

**Universidad de las Ciencias Informáticas
Facultad 2**



**Título: Supervisión de agentes en tiempo real del
Módulo Call-Center de Elastix.**



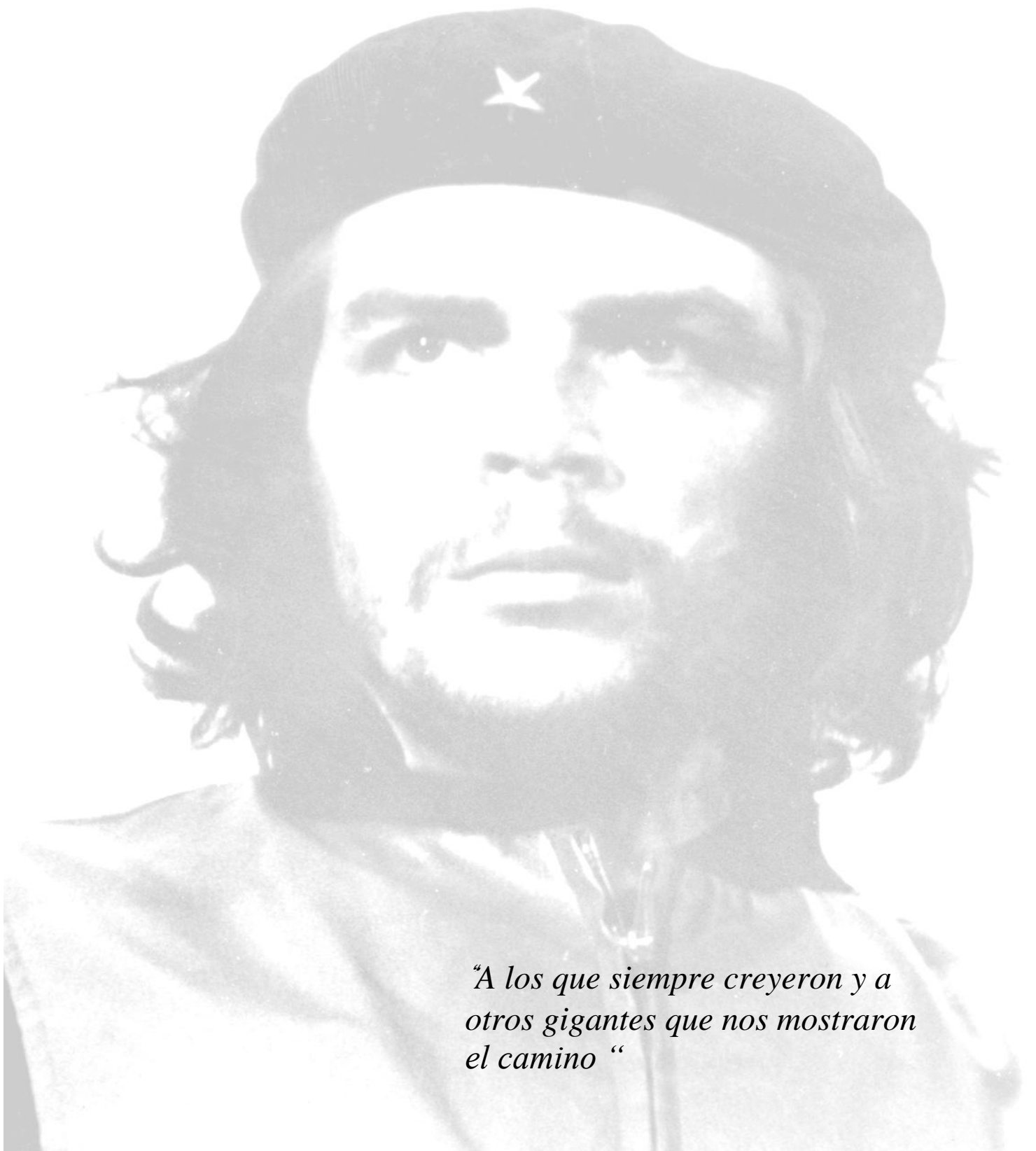
Trabajo de Diploma para optar por el título de
Ingeniero Informático

Autor: Dianelys López Sanchez

Yadier Abel De Quesada Ricardo

Tutor: Ing. Yaislenis Landabe Barbaru.

Ciudad de La Habana, Junio 2010



*“A los que siempre creyeron y a
otros gigantes que nos mostraron
el camino “*

DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis, reconociendo a la Universidad de las Ciencias Informáticas los derechos patrimoniales de manera exclusiva.

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

Dianelys López Sanchez
Firma Autor

Yadier Abel De Quesada Ricardo
Firma Autor

Firma del Tutor

Ing. Yaislenis Landabe Barbaru

Datos de Contacto

ylandabe@uci.cu Ing. Yaislenis Landabe Barbarú, graduada de Ingeniera en Ciencias Informáticas en el 2007. Pertenece al centro de TLM, se desempeña como Subdirectora de formación.

Agradecimientos

Agradezco a mi prima Adislen por ser más que mi prima mi hermana, por entenderme en todos mis momentos buenos y malos.

Agradezco a mi tío Aliesky por ser tan bueno conmigo y por el simple hecho de existir y apoyarme aunque sea en silencio.

Agradezco a mis abuelos maternos por hacer que nuestra pequeña familia siempre esté unida y por darme tanto cariño.

Agradezco a mi tía Katy por ser la tía más linda del mundo, por quererme y por traer al mundo a mis dos primitas Adislenys y Adisleinys que son otro de mis tesoros.

Agradezco a mi papá por la ayuda que me ha brindado y espero que se sienta orgulloso de mí.

Agradezco a mi compañero de tesis que es la persona perfecta para que me acompañara en la realización de mi trabajo de diploma porque cuando estaba alterada tenía un poder para calmarme enorme.

Agradezco a mis tíos Isabelita y Jacinto por ser tan buenos conmigo y darme apoyo y ante todo por ser parte de mi familia.

Agradezco a Diorbis por ser mi amigo incondicional y por quererme tanto dándome la oportunidad de conocer tantas cosas que siempre he valorado.

Agradezco a mi profesora Ayme Hernández Calzada y a la profesora Maura Berta Hidalgo Rosales por aconsejarme tanto y por querer siempre el bien para mí.

Agradezco a mis compañeros de la vocacional Asley, Yurisdan y YUNET.

Agradezco a mis mejores amigos Maipu por ser mi compañera inseparable, Puchito por ser quien me hace reír cuando me siento triste, Toña por estar de acuerdo en todas mis locuras, Danae por ser mi confidente, Malena porque siempre estaba ahí para darme un buen consejo y decirme lo que hago mal y Mayrielis por hacerme bailar cuando no tenía buenos ánimos, a todos ellos gracias por estar siempre presentes en los buenos y malos

momentos que pasé en estos 5 años y en especial este último que no fue muy fácil para mi.

Agradezco a Alexis por haberla conocido, por estimarme y valorarme como persona y amiga.

Agradezco a Dayán por ser mi amigo y confidente en gran parte de mi 5to año.

Agradezco a mis amigos de todos estos 5 años a los que llegaron al final y a los que quedaron en el camino de nuestros estudios.

Agradezco a un profesor y amigo como Ramón Alexander Anglada que me ayudó mucho en mi estancia en la universidad.

Agradezco a todos los profesores de nuestro proyecto y en especial a la profesora Arianna Pérez Carmenate.

Agradezco a mi tutora Yaislenis Landabe y al que en un principio también lo fue Erick Pérez Castillo.

Agradezco a los profesores que me han educado y me hacen ser mejor cada día en lo personal y lo profesional.

Agradezco a la UCA y en especial a la Facultad 2 porque en ella me hice una profesional y conocí a muchas personas especiales.

Dianelys

Agradezco a toda la gente linda que me quieres y me ayuda, Luisito.

Agradezco a alguien a quien quiero como a una abuela, Leonor.

Agradezco a todos los que han estado conmigo estos 5 años de estudio, a los locos del 60

Los Mariachis.

Agradezco a mi novia por quererme tanto y darme apoyo.

Agradezco a mi compañera de tesis.

Agradezco a todos mis profesores y a mis compañeros de la vocacional.

Yadier

Dedicatoria

*... a mi abuela por ser mi apoyo en todos los momentos de mi vida
y por ser la persona más especial que he conocido,*

*... a mi hermana por darme tanto cariño y ser la persona a la que
más quiero en el mundo.*

*...a mi mamá por ser mi mejor amiga, por enseñarme a dar mis
primeros pasos en la vida y espero que siempre se sienta orgullosa
de mi.*

Dianelys

*Dedico mi trabajo de diploma a todas las personas especiales en
mi vida y en especial a las que más quiero:*

*...a mi mamá, que se lo merece por ser la mejor madre del
mundo.*

*...a mi sobrino, por darle alegría a mi vida con su pequeña
sonrisa.*

...a mi hermana, por apoyarme y ayudarme tanto.

*...a mi papá que donde quiera que esté espero esté orgulloso de
mi.*

Yadier

Resumen

Los sistemas de supervisión son considerados de gran importancia dentro de los Call-Center implementados actualmente. Existe gran diversidad de ellos en el mundo, específicamente en Cuba se cuenta con el Elastix el cual es una distribución orientada a las Comunicaciones Unificadas, añade su propio conjunto de utilidades y permite la creación de módulos. Este software posee un módulo de Call-Center el cual asiste centros de llamadas masivas donde interactúan agentes y abonados. Este módulo no cuenta con un subsistema de supervisión para estos agentes el cual, mediante un supervisor, controle todas las actividades realizadas por dichos agentes. El presente trabajo tiene como objetivo el desarrollo de un subsistema de supervisión de agentes en tiempo real, obteniendo como resultado un conjunto de funcionalidades que posibilitan que el sistema Elastix se consolide como un paquete de software disponible para la telefonía de código abierto.

PALABRAS CLAVE

Sistemas, Supervisión, Agentes, Abonados, Supervisor, Elastix, Call-Center

ÍNDICE

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	14
1 Introducción	14
1.1 Conceptos fundamentales.....	14
1.2 Estudio del arte.	14
1.3 Elección de la metodología	16
1.4 Herramienta de Modelado.....	17
1.4.1 BizAgi.....	18
1.5 Gestor de Base de Datos	18
1.6 Lenguajes de desarrollo	19
1.6.1 Lenguaje de programación del lado del servidor	19
1.6.1.1 Hypertext Pre-processor (PHP)	19
1.6.2 Lenguaje de programación del lado del cliente.....	19
1.6.2.1 Javascript	19
1.6.2.2 HTML	19
1.6.2.3 Estilo CSS	20
1.7 Servidor Web.....	20
1.7.1 Servidor Web Apache.....	20
1.8 Dreamweaver.....	21
1.9 Framework Neo.....	21
1.10 Conclusiones.....	22
CAPITULO 2: PLANIFICACIÓN Y CARACTERÍSTICAS DEL SISTEMA.....	23
2.1 Introducción	23
2.2 Propuesta del sistema.....	23
2.3 Modelación de procesos del sistema.....	23
2.4 Características no funcionales del sistema.....	24
2.5 Historias de usuario.....	25
Tabla 2: Muestra de una Historia de usuario	26
Tabla 3: Historia de usuario Nro. 1: Insertar supervisor	27
Tabla 4: Historia de usuario Nro. 2: Modificar supervisor	27
Tabla 5: Historia de usuario Nro. 3: Eliminar supervisor	28
Tabla 6: Historia de usuario Nro. 4: Autenticar supervisor.....	29

Tabla 7: Historia de usuario Nro. 5: Supervisar agentes en tiempo real.	30
Tabla 8: Historia de usuario Nro. 6: Intervenir llamadas.	31
Tabla 9: Historia de usuario Nro. 7: Colgar llamada.	32
Tabla 10: Historia de usuario Nro. 8: Desconectar agente.	33
Tabla 11: Historia de usuario Nro. 9: Transferir llamada.	34
2.6 Planificación.	35
2.6.1 Estimación de esfuerzo por historias de usuarios.	35
2.6.2 Plan de Iteraciones.	36
2.6.2.1 Iteración 1.	36
2.6.2.2 Iteración 2.	36
2.6.2.3 Iteración 3.	36
2.6.3 Plan de duración de las iteraciones.	37
2.6.4 Plan de entregas.	37
2.7 Conclusiones.	38
CAPÍTULO 3: DISEÑO, IMPLEMENTACIÓN Y PRUEBA	39
3.1 Introducción.	39
3.2 Patrón de Arquitectura	39
3.2.1 Modelo Vista Controlador (MVC)	39
3.2.3 Ventajas del uso del MVC	42
3.3 Patrones de Diseño.	42
3.3.1 Patrones para Asignar Responsabilidades (GRASP)	42
3.4 Modelo Físico de la Base de Datos	44
3.4.1 Especificaciones de la Base de Datos empleada	44
3.5 Tarjetas Clase – Responsabilidad – Colaborador.	45
Tabla 16: Tarjeta CRC Clase Agente	45
Tabla 17: Tarjeta CRC Clase Supervisores.	45
Tabla 18: Tarjeta CRC Clase CSupervision	46
3.6 Tareas de Ingeniería	46
3.7 Pruebas.	51
3.7.1 Pruebas unitarias	51
3.7.2 Pruebas de aceptación.	51
CAPÍTULO 4: ESTUDIO DE LA FACTIBILIDAD	53

4.1	Introducción	53
4.2	Modelo matemático COCOMO II.....	53
4.3	Características del Proyecto.....	54
4.3.1	Entradas externas.	54
4.3.2	Salidas Externas	54
4.3.3	Consultas Externas.	55
4.3.4	Archivos Lógicos Internos.	55
4.3.5	Archivos de Interfaz Externos.....	56
4.3.6	Puntos de Función Desajustados.	56
4.4	Cálculos de instrucciones fuentes, esfuerzo, tiempo de desarrollo, cantidad de hombres y costo....	57
4.4.1	Cálculo de instrucciones fuentes.	57
4.4.2	Cálculo de esfuerzo nominal.	58
4.4.2.1	Precedentes (PREC).	59
4.4.2.2	Flexibilidad de desarrollo (FLEX).	59
4.4.2.3	Cohesión del equipo (TEAM).	59
4.4.2.4	Solución de riesgos (RESL).	59
4.4.2.5	Madurez del proceso (PMAT).	59
4.5	Ajuste del esfuerzo nominal.	60
4.5.1	Clasificados en categorías	60
4.5.2	Calculo del tiempo de desarrollo del software.	61
4.5.3	Cálculo del costo total del proyecto.	62
4.6	Resultados.	63
4.7	Análisis de costo.	63
4.8	Conclusiones.....	64
	CONCLUSIONES GENERALES	65
	RECOMENDACIONES.....	66
	REFERENCIA BIBLIOGRÁFICA	67
	BIBLIOGRAFÍA.....	68
	ANEXO I TABLAS DEL COCOMO II	70
	ANEXO II CASOS DE PRUEBA.....	72
	GLOSARIO DE TÉRMINOS	80

INTRODUCCIÓN

Con el creciente desarrollo tecnológico y los avances en los medios de comunicación donde el acceso a la información es una realidad y la necesidad de mantener el contacto en tiempo real con los clientes y los colaboradores es vital; surge el concepto de las Comunicaciones Unificadas que no son más que soluciones que reducen los inconvenientes y los retrasos que se producen al intentar localizar o contactar con personas.

Las Comunicaciones Unificadas llevan la convergencia más allá de la telefonía IP, integrando presencia, mensajería instantánea, mensajería unificada, conferencia Web y videoconferencia, en una plataforma común accesible desde cualquier medio. Sus ventajas, por tanto, son evidentes no sólo en cuanto al incremento de la productividad y la agilización de la toma de decisiones, también avanzan ya hacia una más profunda y sencilla integración con las aplicaciones y procesos de negocio.

El sistema Elastix es un ejemplo de distribución libre de Servidor de Comunicaciones Unificadas, este implementa gran parte de su funcionalidad sobre cuatro programas de software muy importantes como son Asterisk, HylaFAX, Openfire y Postfix. Estos brindan las funciones de Private Branch Exchange (PBX), Fax, Mensajería Instantánea y Correo electrónico respectivamente.

Elastix es un software aplicativo que integra las mejores herramientas disponibles para PBX basados en Asterisk en una interfaz simple y fácil de usar. Además añade su propio conjunto de utilidades y permite la creación de módulos de terceros.

El módulo de Call-Center específicamente de Elastix, es un paquete de software diseñado para asistir en la implementación de centros de llamadas masivas con interacción entre operadores contratados por el administrador de la central telefónica llamados "agentes" y personas contactadas a través de llamadas telefónicas, denominadas "abonados". Este módulo cuenta con muchas funcionalidades básicas ya implementadas, pero carece de algunos conceptos más complejos como por ejemplo la supervisión en tiempo real de los agentes.

La Universidad de las Ciencias Informáticas (UCI) está en camino a convertirse en una potente empresa productora de software. Como consecuencia de esto la UCI desde sus inicios posee vínculos con algunos países a los cuales le proporciona software solicitados por los mismos, un ejemplo de esto es el convenio

entre Cuba y Venezuela lo que abrió el camino a la UCI como proveedora de Soluciones Tecnológicas Integrales.

Actualmente la UCI está desplegando el sistema Elastix en la república Bolivariana de Venezuela sin contar con el subsistema de supervisión que formaría una parte importante en el control de los agentes. Por lo que sería muy útil agregarle el concepto de supervisión al módulo Call-Center para un mejor desempeño, aumentándole las funcionalidades que este subsistema puede brindar, adaptándolo mejor a las necesidades que requieren los sistemas que se están desplegando.

El módulo Call-Center no posee en su versión actual un sistema de supervisión en tiempo real, el cuál incluya los agentes que están en línea, las llamadas que están siendo atendidas por cada agente, así como la intervención, el redireccionamiento e interrupción de llamadas, por lo que se plantea el siguiente

Problema Científico: ¿Cómo supervisar los agentes en tiempo real del módulo Call-Center de Elastix?

El objeto de estudio del presente trabajo son los subsistemas de supervisión de agentes en tiempo real orientado al **Campo de acción:** Proceso de supervisión en tiempo real para el módulo de Call-Center en el Elastix.

El problema anterior arrojó como **Objetivo general:** Adicionar al módulo Call-Center de Elastix un subsistema de supervisión de agentes en tiempo real para lograr monitorear el trabajo de los mismos dentro del sistema.

Las tareas investigativas a desarrollar para cumplir estos objetivos son:

1. Análisis del sistema Asterisk para comprender la integración que posee con el sistema Elastix.
2. Análisis del framework Neos para comprender su funcionamiento y la forma en la que está estructurado.
3. Análisis del subsistema Call-Center del Elastix para comprender las deficiencias que posee el mismo.
4. Selección y análisis de las interfaces para interactuar con el Asterisk.
5. Estudio de sistemas de Call-Center para profundizar en sus principales características las cuales los hacen más completos y eficientes.
6. Selección de la metodología a utilizar.

Idea a defender:

El subsistema de supervisión de agentes en tiempo real del módulo Call-Center integrado al Elastix permite monitorear y mostrar la información de dichos agentes para así tener un mayor control sobre las llamadas y el personal involucrado en dicho subsistema.

Métodos teóricos:

- **Análisis y Síntesis:** para el procesamiento de la información del negocio de la organización y arribar a las conclusiones de la investigación, así como para precisar las características del trabajo a realizar.
- **Histórico - Lógico:** Para determinar las tendencias actuales de los sistemas de integración, de los modelos de desarrollo y las técnicas, lenguajes y herramientas utilizadas durante la investigación.

Métodos empíricos:

- **Observación:** Para la percepción selectiva de las restricciones y propiedades del sistema y para la determinación de la problemática que da origen a la investigación. La observación que se realiza es estructurada ya que previamente se establecen los aspectos que se desean observar sistematizando los detalles más significativos para la investigación

El presente trabajo de diploma está dividido en 4 capítulos:

Capítulo 1 Fundamentación Teórica: Contiene una descripción del proceso que será automatizado y un breve estudio de los principales conceptos tratados en el trabajo. Se exponen las principales características de las tecnologías y herramientas seleccionadas para la solución del problema actual.

Capítulo 2 Planificación y Características del Sistema: En este capítulo se hace una propuesta del sistema que consta de una modelación de los procesos, se exponen las características no funcionales, se plantean de forma general las Historias de usuario(HU) y por último se lleva a cabo una estimación de planificación basada en dichas HU.

Capítulo 3. Diseño, Implementación y Prueba: Contiene la arquitectura del sistema y con ello los patrones arquitectónicos empleados. Se procederá a abordar las HU definidas y Tareas de la ingeniería en cada iteración para luego hacer una revisión de las mismas y llevar a cabo las pruebas de la aplicación.

Capítulo 4. Estudio de la factibilidad: Describe paso a paso y de forma profunda el proceso de estudio de la factibilidad.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1 Introducción

A lo largo del desarrollo de este capítulo se profundizan algunos conceptos fundamentales para el desarrollo de un módulo que permita la supervisión de agentes en tiempo real del módulo de Call-Center de Elastix. También se exponen las principales características de las tecnologías, herramientas, lenguajes y técnicas de desarrollo propuestas para la solución del problema actual, así como la justificación de la selección de las mismas.

1.1 Conceptos fundamentales

Call-Center: es un centro de servicio telefónico que tiene la capacidad de atender altos volúmenes de llamadas, con diferentes objetivos. Su principal enfoque es el de la generación de llamadas de salida y la recepción de llamadas, cubriendo las expectativas de cada una de las campañas implementadas. (1)

Campañas implementadas: son promociones dirigidas al ámbito tecnológico para facilitar el trabajo humano, dando propaganda a diferentes servicios novedosos brindados por el Call-Center.

Subsistema: es un sistema que se ejecuta sobre un sistema operativo, este puede ser un shell (intérprete de comandos) del sistema operativo primario o puede ser una máquina virtual.

Supervisión: es la observación regular y el registro de las actividades que se llevan a cabo en un proyecto o programa. Es un proceso de recogida rutinaria de información sobre todos los aspectos del proyecto. Supervisar es controlar qué tal progresan las actividades del proyecto.

1.2 Estudio del arte.

En la actualidad existe gran diversidad de sistemas Call-Center como son el VICIDIAL, el TELEPROM AGENTS MONITOR y X-Series Call-Center donde cada uno posee características diferentes las cuales se exponen a continuación:

El VICIDIAL Call-Center Suite es un conjunto de programas que están diseñados para interactuar con el sistema Asterisk Open Source PBX de teléfono para actuar como un complemento de entrada y salida privado del centro de llamadas. La interfaz de agente es un conjunto interactivo de páginas web que

funcionan a través de un navegador web para dar información en tiempo real. La interfaz de administración es también basada en la web y ofrece la posibilidad de ver muchos informes en tiempo real e informes resumidos, detalles de las campañas así como las opciones de los agentes y su configuración. En este momento hay más de 1000 instalaciones de VICIDIAL en producción en más de 60 países de todo el mundo, algunos con más de 300 plazas de agente y varios con múltiples ubicaciones. (2)

TELEPROM AGENTS MONITOR es un programa para control, monitoreo y administración de tareas de operadores de Call-Center cuyos objetivos son los siguientes:

- Controlar y monitorear en tiempo real las acciones de los agentes.
- Administrar en forma centralizada los equipos Teleprom instalados en sus puestos de trabajo.
- Generar reportes de sus tareas telefónicas.

Se puede observar gráficamente la ubicación física del puesto de trabajo de cada agente, y su estado telefónico; además de supervisar los tiempos de dichos estados, ya sea de una comunicación telefónica en curso, el tiempo transcurrido sin una comunicación, los tiempos auxiliares (break, toilette, capacitación, coaching) o las situaciones anormales (bloqueo por incumplimiento de tiempos límite, temporalmente inhabilitado). (3)

X-Series permite tener todas las herramientas que un Call-Center requiere tales como: manejo de agentes, manejo de colas de servicio, monitoreo de llamadas y un sistema de estadísticas que le permitirá saber y medir el comportamiento de su Call-Center y sus agentes. (4)

Posee como funciones de monitoreo:

- Monitoreo Silencioso
- Monitoreo Privado
- Monitoreo en Conferencia
- Visualización de Operación en tiempo real

Como parte de estas soluciones implementadas en el mundo, el Software Elastix posee un subsistema Call-Center el cual no posee todas las funcionalidades de los anteriormente explicados por lo que el siguiente trabajo pretende agregar el concepto de supervisión de agentes, ya que en estos momentos se puede ver al agente conectado pero no se puede supervisar el trabajo del mismo.

1.3 Elección de la metodología

El desarrollo de software no es una tarea fácil. Prueba de ello es que existen numerosas propuestas de metodologías que inciden en distintas dimensiones del proceso de desarrollo. Por una parte se tienen aquellas propuestas más tradicionales que se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, las herramientas y notaciones que se usarán. Otra aproximación es centrarse en otras dimensiones, como por ejemplo el factor humano o el producto software. Esta es la filosofía de las metodologías ágiles, las cuales dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas.

Las dos metodologías más conocidas por el equipo del software son Rational Unified Process(RUP) como metodología tradicional y Extreme Programming (XP) la cual representa a las metodologías ágiles. A continuación se realiza una comparación entre ellas y se exponen las principales características tanto del equipo de desarrollo como del propio software a implementar.

Tabla 1: Comparación entre RUP vs XP

RUP	XP
Forma disciplinada de asignar tareas y responsabilidades en una empresa de desarrollo.	Nace en busca de simplificar el desarrollo del software y que se lograra reducir el costo del proyecto.
Un cambio en las etapas de vida del sistema incrementaría notablemente el costo.	Reduce el costo del cambio en las etapas de vida del sistema.
Requiere un grupo grande de programadores para trabajar con esta metodología.	Se requiere un grupo pequeño de programadores para trabajar con esta metodología entre 2 – 15 personas y estas irán aumentando conforme sea necesario.
Define un manajo entero de las actividades y de los artefactos que se necesitan elegir para construir su propio proceso individual.	Combina las que han demostrado ser las mejores prácticas de desarrollo de software, y las lleva al extremo.
El cliente no es parte del equipo.	El cliente es parte del equipo, determina qué construir y cuándo, establece las pruebas funcionales.
Para macro proyectos.	Para proyectos de corto plazo.
Emplea un gran equipo para su desarrollo.	Para proyectos de pequeño equipo.

Se pueden argumentar otras características de XP la cual es una de las metodologías de desarrollo de software más exitosas en la actualidad. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. El manejo del cambio se convierte en parte sustantiva del proceso, el costo del cambio no depende de la fase o etapa y no introduce funcionalidades antes que sean necesarias.

1.3.1 Metodología seleccionada

El equipo de desarrollo está compuesto por dos personas, la experiencia productiva es media, el cliente se encuentra en la universidad específicamente dentro del proyecto para el cual se está conformando la aplicación y forma parte del equipo de desarrollo, la disponibilidad de tiempo es limitada.

Por todo lo antes planteado se llega a la conclusión que la metodología más conveniente es XP ya que RUP no es recomendable para este trabajo de diploma basándose en las características del equipo de desarrollo y del software a implementar.

1.4 Herramienta de Modelado

Las Herramientas Case (Ingeniería de Software Asistida por Computadoras) son la mejor base para el proceso de análisis y desarrollo de software. Se puede definir a las Herramientas Case como un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un software. (5)

Existen varias Herramientas Case como son:

- CASE GENEXUS.
- System Architect: herramientas CASE para Análisis y Diseño, incluye técnicas estructuradas y orientadas a objetos.
- Win A&D: herramientas CASE para Análisis y Diseño, incluye técnicas estructuradas y orientadas a objetos.
- CRADLE: conjunto de herramientas CASE integradas que dan soporte a la Planificación estratégica, Análisis y Diseño.
- PowerDesigner 7.0: herramienta CASE de Análisis y Diseño incluye capacidades de generación relacional y con orientación a objetos.

- SilverRun: Conjunto integrado de de herramientas CASE para el modelado de negocios.(4)

1.4.1 BizAgi

Se decide utilizar BizAgi ya que permite realizar diagramas y documentar tus procesos de la manera rápida y fácil. Posibilita diseñar, integrar, automatizar y monitorear los procesos de negocio por medio de un ambiente gráfico. Posee una interfaz sencilla y fácil de aprender. Permite alinear y organizar los recursos de manera que compartan y contribuyan los conocimiento para modelar los procesos, maximizan la agilidad y la eficiencia que se tiene para diagramar los procesos la organización, sirve como soporte al 100% sobre el único estándar para diagramar procesos y se podrá documentar los procesos de acuerdo al diagrama del proceso y está considerado como una manera más rápida, sencilla y sin costo de implementación.

1.5 Gestor de Base de Datos

Los Sistema Gestores de Bases de Datos (SGBD) son software que proporcionan servicios para la creación, el almacenamiento, el procesamiento y la consulta de la información almacenada en base de datos de forma segura y eficiente. Un SGBD actúa como un intermediario entre las aplicaciones y los datos. (6)

Existen varios SGBD como por ejemplo PostgreSQL, SQLite, Apache Derby entre otros, pero para este trabajo de diploma se utiliza como gestor de base de datos, MySQL, debido a que este es utilizado en la plataforma Elastix.

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario. Es de software libre, muy utilizado en aplicaciones web, sobre plataformas Linux/Windows-Apache-MySQL-PHP/Perl/Python. Su prestigio como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL, también es muy destacable, la condición de Open Source de MySQL, que hace que su utilización sea gratuita e incluso se pueda modificar con total libertad, pudiendo descargar su código fuente. Esto ha favorecido muy positivamente en su desarrollo y continuas actualizaciones, para hacer de MySQL una de las herramientas más utilizadas por los programadores orientados a Internet.

1.6 Lenguajes de desarrollo

1.6.1 Lenguaje de programación del lado del servidor

1.6.1.1 Hypertext Pre-processor (PHP)

PHP (acrónimo de Hypertext Pre-processor), es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica. Se utilizan para complementar el HTML (lenguaje para crear páginas Web) para hacer una Web más interactiva y funcional. Gracias a estos lenguajes se pueden desarrollar cualquier tipo de servicios a través de su sitio Web. Además, PHP permite configurar el servidor de modo que se permita o rechacen diferentes usos, lo que puede hacer al lenguaje más o menos seguro dependiendo de las necesidades de cada cual.

Con PHP se puede hacer cualquier cosa que se pueden realizar con un script CGI, como el procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas. Un sitio con páginas dinámicas es el que permite interactuar con el visitante, de modo que cada usuario que visita la pagina vea la información modificada para requisitos articulares.

1.6.2 Lenguaje de programación del lado del cliente

1.6.2.1 Javascript

Javascript es un lenguaje de programación interpretado. Se utiliza principalmente en su forma del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

1.6.2.2 HTML

Se propone utilizar como lenguaje de programación en el lado del cliente HTML siglas de Hyper Text Markup Language (Lenguaje de Marcado de Hipertexto) ya que es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes.

HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un *script* (por ejemplo Javascript). Es el lenguaje de marcado predominante para la elaboración de las páginas web.

1.6.2.3 Estilo CSS

Las hojas de estilo en cascada (en inglés Cascading Style Sheets), CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML).

Estas hojas de estilo permiten en este trabajo seguir las pautas del diseño de la plataforma web de Elastix.

1.7 Servidor Web

Un servidor web es un programa que sirve para atender y responder a las diferentes peticiones de los navegadores, proporcionando los recursos que soliciten usando el protocolo HTTP o el protocolo HTTPS (la versión cifrada y autenticada). Un servidor web es un programa que se ejecuta continuamente en un computador, manteniéndose a la espera de peticiones de ejecución que le hará un cliente o un usuario de Internet. El servidor web se encarga de contestar a estas peticiones de forma adecuada, entregando como resultado una página web o información de todo tipo de acuerdo a los comandos solicitados. (7)

1.7.1 Servidor Web Apache

Apache es un servidor web flexible, rápido y eficiente, continuamente actualizado y adaptado a los nuevos protocolos (HTTP). Entre sus características destacan:

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Apache es una tecnología gratuita de código fuente abierta.
- Es un servidor altamente configurable de diseño modular.
- Trabaja con gran cantidad de lenguajes como Perl, PHP y otros de script.
- Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor.
- Tiene una alta configurabilidad en la creación y gestión de logos.

1.8 Dreamweaver

Dreamweaver es la herramienta de diseño de páginas web más avanzada. Aunque sea un experto programador de HTML el usuario que lo maneje, siempre se encontrarán en este programa razones para utilizarlo, sobre todo en lo que a productividad se refiere.

Cumple perfectamente el objetivo de diseñar páginas con aspecto profesional, y soporta gran cantidad de tecnologías, además muy fáciles de usar:

- Hojas de estilo y capas
- Javascript para crear efectos e interactividades
- Inserción de archivos multimedia

Es un programa que se puede actualizar con componentes, que fabrica tanto Macromedia como otras compañías, para realizar otras acciones más avanzadas.

En resumen, el programa es realmente satisfactorio, incluso el código generado es de buena calidad. La única limitante consiste en que al ser tan avanzado, puede resultar un poco difícil su manejo para personas menos experimentadas en el diseño de webs. (8)

1.9 Framework Neo

El uso de los framework actualmente se ha convertido en un punto importante para la ingeniería de software ya que están compuestos por componentes que se pueden personalizar o intercambiar cuando se desarrolla una aplicación. El uso de un framework persigue tres objetivos importantes, acelerar el proceso del desarrollo de una aplicación, reutilización de código ya existente y una correcta aplicación de los patrones de la ingeniería.

La naturaleza de la arquitectura de este framework es MVC2 (Modelo Vista Controlador), el número se refiere a la versión del mismo, y su desarrollo en la Programación Orientada a Objetos (POO). Posee algunos tipos de datos especiales, validaciones automáticas, recursos de autorización y autenticación, informes y otras herramientas utilitarias que facilitan y optimizan las operaciones.

1.10 Conclusiones

En este capítulo se realizó un análisis de los principales sistemas Call-Center existentes en el mundo, así como los conceptos de más relevancia para llegar a un mejor entendimiento del sistema a implementar, también se llevó a cabo una investigación de las principales metodologías de desarrollo de software, con el fin de justificar la empleada teniendo en cuenta las características del sistema a desarrollar. Se realizó un estudio de las principales herramientas y tecnologías a utilizar para la realización de este trabajo teniendo en cuenta criterios que hagan óptimo al sistema.

CAPITULO 2: PLANIFICACIÓN Y CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción

En el presente capítulo se elabora una propuesta del sistema a implementar, se exponen las características del sistema para un mejor entendimiento, así como la descripción y modelación de los procesos, detallando las historias de usuarios que proporcionan una mejor visión sobre lo que el cliente desea.

2.2 Propuesta del sistema

El software propone realizar dos subsistemas dentro del módulo Call-Center de Elastix donde el primero tiene como objetivo principal gestionar a los supervisores; ya sea insertarlos modificarlos o eliminarlos, y el segundo consiste en la supervisión de los agentes asignados a cada supervisor. Ambos subsistemas tendrían como objetivo general la supervisión en tiempo real de los agentes conectados al módulo Call-Center de Elastix.

2.3 Modelación de procesos del sistema

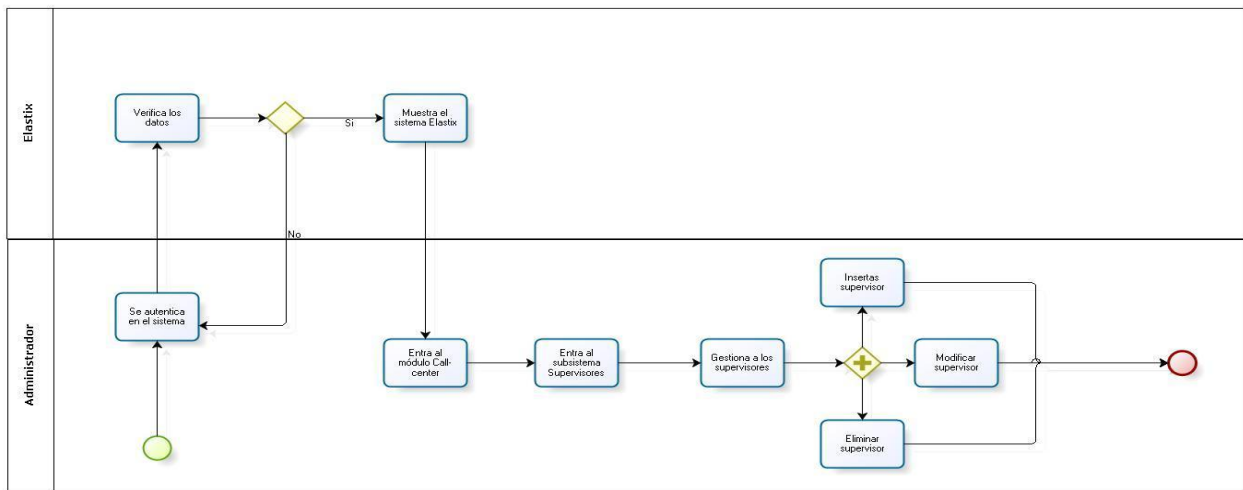


Figura 1 Proceso del sistema Gestión de supervisores

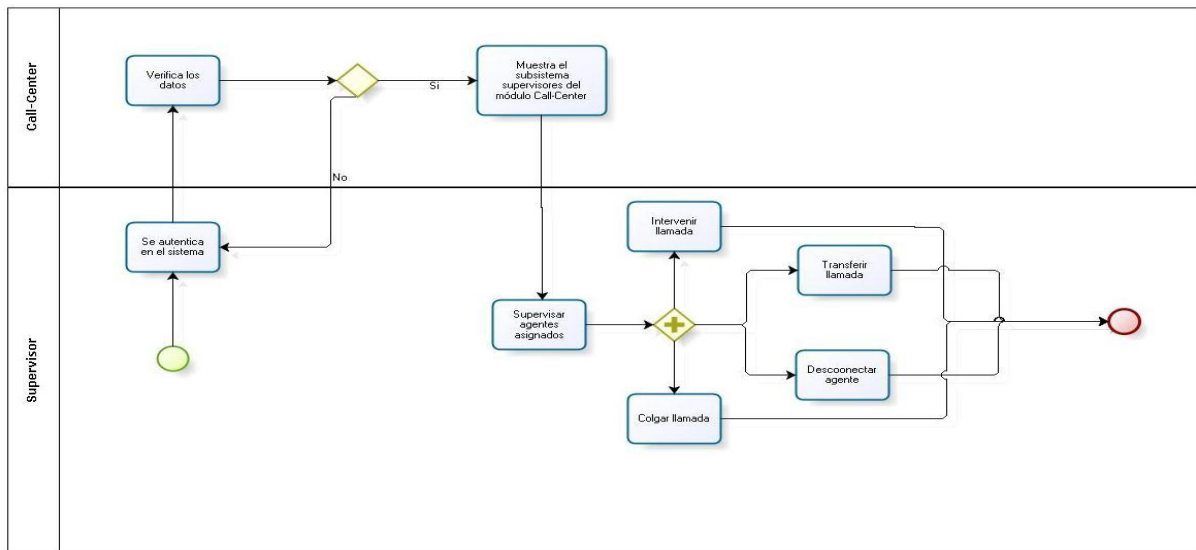


Figura 2 Proceso del sistema Supervisión de agentes

2.4 Características no funcionales del sistema

Seguridad

- **Confiabilidad:** La información manejada por el sistema debe estar protegida de acceso no autorizado. Se evidencia mediante la autenticación.
- **Integridad:** La información manejada por el sistema debe ser objeto de cuidadosa protección contra la corrupción y estados de inconsistencia. Se evidencia mediante las auditorías de seguridad.
- **Disponibilidad:** La aplicación deberá estar disponible en todo momento para aquellas personas con acceso a la información, se asegura mediante la instalación de grupos electrógenos.

Fiabilidad

- Prever contingencias para eventos de caída del sistema.

Capacidad

- Considerar características técnicas mínimas para la ejecución en clientes.
- Soportar múltiples conexiones al sistema de manera simultánea.

Funcionalidad

- Mínima cantidad de páginas para ejecutar todas las funciones posibles, es decir, agrupar funciones afines en las mismas páginas.
- Considerar el crecimiento esperado en el volumen de datos.

Usabilidad

- Se necesitará una preparación previa para operar con el sistema. Se requiere un nivel medio o alto de conocimientos de computación, aunque el manejo de la aplicación es sencillo, permitiendo la fácil comprensión por el usuario.

Hardware

- Para la instalación de la aplicación se debe disponer de una computadora de 512 MB de RAM o superior, 160 GB de disco duro o superior y la instalación del php 5.1.

2.5 Historias de usuario

Las historias de usuario son escritas por el cliente, el cual en su propio lenguaje, describe lo que el sistema debe realizar. Las mismas deben tener el detalle mínimo como para que los programadores puedan realizar una estimación poco riesgosa del tiempo que llevará su desarrollo. Cuando llega el momento de la implementación, los desarrolladores dialogarán directamente con el cliente para obtener todos los detalles necesarios.

La prioridad en el negocio:

Alta: Se le otorga a las HU que resultan funcionalidades fundamentales en el desarrollo del sistema, a las que el cliente define como principales para el control integral del sistema.

Media: Se le otorga a las HU que resultan para el cliente como funcionalidades a tener en cuenta, sin que estas tengan una afectación sobre el sistema que se esté desarrollando.

Baja: Se le otorga a las HU que constituyen funcionalidades que sirven de ayuda al control de elementos asociados al equipo de desarrollo, a la estructura y no tienen nada que ver con el sistema en desarrollo.

El riesgo en su desarrollo:

Alta: Cuando en la implementación de las HU se consideran la posible existencia de errores que lleven a la inoperatividad del código.

Media: Cuando pueden aparecer errores en la implementación de la HU que puedan retrasar la entrega de la versión.

Baja: Cuando pueden aparecer errores que serán tratados con relativa facilidad sin que traigan perjuicios para el desarrollo del proyecto. El cliente y el equipo de desarrollo trabajan en conjunto para definir como agrupar las HU para su lanzamiento.

A continuación se muestran las Historias de usuarios definidas:

Tabla 2: Muestra de una Historia de usuario

Historia de Usuario	
Número: (Número de la historia de usuario incremental en el tiempo)	Nombre de Historia de Usuario: (El nombre de la historia de usuario sería para identificarlas mejor entre los desarrolladores y el cliente)
Modificación de Historia de Usuario Número: (si sufrió alguna modificación anterior)	
Usuario: (Involucrados en el desarrollo de la HU)	Iteración Asignada: (Número de la iteración)
Prioridad en negocio: (Alta / Media / Baja)	Puntos estimados: (El tiempo estimado que se demorará el desarrollo de la HU)
Riesgo en Desarrollo: (Alta / Media / Baja)	Puntos Reales: (El tiempo que se demoró en realidad el desarrollo de la HU)
Descripción: (Breve descripción de la HU)	
Observaciones: Señalamiento o advertencia del sistema.	
Prototipo de interface:	

Tabla 3: Historia de usuario Nro. 1: Insertar supervisor


Historia de Usuario	
Número: 1	Nombre de Historia de Usuario: Insertar supervisor.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Dianelys López Sanchez Yadier Abel De Quesada	Iteración Asignada: 1
Prioridad en negocio: Medio	Puntos estimados: 1
Riesgo en Desarrollo: Bajo	Puntos Reales: 1
Descripción: El administrador del sistema introduce los datos del supervisor: usuario, contraseña y listado de agentes registrados, donde el administrador decidirá que agentes supervisará el mismo.	
Observaciones: El administrador del sistema debe estar previamente autenticado.	
Prototipo de interface:	
	

Tabla 4: Historia de usuario Nro. 2: Modificar supervisor

Historia de Usuario	
Número: 2	Nombre de Historia de Usuario: Modificar supervisor.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Dianelys López Sanchez	Iteración Asignada: 1

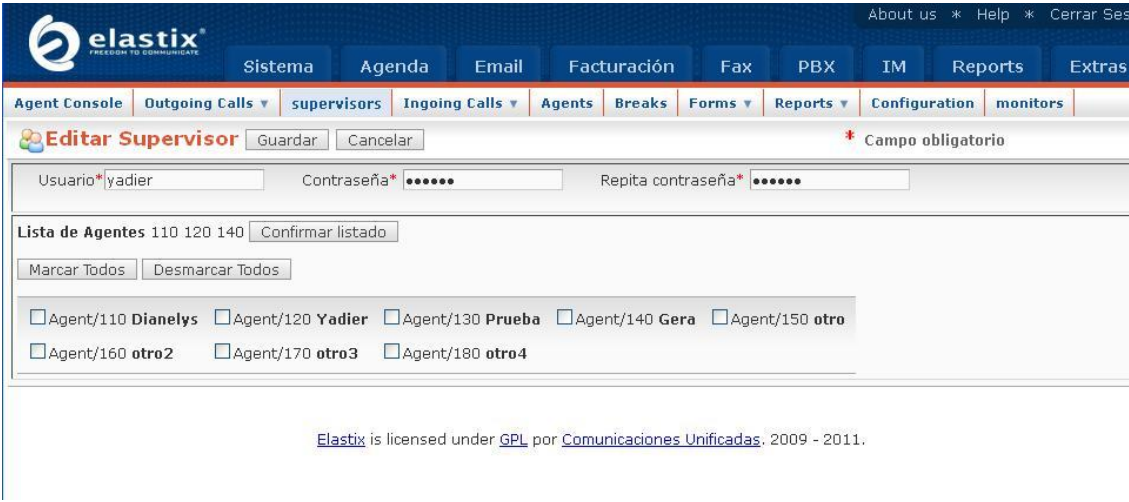
Yadier Abel De Quesada	
Prioridad en negocio: Medio	Puntos estimados: 1
Riesgo en Desarrollo: Bajo	Puntos Reales: 1
Descripción: El administrador, una vez registrado el supervisor, tiene la posibilidad de modificar la contraseña y los agentes que le fueron asignados previamente.	
Observaciones: El administrador del sistema debe estar previamente autenticado.	
Prototipo de interface:	
	

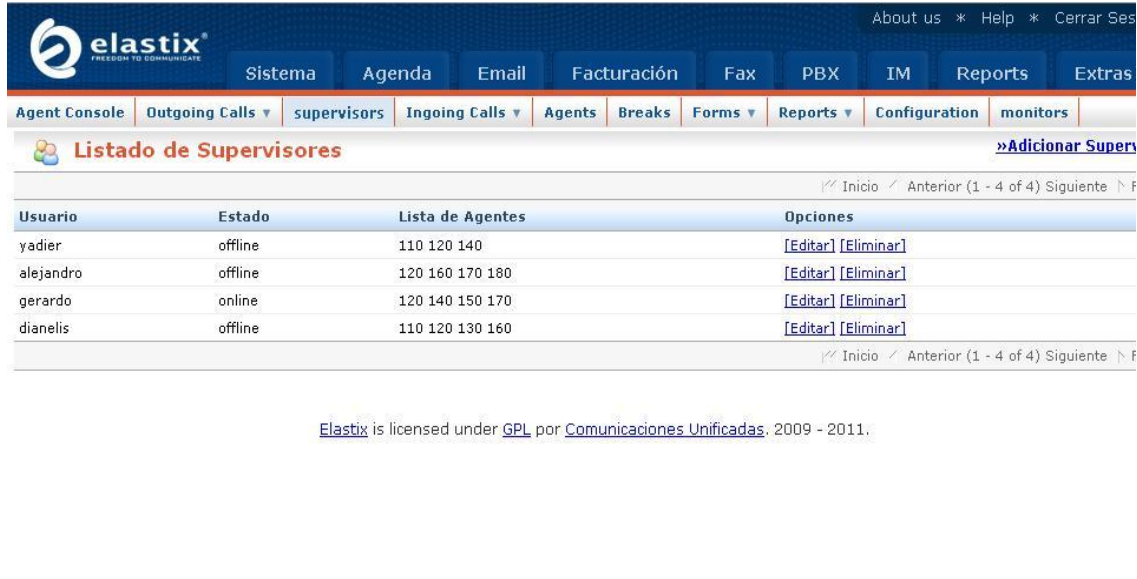
Tabla 5: Historia de usuario Nro. 3: Eliminar supervisor

Historia de Usuario	
Número: 3	Nombre de Historia de Usuario: Eliminar supervisor.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Dianelys López Sanchez Yadier Abel De Quesada	Iteración Asignada: 1
Prioridad en negocio: Medio	Puntos estimados: 1
Riesgo en Desarrollo: Bajo	Puntos Reales: 1
Descripción: El administrador tiene el permiso de eliminar a un supervisor ya	

registrado.

Observaciones: El administrador del sistema debe estar previamente autenticado.

Prototipo de interface:



The screenshot shows the 'Listado de Supervisores' page in the Elastix system. The page has a dark blue header with the Elastix logo and navigation tabs: Sistema, Agenda, Email, Facturación, Fax, PBX, IM, Reports, and Extras. Below the header is a sub-menu with options like Agent Console, Outgoing Calls, supervisors, Ingoing Calls, Agents, Breaks, Forms, Reports, Configuration, and monitors. The main content area displays a table of supervisors with columns for 'Usuario', 'Estado', 'Lista de Agentes', and 'Opciones'. The table lists four supervisors: yadier (offline), alejandro (offline), gerardo (online), and danielis (offline). Each supervisor has a list of agent numbers and 'Editar' and 'Eliminar' buttons. The page also includes pagination controls and a footer note: 'Elastix is licensed under GPL por Comunicaciones Unificadas. 2009 - 2011.'

Tabla 6: Historia de usuario Nro. 4: Autenticar supervisor.

Historia de Usuario	
Número: 4	Nombre de Historia de Usuario: Autenticar supervisor.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Dianelys López Sanchez Yadier Abel De Quesada	Iteración Asignada: 1
Prioridad en negocio: Alta	Puntos estimados: 1
Riesgo en Desarrollo: Medio	Puntos Reales:
Descripción: Consiste en la entrada de datos como usuario y contraseña del supervisor.	
Observaciones: El supervisor debe estar previamente registrado en el sistema.	
Prototipo de interface:	

Bienvenido a la consola de Supervisión

Supervisor:*

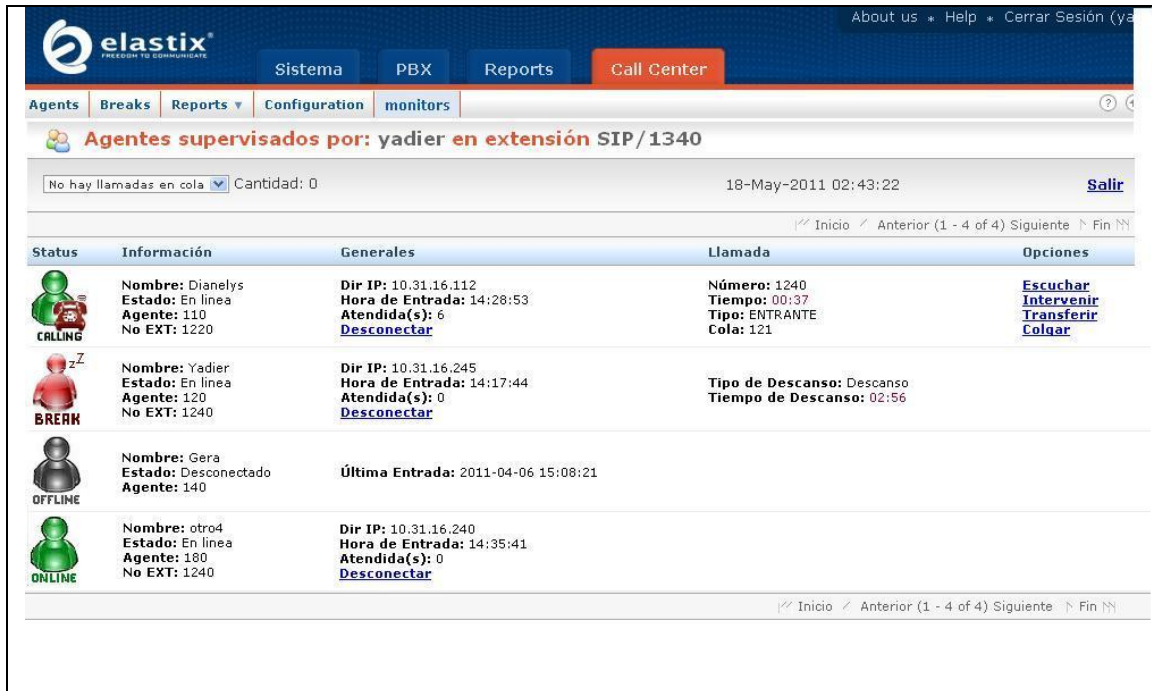
Contraseña:*

Extensión:

* Campo obligatorio

Tabla 7: Historia de usuario Nro. 5: Supervisar agentes en tiempo real.

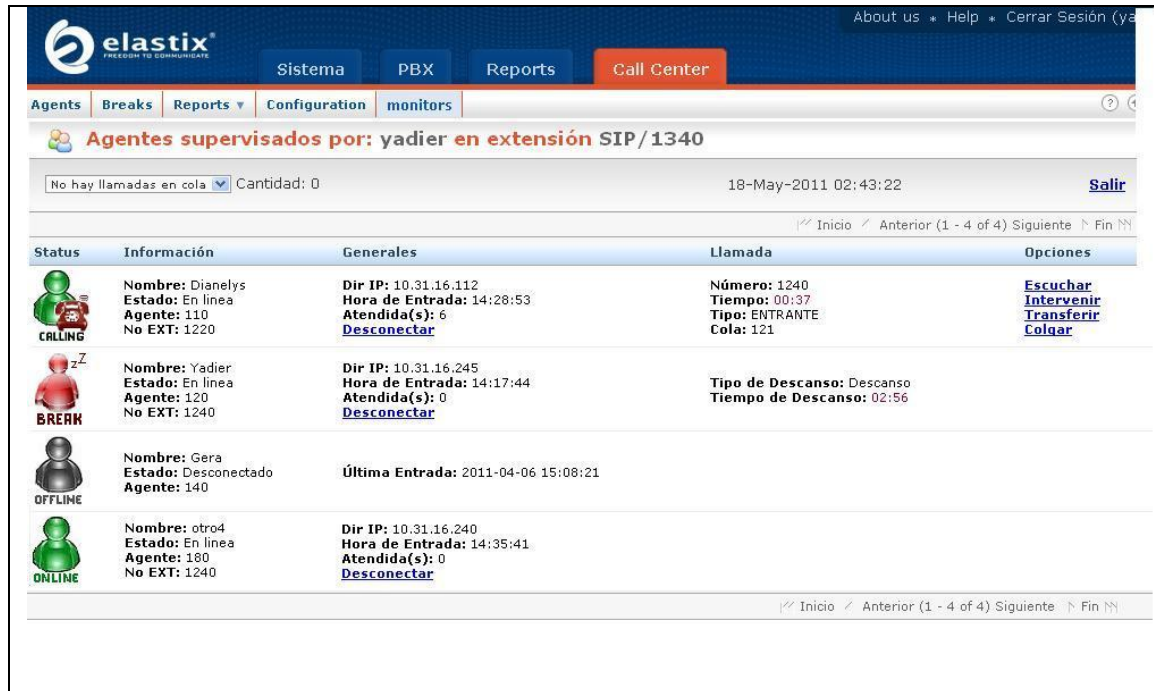
Historia de Usuario	
Número: 5	Nombre de Historia de Usuario: Supervisar agentes en tiempo real.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Dianelys López Sanchez Yadier Abel De Quesada	Iteración Asignada: 1
Prioridad en negocio: Alta	Puntos estimados: 1
Riesgo en Desarrollo: Alta	Puntos Reales: 1
Descripción: Consiste en mostrar todos los agentes registrados en el sistema que le fueron asignados previamente por el administrador.	
Observaciones: El supervisor debe estar autenticado en el sistema.	
Prototipo de interface:	



The screenshot shows the Elastix Call Center interface. At the top, there are navigation tabs: Sistema, PBX, Reports, and Call Center. Below these, there are sub-tabs: Agents, Breaks, Reports, Configuration, and monitors. The main heading is "Agentes supervisados por: yadier en extensión SIP/1340". Below this, there is a status bar showing "No hay llamadas en cola" and "Cantidad: 0". The date and time are "18-May-2011 02:43:22". There is a "Salir" button. Below the status bar, there is a table with columns: Status, Información, Generales, Llamada, and Opciones. The table lists four agents: Dianelys (CALLING), Yadier (BREAK), Gera (OFFLINE), and otro4 (ONLINE). Each agent's row contains their name, state, IP address, entry time, and other call-related details.

Tabla 8: Historia de usuario Nro. 6: Intervenir llamadas.

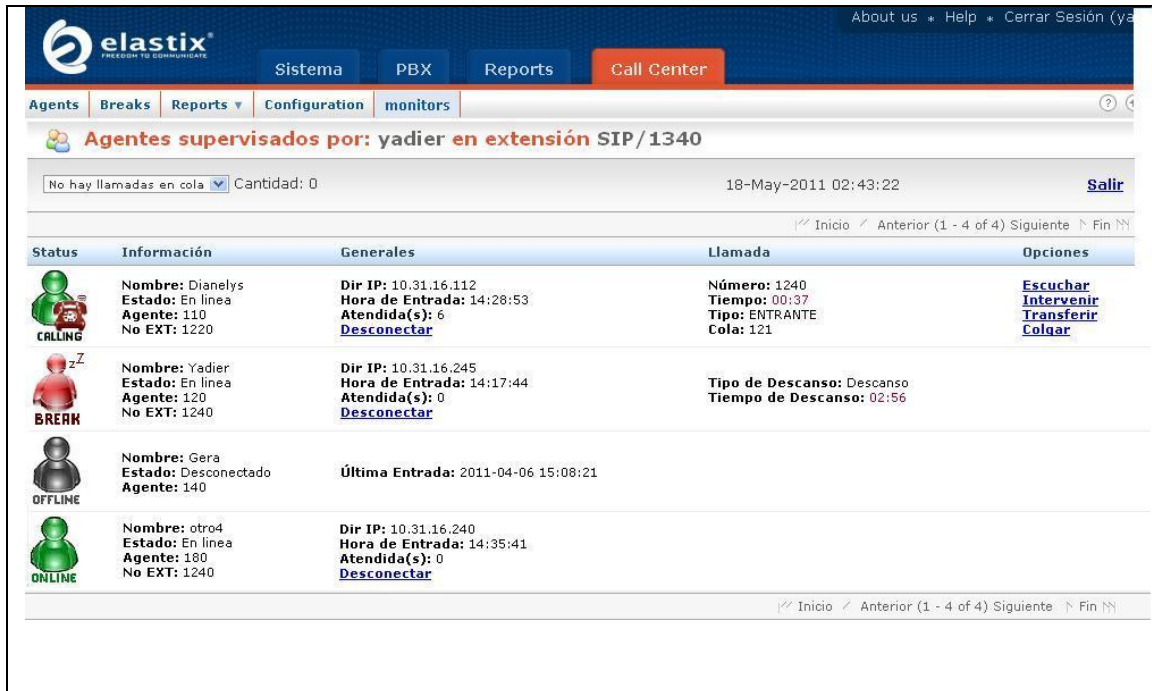
Historia de Usuario	
Número: 6	Nombre de Historia de Usuario: Intervenir llamadas.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Dianelys López Sanchez Yadier Abel De Quesada	Iteración Asignada: 1
Prioridad en negocio: Alta	Puntos estimados: 3
Riesgo en Desarrollo: Alta	Puntos Reales: 3
Descripción: Mediante la intervención de las llamadas el supervisor puede escuchar e intervenir el canal.	
Observaciones: El supervisor debe estar autenticado en el sistema y tener a dicho agente dentro de los que le fue asignado previamente por el administrador.	
Prototipo de interface:	



The screenshot shows the Elastix Call Center interface. At the top, there are navigation tabs for 'Sistema', 'PBX', 'Reports', and 'Call Center'. Below this, there are sub-tabs for 'Agents', 'Breaks', 'Reports', 'Configuration', and 'monitors'. The main heading is 'Agentes supervisados por: yadier en extensión SIP/1340'. Below this, there is a status bar showing 'No hay llamadas en cola' and 'Cantidad: 0'. The main content area displays a list of agents with columns for 'Status', 'Información', 'Generales', 'Llamada', and 'Opciones'. The agents listed are: Dianelys (CALLING), Yadier (BREAK), Gera (OFFLINE), and otro4 (ONLINE). Each agent entry includes details such as name, state, IP address, and call statistics.

Tabla 9: Historia de usuario Nro. 7: Colgar llamada.

Historia de Usuario	
Número: 7	Nombre de Historia de Usuario: Colgar llamada.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Dianelys López Sanchez Yadier Abel De Quesada	Iteración Asignada: 1
Prioridad en negocio: Media	Puntos estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1
Descripción: El supervisor debe darle fin a la comunicación entre un abonado y el agente.	
Observaciones: El supervisor debe estar autenticado en el sistema y tener a dicho agente dentro de los que le fue asignado previamente por el administrador.	
Prototipo de interface:	

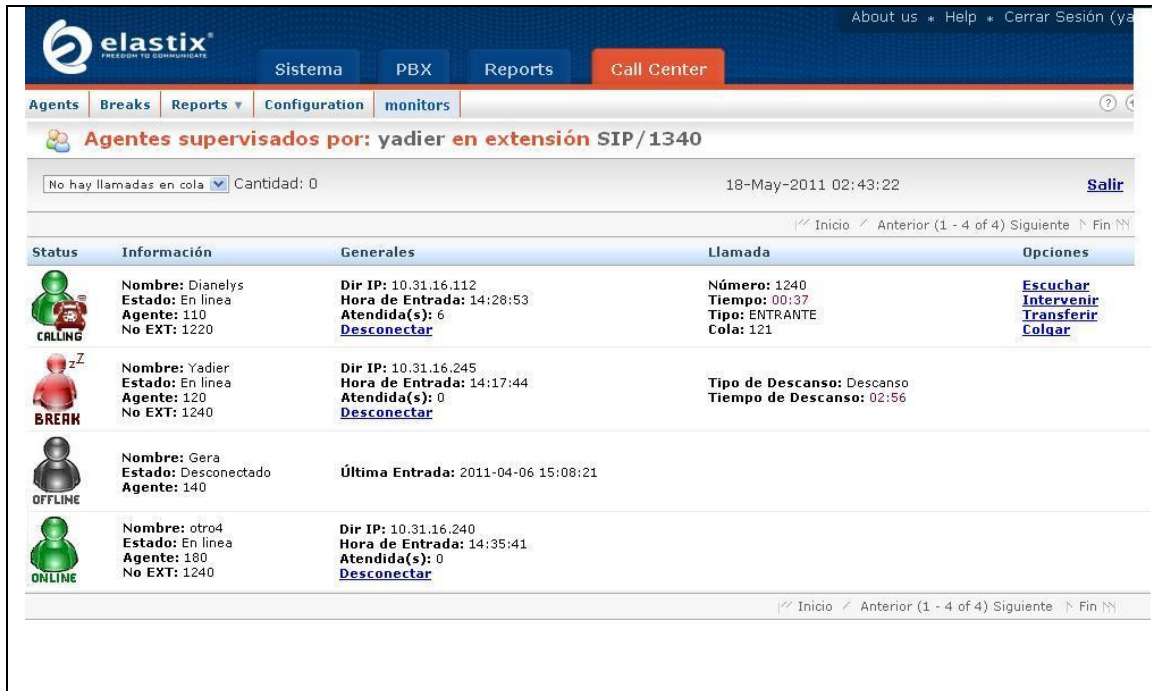


The screenshot shows the Elastix Call Center interface. At the top, there are navigation tabs: Sistema, PBX, Reports, and Call Center. Below this, there are sub-tabs: Agents, Breaks, Reports, Configuration, and monitors. The main heading is "Agentes supervisados por: yadier en extensión SIP/1340". Below this, there is a status bar showing "No hay llamadas en cola" and "Cantidad: 0". The main content area displays a list of agents with columns for Status, Información, Generales, Llamada, and Opciones. The agents listed are: Dianelys (CALLING), Yadier (BREAK), Gera (OFFLINE), and otro4 (ONLINE).

Status	Información	Generales	Llamada	Opciones
CALLING	Nombre: Dianelys Estado: En línea Agente: 110 No EXT: 1220	Dir IP: 10.31.16.112 Hora de Entrada: 14:28:53 Atendida(s): 6 Desconectar	Número: 1240 Tiempo: 00:37 Tipo: ENTRANTE Cola: 121	Escuchar Intervenir Transferir Colgar
BREAK	Nombre: Yadier Estado: En línea Agente: 120 No EXT: 1240	Dir IP: 10.31.16.245 Hora de Entrada: 14:17:44 Atendida(s): 0 Desconectar	Tipo de Descanso: Descanso Tiempo de Descanso: 02:56	
OFFLINE	Nombre: Gera Estado: Desconectado Agente: 140	Última Entrada: 2011-04-06 15:08:21		
ONLINE	Nombre: otro4 Estado: En línea Agente: 180 No EXT: 1240	Dir IP: 10.31.16.240 Hora de Entrada: 14:35:41 Atendida(s): 0 Desconectar		

Tabla 10: Historia de usuario Nro. 8: Desconectar agente.

Historia de Usuario	
Número: 8	Nombre de Historia de Usuario: Desconectar agente.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Dianelys López Sanchez Yadier Abel De Quesada	Iteración Asignada: 1
Prioridad en negocio: Media	Puntos estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1
Descripción: Se pone de manifiesto cuando el supervisor puede lograr que un agente activo y permaneciendo dentro del sistema recese sus funciones.	
Observaciones: El supervisor debe estar autenticado en el sistema y tener a dicho agente dentro de los que le fue asignado previamente por el administrador.	
Prototipo de interface:	

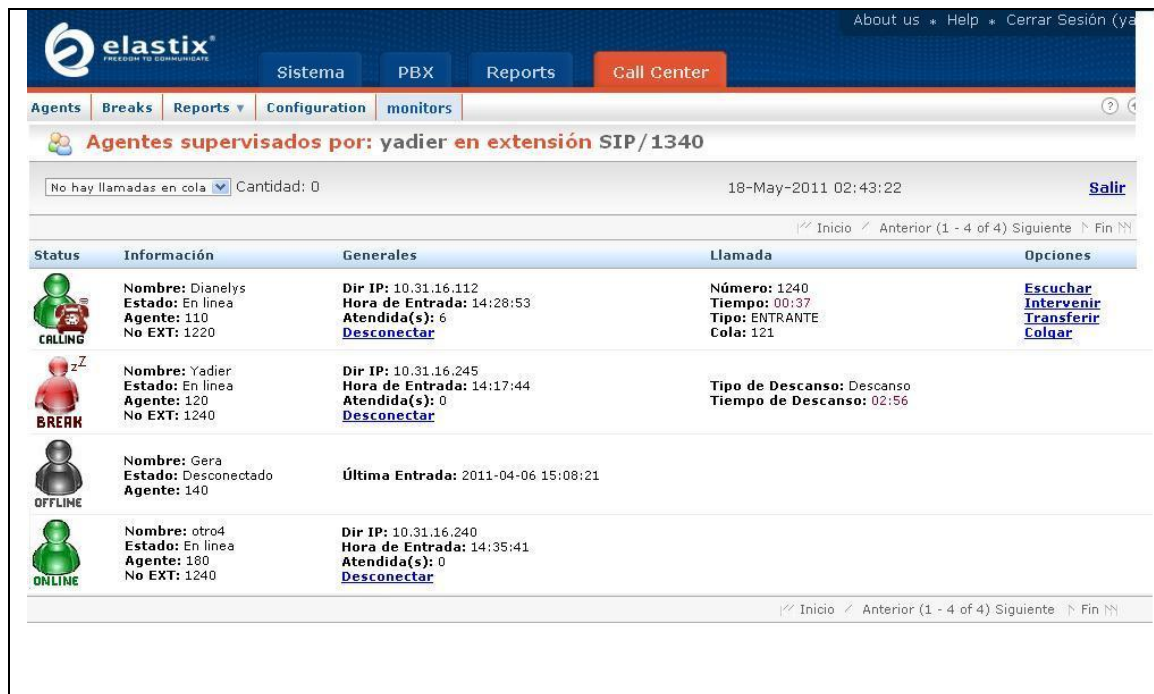


The screenshot shows the Elastix Call Center interface. At the top, there are navigation tabs: Sistema, PBX, Reports, and Call Center. Below these, there are sub-tabs: Agents, Breaks, Reports, Configuration, and monitors. The main heading is "Agentes supervisados por: yadier en extensión SIP/1340". Below this, there is a status bar showing "No hay llamadas en cola" and "Cantidad: 0". The main content area displays a list of agents with columns for Status, Información, Generales, Llamada, and Opciones. The agents listed are Dianelys (CALLING), Yadier (BREAK), Gera (OFFLINE), and otro4 (ONLINE).

Status	Información	Generales	Llamada	Opciones
CALLING	Nombre: Dianelys Estado: En línea Agente: 110 No EXT: 1220	Dir IP: 10.31.16.112 Hora de Entrada: 14:28:53 Atendida(s): 6 Desconectar	Número: 1240 Tiempo: 00:37 Tipo: ENTRANTE Cola: 121	Escuchar Intervenir Transferir Colgar
BREAK	Nombre: Yadier Estado: En línea Agente: 120 No EXT: 1240	Dir IP: 10.31.16.245 Hora de Entrada: 14:17:44 Atendida(s): 0 Desconectar	Tipo de Descanso: Descanso Tiempo de Descanso: 02:56	
OFFLINE	Nombre: Gera Estado: Desconectado Agente: 140	Última Entrada: 2011-04-06 15:08:21		
ONLINE	Nombre: otro4 Estado: En línea Agente: 180 No EXT: 1240	Dir IP: 10.31.16.240 Hora de Entrada: 14:35:41 Atendida(s): 0 Desconectar		

Tabla 11: Historia de usuario Nro. 9: Transferir llamada.

Historia de Usuario	
Número: 9	Nombre de Historia de Usuario: Transferir llamada.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Dianelys López Sanchez Yadier Abel De Quesada	Iteración Asignada: 1
Prioridad en negocio: Media	Puntos estimados: 1
Riesgo en Desarrollo: Alta	Puntos Reales: 1
Descripción: Consiste en redireccionar una llamada manteniendo al abonado en línea y la misma ser tomada por el supervisor y no por otros agentes.	
Observaciones: El supervisor debe estar autenticado en el sistema y tener a dicho agente dentro de los que le fue asignado previamente por el administrador.	
Prototipo de interface:	



The screenshot shows the Elastix Call Center monitoring interface. At the top, there are navigation tabs for 'Sistema', 'PBX', 'Reports', and 'Call Center'. Below this, there are sub-tabs for 'Agents', 'Breaks', 'Reports', 'Configuration', and 'monitors'. The main heading is 'Agentes supervisados por: yadier en extensión SIP/1340'. Below the heading, there is a status bar showing 'No hay llamadas en cola' and 'Cantidad: 0'. The main content area displays a list of agents with the following columns: Status, Información, Generales, Llamada, and Opciones. The agents listed are: Dianelys (CALLING), Yadier (BREAK), Gera (OFFLINE), and otro4 (ONLINE). Each agent entry includes their name, state, IP address, and various call statistics.

Status	Información	Generales	Llamada	Opciones
CALLING	Nombre: Dianelys Estado: En línea Agente: 110 No EXT: 1220	Dir IP: 10.31.16.112 Hora de Entrada: 14:28:53 Atendida(s): 6 Desconectar	Número: 1240 Tiempo: 00:37 Tipo: ENTRANTE Cola: 121	Escuchar Intervenir Transferir Colgar
BREAK	Nombre: Yadier Estado: En línea Agente: 120 No EXT: 1240	Dir IP: 10.31.16.245 Hora de Entrada: 14:17:44 Atendida(s): 0 Desconectar	Tipo de Descanso: Descanso Tiempo de Descanso: 02:56	
OFFLINE	Nombre: Gera Estado: Desconectado Agente: 140	Última Entrada: 2011-04-06 15:08:21		
ONLINE	Nombre: otro4 Estado: En línea Agente: 180 No EXT: 1240	Dir IP: 10.31.16.240 Hora de Entrada: 14:35:41 Atendida(s): 0 Desconectar		

2.6 Planificación

En la fase de planificación se realiza la estimación del esfuerzo que costará la implementación de cada historia de usuario, como en la metodología ágil XP las métricas son libres, puede utilizarse cualquier criterio para medir el desempeño del proyecto en cuestión. Una de las métricas más utilizadas en este tipo de metodología es la medida de puntos; un punto en esta métrica es considerado como una semana de trabajo, donde los miembros de los equipos de desarrollo trabajan sin interrupciones.

2.6.1 Estimación de esfuerzo por historias de usuarios.

Para el desarrollo de la aplicación propuesta se realizó una estimación de esfuerzo por cada una de las historias de usuario identificadas, resultados que se muestran a continuación:

Tabla 12: Estimación de esfuerzo por HU

Historia de Usuario	Puntos de estimación
Insertar supervisor	1
Modificar supervisor	1

Eliminar Supervisor	1
Autenticar supervisor	1
Supervisar agentes en tiempo real	1
Intervenir llamadas	3
Colgar llamada	1
Desconectar agente	1
Transferir llamada	1

2.6.2 Plan de Iteraciones.

Después de ser identificadas las historias de usuario y estimado el esfuerzo dedicado a la realización de cada una de estas historias de usuario se procede a la realización de la planificación de la etapa de implementación del presente trabajo. Para un mejor desempeño del equipo de desarrollo se estableció una división de la implementación en 3 iteraciones:

2.6.2.1 Iteración 1

En la iteración 1 se implementarán las historias de usuario 1, 2, 3 y 4. Al finalizar esta iteración se contará con la primera versión de prueba de la aplicación.

2.6.2.2 Iteración 2

En la iteración 2 se implementarán las historias de usuario 5 y 6 las cuales como las demás son prioritarias para la aplicación, siendo agregadas, como en la iteración anterior, el concepto de supervisión.

2.6.2.3 Iteración 3

La iteración 3 tendrá como objetivo principal la implementación de las historias de usuario 7, 8 y 9 que son las encargadas de complementar el resultado de lo implementado en las iteraciones anteriores. Al terminar esta iteración se contará con una versión 1.0 del producto final y como resultado de esta, el sistema se pondrá en funcionamiento para evaluar su desempeño.

2.6.3 Plan de duración de las iteraciones.

El plan de duración de las iteraciones es el encargado de mostrar las historias de usuario que serán implementadas en cada una de las iteraciones, así como la duración estimada y el orden de implementación de cada una de ellas.

Tabla 13: Plan de duración de iteraciones

Iteración	Orden de la Historias de usuario a implementar.	Duración total
1	Insertar supervisor Modificar supervisor Eliminar supervisor Autenticar supervisor	4 semanas
2	Supervisar agentes en tiempo real. Intervenir llamadas	4 semanas
3	Colgar llamada Desconectar agente Transferir llamada	3 semanas

2.6.4 Plan de entregas

Se presenta el plan de entrega estimado para la fase de implementación, detallando la fecha de fin de cada iteración, los productos obtenidos, así como el módulo sobre el cual se está implementando.

SGS: subsistema de gestión de supervisores

SSATR: subsistema de supervisión de agentes en tiempo real

Tabla 14: Plan de entregas

Módulo	Final de la Iteración 1 (30 de marzo del 2011)	Final de la Iteración 2 (30 de abril del 2011)	Final de la Iteración 3 (20 de mayo del 2011)
Call-Center	SGS v0.1	SSATR 0.1	SSATR v1.0

2.7 Conclusiones

En este capítulo se han expuesto las principales características del sistema, en función de realizar un sistema capaz de cumplir con las expectativas del cliente. A favor de dicho sistema se expone una propuesta de los procesos que serán objetos de automatización y la planificación que se planteó el equipo de desarrollo, compuesto por iteraciones donde de forma incremental se implementará el sistema propuesto. Además, se obtiene un listado de características no funcionales que la aplicación debe cumplir.

CAPÍTULO 3: DISEÑO, IMPLEMENTACIÓN Y PRUEBA

3.1 Introducción

El momento cumbre del desarrollo de un proyecto es la implementación de cada una de las funcionalidades requeridas por el mismo. Realmente reconforta ver como lo que ha sido planificado se materializa poco a poco. También es conocido que existe otro momento, estrechamente vinculado a la implementación, al cual en muchas ocasiones no se le brinda la importancia que requiere. Es el caso de las pruebas que se realizan al sistema, las cuales no deben ser vistas como trabas sino como pilares que garantizan que la implementación transite por caminos firmes. En la actualidad ambos momentos están muy entrelazados debido a que existe la tendencia de diseñar las pruebas antes que la implementación. Las características del desarrollo inherente a la implementación y prueba a las cuales se hace alusión en el presente capítulo se encuentran condicionadas a las propuestas por el framework de desarrollo utilizado.

La metodología XP plantea que la implementación de un producto debe realizarse de forma iterativa, esta característica trae consigo que después del desarrollo de cada iteración se obtenga un producto funcional que debe ser mostrado al cliente y previamente probado para incrementar la visión de los desarrolladores y clientes de posibles cambios. En el presente capítulo se detalla la construcción del sistema, el diseño de la base de datos, además de exponer las tareas generadas por cada HU y la arquitectura del sistema, así como las pruebas de aceptación efectuadas sobre el mismo.

3.2 Patrón de Arquitectura

Los sistemas de software crecen proporcionalmente, hoy se han perfeccionado tanto los algoritmos, que los códigos han dejado de convertirse en un problema. Actualmente el diseño de los sistemas constituye un nuevo reto, los patrones arquitectónicos nos expresan la estructura fundamental para el sistema de software a desarrollar. Para el desarrollo del sistema el patrón de arquitectura para aplicaciones web que se utilizó es el Modelo -Vista -Controlador (MVC).

3.2.1 Modelo Vista Controlador (MVC)

Elastix está basado en un patrón clásico del diseño web conocido como arquitectura MVC, que está formado por tres niveles: modelo, vista y controlador el cual separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón de llamada y retorno MVC, se ve

Supervisión de agentes en tiempo real del Módulo Call-Center de Elastix

frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista. (9)

Esta arquitectura se encuentra compuesta por tres capas principales en donde se encapsula el concepto modular, obteniendo así tres características claves de diseño y desarrollo que son: Mantenibilidad, Modularidad y Reutilización; consiguiendo con esto un modelo escalable lo cual es sumamente útil para la Web. (10)

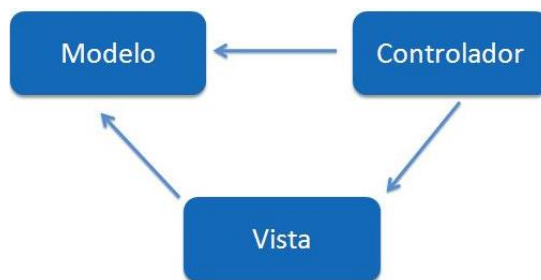


Figura 3 Patrón arquitectónico MVC

Es importante destacar que el framework NEO establece un estándar en su contenido o dentro de una carpeta. La carpeta **themes** define la capa vista, la carpeta **libs** define la capa modelo y el archivo **index.php** define la capa control. (11)

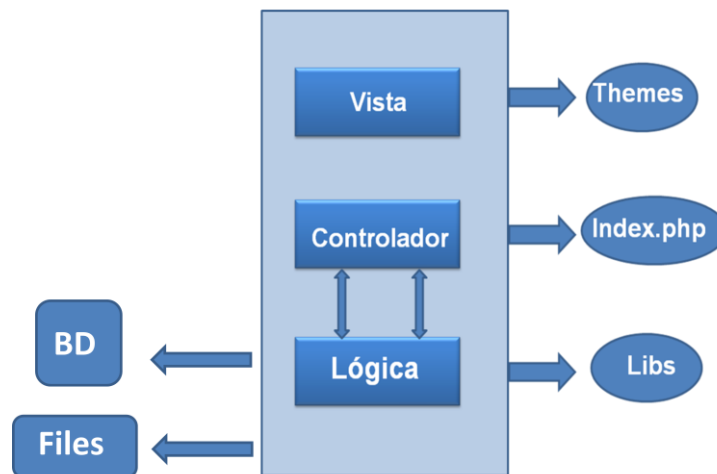


Figura 4. Framework Neo basado en arquitectura MVC2

Tabla 15: Definiciones de carpetas

Carpeta	Descripción
themes/	Están la definiciones de plantillas (HTML, javascript, CSS)
index.php	El punto de entrada común para todas las pantallas de la interface de Elastix. Es quien decide qué pantalla mostrar dependiendo de los parámetros recibidos.
libs/	Aquí se encuentran las librerías (código reciclable)

A continuación se muestran las capas basadas en la arquitectura Modelo Vista Controlador empleada por el framework Neo:

Capa de Vista

Esta capa está compuesta por el código HTML, por los estilos (CSS) y los scripts (como javascript).

Capa de Control

Esta capa es la encargada de recoger y entregar los datos a la capa de vista. Para esto se usan sesiones y una librería de manejo de plantillas llamada Smarty.

Para recoger los datos se usan los métodos GET y POST establecidos en los formularios HTML (etiquetas form).

Capa de Modelo

La participación de esta capa es para realizar transacciones, obtener y enviar datos a la capa de datos. Típicamente el modelo de clases contendrá funciones para consultar, insertar y actualizar información de la base de datos. Idealmente debería ser independiente de la base de datos utilizada y notifica los cambios que se hacen en el sistema.

3.2.3 Ventajas del uso del MVC

La ventaja fundamental de esta división en capas se encuentra centrada en la medida de que cada una de estas capas puede ser sustituida sin afectar a las otras ya que provee una separación total entre la lógica del negocio y presentación, además pueden existir diferentes vistas para un mismo modelo trayendo como resultado que la división de código de este estilo arquitectónico haga más fácil la portabilidad y la adaptación a los requerimientos del usuario. Al existir la separación de vistas, controladores y modelos es más sencillo realizar labores de mejora como:

- Agregar nuevas vistas.
- Agregar nuevas formas de recolectar las órdenes del usuario.
- Modificar los objetos de negocios bien sea para mejorar el performance o para migrar la tecnología.
- Las labores de mantenimiento simplificadas.
- Las correcciones solo se deben hacer en un solo lugar.

3.3 Patrones de Diseño

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces. Un patrón de diseño es una solución a un problema de diseño. Para que una solución sea considerada un patrón debe poseer ciertas características, una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores. Otra es que debe ser reusable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias.

3.3.1 Patrones para Asignar Responsabilidades (GRASP)

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. Dentro de los patrones GRASP utilizados en el desarrollo del sistema se encuentran los siguientes:

Experto: Asignar una responsabilidad a la clase que tiene la información necesaria para cumplirla.

La clase Supervisores y la clase Agente poseen la información requerida para cumplir las responsabilidades que le corresponden.

Creador: Asignar a la clase B la responsabilidad de crear una instancia de clase A, si se cumple una de las siguientes condiciones:

- B contiene. A
- B registra A
- B agrega A
- B utiliza A muy de cerca
- B tiene los datos de inicialización de A.

La clase Supervisores y la clase Agente, ambas son responsables de crear una instancia de la clase paloSantoDB.

Controlador: Un Controlador es un objeto de interfaz no destinada al usuario que se encarga de manejar un evento del sistema. Define además el método de su operación.

La clase CSupervision es responsable de atender varios eventos del sistema, dicha clase es la encargada de controlar los estados de los agentes y sus acciones correspondientes.

Alta Cohesión: es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme.

La clase Supervisores se le asignan responsabilidades con el objetivo de trabajar en la misma área de aplicación y que no tienen mucha complejidad.

Bajo Acoplamiento: es una medida de la fuerza con que una clase está conectada a otras clases, con que las conoce y con que recurre a ellas. Acoplamiento bajo significa que una clase no depende de muchas clases.

Las clases se les asignan responsabilidades de forma tal que solo se comunican con la clase paloSantoDB y las clases controladoras correspondientes a cada una de ellas. Ejemplo: la clase Supervisores solo se relaciona con la clase paloSantoDB.

3.4 Modelo Físico de la Base de Datos

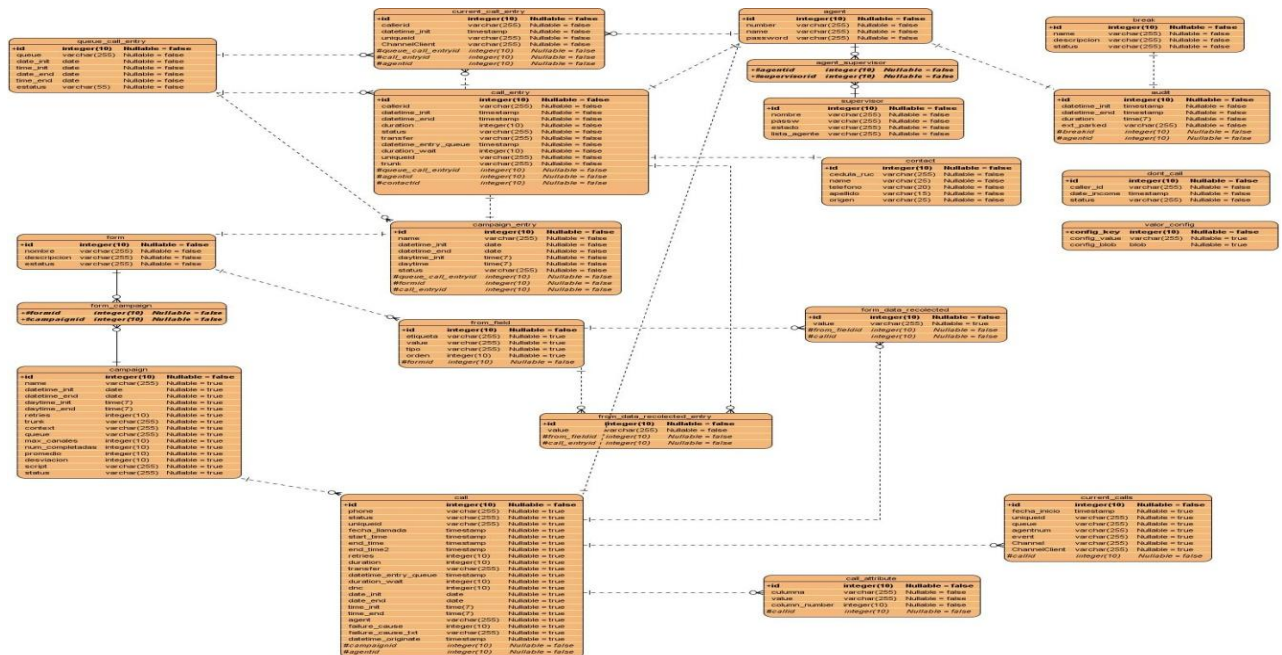


Figura 5. Base de Datos del módulo Call-Center

3.4.1 Especificaciones de la Base de Datos empleada

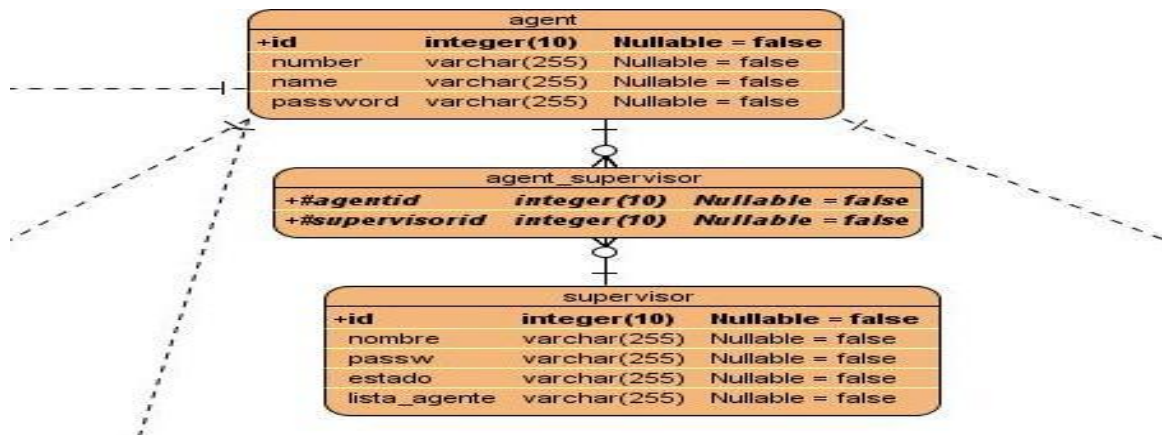


Figura 6. Fragmento de la Base de Datos del módulo Call-Center

Inicialmente la base de datos del Call-Center contaba con 19 tablas, el sistema para su implementación le agrega la tabla supervisor que tiene una relación de mucho a mucho con la tabla agente por lo que se crea la nueva tabla con las llaves primaria de cada una de ellas.

3.5 Tarjetas Clase – Responsabilidad – Colaborador

Sin dudas una de las tareas más importantes en cuanto a diseño de una aplicación se refiere la constituye el definir correctamente las clases que contendrán la lógica del negocio. La metodología XP no obliga a la realización de diagramas UML para la presentación de las mismas y en su lugar propone el uso de tarjetas CRC (Clase-Responsabilidad-Colaboración), técnica que ayuda a evitar el enfoque procedimental destacando la orientación a objetos.

El nombre de la clase se coloca en forma de título en la tarjeta, en la parte izquierda las funcionalidades (responsabilidades) y en la parte derecha las clases que se implican en cada funcionalidad (colaboración). A continuación se describen las tarjetas CRC pertenecientes, permitiendo definir y simular los escenarios que garantizan el buen funcionamiento del diseño.

Tabla 16: Tarjeta CRC Clase Agente

Clase: Agente	
Responsabilidad	Colaboración
Transferir llamadas Desconectar agente Colgar llamada Intervenir llamada	AGI_AsteriskManager paloSantoDB

Tabla 17: Tarjeta CRC Clase Supervisores

Clase: Supervisores	
Responsabilidad	Colaboración
Gestionar supervisores	paloSantoDB

Tabla 18: Tarjeta CRC Clase CSupervision

Clase: CSupervision	
Responsabilidad	Colaboración
Monitorear en tiempo real	paloSantoDB paloSantoGrid Agentes

3.6 Tareas de Ingeniería

Las tareas de la ingeniería son escritas por el equipo de desarrollo a partir de las historias de usuario elaboradas por el cliente, las cuales describen las tareas correspondientes a cada Historia de usuario, dando un detalle más profundo para realizar una implementación de las mismas y estimando un tiempo más cercano a la realidad para cada una de ellas.

Tabla 19: Muestra de una Tarea de Ingeniería

Tarea de Ingeniería	
Número Tarea: [Los números deben ser consecutivos]	Número Historia de Usuario: [Número de la historia de usuario a la que pertenece la tarea]
Nombre Tarea: [Nombre que identifica a la tarea.]	
Tipo de Tarea : [Las tareas pueden ser de: Desarrollo, Corrección, Mejora, Otra(Especificar)]	Puntos Estimados: [Tiempo en semanas que se le asignará. (Estimado)]
Programador Responsable: [Nombre y Apellidos del programador]	
Descripción: [Breve descripción de la tarea.]	

Tabla 20: Tarea de Ingeniería Nro. 1: Insertar Supervisor

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 1
Nombre Tarea: Insertar Supervisor	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Yadier Abel De Quesada Ricardo Dianelys López Sanchez	
Descripción: El administrador del sistema introduce los datos del supervisor como son: usuario, contraseña, verificación de la misma y listado de agentes asignados.	

Tabla 21: Tarea de Ingeniería Nro. 1: Modificar supervisor

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 2
Nombre Tarea: Modificar supervisor	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Yadier Abel De Quesada Ricardo Dianelys López Sanchez	
Descripción: El administrador, una vez registrado el supervisor, puede modificar los campos: contraseña, verificación de la misma y listado de agentes asignados.	

Tabla 22: Tarea de Ingeniería Nro. 1: Eliminar supervisor

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 3
Nombre Tarea: Eliminar supervisor	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Yadier Abel De Quesada Ricardo Dianelys López Sanchez	
Descripción: El administrador tiene el permiso de eliminar a un supervisor ya	

registrado.

Tabla 23: Tarea de Ingeniería Nro. 1: Autenticar supervisor

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 4
Nombre Tarea: Autenticar supervisor	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Yadier Abel De Quesada Ricardo Dianelys López Sanchez	
Descripción: El supervisor, anteriormente registrado en el sistema por el administrador, procede a entrar al sistema mediante su usuario y contraseña asignada, ambos como campos obligatorios y por último la extensión.	

Tabla 24: Tarea de Ingeniería Nro. 1: Supervisar agentes en tiempo real

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 5
Nombre Tarea: Supervisar agentes en tiempo real	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Yadier Abel De Quesada Ricardo Dianelys López Sanchez	
Descripción: El sistema muestra un listado de agentes con su estado, información general del mismo (nombre, estado, número de agente, número de extensión), así como los datos de las llamadas como son tipo, número y tiempo, un sistema de estadísticas que posee la cantidad de llamadas atendidas, la cantidad de llamadas perdidas y el total entre ambas.	

Tabla 25: Tarea de Ingeniería Nro. 1: Escuchar canal

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 6
Nombre Tarea: Escuchar canal	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Yadier Abel De Quesada Ricardo Dianelys López Sanchez	
Descripción: El supervisor tiene la posibilidad de escuchar la llamada entre un agente y un abonado sin tener el permiso de hablar con los mismos o sea espiar de forma pasiva el canal.	

Tabla 26: Tarea de Ingeniería Nro. 2: Intervenir canal

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 6
Nombre Tarea: Intervenir canal	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Yadier Abel De Quesada Ricardo Dianelys López Sanchez	
Descripción: Mediante esta tarea el supervisor interviene el canal dándole la oportunidad de hablar solo con el agente, espiando así de forma activa el canal.	

Tabla 27: Tarea de Ingeniería Nro. 1: Colgar llamada

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 7
Nombre Tarea: Colgar llamada	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1

Programador Responsable: Yadier Abel De Quesada Ricardo Dianelys López Sanchez
Descripción: Consiste en suspender la llamada sin la necesidad de sacar al agente del sistema eliminando así su comunicación con el abonado y dejándolo libre para futuras atenciones.

Tabla 28: Tarea de Ingeniería Nro. 1: Desconectar agente

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 8
Nombre Tarea: Desconectar agente	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Yadier Abel De Quesada Ricardo Dianelys López Sanchez	
Descripción: Consiste en sacar al agente del sistema, manteniéndolo inactivo dentro del mismo, cambiando así su estado, que pasaría de conectado a desconectado.	

Tabla 29: Tarea de Ingeniería Nro. 1: Transferir llamada

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 9
Nombre Tarea: Transferir llamada	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Yadier Abel De Quesada Ricardo Dianelys López Sanchez	
Descripción: Consiste en redireccionar una llamada manteniendo al abonado en línea y esta ser tomada por el supervisor liberando al agente de esta responsabilidad.	

3.7 Pruebas

XP anima a probar constantemente tanto como sea posible. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección. También permite aumentar la seguridad de evitar efectos colaterales no deseados a la hora de realizar modificaciones y refactorizaciones. XP divide las pruebas del sistema en dos grupos: pruebas unitarias y pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida diseñadas por el cliente final. Este tipo de pruebas, constituyen uno de los mayores avances en la programación desde la aparición de la orientación a objetos y aunque son recomendadas, su uso o no, queda a merced de los desarrolladores.

3.7.1 Pruebas unitarias

Son las encargadas de que en el transcurso de la implementación de un sistema, cada uno de los desarrolladores tiene que ir probando constantemente lo que va obteniendo para garantizar que las funcionalidades exigidas por el cliente estén siendo implementadas correctamente. Estas no generan artefactos y no son directamente palpables para el cliente.

3.7.2 Pruebas de aceptación

Las pruebas de aceptación o funcionales son la mejor forma de probar la aplicación de extremo a extremo: desde la petición realizada por un navegador hasta la respuesta enviada por el servidor. Las pruebas funcionales prueban todas las capas de la aplicación: el sistema de enrutamiento, el modelo, las acciones y las plantillas. En realidad, son muy similares a lo que se hace manualmente cada vez que se añade o modifica una acción y se prueban dichos cambios en el navegador para comprobar que todo funciona bien al pulsar sobre los enlaces y botones y que todos los elementos se muestran correctamente en la página. En otras palabras, lo que se hace es probar un escenario correspondiente a la historia de usuario que se acaba de implementar en la aplicación.

Dichas pruebas estarán divididas en las siguientes secciones:

- **Código:** servirá como identificador de la prueba realizada, a su vez será sugerente al nombre de la prueba a la que hace referencia.
- **Número Historia de Usuario:** tendrá el número de la HU a la que hace referencia la prueba a realizar.
- **Nombre:** nombre que se le da a la prueba a realizar.

- **Descripción:** se describe la funcionalidad que se desea probar.
- **Condiciones de Ejecución:** mostrará las condiciones que deben cumplirse para poder llevar a cabo el caso de prueba.
- **Entradas / Pasos de Ejecución:** se hará la descripción de cada uno de los pasos seguidos durante el desarrollo de la prueba, se tendrá en cuenta cada una de las entradas que hace el usuario con el objetivo de ver si se obtiene el resultado esperado.
- **Resultado esperado:** se hará una breve descripción del resultado que se espera obtener con la prueba realizada.
- **Evaluación de la prueba:** acorde al resultado de la prueba realizada se emitirá una evaluación sobre la misma:
 1. Bien: cuando el resultado de la prueba es exactamente el esperado por el usuario.
 2. Parcialmente bien: cuando el resultado no es completamente el esperado por el cliente o usuario de la aplicación y muestra resultados erróneos o fuera de contexto.
 3. Mal: cuando el resultado de la prueba realizada genera un error de codificación en la aplicación o muestra como resultado elementos no deseados o fuera de contexto, trayendo como consecuencia que la funcionalidad requerida por el cliente no tenga resultado, lo que invalida también la UH.

Las pruebas de aceptación correspondiente a cada una de las HU se encuentran en el Anexo II.

3.8 Conclusiones

En este capítulo se describieron las fases de diseño, implementación y prueba. Se realizó una breve descripción del diseño del sistema, el cual se realizó basado en la simplicidad. Se detallaron cada una de las tareas que se realizaron en las tres iteraciones del sistema. Se abordó acerca de la importancia de las pruebas en la metodología XP, en especial las pruebas de aceptación, ya que éstas miden la satisfacción del cliente con el producto desarrollado.

CAPÍTULO 4: ESTUDIO DE LA FACTIBILIDAD

4.1 Introducción

En este capítulo como parte del proceso de planificación de desarrollo de software tiene una vital importancia la estimación, la cual consiste en determinar con cierto grado de certeza los recursos necesarios para el desarrollo del mismo, ya sean recursos de hardware, software, esfuerzo, tiempo y costo. En el presente capítulo se realizará un estudio de factibilidad para la realización del sistema propuesto mediante una estimación de tamaño, esfuerzo y planificación necesaria para llevar a cabo el mismo.

4.2 Modelo matemático COCOMO II.

COCOMO, propuesto y desarrollado por Barry Boehm, es uno de los modelos de estimación de costos mejor documentados y utilizados. El modelo permite determinar el esfuerzo y tiempo que se requiere en un proyecto de software a partir de una medida del tamaño del mismo expresada en el número de líneas de código que se estimen generar para la creación del producto software.

Consiste básicamente en la aplicación de ecuaciones matemáticas sobre los Puntos de Función sin ajustar o la cantidad de líneas de código (SLOC, Source Lines of Code) estimados para un proyecto. Estas ecuaciones se encuentran ponderadas por ciertos factores de costo (cost drivers) que influyen en el esfuerzo requerido para el desarrollo del software.

Está compuesto por tres modelos que se adaptan tanto a las necesidades de los diferentes sectores descritos, como al tipo y cantidad de información disponible en cada etapa del ciclo de vida de desarrollo, lo que se conoce por granularidad de la información.

El estudio de factibilidad de este sistema se realizó mediante el modelo de diseño temprano que se utiliza en las primeras etapas del desarrollo, en las cuales se evalúan las alternativas de hardware y software de un proyecto. En estas etapas se tiene poca información, se conoce muy poco del tamaño del producto a ser desarrollado, de la naturaleza de la plataforma, del personal a ser incorporado al proyecto o aspectos específicos del proceso a utilizar. Ese nivel de detalle en este modelo es consistente con el nivel general de información disponible y con el nivel general de estimación detallada que es necesaria en estas etapas, lo que concuerda con el uso de Puntos de Función. Para estimar tamaño usa Puntos de Función No Ajustados como métrica de medida y un número reducido de factores de costo.

4.3 Características del Proyecto.

El primer paso a llevar a cabo para la estimación del proyecto consiste en la obtención de los Puntos de Función desajustados, los cuales están dados por la suma de cada una de las entradas, las salidas y las consultas externas del sistema, así como los archivos lógicos internos y de interfaz externos. A continuación se muestran cada una de estas características aplicadas al software en cuestión.

4.3.1 Entradas externas.

Se definen como un proceso elemental mediante el cual ciertos datos cruzan la frontera del sistema desde afuera hacia adentro. En el caso particular de la aplicación propuesta se cuenta con ocho entradas externas, especificadas en la siguiente tabla:

Tabla 30: Entradas Externas

Nombre de la entrada externa.	Cantidad de ficheros.	Cantidad de elementos de datos.	Clasificación (Simple, Media, Compleja)
Insertar supervisor	1	4	Simple
Modificar supervisor	1	2	Simple
Eliminar supervisor	1	1	Simple
Autenticar supervisor	1	3	Simple
Intervenir llamada	1	1	Compleja
Colgar llamada	1	1	Media
Desconectar agente	1	1	Media
Transferir llamada	1	1	Compleja
TOTAL		14	

4.3.2 Salidas Externas

Se definen como un proceso elemental con componentes de entrada y de salida mediante el cual datos simples y datos derivados (esto es, datos que se calculan a partir de otros datos) cruzan las fronteras del sistema desde adentro hacia afuera. Las salidas externas vinculadas al proyecto se describen a continuación. (12)

Tabla 31: Salidas Externas

Nombre de la salida externa.	Cantidad de ficheros.	Cantidad de elementos de datos.	Clasificación (Simple, Media. Compleja)
Supervisar agentes en tiempo real	1	6	Compleja
TOTAL		6	

4.3.3 Consultas Externas.

Se definen como un proceso elemental con componentes de entrada y de salida donde un actor del sistema rescata datos de uno o más Archivos Lógicos Internos o Archivos de Interfaz Externos. Los datos de entrada no actualizan ni mantienen ningún archivo (lógico interno o de interfaz externo) y los datos de salida no contienen datos derivados (es decir, los datos de salida son básicamente los mismos que se obtienen de los archivos). (12) En este caso no fueron identificados.

Tabla 32: Consultas externas

Nombre de la Petición.	Cantidad de ficheros.	Cantidad de elementos de datos.	Clasificación (Simple, Media. Compleja)
TOTAL		0	

4.3.4 Archivos Lógicos Internos.

Constituyen un grupo de datos relacionados lógicamente e identificables por el usuario, que residen enteramente dentro de los límites del sistema y se mantienen a través de entradas externas. (12)

Tabla 33: Fichero lógico interno.

Nombre de la petición.	Cantidad de ficheros.	Cantidad de elementos de datos.	Clasificación (Simple, Media. Compleja)
Base de Datos: Call-Center	1	3 (tablas)	Media
TOTAL		3	

4.3.5 Archivos de Interfaz Externos.

Son un grupo de datos relacionados lógicamente e identificables por el usuario, que se utilizan solamente para fines de referencia. Los datos residen enteramente fuera de los límites del sistema y se mantienen por las Entradas Externas de otras aplicaciones, es decir, cada Archivo de Interfaz Externo es un Archivo Lógico Interno de otra aplicación. (12)

Tabla 34: Interfaces externas

Nombre de la interfaz externa.	Cantidad de ficheros.	Cantidad de elementos de datos.	Clasificación (Simple, Media, Compleja)
TOTAL		0	

4.3.6 Puntos de Función Desajustados.

La siguiente tabla está basada en las características del sistema anteriormente expuestas, el producto de la cantidad existente de cada una de ellas y el peso correspondiente a las mismas dan como resultado final, los puntos de función desajustados pertenecientes al proyecto.

Tabla 35: Puntos de función desajustados.

Elementos.	Simple.		Medio.		Complejo.		Subtotal.
	No.	Peso.	No.	Peso.	No.	Peso.	
Entradas externas.	10	3	2	4	2	6	50
Salidas externas.	0	4	0	5	6	7	42
Consultas externas.	0	3	0	4	0	6	0
Fichero lógico interno.	0	7	3	10	0	15	30
Fichero interfaz externo.	0	5	0	7	0	10	0
Total (UFP):							122

Apoyarse en la tabla de peso del factor de complejidad. [Anexo I]

4.4 Cálculos de instrucciones fuentes, esfuerzo, tiempo de desarrollo, cantidad de hombres y costo.

4.4.1 Cálculo de instrucciones fuentes.

Una vez obtenida la cantidad total de puntos de función desajustados pertenecientes al proyecto, se procede a calcular la cantidad de instrucciones fuentes del mismo, para lo cual se utiliza la ecuación que se muestra seguidamente:

$$SLOC = UFP \times ratio$$

En la cual:

SLOC: Cantidad de instrucciones fuentes.

UFP: Puntos de función desajustados. (122)

ratio : Conversión de puntos de función desajustados a líneas de código para el lenguaje PHP. (59)

Según la ecuación anterior se obtienen los siguientes resultados:

$$SLOC = 122 \times 59$$

$$SLOC = 7198 = 7.1 \text{ KSLOC}$$

KSLOC : Tamaño del software a desarrollar.

Las características del sistema y valores obtenidos anteriormente han sido plasmadas en la siguiente tabla:

Tabla 36: Características del sistema

Características	Valor
Puntos de función desajustados.	122
Lenguaje (PHP).	59
Instrucciones fuentes por puntos de función.	7198
Instrucciones fuentes.	7.2 KSLOC

4.4.2 Cálculo de esfuerzo nominal.

A continuación se procede al cálculo del esfuerzo nominal, ecuación que se toma como base tanto en el método de diseño preliminar, al cual se hace alusión anteriormente, como en el modelo Post arquitectura, ambos definidos por COCOMO II.

$$PM_{Nominal} = Ax(Size)^B$$

$$PM_{Nominal} = 2.94 x (7.2)^{1.02}$$

$$PM_{Nominal} = 22.05 \text{ meses/hombres}$$

Donde:

$PM_{Nominal}$: Esfuerzo nominal requerido en meses /hombres.

$Size$: Tamaño estimado del software en Puntos de Función sin Ajustar. (7.2 KSLOC)

A : Constante que se utiliza para capturar los efectos multiplicativos en el esfuerzo requerido de acuerdo al crecimiento de tamaño del software. (2.94).

B : Constante denominada Factor escalar y su valor está dado por la resultante de los aspectos positivos sobre los negativos que presenta el proyecto.

$$B = 0.91 + 0.01 x (W_i)$$

$$B = 0.91 + 0.01 x 10.71$$

$$B = 1.0171$$

Donde:

W_i : Variables escalares que indican las características que el proyecto presenta en lo que a su complejidad y entorno de desarrollo se refiere.

4.4.2.1 Precedentes (PREC).

El factor de precedencia (PREC) toma en cuenta el grado de experiencia previa en relación al producto a desarrollar, tanto en aspectos organizacionales como en el conocimiento del software y hardware a utilizar.

4.4.2.2 Flexibilidad de desarrollo (FLEX).

El factor de flexibilidad (FLEX) considera el nivel de exigencia en el cumplimiento de los requerimientos preestablecidos, plazos de tiempos y especificaciones de interfaz.

4.4.2.3 Cohesión del equipo (TEAM).

El factor de escala denominado Cohesión del Equipo tiene en cuenta las dificultades de sincronización entre los participantes del proyecto: usuarios, clientes y desarrolladores. Estas dificultades pueden surgir por diferencias culturales, dificultad en la conciliación de objetivos, falta de experiencia y familiaridad con el trabajo en equipo.

4.4.2.4 Solución de riesgos (RESL).

Este factor involucra aspectos relacionados al conocimiento de los ítems de riesgo crítico y al modo de abordarlos dentro del proyecto.

4.4.2.5 Madurez del proceso (PMAT).

El procedimiento para determinar el PMAT es establecer el porcentaje de cumplimiento de cada una de las Áreas evaluando el grado de cumplimiento de las metas correspondientes.

La siguiente tabla muestra los valores asignados a cada una de estas variables:

Tabla 37: Factores de escala

Nombre	Valor	Justificación
PREC	1.24	Existen varios sistemas similares a nivel mundial pero no a nivel nacional.
FLEX	1.01	Cuenta con una flexibilidad de desarrollo muy alta.
TEAM	1.10	El equipo de desarrollo presenta una alta cohesión.
RESL	4.24	No se identificaron grandes riesgos.

PMAT	3.12	Se cuenta con una alta experiencia necesaria como para que el software cumpla con las funcionalidades requeridas.
Total(SF)	10.71	

4.5 Ajuste del esfuerzo nominal.

El esfuerzo calculado anteriormente es un valor nominal y debe ser ajustado en base a las características del proyecto para lo cual se tiene un conjunto de Multiplicadores de Esfuerzo (MEi) que representan las características del proyecto y expresan su impacto en el desarrollo total del producto de software.

4.5.1 Clasificados en categorías

El esfuerzo nominal de desarrollo de un proyecto de software se ajusta para una mejor estimación mediante factores que se clasifican en cuatro áreas: Producto, Plataforma, Personal y Proyecto.

Los 7 Multiplicadores de Esfuerzo son:

Del Producto: Se refieren a las restricciones y requerimientos sobre el producto a desarrollar.

RCPX: Confiabilidad y Complejidad del producto

RUSE: Requerimiento de Reusabilidad

De la Plataforma: Estos factores analizan la complejidad de la plataforma subyacente.

PDIF: Dificultad de la Plataforma

Del Personal: Estos factores están referidos al nivel de habilidad que posee el equipo de desarrollo.

PERS: Aptitud del Personal

PREX: Experiencia del Personal

Del Proyecto: Estos factores se refieren a las condiciones y restricciones bajo las cuales se lleva a cabo el proyecto.

FCIL: Facilidades

SCED: Cronograma de Desarrollo Requerido

Tabla 38: Multiplicadores de esfuerzo.

Nombre	Valor	Justificación
RCPX	1.74	La complejidad del sistema es alta así como su confiabilidad.

RUSE	0.95	La parte del código que se pretende reutilizar es muy poca.
PDIF	1.00	Uso de memoria y almacenamiento normal, plataforma estable.
PREX	1.00	La experiencia del personal en cuanto a la utilización del lenguaje y herramientas es intermedia.
PERS	0.63	La aptitud del personal es alta.
FCIL	0.73	Se utilizaron entornos de desarrollo integrados, herramientas de modelación y automatización que facilitan el trabajo.
SCED	1.00	El tiempo estimado para el desarrollo del producto se empleó correctamente.
Total(EM)	0.76	

Se encuentra la tabla de multiplicadores de esfuerzo del modelo del diseño temprano. [Anexo I]

$$PM_{ajustado} = PM_{Nominal} \times (ME_t)$$

$$PM_{ajustado} = 22.05 \text{ meses/hombre} \times 0.76$$

$$PM_{ajustado} = 16.76 \text{ meses/hombre}$$

4.5.2 Calculo del tiempo de desarrollo del software.

El tiempo requerido para el desarrollo del proyecto está dado por la siguiente ecuación:

$$TDEV = C \times (PM_{ajustado})^F$$

$$TDEV = 3.64 \times 16.76^{0.26}$$

$$TDEV = 7.57$$

Donde:

C: Constante con valor 3.64.

PM_{ajustado} = 16.76 meses/hombre

F: 0.26.

Calcular

$$F = D + 0.2 \times 0.01x \quad SF$$

$$F = 0.24 + 0.2 \times 0.01 \times 10.71.$$

$$F = 0.26$$

Donde:

D : Constante cuyo valor es 0.24.

SF : Valor de los factores de escala.

4.5.3 Cálculo del costo total del proyecto.

Para el cálculo del costo total correspondiente al proyecto en cuestión, COCOMO II propone la siguiente ecuación:

$$C = CHM \times PM$$

$$C = 271 \times 22.05$$

$$C = 5975.55$$

CALCULAR

Donde:

C:Costo total.

CHM: Costo teniendo en cuenta salario de todos los obreros, el cual se calcula por la siguiente ecuación.

$$CHM = CH \times sal$$

$$CHM = 2.91 \times 100.$$

$$CHM = 291$$

CALCULAR

Donde:

sal : Salario medio por cada trabajador.

CH : Cantidad de personas destinadas al proyecto, lo que es calculado a través de la ecuación:

$$CH = \frac{PM}{TDEV}$$

$$CH = 22.05 / 7.57$$

$$CH = 2.91$$

4.6 Resultados.

En la siguiente tabla se muestran los resultados obtenidos luego de haber efectuado todos los cálculos para determinar el costo y esfuerzo requeridos por el proyecto.

Tabla39: Resultados.

Cálculo	Valor
Esfuerzo	22.05 meses/hombres
Tiempo de desarrollo	7 meses
Cantidad de hombres	2 hombres
Salario medio	100 pesos
Costo	5975.55

4.7 Análisis de costo.

El desarrollo de un producto siempre presenta un costo de producción, el cual se justifica en base a los beneficios reportados por el mismo. El sistema presentado no precisa de grandes gastos, debido en gran medida a la utilización de plataformas, tecnologías y herramientas libres que no requieren del pago de licencia por su uso; su costo está fundamentalmente determinado por el salario devengado por los desarrolladores que toman parte en el sistema.



Supervisión de agentes en tiempo real del Módulo Call-Center de Elastix

4.8 Conclusiones.

En este capítulo se realizó un análisis de factibilidad de la solución propuesta, obteniendo los estimados de tamaño, esfuerzo y planificación. Se llegó a la conclusión de que el desarrollo del sistema propuesto es viable, basándonos en la comparación de los costos de producción con los beneficios que reportará el sistema.



Supervisión de agentes en tiempo real del Módulo Call-Center de Elastix

CONCLUSIONES GENERALES

Con el desarrollo de este trabajo se ha obtenido la personalización de un subsistema de supervisión de agentes en tiempo real. Se puede decir que se ha cumplido satisfactoriamente el objetivo trazado para este trabajo ya que el producto resultante está provisto de un ambiente cómodo, fácil de entender, que cumple con las especificaciones del cliente. Además, el estudio de la factibilidad determinó que el desarrollo de la aplicación es realmente factible para su puesta en funcionamiento. Por otra parte, se aplicó la metodología XP para guiar el proceso de desarrollo del software permitiendo la modelación del sistema en el tiempo estimado, posibilitando así las entregas en tiempo al cliente.

RECOMENDACIONES

Se recomienda:

1. Que la aplicación se utilice en otros marcos de trabajo y que se pueda integrar a otras aplicaciones como un subsistema más.
2. Por las posibilidades de creación de módulos de terceros que posee la aplicación se recomienda agregarle un subsistema que monitoree el trabajo de los supervisores, logrando que el módulo sea lo más eficiente posible.
3. Que se le agreguen otras funcionalidades de apoyo a la supervisión de los agentes que permitan enriquecer la información de los mismos.
4. Probar el subsistema de supervisión en un marco real antes de ser sometido a un posterior despliegue.

REFERENCIA BIBLIOGRÁFICA

1. Andicel. [En línea] [Citado el: 20 de Enero de 2011.] Disponible en:[[http://www.andicel.com.co\(06-01-2011\)](http://www.andicel.com.co(06-01-2011))].
2. Eflo.net. [En línea] [Citado el: 16 de Noviembre de 2010.] Disponible en:[<http://www.eflo.net/vicidial.php>].
3. Teleprom. [En línea] [Citado el: 18 de Noviembre de 2010.] Disponible en:[<http://www.teleprom.com>].
4. Nextortelecom. [En línea] [Citado el: 19 de Noviembre de 2010.] Disponible en:[http://www.nextortelecom.com/nextor_voip/joomla-overview/call-centers].
5. Virtua. [En línea] [Citado el: 10 de Enero de 2011.] Disponible en:[<http://www.virtual.unal.edu.co/cursos/sedes/manizales/4100010/Lecciones/Cap5/HtasCASE.htm>].
6. Cnx. [En línea] [Citado el: 15 de Diciembre de 2010.] Disponible en:[<http://cnx.org/content/m17543/latest/>].
7. Ciberneta. [En línea] [Citado el: 12 de Diciembre de 2010.] Disponible en:[http://www.ciberneta.com/manuales/instalacion_servidor_web/1_conceptos_basicos.php].
8. Desarrolloweb. [En línea] [Citado el: 5 de Enero de 2011.] Disponible en:[<http://www.desarrolloweb.com/articulos/332.php>].
9. Webtutoriales. [En línea] [Citado el: 10 de Febrero de 2011.] Disponible en:[<http://www.webtutoriales.com/tutoriales/programacion/modelo-vista-controlador.54.html>].
10. Programacionweb.net. [En línea] [Citado el: 12 de Enero de 2011.] Disponible en:[<http://www.programacionweb.net/articulos/articulo/?num=505>].
11. **Landívar, Edgar.** *COMUNICACIONES UNIFICADAS CON ELASTIX VOLUMEN 2* . 2009.
12. **Martínez, María del Pilar García, Daniel Crespo Vázquez.** *Implementación de un módulo para el envío de mensajes a celulares con el componente SAM*. [Documento] Habana : Universidad de las Ciencias Informáticas, 2010.

BIBLIOGRAFÍA

1. Proactiva-calidad. [En línea] [Citado el: 20 de febrero de 2011.] Disponible en:[<http://www.proactiva-calidad.com/java/patrones/mvc.html>].
2. Neo framework. [En línea] [Citado el: 17 de Diciembre de 2010.] Disponible en:[<http://www.neoframework.org>].
3. **Grau Abalo, Ricardo, Correa Valdés, Cecilia y Rojas Betancur, Mauricio.** Metodología de la Investigación. [Documento]
4. **Escribano, Gerardo Fernández.** Introducción a Extreme Programming . [Documento]
5. PHP. [En línea] [Citado el: 16 de Diciembre de 2010.] Disponible en:[<http://www.php.net/manual/es/index.php>].
6. [En línea] [Citado el: 29 de Enero de 2011.] Disponible en:[<http://www.elastix.org/es.html>].
7. Itrainonline. [En línea] [Citado el: 20 de Febrero de 2011.] Disponible en:[[http://www.itrainonline.org/itrainonline/spanish/detail1.shtml?cmd\[893\]=x-893-5039497&cmd\[996\]=x-996-5039497](http://www.itrainonline.org/itrainonline/spanish/detail1.shtml?cmd[893]=x-893-5039497&cmd[996]=x-996-5039497)].
8. La Cara Oscura. [En línea] [Citado el: 2 de Marzo de 2011.] Disponible en:[<http://www.lacaraoscura.com/2006/06/07/patrones-de-diseno-para-la-web/>].
9. **Canós, José H., Letelier, Patricio y Penadés, M^a Carmen.** Metodologías Ágiles en el Desarrollo de Software. [Documento] Valencia : Universidad Politécnica de Valencia.
10. **Larman, Craig.** UML y Patrones. Introducción al análisis y diseño orientado a objetos Pearson Educación. [Documento]
11. CAVSI. [En línea] [Citado el: 6 de Enero de 2011.] Disponible en:[<http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>].
12. Slideshare. [En línea] [Citado el: 21 de Enero de 2011.] Disponible en:[<http://www.slideshare.net/alexmerono/sistemas-gestores-de-bases-de-datos>].
13. Todo Dreamweaver. [En línea] [Citado el: 12 de Enero de 2011.] Disponible en:[<http://www.todo-dreamweaver.com/>].
14. Tecnología Diaria. [En línea] [Citado el: 3 de Abril de 2011.] Disponible en:[<http://www.tecnologiadiaria.com/2009/08/codigos-de-inyeccion-sql.html>].
15. Maestros del Web. [En línea] [Citado el: 19 de Marzo de 2011.] Disponible en:[<http://www.maestrosdelweb.com/editorial/inyecsql/>].

16. Antivirus Online.es. [En línea] [Citado el: 20 de Mayo de 2011.] Disponible en:[<http://www.antivirusonline.es/index.php/tipos-de-ataques/inyeccion-sql>].
17. Wigahluk. [En línea] [Citado el: 12 de Abril de 2011.] Disponible en:[<http://wigahluk.wordpress.com/category/software-development/pruebas-unitarias/>].
18. ARCO. [En línea] [Citado el: 11 de Marzo de 2011.] Disponible en:[http://arco.esi.uclm.es/~david.villa/pensar_en_C++/vol2/ch03s02.html].
19. Buenas Tareas. [En línea] [Citado el: 24 de Abril de 2011.] Disponible en:[<http://www.buenastareas.com/temas/pruebas-de-aceptaci%C3%B3n/20>].
20. Scribd. [En línea] [Citado el: 30 de Marzo de 2011.] Disponible en:[<http://es.scribd.com/doc/5004292/comparacion-entre-RUP-y-XP>].
21. RUP vs XP. [Documento] 2001.
22. **Arañó Garcés, Eduardo y Marrero Echemendía, Jessica.** Interfaz para acceso al servidor de correos Zimbra mediante dispositivos móviles. [Documento] Ciudad de la Habana : s.n., 2010.

ANEXO I TABLAS DEL COCOMO II

Peso de Factor de Complejidad.

Tipo de función	Peso del Factor de Complejidad		
	Bajo	Promedio	Alto
Entradas Externas (Inputs)	3	4	6
Salidas Externas (Outputs)	4	5	7
Archivo Lógicos Internos (Archivos)	7	10	15
Archivos Externos de Interfase (Interfases)	5	7	10
Consultas Externas (Queries)	3	4	6

Factores de Escala

Factor de Escala Wj	Muy Bajo	Bajo	Normal	Alto	Muy Alto	Extra
Precedencia PREC	Completamente sin precedentes	Ampliamente sin precedentes	Algún precedente	Generalmente Familiar	Ampliamente Familiar	Completamente Familiar
Flexibilidad en el desarrollo FLEX	Rigurosa	Relajación Ocasional	Alguna Relajación	Conformidad en general	Alguna Conformidad	Metas generales
Arquitectura/ Resolución de riesgo RESL	Poca (20%)	Alguna (40%)	Siempre (60%)	Generalmente 75%)	Principalmente (90%)	Completo (100%)
Cohesión de equipo TEAM	Interacciones difíciles	Interacciones con alguna dificultad	Interacciones básicamente cooperativas	Ampliamente Cooperativas	Altamente Cooperativas	Interacciones Sin Fisuras
Madurez del proceso PMAT	Desarrollado más adelante					

Factores de Escala

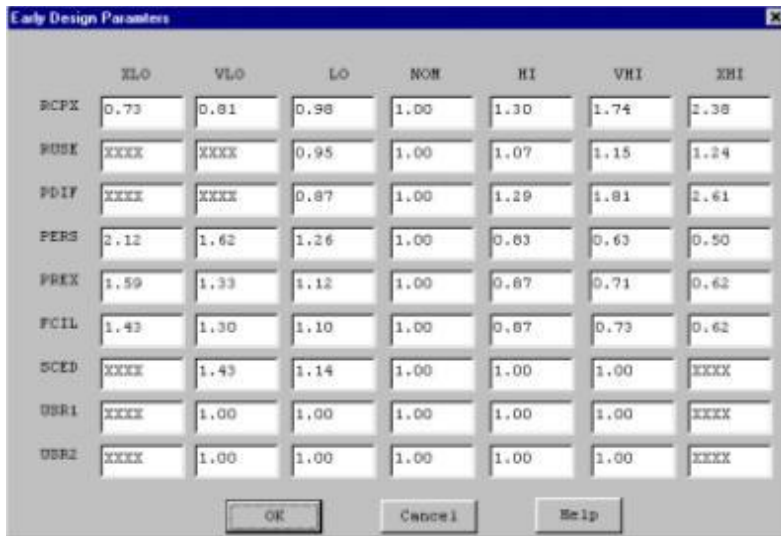


	VLO	LO	NOM	NI	VHI	XHI
PREC	6.20	4.96	3.72	2.48	1.24	0.00
FLEX	5.07	4.05	3.04	2.03	1.01	0.00
RESL	7.07	5.65	4.24	2.83	1.41	0.00
TEAM	5.48	4.38	3.29	2.19	1.10	0.00
PMAT	7.00	6.24	4.68	3.12	1.56	0.00

Factor PMAT de acuerdo al nivel de CMM

Nivel de CMM	PMAT
1 – Mitad inferior	Muy Bajo
1 – Mitad superior	Bajo
2	Nominal
3	Alto
4	Muy Alto
5	Extra Alto

Multiplicadores de Esfuerzo



	XLO	VLO	LO	NOM	HI	VHI	XHI
RCPX	0.73	0.81	0.98	1.00	1.30	1.74	2.38
RUSE	XXXX	XXXX	0.95	1.00	1.07	1.15	1.24
PDIF	XXXX	XXXX	0.87	1.00	1.29	1.81	2.61
PERS	2.12	1.62	1.26	1.00	0.83	0.63	0.50
PREX	1.59	1.33	1.12	1.00	0.87	0.71	0.62
PCIL	1.43	1.30	1.10	1.00	0.87	0.73	0.62
SCED	XXXX	1.43	1.14	1.00	1.00	1.00	XXXX
USR1	XXXX	1.00	1.00	1.00	1.00	1.00	XXXX
USR2	XXXX	1.00	1.00	1.00	1.00	1.00	XXXX

ANEXO II CASOS DE PRUEBA

Prueba #1.1 Insertar supervisor.

Caso de Prueba Aceptación	
Código: 1.1	Número Historia de Usuario: 1
Nombre: Insertar supervisor.	
Descripción: Prueba para verificar la funcionalidad Insertar supervisor.	
Condiciones de Ejecución: Solo el administrador podrá insertar los supervisores.	
Entrada/Pasos de ejecución: Introducir los datos personales del supervisor (usuario, contraseña y verificación de la misma de forma correcta) y asignarle los agentes que el mismo supervisará.	
Resultado Esperado: Muestra el mensaje Se insertó el supervisor.	
Evaluación de la Prueba: Prueba satisfactoria.	

Caso de Prueba Aceptación	
Código: 1.1	Número Historia de Usuario: 1
Nombre: Insertar supervisor.	
Descripción: Prueba para verificar la funcionalidad Insertar supervisor.	
Condiciones de Ejecución: Solo el administrador podrá insertar los supervisores.	
Entrada/Pasos de ejecución: Introducir los datos personales del supervisor (solo usuario)	
Resultado Esperado: Se muestra un mensaje de alerta Se debe poner la contraseña.	
Evaluación de la Prueba: Prueba satisfactoria.	

Caso de Prueba Aceptación

Código: 1.1	Número Historia de Usuario: 1
Nombre: Insertar supervisor.	
Descripción: Prueba para verificar la funcionalidad Insertar supervisor.	
Condiciones de Ejecución: Solo el administrador podrá insertar los supervisores.	
Entrada/Pasos de ejecución: Introducir los datos personales del supervisor (usuario y la contraseña sin llenar el campo de verificación de la misma) y asignarle los agentes que el mismo supervisará.	
Resultado Esperado: Se muestra un mensaje de alerta Las contraseñas no coinciden.	
Evaluación de la Prueba: Prueba satisfactoria.	

Caso de Prueba Aceptación	
Código: 1.1	Número Historia de Usuario: 1
Nombre: Insertar supervisor.	
Descripción: Prueba para verificar la funcionalidad Insertar supervisor.	
Condiciones de Ejecución: Solo el administrador podrá insertar los supervisores.	
Entrada/Pasos de ejecución: Introducir los datos personales del supervisor (usuario, contraseña y verificación de la misma de forma incorrecta) y asignarle los agentes que el mismo supervisará.	
Resultado Esperado: Se muestra un mensaje de alerta Las contraseñas no coinciden.	
Evaluación de la Prueba: Prueba satisfactoria.	

Caso de Prueba Aceptación	
Código: 1.1	Número Historia de Usuario: 1
Nombre: Insertar supervisor.	

Descripción: Prueba para verificar la funcionalidad Insertar supervisor.
Condiciones de Ejecución: Solo el administrador podrá insertar los supervisores.
Entrada/Pasos de ejecución: Introducir los datos personales del supervisor (contraseña y verificación de la misma de forma correcta, obviando el usuario) y asignarle los agentes que el mismo supervisará.
Resultado Esperado: Se muestra un mensaje de alerta Debe poner un nombre de usuario.
Evaluación de la Prueba: Prueba satisfactoria.

Caso de Prueba Aceptación	
Código: 1.1	Número Historia de Usuario: 1
Nombre: Insertar supervisor.	
Descripción: Prueba para verificar la funcionalidad Insertar supervisor.	
Condiciones de Ejecución: Solo el administrador podrá insertar los supervisores.	
Entrada/Pasos de ejecución: Introducir los datos personales del supervisor (usuario, contraseña y verificación de la misma de forma correcta) sin asignarle los agentes que el mismo supervisará.	
Resultado Esperado: Se muestra un mensaje de alerta Debe asignar al menos un agente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #2.1 Modificar supervisor.

Caso de Prueba Aceptación	
Código: 2.1	Número Historia de Usuario: 2
Nombre: Modificar supervisor	
Descripción: Prueba para verificar la funcionalidad Modificar supervisor.	
Condiciones de Ejecución: Solo el administrador podrá modificar los campos personales de cada	

supervisor.
Entrada/Pasos de ejecución: Editar el o los campos a modificar.
Resultado Esperado: Se muestra un mensaje de alerta Supervisor modificado.
Evaluación de la Prueba: Prueba satisfactoria.

Prueba #3.1 Eliminar supervisor

Caso de Prueba Aceptación	
Código: 3.1	Número Historia de Usuario: 3
Nombre: Eliminar supervisor.	
Descripción: Prueba para verificar la funcionalidad de Eliminar supervisor.	
Condiciones de Ejecución: Solo se podrán eliminar aquellos supervisores que hayan sido insertados correctamente y únicamente esta acción la lleva a cabo el administrador del sistema.	
Entrada/Pasos de ejecución: Se procede a eliminar el supervisor.	
Resultado Esperado: Se muestra un mensaje de alerta Supervisor eliminado.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #4.1 Autenticar supervisor.

Caso de Prueba Aceptación	
Código: 4.1	Número Historia de Usuario: 4
Nombre: Autenticar supervisor.	
Descripción: Prueba para verificar la funcionalidad de Autenticar supervisor.	
Condiciones de Ejecución: El supervisor debe estar previamente registrado y activo en el sistema.	
Entrada/Pasos de ejecución: Se procede a autenticar el supervisor introduciendo los datos correspondientes (usuario y contraseña correctamente).	
Resultado Esperado: El supervisor se autenticó correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	



Supervisión de agentes en tiempo real del Módulo Call-Center de Elastix

Caso de Prueba Aceptación	
Código: 4.1	Número Historia de Usuario: 4
Nombre: Autenticar supervisor.	
Descripción: Prueba para verificar la funcionalidad de Autenticar supervisor.	
Condiciones de Ejecución: El supervisor debe estar previamente registrado y activo en el sistema.	
Entrada/Pasos de ejecución: Se procede a autenticar el supervisor introduciendo los datos correspondientes (usuario solamente).	
Resultado Esperado: Se muestra un mensaje de alerta El campo password debe ser llenado.	
Evaluación de la Prueba: Prueba satisfactoria.	

Caso de Prueba Aceptación	
Código: 4.1	Número Historia de Usuario: 4
Nombre: Autenticar supervisor.	
Descripción: Prueba para verificar la funcionalidad de Autenticar supervisor.	
Condiciones de Ejecución: El supervisor debe estar previamente registrado y activo en el sistema.	
Entrada/Pasos de ejecución: Se procede a autenticar el supervisor introduciendo los datos correspondientes (usuario correctamente y contraseña incorrectamente).	
Resultado Esperado: Se muestra un mensaje de alerta La contraseña proporcionada es incorrecta.	
Evaluación de la Prueba: Prueba satisfactoria.	

Caso de Prueba Aceptación	
Código: 4.1	Número Historia de Usuario: 4
Nombre: Autenticar supervisor.	

Descripción: Prueba para verificar la funcionalidad de Autenticar supervisor.
Condiciones de Ejecución: El supervisor debe estar previamente registrado y activo en el sistema.
Entrada/Pasos de ejecución: Se procede a autenticar el supervisor introduciendo los datos correspondientes (solo la contraseña).
Resultado Esperado: Se muestra un mensaje de alerta El campo nombre debe ser llenado.
Evaluación de la Prueba: Prueba satisfactoria.

Prueba #5.1 Supervisar agentes en tiempo real

Caso de Prueba Aceptación	
Código: 5.1	Número Historia de Usuario: 5
Nombre: Supervisar agentes en tiempo real	
Descripción: Prueba para verificar la funcionalidad de Supervisar agentes en tiempo real.	
Condiciones de Ejecución: El Estado del Dialer tiene que estar en ejecución, el supervisor debe estar autenticado en el sistema así como tener asignada una extensión que esté libre.	
Entrada/Pasos de ejecución: Se procede a mostrar todos los estados de los agentes a los que se está supervisando.	
Resultado Esperado: Los datos mostrados fueron correctos.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #6.1 Intervenir llamadas.

Caso de Prueba Aceptación	
Código: 6.1	Número Historia de Usuario: 6
Nombre: Intervenir llamadas.	
Descripción: Prueba para verificar la funcionalidad de Intervenir llamadas.	

Condiciones de Ejecución: El supervisor debe estar autenticado en el sistema y tener a dicho agente dentro de los que le fue asignado previamente por el administrador así como tener asignada una extensión que esté libre y que el estado del Dialer tiene que estar en ejecución.
Entrada/Pasos de ejecución: Se procede a intervenir la llamada.
Resultado Esperado: Se escucha y se interviene el canal correctamente.
Evaluación de la Prueba: Prueba satisfactoria.

Prueba #7.1 Colgar llamada.

Caso de Prueba Aceptación	
Código: 7.1	Número Historia de Usuario: 7
Nombre: Colgar llamada.	
Descripción: Prueba para verificar la funcionalidad de Colgar llamada.	
Condiciones de Ejecución: El supervisor debe estar autenticado en el sistema y tener al agente que le fue previamente asignado atendiendo a una llamada así como el estado del Dialer tiene que estar en ejecución.	
Entrada/Pasos de ejecución: Se procede a colgar la llamada.	
Resultado Esperado: Se colgó la llamada correctamente.	
Evaluación de la Prueba: Prueba satisfactoria	

Prueba #8.1 Desconectar agente.

Caso de Prueba Aceptación	
Código: 8.1	Número Historia de Usuario: 8
Nombre: Desconectar agente.	
Descripción: Prueba para verificar la funcionalidad de Desconectar agente.	

Condiciones de Ejecución: El agente debe estar conectado y el supervisor debe estar autenticado en el sistema y tener a dicho agente dentro de los que le fue asignado previamente por el administrador.
Entrada/Pasos de ejecución: Se procede a desconectar al agente
Resultado Esperado: Se desconectó el agente correctamente.
Evaluación de la Prueba: Prueba satisfactoria

Prueba #9.1 Transferir llamada.

Caso de Prueba Aceptación	
Código: 9.1	Número Historia de Usuario: 9
Nombre: Transferir llamada.	
Descripción: Prueba para verificar la funcionalidad de Transferir llamada.	
Condiciones de Ejecución: El supervisor debe estar autenticado en el sistema y tener a dicho agente dentro de los que le fue asignado previamente por el administrador así como tener asignada una extensión que esté libre y el estado del Dialer tiene que estar en ejecución.	
Entrada/Pasos de ejecución: Se procede a transferir la llamada.	
Resultado Esperado: Se transfirió correctamente la llamada.	
Evaluación de la Prueba: Prueba satisfactoria.	

GLOSARIO DE TÉRMINOS

Abonado: son personas contactadas a través de llamadas telefónicas.

Agente: son operadores contratados por el administrador de la central telefónica.

Asterisk: es un software completo en PBX (Private Branch Exchange o Centralita Telefónica Privadas) que usa el concepto de software libre, se dice que provee todas las características que se esperan de una planta telefónica y mucho más. Es promovido por la empresa Digium que se encarga entre otros aspectos del desenvolvimiento de código fuente y en hardware de telefonía de bajo costo que funciona con Asterisk. Entre los servicios que proporciona Asterisk se encuentra: voicemail con directorios, conferencias, respuesta de voz interactiva, llamada en espera y grabado de llamadas.

Call-Center: Es una unidad funcional dentro de la empresa (o bien una empresa en si misma) diseñada para manejar grandes volúmenes de llamadas telefónicas entrantes y salientes desde y hacia sus clientes, con el propósito de dar soporte a las operaciones cotidianas de la entidad.

Extensión: son las que permiten a los clientes llamar directamente desde su anuncio con mayor facilidad.

Fichero: es un sistema real o virtual de organización de la información mediante una clasificación determinada.

Intervenir: Controlar la comunicación privada entre un abonado y un agente.

PBX: son las siglas en inglés de (Private Branch Exchange), la cual es la red telefónica privada que es utilizada dentro de una empresa. Los usuarios del sistema telefónico PBX comparten un número definido de líneas telefónicas para poder realizar llamadas externas. El sistema PBX conecta las extensiones internas dentro de una empresa y al mismo tiempo las conecta con la red pública.

Redireccionar: es la posibilidad que tiene el supervisor de acceder a una determinada dirección mediante otra, en este caso específicamente se esta está haciendo referencia a un agente.



Supervisión de agentes en tiempo real del Módulo Call-Center de Elastix

SGBD: sistema gestor de base de datos es un conjunto de programas que permite crear y mantener una base de datos, asegurando su integridad, confidencialidad y seguridad.

Supervisar: Inspeccionar por parte de un supervisor el trabajo realizado los agentes que interactúan con el software, para comprobar una acción determinada.

Supervisor: persona especializada que tiene como fin utilizar proporcionadamente los factores que hacen posible la ejecución de los procesos de trabajo, elemento clave dentro del software del que depende el buen desempeño del mismo, se encarga de supervisar los agentes de la aplicación.

Telefonía IP: reúne la transmisión de voz y de datos, lo que posibilita la utilización de las redes informáticas para efectuar llamadas telefónicas. Surge como una alternativa a la telefonía tradicional, brindando nuevos servicios al cliente y una serie de beneficios económicos y tecnológicos.

Transferir: este servicio permite que el supervisor pueda transferir una llamada que tenga en la línea un abonado hacia otro número.

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