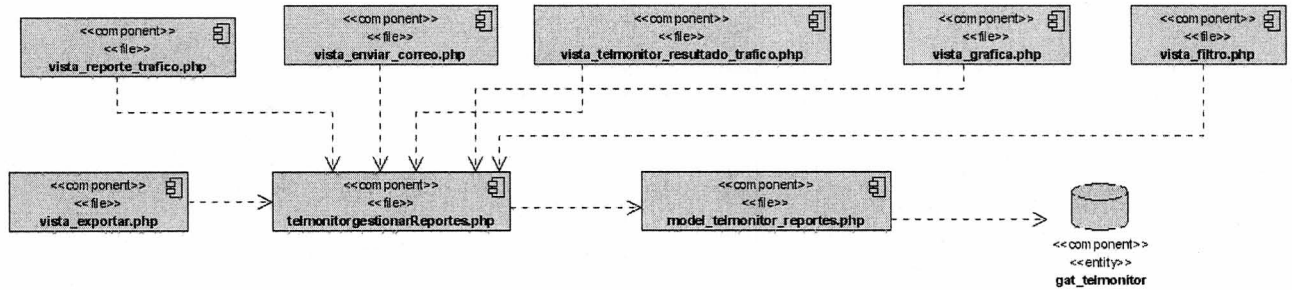


Figura 25 Diagrama de Componentes CU Visualizar Reporte de Tráfico

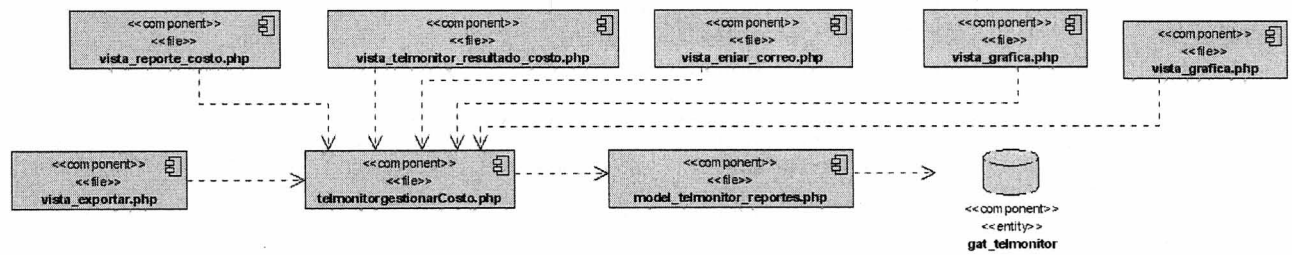


Figura 26 Diagrama de Componentes CU Visualizar Reporte de Costo

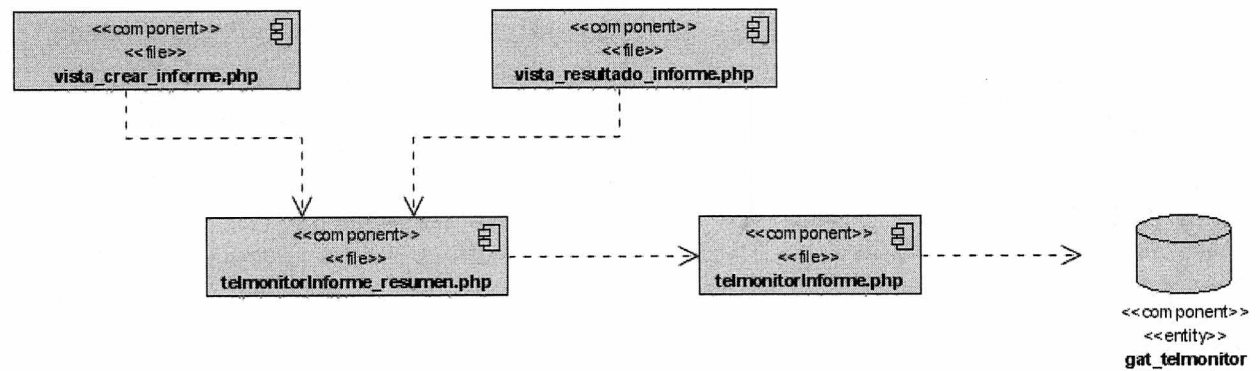


Figura 27 Diagrama de Componentes CU Crear Informe Resumen

CAPITULO 4: IMPLEMENTACION Y PRUEBA

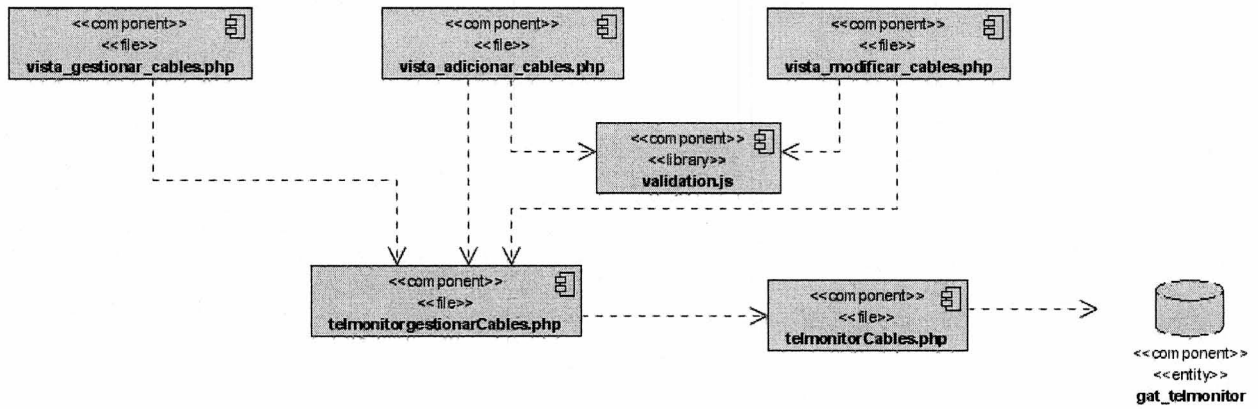


Figura 28 Diagrama de Componentes CU Gestionar Cables

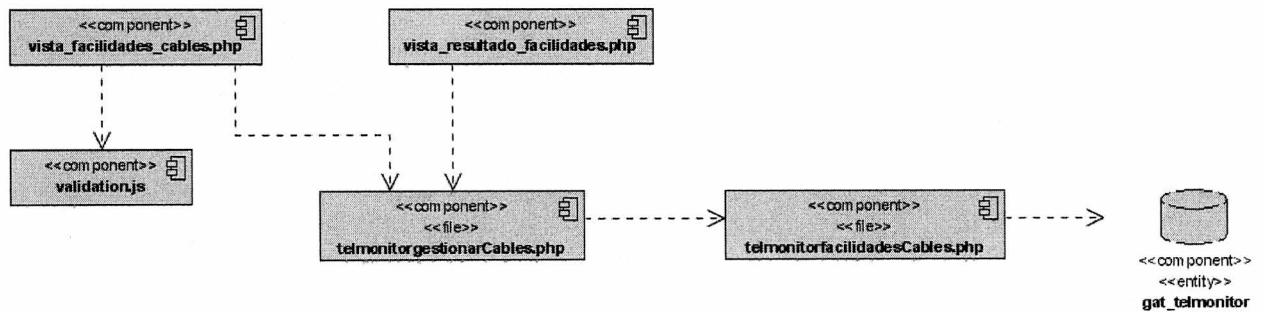


Figura 29 Diagrama de Componentes CU Hallar Facilidades de Cables

4.4 Conclusiones.

En este capítulo pudimos ver como se encuentra distribuida la implementación de nuestro sistema a través de los diagramas de componentes y la distribución de nuestra aplicación a través de equipos de cómputo a través del diagrama de despliegue.

CAPITULO 5: ESTUDIO DE LA FACTIBILIDAD

CAPITULO

5

ESTUDIO DE LA FACTIBILIDAD

5.1 Introducción.

A través de este capítulo se estudiará la factibilidad del sistema a realizar, logrando determinar un aproximado del costo, los recursos y el tiempo que costará desarrollar esta aplicación. Este epígrafe mostrará la estimación realizada al sistema usando el método de estimación por puntos de casos de uso.

5.2 Planificación.

A través de la planificación de un proyecto se puede determinar la cantidad de personas que serán necesarias para su desarrollo, el costo que podrá tener este y los recursos que implicaría su desarrollo.

5.2.1 Cálculo de puntos de casos de uso sin ajustar.

Este valor se calcula a través de la siguiente ecuación:

UUCP = UAW +UUCW donde:

UUCP: Puntos de caso de uso sin ajustar.

UAW: Factor de peso de actores sin ajustar.

UUCW: Factor de peso de los casos de uso sin ajustar.

El factor de peso de los actores sin ajustar se halla teniendo en cuenta la cantidad de actores que interactúan con el sistema, el tipo de actor que es (puede ser una persona u otro sistema que colabora con él) y la forma en que intervienen en la aplicación. Por su parte el factor de peso de los casos de uso sin ajustar se calcula teniendo en cuenta la cantidad de casos de uso existente y la complejidad en cuanto a cantidad de transacciones que posee cada uno de estos.

CAPITULO 5: ESTUDIO DE LA FACTIBILIDAD

5.2.1.1 Factor de peso de actores sin ajustar (UAW).

Tipo de actor	Descripción	Factor de peso	Cantidad de actores	Cantidad * peso
Simple	Otro sistema que interactúa con el nuestro a través de una interfaz grafica.	1	0	0*1
Medio	Otro sistema que interactúa con el nuestro a través de un texto.	2	0	0*2
Complejo	Persona que interactúa con el sistema mediante una interfaz grafica.	3	2	2*3
			Total	6

Tabla 1: Factor de peso de los actores sin ajustar.

El factor de peso de los actores sin ajustar es 6.

5.2.1.2 Factor de peso de los casos de uso sin ajustar (UUCW).

Tipo	Descripción	Peso	Cant*peso
Simple	El caso de uso tiene de 1 a 3 transacciones.	5	0*5
Medio	El caso de uso tiene de 4 a 7 transacciones.	10	2*10
Complejo	El caso de uso tiene más de 8 transacciones.	15	5*15
		Total	95

Tabla 5: Factor de peso de los casos de uso sin ajustar.

Luego se tiene:

$$UUCP = UAW + UUCW$$

$$UUCP = 6 + 95$$

$$UUCP = 101$$

5.2.2 Cálculo de puntos de casos de uso ajustados.

El valor de los puntos de casos de uso ajustados se calcula a través de la siguiente fórmula:

$$UCP = UUCP \times TCF \times EF \text{ donde:}$$

UCP: Puntos de casos de uso ajustados.

UUCP: Puntos de caso de uso sin ajustar.

CAPITULO 5: ESTUDIO DE LA FACTIBILIDAD

TCF: Factor de complejidad técnica

EF: Factor de ambiente.

El valor del factor de complejidad puede ser hallado cuantificando los factores que determinan la complejidad técnica que tendrá el sistema. Por su parte el factor ambiente tiene en cuenta las habilidades y los conocimientos que posee el grupo que desarrollará el software.

5.2.2.1 Factor de complejidad técnica.

Factor.	Descripción.	Peso	Valor	Σ (Peso_i * Valor_i)
T1	Sistema distribuido.	2	5	10
T2	Tiempo de respuesta.	1	4	4
T3	Eficiencia del usuario final.	1	1	1
T4	Procesamiento interno complejo.	1	1	1
T5	El código debe ser reutilizable.	1	5	5
T6	Facilidad de instalación.	0.5	2	1
T7	Facilidad de uso.	0.5	5	2.5
T8	Portabilidad.	2	5	10
T9	Facilidad de cambio.	1	2	2
T10	Concurrencia.	1	4	4
T11	Incluye objetivos especiales de seguridad.	1	3	3
T12	Provee acceso directo a terceras partes.	1	3	3
T13	Se requieren facilidades especiales de entrenamiento a usuarios.	1	1	1
Total				49.5

Tabla 2: Factor de complejidad técnica.

$$TCF = 0.6 + 0.01 \times \Sigma (\text{Peso}_i \times \text{Valor asignado}_i)$$

$$TCF = 0.6 + 0.01 \times 49.5$$

$$TCF = 1.095$$

5.2.2.2 Factor ambiente.

Factor.	Descripción.	Peso.	Valor	Σ (Peso_i * Valor_i)
E1	Familiaridad con el modelo de proyecto utilizado.	1.5	3	4.5

CAPITULO 5: ESTUDIO DE LA FACTIBILIDAD

E2	Experiencia en la aplicación.	0.5	0	0
E3	Experiencia en orientación a objetos.	1	4	4
E4	Capacidad del analista líder.	0.5	3	1.5
E5	Motivación.	1	5	5
E6	Estabilidad de los requerimientos.	2	3	6
E7	Personal part-time.	-1	0	0
E8	Dificultad del lenguaje de programación.	-1	4	-4
Total				17

Tabla 3: Factor ambiente.

$$EF = 1.4 - 0.03 \times \sum (\text{Peso}_i \times \text{Valor asignado}_i)$$

$$EF = 1.4 - 0.03 \times 17$$

$$EF = 0.89$$

Se tiene que:

$$UCP = UUCP \times TCF \times EF.$$

$$UCP = 101 \times 1.095 \times 0.89$$

$$UCP = 98.42955$$

5.2.3 Estimación de esfuerzo.

El esfuerzo a realizar puede ser hallado por la ecuación:

$$E = UCP \times CF$$

Donde:

UCP: Puntos de casos de uso ajustados.

CF: Factor de conversión.

5.2.3.1 Calculando el factor de conversión.

Para hallar el factor de conversión se debe contar cuantos factores de los que afectan al Factor Ambiente están por debajo de 3 para los valores de E1 a E6 mientras que para los valores de E7 y E8 se evalúa cuantos están por encima de 3.

CAPITULO 5: ESTUDIO DE LA FACTIBILIDAD

En caso de que el resultado sea menor o igual que 2 el CF seria 20 horas-hombre, en caso tal que este resultado sea 3 ó 4 se trabajaría con 28 horas-hombre, por el contrario si resulta mayor o igual que 5 lo mejor seria realizarle cambios al proyecto pues el riesgo del fracaso es muy elevado.

Cant EF <3(entre E1-E6)=2

Cant EF>3(E7 y E8)=0

Como el total de EF = 2 se debe trabajar con el factor de conversión 20 horas-hombre.

Ahora se puede calcular E como sigue:

$E=98.42955*20$ horas-hombre

$E=1968.591$ horas-hombre

$E\approx 1968.6$

5.2.4 Calcular el esfuerzo de todo el proyecto

El valor del esfuerzo hallado con anterioridad es el de la fase de implementación por lo que se puede sacar el valor de los demás y el esfuerzo total.

Actividad	% esfuerzo	Valor esfuerzo
Análisis	10%	492.15 horas-hombre
Diseño	20%	984.3 horas-hombre
Implementación	40%	1968.6 horas-hombre
Pruebas	15%	738.225 horas-hombre
Sobrecarga	15%	738.225 horas-hombre
Total	100%	4921.5 horas-hombre

Tabla 4: Esfuerzo total

Si la estimación total es 4921.5 horas-hombre y se calcula que un mes tiene aproximadamente 160 horas laborales suponiendo que se trabaje 8 horas diarias 5 días a la semana. Esto daría un aproximado de $E_t=30.76$ mes – hombre.

Esto quiere decir que una persona terminaría el proyecto en aproximadamente dos años y medio.

CAPITULO 5: ESTUDIO DE LA FACTIBILIDAD

Realizando un análisis del costo que podría tener el proyecto se tiene:

El salario promedio mensual es de \$100.

CH: Cantidad de hombres.

Tiempo: Tiempo total del proyecto.

CH = 2.

CHM= 2*Salario Promedio.

CHM= 2*100

CHM=200 \$/mes.

Costo = CHM * E_t / CH

Costo = 200 * 30.76 / 2

Costo = \$3076.

Tiempo = E_t / CH.

Tiempo = 30.76 / 2

Tiempo = 15.38 meses.

Concluyendo que:

Con los dos trabajadores que se encargan en la actualidad del proyecto este sería terminado en 1 año y 3 meses aproximadamente con un coste de \$3076.

5.3 Beneficios tangibles e intangibles.

El Telmonitor no es un sistema realizado con fines comerciales, su principal propósito consiste en mejorar el trabajo del personal del SubNodo1 en la universidad.

El principal beneficio que se consigue es una aplicación Web de interfaz agradable y dinámica que permita obtener de manera rápida la información de las llamadas que se realizan con el fin de mejorar el servicio telefónico dentro de la universidad.

Por esta razón los beneficios que se consiguen son intangibles:

CAPITULO 5: ESTUDIO DE LA FACTIBILIDAD

- Fácil acceso a la información actualizada.
- Disminución del tiempo empleado en gestionar la información de las llamadas telefónicas.

5.4 Análisis de costo.

El costo de una aplicación de software es evidente por lo que se hace preciso comprender los beneficios que traería su realización con el fin de ver si es viable su construcción o no. Existen dos tipos de beneficios que puede aportar un producto software: beneficios sociales y beneficios económicos. El Telmonitor es un software que aporta beneficios sociales puesto que será empleado en la universidad con el fin de mejorar el servicio telefónico dentro de este centro.

Una vez que este se encuentre en funcionamiento contribuirá a determinar los lugares donde es preciso implantar un plan con medidas preventivas con vistas a mejorar el servicio telefónico o evitar que éste se vea afectado por alguna razón.

No es preciso incurrir en gastos de licencias para su uso debido a que el sistema se encuentra desarrollado con tecnologías de código abierto. La aplicación es portable por lo que no causaría ningún inconveniente la necesidad de migrar a otra plataforma de trabajo.

Realizando un análisis de los costos y los beneficios que trae consigo se concluye que la realización del proyecto es factible.

5.5 Conclusiones.

En el desarrollo de este capítulo se pudo realizar un estimado de los costos que tendría la realización de la aplicación con respecto a los recursos y el dinero. También se realizó un análisis de los productos resultantes de esta realización y si esta resultaría factible o no.

CONCLUSIONES

CONCLUSIONES

Con la culminación de este trabajo de diploma se ha logrado cumplir con los objetivos propuestos. Se ha logrado una aplicación Web, la cual ha pasado por diferentes fases en su desarrollo. Se ha conseguido presentar una aplicación que cumple con lo definido por el cliente, con una gran rapidez a la hora de mostrar sus resultados; una aplicación dinámica que interactúa con el usuario, de esta forma se ha cumplido con las expectativas iniciales y se ha presentado un documento muy detallado que servirá de apoyo a las mejoras que se le quieran agregar a la aplicación realizada.

RECOMENDACIONES

RECOMENDACIONES

Al finalizar este trabajo quedan una serie de recomendaciones entre las que se encuentran: la realización de pruebas al sistema para verificar su correcto funcionamiento; que se integre el sistema realizado con la guía telefónica con el fin de dar a los usuarios la posibilidad de conocer el lugar físico donde se encuentra un número telefónico.

Se recomienda además, expandir el contenido de este trabajo, según las prestaciones de la planta telefónica de la UCI vayan creciendo, para lograr un Sistema Integrado y con mayores funcionalidades.

REFERENCIAS BIBLIOGRAFICAS

REFERENCIAS BIBLIOGRÁFICAS

- [1] Equipo de Netcommerce. Aplicaciones web, 2008. [Disponible en: <http://www.netcommerce.com.mx/desarrollo.asp>].
- [2] José Guillermo Valle, James Gildardo Gutiérrez. Definición de arquitectura cliente servidor, 2005. [Disponible en: <http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml#quees>]
- [3] Josep Casanovas. Artículo original en Alzado.org, Usabilidad y arquitectura del software, 31-07-2004. [Disponible en: <http://www.desarrolloweb.com/articulos/1622.php>]
- [4] Rubén Álvarez, Lenguajes de lado servidor o cliente. [Disponible en: <http://www.desarrolloweb.com/articulos/239.php>]
- [5] El equipo de ciberaula. 2006. [Disponible en: <http://www.ciberaula.com/curso/masterphp/>]
- [6] Joaquín Gracia Murugarren, Conceptos básicos, octubre 3 del 2004. [Disponible en: <http://www.webestilo.com/php/php00.phtml>]
- [7] Raúl Rodas Hinostroza. Características de PHP, febrero 22 2007. [Disponible en: <http://www.linuxcentro.net/linux/staticpages/index.php?page=CaracteristicasPHP>]
- [8] Aníbal de la Torre. Lenguajes del lado servidor o cliente, 2006. [Disponible en: [http://www.adelat.org/media/docum/nuke_publico/lenguajes del lado servidor o cliente.html](http://www.adelat.org/media/docum/nuke_publico/lenguajes_del_lado_servidor_o_cliente.html)]
- [9] Miguel Ángel Álvarez. Que es Java Script. [Disponible en: <http://www.desarrolloweb.com/articulos/25.php>]
- [10]
- [11] Miguel Ángel Álvarez. Zend Studio. [Disponible en: <http://www.desarrolloweb.com/articulos/1178.php>]
- [12] HTMLPOINT, 2006, Aspectos y características generales, [Disponible en: http://www.htmlpoint.com/javascript/corso/js_02.htm]
- [13] Proyecto SOBL Traducciones. Postgree SQL Practico, 2000. [Disponible en: <http://www.sobl.org/traduccion/practical-postgres/node19.html>]
- [14] Autor (es). Por que Postgre SQL, Año. [Disponible en: http://www.efaber.net/formacion/fp/curso_acs/3_a.html]
- [15] Jesús Vegas. El servidor web, 2002-03-21. [Disponible en: <http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node20.html>]

REFERENCIAS BIBLIOGRAFICAS

- [16] Yer. Tutorial de apache, servidor web, 09/05/2002. [Disponible en: <http://www.ayuda-internet.net/tutoriales/manu-apache/manu-apache.html>]
- [17] María A. Mendoza Sánchez (Ing. Informático – UNT, Microsoft Certified Professional – MCP, Analista y Desarrolladora - TeamSoft Perú S.A.C), Metodologías De Desarrollo De Software, Junio 7 del 2004, [Disponible en: [http://www.informatizate.net/articulos/metodologias de desarrollo de software 07062004.html](http://www.informatizate.net/articulos/metodologias%20de%20desarrollo%20de%20software%2007062004.html)]
- [18] Javier J. Gutiérrez. Que es un framework web. [Disponible en: [www.lsi.us.es/~javierj/investigacion ficheros/Framework.pdf](http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf)]
- [19] Maestros del Web, septiembre 11 del 2007, El framework Symfony, una introducción práctica (II parte), [Disponible en: <http://www.maestrosdelweb.com/editorial/el-framework-symfony-una-introduccion-practica-ii-parte/>]
- [20] Febe Ángel Ciudad Ricardo. Utilización del patrón Modelo Vista Controlador en el desarrollo de softwares educativos, Abril del 2006. [Disponible en: <http://www.monografias.com/trabajos43/patron-modelo-vista/patron-modelo-vista.shtml>]

BIBLIOGRAFIA

BIBLIOGRAFIA

- Alba Castro, Mauricio Fernando. "Calidad en la producción de software". En: Calidad de Software. Primer Congreso Nacional de Estudiantes de Ingeniería de Sistemas. Universidad de Manizales. Mayo 1992.
- Booch, Grady. "Object Oriented Design with Applications". Prentice Hall. 1991. 3. Coad, Peter y Yourdon, Edward. "Object Oriented Analysis". Prentice Hall. 1990.
- Dunn, Robert. "Software Quality". Prentice Hall. 1990. Cap 1. 5. Crosby, Philip. "Quality is Free". McGraw Hill. 1979.
- Humphrey, Watts S. "A discipline for software engineering". 1.999.
- MCGARRY, John. "Practical Software Measurement". Addison Wesley. 2001.
- Meyer, Bertrand. "Object Oriented Software Constructions". Englewood Cliffs, Prentice Hall. 1998.
- Pressman, Roger S. "Ingeniería del Software, Un enfoque practico". Tercera edición. McGraw Hill Cap 12.
- VILLALOBOS, Jorge. "La programación orientada por objetos: El paradigma del futuro". En Revista: Sistemas N. 48. 1991.

ANEXOS

ANEXOS

Anexo I

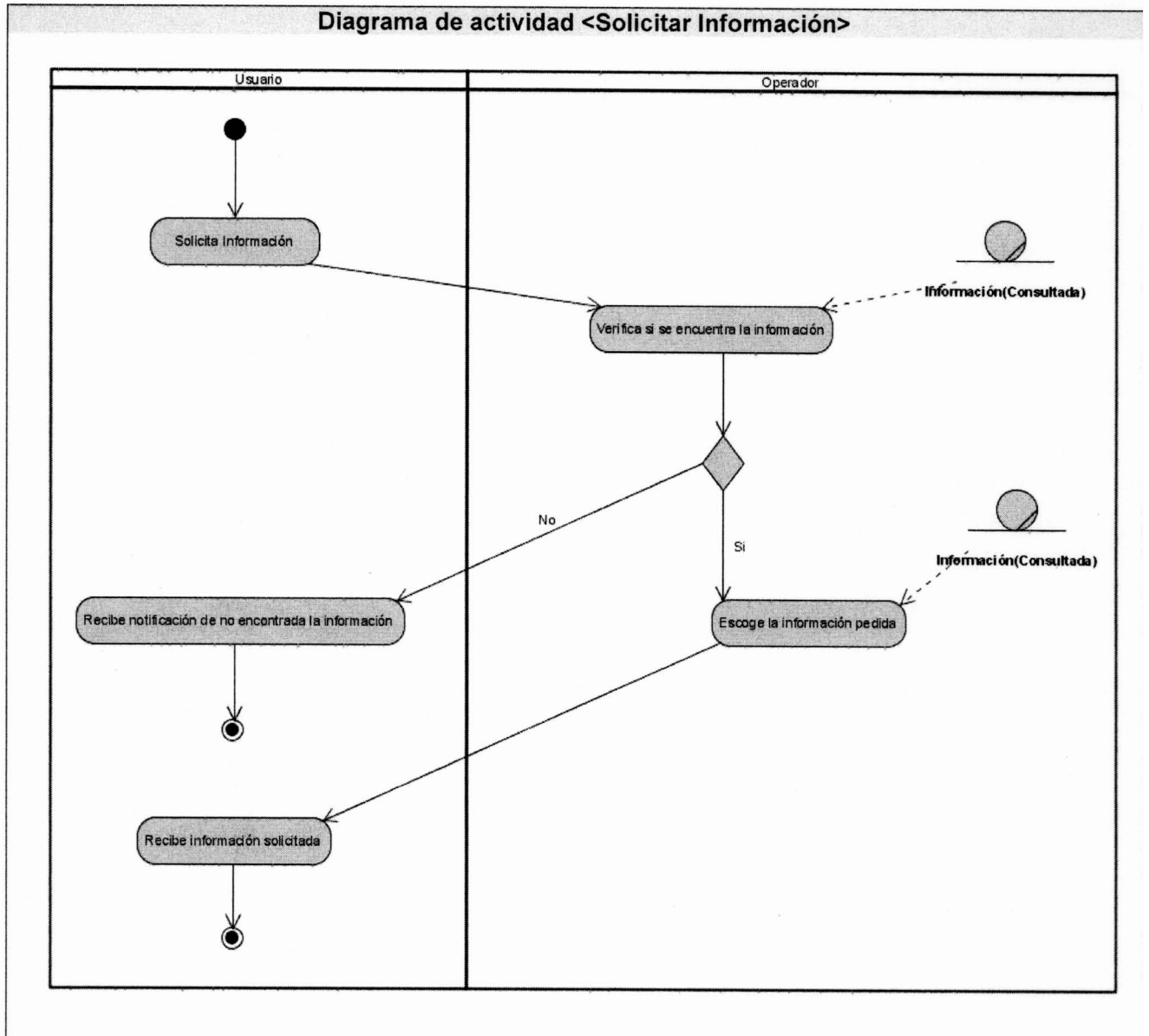
Caso de uso del negocio	Solicitar información.	
Actores	Usuario (inicia).	
Propósito	Este caso de uso tiene como objetivo principal que el usuario pueda pedir y recibir una información determinada que necesite en un momento dado.	
Resumen	El caso de uso inicia cuando el usuario solicita una determinada información relacionada con las llamadas telefónicas. Esta solicitud es hecha a la persona encargada del sistema encargado de archivarla. El operador; nombre que recibe esta persona, luego de buscar la información se la hace llegar al usuario finalizando de esta forma el caso de uso.	
Casos de uso asociados	-	
Curso normal de eventos		
Acción del actor	Respuesta del proceso de negocio	
<p>1. El usuario solicita información sobre llamadas telefónicas.</p> <p>5. El usuario recibe toda la información que fue a buscar.</p>	<p>2. El operador accede al sistema que guarda la información.</p> <p>3. El operador verifica que la información que le fue solicitada se encuentra almacenada en el sistema.</p> <p>4. El operador localiza la información que le fue solicitada.</p>	
Prioridad	Este caso de uso tiene máxima prioridad puesto que en él es donde se va a centrar todo el desarrollo del negocio.	
Otras secciones	<p>4. El administrador no encontró la información que buscaba.</p> <p>5. El usuario recibe la notificación de que la información buscada no se</p>	

ANEXOS

	encuentra archivada.
Mejoras propuestas	El usuario obtendrá una información más completa referente a las llamadas telefónicas y la probabilidad de que la información se pierda será casi nula.

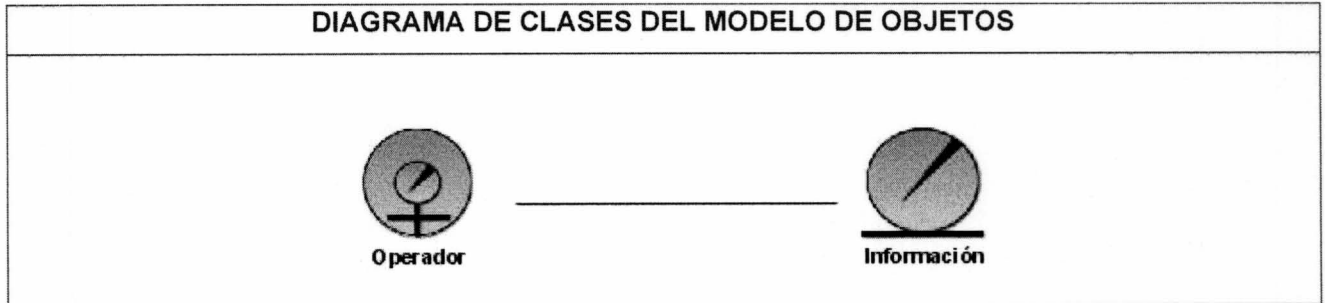
ANEXOS

Anexo II



ANEXOS

Anexo III



Anexo IV

Descripción textual de los casos de uso

Nombre del caso de uso	Autenticar usuario.
Actores	Operador (inicia).
Propósito	Garantizar que solo el personal autorizado acceda al sistema y que cada una de estas personas consulte solo la información a la que tiene permiso garantizando de esta manera la seguridad en el sistema.
Resumen: El caso de uso inicia cuando el usuario introduce su nombre de usuario y contraseña para acceder al sistema. El sistema por su parte, luego de verificar si los datos brindados son correctos le brinda el acceso al sistema según los privilegios que tiene dentro de este. Si los datos son incorrectos el sistema le brinda la oportunidad de volver a suministrarlos.	
Referencias	R4.
Precondiciones	
Poscondiciones	Se habilitan todas las funcionalidades que necesita el usuario autenticado para trabajar con el sistema.
Curso normal de eventos	
Acciones del actor	Respuesta del sistema
1. El operador solicita la entrada al sistema.	2. El sistema muestra una interfaz que permita la introducción de los datos del usuario.

ANEXOS

<p>3. El operador entra usuario y contraseña y elige la opción "Autenticarse".</p>	<p>4. Verifica que el usuario y contraseña entrados sean correctos.</p> <p>5. Le brinda acceso al sistema con los privilegios que posee.</p>
<p>Curso alterno</p>	
<p>Línea 5 a. Campos vacíos.</p>	
<p>5a.1. El operador deja campos sin llenar.</p> <p>5a. 3. Retornar al paso 3.</p>	<p>5a. 2. El sistema muestra un mensaje informando que hay campos vacíos.</p>
<p>Línea 5 b. Datos incorrectos.</p>	
<p>5b. 1. El operador introduce datos incorrectos al sistema.</p>	<p>5b. 2. El sistema muestra una interfaz con un mensaje informando que los datos ingresados son incorrectos.</p>
	
<p>Prioridad</p>	<p>Crítico.</p>

<p>Nombre del caso de uso</p>	<p>Gestionar operador.</p>
<p>Actores</p>	<p>Administrador (inicia).</p>
<p>Propósito</p>	<p>Tener un control de las personas que tienen acceso al sistema y los privilegios que tienen dentro de este.</p>

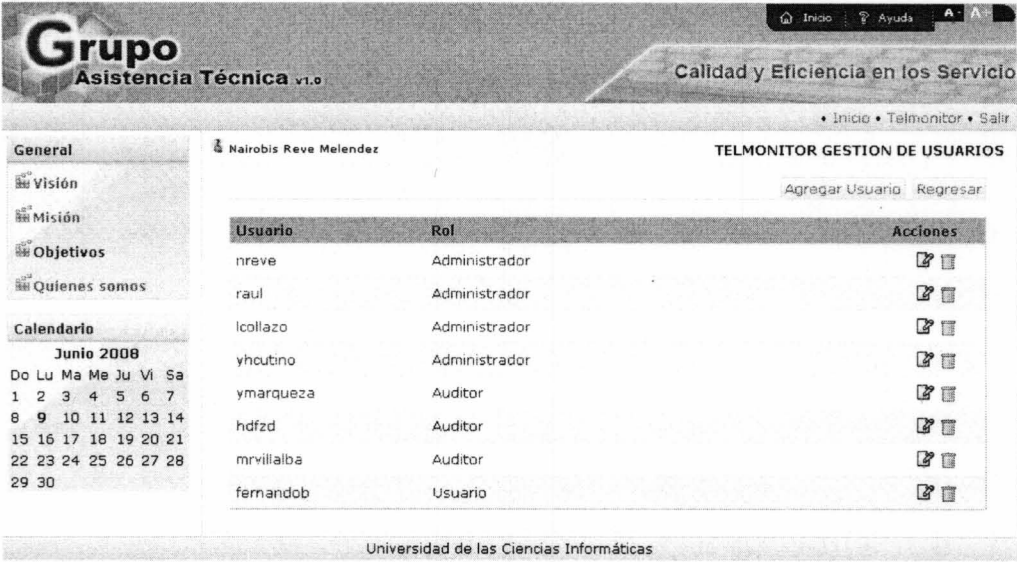
ANEXOS

Resumen: El caso de uso inicia cuando el administrador solicita la gestión de los usuarios. Aquí el usuario tiene la oportunidad de adicionar un nuevo usuario al sistema, modificarlo o eliminarlo totalmente.	
Referencias	R1, R2, R3.
Precondiciones	El administrador se encuentra autenticado en el sistema y tiene los privilegios que le corresponden.
Poscondiciones	Los datos de los usuarios en la base de datos se han actualizado.
Curso normal de eventos	
Acciones del actor	Respuesta del sistema
1. El administrador escoge la opción "Gestión de usuarios" del menú.	2. El sistema muestra una interfaz con el listado de los usuarios y los privilegios que tienen dentro del sistema, conjuntamente muestra una serie de opciones que puede realizar: 2.1. Agregar usuario, ver sección "Adicionar usuario". 2.2. Editar usuario, ver sección "Modificar usuario". 2.3. Eliminar usuario, ver sección "Eliminar usuario".
Curso alterno	
Línea 2 a. No hay usuarios registrados en el sistema.	
2a.1. El administrador no ha registrado usuarios en el sistema.	2a.2. El sistema muestra un mensaje indicando que no hay usuarios registrados en el sistema y retorna al paso 2.
Sección "Adicionar usuario"	
3. El administrador escoge la opción "Agregar usuario".	4. El sistema muestra una interfaz que permite introducir los datos del nuevo usuario.
5. El administrador introduce los datos del nuevo operador y selecciona la opción "Agregar".	6. El sistema valida los datos de entrada.
8. El administrador elige la opción	7. El sistema almacena la información en su base de datos y muestra un mensaje informando que la operación tuvo éxito.
	9. Retornar al paso 2 finalizando así el caso de uso.

ANEXOS

"Continuar".	
Curso alternativo	
Línea 5a. Operación cancelada.	
5a. 1. El administrador escoge la opción "Cancelar".	5a. 2. Retornar al paso 2 finalizando así el caso de uso.
Línea 5 b. Datos incorrectos.	
5b.1. El administrador introduce datos incorrectos.	5b.2. El sistema muestra una interfaz con un mensaje indicando que existen datos incorrectos.
5b.3. El administrador escoge la opción "Continuar".	5b.4. Retornar al paso 4.
Línea 5b.3. Ignorar opción "Continuar".	
5b.3.1. El administrador ignora la opción continuar.	5b.3.2. Retornar al paso 4.
Línea 5c. Campos vacíos.	
5c.1. El administrador deja campos vacíos.	5c.2. El sistema muestra una interfaz con un mensaje informando que ha dejado campos vacíos.
5c.3. Retornar al paso 5.	
Línea 8a. Ignorar opción "Continuar".	
8a.1. El administrador ignora la opción "Continuar".	8a.2. Retornar al paso 2 y finaliza el caso de uso.
Sección "Modificar usuario"	
3. El administrador escoge la opción "Editar usuario".	4. El sistema muestra una interfaz con los datos del usuario a modificar.
5. El administrador cambia los datos que necesite y escoge la opción "Actualizar".	6. El sistema actualiza la información en la base de datos y muestra una interfaz con un mensaje informando el éxito de la operación.
7. El administrador escoge la opción "Continuar".	8. Retornar al paso 2 y finaliza el caso de uso.
Curso alternativo	
Línea 5a. Operación cancelada.	
5a.1. El administrador escoge la opción	5a.2. Retornar al paso 2 y finaliza así el caso de uso.

ANEXOS

"Cancelar".	
Línea 7a. Ignorar Continuar.	
7a.1. El administrador ignora la opción "Continuar".	7a.2. Retornar al paso 2 y finaliza el caso de uso.
Sección "Eliminar usuario"	
3. El administrador escoge la opción "Eliminar usuario".	4. El sistema muestra un mensaje de confirmación para eliminar al usuario.
5. El administrador escoge la opción "Aceptar".	6. El sistema elimina el usuario de su base de datos y retorna al paso 2 acabando así el caso de uso.
Curso alterno	
Línea 5 a. Cancelar operación.	
5a.1. El usuario escoge la opción "Cancelar".	5 a.2. El sistema cancela la operación y retorna al paso 2.
	
Prioridad	Crítico

Nombre del caso de uso	Visualizar reporte de tráfico.
Actores	Operador (inicia).
Propósito	Obtener informe detallado del tráfico telefónico.

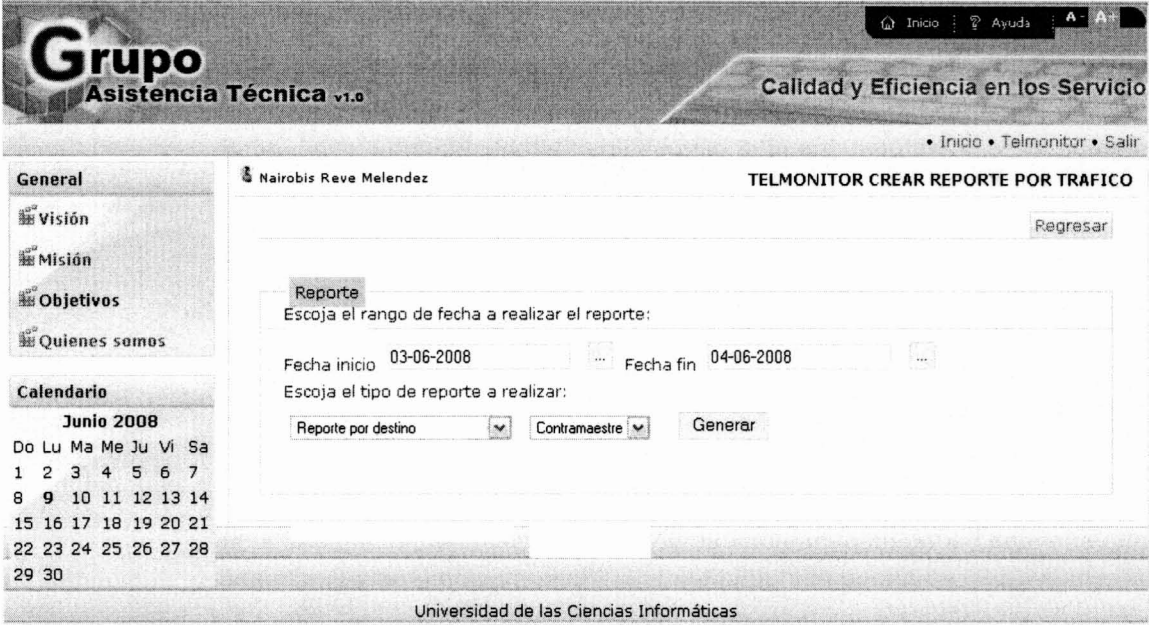
ANEXOS

Resumen: El caso de uso inicia cuando el operador decide ver el comportamiento del tráfico telefónico dentro de la universidad. El operador brinda los detalles del tipo de reporte que desea y el sistema le brinda esta información finalizando de esta manera el caso de uso.	
Referencias	R10 – R33.
Precondiciones	El operador se encuentra autenticado en el sistema.
Poscondiciones	
Curso normal de eventos	
Acciones del actor	Respuesta del sistema
1. El operador escoge la opción "Crear reporte por trafico" del menú.	2. El sistema muestra una interfaz donde se posibilita la entrada de los datos del reporte.
3. El operador escoge los datos del reporte a realizar y selecciona la opción "Generar".	4. El sistema busca en la base de datos y muestra en una tabla toda la información encontrada que coincida con los datos entrados y se habilitan una serie de opciones para el operador: 4.1. Imprimir, ver sección "Imprimir reporte". 4.2. Exportar, ver sección "Exportar reporte". 4.3. Graficar, ver sección "Graficar reporte". 4.4. Enviar por correo, ver sección "Enviar por correo". 4.5. Filtrar, ver sección "Filtrar reporte". 4.6. Ordenar ascendentemente, ver sección "Ordenar ascendentemente". 4.7. Ordenar descendentemente, ver sección "Ordenar descendentemente".
Curso alterno	
Línea 4a. No hay llamadas registradas.	
4a.1. El usuario introduce una información de la cual no se encuentra nada archivado en el sistema.	4a.2. El sistema muestra una interfaz con un mensaje indicando que no hay llamadas registradas que cuenten con las características especificadas y finaliza el caso de uso.
Sección "Imprimir reporte".	
5. El operador escoge la opción de	6. El sistema muestra una interfaz para imprimir la página.

ANEXOS

“Imprimir”.	
7. El operador escoge la opción “Imprimir”.	8. El sistema da al dispositivo la orden de imprimir finalizando así el caso de uso.
Curso alterno	
Línea 7a. Cancelar operación.	
7a.1. El operador escoge la opción de “Cancelar”.	7a.2. El sistema cancela la operación y se retorna al paso 4.
Sección “Exportar reporte”.	
5. El operador escoge la opción “Exportar”.	6. El sistema muestra una interfaz que permita exportar.
7. El operador especifica los datos a entrar y selecciona la opción “Aceptar”.	8. El sistema guarda los datos especificados finalizando el caso de uso.
Curso alterno	
Línea 7a. Cancelar operación.	
7a.1. El operador escoge la opción de “Cancelar”.	7a.2. El sistema cancela la operación y regresa al paso 4.
Sección “Graficar reporte”.	
5. El operador escoge la opción “Graficar”.	6. El sistema muestra una interfaz con la gráfica.
Curso alterno	
Línea 7a. Cancelar operación.	
7a.1. El operador escoge la opción de “Cancelar”.	7a.2. El sistema cancela la operación de graficar.
Sección “Enviar por correo”.	
5. El operador escoge la opción “Enviar por correo”.	6. El sistema muestra una interfaz para el envío de correo.
7. El operador envía el correo finalizando así el caso de uso.	
Sección “Filtrar reporte”.	
5. El operador escoge la opción “Filtrar”.	6. El sistema muestra una interfaz con los datos del filtro

ANEXOS

	realizado y brinda la opción de: Limpiar, ver sección "Limpiar filtro".
Sección "Limpiar filtro".	
7. El operador escoge la opción "Limpiar".	8. Retorna al paso 4.
Sección "Ordenar ascendentemente".	
5. El operador escoge la opción "Ordenar ascendentemente".	6. El sistema le muestra la misma información pero ordenada por la duración de la llamada de menor a mayor.
Sección "Ordenar descendentemente".	
5. El operador escoge la opción "Ordenar descendentemente".	6. El sistema le muestra la misma información pero ordenada por la duración de la llamada de mayor a menor.
	
Prioridad	Crítico

Nombre del caso de uso	Visualizar reporte de costo.
Actores	Operador (inicia).
Propósito	Obtener informe detallado del comportamiento del costo telefónico.
Resumen:	El caso de uso inicia cuando el operador decide ver el comportamiento de los costos

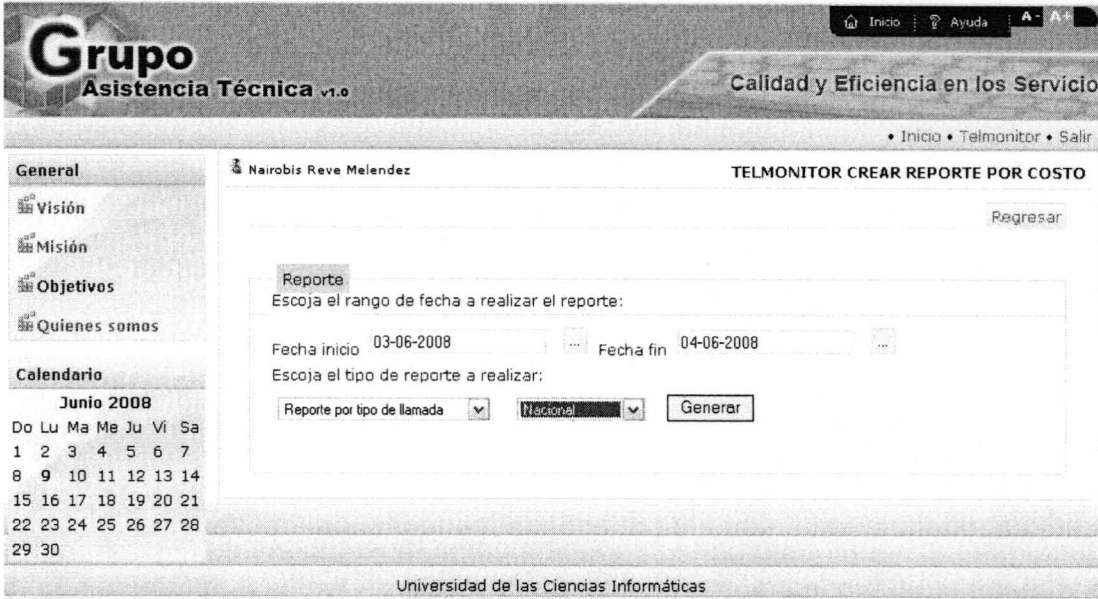
ANEXOS

<p>telefónicos dentro de la universidad. El operador brinda los detalles del tipo de reporte que desea y el sistema le brinda esta información finalizando de esta manera el caso de uso.</p>	
Referencias	R34 – R53.
Precondiciones	
Poscondiciones	
Curso normal de eventos	
Acciones del actor	Respuesta del sistema
<p>1. El operador escoge la opción "Crear reporte por costos" del menú.</p> <p>3. El operador escoge los datos del reporte a realizar y selecciona la opción "Generar".</p>	<p>2. El sistema muestra una interfaz donde se posibilita la entrada de los datos del reporte.</p> <p>4. El sistema busca en la base de datos y muestra en una tabla toda la información encontrada que coincida con los datos entrados y se habilitan una serie de opciones para el operador:</p> <p>4.1. Imprimir, ver sección "Imprimir reporte".</p> <p>4.2. Exportar, ver sección "Exportar reporte".</p> <p>4.3. Graficar, ver sección "Graficar reporte".</p> <p>4.4. Enviar por correo, ver sección "Enviar por correo".</p> <p>4.5. Filtrar, ver sección "Filtrar reporte".</p> <p>4.6. Ordenar ascendentemente, ver sección "Ordenar ascendentemente".</p> <p>4.7. Ordenar descendentemente, ver sección "Ordenar descendentemente".</p>
Curso alterno	
<p>Línea 4a. No hay llamadas registradas.</p>	
	<p>4a.1. El sistema muestra un mensaje indicando que no hay llamadas registradas que cuenten con las características especificadas y finaliza el caso de uso.</p>
Sección "Imprimir reporte".	
<p>5. El operador escoge la opción de "Imprimir".</p> <p>7. El operador escoge la opción</p>	<p>6. El sistema muestra una interfaz para imprimir la página.</p> <p>8. El sistema da al dispositivo la orden de imprimir</p>

ANEXOS

"Imprimir".	finalizando así el caso de uso.
Curso alterno	
Línea 7a. Cancelar operación.	
7a.1. El operador escoge la opción de "Cancelar".	7a.2. El sistema cancela la operación.
Sección "Exportar reporte".	
5. El operador escoge la opción "Exportar".	6. El sistema muestra una interfaz para exportar.
7. El operador especifica los datos a entrar y selecciona la opción "Aceptar".	8. El sistema guarda los datos especificados finalizando el caso de uso.
Curso alterno	
Línea 7a. Cancelar operación.	
7a.1. El operador escoge la opción de "Cancelar".	7a.2. El sistema cancela la operación.
Sección "Graficar reporte".	
5. El operador escoge la opción "Graficar".	6. Muestra una interfaz con el gráfico realizado finalizando así el caso de uso.
Curso alterno	
Línea 7a. Cancelar operación.	
7a.1. El operador escoge la opción de "Cancelar".	7a.2. El sistema cancela la operación.
Sección "Enviar por correo".	
5. El operador escoge la opción "Enviar por correo".	6. El Sistema le muestra una interfaz que permite el envío de correo.
7. El operador envía el correo finalizando así el caso de uso.	
Sección "Filtrar reporte".	
5. El operador escoge la opción "Filtrar".	6. El sistema muestra una interfaz con los datos del filtro realizado y brinda la opción de: Limpiar, ver sección "Limpiar filtro".
Sección "Limpiar filtro".	

ANEXOS

7. El operador escoge la opción "Limpiar".	8. El sistema muestra la información que tenía antes de realizar el filtro.
Sección "Ordenar ascendentemente".	
5. El operador escoge la opción "Ordenar ascendentemente".	6. El sistema le muestra la misma información pero ordenada por la duración de la llamada de menor a mayor.
Sección "Ordenar descendentemente".	
5. El operador escoge la opción "Ordenar descendentemente".	6. El sistema le muestra la misma información pero ordenada por la duración de la llamada de mayor a menor.
	
Prioridad	Crítico.

Nombre del caso de uso	Crear informe resumen.(ok)
Actores	Operador(inicia)
Propósito	Obtener un resumen general del comportamiento del tráfico telefónico.
Resumen: El caso de uso comienza cuando el usuario del sistema decide crear un informe que le resuma el comportamiento del tráfico telefónico en un período determinado; período que puede variar en dependencia de las necesidades del usuario. El caso de uso finaliza cuando el usuario recibe un documento donde se encuentra esta información reflejada.	

ANEXOS

Referencias	R57.	
Precondiciones		
Poscondiciones	Se obtendrá un documento donde se resuma el comportamiento del tráfico telefónico.	
Curso normal de eventos		
Acciones del actor	Respuesta del sistema	
1. El operador escoge la opción "Informe resumen" del menú.	2. El sistema muestra una interfaz que le proporciona al operador introducir el rango de fecha en que quiere enmarcar su resumen.	
3. El operador introduce el rango de fecha y escoge la opción "Crear".	4. El sistema muestra una interfaz con un resumen de la información almacenada en el rango de fecha especificado y se habilita la opción: Imprimir, ver sección "Imprimir informe resumen".	
Curso alterno		
Línea 3a. Operación cancelada.		
3a.1.El operador escoge la opción "Cancelar".	3a.2. El sistema cancela la operación.	
Sección "Imprimir informe resumen."		
5. El operador escoge la opción "Imprimir".	6. El sistema muestra una ventana de impresión.	
7. El operador escoge la opción "Aceptar".	8. El sistema envía la orden de imprimir al dispositivo.	

ANEXOS

Grupo de Asistencia Técnica v1.0

Calidad y Eficiencia en los Servicios

Inicio • Telmonitor • Salir

General

- Visión
- Misión
- Objetivos
- Quienes somos

Calendario

Junio 2008

Do	Lu	Ma	Me	Ju	Vi	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

Fecha inicio 01-06-2008 Fecha fin 11-06-2008

Generar

INFORME RESUMEN

Regresar

Escija el rango de fecha a realizar el informe:

Universdad de l

Listo Intranet local

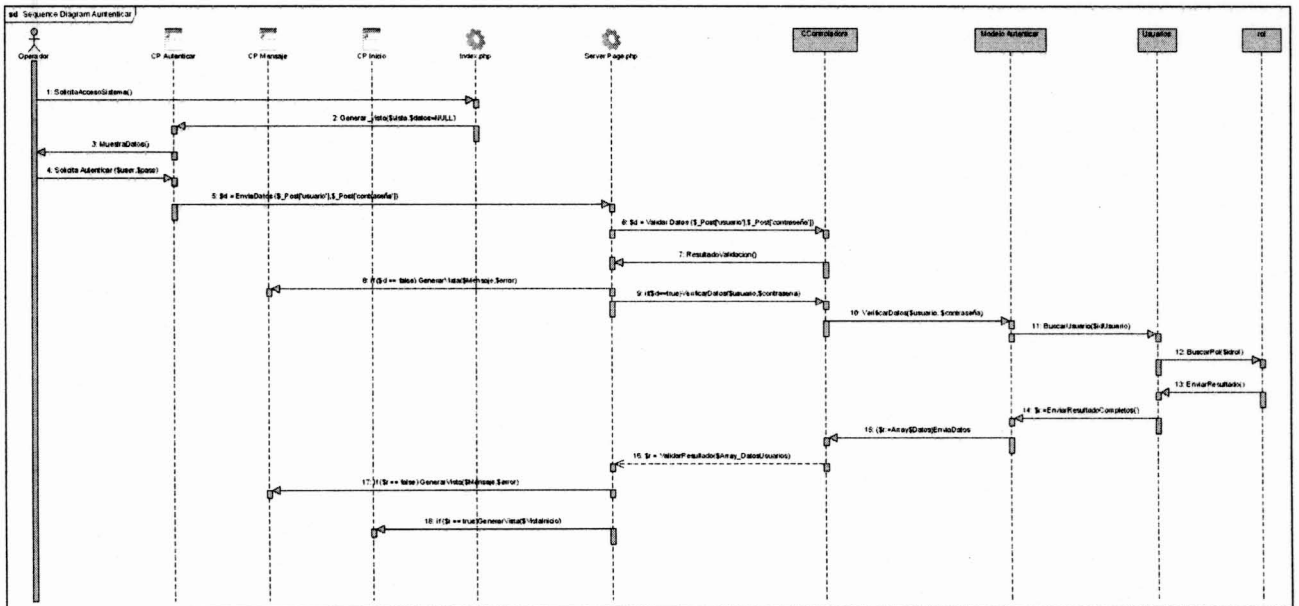
Prioridad Crítico

ANEXOS

Anexo V.

Diagramas de secuencia del diseño:

Diagrama de secuencia caso de uso Autenticar.



ANEXOS

Diagrama de secuencia Caso de uso Gestionar operador, Adicionar operador.

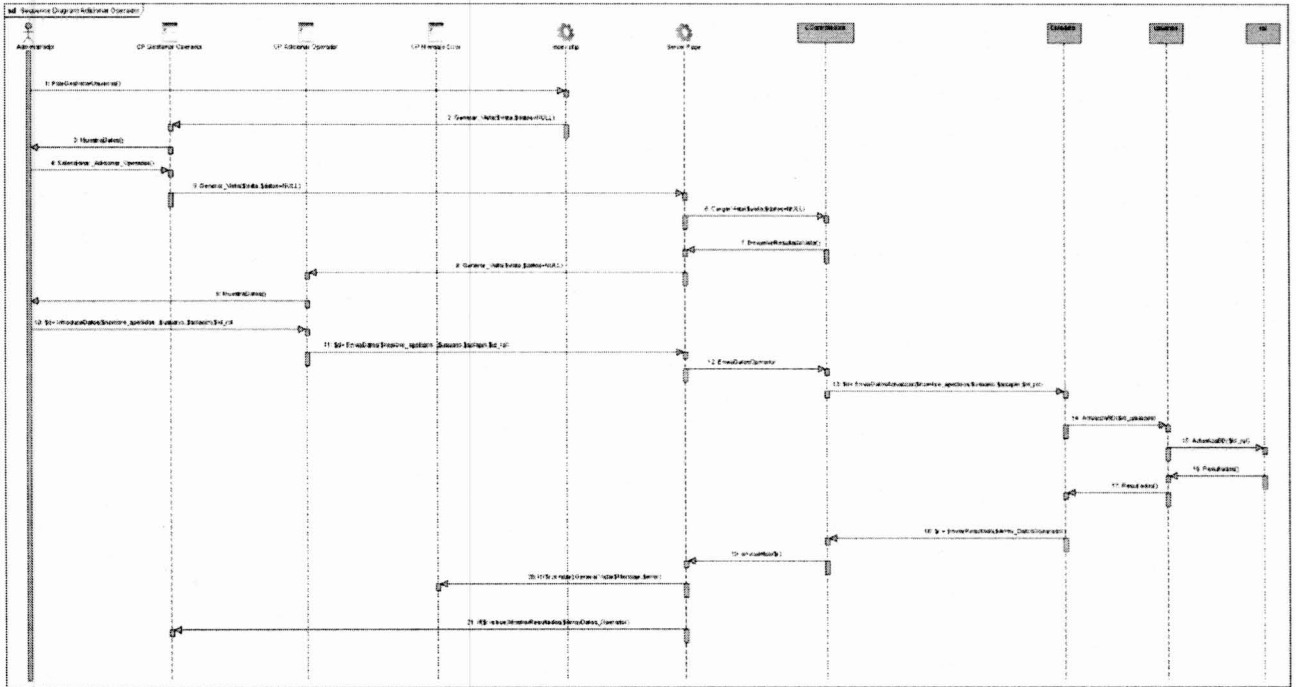
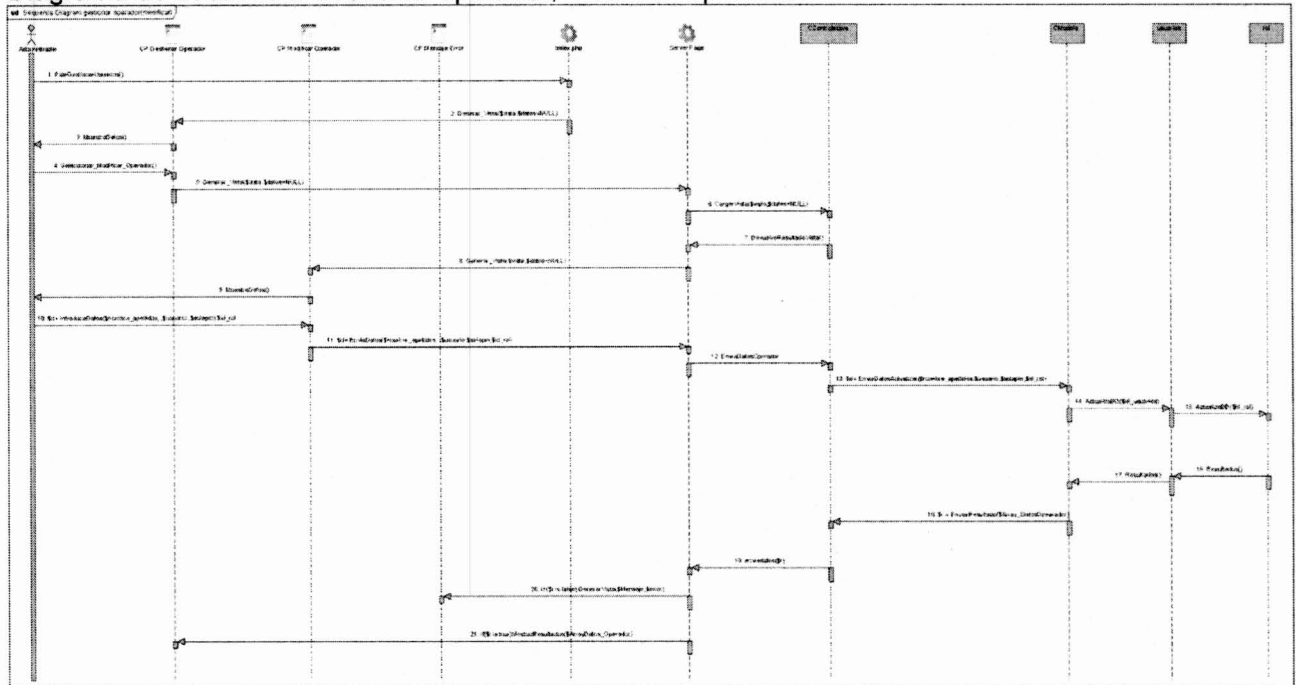


Diagrama de secuencia Gestionar operador, Modificar operador.



ANEXOS

Diagrama de secuencia caso de uso Gestionar operador, Eliminar operador.

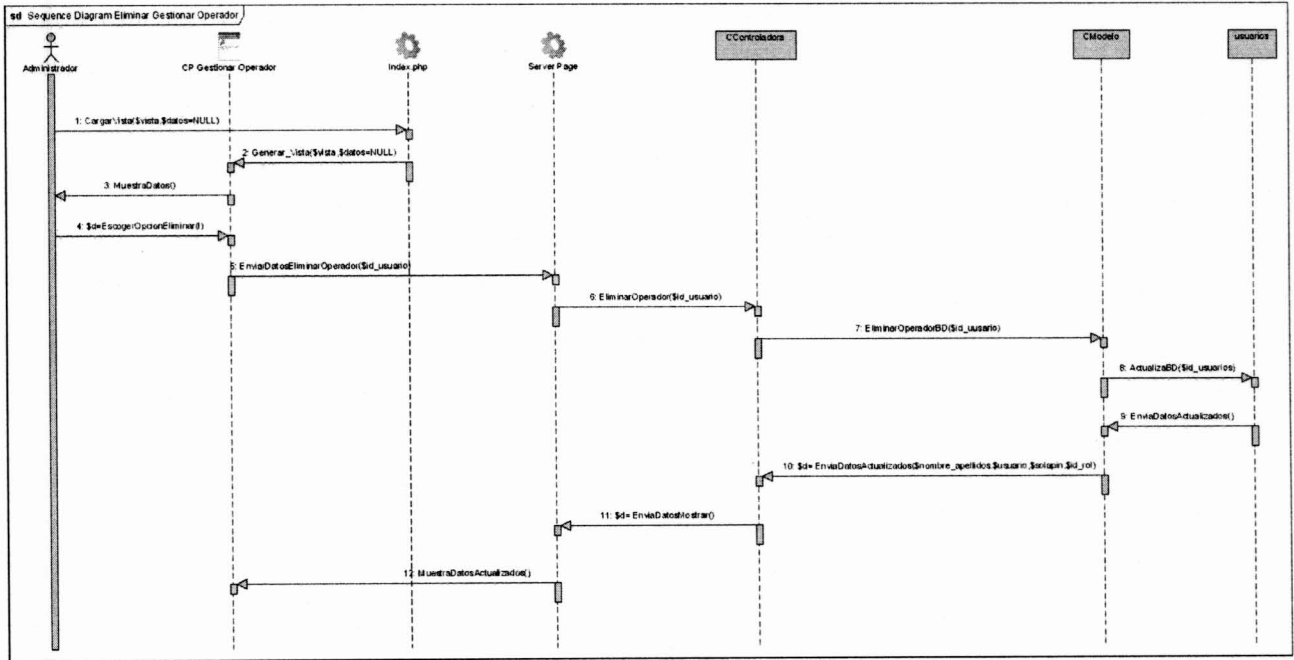
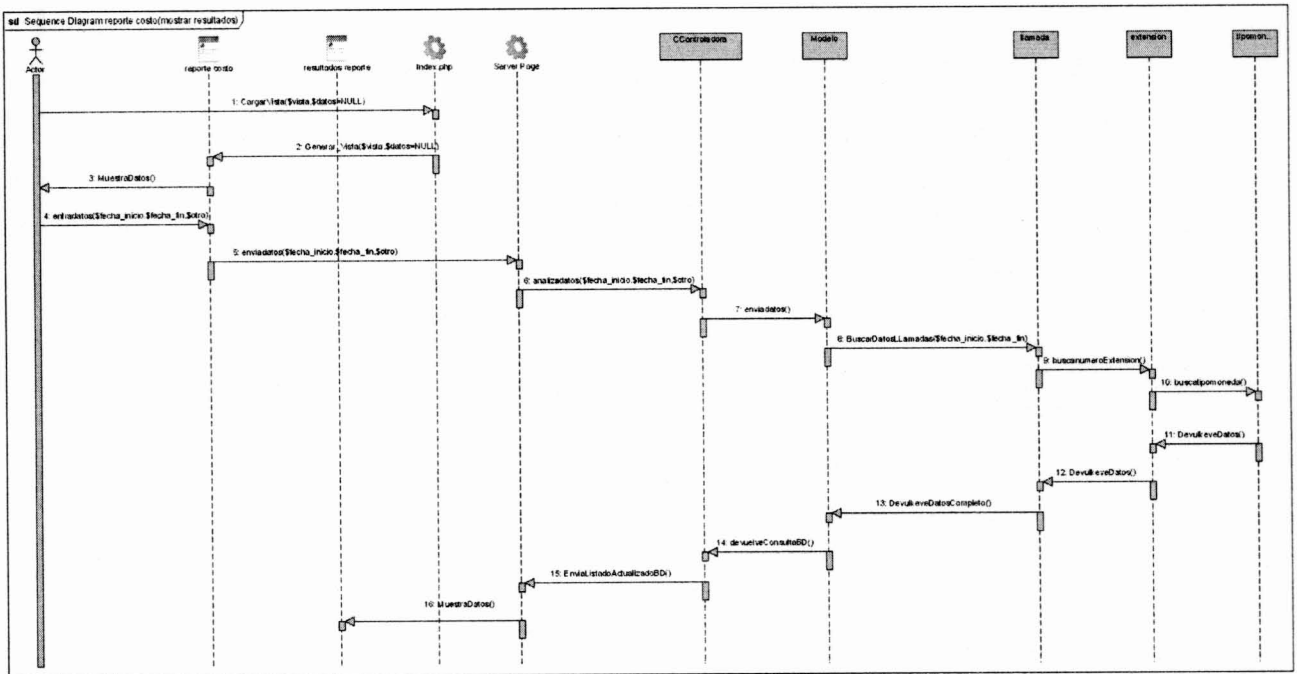


Diagrama de secuencia Visualizar reporte de costo, Mostrar reporte.



ANEXOS

Diagrama de secuencia Visualizar reporte de costo, Enviar por correo.

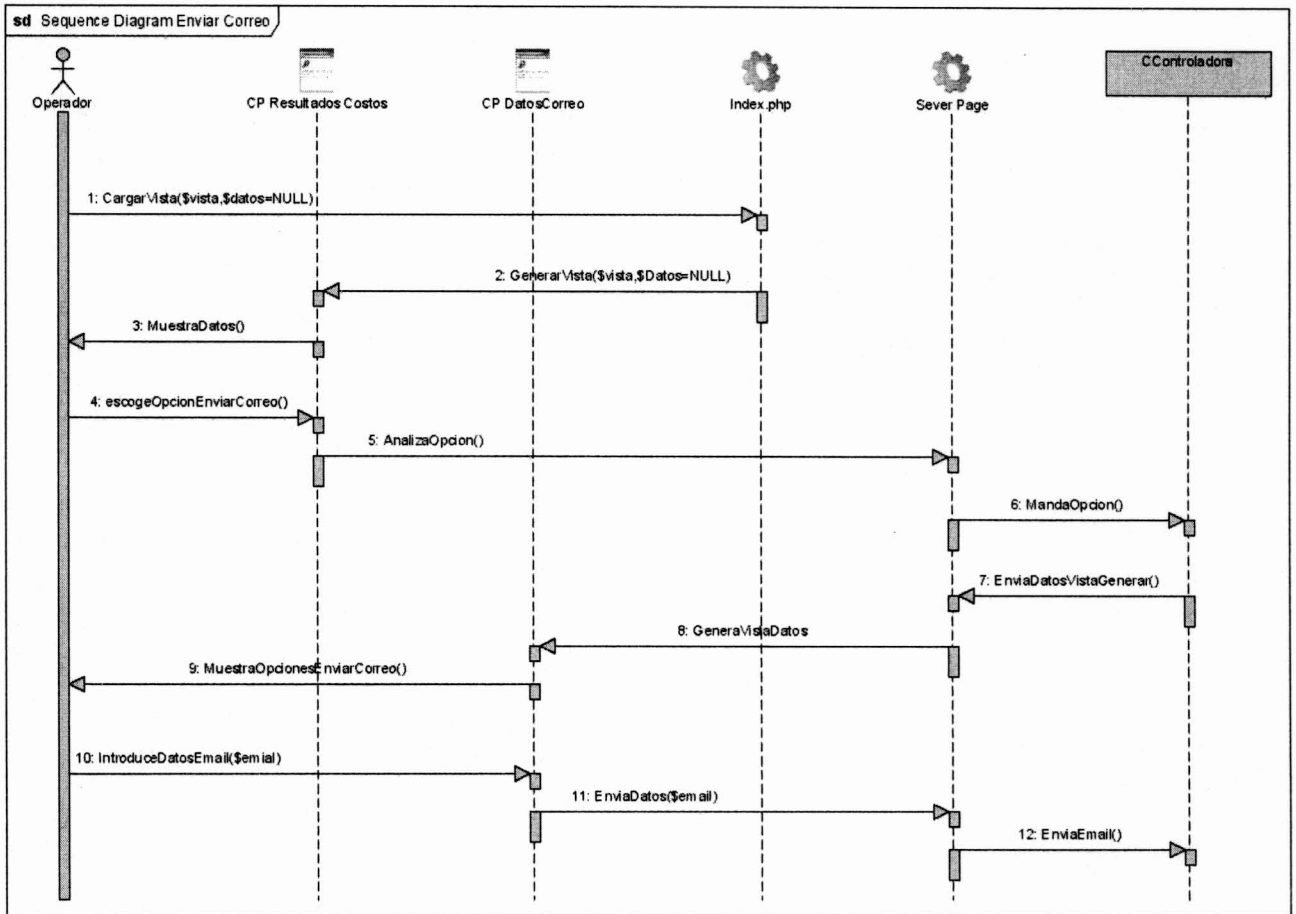
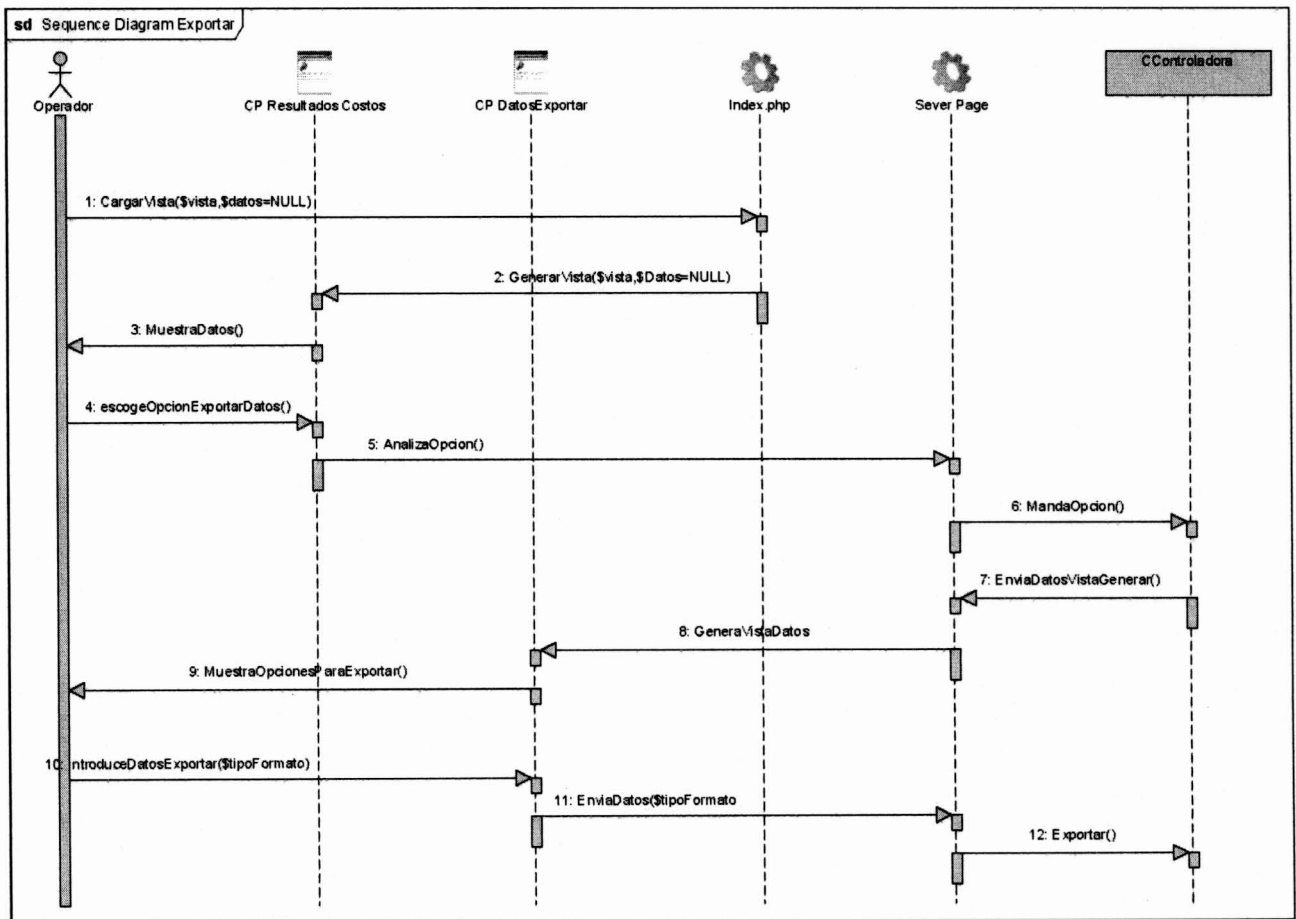


Diagrama de secuencia Visualizar reporte de costo, Exportar.

ANEXOS



ANEXOS

Diagrama de secuencia Visualizar reporte de costo, Filtrar.

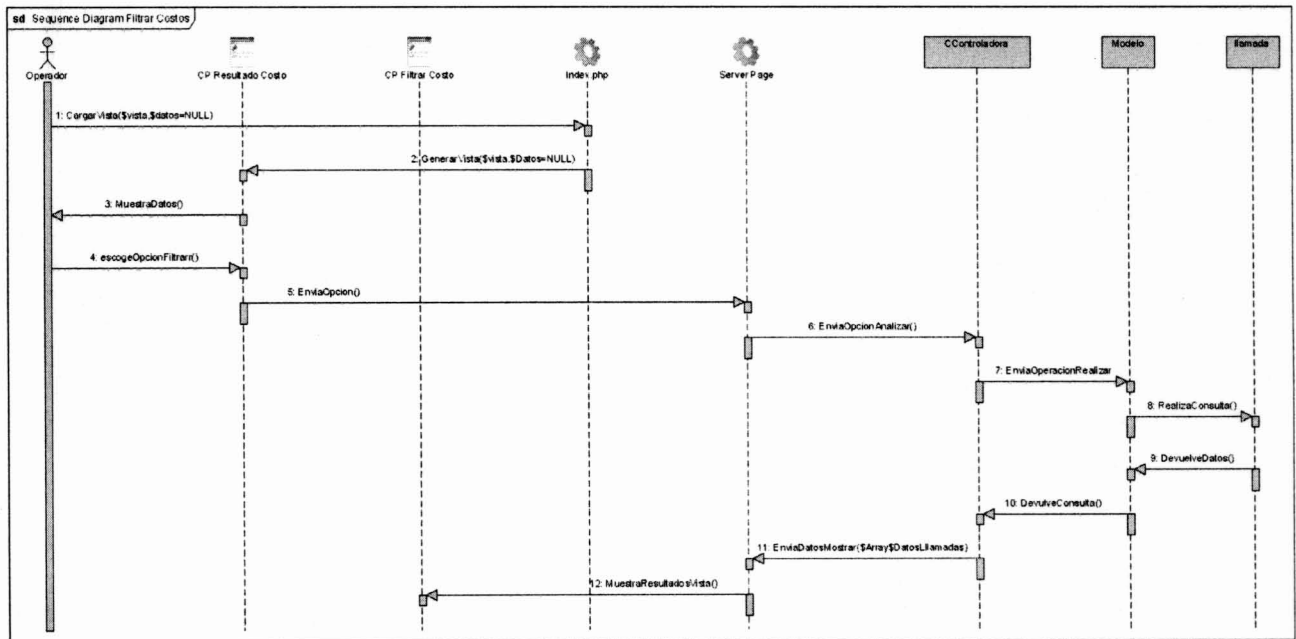
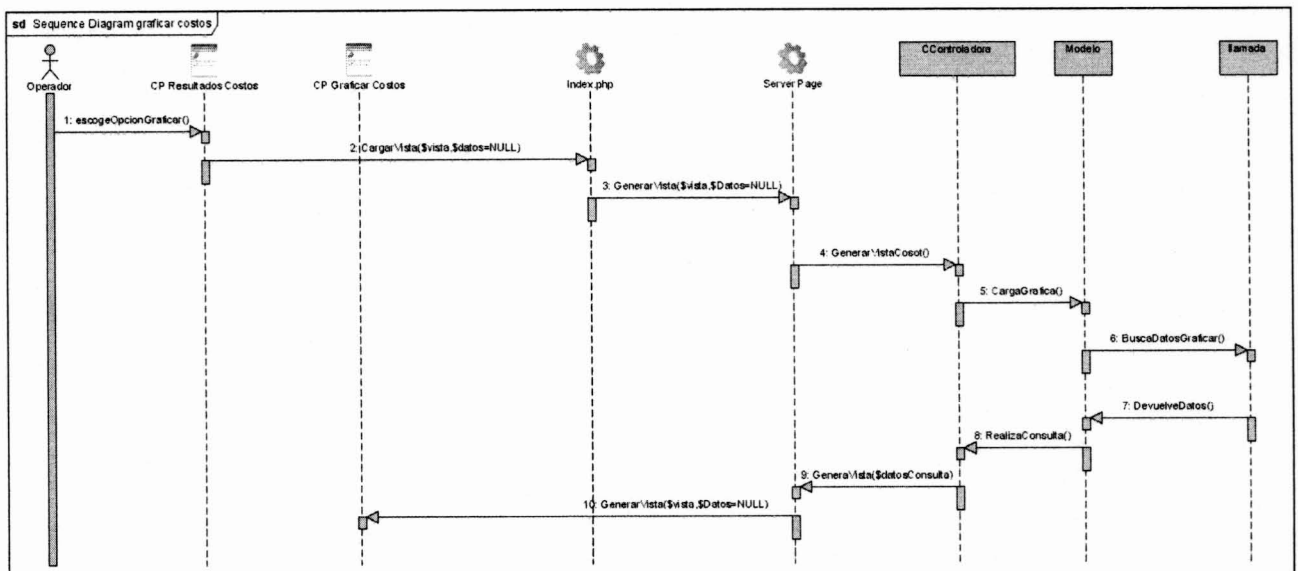
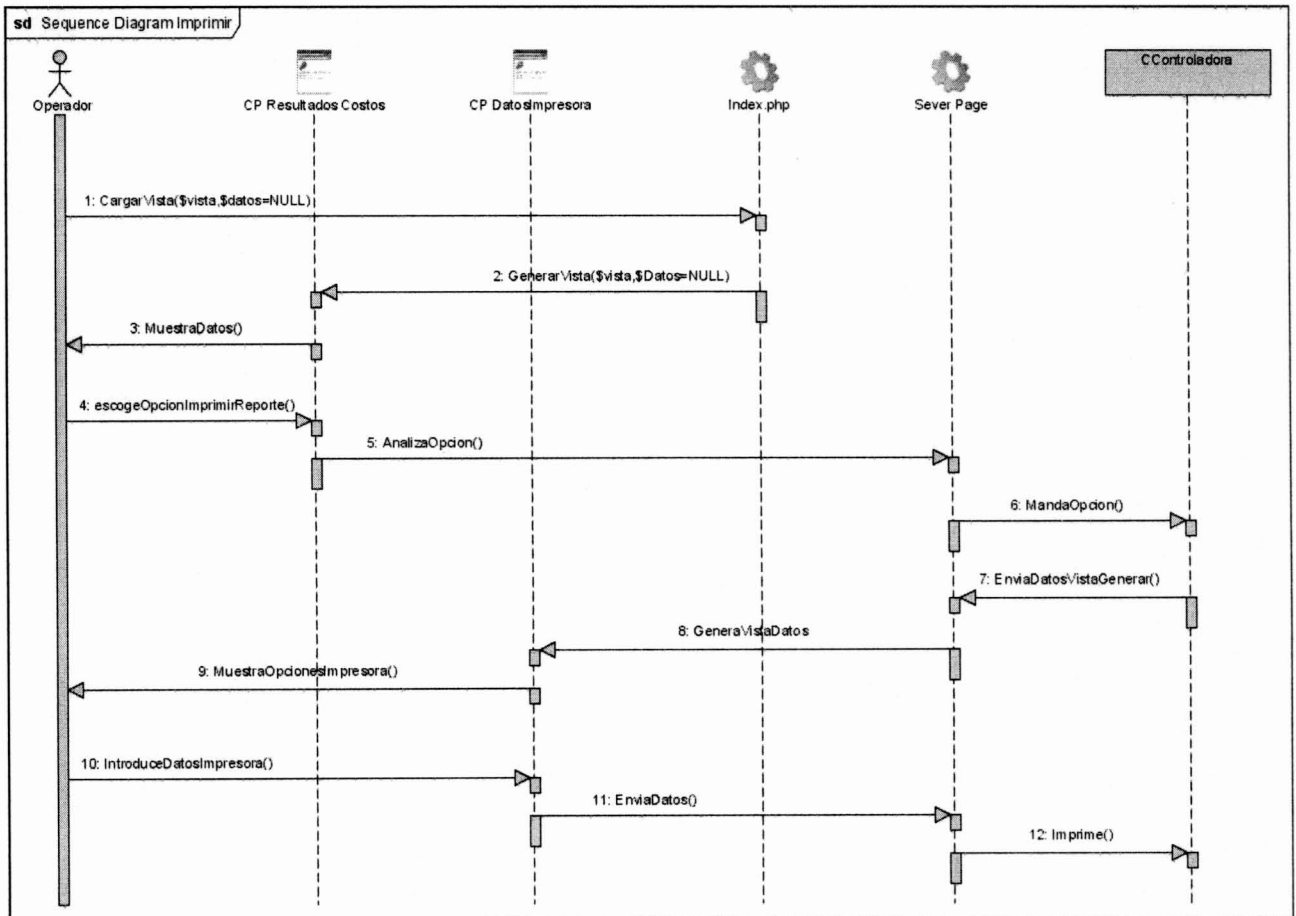


Diagrama de secuencia Visualizar reporte de costo, Graficar.



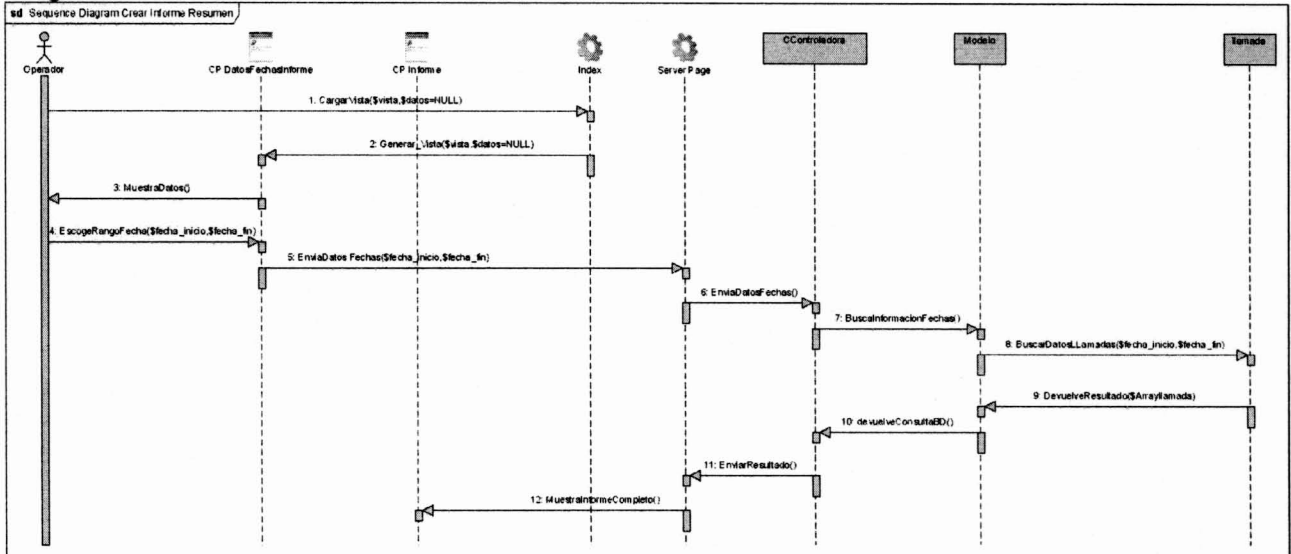
ANEXOS

Diagrama de secuencia Visualizar reporte de costo, Imprimir.



ANEXOS

Diagrama de secuencia Crear informe resumen.



GLOSARIO DE TERMINOS

GLOSARIO DE TERMINOS

Actor: Subsistemas o clases que interactúan directamente con el sistema. Un actor participa en un caso de uso o en conjunto de casos de usos para llevar a cabo un propósito.

Caso de uso: Secuencia de acciones que un sistema ejecuta y que produce un resultado observable para un actor.

Código abierto (Open source): Es el término con el que se conoce al software distribuido y desarrollado libremente. El significado de código abierto es que se puede tener acceso al código fuente, lo cual es un criterio más débil y flexible que el del software libre; un programa de código abierto puede ser software libre.

Diagrama: Es la representación gráfica de un conjunto de elementos. Visualiza un sistema desde diferentes perspectivas.

Framework: En el desarrollo de software es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Un framework puede incluir soporte de programas, bibliotecas para ayudar a desarrollar y unir los elementos que componen un proyecto determinado.

Interfaz: Conjunto de operaciones que posee un nombre y que caracteriza el comportamiento de un elemento.

Lenguaje de Programación: Consiste en un conjunto de reglas sintácticas y semánticas que definen un lenguaje informático.

Tráfico de llamadas: Flujo telefónico que existe en un lugar determinado. Representa a todas las llamadas que entran y salen de un lugar a través de una pizarra telefónica.

Extensión: Número telefónico asignado a un dispositivo físico.

Línea telefónica: Cable que guía la conexión telefónica hasta donde se encuentra el dispositivo físico.

Conmutador: Caja controladora de líneas telefónicas. En teoría ejerce la función que anteriormente realizaba una telefonista, la función de hacer llegar una llamada que se realiza desde un número telefónico hasta su destino.