

**Universidad de las Ciencias Informáticas**

**Facultad 2**



**Trabajo de Diploma para optar por el título de  
Ingeniero Informático.**

**Título:**

**Sistema para la automatización  
del taller de reparaciones de la UCI.**

**Autor(es):**

**Daydier Bernal Calvo  
Carlandis Chaveco Martínez**

**Tutor:**

**Ing. Deivis R. Álvarez Mendoza**

**Co-tutor:**

**Ing. Aliennis Mercedes González Hurtado**

**Junio de 2008. Ciudad de La Habana.**

**Año del 50 Aniversario del Triunfo de la Revolución**

*“Por el poder de erguirse se mide a los hombres”*

*José Martí*

---

## DECLARACIÓN DE AUTORÍA

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

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

**Daydier Bernal Calvo**

\_\_\_\_\_

Firma del Autor

**Deivis R. Álvarez Mendoza**

\_\_\_\_\_

Firma del Tutor

**Carlandis Chaveco Martínez**

\_\_\_\_\_

Firma del Autor

---

## DATOS DE CONTACTO

Tutor: Ing. Deivis R. Álvarez Mendoza.

Instructor Recién Graduado de la Universidad de las Ciencias Informáticas (UCI). Un año de experiencia. Líder del proyecto perteneciente al Grupo de Asistencia Técnica (GAT) de la Facultad 2 de la UCI.

Co-Tutor: Ing. Aliennis Mercedes González Hurtado.

Instructor Recién Graduado de la Universidad de las Ciencias Informáticas (UCI). Un año de experiencia.

## AGRADECIMIENTOS

### *Carlandis:*

Agradezco a mi hermano, mis padres, mi esposa su familia, por darme la estabilidad emocional, económica, sentimental, para poder llegar hasta este logro, que definitivamente no hubiese podido ser realidad sin ustedes.

A todos mis amigos pasados y presentes, pasados por ayudarme a crecer y madurar como persona y presentes por estar siempre conmigo apoyándome en todas las circunstancias posibles, también son parte de esta alegría, los quiero.

A mis tutores por ayudarme y apoyarme en este ultimo escalón para poder alcanzar este sueño.

Y a todos aquellos, que han quedado en los recintos más escondidos de mi memoria, pero que fueron partícipes de este triunfo.

### *Daydier:*

Agradezco a mi papá por siempre estar apoyándome en todas las actividades, a mis hermanas y hermanos, a mis sobrinos, a mis amigos y amigas de esta universidad que son como mi familia, en especial a los hermanos que comenzamos juntos esta carrera en el grupo 2102, a los que continúan y a los que ya no están en la UCI. También a los del grupo 1 que me recibieron como uno más de los suyos. A mi profesor y hermano Aimel que siempre me ha apoyado en todas mis tareas de la universidad. A mis amigos del barrio que son como mi segunda familia por haber estado pendiente de mi desempeño en la carrera y apoyarme siempre.

Agradezco a los compañeros del proyecto del GAT que tanto apoyo nos dieron, a Landy, y a nuestro tutor que ha sido el pilar fundamental en el desarrollo de este trabajo.

## DEDICATORIA

*Carlandis:*

*A mi hijo, por ser la fuente de mi inspiración y motivación.*

*A mi mamá y mi papá por ser los mejores y estar conmigo incondicionalmente, gracias por su amor, cariño, apoyo, confianza y educación.*

*A mi hermano por ser mi ejemplo a seguir.*

*A mi esposa por su paciencia, comprensión y amor.*

*A toda mi familia abuelos abuelas, tíos, tías, primos, primas.*

*A todas aquellas personas, colegas y amigos que me brindaron su apoyo, tiempo y siempre tuvieron confianza en mí.*

*Daydier:*

*A mi mamá, mi ángel guardián que nunca me abandona, por su amor de siempre.*

*A mi papá, por su sacrificio y amor de todos estos años.*

*A mis hermanos y amigos, por siempre estar a mi lado en cada momento.*

*A mi gente del barrio de Centro Habana.*

## RESUMEN

El taller de reparaciones del Grupo de Asistencia Técnica de la UCI es el local encargado de dar solución a cada deficiencia o rotura de los medios tecnológicos que se le asignen. Cada medio defectuoso llega al taller con un reporte asignado que se debe actualizar una vez que se repare el equipo y las distintas reparaciones hechas deben ser almacenadas en una base de conocimiento que permita a los trabajadores consultar los datos registrados en la misma según el tipo de equipo que necesite consultar, además se deben generar estadísticas que muestren los datos de equipos atendidos en el taller en un tiempo determinado. Con el estudio de las diversas herramientas, tecnologías, patrones de arquitectura y diseño que organizan el desarrollo del sistema y haciendo uso de un lenguaje de programación y un gestor de base de datos que cumplen con las facilidades que en la actualidad brinda el uso del software libre, se puede desarrollar una aplicación Web<sup>1</sup> que garantice a los trabajadores del taller de reparaciones la correcta realización de las tareas que se desarrollan en su entorno de trabajo. El uso de una metodología para la ingeniería y gestión de software permite además que cada proceso realizado en la elaboración de este sistema concluya con una aplicación que cumpla con cada funcionalidad que se necesita para la automatización del trabajo en el taller de reparaciones de la UCI.

## PALABRAS CLAVE

Grupo de Asistencia Técnica (GAT), Taller de Reparaciones, Reportes, Base de Conocimientos, Estadísticas.

---

<sup>1</sup> Ver Glosario de Términos.

## ÍNDICE

AGRADECIMIENTOS .....	I
DEDICATORIA .....	II
RESUMEN.....	I
INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA .....	5
1.1. Introducción.....	5
1.2. Sobre los talleres de reparación. ....	5
1.3. Uso de software libre para el desarrollo de este sistema. ....	8
1.4. Sobre aplicaciones Web. ....	10
1.5. Metodología propuesta para el desarrollo del software en este sistema: RUP. ....	13
1.6. Lenguaje de programación Web de lado del servidor propuesto para implementar el sistema: PHP Versión 5.0. ....	14
1.7. Programación del lado del cliente. ....	17
1.8. Uso de Ajax. ....	18
1.8. Framework propuesto para el desarrollo de la aplicación. ....	18
1.9. Lenguaje de modelado para los procesos de ingeniería de software. UML. ....	20
1.11. Servidor Web Apache.....	22
1.12. Herramienta propuesta para la modelación de los diagramas de ingeniería de software para el desarrollo este sistema: Visual Paradigm. ....	23
1.13. Uso de Macromedia Dreamweaver 8.0 y Zend Development Environment. ....	24
1.14. Modelo-Vista-Controlador (MVC) .....	25
1.15. Arquitectura Cliente-Servidor .....	26
1.16. Patrones de Diseño empleados.....	27
1.17. Conclusiones. ....	28
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA .....	29
2.1. Introducción.....	29
2.2. Objeto de Estudio. ....	29
2.3. Objeto de automatización. ....	30
2.4. Información que se maneja. ....	31
2.5. Propuesta de sistema. ....	31
2.6. Modelo de Negocio. ....	33
2.6.1. Descripción del Negocio. ....	33
2.6.2. Actores y Trabajadores del Negocio.....	33
2.6.3. Diagrama de Casos de Uso del Negocio. ....	35
2.6.4. Descripción del Caso de Uso del Negocio y Reglas del Negocio. Ver Anexo 1. ....	35
2.6.5. Diagrama de Actividades. Ver Anexo 2.....	35
2.6.5. Modelo de Objetos. Ver Anexo 3.....	35
2.7. Especificación de los Requisitos de Software. ....	35
2.8. Definición de los Casos de Uso del Sistema. ....	37



2.8.1. Actores del sistema. ....	37
2.8.2. Diagrama de Casos de Uso del Sistema. ....	38
2.8.3. Descripción de los Casos de Uso del Sistema. Ver Anexo 4. ....	38
2.9. Conclusiones. ....	39
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA .....	40
3.1. Introducción. ....	40
3.2. Análisis. ....	40
3.3 Diseño. ....	43
3.3.1 Diagramas de interacción. Ver anexo 5. ....	44
3.3.2 Diagrama de clases del diseño. ....	44
3.4. Diseño de la base de datos. ....	49
3.4.1. Modelo lógico de datos (diagrama de clases persistentes). ....	49
3.4.2. Modelo físico de datos (modelo de datos). ....	50
3.5. Conclusiones. ....	52
CAPÍTULO 4: IMPLEMENTACIÓN .....	53
4.1. Introducción. ....	53
4.2 Diagrama de despliegue. ....	53
4.3. Diagrama de componentes. ....	54
4.4. Conclusiones. ....	58
CAPITULO 5. ESTUDIO DE FACTIBILIDAD. ....	59
5.1 Introducción .....	59
5.2 Método de estimación basado en casos de usos. ....	59
5.3. Beneficios Tangibles e Intangibles. ....	63
5.4. Análisis de costos y beneficios. ....	63
5.5. Conclusiones. ....	64
CONCLUSIONES.....	65
RECOMENDACIONES .....	67
BIBLIOGRAFÍA.....	68
ANEXOS.....	71
GLOSARIO DE TÉRMINOS.....	93

### INTRODUCCIÓN

La vida interna de la UCI, teniendo en cuenta las características del proyecto y su constante desarrollo, está estrechamente vinculada a la tecnología. Cada medio tecnológico es de vital importancia en el desarrollo de la informatización de las tareas de la universidad, ya sea en la residencia, el aula, el laboratorio, incluso en el comedor. En la UCI cada proceso de informatización depende de medios tecnológicos que garantizan la correcta realización de cada actividad. Para mantener en forma la tecnología con que dispone la universidad no solo existen las reglas disciplinarias que sancionan a quienes maltratan la propiedad estatal, sino que existe un taller de reparaciones que se encarga de corregir los defectos y roturas de cada medio, siempre que esté a su alcance.

El taller de reparaciones de medios tecnológicos del Grupo de Asistencia Técnica de la UCI brinda los servicios de reparación de medios tecnológicos o defectuosos pertenecientes a la comunidad universitaria.

El servicio de este taller es de vital importancia para la universidad pues es una prioridad mantener funcionando las computadoras con sus dispositivos que posibilitan el desarrollo de la docencia y la producción en el centro; y no solo se reparan dispositivos de computadoras, también se hacen reparaciones de televisores, teléfonos y otros medios electrónicos de los que dispone la universidad. Este taller brinda servicios las 24 horas del día.

En este taller se reparan los equipos rotos que se reportan desde cualquier punto de la universidad, desde la residencia, la zona administrativa, los docentes y otras áreas. Sus técnicos mantienen un control detallado de cada rotura, sus horarios de entrada y salida, la norma cumplida o a cumplir. En caso de no poder darle solución al problema pasan el mismo a la empresa COPEXTEL<sup>2</sup> que contiene los materiales un poco más avanzados para resolver roturas de un nivel de mayor complejidad.

Con el taller de reparaciones del GAT se puede comunicar vía telefónica o ir personalmente, allí se recogen los datos del equipo defectuoso así como del lugar de donde se efectúa la llamada y los datos de la persona que reporta. Luego el reporte para a los técnicos que se dividen en brigadas de trabajo y

---

<sup>2</sup> Empresa cubana comercializadora de equipos electrónicos y tecnologías.

salen a cumplir con la tarea asignada. Una vez solucionado el problema, se le da de baja al equipo. En caso de no poder resolverse, como decíamos anteriormente, se le asigna la incidencia a COPEXTEL.

Se está utilizando hasta la fecha una aplicación propia del Grupo de Asistencia Técnica, a la que se puede acceder a través de la dirección electrónica <http://gatserver.uci.cu>, que ejecuta las acciones que los trabajadores del taller necesitan para mantener actualizado su entorno laboral y organizar sus propias tareas y gestionar los reportes solicitados. Pero este sistema no presenta un registro de cada incidencia atendida en el taller y que además almacene las soluciones dadas a cada deficiencia según el medio tecnológico reparado y que posibilite al trabajador del taller realizar una búsqueda avanzada de cada incidencia y conocer sus posibles soluciones para así facilitar su labor. Esta aplicación además está desarrollada sobre una plataforma de software propietario y no se adapta a las nuevas herramientas y tecnologías de software libre que se vienen desarrollando de forma acelerada en la universidad. Actualmente los trabajadores del taller no disponen de un registro en formato digital que almacene cada incidencia por equipo y las soluciones dadas, esta información se almacena en papel.

El taller de reparaciones necesita agregar nuevas funcionalidades en su aplicación y modificar otras como es la obtención de estadísticas que se hacen de forma estática y darle más dinamismo a las mismas y obtener gráficas que hagan más novedosa e interesante la búsqueda de datos dentro del sistema.

Por el entorno de trabajo del taller de reparaciones de la UCI, y atendiendo a las necesidades del cliente, además de tener en cuenta los aspectos señalados anteriormente, se hace necesario la creación de un sistema informático para así cumplir con los objetivos señalados. Se tiene como *problema científico*: ¿Cómo mejorar los procesos de gestión de medios tecnológicos dentro del taller de reparaciones del GAT?

Se tiene como *objeto de estudio* los procesos de gestión de medios tecnológicos en talleres de reparación.

Como *campo de acción* tenemos el trabajo de servicios técnicos que se realiza en el taller de reparaciones de la UCI.

Como *objetivo general* se tiene desarrollar un sistema informático para facilitar la gestión y control de los equipos en reparación en el taller de la UCI.

Se cuenta además con los siguientes *objetivos específicos*:

- Dar a conocer las características y ventajas de aplicaciones Web.
- Desarrollar una aplicación Web multiplataforma.
- Adicionar nuevas funcionalidades que faciliten las tareas que se desarrollan en el entorno de trabajo del taller de reparaciones de la UCI

Para el desarrollo de este trabajo se trazaron las siguientes *tareas de investigación*:

- Estudiar el sitio actual del taller y realizar comparaciones con otras aplicaciones existentes que realicen tareas semejantes y adaptarlas al entorno laboral del taller de la UCI.
- Estudiar las normas establecidas para la elaboración de aplicaciones Web teniendo en cuenta las características del sistema a implementar y los requerimientos establecidos por el cliente.
- Investigar en Internet y otras bibliografías disponibles y especializadas las ventajas de las aplicaciones Web así como su uso y funcionamiento.
- Estudiar y conocer a fondo los lenguajes de programación y elementos fundamentales de ingeniería y gestión de software a utilizar para desarrollar la aplicación Web.
- Estudiar los posibles softwares vinculados al campo de acción.
- Estudiar los procesos vinculados al campo de acción.

Para lograr una mejor organización del contenido a tratar, se decidió que el documento estuviera estructurado en capítulos, los cuales se describen a continuación:

- *CAPITULO 1. FUNDAMENTACION TEORICA*: Se describen los principales aspectos y conceptos de relevancia que han sido objeto de análisis a lo largo de la investigación. Se explica el por qué se hace necesario el uso de cada herramienta, metodologías y lenguajes de programación para el sistema que se desea implementar.

- *CAPITULO 2. CARACTERISTICAS DEL SISTEMA:* Se aborda el objeto de estudio, problema y situación problemática, se detalla el objeto de automatización, la información que se maneja, se realiza una descripción general de la propuesta de sistema para el desarrollo de la aplicación Web. Se describen los requisitos funcionales y no funcionales con los que debe contar el sistema así como elementos clases del Negocio.
- *CAPITULO 3. ANALISIS Y DISEÑO DEL SISTEMA:* Aborda el flujo de trabajo Análisis y Diseño. Se definen las clases del análisis y el Modelo de clases de análisis. En el diseño se realizan los modelos de clases del diseño para cada caso de uso, se definen los diagramas de interacción por cada realización de casos de uso, se describen las clases persistentes y la posterior creación del modelo de Datos (Base de Datos).
- *CAPITULO 4. IMPLEMENTACION:* Se aborda todo lo referente al flujo de trabajo Implementación en su desarrollo para esta aplicación, para ello se expone el diagrama de despliegue que corresponde a la aplicación y descrito en los capítulos anteriores. Quedan reflejados los diagramas de componentes que forman la base de la programación de este sistema. También se definen elementos de las pruebas realizadas al sistema a través de su ciclo de vida y la final realización del mismo.
- *CAPITULO 5. ESTUDIO DE FACTIBILIDAD:* Aborda el método estimación aplicado a este sistema, así como que el análisis del costo, el esfuerzo y los beneficios que son procesos de suma importancia durante el desarrollo de una aplicación.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

## 1.1. Introducción.

Para el desarrollo del sistema informático, que gestione cada proceso a realizarse en el taller de reparaciones del GAT, y cumplir con los objetivos trazados para el mismo, es de vital importancia el uso de una aplicación Web que interactúe con el cliente y muestre cada información que este necesite para su trabajo.

En este capítulo se conocerá el taller de reparaciones de la UCI y su situación actual, ejemplos de otros talleres de reparación y su funcionamiento en la actualidad, así como una breve comparación entre otras aplicaciones que presentan una estructura funcional semejante.

En este capítulo se podrá conocer además detalles sobre las aplicaciones Web y así conocer por qué es importante desarrollar una aplicación de este tipo en el sistema que se implementará. Se verán las especificidades del lenguaje de programación propuesto, tanto del lado del cliente como del servidor, el framework de desarrollo Web propuesto, el gestor de base de datos seleccionado y los motivos de su elección para este sistema. Podrá conocer la herramienta de ingeniería y gestión de software empleada, y la metodología usada, haciendo un resumen de sus características fundamentales y motivos de su uso, y cómo se aplican al software que se pretende implementar, entre otros elementos seleccionados para la elaboración de la aplicación del taller de reparaciones del GAT, los patrones arquitectónicos y de diseño empleado.

## 1.2. Sobre los talleres de reparación.

El desarrollo de la informática ha dado lugar a una creciente demanda de medios tecnológicos y otros artefactos relacionados a la ciencia de la informática. Para solucionar los desperfectos y diversos problemas que presentan a diario muchos de estos medios se han creado talleres de reparación de servicios técnicos que no solo abarcan los dispositivos informáticos, sino que reparan otros artefactos relacionados al mundo digital como teléfonos, televisores, videos, equipos de música entre otros artefactos electrónicos. Los talleres de reparación también necesitan de un orden de trabajo y un sistema automatizado para gestionar cada tarea que allí se realice. A continuación se presentarán

algunos ejemplos de talleres de reparación que emplean sistemas informáticos para la gestión de su trabajo. Se verán ejemplos de otros países y de Cuba y dentro de ella la UCI.

Existen varios sistemas informáticos de gestión para talleres de reparación en el mundo. Como resultado de una búsqueda en Internet se encontró el Sistema de gestión de talleres SAT (versión 2007) de Argentina, que es un software de gestión y administración para talleres y servicios técnicos autorizados. SAT son las siglas de 'Sistema Administrativo para Talleres', y no por casualidad comparte las mismas siglas que "Servicio de Asistencia Técnica", denominación utilizada para los talleres de reparación. Datos sobre este sistema se puede encontrar en la dirección Web [www.satnetwork.com.ar](http://www.satnetwork.com.ar). En la actualidad el sistema SAT se utiliza en el taller de la UCI para realizar diversas comparaciones entre sus características principales y funcionalidades operativas y el sistema de Gestión Tecnológica de la UCI, esto se verá más adelante en este capítulo.

Existen además otras aplicaciones que se utilizan en diversos talleres de reparación de computadoras y otros medios electrónicos en otros países y muchas de esta son de libre acceso, o sea, sin licencia para su uso. A continuación se verá una breve descripción de las características de un sistema para la gestión de talleres de reparación publicado en el sitio [www.taringa.net](http://www.taringa.net) que tuvo una amplia aceptación por parte de los trabajadores de taller de reparación en muchas regiones.

Las características de este sistema son:

- Aplicación para que un taller de servicios técnicos almacene los datos de equipos a reparar.
- Está orientado a reparaciones de equipo de cómputo pero puede usarse con otro tipo de reparaciones.
- Si lo requiere puede agregar una contraseña que mantiene la confidencialidad de sus datos.
- Reportes de entrada y salida de equipos.
- Agenda de citas.
- Con búsquedas fáciles pero poderosas en todos los datos.
- Se depura fácilmente.
- Ayuda disponible.

Otras características de este sistema son los requerimientos de hardware que se necesitan, entre ellos mencionamos algunos como:

- Computadora con procesador Intel Celeron, Pentium 3 o 4, AMD Athlon, Athlon 64.
- 128 megabytes RAM en Windows 98, 2000 y Me.
- 256 megabytes RAM en Windows XP.
- Aproximadamente 20 MB de espacio en disco duro.
- Para software ellos proponen es uso de Windows 98, 2000, Me, XP o Vista.

Este sistema se puede descargar desde el sitio anteriormente mencionado o desde <http://www.cpimario.com/cprepair.zip> [1].

Otros ejemplos de sistemas para talleres de reparación se pueden ver en cada país que tenga un desarrollo avanzado de la informática. La mayoría de estos talleres usan sistemas ya creados en otros países y lo comparten libremente mediante los portales Web de distintas comunidades de desarrollo de software libre que existen en la actualidad, otros usan los métodos tradicionales de almacenamiento como son bases de datos en Access, tablas de Excel o en papel, pero la mayoría presenta su propio sistema informático ya sea Web o de escritorio (*desktop* en inglés)<sup>3</sup>. En varios sitios Web y Portales de Internet aparecen anuncios de estos talleres como propaganda. Se puede observar un ejemplo en el sitio Web [www.mercadolibre.com.co](http://www.mercadolibre.com.co) donde se ofertan servicios técnicos a diversos clientes, este es un taller colombiano. Veamos a continuación un ejemplo de sus ofertas:

*Nuestro Servicio de Reparación de computadores se realiza en nuestro taller o a domicilio. Diagnóstico de fallas y averías en Computadores Personales y Portátiles. Instalación de Sistemas Operativos Windows y Linux. Instalación física de redes de computadores en Salas de Internet y oficinas. Remoción de Virus. Instalación de Software adicional, reinstalación de Windows. Instalación de Partes y accesorios. Actualización de computadores a la última tecnología, expansión de memoria RAM. OTROS SERVICIOS Diseño logos para Páginas Web. Publicidad en Internet con Google<sup>4</sup> Adwords. [2]*

---

<sup>3</sup> Ver Glosario de Términos.

<sup>4</sup> Google es uno de los buscadores de Internet más potente y visitado en la actualidad.



Como se puede observar, con el desarrollo de la informática a nivel mundial, se han creado los talleres de servicios técnicos que han informatizado su entorno de trabajo para el control de cada tarea y otros que han compartido sus aplicaciones a colegas de todo el mundo.

En Cuba la empresa COPEXTEL (Empresa cubana comercializadora de equipos electrónicos y tecnologías) se encarga de la reparación de artefactos de computadoras y mantiene un sistema de talleres de servicios técnicos a todo lo largo del archipiélago. COPEXTEL es una sociedad mercantil cubana que inicia sus operaciones en el año 1991, con el objetivo de comercializar equipamiento electrónico de alta tecnología. El crecimiento de su actividad ha propiciado el desarrollado una red de servicios técnicos y de garantía, consultoría y otras prestaciones que enriquece considerablemente sus capacidades comerciales en el mercado cubano. La Corporación cuenta con capacidad para importar y exportar una amplia gama de equipos, partes y piezas, accesorios, muebles, componentes e insumos asociados a las soluciones que ofrece [3].

Esta empresa cuenta con un Portal Web donde presenta sus características, sus unidades de negocio, divisiones comerciales, territoriales, entre otros datos de interés.

En la UCI actualmente COPEXTEL cuenta con una aplicación Web propia, adaptada a su función en el centro (UCIXTEL<sup>5</sup>), que realiza esta tarea. La aplicación Web de UCIXTEL realiza diversas funciones, entre ellas está mostrar información sobre sus reportes y equipos en reparación y los ya reparados, además de procesar datos de sus tareas internas.

La aplicación del Grupo de Gestión Tecnológica “GATSERVER” de la UCI presenta algunas deficiencias y es necesario realizar cambios al mismo y para eso se ha propuesto la implementación de un nuevo sistema que siga las nuevas estrategias trazadas por el grupo de asistencia técnica, con nuevas modificaciones que se necesitan para favorecer y acelerar el trabajo con la calidad requerida. Dentro de este grupo se incluye la nueva aplicación del taller de reparaciones perteneciente al GAT.

En los siguientes epígrafes se podrán ver las tecnologías, lenguajes, metodologías y herramientas empleadas y el por qué de su uso en el desarrollo de esta aplicación.

### 1.3. Uso de software libre para el desarrollo de este sistema.

---

<sup>5</sup> Entidad de COPEXTEL que atiende a la UCI.

¿Qué se conoce como software libre?

El software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. De modo más preciso, se refiere a cuatro libertades de los usuarios del software:

- La libertad de usar el programa, con cualquier propósito.
- La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades. El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias, con lo que se puede ayudar a un compañero o entidad.
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. El acceso al código fuente es un requisito previo para esto [4].

La migración a software libre rompe las barreras que el software propietario mantiene sobre el desarrollo de la comunidad internacional, sobre todo en países pobres, y amplía los campos de conocimientos de la sociedad, aumentando las posibilidades de desarrollar y crear nuevas herramientas que resuelvan los problemas fundamentales de la misma.

La licencia que rige este tipo de software, se denomina GPL (GNU General Public License, Licencia Pública General). Dicha licencia se creó para proteger la integridad del software libre y evitar que nadie restrinja o utilice su circulación [5].

Todo programa o código sujeto a esta licencia permite:

- Descargar, utilizar y copiar libremente dicho programa, siempre que se incluya el copyright correspondiente y todo el texto correspondiente a la licencia.
- Incluir modificaciones y mejoras en dicho software y distribuirlas libremente, siempre que se especifique qué cambios se han hecho en el software original y la fecha de realización.
- Dicho software debe distribuirse libremente, sin coste alguno. Como mucho, se permite cobrar una cuota que costee los gastos de reproducción (por ejemplo, el CD en el que se graba).

La UCI como universidad para la formación de un profesional integral, proveedora de un espacio para el estudio, investigación y el desarrollo de las ciencias informáticas, debe estar a la vanguardia en los últimos adelantos de la

ciencia y la tecnología. Recibe gran importancia el uso de tecnologías y herramientas de software libre, necesarias para el futuro de la informática, que se encuentra en constante evolución.

### 1.4. Sobre aplicaciones Web.

Desde la creación de la Internet se ha ido desarrollando una amplia gama de sitios y aplicaciones Web que facilitan el trabajo de muchas entidades que realizan operaciones diarias con todo tipo de información que comparten entre diversas computadoras, y necesitan además el acceso a la red de redes, y la conexión a servidores de Bases de Datos.

¿Qué es un sitio Web?

Los sitios Web son sistemas interactivos, dinámicos, distribuidos y gráficos de hipertexto que funciona sobre Internet, para intercambiar información. Ofrecen una interfaz simple y consistente para acceder a los recursos de la red de redes.

¿Qué son las aplicaciones Web?

Son sitios especializados en realizar funciones inteligentes, realizando operaciones que interactúan con el cliente dando a este una respuesta a sus solicitudes. Tienen características similares a las páginas Web y son más populares por la facilidad de acceder a ellas a través de navegadores como Internet Explorer y Mozilla Firefox que cargan las aplicaciones mediante la red.

Las aplicaciones Web tienen una gran importancia sobre los programas tradicionales, a continuación se muestran algunas de ellas según un fragmento del artículo *Beneficios De Las Aplicaciones Basadas En Web Y El Anuncio De Microsoft De La Era "En Vivo"* de Noviembre 2, 2005 por Miguel Corsi. Las definiciones son las siguientes [6]:

- *Compatibilidad multiplataforma.* Las aplicaciones Web tienen un camino mucho más sencillo para la compatibilidad multiplataforma que las aplicaciones de software descargables. Varias tecnologías incluyendo Java, Flash, ASP y Ajax permiten un desarrollo efectivo de programas soportando todos los sistemas operativos principales.

- *Actualización.* Las aplicaciones basadas en Web están siempre actualizadas con el último lanzamiento sin requerir que el usuario tome acciones pro-activas, y sin necesitar llamar la atención del usuario o interferir con sus hábitos de trabajo con la esperanza de que va a iniciar nuevas descargas y procedimientos de instalación (algunas veces imposible cuando se está trabajando dentro de grandes organizaciones).
- *Inmediatez de acceso.* Las aplicaciones basadas en Web no necesitan ser descargadas, instaladas y configuradas. El usuario accede a su cuenta online y están listas para trabajar sin importar cuál es su configuración o su hardware.
- *Menos requerimientos de memoria.* Las aplicaciones basadas en Web tienen muchas más razonables demandas de memoria RAM de parte del usuario final que los programas instalados localmente. Al residir y correr en los servidores del proveedor, a esas aplicaciones basadas en Web usa en muchos casos la memoria de las computadoras que ellos corren, dejando más espacio para correr múltiples aplicaciones del mismo tiempo sin incurrir en frustrantes deterioros en el rendimiento.
- *Menos Bugs<sup>6</sup>.* Las aplicaciones basadas en Web deberían ser menos propensas a colgarse y crear problemas técnicos debido a software o conflictos de hardware con otras aplicaciones existentes, protocolos o software personal interno. Con aplicaciones basadas en Web, todos utilizan la misma versión, y todos los bugs pueden ser corregido tan pronto como son descubiertos. Esta es la razón por la cual las aplicaciones basadas en Web deberían tener mucho menos bugs que el software de escritorio descargable tradicional.
- *Precio.* Las aplicaciones basadas en Web no requieren la infraestructura de distribución, soporte técnico y marketing requerido por el software descargable tradicional. Esto permite que las aplicaciones online cuesten una fracción de sus contrapartes descargables si no totalmente gratuitas, mientras que ofrecen componentes adicionales y servicios premium como una opción.
- *Los datos también van online.* Por supuesto con el desplazamiento de las aplicaciones locales a aquellas basadas en Web también los datos que creamos y accedemos van a necesitar experimentar profundos cambios. A nadie le gusta no poder acceder a su propio e-mail cuando está de viaje, o poder recuperar un documento particular cuando se conecta desde un ciber café a 15.000 kilómetros de su oficina.

---

<sup>6</sup> Ver Glosario de Términos.

- *Múltiples usuarios concurrentes.* Las aplicaciones basadas en Web pueden realmente ser utilizadas por múltiples usuarios al mismo tiempo. No hay más necesidad de compartir pantallas o enviar instantáneas cuando múltiples usuarios pueden ver e incluso editar el mismo documento de manera conjunta. Las compañías de conferencia Web y colaboración online están involucradas algunas transformaciones claves y los usuarios necesitan explorar que significa realmente trabajar efectivamente y co-editar documentos juntos.
- *Los datos son más seguros.* Si bien la ruptura de discos no va a desaparecer, es probable que los usuarios escuchen mucho menos del tema. A medida que las compañías se hagan cargo del almacenamiento de los datos del usuario, granjas de almacenamiento de datos redundantes, altamente fiables, serán la norma más que la excepción, y los usuarios van a tener mucho menos riesgo de perder sus datos debido a una ruptura de disco impredecible o a un virus de la computadora.
- *Desarrollar aplicaciones en cualquier lenguaje.* Una vez que las aplicaciones han sido separadas de computadoras locales y sistemas operativos específicos esos pueden también ser escritas en prácticamente cualquier lenguaje de programación. Ya que las aplicaciones Web son esencialmente una colección de programas más que un simple programa, ellas podrían ser escritas en cualquier lenguaje de programación. Mientras que para software escritorio está limitado a usar el mismo lenguaje que el sistema operativo subyacente este no es el caso cuando la aplicación de software es independiente del sistema operativo.

Estas y otras integran la amplia gama de posibilidades que brinda el uso de las aplicaciones Web, demostrando su importancia en el desarrollo de la industria de software actualmente.

El uso de páginas Web dinámicas posibilita además otras ventajas para el usuario:

- Gran número de posibilidades en su diseño y desarrollo.
- El visitante puede alterar el diseño, contenidos o presentación de la página a su gusto.
- En su realización se utilizan diversos lenguajes y técnicas de programación.
- El proceso de actualización es sumamente sencillo, sin necesidad de entrar en el servidor.
- Permite un gran número de funcionalidades tales como bases de datos, foros. etc.
- Pueden realizarse íntegramente software de libre distribución.
- Existe una amplia comunidad de programadores que brinda apoyo desinteresado.
- Cuenta con un gran número de soluciones prediseñadas de libre disposición.

### 1.5. Metodología propuesta para el desarrollo del software en este sistema: RUP.

Se propone como metodología de desarrollo de software el Proceso Unificado Racional (*Rational Unified Process*) ya que es un proceso de desarrollo de software, que junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. Su ciclo de vida se caracteriza por estar centrado en la arquitectura, guiado por casos de uso y ser iterativo e incremental [7].

La metodología RUP se divide en 4 fases el desarrollo del software:

- *Inicio*, El Objetivo en esta etapa es determinar la visión del proyecto.
- *Elaboración*, En esta etapa el objetivo es determinar la arquitectura óptima.
- *Construcción*, En esta etapa el objetivo es llevar a obtener la capacidad operacional inicial.
- *Transición*, El objetivo es llegar a obtener el release<sup>7</sup> del proyecto.

Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los Objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes.

Vale mencionar que el ciclo de vida que se desarrolla por cada iteración, es llevada bajo dos disciplinas:

#### *Disciplina de Desarrollo*

- *Modelo de Negocios*: Se entienden las necesidades del negocio.
- *Requerimientos*: Se trasladan las necesidades del negocio a un sistema automatizado.
- *Análisis y Diseño*: Se trasladan los requerimientos dentro de la arquitectura de software.
- *Implementación*: Se crea el software que se ajuste a la arquitectura y que tenga el comportamiento deseado.
- *Pruebas*: Se asegura que el comportamiento requerido es el correcto y que todo lo solicitado está presente.

#### *Disciplina de Soporte*

---

<sup>7</sup> Versión del producto realizado.

- Configuración y administración del cambio: Se guardan todas las versiones del proyecto.
- Administrando el proyecto: Se administran horarios y recursos.
- Ambiente: Se administra el ambiente de desarrollo.
- Distribución: Se hace todo lo necesario para la salida del proyecto.

La Metodología RUP es más adaptable para proyectos de largo plazo, otra de las características por la cual se propone esta metodología de desarrollo de software y se adapta más a las necesidades de desarrollo de este sistema.

1.6. Lenguaje de programación Web de lado del servidor propuesto para implementar el sistema: PHP Versión 5.0.

En este trabajo se propone realizar la programación de la aplicación Web del lado del servidor usando lenguaje PHP. El PHP (acrónimo de “PHP: Hypertext Preprocessor”), es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor.

¿Qué se conoce por PHP?

Este lenguaje se inició como una modificación a Perl escrita por Rasmus Lerdorf a finales de 1994. Su primer uso fue el de mantener un control sobre quien visitaba su curriculum en su Web. En los siguientes tres años, se fue convirtiendo en lo que se conoce como PHP/FI 2.0. Esta forma de programar llegó a muchos usuarios, pero el lenguaje no tomó el peso actual hasta que Zeev Surasky y Andi Gutmans le incluyeron nuevas características en 1997, que dio por resultado el PHP 3.0.

Según estudios más de un millón de servidores tienen esta capacidad implementada y los números continúan creciendo. La UCI, que está dando los primeros pasos en el desarrollo del software libre, está implementando sus aplicaciones Web en este lenguaje de fácil acceso y que permite una mayor seguridad a los usuarios, así como una reusabilidad a aquellos desarrolladores que deseen adaptar su aplicación a las condiciones de su entorno de trabajo. El PHP se vincula perfectamente al diseño gráfico de una aplicación Web, dando grandes posibilidades para su desarrollo, y permite una conexión a Bases de Datos con mayor facilidad. Es por esto que se selecciona este lenguaje que día a día va tomando auge en toda la comunidad informática internacional, en especial los grupos de desarrollo Web.

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

Comparando PHP con otro lenguaje de programación Web del lado del servidor como ASP, se tiene que este último es software propietario y que PHP es open source<sup>8</sup>. ASP.NET es un intento de Microsoft para introducirse en el mercado del desarrollo Web, y viene a ser como su estándar para su servidor Web, ISS. Asp ha sido atacado por la comunidad open source de desarrollo Web desde que este apareció, y dan numerosas razones para ello, entre las que se encuentran que: es propietario, solo se desarrolla sobre plataforma Microsoft y la lentitud que presenta. PHP es más flexible y no es muy complicado su aprendizaje. Además presenta mucha información al respecto por la gran comunidad de desarrollo de que dispone.

Con PHP es posible crear aplicaciones con una interfaz gráfica para el usuario como mencionábamos anteriormente (llamada GUI), utilizando la extensión PHP-Qt o PHP-GTK. También puede ser usado desde la línea de órdenes, de la misma manera como Perl o Python pueden hacerlo, esta versión de PHP se llama PHP CLI (*Command Line Interface*).

Su interpretación y ejecución se da en el servidor Web, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página Web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente. Además es posible utilizar PHP para generar archivos PDF, Flash, así como imágenes en diferentes formatos, entre otras cosas.

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de Aplicaciones Web muy robustas.

---

<sup>8</sup> Ver Glosario de Términos.



PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como UNIX (y de ese tipo, como Linux o Mac OS X) y Windows, y puede interactuar con los servidores de Web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

Al tratarse de software libre, PHP, puede descargarse y utilizarse en cualquier aplicación, personal o profesional, de manera completamente libre.

PHP también ofrece la integración con las varias bibliotecas externas, que permiten que el desarrollador haga casi cualquier cosa desde generar documentos en pdf hasta analizar código XML [8].

Cuatro de las grandes características de este lenguaje son: Velocidad, estabilidad, seguridad y simplicidad.

- *Velocidad:* No solo la velocidad de ejecución, la cual es importante, sino además no crea demoras en la máquina. Por esta razón no debe requerir demasiados recursos de sistema. PHP se integra muy bien junto a otro software, especialmente bajo ambientes Unix, cuando se configura como módulo de Apache, esta listo para ser utilizado.
- *Estabilidad:* La velocidad no sirve de mucho si el sistema se cae cada cierta cantidad de ejecuciones. Ninguna aplicación es 100% libre de bugs, pero teniendo de respaldo una increíble comunidad de programadores y usuarios es mucho más difícil para los bugs sobrevivir. PHP utiliza su propio sistema de administración de recursos y dispone de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.
- *Seguridad:* El sistema debe poseer protecciones contra ataques. PHP provee diferentes niveles de seguridad, estos pueden ser configurados desde el archivo .ini
- *Simplicidad:* Se les debe permitir a los programadores generar código productivamente en el menor tiempo posible. Usuarios con experiencia en C y C++ podrán utilizar PHP rápidamente.

Estas y otras múltiples razones son las causas de que se emplee este lenguaje en el desarrollo de la aplicación Web para el taller de reparaciones. Para esta aplicación se empleará la versión 5.0 de PHP.

*Sobre PHP 5.0.*

A continuación se verán algunas de las características de PHP 5.0, versión sobre la cual se desarrolló este sistema:

- Completamente nuevo soporte de XML, gracias a la poderosa librería libxml2, inicialmente escrita para GNOME<sup>9</sup>.
- Para los más apresurados, también está la extensión *SimpleXML* para manipular archivos XML como objetos PHP, con una facilidad no conocida hasta ahora.
- Nueva extensión SOAP<sup>10</sup> propia para acceder a *Web Services*.

### 1.7. Programación del lado del cliente.

En la programación Web, el HTML es el lenguaje que permite codificar o preparar documentos de hipertexto, que viene a ser el lenguaje común para la construcción de una página Web. XHTML es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas Web. XHTML es la versión XML de HTML, por lo que tiene, básicamente, las mismas funcionalidades, pero cumple las especificaciones, más estrictas, de XML.

Para realizar validaciones y otras funciones que se ejecuten desde el cliente se usará JavaScript implementado en las distintas instrucciones que facilita HTML dentro de la página Web.

JavaScript es un lenguaje de script multiplataforma, orientado a objetos, es pequeño y ligero; no es útil como un lenguaje independiente, más bien está diseñado para una fácil incrustación en otros productos y aplicaciones, tales como los navegadores Web. Dentro de un entorno anfitrión, JavaScript puede ser conectado a los objetos de su entorno para proveer un control programable sobre éstos.

Todos los navegadores interpretan el código JavaScript integrado dentro de las páginas Web. Para interactuar con una página Web se provee al lenguaje JavaScript de una implementación del DOM (Modelo de Objetos de Documento)<sup>11</sup>.

---

<sup>9</sup> Ver Glosario de Términos.

<sup>10</sup> Ver Glosario de Términos.

<sup>11</sup> Ver Glosario de Términos.

### 1.8. Uso de Ajax.

El uso de Ajax, en este sistema, se evidencia en las validaciones de formularios y en cada vista que utilizan eventos donde se cargan elementos en la misma página de forma interactiva, brindando una rápida respuesta al usuario y facilitando el desarrollo de cada funcionalidad.

AJAX, acrónimo de *Asynchronous JavaScript And XML* (JavaScript asíncrono y XML), es una técnica de desarrollo Web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Éstas se ejecutan en el cliente, es decir, en el navegador de los usuarios y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página sin necesidad de recargarla. Esto significa aumentar la interactividad, velocidad y usabilidad en la misma.

### 1.8. Framework propuesto para el desarrollo de la aplicación.

Para la implementación de esta aplicación se emplea el framework de desarrollo Web para PHP: CodeIgniter. ¿Qué se conoce como framework?

El término framework se refiere a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. Un framework se puede considerar como una aplicación genérica incompleta y configurable a la que se puede añadir las últimas piezas para construir una aplicación concreta.

Los objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones. Un framework Web, por tanto, se define como un conjunto de componentes (por ejemplo clases en Java y descriptores y archivos de configuración en XML) que componen un diseño reutilizable que facilita y agiliza el desarrollo de aplicaciones Web.

CodeIgniter es un entorno de trabajo Web para el desarrollo de aplicaciones Web en PHP que facilita la escritura de código fuente en este lenguaje y reduce el tiempo necesario para lograrlo. Está basado en el patrón Modelo Vista Controlador (MVC). Además CodeIgniter posibilita el desarrollo de proyectos pequeños y medianos, como es el caso de este sistema, y es más fácil su aprendizaje.

Una de las características más interesantes de CodeIgniter es el elevado número de clases que incluye para trabajar con distintos objetos: calendario, bases de datos, correo electrónico, manipulación de imágenes, FTP, lenguaje, tablas, sesiones, compresión ZIP, etcétera [9].

A diferencia de otros framework, CodeIgniter cuenta con una documentación excelente que permite conocer todos los secretos de este entorno de trabajo. CodeIgniter favorece el desarrollo de aplicaciones que se ejecutan en hosting<sup>12</sup> compartido que ejecutan muchas versiones de PHP con diferentes configuraciones.

Para poder ejecutar CodeIgniter se debe contar con un servidor Web como Apache y el lenguaje de programación PHP. También es posible su utilización de forma remota.

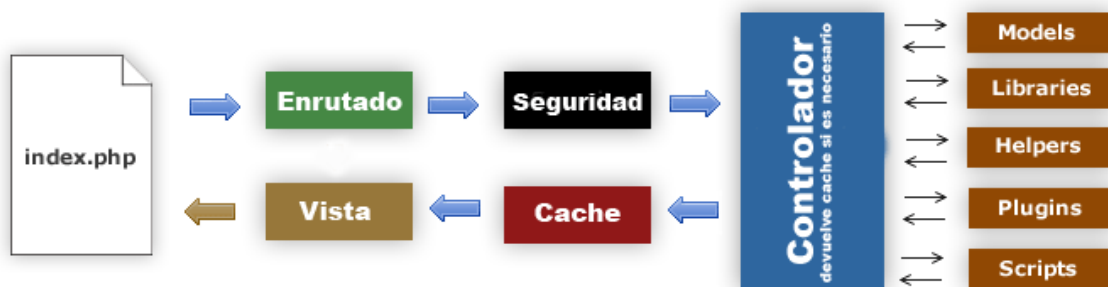


Figura 1.1. Diagrama de flujo de CodeIgniter.

Para una mejor comprensión de este framework se verá a continuación una breve explicación del diagrama de flujo del CodeIgniter observado en la imagen:

1. El `index.php` sirve para inicializar el núcleo de CodeIgniter.
2. El `enrutador` examina la petición HTTP y determina que se debe de hacer.
3. `Seguridad`, antes de que se cargue el controlador se filtra los datos enviados para que estos sean fiables.
4. El `controlador` comprueba si existe `cache`<sup>13</sup>, en caso afirmativo la devuelve pero al hacer la comprobación en este punto podemos llegar a pasar datos dinámicos a la `vista` (se podría lograr que funcionara un módulo de autenticación aún con la `cache` habilitada), si no existe

<sup>12</sup> Ver Glosario de Términos.

<sup>13</sup> Ver Glosario de Términos.

*cache* carga el modelo, librerías, plugins<sup>14</sup>, helpers y todos los recursos necesarios para satisfacer la petición.

5. Una vez que la *Vista* está renderizada<sup>15</sup> es enviada al navegador si la *cache* está habilitada se almacena el resultado en *cache* para la próxima vez que esta URL<sup>16</sup> sea servida [10].

### 1.9. Lenguaje de modelado para los procesos de ingeniería de software. UML.

En la actualidad UML está considerado como el lenguaje estándar en el análisis y diseño de sistemas de cómputo. Mediante este lenguaje es posible establecer la serie de requerimientos y estructuras necesarias para plasmar un sistema de software previo al proceso intensivo de escribir código.

¿Qué es el UML?

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

UML prescribe una notación estándar y semánticas esenciales para el modelado de un sistema orientado a objetos. Previamente, un diseño orientado a objetos podría haber sido modelado con cualquiera de la docena de metodologías populares, causando a los revisores tener que aprender las semánticas y notaciones de la metodología empleada antes que intentar entender el diseño en sí. Ahora con UML, diseñadores diferentes modelando sistemas diferentes pueden sobradamente entender cada uno los diseños de los otros. Para el modelado de este sistema se usa como herramienta el Visual Paradigm.

### 1.10. Servidor de Base de Datos propuesto para el desarrollo de esta aplicación: PostgreSQL.

¿Qué es un sistema de gestión de Base de Datos?

---

<sup>14</sup> Ver Glosario de Términos

<sup>15</sup> Ver Glosario de Términos.

<sup>16</sup> Ver Glosario de Términos.

Los Sistemas de gestión de base de datos son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. En los textos que tratan este tema, o temas relacionados, se mencionan los términos SGBD y DBMS, siendo ambos equivalentes, y acrónimos, respectivamente, de Sistema Gestor de Bases de Datos y *DataBase Management System*, su expresión inglesa.

Entre los SGBD libres que se emplean en el desarrollo de sistemas informáticos:

- PostgreSQL. Licencia BSD.
- MySQL Licencia Dual, depende el uso.
- Firebird basada en la versión 6 de InterBase, Initial Developer's PUBLIC LICENSE Versión 1.0.
- SQLite. Licencia Dominio Público.
- DB2 Express-C.
- Apache Derby.
- Sybase ASE Express Edition para Linux (Edición gratuita para Linux)

PostgreSQL:

El PostgreSQL es un servidor de base de datos objeto relacional libre, liberado bajo la licencia BSD<sup>17</sup>. Como muchos otros proyectos open source, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo, dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group). Es multiplataforma.

Entre sus características fundamentales se encuentran la alta concurrencia que permite, mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés), PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo *commit*<sup>18</sup>. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.

---

<sup>17</sup> Ver Glosario de Términos.

<sup>18</sup> Ver Glosario de Términos.

Un *disparador* o *trigger* se define en una acción específica basada en algo ocurrente dentro de la base de datos. En PostgreSQL esto significa la ejecución de un procedimiento almacenado basado en una determinada acción sobre una tabla específica. PostgreSQL le permitirá crear una amplia funcionalidad a través de su sistema de activación de disparadores (triggers).

Entre sus funciones se tiene que bloques de código que se ejecutan en el servidor pueden ser escritos en varios lenguajes, con la potencia que cada uno de ellos da, desde las operaciones básicas de programación, tales como bifurcaciones y bucles, hasta las complejidades de la programación orientada a objetos o la programación funcional.

Otras de las características por la que ha sido propuesto PostgreSQL es que posee una gran escalabilidad. Es capaz de ajustarse al número de CPUs y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta. Implementa el uso de rollback's<sup>19</sup>, subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz, y ofreciendo soluciones en campos en las que otros SGBD como MySQL no podría realizar. Tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de bases de datos de alto nivel, como puede ser Oracle [11].

Comparando PostgreSQL con otro gestor de base de datos relacionado a PHP como es el MySQL, concluimos que PostgreSQL permite una alta concurrencia y permite desarrollar bases de datos con gran flujo de información.

### 1.11. Servidor Web propuesto. Apache.

¿Qué se entiende por servidor Web?

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

Apache:

---

<sup>19</sup> Ver Glosario de Términos.

Se propone como servidor Web el Apache ya que es un software (libre) servidor HTTP de código abierto. Es aplicable para muchos sistemas operativos, por ejemplo:

- NetWare.
- OS/2.
- Unix.
- Windows 95/98.
- Windows NT 4.0.
- Windows XP Professional.

Apache está diseñado para ser un servidor Web potente y flexible que pueda funcionar en la más amplia variedad de plataformas y entornos. Las diferentes plataformas y entornos, hacen que a menudo sean necesarias diferentes características o funcionalidades. Apache se ha adaptado siempre a una gran variedad de entornos a través de su diseño modular. Este diseño permite a los administradores de sitios y aplicaciones Web elegir qué características van a ser incluidas en el servidor seleccionando y qué módulos se van a cargar, ya sea al compilar o al ejecutar el servidor. El servidor puede personalizarse mejor para las necesidades de cada sitio o aplicación Web [12].

1.12. Herramienta propuesta para la modelación de los diagramas de ingeniería de software para el desarrollo este sistema: Visual Paradigm.

Para la ingeniería de software de a aplicación en cuestión se emplea como herramienta de modelado el Visual Paradigm, esta es una herramienta CASE multiplataforma que utiliza "UML": como lenguaje de modelado.

¿Qué es una herramienta CASE?

Las Herramientas CASE (*Computer Aided Software Engineering*, Ingeniería de Software Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, calculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras.



Visual Paradigm es una herramienta colaborativa, es decir, soporta múltiples usuarios trabajando sobre el mismo proyecto; genera la documentación del proyecto automáticamente en varios formatos como Web o .Pdf, y permite control de versiones. Incorpora el soporte para trabajo en equipo, que permite que varios desarrolladores trabajen a la vez en el mismo diagrama y vean en tiempo real los cambios hechos por sus compañeros de equipo. Existen varias ediciones de VP con prestaciones diferentes. La más interesante puede ser la Community Edition pues, aunque está algo limitada, es gratuita si se usa con fines no comerciales.

Tiene capacidad para la ingeniería directa e inversa en Java, C++, PHP entre otras, presenta además disponibilidad de múltiples versiones para cada necesidad. Es multiplataforma y muy útil para la generación de código fuente en PHP. Tiene la capacidad de crear el esquema de clases a partir de una base de datos y crear la definición de base de datos a partir del esquema de clases.

Cabe destacar igualmente su robustez, usabilidad y portabilidad.

Visual Paradigm es una herramienta a tener en cuenta a la hora de trabajar en un proyecto importante. El diseño es centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad. Esta herramienta usa un lenguaje estándar común para todo el equipo de desarrollo que facilita la comunicación. Permite invertir código fuente de programas, archivos ejecutables y binarios en modelos UML al momento, creando de forma simple toda la documentación [13].

### 1.13. Uso de Macromedia Dreamweaver 8.0 y Zend Development Environment.

Dreamweaver es una herramienta de desarrollo Web de fácil manejo que facilita el diseño, desarrollo y el mantenimiento de sitios y aplicaciones Web. La conjunción de PHP, bases de datos como PostgreSQL o MySQL y Dreamweaver, es uno de los motivos para el uso de esta herramienta en el desarrollo de este trabajo, además que permite el uso de estilos CSS<sup>20</sup> para obtener un producto acorde a las restricciones de diseño del proyecto.

Dreamweaver 8 (Versión utilizada para el desarrollo de este sistema) incluye herramientas para trabajar en aplicaciones que manejan XML, así como mejoras a su manejo de hojas de estilo (CSS). Del lado del XML, incluye interesantes herramientas visuales para incluir contenidos de este formato

---

<sup>20</sup> Ver Glosario de Términos.

como son los RSS<sup>21</sup>, integrándolos fácilmente en sitios Web y aplicaciones. Para el trabajo con CSS ha simplificado la creación y manejo de diferentes estilos, promoviendo los estándares para nuevos usuarios y facilitando su aplicación para usuarios avanzados [14].

Otra de las herramientas empleadas para la implementación del sistema y como apoyo al trabajo realizado sobre Dreamweaver fue el Zend Studio o Zend Development Environment.

Las siguientes características evidencian el por qué del uso de esta herramienta como apoyo al desarrollo del sistema:

- Resaltado de sintaxis, autocompletado de código, ayuda de código y lista de parámetros de funciones y métodos de clase.
- Plegado de código (comentarios, bloques de phpDoc, cuerpo de funciones y métodos e implementación de clases).
- Inserción automática de paréntesis y corchetes de cierre.
- Sangrado automático y otras ayudas de formato de código.
- Emparejamiento (*matching*) de paréntesis y corchetes (si se sitúa el cursor sobre un paréntesis (corchete) de apertura (cierre), Zend Studio localiza el correspondiente paréntesis (corchete) de cierre (apertura)).
- Detección de errores de sintaxis en tiempo real.
- Además que se adapta a diversos sistemas operativos como Windows, Linux y MacOS.

### 1.14. Modelo-Vista-Controlador (MVC)

El patrón conocido como Modelo-Vista-Controlador (MVC) separa el modelado del dominio, la presentación y las acciones basadas en datos ingresados por el usuario en tres clases diferentes:

**Modelo:** Administra el comportamiento y los datos del dominio de aplicación, responde a requerimientos de información sobre su estado (usualmente formulados desde la vista) y responde a instrucciones de cambiar el estado (habitualmente desde el controlador).

**Vista:** Maneja la visualización de la información.

**Controlador:** Controla el flujo entre la vista y el modelo (los datos).

Tanto la vista como el controlador dependen del modelo, el cual no depende de las otras clases. Esta separación permite construir y probar el modelo, independientemente de la representación visual.

---

<sup>21</sup> Ver Glosario de Términos.

Entre las ventajas del estilo Modelo-Vista-Controlador están las siguientes:

**Soporte de múltiples vistas:** Dado que la vista se halla separada del modelo y no hay dependencia directa del modelo con respecto a la vista, la interfaz de usuario puede mostrar múltiples vistas de los mismos datos simultáneamente. Por ejemplo, múltiples páginas de la aplicación Web pueden utilizar el mismo modelo de objetos mostrado de maneras diferentes.

**Adaptación al cambio:** Los requerimientos de interfaz de usuario tienden a cambiar con mayor rapidez que las reglas de negocios. Los usuarios pueden preferir distintas opciones de representación, o requerir soporte para nuevos dispositivos, algo que se espera en futuras versiones de este sistema. Dado que el modelo no depende de las vistas, agregar nuevas opciones de presentación generalmente no afecta al modelo [16].

A continuación se verá un diagrama sencillo que muestra la relación entre el modelo, la vista y el controlador.

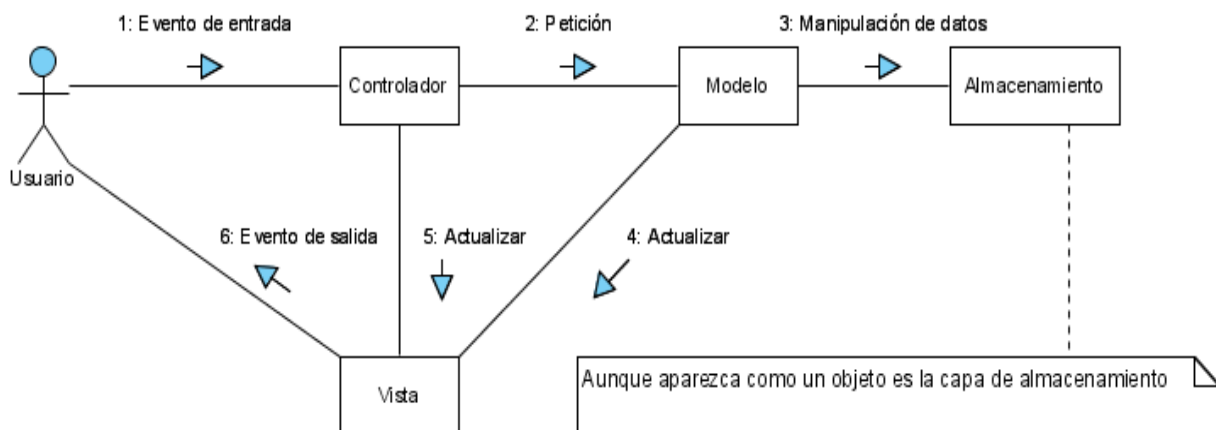


Figura 1.2. Representación gráfica del Modelo-Vista-Controlador.

### 1.15. Arquitectura Cliente-Servidor

Esta consiste en varios clientes distribuidos en diferentes nodos, conectados en red a uno o varios nodos servidores, donde el servidor puede servir a varios clientes a la vez. En los nodos clientes generalmente se encuentra la interfaz de usuario y en los nodos servidores la lógica del negocio. Este es un sistema que será desarrollado en una aplicación Web que contará además con varios nodos clientes que se podrán conectar a uno o varios servidores, según el despliegue a realizarse teniendo en cuenta la alta demanda y el tamaño de los datos a manejarse, por lo tanto el uso de este patrón

arquitectónico es muy importante y necesario teniendo en cuenta las características de este sistema [16].

### 1.16. Patrones de Diseño empleados.

Los Patrones de Diseño, en su forma más sencilla, son buenas soluciones que pueden ser aplicadas a problemas recurrentes que surgen durante el diseño de un sistema o aplicación.

Los patrones de diseño empleados en el desarrollo de este sistema entran en el grupo de los denominados patrones básicos GRASP (patrones de los principios generales para asignar responsabilidades). A continuación se verán algunos de estos patrones más usados en este sistema.

*Patrón\_Experto:* Este patrón asigna una responsabilidad al experto en la información. Una clase contiene toda la información necesaria para cumplir una responsabilidad.

*Patrón Creador:* El patrón creador ayuda a identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases. La nueva instancia deberá ser creada por la clase que:

- Tiene la información necesaria para realizar la creación del objeto.
- Usa directamente las instancias creadas del objeto.
- Almacena o maneja varias instancias de la clase.

*Patrón Controlador:* El patrón controlador es un patrón que sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado.

*Patrón Bajo Acoplamiento:* Es la idea de tener las clases lo menos ligadas entre sí que se pueda. De tal forma que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, potenciando la reutilización, y disminuyendo la dependencia entre las clases.

*Patrón Alta Cohesión:* Con el uso de este patrón la información que almacena una clase debe de ser coherente y está en la mayor medida de lo posible relacionada con la clase [17].

### 1.17. Conclusiones.

Después de estudiar el campo de acción y sus características fundamentales, se ha llegado a la conclusión de que es importante realizar una aplicación Web, teniendo en cuenta la facilidad de acceso al mismo por los usuarios. Dadas las características fundamentales, y ventajas, del uso de software libre se creará un sistema programado en PHP, a través del framework para desarrollo Web: CodeIgniter, con un fácil y rápido acceso al sistema de base de datos empleado (PostgreSQL) y un modelado de cada diagrama propuesto en cada flujo de trabajo de RUP en la herramienta Visual Paradigm, que agiliza y facilita cada tarea a realizar durante el desarrollo del software. Además se pudo conocer en esencia los patrones arquitectónicos y de diseño empleados en el desarrollo de este sistema.

Es de vital importancia el dominio de estas herramientas, metodologías, lenguajes y tecnologías, y ver cómo se pueden aplicar para desarrollar un sistema de máxima calidad que cumpla con los requisitos propuestos y brinde al cliente una versión del producto que satisfaga sus intereses.

## CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

### 2.1. Introducción.

En este Capítulo se presentará una descripción del negocio del problema a resolver. Se podrá ver además los diversos casos de uso del sistema y los requerimientos funcionales y no funcionales con los cuales se realizará posteriormente un análisis y diseño para concluir con la implementación, prueba y presentación de la primera versión de este sistema a automatizar, pero esto se conocerá en próximos capítulos. Teniendo en cuenta la metodología RUP se comienza este trabajo de modelación del negocio en la fase de inicio<sup>22</sup>.

### 2.2. Objeto de Estudio.

Se necesita crear un sistema que debe actualizar cada reporte que llegue al taller, y se debe tener el estricto control de cada uno de ellos por una persona encargada de gestionar cada proceso que allí se ejecute.

El taller de reparaciones de la UCI es un medio laboral donde se reparan los medios tecnológicos de la universidad que llegan al mismo. El objetivo del taller es dar soporte técnico a cada equipo electrónico de la universidad, siempre que exista la disponibilidad para darle soporte. El trabajo en el taller presenta diversos procesos que se interrelacionan uno con otro; un proceso no puede realizarse sin la ejecución de una actividad anterior. El personal ajeno al taller (estudiantes, trabajadores, profesores) cuando tienen un problema con un equipo electrónico envía una solicitud al Grupo de Asistencia Técnica (GAT). Esta solicitud es recibida y se crea un reporte que se le asigna a la Brigada de Atención al Usuario (BAU) que es la encargada de resolver cada deficiencia según el lugar donde se encuentre (residencia, docentes, aéreas administrativas). Al taller llegan las solicitudes de los reportes de equipos defectuosos que no pueden ser reparados por la BAU y los equipos del área de recambio que presenten algún defecto. El área de recambio es un local que contiene equipos para ser cambiados por otros que están defectuosos, pero en caso de presentar algunos desperfectos se envían directamente al taller. Si no se puede arreglar el equipo en el taller este reasigna el medio tecnológico a UCIXTEL, que se encarga de la reparación del equipo, y si no tiene solución el equipo es

---

<sup>22</sup> Ver Epígrafe 1.5.

dado de baja técnica. En la actualidad existe un sistema (GATSERVER) que realiza algunas de estas actividades, pero no cumple con algunos requisitos que los trabajadores del taller necesitan que el mismo tenga como por ejemplo:

- El sistema debe adaptarse a las nuevas herramientas y tecnologías de software libre y dejar plasmado, además del cambio de estado del reporte, la solución dada a cada incidencia y almacenar esta información en un historial.
- Presentar además una base de conocimiento que contenga todas las posibles soluciones que se pueden dar a la rotura de cualquier equipo que se repare en el taller.

### 2.3. Objeto de automatización.

Serán objeto de automatización los procesos de gestión de cada reporte llegado al taller, los registros de datos y la búsqueda en la base de conocimiento y generar estadísticas de los equipos reparados en el taller y los enviados a UCIXTEL. La Brigada de Atención al Usuario (BAU) recibe reportes de equipos defectuosos, si no pueden dar solución al problema asignado ese reporte pasa al taller. El área de recambio almacena equipos que pueden estar también defectuosos y estos pasan también con su reporte asignado al taller. En el taller se soluciona el problema, cada reporte actualizado es almacenado en un historial de reportes propios del taller. Pero en caso de no poder reparar el equipo se pasa el reporte a UCIXTEL. Si UCIXTEL no puede dar soporte técnico al equipo, este medio se declara como baja técnica. Si el taller puede dar el soporte técnico al equipo, una vez finalizado el mismo se actualiza el reporte con los arreglos hechos y se almacenan los datos del reporte en el historial. Además es objeto de automatización la creación de una base de conocimientos que registre cada equipo, incidencia y solución que entre al taller y que facilite la consulta de estos datos posteriormente. A continuación se verán los procesos más importantes del sistema del taller de reparaciones que son objeto de automatización.

- El personal con acceso al sistema deben ser los técnicos que trabajan en el taller, que deben autenticarse para poder acceder al mismo, para así mantener segura la información que allí se procesa.
- Se debe crear y mantener actualizada una base de conocimientos que contenga toda la información de cada equipo así como sus incidencias y la solución a cada una.

- Una vez recibido el reporte, los técnicos se disponen a reparar el equipo y luego entregan el reporte con los datos actualizados del mismo. Cada reporte atendido debe ser almacenado en un historial del taller.
- En caso de no poder reparar el equipo se reasigna el reporte a UCIXTEL y se almacenan sus datos en el historial del taller.
- El sistema debe ser capaz de mostrar estadísticas de los reportes que han sido tratados por el taller, mostrando la información de cada uno según un intervalo de fechas dadas, el tipo de equipo y el total, además de la información almacenada en el historial.
- El sistema debe ser capaz de graficar los resultados de las estadísticas.

### 2.4. Información que se maneja.

En esta aplicación debe manejarse una serie de datos de los reportes llegados al taller, que deben ser bien comprobados antes de proceder a su actualización. El taller recibe la solicitud de cada reporte que se identifican por un número único del mismo y los datos de cada deficiencia que presenta el equipo, el tipo de equipo, la descripción del problema según el usuario, la fecha y la hora de llegada del taller, el estado y la clave entre otros datos a manejar por el técnico en este sistema. Estos datos son necesarios para elaborar estadísticas que muestran en detalle las características de los reportes tratados por el taller. El técnico para devolver el reporte, una vez reparado el equipo, debe actualizar el mismo con los datos de los arreglos realizados al medio tecnológico. Una vez enviado el reporte, estos datos se almacenan en el historial del taller para su posterior consulta.

Otra información que maneja el sistema son los datos de cada equipo, incidencia y solución que se almacena en la base de conocimientos, para luego ser consultada por el usuario en un buscador. También se manejan los datos de usuarios que tienen acceso al taller. El sistema de estadísticas devuelve al usuario los totales de reparaciones realizadas a un equipo determinado y los que han sido enviados a UCIXTEL en un intervalo de tiempo dado. También devuelve los datos de los reportes atendidos y enviados a COPEXTEL en un intervalo de tiempo dado y por un equipo determinado.

### 2.5. Propuesta de sistema.



Se propone, para cumplir con todo lo anteriormente expuesto, elaborar una aplicación Web que sea de fácil acceso, con una interfaz diseñada según los estándares predeterminados por la universidad para el diseño de sitios Web y el proyecto del Grupo de Asistencia Técnica.

Este sistema debe realizar cada funcionalidad que se le implemente con rapidez, cada dato debe ser gestionado en un tiempo de ejecución que será lo más óptimo posible. El sistema debe mostrar con claridad al usuario los resultados cada operación que se realice. Debe mostrar solo los datos que el usuario solicite. El sistema no debe entretener al usuario con mensajes ajenos a las operaciones del sistema, ni informaciones no relevantes para el mismo. El sistema debe mantener una conexión activa con UCIXTEL y los otros sistemas del grupo de asistencia técnica. El sistema mantiene un estricto control de cada usuario al permitir solamente el acceso del personal autorizado a interactuar con las funcionalidades de la aplicación. Se debe mostrar cada reporte llegado al sistema con su origen, tipo de equipo, estado, clave, fecha de realizado el reporte, y la descripción del problema. El sistema brinda la opción de incluir nuevos datos, como los arreglos realizados al equipo, registros en la base conocimiento y autorizar nuevos usuarios al sistema. Deben mostrarse bien claras las opciones de guardar, buscar, eliminar y actualizar, que son tareas fundamentales para gestionar los datos dentro del taller.

El sistema tiene que dar la opción de realizar estadísticas según los datos entrados por el usuario, estos datos son los intervalos de fechas de un tipo equipo seleccionado, la cantidad de reportes actualizados y enviados a UCIXTEL, así como el total. De cada estadística sacada se deben crear gráficos que muestren de forma visual y más atractiva la información mostrada al usuario.

La base de conocimientos debe ser una de las opciones de mayor importancia a la hora de interactuar con la aplicación Web. El técnico puede adicionar, actualizar y eliminar cada dato de la base de conocimientos y hacer una búsqueda de la información almacenada en la misma. El sistema debe presentar la opción de almacenar los datos de los equipos reparados en el taller, y poder mostrar los mismos cuando el usuario lo necesite.

Comparado con otras soluciones existentes para este sistema se tiene que la nueva aplicación cumple con cada requisito establecido por el cliente, presenta funcionalidades que facilitan el control del trabajo en el taller, presenta mayor interacción con el usuario, es más atractivo a la vista del usuario y presenta una mejor organización en cada opción que presenta el sistema.

### 2.6. Modelo de Negocio.

#### 2.6.1. Descripción del Negocio.

En la actualidad el grupo de asistencia técnica (GAT) ha modificado su estrategia de trabajo para la atención a los clientes que solicitan los servicios de este grupo. La brigada de atención a usuarios (BAU) realiza los arreglos de cada medio tecnológico que se reporta como roto o defectuoso. Existe además un área de recambio de donde se extraen equipos para cambiarlos por los defectuosos durante el tiempo de su reparación o permanentemente según sea la gravedad de la incidencia. Cuando la BAU no puede reparar un equipo solicita los servicios del taller de reparaciones que se encarga de reparar el equipo asignado, y actualiza el reporte del mismo con los nuevos datos que se le introducen.

También los equipos que salen del área de recambio si presentan algún desperfecto son enviados al taller con una solicitud de reporte asignada. Igualmente si el taller no dispone de los medios necesarios para darle solución al problema se pasa el reporte a COPEXTEL. Los reportes que llegan al taller tienen que ser actualizados siempre que se realice cualquier trabajo con el equipo asignado. Cada reporte actualizado debe ser almacenado en un historial que puede ser consultado posteriormente. Los responsables de gestionar las solicitudes de los reportes que llegan al taller son los técnicos del mismo. Se dispone de una base de conocimientos que permite guiar al técnico en su trabajo. Esta base de conocimientos tiene las posibles soluciones a cada rotura o desperfecto de los posibles equipos que llegan al taller. Se pretende desarrollar una aplicación que posibilite a los técnicos del taller gestionar cada tarea de actualización de los reportes y enviar los mismos a UCIXTEL, en caso de no poder repararlos, o al área de procedencia de cada reporte una vez terminada la reparación, la actualización del reporte, su almacenamiento en el historial del taller, y también generar estadísticas de cada reporte almacenado.

#### 2.6.2. Actores y Trabajadores del Negocio.

Un actor del negocio es cualquier individuo, grupo, organización, compañía o máquina que actúan recíprocamente con el negocio y desempeña un rol determinado.

Un trabajador del negocio representa a personas o sistemas dentro del negocio que son los que realizan las actividades que están comprendidas dentro de un caso de uso [16].

A continuación se representan en la siguiente tabla los roles identificados para este negocio y su justificación atendiendo a los procesos con los cuales interactúan.

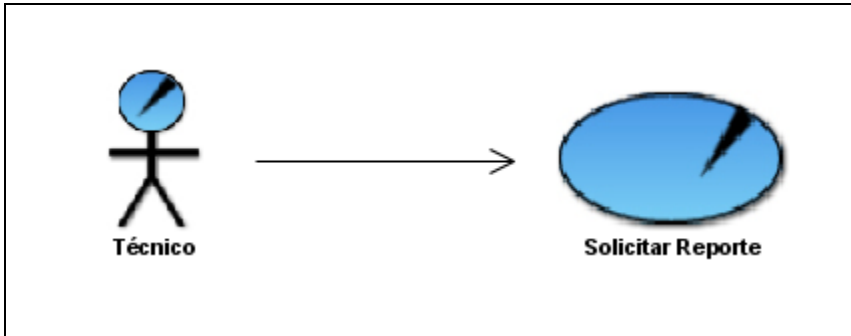
*Tabla 2.1. Actores Del Negocio*

<b>Actor</b>	<b>Descripción</b>
Cliente	Personal que solicita la gestión del reporte al taller. Pueden ser los técnicos del BAU, trabajadores del área de recambio o cualquier directivo del GAT que necesite esta información.

*Tabla 2.2. Trabajadores Del Negocio*

<b>Trabajador</b>	<b>Descripción</b>
Técnico	Persona encargada de recibir los reportes, actualizarlos, almacenarlos y enviarlos a las aéreas correspondientes. Es el encargado además de reparar los equipos defectuosos que llegan al taller.

### 2.6.3. Diagrama de Casos de Uso del Negocio.



*Figura 2.1. Diagrama de Caso de Uso del Negocio.*

### 2.6.4. Descripción del Caso de Uso del Negocio y Reglas del Negocio. Ver Anexo 1.

### 2.6.5. Diagrama de Actividades. Ver Anexo 2.

### 2.6.5. Modelo de Objetos. Ver Anexo 3.

### 2.7. Especificación de los Requisitos de Software.

Entre los requerimientos funcionales del sistema tenemos:

- R1.Gestionar Base de Conocimientos.
  - R1.1. Adicionar Datos.
  - R1.2. Actualizar Datos.
  - R1.3. Eliminar Datos.
  
- R2.Mostrar los datos de la base de conocimientos.
  - R2.1. Buscar Equipo.
  - R2.2. Mostrar Equipo.
  - R2.3. Mostrar Incidencia.

- R2.4. Mostrar Solución.
- R3. Gestionar usuario.
  - R3.1. Adicionar Datos del Usuario.
  - R3.2. Mostrar Datos del Usuario.
  - R3.3. Eliminar Datos del Usuario.
  - R3.4. Actualizar Datos del Usuario.
- R4. Gestionar Reportes.
  - R4.1. Actualizar los reportes.
  - R4.2. Enviar los reportes actualizados.
  - R4.3. Almacenar los reportes en el Historial.
  - R4.4. Mostrar los reportes.
- R5. Generar Estadísticas.
  - R5.1. Mostrar Intervalo de Fechas.
  - R5.2. Mostrar Cantidad de Reportes Actualizados.
  - R5.3. Mostrar Cantidad de Reportes enviados a COPEXTEL.
  - R5.4. Mostrar Total.
  - R5.5. Seleccionar Tipo de Equipo.
  - R5.6. Generar gráficos de estadísticas.
- R6. Autenticar Usuario.

Entre los requerimientos no funcionales, o sea, características o capacidades del sistema, se tiene los siguientes:

- *Requerimientos de apariencia o interfaz externa*

Interfaz de usuario amigable y fácil de entender sin mucho entrenamiento por parte del cliente. Colores suaves e información legible que evite cualquier pérdida por parte de un usuario sin conocimiento de la aplicación.

- *Requerimientos de Usabilidad*

Consistencia en la interfaz de usuario, Documentación de usuario.

- *Requerimientos de Rendimiento*

Sistema rápido y eficiente en la búsqueda de datos. Los envíos de reportes y de respuestas del sistema deben ser rápidos y exactos.

- *Requerimientos de Soporte*

Es fácil de mantener por los desarrolladores del proyecto, ya que el sistema está desarrollado sobre herramientas y tecnologías de software libre y un lenguaje de programación de código abierto.

- *Requerimientos de Portabilidad*

Fácil de migrar al sistema operativo Linux.

- *Requerimientos de Seguridad*

Confidencialidad: Asignar permisos limitados a usuarios teniendo en cuenta el nivel de acceso a la aplicación y al trabajo a realizar en ella.

Integridad: La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes.

Disponibilidad: Significa que los usuarios autorizados se les garantizará el acceso a la información y que los dispositivos o mecanismos utilizados para lograr la seguridad no ocultarán o retrasarán a los usuarios para obtener los datos deseados en un momento dado.

- *Requerimientos de Software.*

Sistema Operativo Windows XP o Superior, Sistema UNIX (Linux).

- *Requerimientos de Hardware.*

PC o Servidor con microprocesador Pentium III o superior, memoria RAM 256 Mb o superior.

Memoria física superior a 500 Mb.

## 2.8. Definición de los Casos de Uso del Sistema.

### 2.8.1. Actores del sistema.

*Tabla 2.3. Actores del sistema.*

Actores	Justificación
Técnico	Persona con acceso a interactuar con todas las funcionalidades del sistema.
Administrador	Técnico con permiso de agregar o quitar privilegios de usuarios del sistema.

2.8.2. Diagrama de Casos de Uso del Sistema.

Un diagrama de casos de uso del sistema representa gráficamente a los procesos y su interacción con los actores.

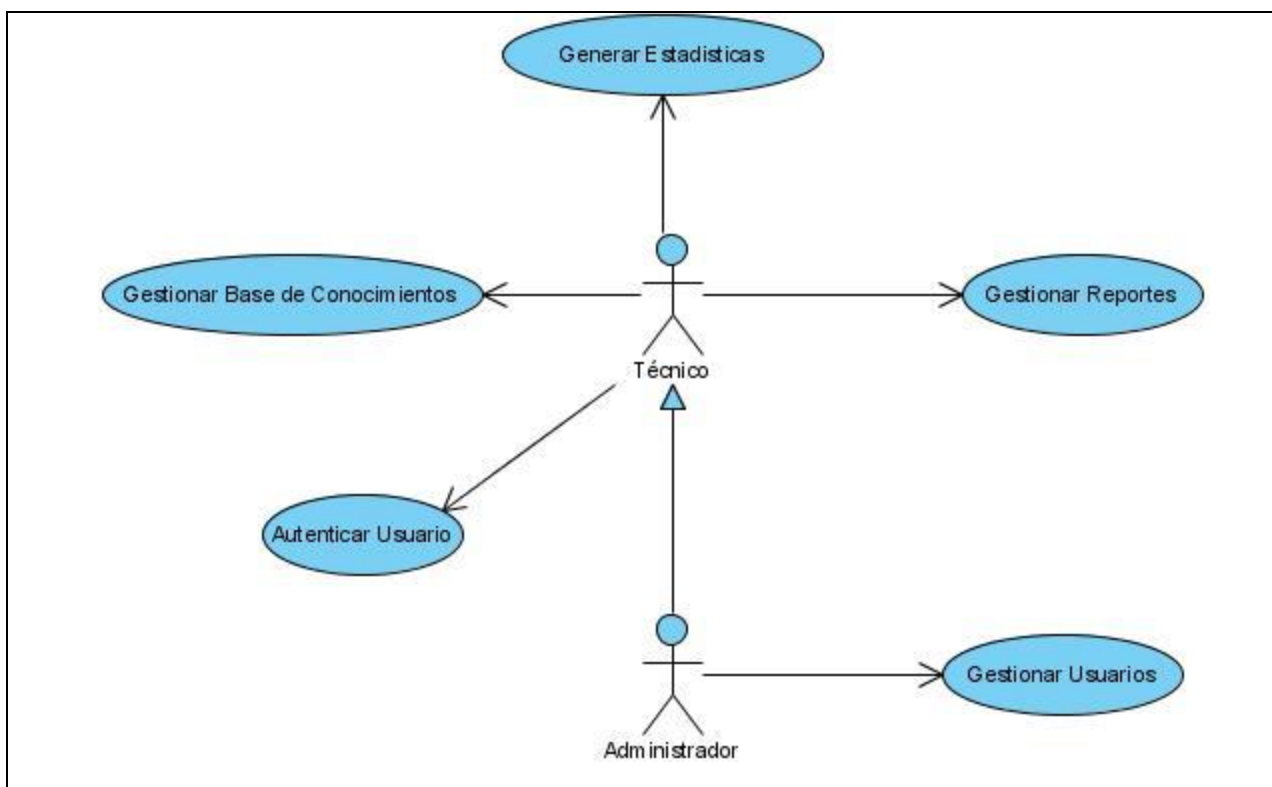


Figura 2.2. Diagrama de Casos de Uso del Sistema.

2.8.3. Descripción de los Casos de Uso del Sistema. Ver Anexo 4.

### 2.9. Conclusiones.

En este capítulo se ha conocido de forma general las características generales del sistema informático a desarrollar, una descripción detallada del proceso de negocio del mismo y los casos de uso a implementar. Además se pudo conocer los diversos requerimientos que dan funcionalidad a esta aplicación Web.

Es de vital importancia mantener bien establecidos los requerimientos funcionales y cada caso de uso del sistema, para luego realizar un correcto análisis y diseño del mismo, poniendo en práctica lo establecido en este capítulo. A partir de ellos se procede entonces a la implementación de la aplicación.



## CAPÍTULO 3. ANÁLISIS Y DISEÑO DEL SISTEMA

### 3.1. Introducción.

En este capítulo se conocerán los procesos de análisis y diseño desarrollados para este sistema y los modelos de datos que se utilizan para el desarrollo del mismo. Además se hará referencia a los diversos diagramas de interacción que explican en detalle cada evento por caso de uso y dan una comprensión mas detallada del funcionamiento del sistema. De cada diagrama de clases se podrá obtener la lógica funcional del sistema, que luego posibilite la implementación del sistema, adaptada a las condiciones del framework empleado y manteniendo una correcta conexión a la base de datos.

### 3.2. Análisis.

El análisis consiste en obtener una visión del sistema que se preocupa por ver lo que hace el mismo de modo que sólo se interesa por los requisitos funcionales, transformar los requisitos funcionales en un diseño de clases viendo las relaciones e interacción que existe entre ellos, y teniendo en cuenta en el proceso una arquitectura robusta que permita adaptar el sistema al entorno de implementación que se está desarrollando.

Una de las principales salidas o artefactos del análisis es el Diagrama de Clases de Análisis. En este se representan las Clases de Análisis (clase Interfaz, clase controlador y clases entidad), sus relaciones entre sí y con los actores del sistema [16].

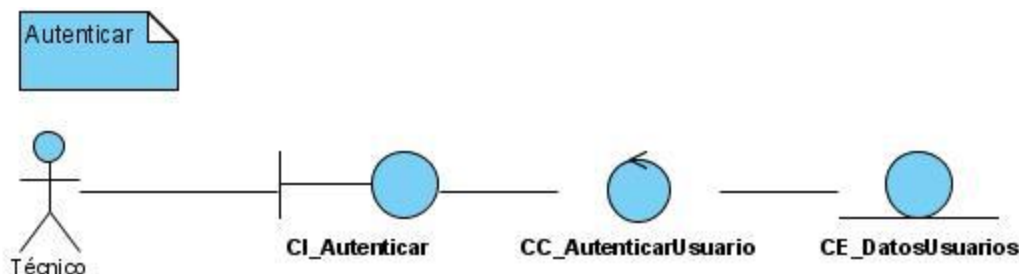


Figura 3.1. Diagrama de Clases del Análisis del CUS Autenticar.

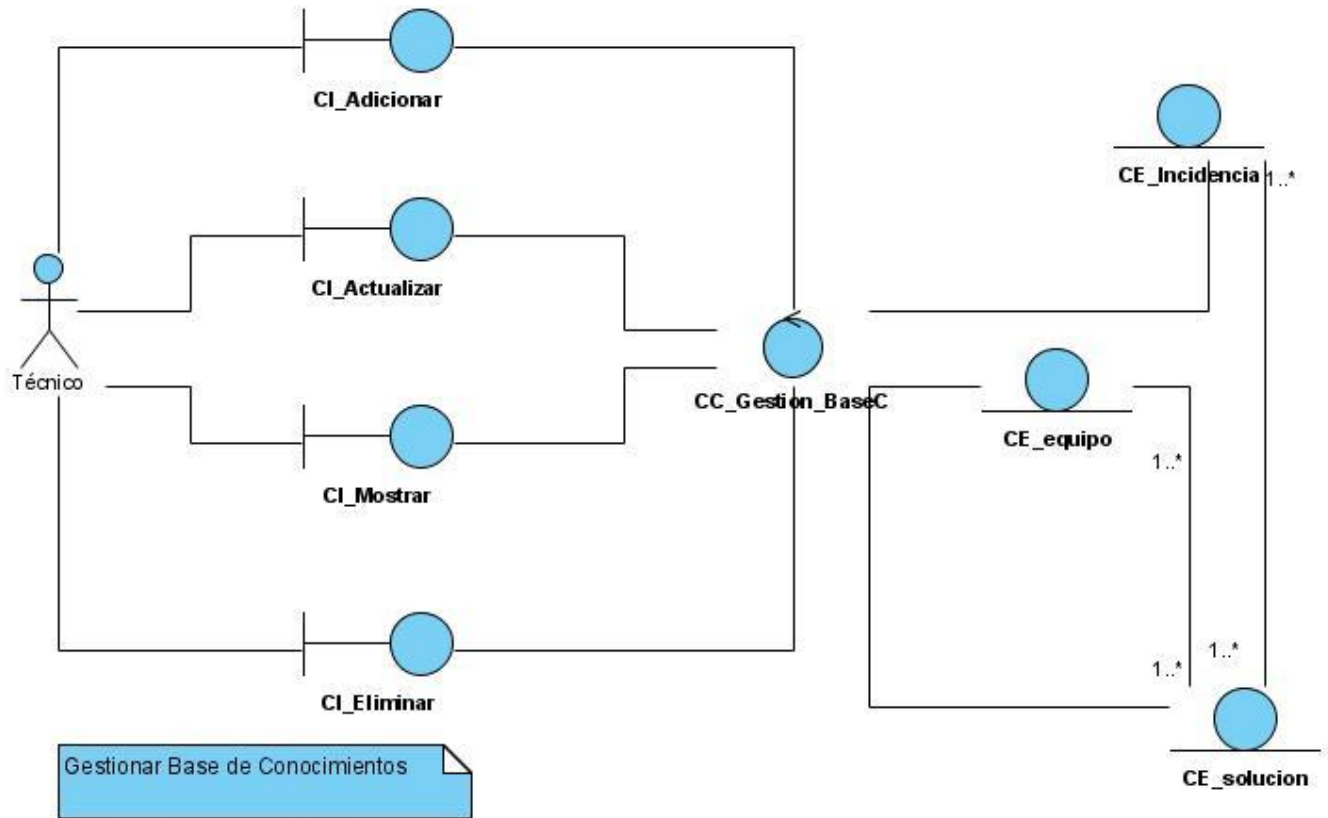


Figura 3.2. Diagrama de Clases del Análisis del CUS Gestionar Base de Conocimientos.

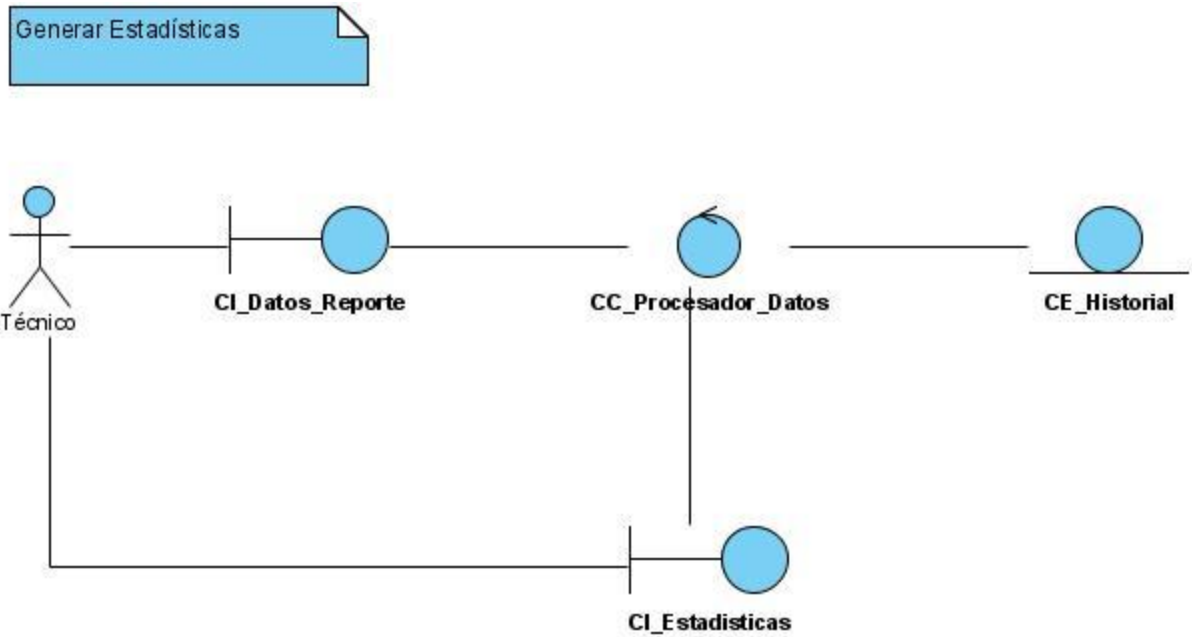


Figura 3.3. Diagrama de Clases del Análisis del CUS Generar Estadísticas.

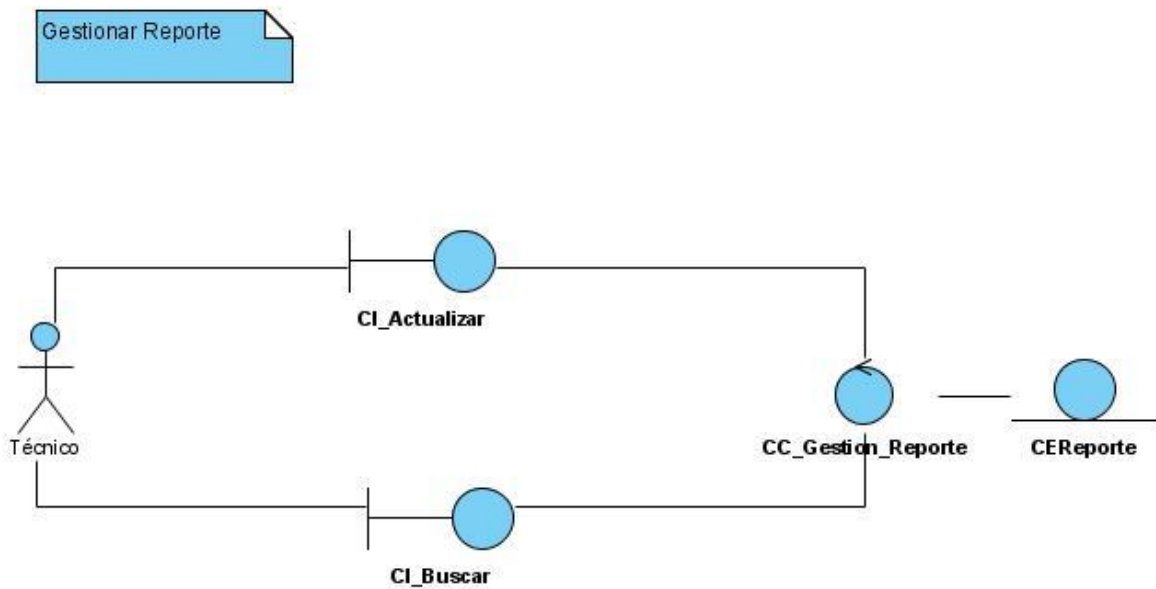


Figura 3.4. Diagrama de Clases del Análisis del CUS Gestionar Reporte.

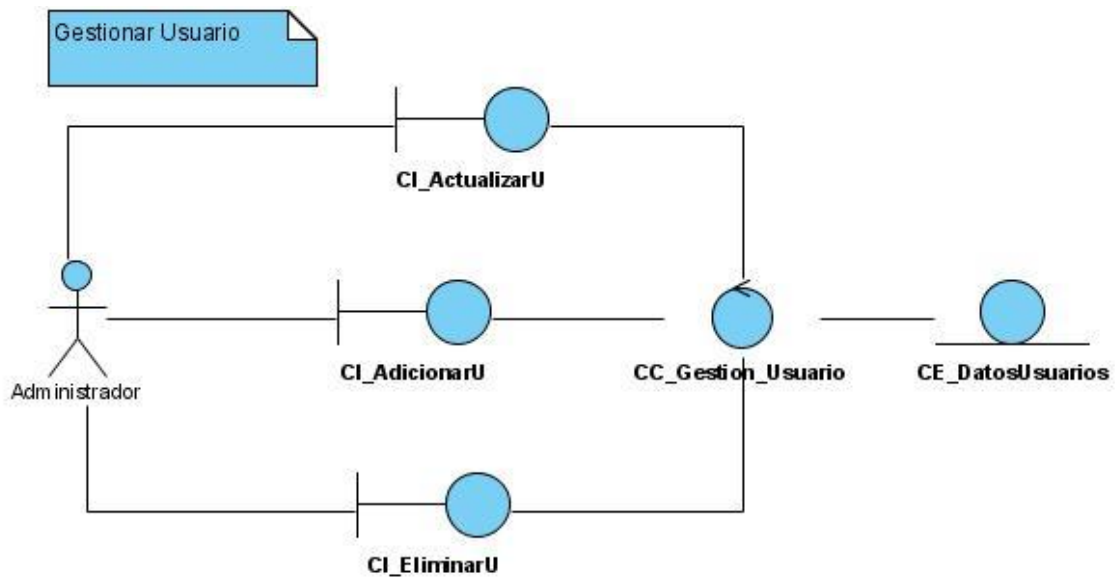


Figura 3.5. Diagrama de Clases del Análisis del CUS Gestionar Usuario.

### 3.3 Diseño.

Con el diseño se logra la comprensión de los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, Sistemas Operativos, tecnologías de distribución y concurrencia y tecnologías de interfaz de usuario y se crea una entrada apropiada y un punto de partida para actividades de implementación, capturando los requisitos o subsistemas individuales, interfaces y clases [16].

Una de las principales salidas del diseño lo constituyen el Diagrama de Clases del Diseño, el Diagrama de Clases Persistentes, el Modelo de Datos y el Diagrama de Despliegue, los cuales se podrán conocer en este capítulo.

3.3.1 Diagramas de interacción. Ver anexo 5.

3.3.2 Diagrama de clases del diseño.

Un Diagrama de Clases muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema, esto incluye modelar el vocabulario del sistema, modelar las colaboraciones o modelar esquemas. Los diagramas de clases también son la base para un par de diagramas relacionados: los diagramas de componentes y los diagramas de despliegue. Los diagramas de clases son importantes no sólo para visualizar, especificar y documentar modelos estructurales, sino también para construir sistemas ejecutables, aplicando ingeniería directa e inversa. Además se pone en práctica el uso del patrón Modelo-Vista-Controlador representando cada clase según su función, desde las vistas con sus respectivos estereotipos Web, hasta las clases controlados y las pertenecientes al modelo con su acceso a datos.

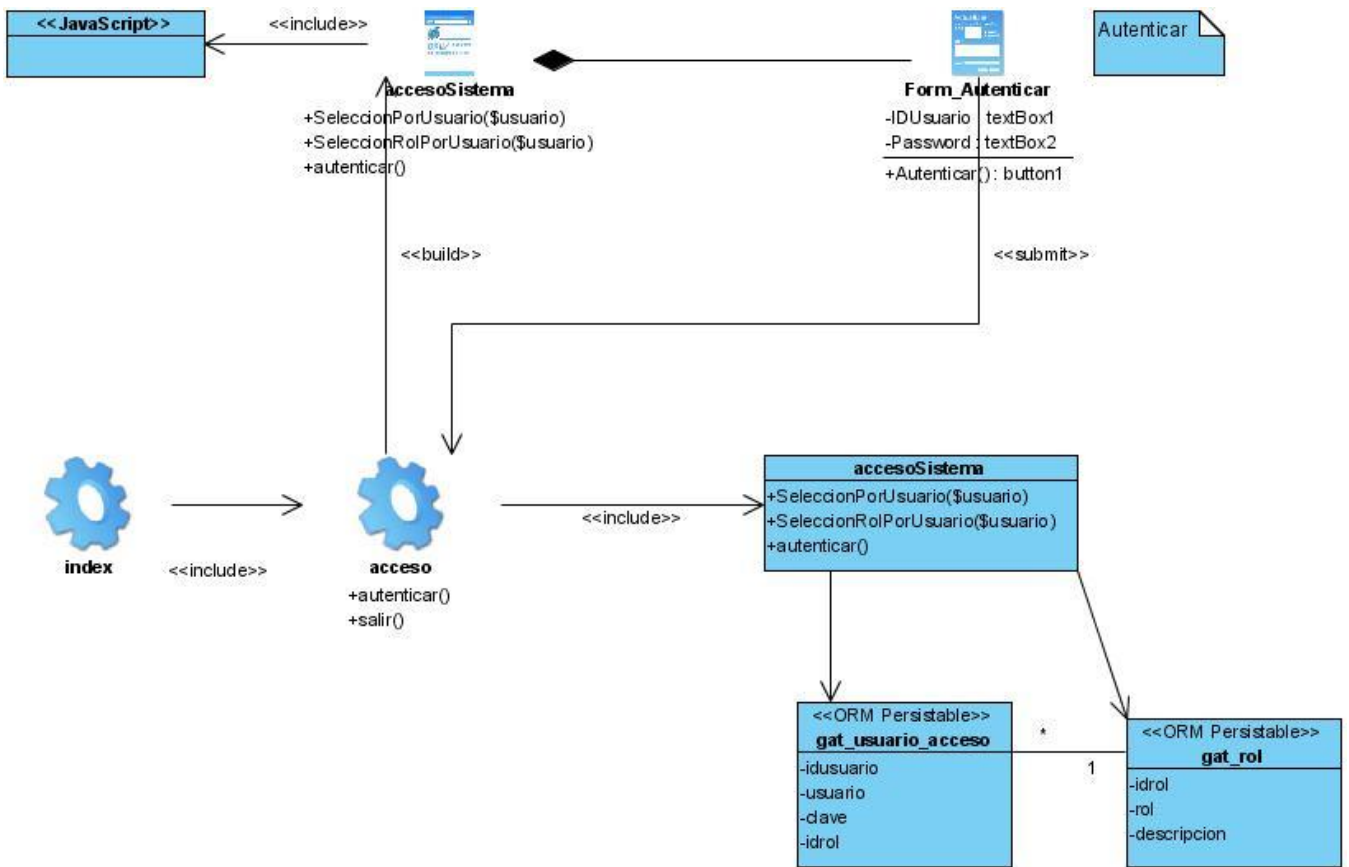


Figura 3.6. Diagrama de Clases del Diseño del CUS Autenticar.

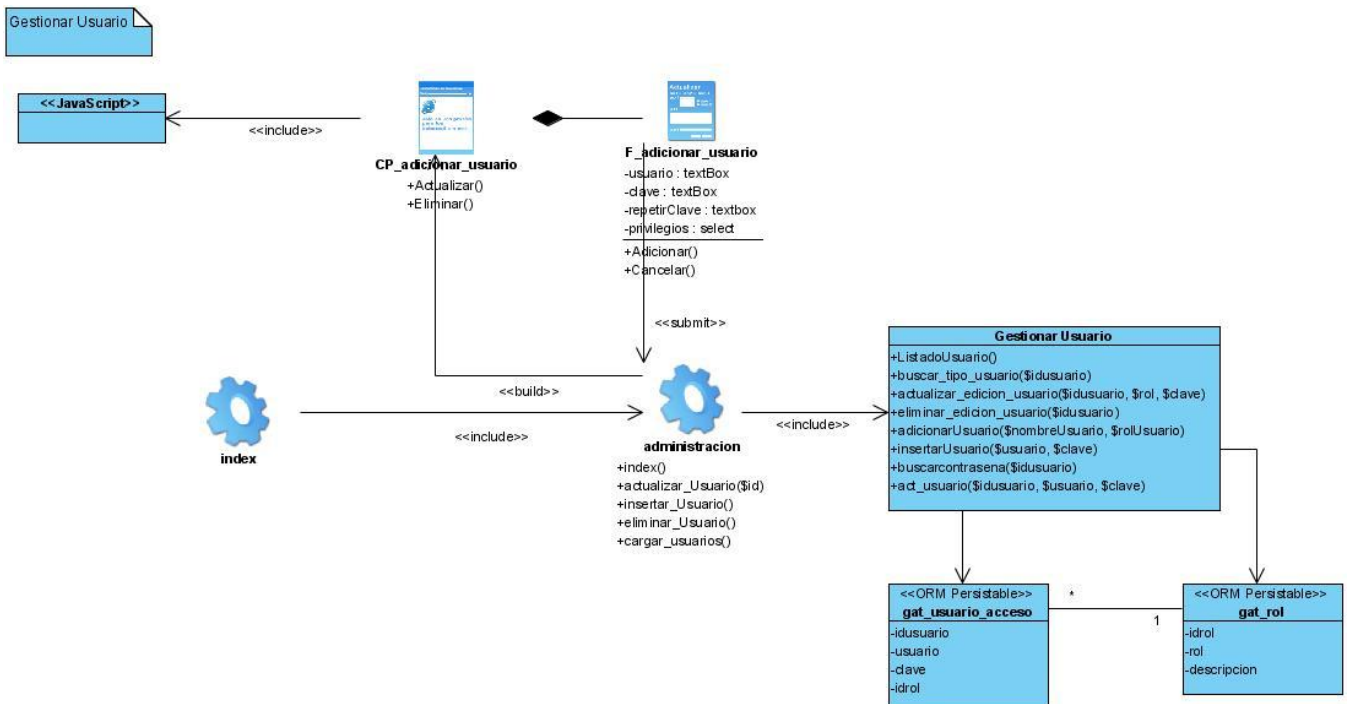


Figura 3.7. Diagrama de Clases del Diseño del CUS Gestionar Usuario.

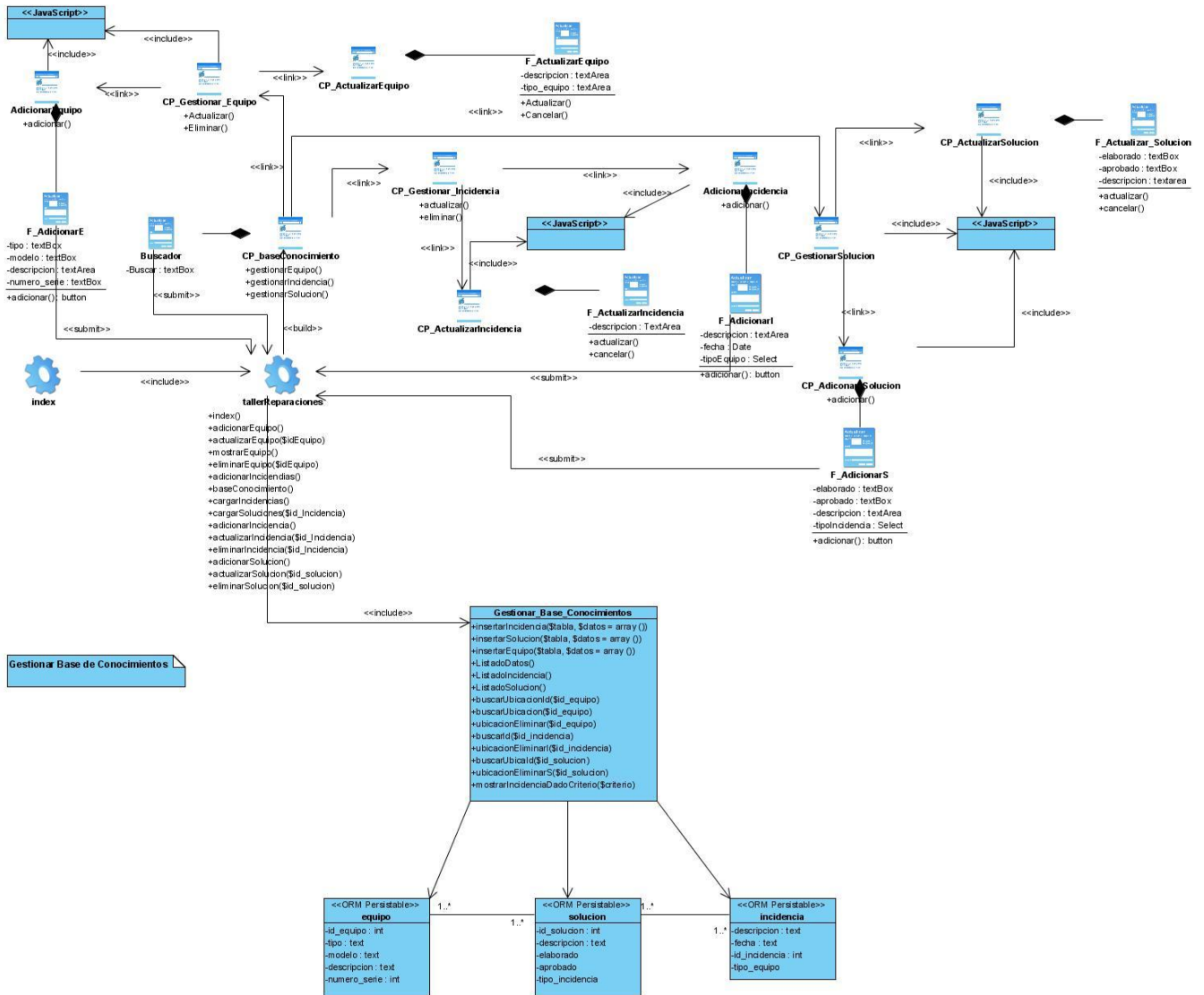


Figura 3.8. Diagrama de Clases del Diseño del CUS Gestionar Base de Conocimiento.



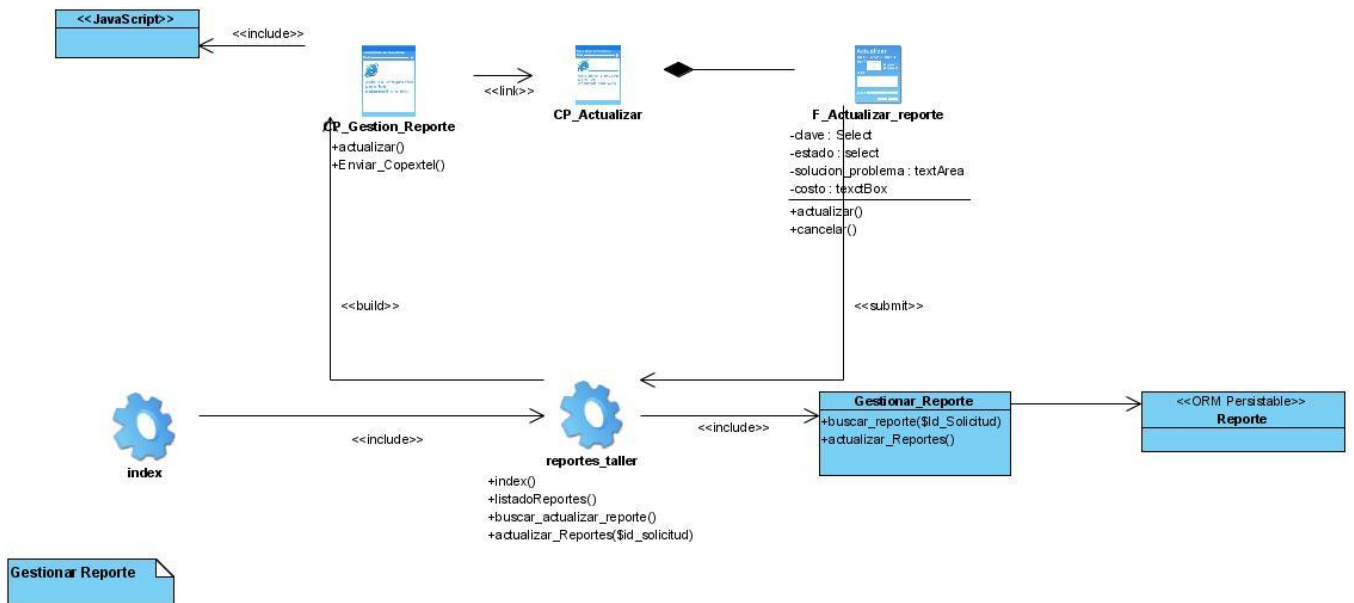


Figura 3.9. Diagrama de Clases del Diseño del CUS Gestionar Reporte.

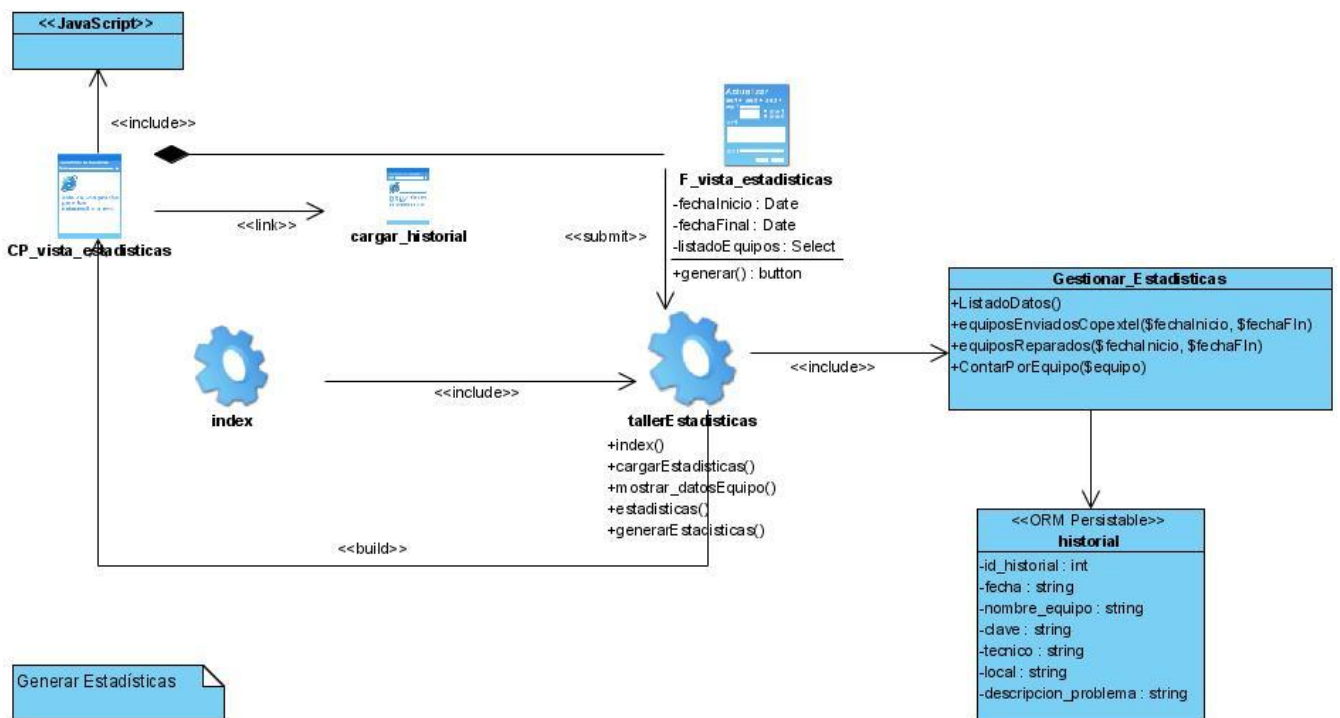


Figura 3.10. Diagrama de Clases del Diseño del CUS Generar Estadísticas.

### 3.4. Diseño de la base de datos.

En este sistema se emplean tres modelos de datos que posibilitan el desarrollo de cada funcionalidad, dando cumplimiento a los requerimientos funcionales del sistema.

- El primer modelo es la base de datos de usuarios, que registra los datos de los usuarios autenticados en el sistema y el rol correspondiente que restringe el acceso a otros módulos del proyecto.
- El segundo modelo es la base de conocimientos que registra los campos correspondientes a equipos, incidencias y soluciones que se dan en el taller.
- El tercer y último modelo en presentar representa una tabla que almacena el historial de los reportes tratados en el taller, que almacenan toda la información del mismo y sobre las cuales se pueden realizar estadísticas posteriormente.

Los procesos de actualización de reportes se realizan sobre los datos de las solicitudes que se envían al taller desde el Sistema de Gestión Tecnológica de la Información que es el sistema encargado de gestionar las solicitudes de cada equipo roto o defectuoso y emitir los reportes al taller de reparaciones.

#### 3.4.1. Modelo lógico de datos (diagrama de clases persistentes).

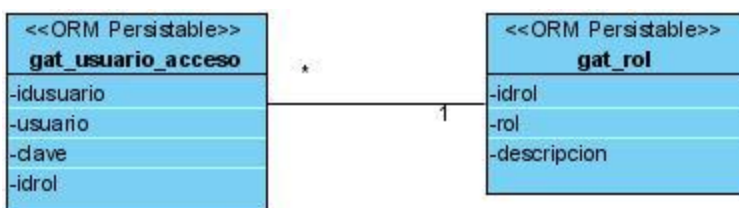


Figura 3.11. Diagrama de Clases Persistentes (Datos Usuario).

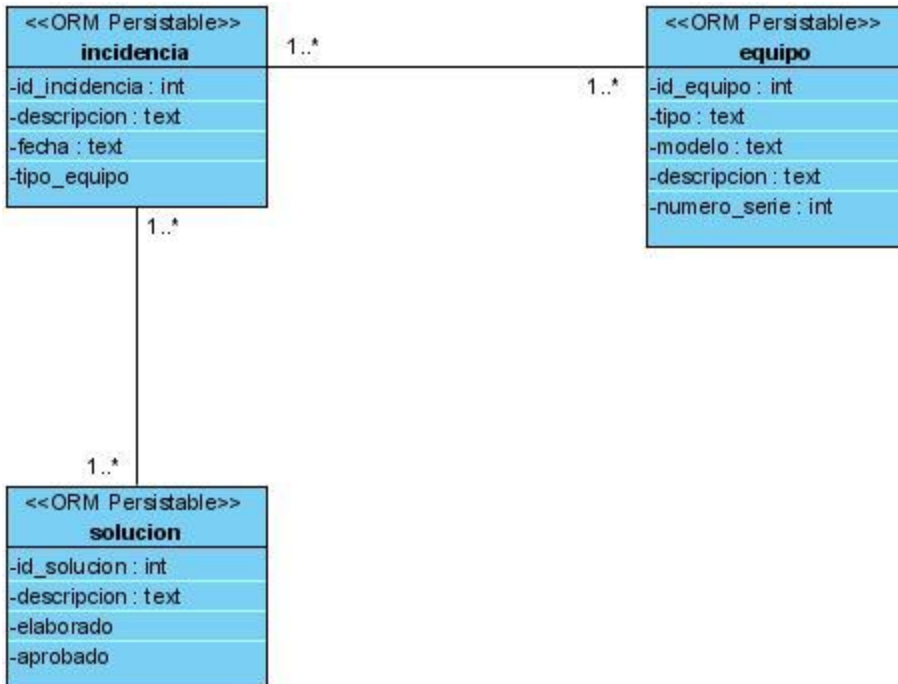


Figura 3.12. Diagrama de Clases Persistentes (Base de Conocimientos).

3.4.2. Modelo físico de datos (modelo de datos).

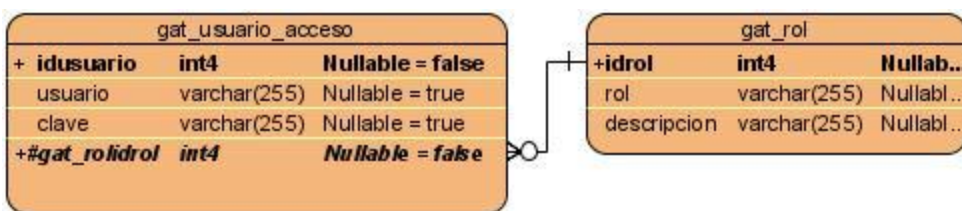


Figura 3.13. Modelo de Datos (DatosUsuario).

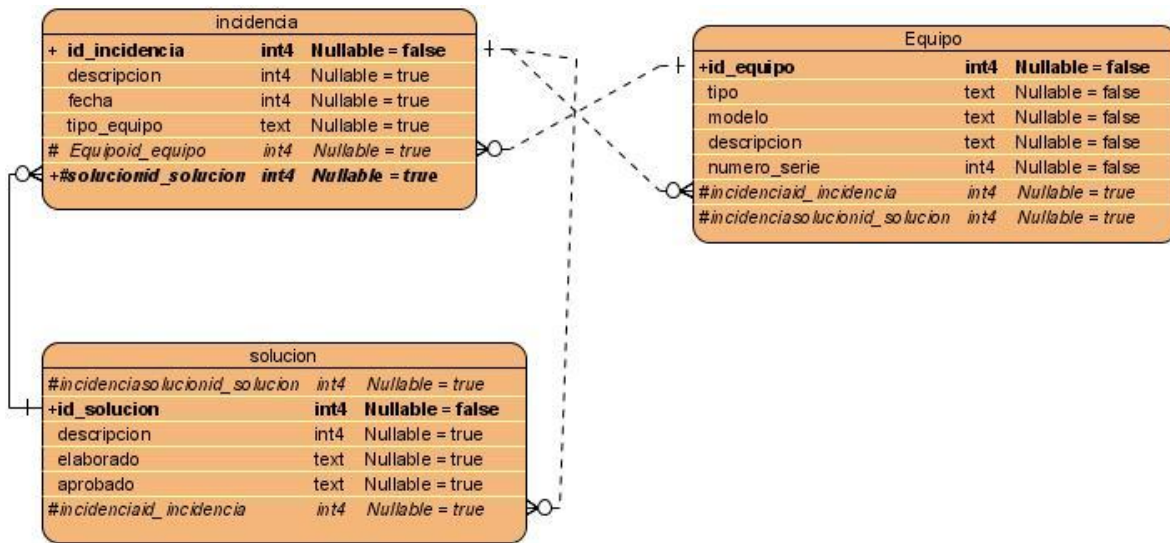


Figura 3.14. Modelo de Datos (Base de Conocimientos).

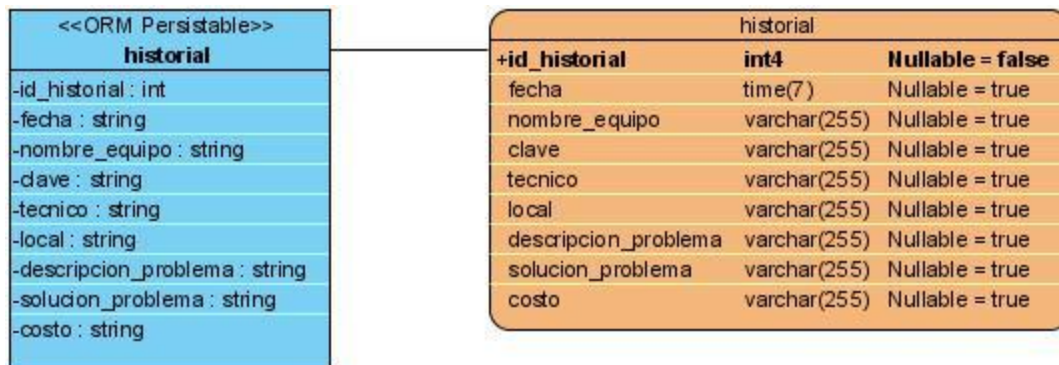


Figura 3.14. Modelo lógico y de datos correspondientes a la tabla Historial.

### 3.5. Conclusiones.

En este capítulo se pudo conocer los artefactos generados en el flujo de análisis y diseño que permiten observar en detalle las distintas características del sistema y las ideas bases de una futura implementación. Además se tuvo conocimiento de los distintos modelos de datos que se emplean y los detalles de cada uno.

Han sido definidas las clases de análisis y las clases del diseño así como sus relaciones en el Diagrama de Clases del Diseño. Se definió el Diagrama de Clases Persistentes, y a través de este, el Modelo de Datos.

## CAPÍTULO 4. IMPLEMENTACIÓN

### 4.1. Introducción.

En este capítulo se conocerán los distintos diagramas de implementación que representan los elementos físicos del sistema y el despliegue realizado por el proyecto.

### 4.2 Diagrama de despliegue.

Un Diagrama de Despliegue es un diagrama que muestra la configuración de los nodos que participan en la ejecución. Se modelan la topología del hardware sobre el que se ejecuta el sistema y la distribución física del sistema.

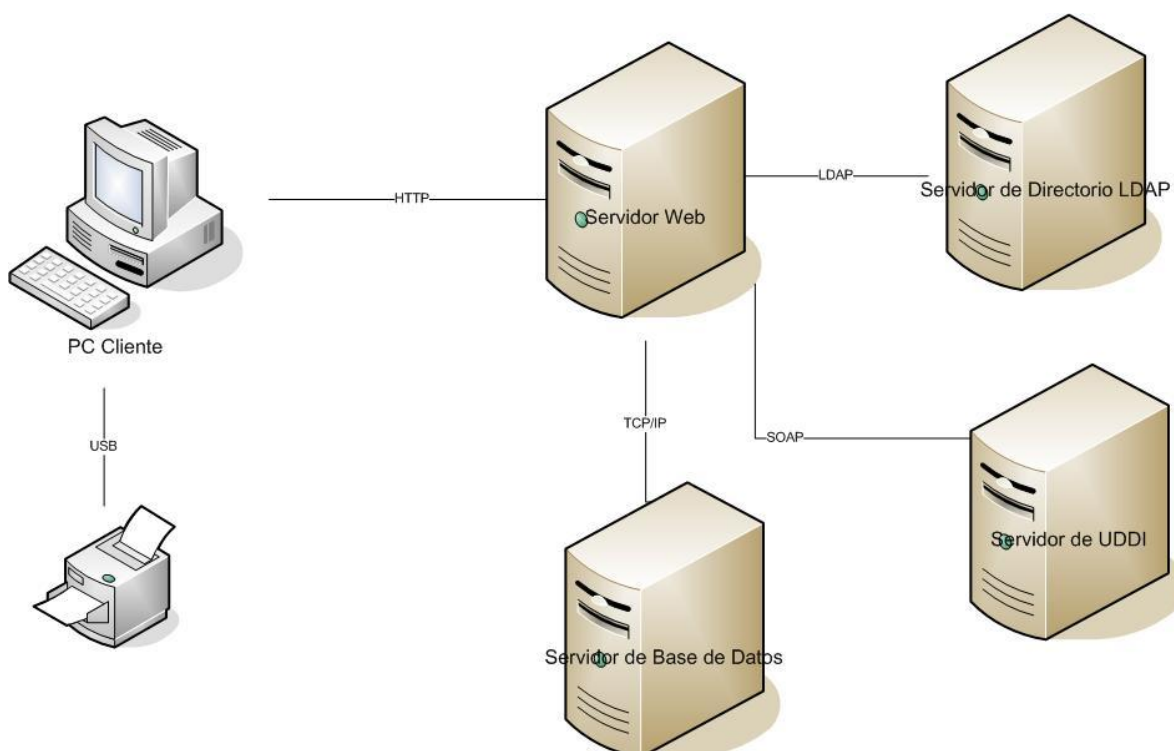


Figura 4.1. Representación del despliegue.

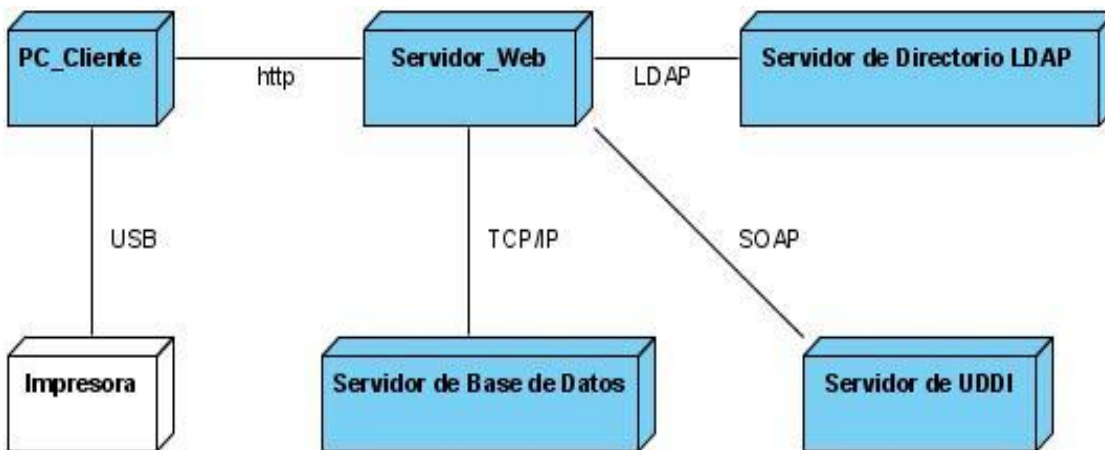


Figura 4.2. Diagrama de Despliegue según UML.

### 4.3. Diagrama de componentes.

El diagrama de componentes describe los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable.

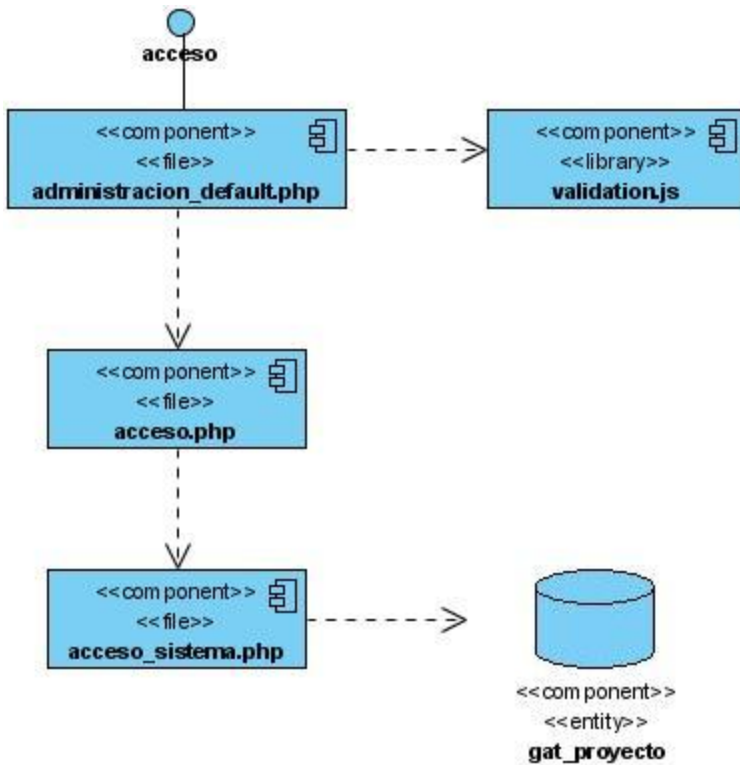


Figura 4.3. Modelo de Componente. Caso de Uso Autenticar.

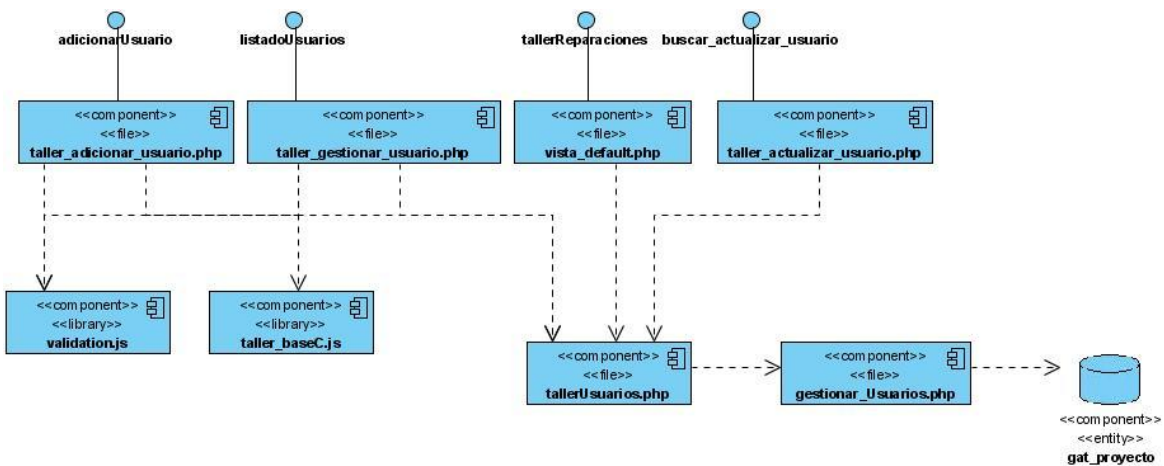


Figura 4.4. Modelo de Componente. Caso de Uso Gestionar Usuario.



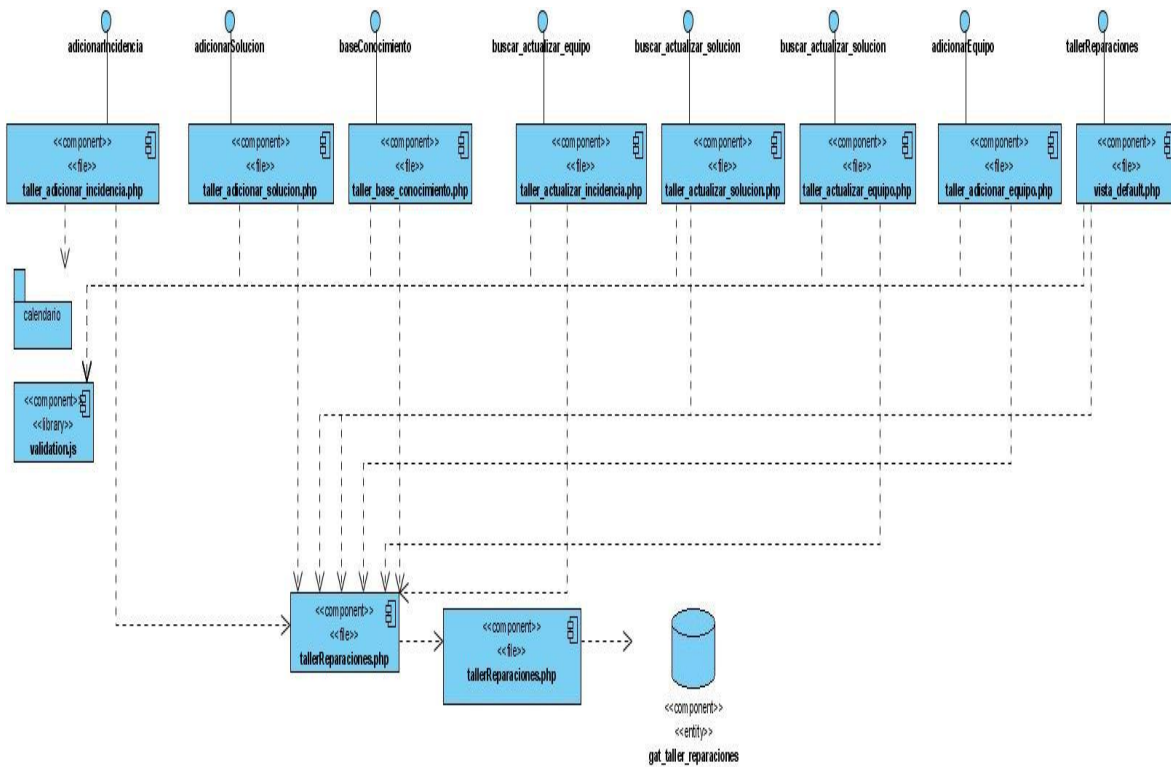


Figura 4.5. Modelo de Componente. Caso de Uso Gestionar Base de Conocimiento.

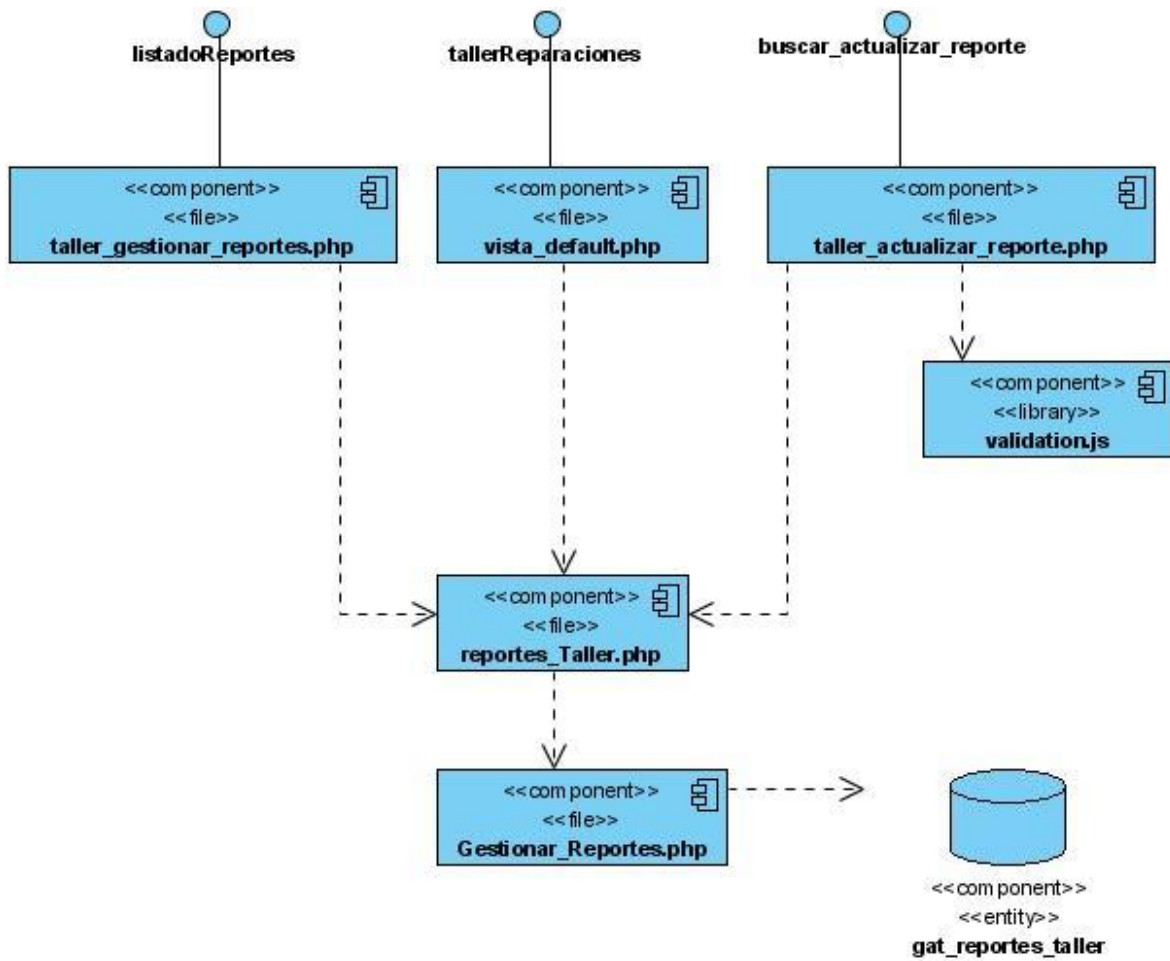


Figura 4.6. Modelo de Componente. Caso de Uso Gestionar Reporte.

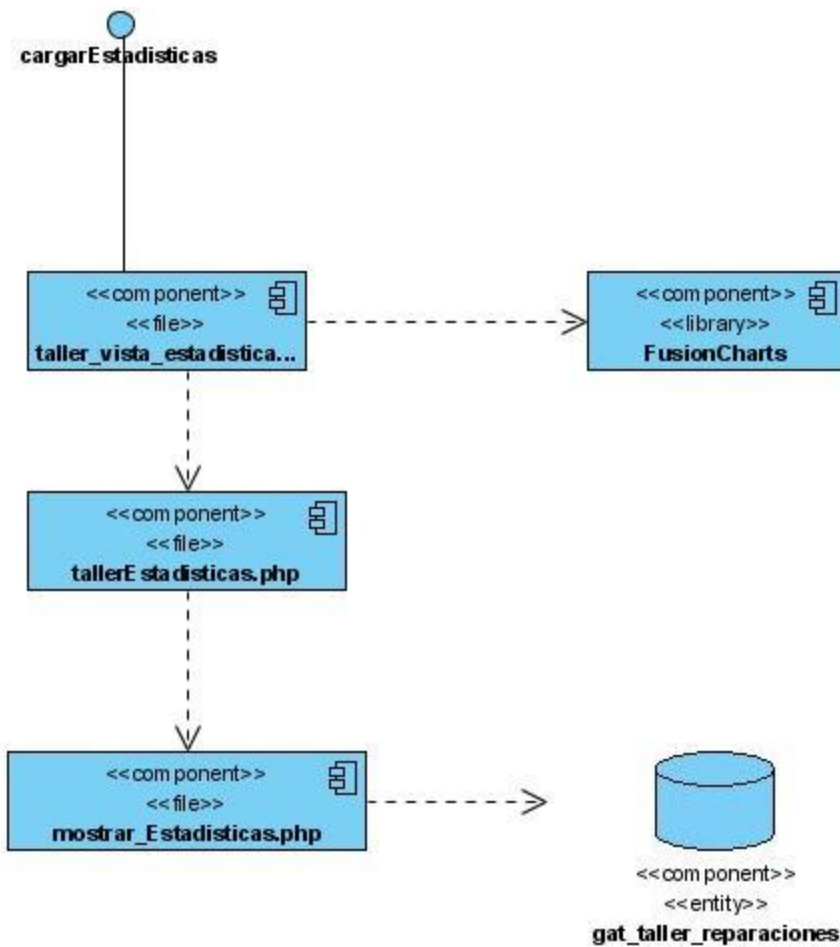


Figura 4.7. Modelo de Componente. Caso de Uso Generar Estadísticas.

#### 4.4. Conclusiones.

En este capítulo se pudieron observar los distintos artefactos creados en el flujo de trabajo de implementación que ofrece la metodología RUP. Estos diagramas ofrecen una idea lógica de cada componente a implementar en el sistema, y que mantiene el orden representado en cada diagrama de clases del diseño, y da solución a los requisitos del sistema.

## CAPITULO 5. ESTUDIO DE FACTIBILIDAD.

### 5.1 Introducción

Este capítulo presenta el método estimación aplicado a este sistema, ya que el análisis del costo, el esfuerzo y los beneficios, es de suma importancia durante el desarrollo de una aplicación. Se realiza el estudio de factibilidad mediante el análisis de Puntos de Casos de Uso, método de estimación de tiempo de desarrollo del proyecto, a partir de las características de sus requisitos, expresados en los casos de uso propuestos.

### 5.2 Método de estimación basado en casos de usos.

El análisis de Puntos de Casos de Uso es una de las alternativas posibles para la estimación del esfuerzo en proyectos basados en Casos de Uso, se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.

Paso 1. Cálculo de Puntos de Casos de Uso sin ajustar.

$$UUCP = UAW + UUCW$$

$$UUCP = 9+25=34$$

Donde:

UUCP: Puntos de Casos de Uso sin ajustar.

UAW: Factor de Peso de los Actores sin ajustar.

UUCW: Factor de Peso de los Casos de Uso sin ajustar.

*Tabla 5.1: Factor de Peso de los Actores sin ajustar.*

Tipo de actor	Descripción	Factor de peso	Actores	Total
Simple	Sistema con sistema a través de interfaz de programación.	1	1	1
Medio	Sistema con sistema mediante protocolo de interfaz basada en texto.	2	1	2
Complejo	Persona que interactúa con el sistema	3	2	6

	mediante interfaz gráfica.			
$UAW = \sum (\text{Factor} * \text{Actores})$		UAW	9	

$$UAW = \sum (\text{cant actores} * \text{peso})$$

$$UAW = 9$$

Tabla 5.2: Factor de Peso de los Casos de Uso sin ajustar.

Tipo de CU	Descripción	Peso	Cantidad de CU	Total
Simple	El caso de uso tiene de 1 a 3 transacciones.	5	4	20
Medio	El caso de uso tiene de 4 a 7 transacciones.	10	0	0
Complejo	El caso de uso tiene más de 8 transacciones.	15	1	15

$$UUCW = \sum (\text{cant CU} * \text{peso})$$

$$UUCW = 35$$

Paso 2. Cálculo de Puntos de Casos de Uso ajustados.

$$UCP = UUCP * TCF * EF$$

$$UCP = 19 * 1.03 * 0.41 = 8.0237$$

Donde:

UCP: Puntos de Casos de Uso ajustados.

UUCP: Puntos de Casos de Uso sin ajustar.

TCF: Factor de complejidad técnica.

EF: Factor de ambiente.

El factor de complejidad técnica (TCF) se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada factor se cuantifica en un valor desde 0 (aporte irrelevante) hasta 5 (aporte muy relevante).

Tabla 5.3: Factor de Complejidad Técnica.

Factor	Descripción	Peso	Valor asignado	Total
T1	Sistema distribuido.	2	0	0
T2	Tiempo de respuesta.	1	1	1
T3	Eficiencia del usuario final.	1	1	1
T4	Procesamiento interno complejo.	1	1	1
T5	El código debe ser reutilizable.	3	3	9
T6	Facilidad de instalación.	1	0	0
T7	Facilidad de uso.	3	4	12
T8	Portabilidad	2	1	2
T9	Facilidad de cambio.	3	2	6
T10	Concurrencia.	2	4	8
T11	Incluye objetivos especiales de seguridad.	3	3	9
T12	Provee acceso directo a terceras partes	2	5	10
T13	Se requieren facilidades especiales de entrenamiento de usuarios	1	1	1

$$TCF = 0.6 + 0.01 * \sum (\text{peso} * \text{valor asignado})$$

$$TCF = 0.61 + 0.01 * 60 = 1.2$$

El factor de ambiente (EF) está relacionado con las habilidades y entrenamiento del grupo de desarrollo que realiza el sistema. Cada factor se cuantifica con un valor desde 0 (aporte irrelevante) hasta 5 (aporte muy relevante).

Tabla 5.4: Factor de Ambiente.

Factor	Descripción	Peso	Valor asignado	Total
E1	Familiaridad con el modelo de proyecto utilizado.	1	3	3
E2	Experiencia en la aplicación.	2	2	4
E3	Experiencia en la orientación a objetivos.	2	5	10

E4	Capacidad del analista líder.	1	5	5
E5	Motivación.	1	5	5
E6	Estabilidad de requerimientos.	2	3	6
E7	Personal Part-Time.	1	0	0
E8	Dificultad del lenguaje de programación.	1	0	0

$$EF = 1.4 - 0.03 * \sum (\text{peso} * \text{valor asignado})$$

$$EF = 1.4 - 0.03 * 33 = 0.41$$

Paso 3. Estimación del esfuerzo.

$$E = UCP * CF$$

$$E = 21.648 * 20$$

$$E = 432.96 \text{ Horas-Hombres}$$

Donde:

E: Esfuerzo estimado en horas-hombre.

UCP: Puntos de Casos de Uso ajustados.

CF: Factor de conversión.

Para obtener el factor de conversión (CF) se cuentan cuantos valores de los que afectan el factor ambiente (E1...E6) están por debajo de la media (3), y los que están por arriba de la media para los restantes (E7, E8). Si el total es 2 o menos se utiliza el factor de conversión 20 Horas-Hombre / Punto de Casos de Uso. Si el total es 3 o 4 se utiliza el factor de conversión 28 Horas-Hombre / Punto de Casos de uso. Si el total es mayor o igual que 5 se recomienda efectuar cambios en el proyecto ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

En este caso se puede decir que:

$$CF = 20 \text{ Horas-Hombre}$$

Paso 4. Calcular esfuerzo total del proyecto.

*Tabla 5.5: Esfuerzo del proyecto.*

Actividad	Porcentaje %	Horas- Hombres
-----------	-----------------	-------------------

Análisis	10	108.24
Diseño	20	216.48
Implementación	40	432.96
Pruebas	15	162.36
Sobrecarga (otras actividades)	15	162.36
Total	100	1082.4

Tabla 5.6: Resumen.

<b>Esfuerzo Total (Horas--Hombre)</b>	ET1	1082.4
<b>Esfuerzo Total (Mes--Hombre)</b>	ET2	4.51
<b>Salario Promedio</b>	SM	198
<b>Cantidad de Hombres</b>	CH	1
<b>Costo Hombre--Mes</b>	CHM	198
<b>Costo Total</b>	Costo	892.98

### 5.3. Beneficios Tangibles e Intangibles.

El Sistema para la automatización del taller de reparaciones de la UCI no es un producto utilizado con fines comerciales u otros intereses similares, ha sido desarrollado para facilitar el trabajo de gestión de cada solicitud de reportes que llegan al taller de reparaciones del Grupo de Asistencia Técnica de la UCI y mantener un control de cada reparación realizada y mostrar sus estadísticas. El beneficio principal que reporta el sistema es contar una solución informática que posibilite la gestión de cada reporte y facilite la consulta de cada incidencia por equipo y sus soluciones a los técnicos y jefes del taller de reparaciones. Si hablamos de beneficios intangibles que reporta el sistema, podemos mencionar la posibilidad que brinda este sistema a los trabajadores del taller de generar los reportes que llegan al mismo y mantener actualizados los registros que se almacenan en la base de conocimientos y brindar un mejor servicio al usuario y facilitar su trabajo dentro del local.

### 5.4. Análisis de costos y beneficios.

Este sistema no requiere de inversión de software porque las herramientas y la tecnología empleadas para su desarrollo son libres como se describió en el capítulo 1 de este trabajo, por lo que una vez analizado el costo del proyecto y los beneficios que este reporta se puede concluir que el sistema



desarrollado es factible y que su uso a facilitar el trabajo dentro del taller de reparaciones del GAT de la UCI.

### 5.5. Conclusiones.

En este capítulo se presentó la estimación por Puntos de Caso de Uso que resultó muy efectiva para estimar el esfuerzo del proyecto teniendo en cuenta los factores que influyen en el desarrollo del software. Además se realizó el análisis de los beneficios tangibles e intangibles, los costos del mismo y los beneficios que proporciona como sistema, lo cual permite valorar cuán factible sería el desarrollo de la aplicación, también se tuvo conocimiento de cómo optimizar los recursos empleados para su desarrollo.

### CONCLUSIONES

Con este trabajo se ha abordado cada temática relacionada al proceso de desarrollo del sistema informático para facilitar la gestión y el control de los equipos en el taller de reparaciones del Grupo de Asistencia Técnica de la UCI.

Se realizó un estudio de distintos softwares vinculados al campo de acción. Estos software para talleres presentan casi la misma estructura funcional del sistema a implementar para el taller de la UCI, pero cada una de sus diferencias aportaron a este trabajo ideas para luego adaptarlas a las necesidades del taller de reparaciones del GAT.

Cada actividad vinculada al taller de reparaciones fue estudiada con detenimiento y como resultado de ese estudio se obtuvo algunas funcionalidades que se le incorporaron al sistema y que facilitan el procesos de gestión de reportes y el registro de acciones de los técnicos en el taller, además que brindan la posibilidad de mostrar a cualquier persona interesada cada dato almacenado de soluciones a roturas o desperfectos, así como otros datos de interés y novedosos como los almacenados en la base de conocimientos, con sus respectivas opciones de modificar la información en caso de errores y de adicionar nuevos registros. Culminó el ciclo de desarrollo con la obtención de una aplicación Web, cumpliendo con los requerimientos establecidos por el cliente y las normas establecidas por la dirección del proyecto, haciendo uso de cada herramienta y tecnología propuesta para desarrollar el sistema del taller de reparaciones.

Se han explicado en detalle los motivos por los cuales fueron seleccionados las herramientas, lenguajes, metodología y tecnologías que se utilizaron para la realización de esta aplicación Web, sobre todo por su vinculación al desarrollo del software libre y las facilidades que este brinda. El conocimiento y dominio de estas herramientas, lenguajes, metodología y tecnologías es el fruto de un estudio intensivo, obteniéndose información de las distintas referencias bibliográficas, fundamentalmente artículos y otras publicaciones en sitios Web nacionales e internacionales, además de consultar las ayudas de las distintas herramientas y tecnologías empleadas y libros catalogados con un alto nivel científico. Se dieron a conocer las características y ventajas del uso de las aplicaciones Web para el desarrollo de un sistema que cumpla los requerimientos de este y facilite la interacción con el usuario.

Con el uso del software libre, en el desarrollo de este sistema, se garantiza que, en un futuro no muy lejano, la aplicación se adapte sin ningún inconveniente a cualquier cambio de sistema operativo que se realice en el taller, como la migración de Windows a Linux. Facilita, además, las actividades de soporte y cambios en la aplicación ya que se emplean lenguajes, herramientas y tecnologías relacionadas al software libre.

Se puede concluir que fueron vencidos los objetivos trazados y se ha cumplido además con el orden de trabajo establecido por la metodología de desarrollo de software seleccionada, logrando así los diversos artefactos de cada flujo hasta lograr un producto con las características requeridas, y siguiendo los patrones establecidos. Este sistema trae como beneficios al taller una nueva forma de registrar las incidencias atendidas en el mismo. Permite almacenar los datos de cada reporte atendido con los arreglos hechos al medio tecnológico. Estos datos pueden ser consultados con posterioridad en las tablas del historial, y ver los resultados de las estadísticas en forma de gráfico. Ayuda a los técnicos a agilizar su trabajo, al permitirles consultar una base de conocimiento con nuevos datos sobre los equipos rotos, sus desperfectos y posibles soluciones.

Con este trabajo se ha incentivado el estudio del software libre y todo lo relacionado con su uso en aplicaciones Web, además de incrementar la investigación científica y la puesta en práctica de los resultados obtenidos en cada una de estas investigaciones para el bien de la sociedad, en el presente y futuro, y así contribuir al desarrollo de las ciencias informáticas en nuestro país.

### RECOMENDACIONES

Se recomienda continuar con el estudio de cada proceso vinculado al taller de reparaciones de la UCI ya que estos pueden cambiar con el paso del tiempo que exige cambios, y entonces sería necesario reajustar los requerimientos del sistema y modificar la aplicación Web añadiéndole o eliminándole algunas funcionalidades.

Es recomendable el estudio de las diversas herramientas y lenguajes usados para el desarrollo de esta aplicación Web, pues es de vital importancia el dominio de las mismas y su evolución en el mundo de la informática.

También se recomienda el estudio del framework CodeIgniter ya que este facilita el desarrollo de aplicaciones Web usando PHP , facilita el aprendizaje de este lenguaje y además por los distintos recursos que brinda al permitir el uso de diversas clases, librerías, y la amplia y variada documentación que presenta. Además se recomienda el uso de otros framework de desarrollo para PHP que permiten implementar proyectos de mayor alcance y complejidad.

Se recomienda el empleo de nuevas tecnologías y herramientas relacionadas al software libre y que posibiliten la ejecución de esta aplicación Web en diversos navegadores que presentan otros sistemas operativos como Linux.

Como última recomendación se tiene incentivar la investigación científica de nuevos productos informáticos que facilitan el desarrollo de software, sobre todo el software libre, y que pueden dar en un futuro no muy lejano soluciones mas óptimas a cada problema que se presente en nuestro quehacer diario.

### BIBLIOGRAFÍA

#### Referencias bibliográficas:

1. Consultado en: enero de 2008. *“Sistema de gestión de taller de reparaciones de PC.”*  
Disponible en: <http://www.taringa.net/posts/downloads/829214/Sistema-de-gestion-de-taller-de-reparacion-de-PC.html>.
2. Consultado en: enero de 2008. Disponible en: [www.mercadolibre.com.co](http://www.mercadolibre.com.co).
3. Consultado en: enero del 2008. Disponible en: [www.copextel.com.cu](http://www.copextel.com.cu).
4. Consultado en: febrero de 2008. Disponible en: <http://www.gnu.org/philosophy/free-sw.es.html>.
5. Consultado en: enero de 2008. Disponible en:  
<http://www.es.gnu.org/modules/content/index.php?id=42>.
6. Consultado en: diciembre de 2007 *“Beneficios De Las Aplicaciones Basadas En Web Y El Anuncio De Microsoft De La Era “En Vivo”.* Disponible en:  
[http://www.masternewmedia.org/es/aplicaciones\\_web/temas\\_de\\_aplicaciones\\_web/Beneficios\\_De\\_Las\\_Aplicaciones\\_Basadas\\_En%20Web\\_Y\\_El\\_Anuncio\\_De\\_Microsoft\\_De\\_La\\_Era\\_En\\_Vivo.htm](http://www.masternewmedia.org/es/aplicaciones_web/temas_de_aplicaciones_web/Beneficios_De_Las_Aplicaciones_Basadas_En%20Web_Y_El_Anuncio_De_Microsoft_De_La_Era_En_Vivo.htm).
7. Consultado en: enero de 2008. Disponible en:  
<http://es.answers.yahoo.com/question/index?qid=20070516074647AAAd0Mvn>.
8. Consultado en: enero de 2008. Disponible en:  
<http://www.maestrosdelweb.com/editorial/phpintro/>.

## Referencias bibliográficas y bibliografía

---

9. Consultado en: abril de 2008. Disponible en:  
<http://nideaderedes.urlansoft.com/2007/02/27/code-igniter-framework-php/>.
10. Consultado en: junio de 2008. Disponible en: <http://www.tufuncion.com/codeigniter-php>.
11. Consultado en: enero de 2008. Disponible en: <http://www.postgresql.cl/entrevistas/aherrera.htm>.
12. Consultado en: abril de 2008. Disponible en:  
<http://www.linuxparatodos.net/portal/staticpages/index.php?page=servidor-web>.
13. Consultado en: enero de 2008. Disponible en: <http://slion2000.blogspot.com/2007/04/visual-paradigm-una-herramienta-de-lo.html>.
14. Consultado en: junio de 2008. Disponible en:  
<http://www.idg.es/macworld/content.asp?idart=179855>.
15. Consultado en: febrero de 2008. “*Metodologías de desarrollo de software.*” Disponible en:  
[http://www.informatizate.net/articulos/metodologias\\_de\\_desarrollo\\_de\\_software\\_07062004.html](http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html)
16. Jacobson, I. y Booch, G. y Rumbaugh, J.” *El Proceso Unificado de Desarrollo de Software, Volumen I y II.*” La Habana: s.n., 2004.
17. Larman, Craig.” *UML y Patrones*”.

### Webgrafía:

- Consultado en: diciembre de 2007. Disponible en:  
[http://www.intellia.com.mx/esp/servicios/aplicaciones\\_web\\_a\\_la\\_medida.php](http://www.intellia.com.mx/esp/servicios/aplicaciones_web_a_la_medida.php).
- Consultado en: diciembre de 2007. Disponible en:  
<http://www.microsoft.com/latam/windowsserver2003/evaluation/whyupgrade/top10nt.mspcx>.
- Consultado en: diciembre de 2007. Disponible en: <http://www.fsfla.org/?q=node/18>.

## *Referencias bibliográficas y bibliografía*

---

- Consultado en: enero de 2008. Disponible en:  
[http://groups.google.com/group/CafeBSD/browse\\_thread/thread/17884ad5a16bf904](http://groups.google.com/group/CafeBSD/browse_thread/thread/17884ad5a16bf904).
- Consultado en: febrero de 2008. Disponible en: [http://www.ati.es/article.php3?id\\_article=294](http://www.ati.es/article.php3?id_article=294).
- Consultado en: abril de 2008. "Framework". Disponible en:  
[http://www.lsi.us.es/~javierj/investigacion\\_ficheros/Framework.pdf](http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf).
- Consultado en: abril de 2008. "Arquitectura de programación." Disponible en:  
<http://elcodigok.blogspot.com/2007/09/arquitectura-de-programacin-en-3-capas.html>.

**ANEXOS**

Anexo 1. Descripción de los Casos de Uso del Negocio:

*Tabla 1. Descripción de los CUN.*

<b>Caso de Uso:</b>	<b>Solicitar Reporte</b>	
<b>Actores:</b>	<b>Cliente</b>	
<b>Trabajadores:</b>	<b>Técnico</b>	
<b>Resumen:</b>	<p>El caso de uso se inicia cuando el cliente solicita al taller un servicio de reparación de un equipo defectuoso enviando un reporte que se le hace llegar al técnico desde la BAU o del área de recambio y este se dispone a reparar el equipo. El técnico revisa en la base de conocimientos en busca de las posibles soluciones al desperfecto del equipo. Cuando repara el equipo, actualiza y almacena el reporte, lo envía de regreso al BAU o área de recambio, según sea el caso. Si no puede repararlo el técnico actualiza el reporte con los datos del desperfecto y envía el reporte a UCIXTEL. El técnico busca los datos de los reportes actualizados y obtiene estadísticas de los mismos.</p>	
<b>Precondiciones:</b>	Los reportes tienen que llegar con sus datos correctos al taller.	
<b>Flujo Normal de Eventos</b>		
<b>Sección “Actualizar”</b>		
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>	
1. El cliente envía reporte.	<p>1.1. El técnico recibe el reporte.</p> <p>1.2.Consulta la base de conocimientos</p>	



	<p>1.3. Verifica si puede arreglar equipo</p> <p>1.4. Actualiza reporte.</p> <p>1.5. Envía el reporte a su destino.</p>
<b>Flujos Alternos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>
	<p>1.3. Si no puede reparar el equipo envía el reporte a COPEXTEL.</p>
<b>Poscondiciones</b>	<p>Se actualiza el reporte y se envía a la entidad que le corresponde.</p>
<b>Sección "Buscar Estadísticas"</b>	
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>
<p>2. El cliente solicita estadísticas de los reportes.</p>	<p>2.1. El técnico busca los datos del reporte.</p> <p>2.2. Se muestran las estadísticas.</p>

<b>Flujos Alternos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>
	2.3. No se encuentran los datos del equipo solicitado.
<b>Poscondiciones</b>	Se muestran las estadísticas del reporte.

Reglas del Negocio:

- Los reportes deben tener sus datos exactos antes de llegar al taller.
- El técnico debe registrar los arreglos hechos al equipo.
- El técnico debe revisar la base de conocimientos antes de ejecutar cualquier reparación, esta debe estar actualizada y documentada lo mejor posible.
- Cada reporte debe ser almacenado en un historial para luego ser consultado por el personal correspondiente, en caso que sea necesario tener esa información.

Anexo 2. Diagrama de Actividades:

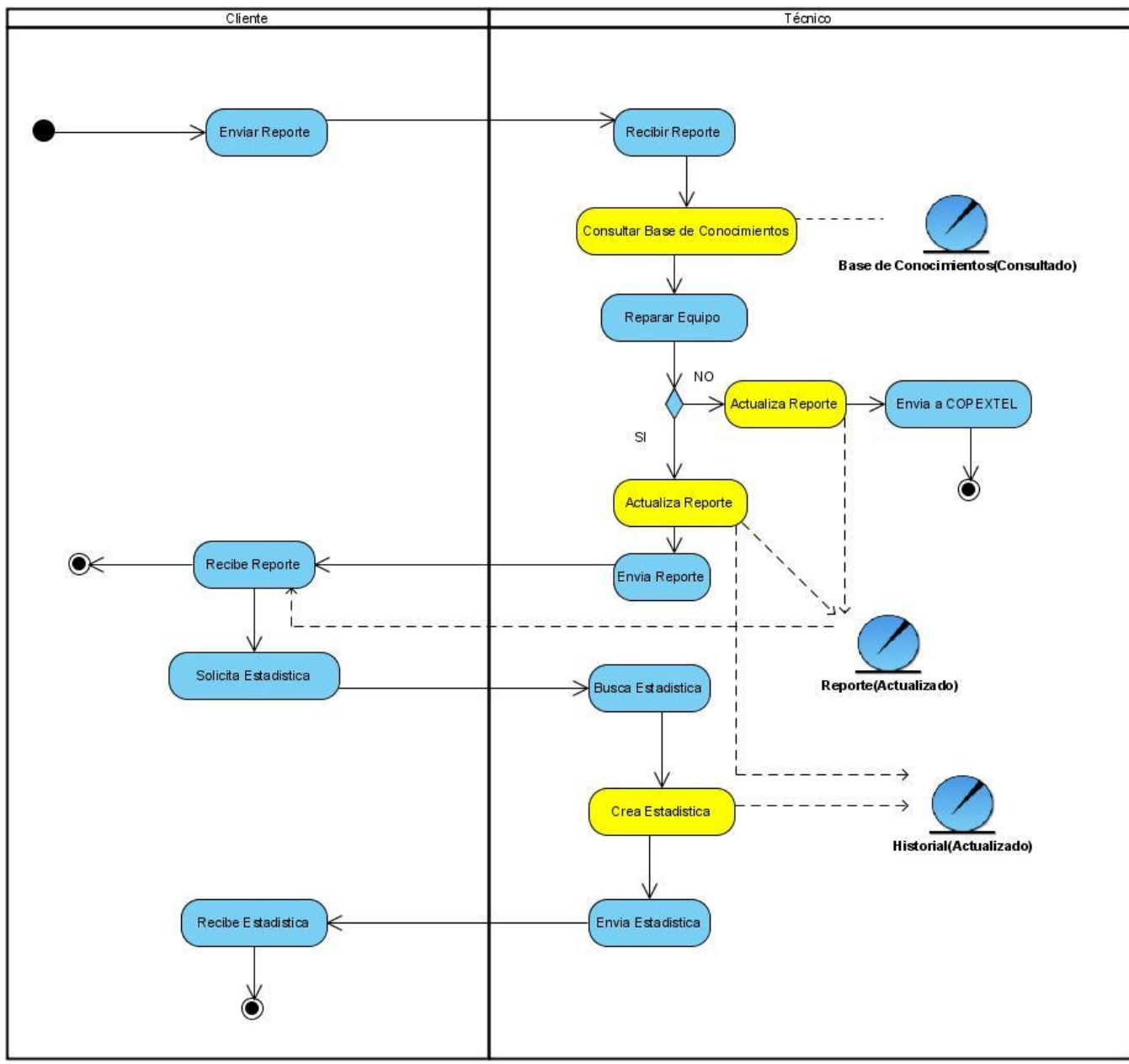


Figura 2.1. Diagrama de Actividades.

Anexo 3. Modelo de Objetos:

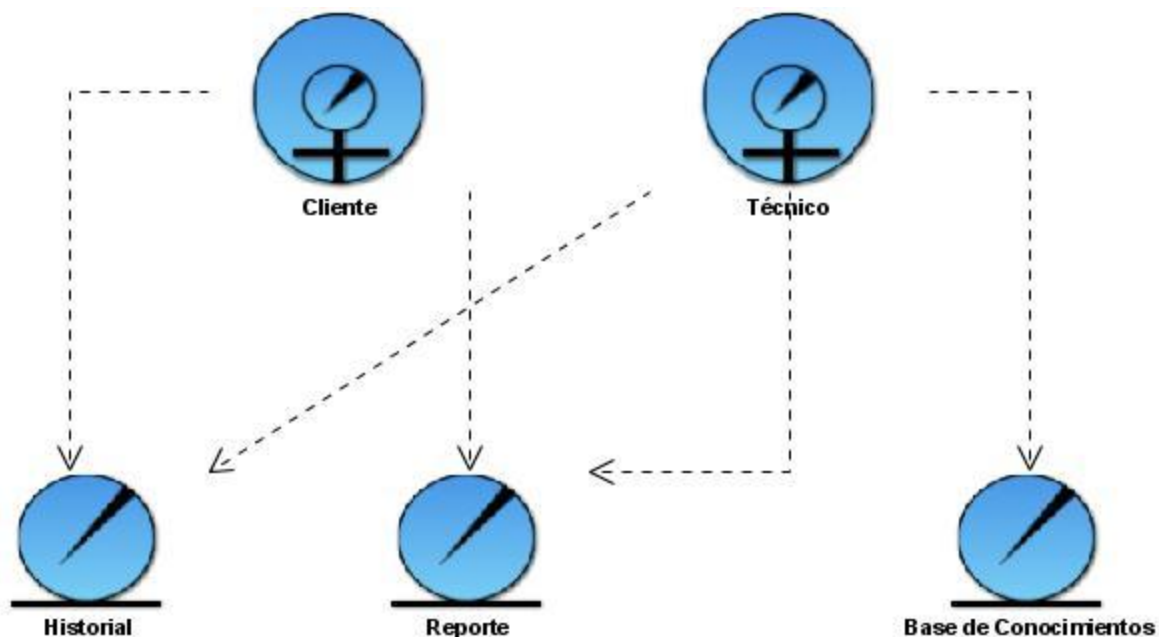


Figura 3.1. Modelo de Objetos.

Anexo 4. Descripción de los casos de uso del sistema.

Tabla 4.1. Descripción del CUS Autenticar.

Caso de Uso	
1	<b>Autenticar</b>
<b>Propósito</b>	
<b>Actores:</b> Técnico	
<b>Resumen:</b>	El caso de uso se inicia cuando el técnico se autentica para entrar al sistema a realizar su trabajo.
<b>Referencias:</b>	R7
<b>Acción del actor.</b>	<b>Respuesta del sistema.</b>
1. El técnico entra su usuario y contraseña.	1.1. El sistema verifica los datos. 1.2. El sistema permite entrar al técnico al

	sistema.
<b>Flujo Alternativo</b>	
<b>Acción del Actor</b>	<b>Respuesta del sistema.</b>
1.1. El técnico no puede entrar al sistema.	1.2. Los datos son incorrectos y el sistema no permite el acceso a la información. 1.3. El sistema vuelve a pedir los datos del técnico.

Tabla 4.2. Descripción del CUS Gestionar Base de Conocimiento.

<b>Caso de Uso</b>	
<b>3</b>	<b>Gestionar Base de Conocimiento</b>
<b>Propósito</b>	
<b>Actores:</b> Técnico	
<b>Resumen:</b>	El caso de uso se inicia cuando el técnico accede a la base de conocimientos para interactuar con ella: adicionar, actualizar, eliminar y buscar datos.
<b>Referencias:</b>	R1, R1.1, R1.2, R1.3, R2, R2.1, R2.2, R2.3, R2.4.
<b>Escenario: Adicionar</b>	
<b>Acción del actor.</b>	<b>Respuesta del sistema.</b>
1. El técnico adiciona datos de equipos, incidencia o solución a la base de conocimientos.	1.1. El sistema adiciona datos de equipos, incidencia o solución en la base de conocimientos.
<b>Escenario: Actualizar</b>	
<b>Acción del Actor.</b>	<b>Respuesta del sistema.</b>

1. El técnico selecciona un equipo, incidencia o solución y actualiza sus datos.	1.1. El sistema actualiza los datos del equipo, incidencia o solución seleccionada.
<b>Escenario: Buscar</b>	
<b>Acción del Actor.</b>	<b>Respuesta del sistema.</b>
1. El técnico entra el nombre de un equipo	1.1. El sistema busca el equipo. 1.2. El sistema muestra el equipo, incidencia y la solución correspondiente.
<b>Escenario: Eliminar</b>	
<b>Acción del Actor.</b>	<b>Respuesta del sistema.</b>
1. El técnico selecciona un equipo, incidencia o solución y da la opción de eliminar.	1.1. El sistema pregunta al técnico si desea eliminar el equipo, incidencia o solución seleccionada.
2. El técnico marca la opción de eliminar.	2.1. El sistema elimina el equipo, incidencia o solución seleccionada.
<b>Flujo Alternativo</b>	
<b>Acción del Actor</b>	<b>Respuesta del sistema.</b>
2. El técnico marca la opción de no eliminar.	

Tabla 4.3. Descripción del CUS Gestionar Reportes.

<b>Caso de Uso</b>	
<b>3</b>	<b>Gestionar Reportes</b>
<b>Propósito</b>	
<b>Actores:</b> Técnico	

<b>Resumen:</b>	El caso de uso se inicia cuando el técnico actualiza el reporte una vez realizada la revisión del equipo o su reparación y lo envía, luego almacena los datos del reporte en el historial de reportes del taller.
<b>Referencias:</b>	R4, R4.1, R4.2, R4.3, R4.4
<b>Escenario: Actualizar Reporte</b>	
<b>Acción del actor.</b>	<b>Respuesta del sistema.</b>
	1. El sistema muestra los datos del reporte.
2. El técnico selecciona el reporte a actualizar.	2.1. El sistema pregunta al técnico si desea actualizar o no.
3. El técnico selecciona actualizar los datos del reporte.	
4. El técnico actualiza los datos del reporte.	4.1. El sistema recibe los datos actualizados del reporte.
5. El técnico envía los datos del reporte actualizado.	3.1. El sistema envía el reporte actualizado y además almacena una copia del reporte en el historial del taller.
<b>Flujo Alternativo</b>	
<b>Acción del Actor.</b>	<b>Respuesta del sistema.</b>
3. El técnico no actualiza los datos del reporte.	

Tabla 4.4. Descripción del CUS Generar Estadísticas.

<b>4</b>	<b>Generar Estadísticas</b>	
<b>Propósito</b>		
<b>Actores:</b> Técnico		
<b>Resumen:</b>	El caso de uso se inicia cuando el actor entra los datos de un reporte en específico y el sistema le devuelve las estadísticas en forma de gráficos o tablas.	
<b>Referencias:</b>	R5, R5.1, R5.2, R5.3, R5.4, R5.5, R5.6.	
<b>Acción del actor.</b>	<b>Respuesta del sistema.</b>	
<p>1. El técnico selecciona el tipo de estadísticas a consultar, sea gráfico o tablas del historial.</p> <p>2. El técnico entra un intervalo de fechas de un equipo seleccionado previamente.</p>	<p>1.1. El sistema procesa los datos entrados.</p> <p>1.2. El sistema crea los gráficos o tablas con el resultado de la búsqueda.</p> <p>1.3. El sistema muestra los gráficos o tablas con el resultado de la búsqueda.</p>	
<b>Flujo Alternativo</b>		
<b>Acción del Actor</b>	<b>Respuesta del sistema.</b>	
	1.2. No existen los datos entrados y el sistema muestra un mensaje de error.	

Tabla 4.5. Descripción del CUS Gestionar Usuario.

<b>Caso de Uso</b>		
<b>3</b>	<b>Gestionar Usuario</b>	



<b>Propósito</b>	
<b>Actores:</b> Técnico	
<b>Resumen:</b>	El caso de uso se inicia cuando el administrador adiciona, actualiza, busca y elimina a los nuevos usuarios que accederán al sistema.
<b>Referencias:</b>	R3, R3.1, R3.2, R3.3, R3.4
<b>Escenario: Adicionar</b>	
<b>Acción del actor.</b>	<b>Respuesta del sistema.</b>
6. El administrador adiciona datos del usuario.	1.2. El sistema adiciona datos del usuario.
<b>Escenario: Adicionar</b>	
<b>Acción del Actor.</b>	<b>Respuesta del sistema.</b>
1. El administrador selecciona un usuario y actualiza sus datos.	1.1. El sistema actualiza los datos del usuario seleccionado.
<b>Escenario: Buscar</b>	
<b>Acción del Actor.</b>	<b>Respuesta del sistema.</b>
1. El administrador entra el nombre de un usuario.	1.1. El sistema busca el usuario. 1.2. El sistema muestra los datos del usuario.
<b>Escenario: Eliminar</b>	
<b>Acción del Actor.</b>	<b>Respuesta del sistema.</b>
3. El administrador selecciona un usuario y da la opción de eliminar. 4. El administrador marca la opción de eliminar.	6.1. El sistema pregunta al técnico si desea eliminar al usuario seleccionado. 2.1. El sistema elimina al usuario.

Flujo Alternativo	
Acción del Actor	Respuesta del sistema.
2. El administrador marca la opción de no eliminar.	

Anexo 5. Diagramas de Interacción.

Diagramas de Colaboración. Clases del Análisis.

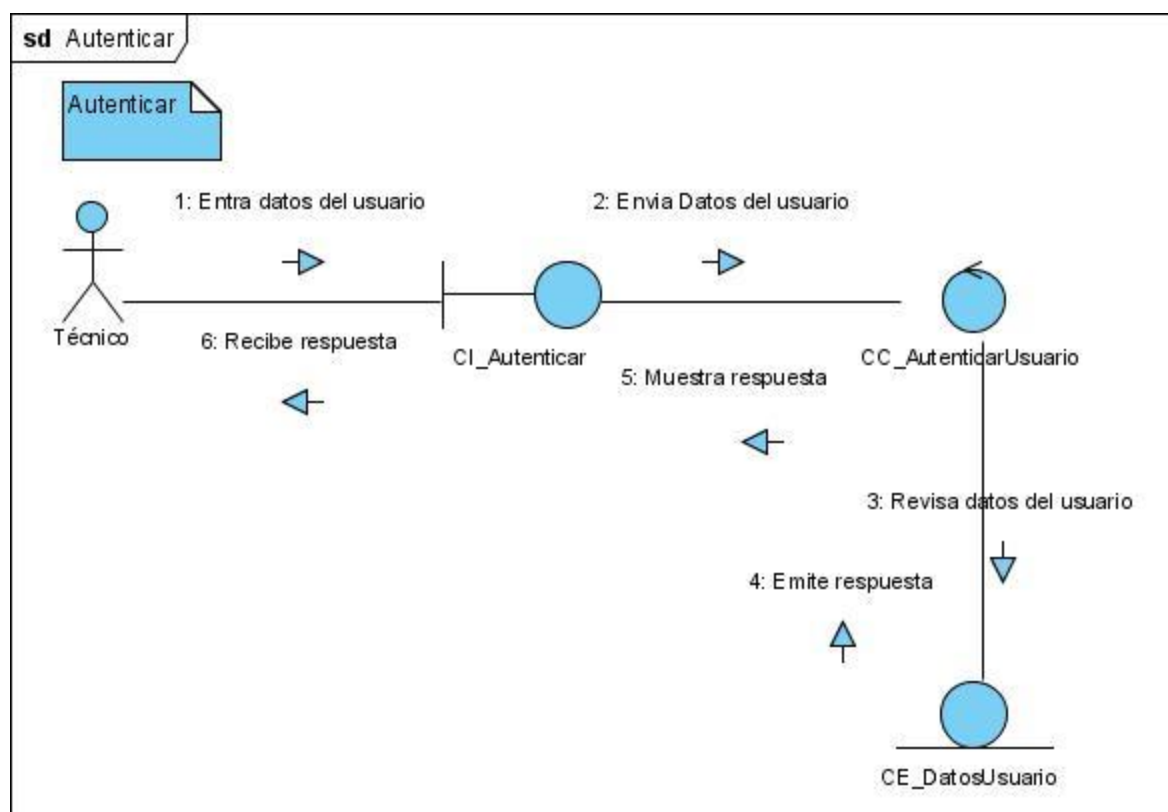


Figura 5.1. Diagrama de colaboración del Análisis del CUS Autenticar.

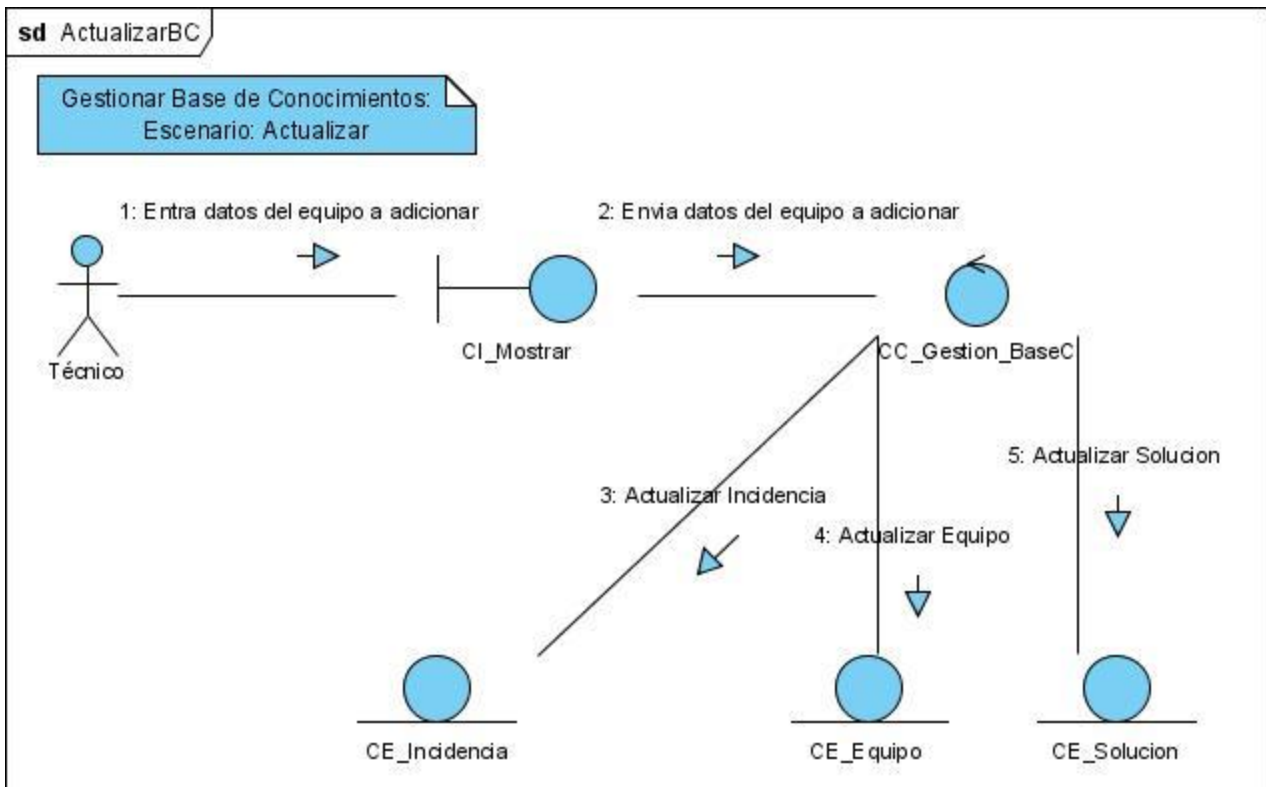


Figura 5.2. Diagrama de colaboración del Análisis del CUS Gestionar Base de Conocimientos. Escenario Actualizar.

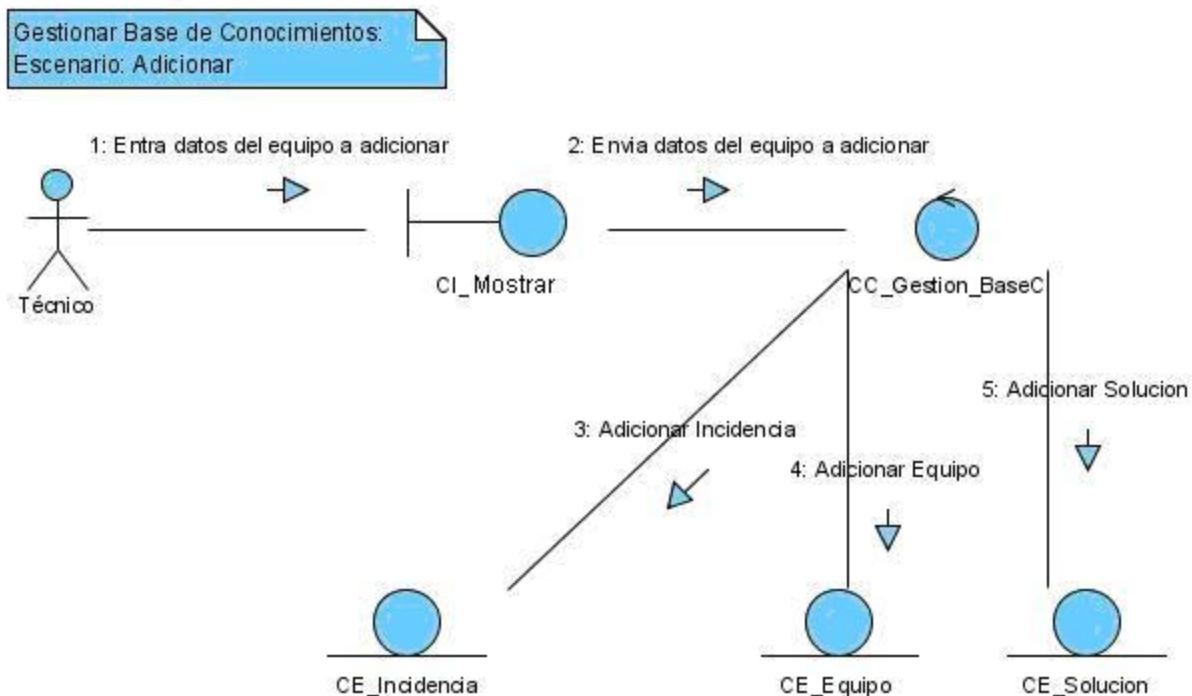


Figura 5.3. Diagrama de colaboración del Análisis del CUS Gestionar Base de Conocimientos. Escenario Adicionar.

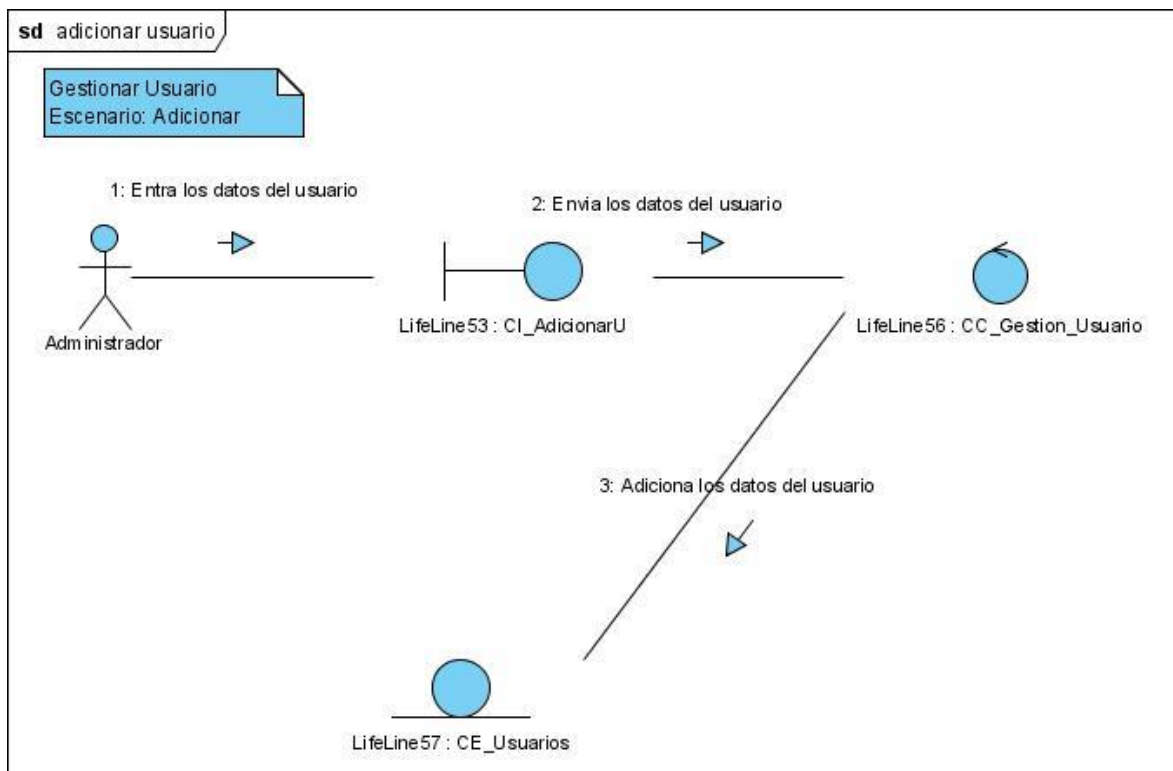


Figura 5.4. Diagrama de colaboración del Análisis del CUS Gestionar Usuario. Escenario Adicionar.

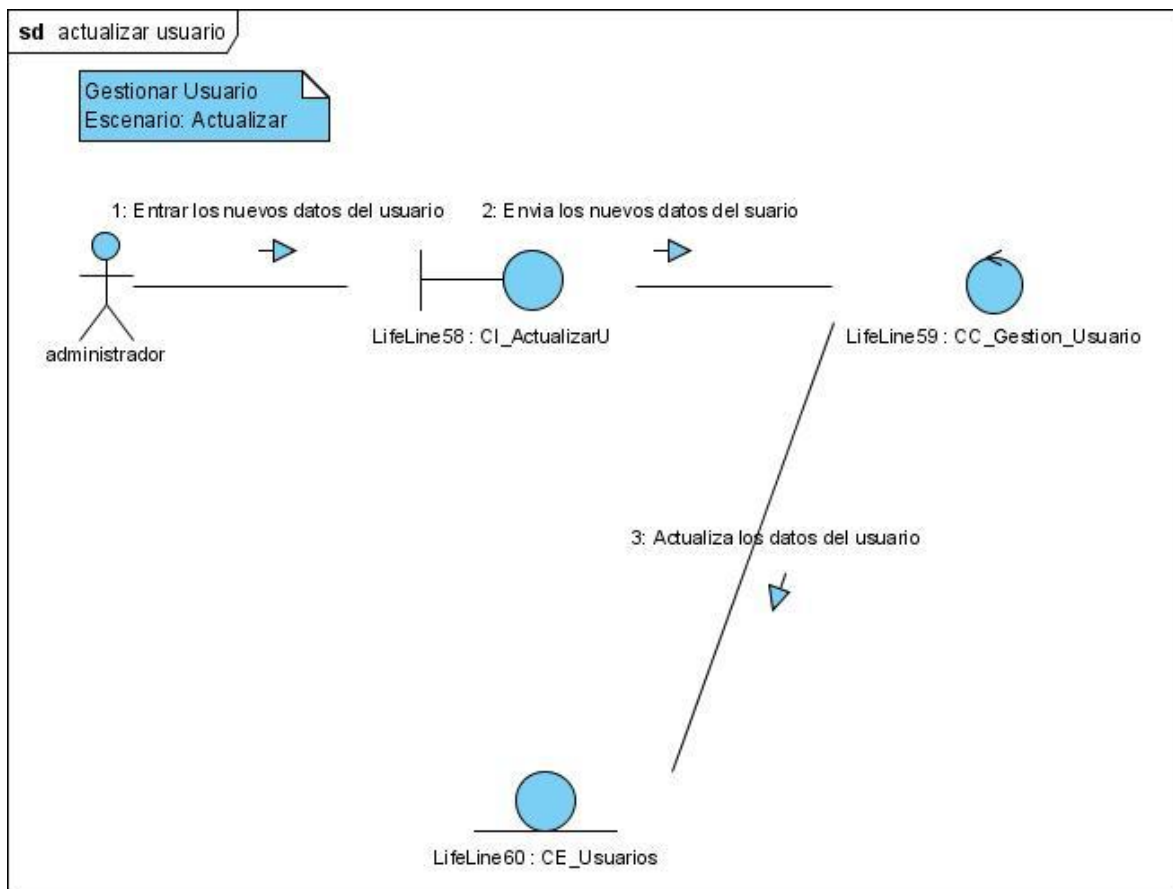


Figura 5.5. Diagrama de colaboración del Análisis del CUS Gestionar Usuario. Escenario Actualizar.

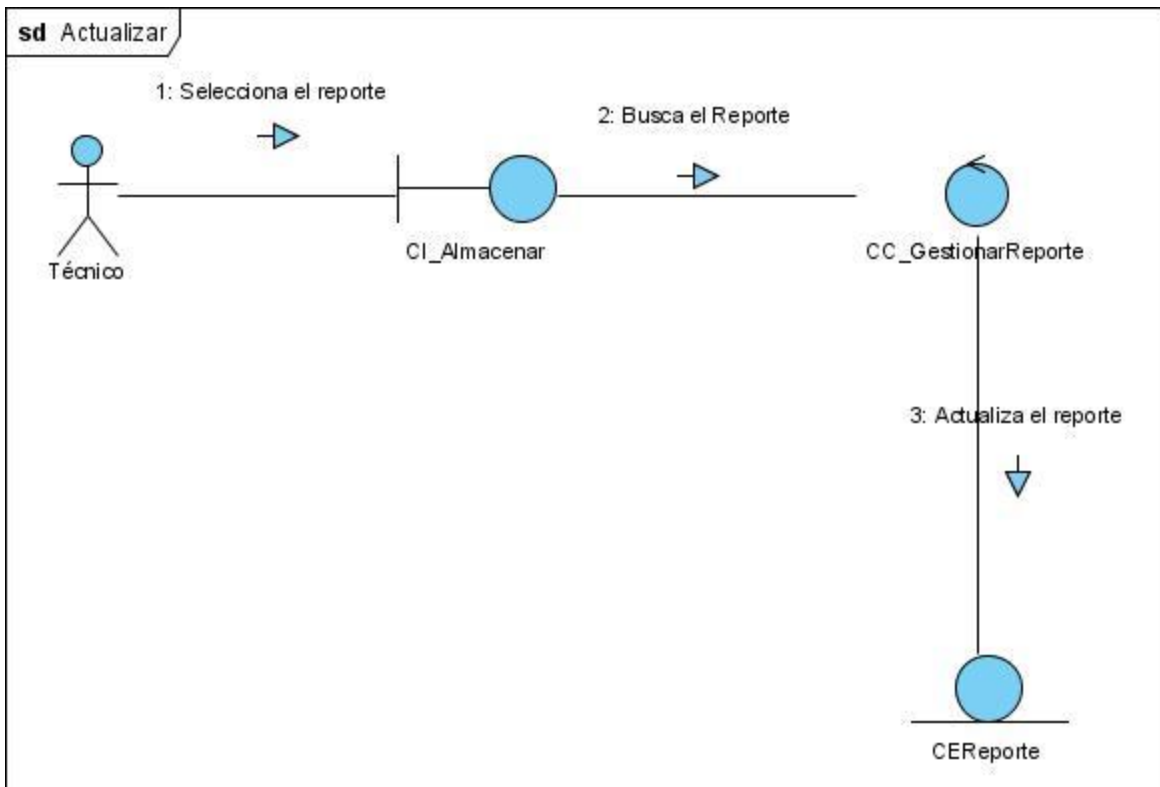


Figura 5.6. Diagrama de colaboración del Análisis del CUS Gestionar Reporte. Escenario Actualizar Reporte.

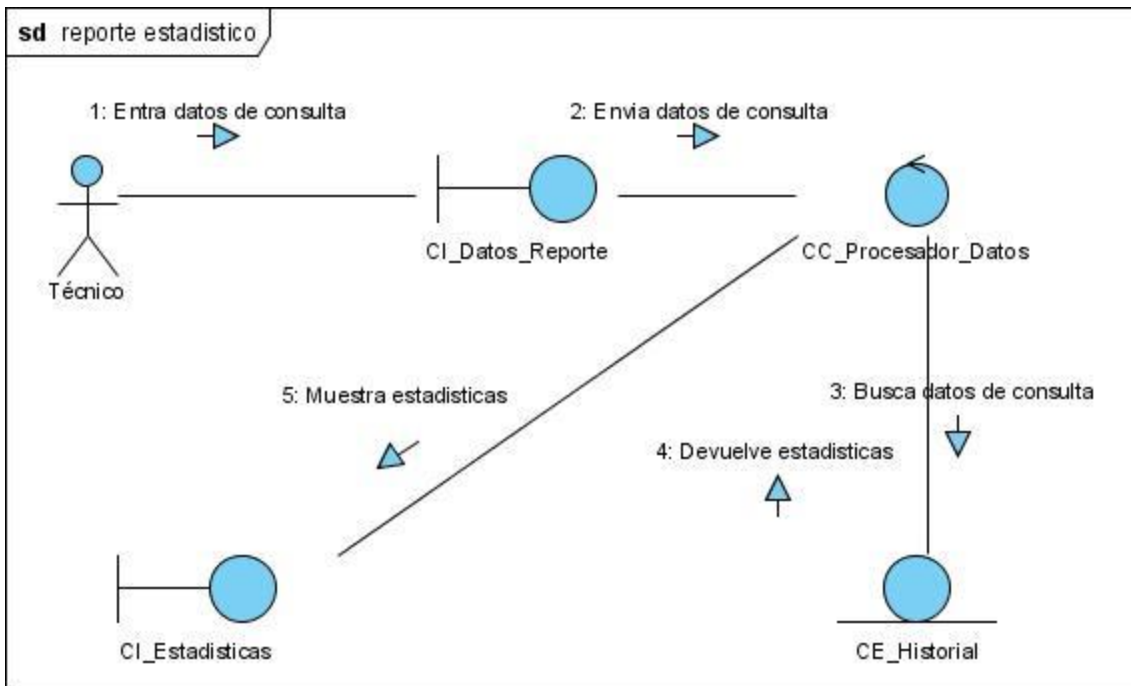


Figura 5.7. Diagrama de colaboración del Análisis del CUS Generar Estadísticas.

Diagramas de Secuencia. Clases del Diseño.



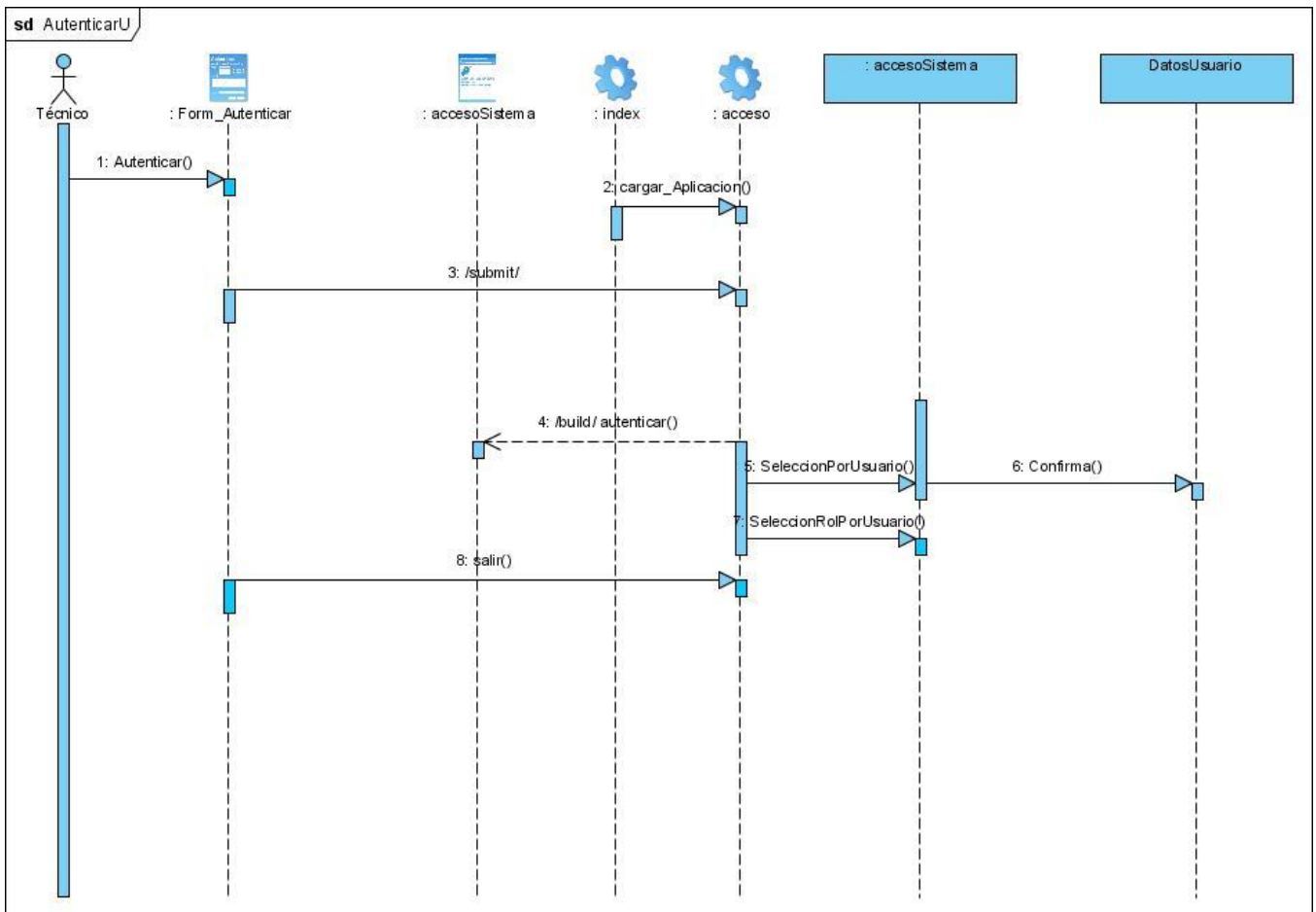


Figura 5.8. Diagrama de secuencia del diseño del CUS Autenticar.

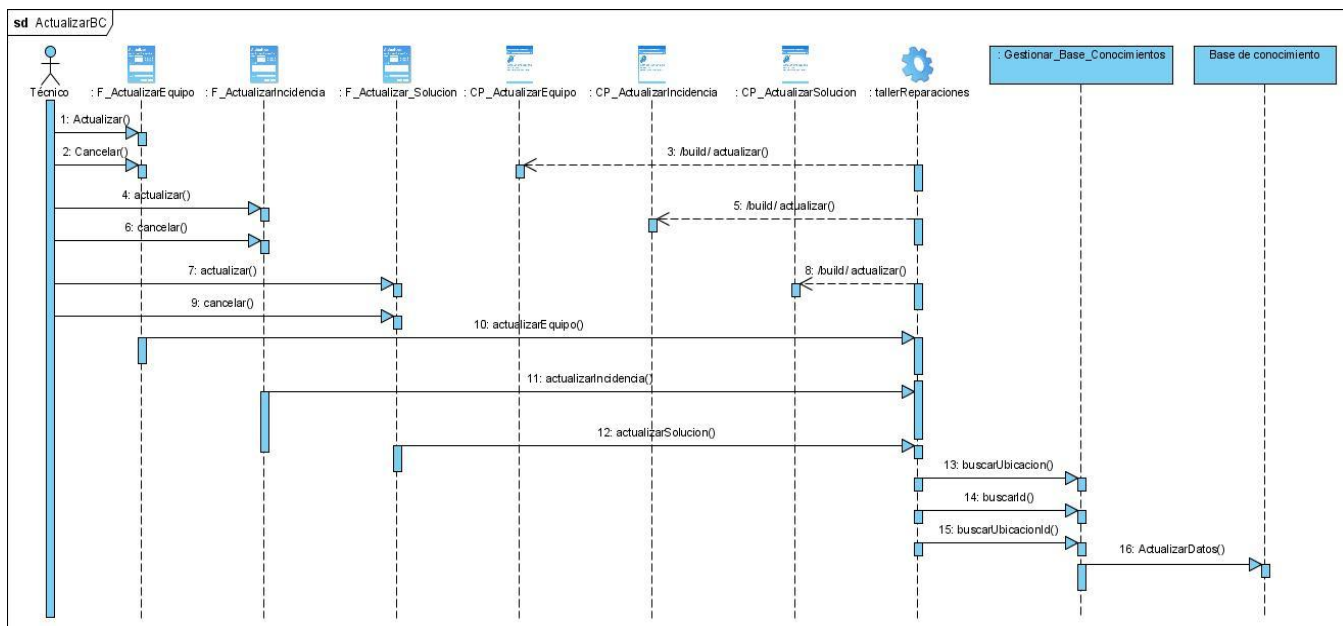


Figura 5.9. Diagrama de secuencia del diseño del CUS Gestionar Base de Conocimientos. Escenario Actualizar.

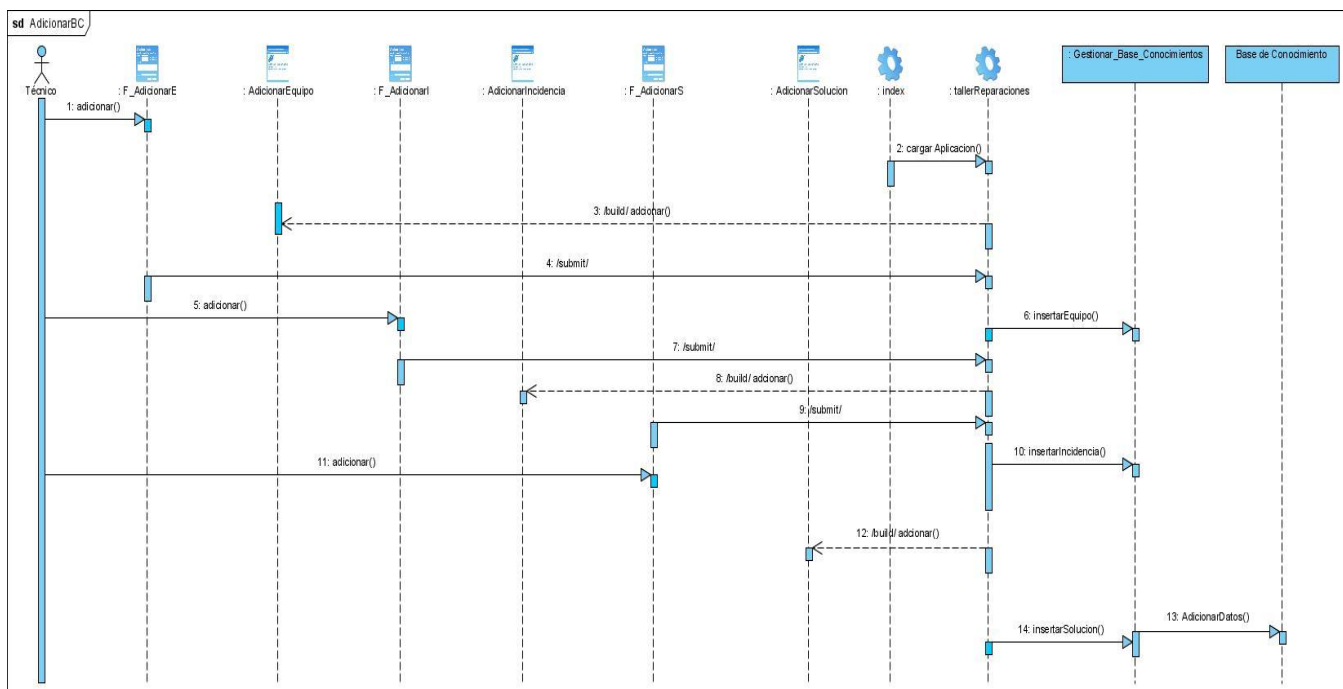


Figura 5.10. Diagrama de secuencia del diseño del CUS Gestionar Base de Conocimientos. Escenario Adicionar.

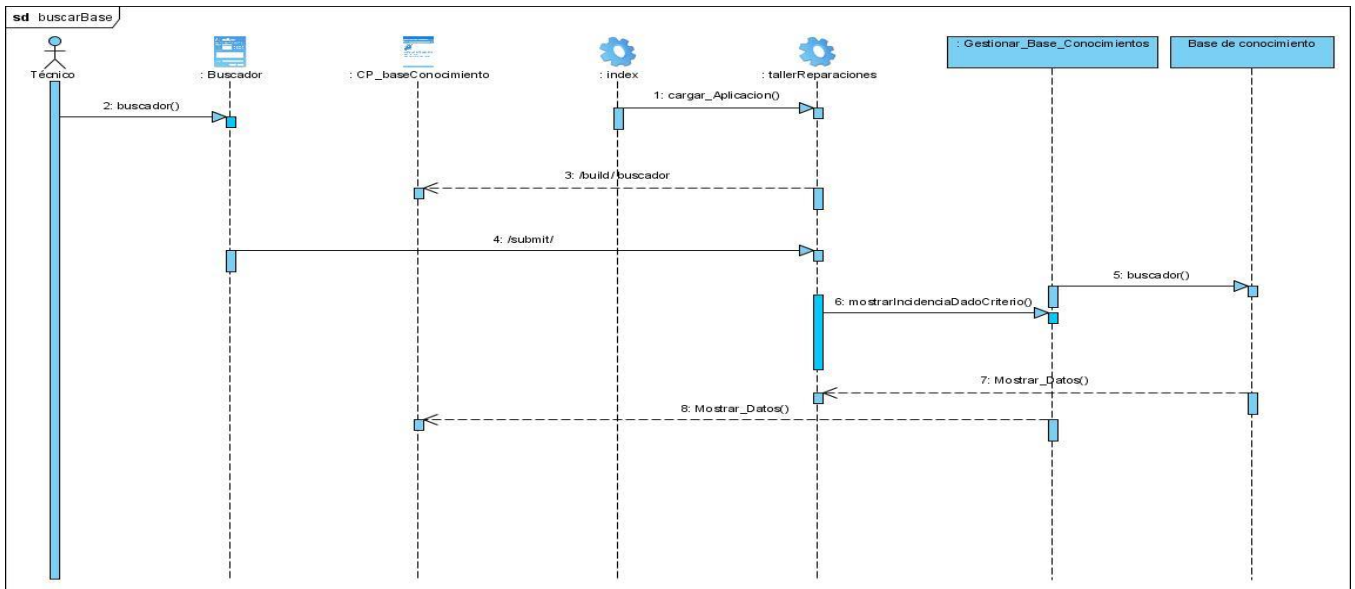


Figura 5.11. Diagrama de secuencia del diseño del CUS Gestionar Base de Conocimientos. Escenario Buscar.

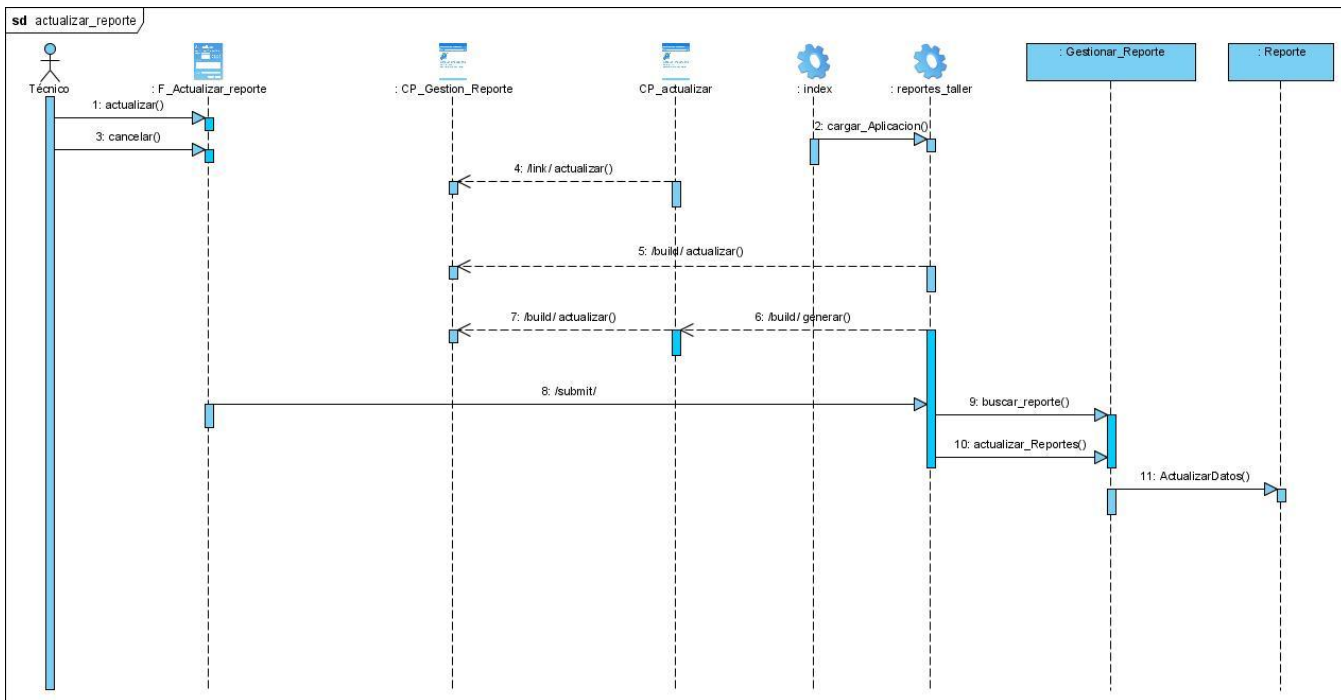


Figura 5.12. Diagrama de secuencia del diseño del CUS Gestionar Reporte. Escenario Actualizar Reporte.

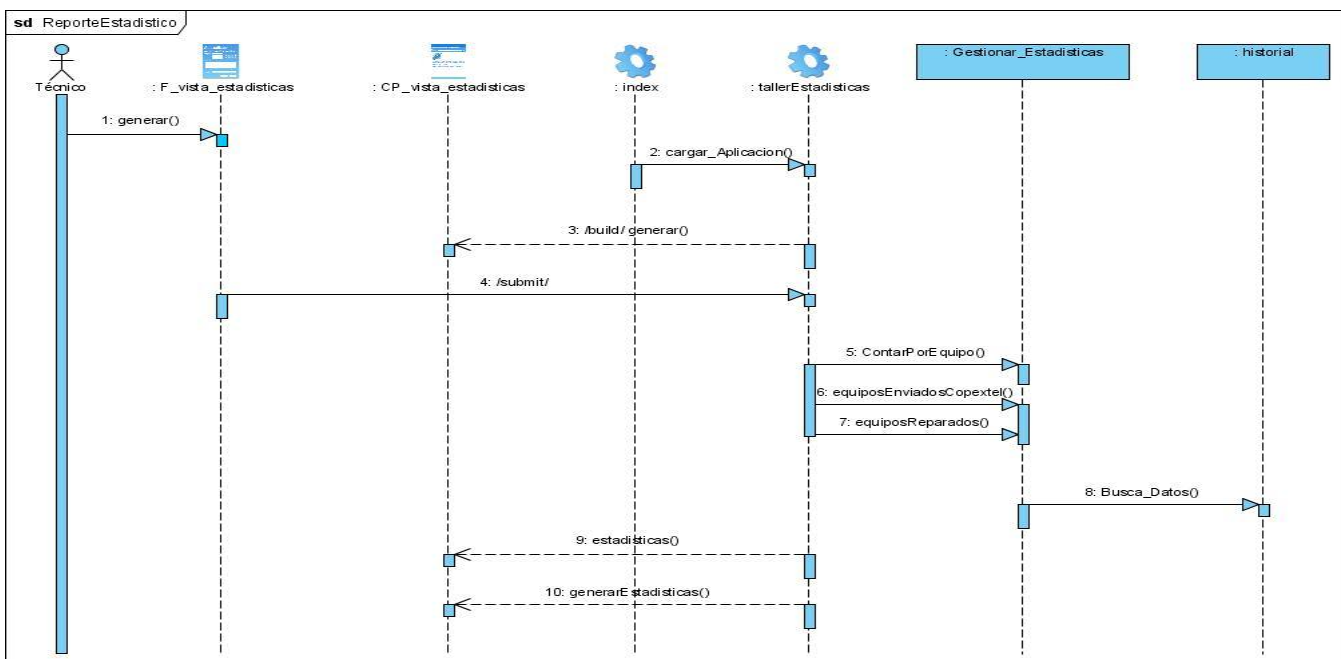


Figura 5.13. Diagrama de secuencia del diseño del CUS Generar Estadísticas.

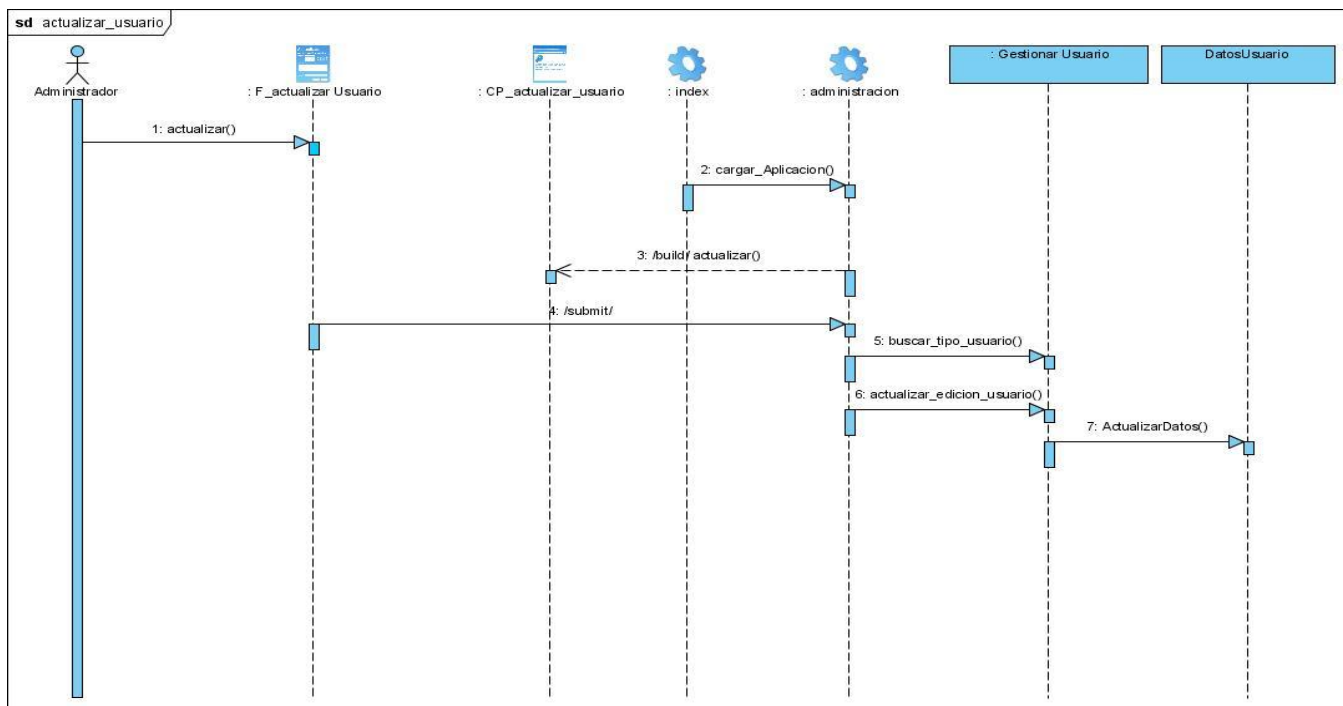


Figura 5.14. Diagrama de secuencia del diseño del CUS Gestionar Usuarios. Escenario Actualizar.

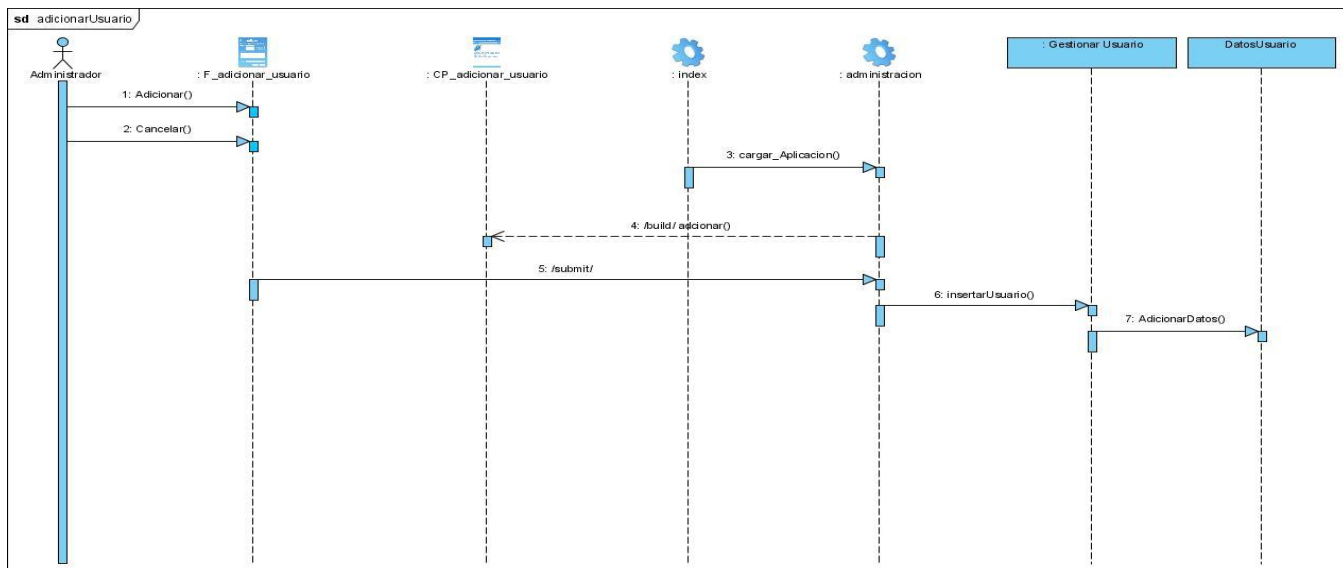


Figura 5.15. Diagrama de secuencia del diseño del CUS Gestionar Usuarios. Escenario Adicionar.

### GLOSARIO DE TÉRMINOS.

*Web:* World Wide Web (o la "Web") o Red Global Mundial es un sistema de documentos de hipertexto y/o hipermedios enlazados y accesibles a través de Internet.

*Aplicaciones de escritorio o desktop:* Las aplicaciones de escritorio almacenan tradicionalmente sus datos en un archivo, aunque algunos tienen la capacidad de colaborar con otros usuarios, o almacenar sus datos en una base de datos central.

*Bug:* Un defecto de software (computer bug en inglés), es el resultado de un fallo o deficiencia durante el proceso de creación de programas de ordenador o computadora (software).

*Open Source:* Código abierto (en inglés open source) es el término con el que se conoce al software distribuido y desarrollado libremente.

*SOAP:* (Siglas de Simple Object Access Protocol) es un protocolo estándar creado por Microsoft, IBM y otros, está actualmente bajo el auspicio de la W3C que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. SOAP es uno de los protocolos utilizados en los servicios Web.

*GNOME:* Es un entorno de escritorio para sistemas operativos de tipo Unix bajo tecnología X Windows.

*DOM:* El Modelo de Objetos del Documento (DOM) es una interfaz de programación de aplicaciones (API) para documentos HTML y XML. Define la estructura lógica de los documentos y el modo en que se accede y manipula un documento.

*Hosting o alojamiento Web:* Es el servicio que provee a los usuarios de Internet un sistema para poder almacenar información, imágenes, vídeo, o cualquier contenido accesible vía Web.

*Caché:* Es un conjunto de datos duplicados de otros originales, con la propiedad de que los datos originales son costosos de acceder, normalmente en tiempo, respecto a la copia en el caché. Cuando

se accede por primera vez a un dato, se hace una copia en el caché; los accesos siguientes se realizan a dicha copia, haciendo que el tiempo de acceso medio al dato sea menor.

*Plugin o componente enchufable:* Es una aplicación informática que interactúa con otra aplicación para aportarle una función o utilidad específica, generalmente muy específica.

*Renderizar:* Es el proceso de generar una imagen desde un modelo.

*URL:* Significa Uniform Resource Locator, es decir, localizador uniforme de recurso. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.

*Licencia BSD:* La licencia BSD es la licencia de software otorgada principalmente para los sistemas BSD (Berkeley Software Distribution). Pertenece al grupo de licencias de software Libre. La licencia BSD al contrario que la GPL permite el uso del código fuente en software no libre.

*Commit:* En el contexto de la Ciencia de la computación y la gestión de datos, commit(acción de cometer) se refiere a la idea de hacer que un conjunto de cambios "tentativos, o no permanentes" se conviertan en permanentes. Un uso popular es al final de una transacción de base de datos.

*RollBack:* En tecnologías de base de datos, un rollback es una operación que devuelve a la base de datos a algún estado previo.

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

*RSS:* Es un sencillo formato de datos que es utilizado para redifundir contenidos a suscriptores de un sitio Web.

*USB:* El Universal Serial Bus (bus universal en serie) es un puerto que sirve para conectar periféricos a una computadora.

*LDAP*: (Lightweight Directory Access Protocol) es un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

*FTP (File Transfer Protocol)*: Es un protocolo de transferencia de archivos entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor.