

**UNIVERSIDAD NACIONAL AUTONOMA DE NICARAGUA**

**UNAN - MANAGUA**

**FACULTAD DE CIENCIAS E INGENIERIAS**

**DEPARTAMENTO DE COMPUTACION**



**METODOLOGIAS ÁGILES DE DESARROLLO DE SOFTWARE**

**LA PROGRAMACIÓN EXTREMA APLICADA AL DESARROLLO DEL SISTEMA DE INFORMACIÓN WEB  
PARA LA GESTIÓN DE PRESUPUESTOS EN CEM COMUNICACIONES S.A.**

**AUTORES:**

- 1. BR. MARCO ANTONIO FONSECA VARGAS**
- 2. BR. ENGELS ISMAEL OBREGÓN HERNANDEZ**
- 3. BR. LEONTE DERWING ESPINOZA JAEN**

**TUTOR:**

**MSC. AINOA CALERO**

## DEDICATORIA

Con todo mi cariño y mi amor primeramente a Dios y a la persona que hizo todo en la vida para que yo pudiera lograr mis metas, por motivarme y darme la mano cuando sentía que el camino se terminaba, a usted por siempre mi corazón y mi agradecimiento.

Mi madre Catalina Vargas.

***Marco Fonseca***

En primer lugar a Dios por estar conmigo en cada paso que doy y por haberme dado la fortaleza suficiente para culminar mi carrera. En segundo lugar con todo mi amor a mi madre por brindarme su apoyo incondicional, sus sabios consejos y por alentarme siempre a seguir hacia adelante para cumplir mis objetivos propuestos.

***Engels Obregón***

Dedico este proyecto a mis padres, que han sido la motivación y principal sostén para continuar a lo largo de esta carrera y a mi esposa como pilar fuerte quien me incentiva a luchar y no desfallecer a pesar de las dificultades.

***Leonte Espinoza***

## **AGRADECIMIENTOS**

Agradecemos primeramente a Dios por brindarnos la oportunidad y las fuerzas necesarias para culminar nuestros estudios superiores, a nuestras familias por el apoyo incondicional que siempre nos han dado y a nuestros profesores por compartir con nosotros su valioso conocimiento intelectual y humano.

## Contenido del Documento

1.	RESUMEN .....	14
2.	INTRODUCCIÓN AL TEMA Y SUBTEMA.....	15
3.	JUSTIFICACIÓN.....	17
4.	OBJETIVOS .....	19
4.1.	Objetivo General .....	19
4.2.	Objetivos Específicos .....	19
5.	MARCO TEÓRICO.....	20
5.1.	Origen de la Metodologías Ágiles del Desarrollo de Software .....	20
5.2.	Principales Metodologías Ágiles.....	22
5.3.	Metodología Programación Extrema (XP).....	25
5.4.	Herramientas de la metodología XP .....	25
5.5.	Roles de la metodología XP .....	32
5.6.	Fases de la programación extrema .....	33
5.7.	Prácticas de la metodología XP .....	35
5.8.	Acerca de los sistemas WEB.....	37
5.9.	Herramientas para el desarrollo de Software .....	40
6.	MARCO METODOLÓGICO.....	52
6.1.	Tipo de investigación.....	52
6.2.	Diseño de la investigación .....	52
6.3.	Fuentes de información .....	52
6.4.	Instrumentos de Recolección de Datos.....	53
7.	APLICACIÓN DE XP AL DESARROLLO DEL SISTEMA .....	54
7.1.	Descripción del Cliente.....	54
7.2.	Planificación .....	55
7.3.	Historias de Usuario .....	55
7.4.	Metáfora del Sistema de Gestión de Presupuestos.....	56
7.5.	Asignación de Roles.....	57
7.6.	Plan de Entregas.....	57

8.	CICLO DE VIDA DEL SISTEMA DE GESTIÓN DE PRESUPUESTOS.....	58
8.1.	Primera Iteración.....	58
8.2.	Segunda Iteración.....	71
8.3.	Tercera Iteración .....	82
9.	CONCLUSIONES .....	87
10.	BIBLIOGRAFIA .....	89
11.	RESULTADOS .....	90
11.1.	RESULTADO 1: Historias de usuario .....	91
11.2.	RESULTADO 2: Tareas de ingeniería.....	97
11.3.	RESULTADO 3: Pruebas de Aceptación. ....	114
11.4.	RESULTADO 4: Tarjetas CRC .....	122
11.5.	RESULTADO 5: Diagrama de clases del Sistema de Gestión de Presupuestos.....	126
11.6.	RESULTADO 6: Bitácora de reuniones.....	127

## **Listado de Tablas**

Tabla 1. Formato de Historia de usuario

Tabla 2. Formato de Tarea de Ingeniería

Tabla 3. Formato de Prueba de Aceptación

Tabla 4. Formato de Tarjeta CRC

Tabla 5. Ciclo de vida de una aplicación ASP .NET

Tabla 6. Asignación de Roles del Proyecto

Tabla 7. Plan de entrega del proyecto

Tabla 8. Historias de usuario de la primera iteración

Tabla 9. Tareas de ingeniería de la primera iteración

Tabla 10. Pruebas de aceptación de la primera iteración

Tabla 11. Historias de usuario de la segunda iteración

Tabla 12. Tareas de ingeniería de la segunda iteración

Tabla 13. Pruebas de aceptación de la segunda iteración

Tabla 14. Historias de usuario de la segunda iteración

Tabla 15. Tareas de ingeniería de la tercera iteración

Tabla 16. Pruebas de aceptación de la tercera iteración

Tabla 17. Historia de usuario Acceso al sistema

Tabla 18. Historia de usuario Creación de Proveedores

Tabla 19. Historia de usuario Creación de clientes

Tabla 20. Historia de usuario Creación de productos y servicios

Tabla 21. Historia de usuario Creación de Proyección de Gastos

Tabla 22. Historia de usuario Creación de búsquedas y Filtros

Tabla 23. Historia de usuario Creación de Marca

Tabla 24. Historia de usuario Administración de usuarios

Tabla 25. Historia de usuario Creación de Presupuestos

Tabla 26. Historia de usuario Aprobación de Presupuestos

Tabla 27. Historia de usuario Generación de Reportes de Presupuestos

Tabla 28. Tarea de ingeniería 1 para Historia de usuario 1

Tabla 29. Tarea de ingeniería 2 para Historia de usuario 1

Tabla 30. Tarea de ingeniería 3 para Historia de usuario 1

Tabla 31. Tarea de ingeniería 1 para Historia de Usuario 2

Tabla 32. Tarea de ingeniería 2 para Historia de usuario 2

Tabla 33. Tarea de ingeniería 3 para Historia de usuario 2

Tabla 34. Tarea de ingeniería 1 para historia de usuario 3

Tabla 35. Tarea de ingeniería 2 para historia de usuario 3

Tabla 36. Tarea de ingeniería 3 para historia de usuario 3

Tabla 37. Tarea de ingeniería 1 para Historia de usuario 4

Tabla 38. Tarea de ingeniería 2 para Historia de usuario 4

Tabla 39. Tarea de ingeniería 3 para Historia de usuario 4

Tabla 40. Tarea de ingeniería 4 para Historia de usuario 4

Tabla 41. Tarea de ingeniería 5 para Historia de usuario 4

Tabla 42. Tarea de ingeniería 1 para Historia de usuario 5

Tabla 43. Tarea de ingeniería 2 para Historia de usuario 5

Tabla 44. Tarea de ingeniería 3 para Historia de usuario 5

Tabla 45. Tarea de ingeniería 4 para Historia de usuario 5

Tabla 46. Tarea de ingeniería 1 para Historia de usuario 6

Tabla 47. Tarea de ingeniería 2 para Historia de usuario 6

Tabla 48. Tarea de ingeniería 1 para Historia de usuario 7

Tabla 49. Tarea de ingeniería 2 para Historia de usuario 7

Tabla 50. Tarea de ingeniería 3 para Historia de usuario 7

Tabla 51. Tarea de ingeniería 1 para Historia de usuario 8

Tabla 52. Tarea de ingeniería 2 para Historia de usuario 8

Tabla 53. Tarea de ingeniería 3 para Historia de usuario 8

Tabla 54. Tarea de ingeniería 1 para Historia de usuario 9

Tabla 55. Tarea de ingeniería 2 para Historia de usuario 9

Tabla 56. Tarea de ingeniería 3 para Historia de usuario 9

Tabla 57. Tarea de ingeniería 4 para Historia de usuario 9

Tabla 58. Tarea de ingeniería 1 para Historia de usuario 10

Tabla 59. Tarea de ingeniería 2 para Historia de usuario 10

Tabla 60. Tarea de ingeniería 1 para Historia de usuario 11

Tabla 61. Tarea de ingeniería 2 para Historia de usuario 11

Tabla 62. Caso de prueba acceso al sistema

Tabla 63. Caso de prueba creación de Proveedor

Tabla 64. Caso de prueba edición de Proveedor

Tabla 65. Caso de prueba creación de Clientes

Tabla 66. Caso de prueba edición de Clientes

Tabla 67. Caso de prueba creación de Servicios

Tabla 68. Caso de prueba creación de nueva Proyección de gastos

Tabla 69. Caso de prueba búsquedas en los formularios

Tabla 70. Caso de prueba creación de Marcas

Tabla 71. Caso de prueba edición de Marcas

Tabla 72. Caso de prueba creación de usuarios

Tabla 73. Caso de prueba edición de usuarios

Tabla 74. Caso de prueba creación de presupuestos

Tabla 75. Caso de prueba aprobación de presupuestos

Tabla 76. Caso de prueba generación de reportes de presupuestos

Tabla 77. Tarjeta CRC Ejecutivo

Tabla 78. Tarjeta CRC Cliente

Tabla 79. Tarjeta CRC Marca

Tabla 80. Tarjeta CRC Departamento

Tabla 81. Tarjeta CRC Rubro

Tabla 82. Tarjeta CRC Proveedores

Tabla 83. Tarjeta CRC Proyección de gastos

Tabla 84. Tarjeta CRC Proyección de Categoría

Tabla 85. Tarjeta CRC Proyección de Presupuesto

Tabla 86. Tarjeta CRC Cotizaciones

Tabla 87. Tarjeta CRC Usuario

### **Listado de Figuras**

Figura 1. Fases de la metodología de Programación Extrema.

Figura 2. Mejores Prácticas de XP

Figura 3. Proceso de la primera solicitud ASP .NET

Figura 4. Asignación de un objeto a la solicitud

## **Listado de Imágenes**

Imagen 1. Bosquejo de la pantalla de iniciar Sesión

Imagen 2. Bosquejo de la pantalla Catalogo de Clientes

Imagen 3. Bosquejo de la pantalla Crear nuevo cliente (Información General)

Imagen 4. Bosquejo de la pantalla Crear nuevo cliente (Referencias)

Imagen 5. Bosquejo de la pantalla Crear nuevo cliente (Autorización de Presupuestos)

Imagen 6. Bosquejo de la pantalla Crear de nuevo cliente (Pagos)

Imagen 7. Bosquejo de la pantalla Catalogo de Proveedores

Imagen 8. Bosquejo de la pantalla Crear nuevo Proveedor

Imagen 9. Bosquejo de la pantalla Crear nuevo Proveedor (Contactos)

Imagen 10. Bosquejo de la pantalla Crear nuevo Proveedor (Categorías)

Imagen 11. Pantalla Iniciar sesión

Imagen 12. Pantalla de Página inicial

Imagen 13. Pantalla Catálogo de Clientes

Imagen 14. Pantalla Crear nuevo Cliente (Información General)

Imagen 15. Pantalla Crear nuevo Cliente (Referencias)

Imagen 16. Pantalla Crear nuevo Cliente (Autorización de presupuestos)

Imagen 17. Pantalla Crear nuevo Cliente (Pagos)

Imagen 18. Pantalla Catálogo de Proveedores

Imagen 19. Pantalla Crear nuevo proveedor (Información General)

Imagen 20. Pantalla Crear nuevo proveedor (Contacto)

Imagen 21. Pantalla Crear nuevo proveedor (Categorías)

Imagen 22. Bosquejo de la pantalla Nueva proyección de gastos

Imagen 23. Bosquejo de la pantalla Búsqueda y filtrado

Imagen 24. Bosquejo de la pantalla Catálogo de marcas

Imagen 25. Bosquejo de la pantalla Crear nueva Marca (Información General)

Imagen 26. Bosquejo de la pantalla Crear nueva Marca (Referencias)

Imagen 27. Bosquejo de la pantalla Crear nueva Marca (Autorización de presupuestos)

Imagen 28. Bosquejo de la pantalla Crear nueva Marca (Pagos)

Imagen 29. Pantalla Catalogo de Clientes (mejorada)

Imagen 30. Pantalla Crear nuevo cliente (mejorada)

Imagen 31. Pantalla Catálogo de Proveedores (mejorada)

Imagen 32. Pantalla Crear nuevo Proveedor (mejorada)

Imagen 33. Pantalla buscar y filtrar Cliente

Imagen 34. Pantalla buscar y filtrar Proveedor

Imagen 35. Pantalla buscar y filtrar Marca

Imagen 36. Pantalla catálogo de Marcas

Imagen 37. Pantalla crear nueva Marca

Imagen 38. Pantalla crear nueva Proyección de Gastos

Imagen 39. Bosquejo de la pantalla Administración de usuarios

Imagen 40. Bosquejo de la pantalla nuevo Presupuesto

Imagen 41. Bosquejo de la pantalla catálogo de Presupuestos

Imagen 42. Pantalla reporte de catálogo de Clientes

Imagen 43. Pantalla reporte de catálogo de Proveedores

## **1. RESUMEN**

El presente proyecto está enfocado en aplicar las técnicas de la metodología ágil de Programación Extrema para el desarrollo de un sistema web de Gestión de Presupuestos en la Agencia de Publicidad CEM Comunicaciones S.A. La razón para realizar dicho sistema se debe a la problemática actual del manejo de la información en el área operativa de la Agencia, por lo tanto uno de los objetivos que se ha planteado es el de reducir los tiempos de ejecución de los procesos que se realizan en el departamento, los cuales incluyen la creación de Proyecciones de gastos, creación de Presupuestos o cotizaciones, manejo de compras a proveedores, además de permitir realizar consultas de dicha información en cualquier momento, información que será organizada en reportes.

Para el logro de los objetivos propuestos las técnicas de recolección de datos que se utilizaron fueron las observaciones directas y entrevistas no estructuradas a los colaboradores de la Agencia. Para el desarrollo del sistema se utilizaron diversas tecnologías como el lenguaje de programación ASP.NET y el sistema manejador de base de datos SQL SERVER, así como también las herramientas propias que abarca la metodología de Programación Extrema.

Una vez aplicadas las fases y prácticas de la metodología de Programación Extrema se pudo concluir que con la implantación del nuevo sistema se logra un mejor desempeño de las labores del Departamento debido a la realización de los procesos en forma automatizada.

## **2. INTRODUCCIÓN AL TEMA Y SUBTEMA**

Las aplicaciones de sistemas de información tienen su origen en casi todas las áreas de una empresa y están relacionadas con todos los problemas de la misma. Desde hace muchos años el principal objetivo en el desarrollo de software ha sido diseñarlo e implementarlo en menos tiempo y que su coste menor.

Es por tal razón que en empresas con altos índices de cambio en requerimientos y que no quieren perder calidad en el desarrollo de software eligen la metodologías ágiles como norma para realizar el desarrollo. El origen de las metodologías ágiles está ligado a los constantes inconvenientes que se presentaban en proyectos con ciertas características en los cuales la utilización de la metodologías tradicionales era motivo de fracaso.

En CEM Comunicaciones, agencia de publicidad de Grupo Pellas donde hay un enorme número de requerimientos y cambios por la alta variación del mercado, tendencias y tecnología ha decidido utilizar la metodología ágil de Programación Extrema para desarrollar un sistema de información web que realice de forma automatizada la gestión de presupuestos y que esto ayude a la alta gerencia a la toma de decisiones.

Durante el desarrollo de este proyecto hemos elaborado el presente documento donde se expone la experiencia real obtenida durante la aplicación de la metodología de Programación Extrema. El documento cuenta con cuatro partes principales. La primera parte es el Marco teórico, realizando primeramente una introducción breve de las metodologías ágiles de programación resaltando el manifiesto ágil como su punto de partida, también se hace un recorrido teórico completo de la metodología de Programación Extrema, iniciando con los principios sobre los cuales se basa dicha metodología y finalizando con las mejores prácticas orientadas por la metodología.

La segunda parte es el Marco metodológico en el cual primeramente se hace una breve presentación del cliente y el tipo de negocio donde desarrolló el sistema, también se aborda un poco acerca del desarrollo de sistemas web debido a que el sistema fue desarrollado en este ambiente y finalmente se describen las herramientas de desarrollo de software que se utilizaron para llevar a cabo el proyecto.

La tercera parte muestra la Aplicación de la metodología XP al desarrollo del sistema donde se incluyen las fases del proyecto que son Planificación, Historias de usuarios (Requerimientos), metáfora del Sistema, Asignación de roles y el Plan de entregas.

En la cuarta y última parte del documento se muestra el ciclo de vida del sistema el cual fue dividido en 3 iteraciones, aquí se detalla el proceso que se realizó en cada iteración y los resultados obtenidos.

Al final del documento el lector encontrará las conclusiones de la experiencia que obtuvo el grupo con la aplicación de las técnicas de la metodología de Programación Extrema.

### 3. JUSTIFICACIÓN

Con el fin de conocer a fondo metodologías ágiles para desarrollo de software en particular la Programación Extrema, se implementó un diseño de sistema automatizado que permita ejecutar de manera rápida y eficaz las tareas llevadas a cabo por los trabajadores de la Agencia CEM Comunicaciones S.A, las cuales anteriormente se realizaban de forma manual provocando dificultades y retrasos para la presentación de informes, logrando con este sistema un apoyo directo a la Gerencia en el análisis y toma de decisiones en menor tiempo, además en la actualidad con los avances tecnológicos en el área del manejo de información, no se concibe que pequeñas y sobre todo grandes empresas no tengan automatizados parte o la gran mayoría de sus procesos.

Esta metodología de desarrollo de Programación Extrema es más eficiente y controlable que los métodos tradicionales en general.

Durante los últimos años en la agencia CEM Comunicaciones ha existido la creciente necesidad de obtener la información de manera más ágil y en el momento solicitado, esto ha creado conflictos durante la toma de decisiones gerenciales inmediatas, además el proceso para obtener datos de los diferentes departamentos es complicado y desorganizado. La gerencia obtiene reportes e información sólo de la persona encargada, cabe señalar que si esta persona está fuera de la empresa el flujo de información se interrumpe por lo tanto no se puede tomar ninguna decisión. Además se carece de un registro histórico que contemple los datos revisados en periodos anteriores, es debido a lo antes expuesto que un análisis estadístico sobre ingresos de clientes es difícil y demorado.

La agencia CEM Comunicaciones S. A. requiere automatizar y registrar de forma más efectiva los ingresos mensuales de cada representante de ventas, los cuales son ingresados por rubro y dichos rubros están clasificados por departamentos. El ingreso de estos datos se hace de manera manual, mensualmente cada ejecutivo llena un documento de Excel y luego lo envía por correo a un supervisor, quien une y consolida toda la información con el fin de proporcionar a la gerencia los reportes correspondientes de los ingresos globales e individuales de la empresa.

Basado en el buen resultado que obtiene la empresa al usar un sistema automatizado en el área de contabilidad, se llegó a tomar la decisión de implementar un software que permita darle solución a esta problemática. El SGP (Sistema de Gestión de Presupuestos) a como lo hemos denominado, recibirá como datos de entrada toda la información relacionada para la venta de un servicio así como también los datos de proveedores, clientes y usuarios que trabajaran en el sistema, estos datos se procesaran y serán mostrados en reportes tales como: lista de presupuestos, lista de cotizaciones, proyecciones de gastos, compras a proveedores, presupuestos autorizados, presupuestos anulados, reporte de datos de los clientes y reporte de datos de los proveedores. El SGP hará uso de filtros que faciliten al usuario encontrar la información que requiera.

## 4. OBJETIVOS

### 4.1. Objetivo General

Diseñar un sistema de gestión de presupuestos para la agencia CEM Comunicaciones aplicando las metodologías ágiles de desarrollo de sistemas.

### 4.2. Objetivos Específicos

- Conocer la metodología ágil de Programación Extrema y las técnicas que esta utiliza para el desarrollo de software.
- Identificar las ventajas y desventajas que se obtienen al utilizar la metodología de Programación Extrema.
- Facilitar el proceso de trabajo con la automatización de las tareas de gestión de presupuesto y presentación de informes.

## 5. MARCO TEÓRICO

### 5.1. Origen de la Metodologías Ágiles del Desarrollo de Software

Para tener una visión global y comprender el origen y la razón de ser de las metodologías ágiles hay que remontarse al año 1968, momento en que se le puso nombre a los continuos retrasos, sobrecostes y deficiencias de calidad o utilidad que se producían en el desarrollo de software denominándolo como “Crisis del software”. Un problema que ocurría reiteradamente y que había que rectificar. La solución que se buscó pasaba por:

- El desarrollo de una ingeniería del software. Un cuerpo científico de conocimiento y prácticas especializado.
- La aplicación de la gestión predictiva (clásica) de los proyectos. Basada en la planificación del trabajo, su ejecución y posterior control con el claro objetivo de garantizar el cumplimiento de lo planificado en términos de tiempo, coste y calidad.
- La producción basada en procesos. El objetivo es garantizar la calidad del resultado final a través del uso en la producción de unos procesos de calidad.

La realidad fue que la gestión de proyectos predictiva no funcionaba como debiera en entornos inestables, con gran competencia y cuando los requisitos iniciales cambiaban. Entonces, en 1986, Takeuchi & Nonaka publicaron un artículo llamado “The New Product Development Game” donde daban ejemplos de empresas punteras que estaban obteniendo buenos resultados en ese tipo de entornos y bajo esas circunstancias aplicando metodologías que contravenían la hasta entonces forma de gestionar proyectos. Se empezó a vislumbrar el nacimiento del “agilismo” que surge como contraposición de los modelos de desarrollo basados en procesos.

Estos nuevos métodos de desarrollo surgen como respuesta a un nuevo entorno cambiante, muy competitivo, donde los lanzamientos de productos y las mejoras son cada vez más continuos y se producen en menores intervalos de tiempo y el valor en alza que otorga la ventaja competitiva para estar en los primeros puestos de un sector es la innovación. Autores como el mencionado Ries, además apuntan a que esa innovación ha de producirse en el menor tiempo posible y con el menor esfuerzo necesario, otorgando a la variable tiempo y a la inversión racionalizada una importancia primordial.

La definición moderna de desarrollo ágil de software evolucionó a mediados de la década de 1990 como parte de una reacción contra los métodos muy estructurados y estrictos extraídos del modelo de desarrollo en cascada. El proceso originado del uso del modelo en cascada era visto como burocrático, lento, degradante e inconsistente con las formas de desarrollo de software que realmente realizaban un trabajo eficiente.

Los métodos de desarrollo ágil e iterativo pueden ser vistos como un retroceso a las prácticas observadas en los primeros años del desarrollo de software (aunque en ese tiempo no había metodologías formales). Inicialmente, los métodos ágiles fueron llamados métodos de "peso liviano".

En el año 2001, miembros prominentes de la comunidad practicante de las metodologías ágiles se reunieron en Snowbird, Utah, y adoptaron el nombre de "Métodos ágiles". Poco después, algunas de estas personas formaron la "Alianza ágil", una organización sin fines de lucro que promueve el desarrollo ágil de aplicaciones.

Además de un contacto útil entre los líderes de procesos, existía también la idea de emitir una declaración conjunta en favor de procesos de desarrollo de software ágiles, el resultado es una declaración de los valores y principios comunes de los procesos ágiles conocida como **Manifiesto Ágil** (Principios del manifiesto Ágil, 2001), El manifiesto detalla cuatro valores clave para habilitar equipos de alto rendimiento, los cuales se mencionan a continuación:

- ✓ Valorar más a los individuos y su iteración que a los procesos y las herramientas.
- ✓ Valorar más a el software que funciona que a la documentación exhaustiva.
- ✓ Valorar más la colaboración con el cliente que la negociación contractual.
- ✓ Valorar más la respuesta al cambio que el seguimiento de un plan.

De los valores anteriores se desprenden los principios que constituyen las ideas centrales del desarrollo ágil.

1. Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.
2. Dar la bienvenida a los cambios. Los Métodos Ágiles capturan los cambios para que el cliente tenga una ventaja competitiva.

3. Entregar frecuentemente software que funcione, desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre una entrega y la siguiente.
4. La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto.
5. Construir el proyecto entorno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir el trabajo.
6. El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.
7. El software que funciona es la medida principal de progreso.
8. Los procesos ágiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante.
9. La atención continua a la calidad técnica y al buen diseño mejora la agilidad.
10. La simplicidad es esencial.
11. Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.
12. En intervalos regulares, el equipo reflexiona respecto de cómo llegar a ser más efectivo, y según esto ajusta su comportamiento.

## **5.2. Principales Metodologías Ágiles**

### **SCRUM**

Desarrollada por Ken Schwaber, Jeff Sutherland y Mike. Define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Está especialmente indicada para proyectos con un rápido cambio de requisitos. Sus principales características se pueden resumir en dos: El desarrollo de software se realiza mediante iteraciones denominadas sprints, con una duración de 30 días, el resultado de cada sprint es un incremento ejecutable que se muestra al cliente. La segunda característica importante son las reuniones a lo largo del proyecto, entre ellas destaca la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración (Schwaber, Beedle, Martin, 2001).

### Crystal Methodologies

Se trata de un conjunto de metodologías para el desarrollo de software caracterizadas por estar centradas en las personas que componen el equipo y la reducción al máximo del número de artefactos producidos (Cockburn, 2004, p 336). Han sido desarrolladas por Alistair Cockburn. El desarrollo de software se considera un juego cooperativo de invención y comunicación, limitado por los recursos a utilizar. El equipo de desarrollo es un factor clave, por lo que se deben invertir esfuerzos en mejorar sus habilidades y destrezas, así como tener políticas de trabajo en equipo definidas. Estas políticas dependerán del tamaño del equipo, estableciéndose una clasificación por colores, por ejemplo Crystal Clear (3 a 8 miembros) y Crystal Orange (25 a 50 miembros).

### Dynamic Systems Development Method (DSDM)

Define el marco para desarrollar un proceso de producción de software. Nace en 1994 con el objetivo de crear una metodología RAD (Desarrollo Rápido de Aplicaciones) unificada. Sus principales características son: es un proceso iterativo e incremental y el equipo de desarrollo y el usuario trabajan juntos. Propone cinco fases: estudio de viabilidad, estudio del negocio, modelado funcional, diseño y construcción, y finalmente implementación. Las tres últimas son iterativas, además de existir retroalimentación a todas las fases (Stapleton, 1997).

### Adaptive Software Development (ASD).

Su impulsor fué Jim Highsmith. Sus principales características son: iterativo, orientado a los componentes software más que a las tareas y tolerante a los cambios. El ciclo de vida que propone tiene tres fases esenciales: especulación, colaboración y aprendizaje. En la primera de ellas se inicia el proyecto y se planifican las características del software; en la segunda desarrollan las características y finalmente en la tercera se revisa su calidad y se entrega al cliente. La revisión de los componentes sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo (Highsmith, 2000).

Feature Driven Development (FDD).

Define un proceso iterativo que consta de 5 pasos. Las iteraciones son cortas (hasta 2 semanas). Se centra en las fases de diseño e implementación del sistema partiendo de una lista de características que debe reunir el software. Sus impulsores son Jeff De Luca y Peter Coad.

Lean Development (LD).

Definida por Bob Charette's a partir de su experiencia en proyectos con la industria japonesa del automóvil en los años 80 y utilizada en numerosos proyectos de telecomunicaciones en Europa. En LD, los cambios se consideran riesgos, pero si se manejan adecuadamente se pueden convertir en oportunidades que mejoren la productividad del cliente. Su principal característica es introducir un mecanismo para implementar dichos cambios (Poppendieck, 2003).

Extreme Programming (XP).

Fue la que le dio impulso al movimiento actual de metodologías ágiles. Entre los principios más importantes de esta metodología, se pueden mencionar: en cada iteración se determina el alcance de la próxima iteración combinando prioridades del negocio y estimaciones técnicas, definiéndose la estrategia de planeamiento durante el proceso de desarrollo. Las entregas son frecuentes y continuas. Cada versión liberada debe ponerse en producción rápidamente. El desarrollo es guiado a través de historias de usuario simples. El cliente o un representante del cliente son integrados al equipo de desarrollo, quien está disponible todo el tiempo para responder preguntas del equipo relacionadas con las reglas de negocio. El cliente es responsable por escribir los casos de prueba, demostrando que la funcionalidad está finalizada. Recomienda que el desarrollo de las funciones del producto sea realizado por dos personas en el mismo puesto – programación por pares. Cuando se finaliza la implementación de una funcionalidad, se deben corregir todos los defectos encontrados (Beck, 2000).

### 5.3. Metodología Programación Extrema (XP)

Para realizar este proyecto se nos ha asignado aplicar la Metodología Ágil de Programación Extrema (Extreme Programming, XP), esto con el fin de aprender todas las técnicas que usa dicha metodología y una vez aplicada identificar los beneficios obtenidos durante y posterior al desarrollo del proyecto. A continuación explicaremos un poco más detallado en qué consiste la metodología.

XP (Beck, 1999) es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en retroalimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

### 5.4. Herramientas de la metodología XP

#### **Las Historias de Usuario**

Son la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas. Las historias de usuario son un artefacto de requerimientos de muy alto nivel.

Beck (Beck, 1999) presenta un ejemplo de Historia de usuario con el siguiente contenido:

- Fecha.
- Tipo de actividad (nueva, corrección, mejora).
- Prueba funcional.

- Número de historia.
- Prioridad técnica y del cliente.
- Referencia a otra historia previa.
- Riesgo.
- Estimación técnica.
- Descripción.
- Notas.
- Lista de seguimientos con la fecha.
- Estado
- Cosas por terminar.
- Comentarios

En la Tabla 1 se muestra el formato de una Historia de usuario.

Tabla 1. Formato de Historia de usuario

<b>Historia de Usuario</b>	
<b>Número:</b>	<b>Usuario:</b>
<b>Nombre historia:</b>	
<b>Prioridad en negocio:</b>  Alta / Media/ baja	<b>Riesgo en desarrollo:</b>  Alta/Media/Baja
<b>Puntos estimados:</b>	<b>Iteración asignada:</b>
<b>Programador responsable:</b>	
<b>Descripción:</b>	
<b>Observaciones:</b>	

## Contenido de las Historias de Usuario

### Descripción Escrita

La descripción escrita es sólo una parte muy pequeña de una historia de usuario. La parte más importante es la conversación, donde el usuario explica al desarrollador qué es exactamente lo que quiere y los detalles de la funcionalidad.

El desarrollador puede escribir anotaciones en la parte trasera del post-it, pero si lo hace tiene que ser una pequeña anotación de pocas palabras.

La forma de redactar la descripción escrita para una historia de usuario es:

***Como <rol de usuario>, quiero <función del sistema> para poder <valor de negocio>***

Es importante que en la descripción conste el para qué, esto permite tomar decisiones con respecto al diseño y de cómo resolver el problema. Si el usuario no explica para qué quiere esa funcionalidad, es probable que el desarrollador no entienda toda la perspectiva del requerimiento y no codifique lo que el usuario necesita.

Algunas consideraciones importantes al escribir las historias de usuario:

***El cliente escribe las historias del usuario.***

Los stakeholders (interesados o involucrados en un problema determinado, en este caso el cliente) del proyecto deben escribir las historias de usuario, no los diseñadores. Las historias de usuario son bastante simples, las personas pueden aprender a escribirlas en pocos minutos, no tiene sentido que los expertos las escriban.

***Use la herramienta más simple.***

Las historias del usuario son escritas en tarjetas de índice. Estas tarjetas son muy fáciles trabajar y son por consiguiente una técnica modelada inclusiva.

***Indique el esfuerzo estimado.***

Se debe incluir una estimación del esfuerzo en llevar a cabo la historia de usuario. Una manera de estimar es asignar puntos a cada tarjeta, una indicación relativa de cuánto tiempo tomará que un par de programadores lleven a cabo la historia. El equipo sabe entonces que si actualmente toma 2.5 horas por punto; para una historia que tenga una estimación de 4, tardará alrededor de 10 horas para llevarla a cabo.

***Indique la prioridad.***

Los requisitos, incluso defectos identificados como parte de las actividades de comprobación paralelas independientes o por el esfuerzo de operaciones y soporte, son priorizados por los stakeholders de su proyecto (o representantes del proyecto como los dueños) y agregadas en el lugar apropiado. Se puede mantener fácilmente un conjunto de requisitos priorizados moviendo las tarjetas apropiadamente alrededor. Se puede ver que la tarjeta de historia de usuario incluye una indicación de la prioridad; Se puede usar una escala 1 a 10, siendo 1 la prioridad más alta. Otras priorizaciones son posibles, a menudo se usan prioridades de Alta/Media/Baja en lugar de números y algunas personas asignarán a cada tarjeta un único número de orden de prioridad (por ejemplo 344, 345,...).

Se debe escoger la mejor estrategia que trabaje bien para el equipo. Si la prioridad cambio algún punto en el pasado, es importante que el equipo pueda mover la tarjeta a otro punto en la pila. La estrategia de priorización debe soportar esta clase de actividades.

***Incluya un único identificador.***

Las tarjetas de historias de usuario tienen un único identificador, por ejemplo 173. La razón para hacer esto, es que si necesita podría mantener alguna clase de trazabilidad entre la historia de usuario y otros artefactos, en particular las pruebas de aceptación.

## Tareas de Ingeniería

Las tareas de ingeniería describen las actividades que se realizarán en el proceso descrito en una historia de usuario y se realizan teniendo en cuenta los siguientes datos:

- Está relacionada con el número de historia.
- Se debe poner el nombre de la tarea.
- Los puntos estimados se pondrán dependiendo el tiempo estimado que se llevará en realizar esta tarea.
- Fecha de inicio y fin de la tarea.
- El nombre del programador responsable.
- Descripción de los puntos a tomar en cuenta para realizar la tarea.

En la tabla 2 se muestra el formato de una Tarea de ingeniería.

Tabla 2. Formato de Tarea de Ingeniería

Tarea	
Número tarea:	Número historia:
Nombre tarea:	
Tipo de tarea : Desarrollo / Corrección / Mejora / Otra	Puntos estimados:
Fecha inicio:	Fecha fin:
Programador responsable:	
Descripción:	

## Pruebas de aceptación

En una historia de usuario es importante que quede bien definido cómo se va a aceptar, para esto las Pruebas de Aceptación automatizadas son ideales.

Deben ser automatizadas porque deben “abrazar al cambio” (lema fundamental del manifiesto ágil), se debe reaccionar de forma rápida ante él.

En un negocio donde los cambios se dan constantemente y el código es refactorado (simplificado para hacerlo más flexible y así facilitar los posteriores cambios) todos los días, las pruebas de regresión se multiplican exponencialmente. Para un encargo de pruebas (Tester) sería tediosa y muy susceptible a errores la tarea de probar lo mismo todos los días de forma manual. Cuando un miembro del equipo está conversando con el usuario sobre una historia de usuario, un tipo de anotación que puede hacer en la cara posterior del post-it es el de recordatorio de cómo testear la historia.

El cliente a menudo tiene muchas dificultades al escribir la funcionalidad de las pruebas, como desarrollador debe escribir las pruebas unitarias.

Para obtener una prueba de aceptación se debe seguir los siguientes pasos:

1. Identificar todas las acciones en la historia.
2. Por cada acción escribir dos pruebas.
3. Para algunos datos, suministrar las entradas que debería tener éxito, y llenar cualquier resultado satisfactorio.
4. Para otros datos, suministrar las entradas que tengan una acción fallida, y llenar la respuesta que debería tener.

En la Tabla 3 se muestra el formato de una prueba de aceptación.

Tabla 3. Formato de Prueba de Aceptación

Caso de Prueba	
Número Caso de Prueba:	Número Historia de Usuario:
Nombre Caso de Prueba:	
Descripción:	
Condiciones de ejecución:	
Entradas:	
Resultado esperado:	
Evaluación:	

### Tarjetas CRC

Cada tarjeta CRC representa una clase en la programación orientada a objetos y define sus responsabilidades (lo que ha de hacer) y la colaboración con las otras clases (cómo se comunica con ellas).

En la Tabla 4 se muestra el formato de una Tarjeta CRC

Tabla 4. Formato de Tarjeta CRC

Nombre de la clase.	
Responsabilidades	Colaboradores

### **Cómo crear modelos CRC**

#### **Encontrar las clases.**

Es una tarea fundamental porque se identifica los bloques de nuestra aplicación. Una buena regla es buscar de tres a cinco clases principales. Se pueden incluir otras clases inclusive las que representen a los actores.

#### **Encontrar las responsabilidades.-**

Aquí se debe preguntar qué es lo que hace una clase y así saber cuál es la información que se desea mantener de ésta.

#### **Definir los colaboradores.-**

Una clase a menudo no tiene la información suficiente para cumplir sus responsabilidades. Por consiguiente, debe colaborar (trabajo) con otras clases para conseguir hacer su trabajo. Para identificar los colaboradores para cada responsabilidad debe hacerse la pregunta. ¿Tiene la habilidad para cumplir esta responsabilidad? Si no es así se busca una clase que tenga la habilidad de cumplir la funcionalidad perdida. Haciendo esto descubrirá la necesidad de nuevas responsabilidades en otras clases o a menudo puede necesitar una nueva clase o dos.

## **5.5. Roles de la metodología XP**

Los roles de acuerdo con la propuesta original de Kent Beck (Beck, 1999) son:

- Programador. El programador escribe las pruebas unitarias y produce el código del sistema.
- Cliente. Escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio.

- Encargado de pruebas (Tester). Ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.
- Encargado de seguimiento (Tracker). Proporciona realimentación al equipo. Verifica el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, para mejorar futuras estimaciones. Realiza el seguimiento del progreso de cada iteración.
- Entrenador (Coach). Es responsable del proceso global. Debe proveer guías al equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente.
- Consultor. Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto, en el que puedan surgir problemas.
- Gestor (Big boss). Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación.

## **5.6. Fases de la programación extrema**

La metodología XP se desarrolla en 4 fases:

1. Planeación: Se comienza a interactuar con el cliente y el resto del grupo de desarrollo para descubrir los requerimientos del sistema, se definen los puntos de estimación de las historias de usuario, los criterios de las pruebas de aceptación y se identifica el número y tamaño de las iteraciones.
2. Diseño: Se debe realizar un diseño sencillo pero que cumpla con los requerimientos de las historias de usuario. Se elaboran las tarjetas CRC y los bosquejos, los cuales nos ayudan a obtener los prototipos del sistema.
3. Codificación: En esta fase se emplean los métodos de programación en parejas, se realizan las pruebas unitarias y la integración continua del código.  
  
Integración continua: Se refiere a que cada desarrollador siempre debe estar trabajando en la última versión del software para evitar problemas y retrasos.

- 4. Pruebas de aceptación: Se ejecutan las pruebas de aceptación y se determina si hay errores a corregir.

Una vez aprobadas las pruebas de aceptación se procede con el lanzamiento o puesta en producción, en donde lograremos observar el incremento del software y la velocidad calculada del proyecto la cual se establece en dependencia de cuantas historias de usuario se realizaron en la iteración.

En la Figura 1 podemos ver las fases de la metodología de Programación Extrema.

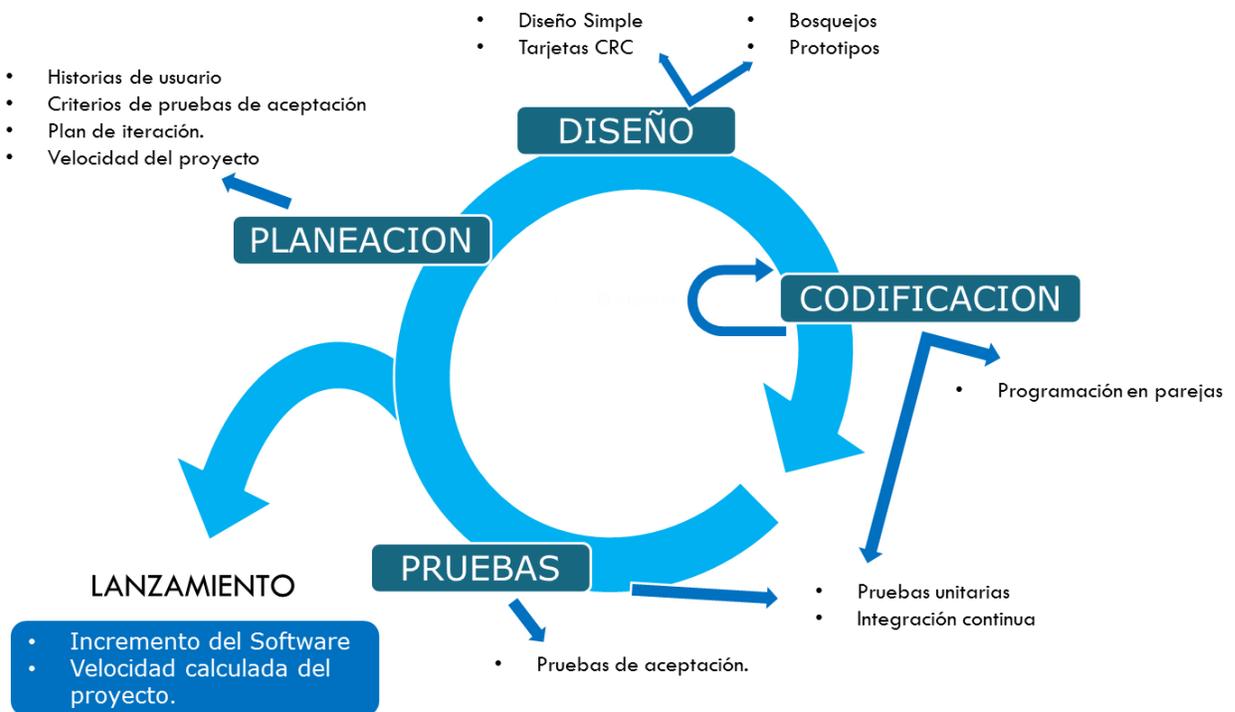


Figura 1. Fases de la metodología de Programación Extrema.

## 5.7. Prácticas de la metodología XP

La principal suposición que se realiza en XP es la posibilidad de disminuir la mítica curva exponencial del costo del cambio a lo largo del proyecto, lo suficiente para que el diseño evolutivo funcione. Esto se consigue gracias a las tecnologías disponibles para ayudar en el desarrollo de software y a la aplicación disciplinada de las siguientes prácticas.

- El juego de la planificación. Hay una comunicación frecuente el cliente y los programadores. El equipo técnico realiza una estimación del esfuerzo requerido para la implementación de las historias de usuario y los clientes deciden sobre el ámbito y tiempo de las entregas y de cada iteración.
- Entregas pequeñas. Producir rápidamente versiones del sistema que sean operativas, aunque no cuenten con toda la funcionalidad del sistema. Esta versión ya constituye un resultado de valor para el negocio. Una entrega no debería tardar más de 3 meses.
- Metáfora. El sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo. Una metáfora es una historia compartida que describe cómo debería funcionar el sistema (conjunto de nombres que actúen como vocabulario para hablar sobre el dominio del problema, ayudando a la nomenclatura de clases y métodos del sistema).
- Diseño simple. Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto.
- Pruebas. La producción de código está dirigida por las pruebas unitarias. Éstas son establecidas por el cliente antes de escribirse el código y son ejecutadas constantemente ante cada modificación del sistema.
- Refactorización (Refactoring). Es una actividad constante de reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible para facilitar los posteriores cambios. Se mejora la estructura interna del código sin alterar su comportamiento externo.

- Programación en parejas. Toda la producción de código debe realizarse con trabajo en parejas de programadores. Esto conlleva ventajas implícitas (menor tasa de errores, mejor diseño, mayor satisfacción de los programadores).
- Propiedad colectiva del código. Cualquier programador puede cambiar cualquier parte del código en cualquier momento.
- Integración continua. Cada pieza de código es integrada en el sistema una vez que esté lista. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día.
- 40 horas por semana. Se debe trabajar un máximo de 40 horas por semana. No se trabajan horas extras en dos semanas seguidas. Si esto sucede, probablemente está ocurriendo un problema que debe corregirse. El trabajo extra desmotiva al equipo.
- Cliente in-situ. El cliente tiene que estar presente y disponible todo el tiempo para el equipo. Éste es uno de los principales factores de éxito del proyecto XP. El cliente conduce constantemente el trabajo hacia lo que aportará mayor valor de negocio y los programadores pueden resolver de manera inmediata cualquier duda asociada. La comunicación oral es más efectiva que la escrita.
- Estándares de programación. XP enfatiza que la comunicación de los programadores es a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación para mantener el código legible.

El mayor beneficio de las prácticas se consigue con su aplicación conjunta y equilibrada puesto que se apoyan unas en otras. Esto se ilustra en la figura 2, donde una línea entre dos prácticas significa que las dos prácticas se refuerzan entre sí. La mayoría de las prácticas propuestas por XP no son novedosas sino que en alguna forma ya habían sido propuestas en ingeniería del software e incluso demostrado su valor en la práctica. El mérito de XP es integrarlas de una forma efectiva y complementarlas con otras ideas desde la perspectiva del negocio, los valores humanos y el trabajo en equipo.

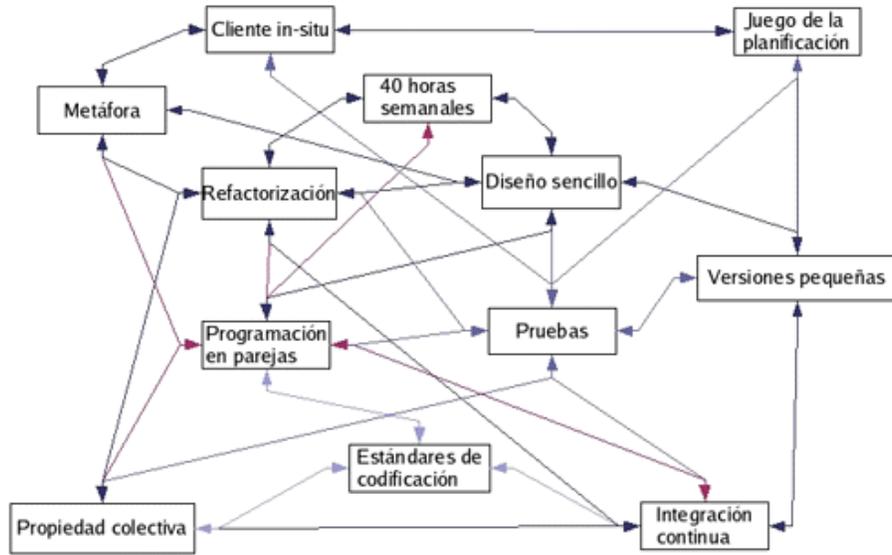


Figura 2. Mejores Prácticas de XP

## 5.8. Acerca de los sistemas WEB

Gracias a la web hoy en día estamos conectados con personas alrededor del mundo, en solo algunos escasos años la web ha evolucionado de forma impresionante. Se ha transformado de aquellas páginas estáticas y sin color que eran un día, a lo que son hoy en día, páginas dinámicas con diseños hermosos gracias a su integración con una variación de lenguajes de programación web.

En poco menos de 10 años ha transformado los sistemas informáticos, ha roto las barreras físicas (distancia), económicas y lógicas (empleo de sistemas web con distintos sistemas operativos y protocolos) y ha abierto un abanico de nuevas posibilidades. Y una de las partes donde más se ha desarrollado es en las aplicaciones web.

Una aplicación Web es aquella que es contactada por un usuario mediante un navegador, usando las capas de comunicación entre el navegador y el servidor web ya sea local (intranet) o en el internet (extranet). El servidor manda las respuestas al cliente, y este las muestra en el navegador. Las aplicaciones basadas en web son programas que corren en servidores web y utilizan páginas web como la interfaz de usuario. Para el usuario promedio este nuevo tipo de software serán más

sencillo, económico, más móvil, más confiable, y a menudo más poderoso que el software de escritorio.” (Graham ,2001).

Cada vez es más necesario adaptar el software y los sistemas online a la lógica de negocio de cada cliente. Por este motivo, no sólo es necesario que las empresas dispongan de una web corporativa, un comercio virtual o una extranet para gestionar los pedidos de sus clientes o distribuidores, sino que es muy aconsejable una integración total con sus sistemas propietarios de facturación, nóminas, proveedores, previsiones, etc. para obtener un sistema integral y eficiente en todas las áreas de su corporación.

### **Ventajas del uso de aplicaciones WEB**

Las aplicaciones web ya sea en el internet o en el intranet presentan una serie de ventajas con respecto al software de escritorio, con lo cual se logra aprovechar y utilizar los recursos de una forma mucho más práctica que el software tradicional.

### **Actualizaciones**

Dado que las aplicaciones web no necesitan ser descargadas, instaladas y configuradas en cada uno de los equipos, un usuario puede acceder de forma remota mediante un navegador Web y comenzar a trabajar sin importar la configuración y el hardware de su equipo. El usuario puede obtener información gerencial desde su Smartphone, notebook con módem 3G, o cualquier otro dispositivo portátil en cualquier lugar del mundo, a cualquier hora. Sin importar al lugar donde vaya mientras tenga acceso a internet tendrá acceso a todos sus documentos y cualquier operación que desea realizar en la aplicación, viendo que hoy en día existe internet casi en cualquier lugar siempre tendrás tus documentos listos para editar, ver y modificarlos.

Como la aplicación web está alojada en un servidor y quienes se encargan de actualizarla son los administradores del sistema, no tendremos que preocuparnos por tener o no la última versión del software y las actualizaciones de seguridad necesarias. Los usuarios solo tendrán que preocuparse de usar la aplicación. 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, esto lleva a un ciclo de actualizaciones mucho más tardada (6-12 meses) causada por los extensos procedimientos de una actualización de una aplicación de escritorio.

### **Seguridad**

Dado que en una aplicación web la información se almacena de forma centralizada y se realizan copias de seguridad de los servidores, los usuarios van a tener mucho menos riesgo de perder sus datos ocasionado por un daño en sus discos duros o un virus en su computador siempre usarán la versión más actualizada y sincronizada.

### **Compatibilidad**

Gracias a las aplicaciones web ya no tienes que dejar de usar tu sistema favorito. Windows, Mac, Linux o cualquier otro sistema operativo pueden ser usados para acceder a una aplicación web. “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.” (Alan, 2013)

### **Errores y Fallas**

Las aplicaciones web son menos propensas a crear problemas técnicos debido a conflictos con el software, hardware o con otras aplicaciones existentes. Con las aplicaciones web todos los usuarios utilizan la misma versión y por esto motivo los posibles bugs (fallas) pueden ser corregidos tan pronto como sean descubiertos, y ningún usuario habrá actualizado individualmente la aplicación. En un pequeño estudio de un desarrollador se encontró que de 50 clientes pidiendo soporte técnico 15 de ellos pedían apoyo para aplicaciones de escritorio mientras que solo 3 pedían ayuda para versiones Web. La mayoría de ellos pedían soporte por las siguientes razones: errores de instalación, pérdida de código de registración y bugs encontrados en versiones antiguas del programa encontradas en páginas de descarga. ¿Porque esto no existe en las aplicaciones web? No hay problemas de instalación porque no existe la instalación. Y no habrá problema de bugs por que los usuarios siempre tendrán acceso a la versión más actualizada.

## 5.9. Herramientas para el desarrollo de Software

### MICROSOFT SQL SERVER

El lenguaje de consulta estructurado o SQL (por sus siglas en inglés Structured Query Language) es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en éstas. Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo efectuar consultas con el fin de recuperar de una forma sencilla información de interés de una base de datos, así como también hacer cambios sobre ella.

El SQL es un lenguaje de acceso a bases de datos que explota la flexibilidad y potencia de los sistemas relacionales permitiendo gran variedad de operaciones en éstos últimos, por esta razón hemos elegido SQL como nuestro manejador de Base de datos.

Es un lenguaje declarativo de "alto nivel" o "de no procedimiento", que gracias a su fuerte base teórica y su orientación al manejo de conjuntos de registros, y no a registros individuales, permite una alta productividad en codificación y la orientación a objetos. De esta forma una sola sentencia puede equivaler a uno o más programas que se utilizarían en un lenguaje de bajo nivel orientado a registros.

### Componentes del SQL

El lenguaje SQL está compuesto por comandos, cláusulas, operadores y funciones de agregado. Estos elementos se combinan en las instrucciones para crear, actualizar y manipular las bases de datos.

### Comandos

Existen dos tipos de comandos SQL:

- ✓ Los DDL (Lenguaje de Definición de Datos) que permiten crear y definir nuevas bases de datos, campos e índices.

- ✓ Los DML (Lenguaje de Manipulación de Datos) que permiten generar consultas para ordenar, filtrar y extraer datos de la base de datos.

### **Comandos DDL**

CREATE: Utilizado para crear nuevas tablas, campos e índices.

DROP: Empleado para eliminar tablas e índices.

ALTER: Utilizado para modificar las tablas agregando campos o cambiando la definición de los campos.

### **Comandos DML**

SELECT: Utilizado para consultar registros de la base de datos que satisfagan un criterio determinado.

INSERT: Utilizado para cargar lotes de datos en la base de datos en una única operación.

UPDATE: Utilizado para modificar los valores de los campos y registros especificados.

DELETE: Utilizado para eliminar registros de una tabla de una base de datos.

### **Cláusulas**

Las cláusulas son condiciones de modificación utilizadas para definir los datos que desea seleccionar o manipular.

FROM: Utilizada para especificar la tabla de la cual se van a seleccionar los registros.

WHERE: Utilizada para especificar las condiciones que deben reunir los registros que se van a seleccionar.

GROUPBY: Utilizada para separar los registros seleccionados en grupos.

HAVING: Utilizada para expresar la condición que debe satisfacer cada grupo.

ORDERBY: Utilizada para ordenar los registros seleccionados de acuerdo con un orden específico.

### **Operadores Lógicos**

AND: Es el "y" lógico. Evalúa dos condiciones y devuelve un valor de verdad sólo si ambas son ciertas.

OR: Es el "o" lógico. Evalúa dos condiciones y devuelve un valor de verdad si alguna de las dos es cierta.

NOT: Negación lógica. Devuelve el valor contrario de la expresión.

### **Operadores de Comparación**

< Menor que

> Mayor que

<> Distinto de

<= Menor o Igual que

>= Mayor o Igual que

= Igual que

BETWEEN: Utilizado para especificar un intervalo de valores.

LIKE: Utilizado en la comparación de un modelo

IN: Utilizado para especificar registros de una base de datos

### **Funciones de Agregado**

Las funciones de agregado se usan dentro de una cláusula SELECT en grupos de registros para devolver un único valor que se aplica a un grupo de registros.

AVG: Utilizada para calcular el promedio de los valores de un campo determinado.

COUNT: Utilizada para devolver el número de registros de la selección.

SUM: Utilizada para devolver la suma de todos los valores de un campo determinado.

MAX: Utilizada para devolver el valor más alto de un campo especificado.

MIN: Utilizada para devolver el valor más bajo de un campo especificado.

### **VISUAL STUDIO 2010**

Visual Studio es un conjunto de herramientas de desarrollo basadas en componentes y otras tecnologías para compilar aplicaciones eficaces de alto rendimiento. Además, Visual Studio está optimizado para el diseño, el desarrollo y la implementación en equipo de soluciones empresariales.

### **ASP .NET**

ASP.NET es un modelo de desarrollo Web unificado que incluye los servicios necesarios para crear aplicaciones Web empresariales con el código mínimo. ASP.NET forma parte de .NET Framework y al codificar las aplicaciones ASP.NET tiene acceso a las clases en .NET Framework. El código de las aplicaciones puede escribirse en cualquier lenguaje compatible con el Common Language Runtime (CLR), entre ellos Microsoft Visual Basic y C#. Estos lenguajes permiten desarrollar aplicaciones ASP.NET que se benefician del Common Language Runtime, seguridad de tipos, herencia, etc. Para el desarrollo de nuestro proyecto hemos hecho uso del lenguaje Visual Basic.

### **Proyectos de sitios y aplicaciones web ASP.NET**

Con Visual Studio 2010, se puede crear diferentes tipos de proyectos ASP.NET, que incluyen sitios web, aplicaciones web, servicios Web y controles de servidor AJAX.

Hay una diferencia entre los proyectos de sitios web y los proyectos de aplicaciones web. Algunas características solo funcionan con proyectos de aplicaciones web, como MVC y ciertas herramientas para automatizar la implementación web. Otras características, como los datos dinámicos, funcionan tanto con los proyectos de sitios web como con los proyectos de aplicaciones web.

### **Referencia de ASP.NET API**

A continuación se indican algunos de los espacios de nombres más importantes de la biblioteca de clases de .NET Framework que pertenecen a ASP.NET:

#### **System.Web**

Proporciona clases e interfaces que habilitan la comunicación entre el explorador y el servidor. Este espacio de nombres incluye la clase `HttpRequest`, que proporciona amplia información sobre la solicitud HTTP actual; la clase `HttpResponse`, que administra la salida HTTP al cliente; y la clase `HttpServerUtility`, que proporciona acceso a las utilidades y procesos del servidor.

`System.Web` también incluye clases para la manipulación de cookies, la transferencia de archivos, la información de excepciones y el control de la memoria caché de resultados.

#### **System.Web.ApplicationServices**

Proporciona clases que proporcionan acceso a los servicios de autenticación de formularios, funciones y aplicaciones de perfiles de ASP.NET como servicios de Windows Communication Foundation (WCF).

#### System.Web.Handlers

Contiene clases del controlador HTTP que procesan las solicitudes HTTP a un servidor Web. Una página de formularios Web Forms de ASP.NET (archivo .aspx) es un formulario especial de un controlador HTTP.

#### System.Web.Services

Se compone de las clases que permiten crear servicios Web XML mediante clientes de servicios Web XML y ASP.NET. Los servicios Web XML son aplicaciones que permiten intercambiar mensajes en un entorno de correspondencia imprecisa mediante protocolos estándar como HTTP, XML, XSD, SOAP y WSDL. Los servicios Web XML permiten compilar aplicaciones modulares que son interoperables en una amplia variedad de implementaciones, plataformas y dispositivos.

#### System.Web.UI

Proporciona clases e interfaces que permiten crear controles de servidor ASP.NET y páginas web ASP.NET para la interfaz de usuario de las aplicaciones web ASP.NET. Este espacio de nombres incluye la clase Control, que proporciona a todos los controles de servidor HTML, los controles de servidor web y los controles de usuario un conjunto de funciones comunes. También incluye el control Page, que se genera automáticamente siempre que se realiza una solicitud de un archivo .aspx en una aplicación Web ASP.NET. También incluye clases que proporcionan a los controles de servidor funciones de enlace de datos, la capacidad de guardar el estado de vista de un determinado control o página y la función de análisis.

#### **Marco de trabajo de páginas y controles**

El marco de trabajo de páginas y controles ASP.NET es un marco de trabajo de programación que se ejecuta en un servidor web para generar y representar de forma dinámica páginas Web ASP.NET. Las páginas Web ASP.NET se pueden solicitar a cualquier explorador o dispositivo del cliente y ASP.NET representa el marcado (como HTML) al explorador que realizó la solicitud. Como norma, puede utilizar la misma página para varios exploradores, porque ASP.NET representa el marcado adecuado para el explorador que realiza la solicitud. Sin embargo, puede diseñar una

página Web ASP.NET para ejecutarse en un explorador determinado y aprovechar así todas las características de ese explorador.

Las páginas Web ASP.NET están completamente orientadas a objetos. En las páginas Web ASP.NET se puede trabajar con elementos HTML que usen propiedades, métodos y eventos. El marco de trabajo de páginas ASP.NET quita los detalles de implementación relacionados con la separación de cliente y servidor inherente a las aplicaciones Web presentando un modelo unificado que responde a los eventos de los clientes en el código que se ejecuta en el servidor. El marco de trabajo también mantiene automáticamente el estado de la página y de los controles que contenga durante el ciclo vital de procesamiento de la página.

### **Compilador de ASP.NET**

Compila todo el código de ASP.NET, lo que permite el establecimiento inflexible de tipos, las optimizaciones de rendimiento y el enlace en tiempo de compilación, entre otras ventajas. Una vez que se ha compilado el código, el Common Language Runtime compila una vez más código de ASP.NET en código nativo, lo que permite un mayor rendimiento.

### **Infraestructura de seguridad**

Además de las características de seguridad de .NET, ASP.NET proporciona una infraestructura de seguridad avanzada para autenticar y autorizar el acceso de los usuarios y realizar otras tareas relacionadas con la seguridad. Puede autenticar usuarios con la autenticación de Windows suministrada por IIS o puede administrar la autenticación con su propia base de datos de usuario utilizando la autenticación mediante formularios ASP.NET y la pertenencia de ASP.NET.

### **Ciclo de vida de una aplicación ASP.NET**

En ASP.NET, deben producirse varios pasos de procesamiento para que una aplicación ASP.NET se inicialice y procese las solicitudes. Además, ASP.NET es sólo una parte de la arquitectura de servidor web que atiende las solicitudes realizadas por los exploradores. Es importante que

---

**LA PROGRAMACIÓN EXTREMA APLICADA AL DESARROLLO DEL SISTEMA DE INFORMACIÓN WEB PARA LA GESTIÓN DE PRESUPUESTOS EN CEM COMUNICACIONES S.A.**

comprender el ciclo de vida de la aplicación para que se pueda escribir código en la fase apropiada del ciclo y conseguir el efecto deseado.

En la tabla 5 se describen las fases del ciclo de vida de la aplicación ASP.NET.

Tabla 5. Ciclo de vida de una aplicación ASP .NET

Fase	Descripción
El usuario solicita un recurso de aplicación del servidor web.	El ciclo de vida de una aplicación ASP.NET se inicia con una solicitud enviada por un explorador al servidor web (para las aplicaciones ASP.NET, normalmente es IIS). ASP.NET es una extensión ISAPI bajo el servidor web. Cuando un servidor web recibe una solicitud, examina la extensión de nombre de archivo del archivo solicitado, determina la extensión ISAPI que debería procesar dicha solicitud y, a continuación, pasa ésta a la extensión ISAPI apropiada. ASP.NET procesa las extensiones de nombre de archivo que tiene asignadas, como .aspx, .ascx, .ashx y .asmx.
ASP.NET recibe la primera solicitud para la aplicación.	Cuando ASP.NET recibe la primera solicitud para cualquier recurso de una aplicación, una clase denominada Application Manager crea un dominio de aplicación. Los dominios de aplicación proporcionan aislamiento entre aplicaciones para las variables globales y permiten descargar cada aplicación de forma independiente. Dentro de un dominio de aplicación, se crea una instancia de la clase denominada HostingEnvironment, que proporciona acceso a la información sobre la aplicación, como el nombre de la carpeta en la que está almacenada la aplicación.  En la siguiente figura 3 se muestra esta relación:

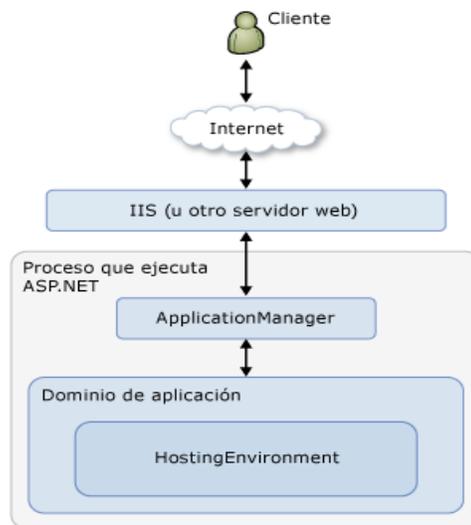


Figura 3. Proceso de la primera solicitud ASP .NET

Se crean los objetos de núcleo ASP.NET para cada solicitud.

Una vez creados el dominio de aplicación y una instancia del objeto HostingEnvironment, ASP.NET crea e inicializa objetos de núcleo, como HttpContext, HttpRequest y HttpResponse.

La clase HttpContext contiene objetos específicos de la solicitud de aplicación actual, como los objetos HttpRequest y HttpResponse. El objeto HttpRequest contiene datos sobre la solicitud actual, entre los que se incluyen las cookies e información del explorador. El objeto HttpResponse contiene la respuesta que se envía al cliente, la cual incluye todos los resultados representados y las cookies.

Se asigna un objeto HttpRequest a la solicitud.

Una vez que se han inicializado todos los objetos principales de la aplicación, ésta se inicia creando una instancia de la clase HttpRequest. Si la aplicación tiene un archivo Global.asax, ASP.NET crea una instancia de la clase Global.asax derivada de la

clase `HttpApplication` y la utiliza para representar la aplicación.

En la figura 4 se muestra esta relación:

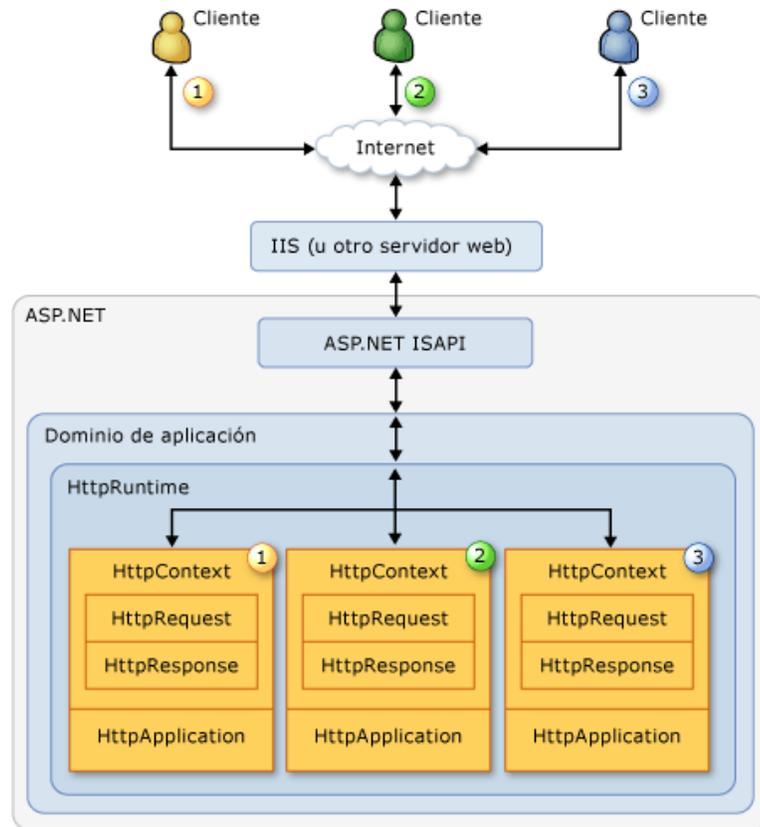


Figura 4. Asignación de un objeto a la solicitud

Canalización de  
HttpApplication  
procesa la  
solicitud.

La clase `HttpApplication` ejecuta los eventos siguientes mientras se procesa la solicitud. Dichos eventos son de particular interés para los desarrolladores que desean extender la clase `HttpApplication`.

1. Valida la solicitud, que examina la información enviada por el explorador y determina si contiene formato potencialmente malintencionado
2. Realiza la asignación de direcciones URL si se ha configurado alguna dirección URL en la sección `Url Mappings Section` del archivo `Web.config`.
3. Genera el evento `BeginRequest`.
4. Genera el evento `AuthenticateRequest`.
5. Genera el evento `PostAuthenticateRequest`.
6. Genera el evento `AuthorizeRequest`.
7. Genera el evento `PostAuthorizeRequest`.
8. Genera el evento `ResolveRequestCache`.
9. Genera el evento `PostResolveRequestCache`.
10. Basándose en la extensión de nombre de archivo del recurso solicitado (asignada en el archivo de configuración de la aplicación), selecciona una clase que implemente `IHandler` para procesar la solicitud. Si la solicitud es para un objeto (página) derivado de la clase `Page` y es necesario compilar la página, ASP.NET la compila antes de crear una instancia de ella.
11. Genera el evento `PostMapRequestHandler`.
12. Genera el evento `AcquireRequestState`.

13. Genera el evento PostAcquireRequestState.
14. Genera el evento PreRequestHandlerExecute.
15. Llama al método ProcessRequest de la clase IHttpHandler apropiada para la solicitud. Por ejemplo, si la solicitud es para una página, la controla la instancia de la página actual.
16. Genera el evento PostRequestHandlerExecute.
17. Genera el evento ReleaseRequestState.
18. Genera el evento PostReleaseRequestState.
19. Realiza el filtrado de respuestas si se define la propiedad Filter.
20. Genera el evento UpdateRequestCache.
21. Genera el evento PostUpdateRequestCache.
22. Genera el evento EndRequest.
23. Genera el evento PreSendRequestHeaders.
24. Genera el evento PreSendRequestContent.

## 6. MARCO METODOLÓGICO

### 6.1. Tipo de investigación

Debido a la naturaleza del presente proyecto, se ha realizado el tipo de investigación de campo, ya que como su definición lo indica este tipo de investigación constituye un proceso sistemático, riguroso y racional de recolección, tratamiento, análisis y presentación de datos, basado en una estrategia de recolección directa de las informaciones necesarias para la investigación. Para el diseño del sistema de Gestión de presupuestos se inicia con una investigación preliminar del ciclo de vida del desarrollo del software y luego con la aplicación de las fases de la metodología XP.

### 6.2. Diseño de la investigación

Diseño experimental: En ellos el investigador desea comprobar los efectos de una intervención específica, en este caso el investigador tiene un papel activo, pues lleva a cabo una intervención.

En nuestro proyecto aplicamos este tipo de diseño ya que el equipo de desarrollo siempre tuvo un papel activo realizando una intervención con el desarrollo del sistema de gestión de presupuestos para comprobar los efectos que causaba este en la agencia Cem Comunicaciones S.A.

### 6.3. Fuentes de información

Se define como fuente de información a los materiales o productos, originales o elaborados, que aportan noticias o testimonios a través de los cuales se accede al conocimiento, cualquiera que este sea” (Carrizo, 2000:38). Esta definición permite considerar como fuente de información cualquier objeto o situación que nos transmita algo que, tras su interpretación y elaboración, genera conocimiento. Las fuentes de información se dividen en dos: las fuentes Primarias y las fuentes secundarias.

Las fuentes primarias contienen artículos o informes que exponen observaciones originales o los resultados de la investigación experimental o de campo, los cuales comprenden contribuciones nuevas al conocimiento.

Las Fuentes Secundarias son registros que proceden también de un contacto con la práctica, pero que ya han sido procesados por otros investigadores.

#### 6.4. Instrumentos de Recolección de Datos

Un instrumento de recolección de datos es cualquier recurso del que pueda valerse el investigador para acercarse a los fenómenos y extraer de ellos información. Existen dos tipos de técnicas para la recolección de datos las cuales se dividen en: Técnicas para datos primarios y Técnicas para datos secundarios.

La técnica para la recolección de datos primarios utilizada en el desarrollo de este proyecto fue la entrevista, que es una técnica mediante la cual una persona obtiene información directa de otra, en nuestro caso se entrevistaron a las personas que integran el departamento de operaciones de la agencia Cem comunicaciones, ya que estas tienen un conocimiento a profundidad del proceso que se lleva a cabo para la elaboración de los presupuestos a los clientes.

En el caso de la recolección de datos secundarios se utilizaron los artículos y libros que sirvieron de apoyo para fortalecer el conocimiento que tenía el equipo acerca de la metodología de programación extrema.

## 7. APLICACIÓN DE XP AL DESARROLLO DEL SISTEMA

### 7.1. Descripción del Cliente.

CEM Comunicaciones S.A es una agencia de comunicación integral, nació en 1992 como parte del grupo Pellas y en 2006 se alió estratégicamente con JWT para brindar servicios profesionales en el campo publicitario.

Entre los servicios que ofrece la agencia están:

- ✓ Planeamiento Estratégico Creativo
- ✓ Ingeniería Gráfica Creativa
- ✓ Relaciones Públicas Digitales
- ✓ Diseño Gráfico
- ✓ Planeamiento Estratégico de Medios
- ✓ Producción Audiovisual
- ✓ Servicios de BTL
- ✓ Planeamiento Estratégico de Campañas
- ✓ Comunicación Digital e Interactiva
- ✓ Producción e Impresos

## 7.2. Planificación

Sistema de Gestión de Presupuestos para la agencia CEM COMUNICACIONES S.A.

Se desea que el SGP (Sistema de Gestión de Presupuestos) gestione los presupuestos, lista de cotizaciones, proyecciones de gastos, compras a proveedores, presupuestos autorizados, presupuestos anulados, reporte de datos de los clientes y reporte de datos de los proveedores.

Para entrega de este proyecto nos estaremos centrando en siguientes módulos:

- ✓ Presupuestos
- ✓ Servicios
- ✓ Proveedores
- ✓ Clientes
- ✓ Departamentos
- ✓ Administración de usuarios.

Para esto al reunirnos con el usuario se han definido las siguientes historias de usuario.

## 7.3. Historias de Usuario

A continuación se muestra un listado de las historias de usuarios para el Sistema de Gestión de Presupuestos para CEM COMUNICACIONES S.A.

- ✓ Acceso al sistema
- ✓ Creación de Proveedores
- ✓ Creación de Clientes
- ✓ Creación de Servicios
- ✓ Creación de Proyección de Gastos
- ✓ Creación de búsquedas y filtros
- ✓ Administración de Usuarios
- ✓ Creación de Presupuestos
- ✓ Aprobación de Presupuestos
- ✓ Generación de Reportes de Presupuestos

#### 7.4. Metáfora del Sistema de Gestión de Presupuestos.

La empresa CEM COMUNICACIONES S.A registra los presupuestos emitidos y los estados en que se encuentra cada uno de ellos como anulado, aprobado y pendiente; los presupuestos son emitidos con diferentes tipos de servicios de diferentes proveedores, todo servicio o producto que está en el presupuesto contiene la comisión de la agencia. El contrato como parte del compromiso que toma el cliente con la agencia es realizado a partir de un presupuesto, el mismo muestra todos los datos que tiene el presupuesto pero con la salvedad que esté en su encabezado muestra los campos de contrato con sus políticas impresas en su físico.

Los productos y servicios son aquellos que la agencia ofrece y que los mismos son distribuidos por los proveedores que tiene la agencia como alianza estratégica. La comisión de ganancia de la agencia se calcula automáticamente y esta solo puede ser modificada por el jefe de área o a la persona que designe el mismo.

Los departamentos son las divisiones productivas de la agencia los cuales se ven reflejados en la facturación de productos y servicios.

Se deberá obtener reportes tales como: lista de presupuestos, lista de cotizaciones, proyecciones de gastos, compras a proveedores, presupuestos autorizados, presupuestos anulados, reporte de datos de los clientes y reporte de datos de los proveedores.

También deberá existir un módulo con seguridad de acceso donde solo usuarios con perfil de administrador puedan acceder para realizar la creación de usuarios y sus contraseñas iniciales y adicionalmente se pueda cambiar los datos de configuración principales como el nombre de la compañía y sus datos comunes, así también podrán hacer modificaciones al tipo de cambio de la moneda actual para todo el sistema.

## 7.5. Asignación de Roles

En la tabla 6 se muestran los roles que fueron asignados a cada integrante en el desarrollo del proyecto.

Tabla 6. Asignación de Roles del Proyecto

Roles	Asignado a:
<b>Programador</b>	Engels Obregón, Marco Fonseca
<b>Cliente</b>	Geovani Geanoli, Jairo Narváez
<b>Encargado de pruebas (Tester)</b>	Leonte Espinoza
<b>Encargado de seguimiento (Tracker)</b>	Leonte Espinoza
<b>Entrenador (Coach)</b>	Marco Fonseca
<b>Consultor</b>	Marco Fonseca, Engels Obregón, Leonte Espinoza
<b>Gestor (Big Boss)</b>	Marco Fonseca

## 7.6. Plan de Entregas.

De acuerdo a las historias de usuarios consideradas para el desarrollo del sistema, se ha realizado el siguiente plan de entrega, el cual muestra que historias de usuario se llevaran a cabo en cada iteración. Este plan se ha desarrollado tomando en cuenta el esfuerzo y la prioridad de cada historia de usuario.

En la tabla 7 se detalla el plan de entrega del proyecto.

Tabla 7. Plan de entrega del proyecto

HISTORIA	ITERACIÓN	PRIORIDAD	ESFUERZO	FECHA DE INICIO	FECHA FINAL
HISTORIA 1	1	MEDIA	2	16-dic-13	21-dic-13
HISTORIA 2	1	ALTA	4	23-dic-13	30-dic-13
HISTORIA 3	1	ALTA	2	02-ene-14	20-ene-14
HISTORIA 4	1	ALTA	6	21-ene-14	15-feb-14
HISTORIA 5	2	ALTA	4	16-feb-14	27-feb-14

HISTORIA 6	2	MEDIA	10	28-feb-14	03-mar-14
HISTORIA 7	3	ALTA	8	05-mar-14	15-mar-14
HISTORIA 8	3	ALTA	9	16-mar-14	05-abr-14
HISTORIA 9	3	ALTA	10	06-abr-14	28-abr-14
HISTORIA 10	3	ALTA	8	29-abr-14	03-may-14
HISTORIA 11	3	ALTA	10	04-may-14	20-may-14

## **8. CICLO DE VIDA DEL SISTEMA DE GESTIÓN DE PRESUPUESTOS.**

### **8.1. Primera Iteración**

Para esta iteración se han desarrollado los módulos de Acceso al Sistema, Proveedores, Clientes y Servicios, para esto se han utilizado y aplicado las herramientas que destaca la metodología XP.

A continuación en las tablas 8, 9, 10 y en las imágenes 1-10 se muestra un resumen de las herramientas utilizadas en esta iteración y por último en las imágenes 11-21 se presentan las capturas de pantalla de los módulos desarrollados. En la sección de Resultados del documento se detalla cada una de estas herramientas, en el Resultado 1 las Historias de usuarios, en el Resultado 2 las tareas de ingeniería y en el Resultado 3 las pruebas de aceptación.

Tabla 8. Historias de usuario de la primera iteración

<b>Número</b>	<b>Nombre</b>
1	Acceso al Sistema
2	Creación de Proveedores
3	Creación de Clientes
4	Creación de Servicios

Tabla 9. Tareas de ingeniería de la primera iteración

<b>Numero de Tarea</b>	<b>Número de Historia</b>	<b>Nombre de la Tarea</b>
1	1	Diseño de interfaz del acceso al sistema.
2	1	Validación de usuarios en la BD.
3	1	Adaptación de la Base de datos para los usuarios.
4	2	Diseñar interfaz de usuario para la creación de proveedores.
5	2	Diseño de la base de datos de Proveedores.
6	2	Grabar Proveedores en la Base de Datos.
7	3	Creación de Interfaz usuario para la creación de clientes.
8	3	Diseño de base de datos de clientes
9	3	Grabar Clientes en la Base de Datos.
10	4	Registro de servicios impresos de la agencia.
11	4	Escoger automáticamente el proveedor del servicio.
12	4	Actualizar la comisión de la agencia.
13	4	Adaptación de la Base de Datos para la grabación de Servicios.
14	4	Guardar los servicios en la Base de Datos.

Tabla 10. Pruebas de aceptación de la primera iteración

<b>Numero de Prueba</b>	<b>Número de Historia</b>	<b>Nombre de la Prueba</b>
1	1	Acceso al Sistema
2	2	Creación de Proveedor
3	2	Edición de proveedor
4	3	Creación de Clientes
5	3	Edición de Clientes
6	2	Creación de Servicios

Bosquejos

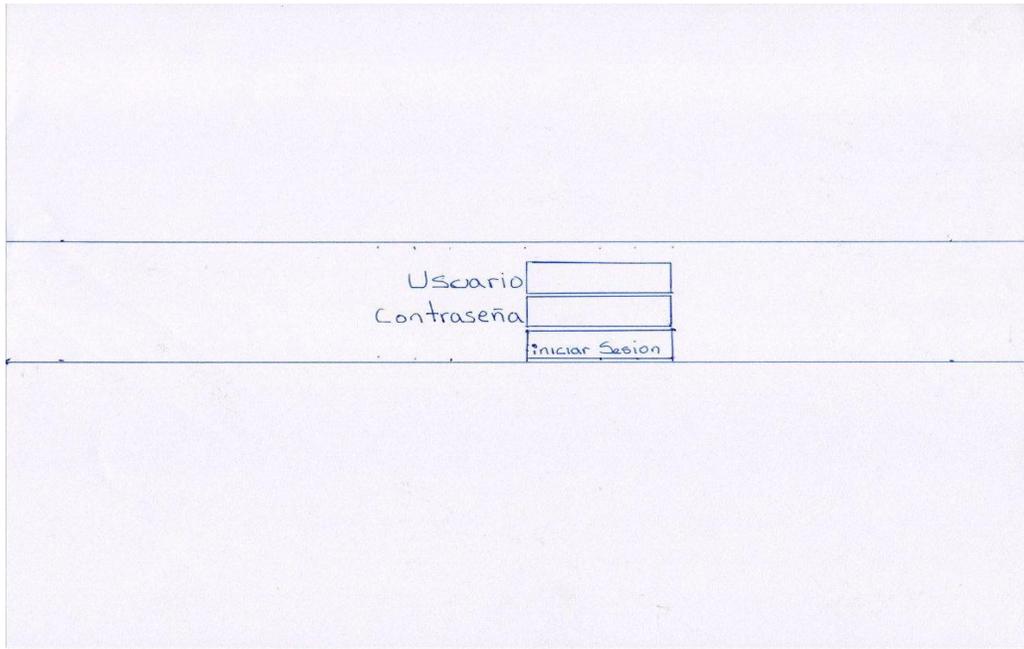


Imagen 1. Bosquejo de la pantalla iniciar Sesión

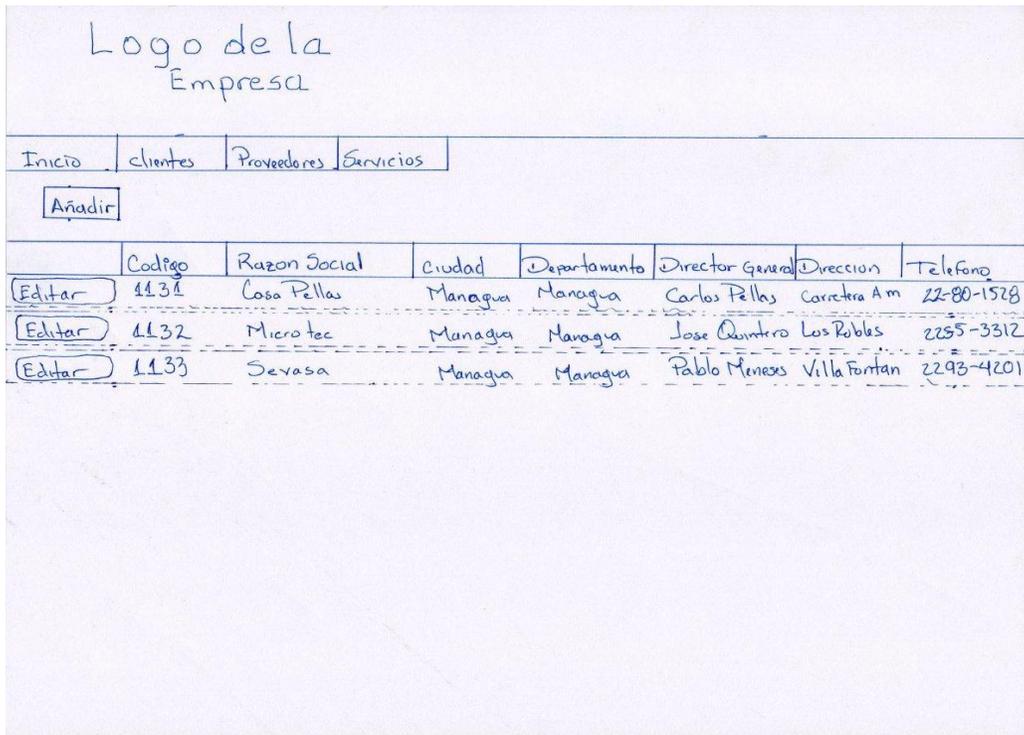


Imagen 2. Bosquejo de la pantalla Catálogo de Clientes

Logo de la Empresa

Inicio	clientes	Proveedores	Servicios
--------	----------	-------------	-----------

Guardar Actualizar Cerrar

Código Cliente

Razon Social

Representante legal

Ciudad  Departamento  Código Postal

Director General

Numero Ruc

Giro de la Empresa  Años de Operación

Dirección

Telefono  Fax

Correo Electronico

Información General	Referencias	Autorización de Presupuestos	Pagos
---------------------	-------------	------------------------------	-------

Imagen 3. Bosquejo de la pantalla Crear nuevo cliente (Información General)

Logo de la Empresa

Inicio	clientes	Proveedores	Servicios
--------	----------	-------------	-----------

Guardar Actualizar Cerrar

Referencias Bancarias

1

2

3

Referencias Comerciales

	Compañía	Telefono
1	<input type="text"/>	<input type="text"/>
2	<input type="text"/>	<input type="text"/>
3	<input type="text"/>	<input type="text"/>

Información General	Referencias	Autorización de Presupuestos	Pagos
---------------------	-------------	------------------------------	-------

Imagen 4. Bosquejo de la pantalla Crear nuevo cliente (Referencias)

Logo de la Empresa

Inicio	clientes	Proveedores	servicios
--------	----------	-------------	-----------

Nombre	<input type="text"/>
Cedula	<input type="text"/>
Cargo que desempeña	<input type="text"/>
Telefono	<input type="text"/>
Celular	<input type="text"/>
Correo Electronico	<input type="text"/>

Informacion General	Referencias	Autorización de Presupuestos	Pagos
---------------------	-------------	------------------------------	-------

Imagen 5. Bosquejo de la pantalla Crear de nuevo cliente (Autorización de Presupuestos)

Logo de la Empresa

Inicio	clientes	Proveedores	servicios
--------	----------	-------------	-----------

Dias de Pago	<input type="text"/>
Encargado de Pagos	<input type="text"/>
Horario de Pagos	<input type="text"/>
Telefono Ejecutivo del depto de Contabilidad	<input type="text"/>
Correo Electronico	<input type="text"/>

Informacion General	Referencias	Autorización de Presupuesto	Pagos
---------------------	-------------	-----------------------------	-------

Imagen 6. Bosquejo de la pantalla Crear de nuevo cliente (Pagos)

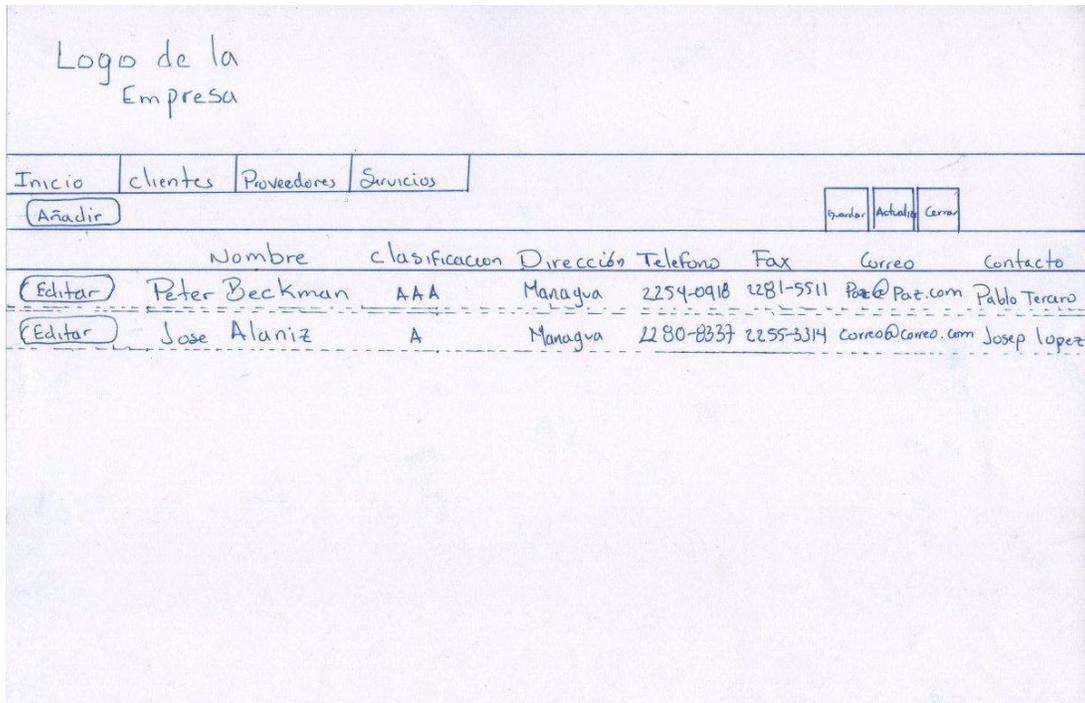


Imagen 7. Bosquejo de la pantalla Catálogo de Proveedores

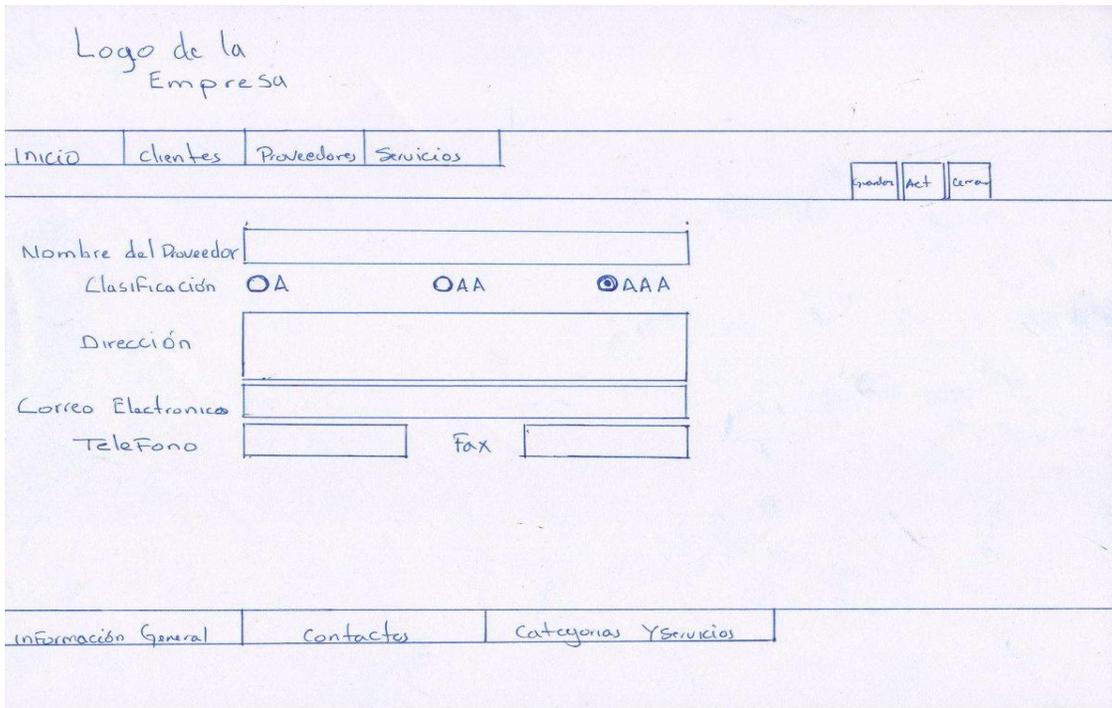


Imagen 8. Bosquejo de la pantalla Crear nuevo Proveedor

Logo de la Empresa

Inicio | Clientes | Proveedores | Servicios

Guardar | Act | Cerrar

Nombre

Cargo

Correo Electronico

Telefono 1  Telefono 2  Fax

Información General | Contactos | Categorías y Servicios

Imagen 9. Bosquejo de la pantalla Crear nuevo Proveedor (Contactos)

Logo de la Empresa

Inicio | Clientes | Proveedores | Servicios

Guardar | Actualiz | Cerrar

Agregar Servicios

Categorías

Servicios

Información General | Contactos | Categorías y Servicios

Imagen 10. Bosquejo de la pantalla Crear nuevo Proveedor (Categorías)

## Capturas de Pantalla

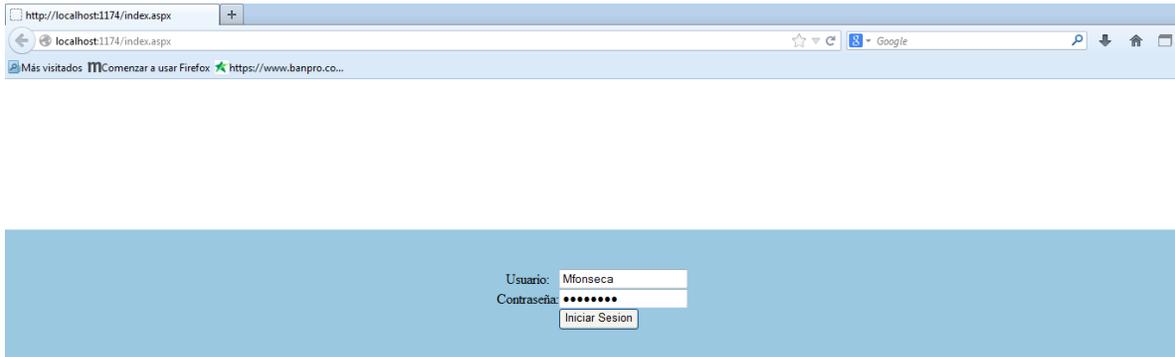


Imagen 11. Pantalla Iniciar sesión



Imagen 12. Pantalla de Página inicial

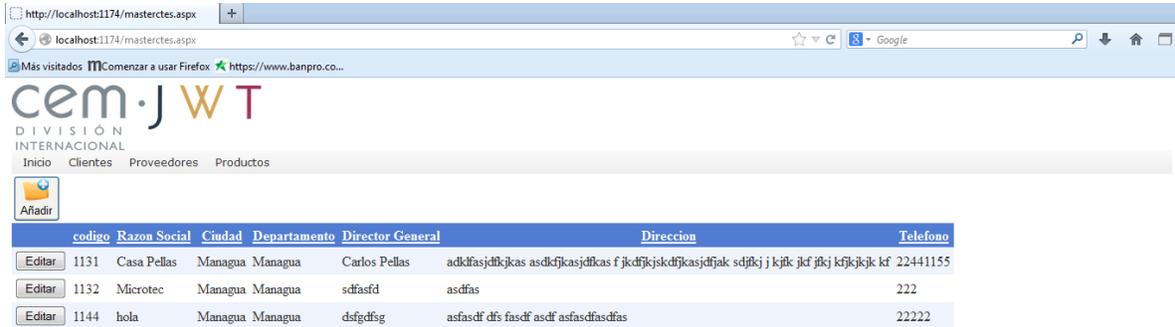


Imagen 13. Pantalla Catálogo de Clientes

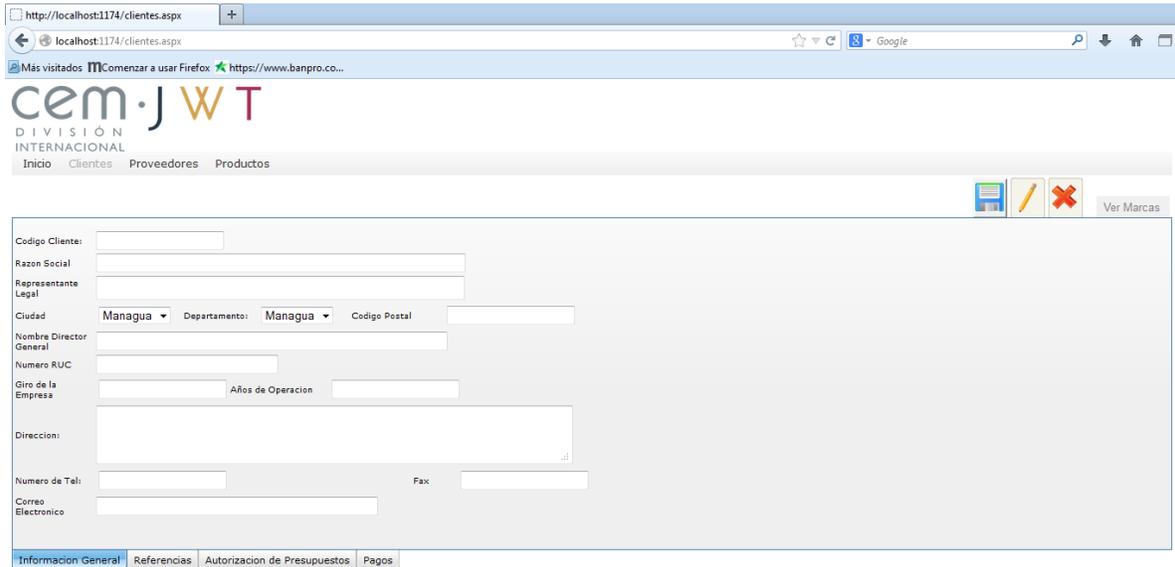


Imagen 14. Pantalla Crear nuevo Cliente (Información General)

The screenshot shows a web browser window with the URL `http://localhost:1174/clientes.aspx`. The page header includes the logo for 'cem·J W T DIVISION INTERNACIONAL' and a navigation menu with 'Inicio', 'Clientes', 'Proveedores', and 'Productos'. The main content area is titled 'REFERENCIAS' and contains two sections: 'REFERENCIAS BANCARIAS' with three input fields, and 'REFERENCIAS COMERCIALES' with two columns of input fields labeled 'COMPAÑIA' and 'TELEFONO'. A breadcrumb trail at the bottom shows 'Información General' > 'Referencias' > 'Autorización de Presupuestos' > 'Pagos'.

Imagen 15. Pantalla Crear nuevo Cliente (Referencias)

The screenshot shows the same web browser window, but the main content area is now titled 'Autorización de presupuestos'. It contains a form with the following fields: 'Nombre', 'cedula', 'Cargo que desempeña', 'Telefono', 'Celular', and 'Correo Electronico'. The breadcrumb trail at the bottom is updated to 'Información General' > 'Referencias' > 'Autorización de Presupuestos' > 'Pagos'. A small window titled 'Documental - Microsoft Word' is visible at the bottom of the browser.

Imagen 16. Pantalla Crear nuevo Cliente (Autorización de presupuestos)

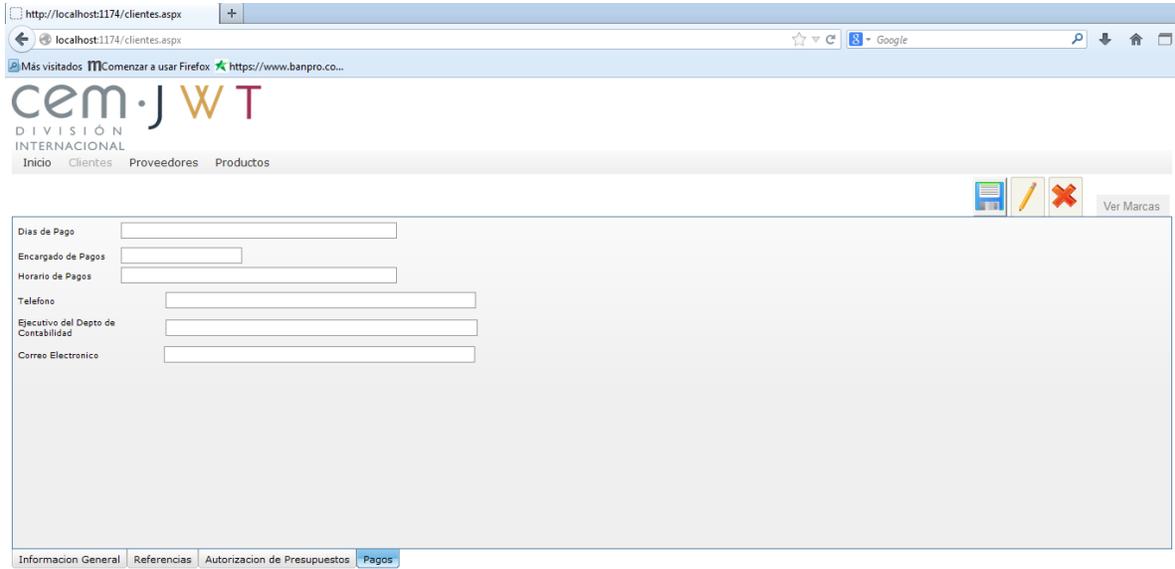


Imagen 17. Pantalla Crear nuevo Cliente (Pagos)



Imagen 18. Pantalla Catálogo de Proveedores

The screenshot shows a web browser window with the URL `http://localhost:1174/crearprov.aspx`. The page header includes the logo for 'cem·J W T DIVISIÓN INTERNACIONAL' and a navigation menu with 'Inicio', 'Clientes', 'Proveedores', and 'Productos y Servicios'. The main content area is a form titled 'Crear nuevo proveedor (Información General)'. The form contains the following fields and options:

- Nombre del Proveedor:
- Clasificación:  A,  AA,  AAA
- Dirección:
- Teléfono:  Fax:
- Correo electrónico:

At the bottom of the form, there are three tabs: 'Información General' (which is selected), 'Contactos', and 'Categorías y Servicios'.

Imagen 19. Pantalla Crear nuevo proveedor (Información General)

The screenshot shows the same web browser window as in Image 19, but the 'Contactos' tab is now selected. The form contains the following fields:

- Nombre:
- Cargo:
- Teléfono1:  Teléfono2:  Fax:
- Correo:

The 'Información General' tab is now unselected, and the 'Contactos' tab is highlighted in blue.

Imagen 20. Pantalla Crear nuevo proveedor (Contacto)

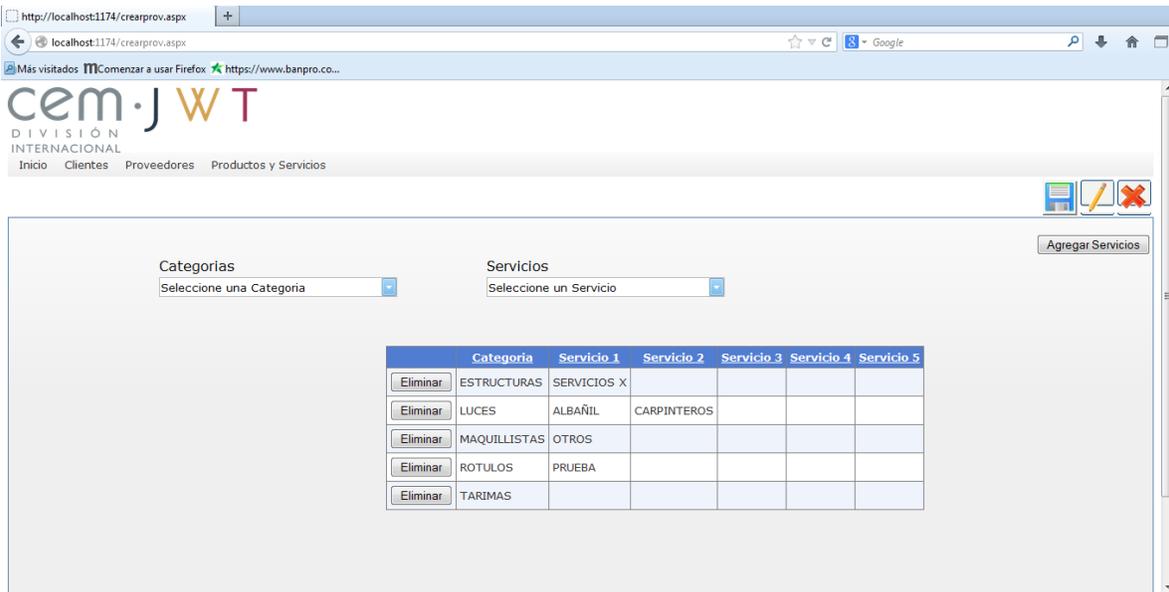


Imagen 21. Pantalla Crear nuevo proveedor (Categorías)

## Resultados.

Como resultado de la entrega de la primera iteración, el cliente quedó satisfecho con la funcionalidad de los módulos desarrollados, sin embargo solicitó cambios visuales en los formularios del Sistema que será una de las prioridades a desarrollar en la siguiente iteración.

## 8.2. Segunda Iteración

Para esta iteración se ha desarrollado el módulo de Proyecciones de Gastos y se han diseñado y codificado las búsquedas y los filtros del sistema, para esto se han utilizado y aplicado las herramientas que destaca la metodología XP.

A continuación en las tablas 11, 12, 13 y en las imágenes 22-28 se muestra un resumen de las herramientas utilizadas en esta iteración y por último en las imágenes 22-38 se presentan las capturas de pantalla de los módulos desarrollados. En la sección de Resultados del documento se detalla cada una de estas herramientas, en el Resultado 1 las Historias de usuarios, en el Resultado 2 las tareas de ingeniería y en el Resultado 3 las pruebas de aceptación.

Tabla 11. Historias de usuario de la segunda iteración

Número	Nombre
5	Creación de la Proyección de Gasto
6	Creación de búsquedas y filtros.
7	Creación de Marcas

Tabla 12. Tareas de ingeniería de la segunda iteración

Numero de Tarea	Número de Historia	Nombre de la Tarea
15	5	Diseño de interfaz de Creación de una nueva Proyección de Gastos
16	5	Adaptación de la Base de Datos para Proyecciones de Gastos
17	5	Codificación del módulo de Proyección de Gastos
18	5	Diseño de la Interfaz de Catálogo de Proyecciones de Gastos
19	6	Diseño de la Interfaz de Búsquedas y filtros.
20	6	Definir las consultas a la Base de datos para realizar las búsquedas y filtrado.
21	7	Diseño de la Interfaz de Creación de Marcas de los clientes.
22	7	Adaptación de la Base de Datos para las Marcas
23	7	Diseño de la Interfaz de Catálogo de Marcas

Tabla 13. Pruebas de aceptación de la segunda iteración

Numero de Prueba	Número de Historia	Nombre de la Prueba
7	5	Proyección de Gastos
8	6	Realizar Búsquedas en los formularios.
9	7	Creación de Marcas
10	7	Edición de Marcas

**Bosquejos**

Imagen 22. Bosquejo de la pantalla Nueva proyección de gastos

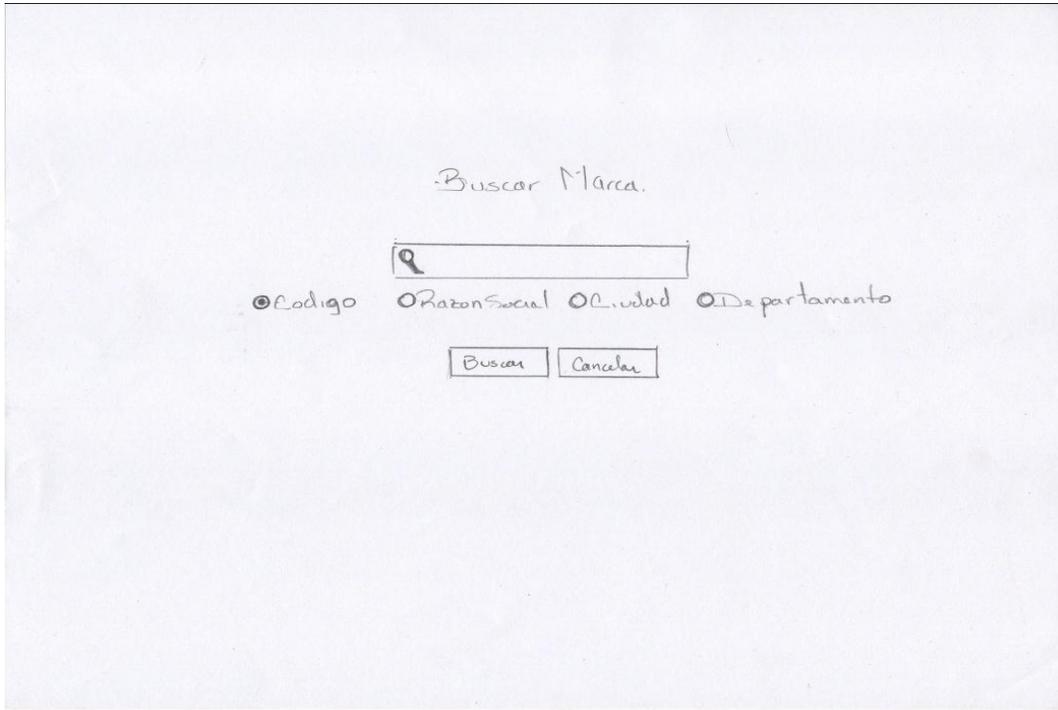


Imagen 23. Bosquejo de la pantalla Búsqueda y filtrado

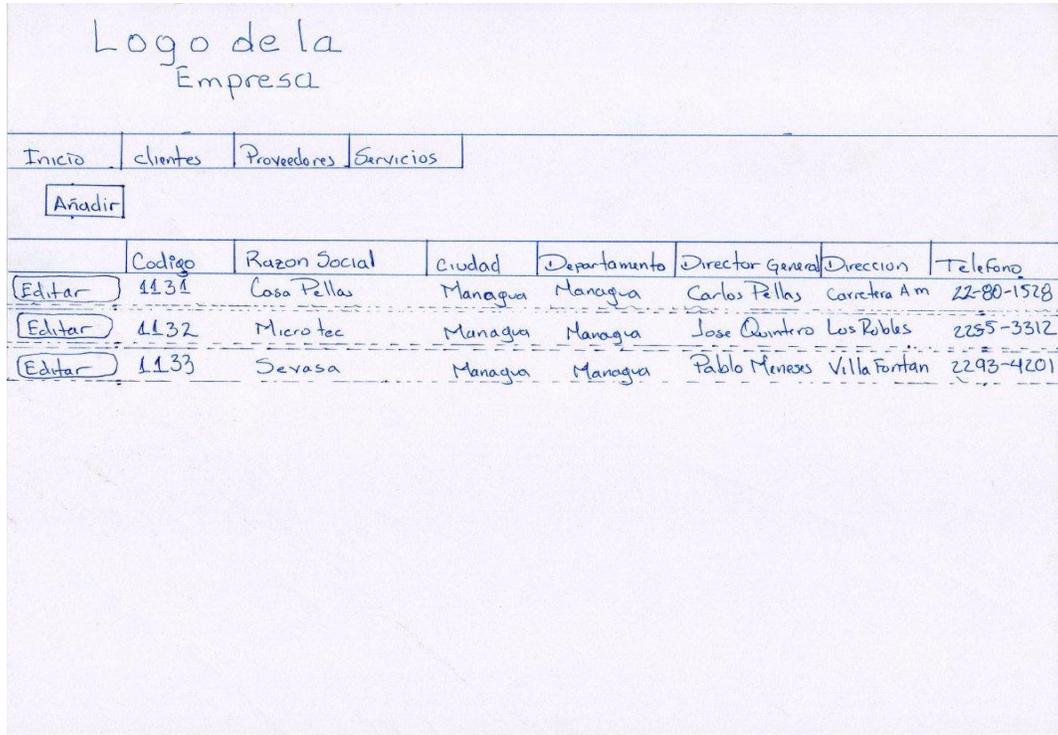


Imagen 24. Bosquejo de la pantalla Catálogo de marcas



Logo de la Empresa

Inicio	clientes	Proveedores	servicios
--------	----------	-------------	-----------

Nombre	<input type="text"/>
Cedula	<input type="text"/>
Cargo que desempeña	<input type="text"/>
Telefono	<input type="text"/>
Celular	<input type="text"/>
Correo Electronico	<input type="text"/>

Informacion General	Referencias	Autorización de Presupuestos	Pagos
---------------------	-------------	------------------------------	-------

Imagen 27. Bosquejo de la pantalla Crear nueva Marca (Autorización de presupuestos)

Logo de la Empresa

Inicio	clientes	Proveedores	servicios
--------	----------	-------------	-----------

Dias de Pago	<input type="text"/>
Encargado de Pagos	<input type="text"/>
Horario de Pagos	<input type="text"/>
Telefono Ejecutivo del depto de Contabilidad	<input type="text"/>
Correo Electronico	<input type="text"/>

Informacion General	Referencias	Autorización de Presupuestos	Pagos
---------------------	-------------	------------------------------	-------

Imagen 28. Bosquejo de la pantalla Crear nueva Marca (Pagos)

Capturas de Pantalla



Imagen 29. Pantalla Catálogo de Clientes (mejorada)

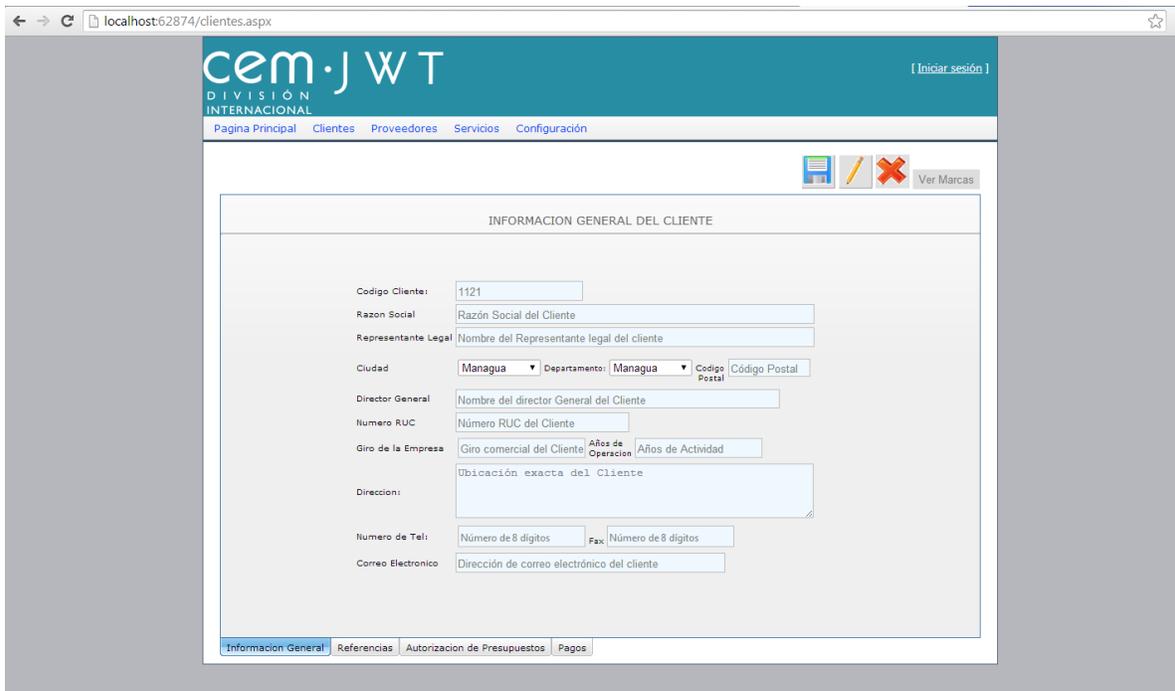


Imagen 30. Pantalla Crear nuevo cliente (mejorada)

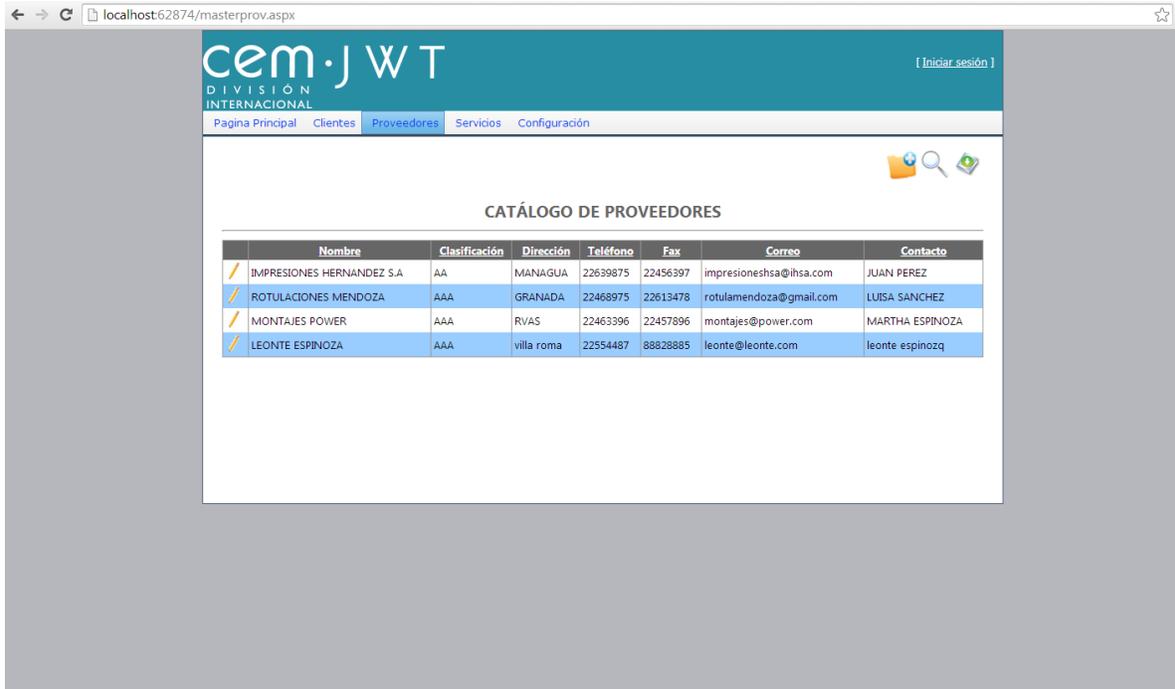


Imagen 31. Pantalla Catálogo de Proveedores (mejorada)

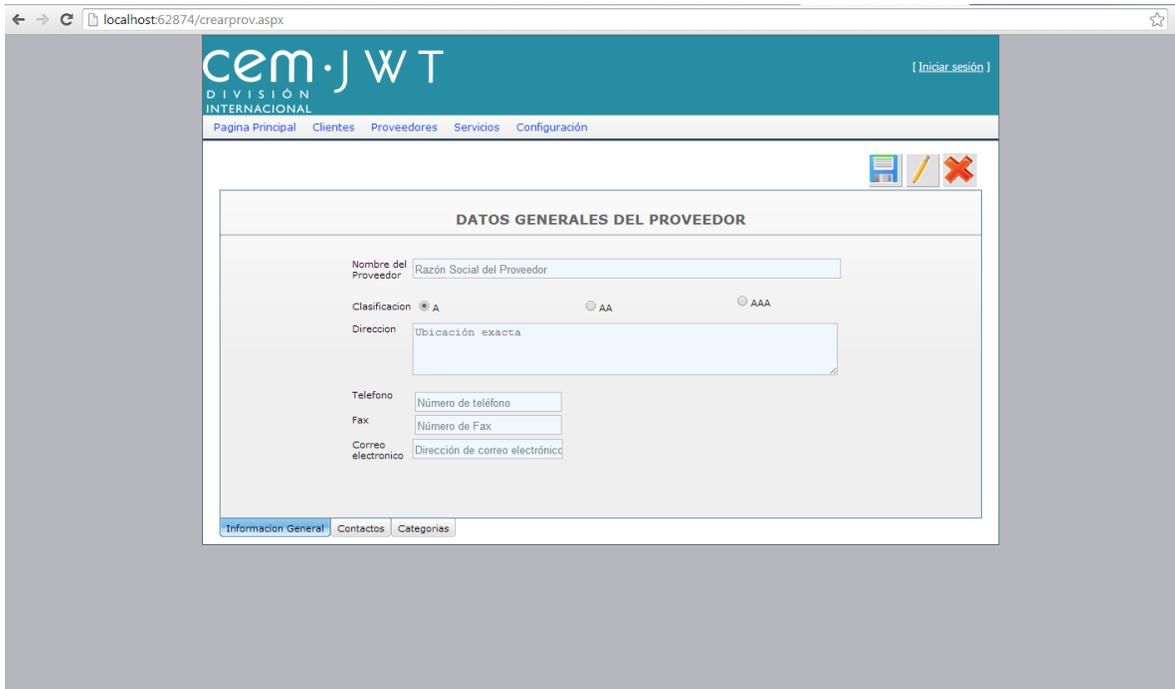


Imagen 32. Pantalla Crear nuevo Proveedor (mejorada)

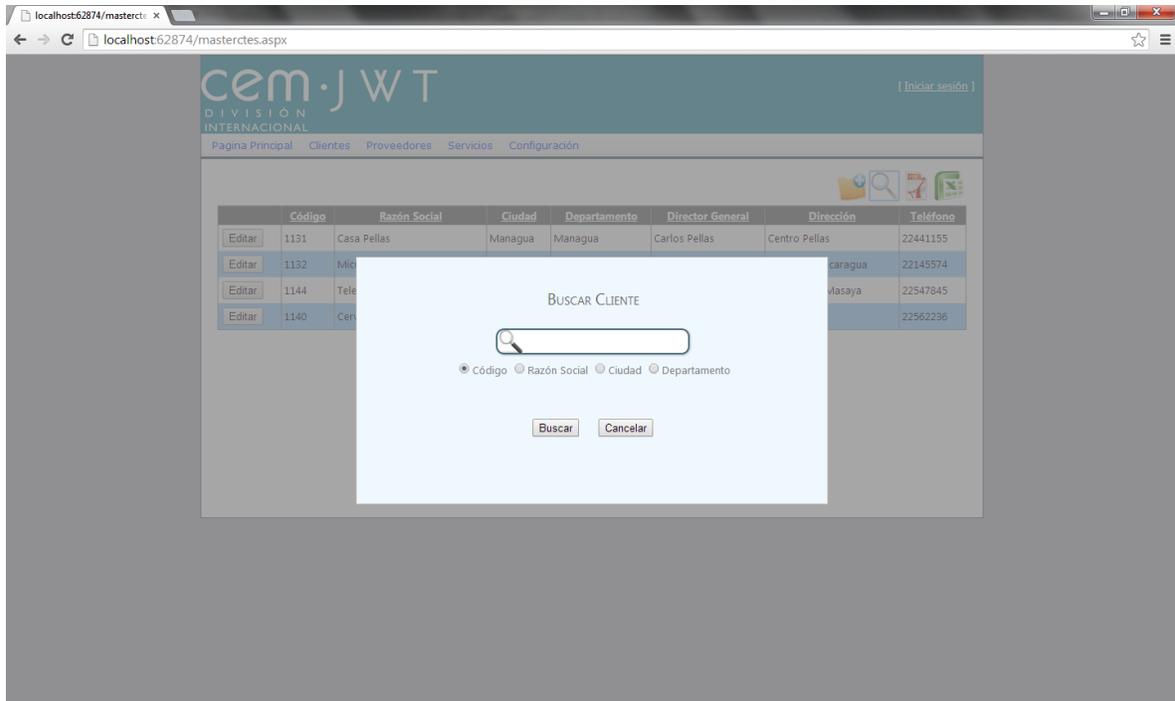


Imagen 33. Pantalla buscar y filtrar Cliente



Imagen 34. Pantalla buscar y filtrar Proveedor

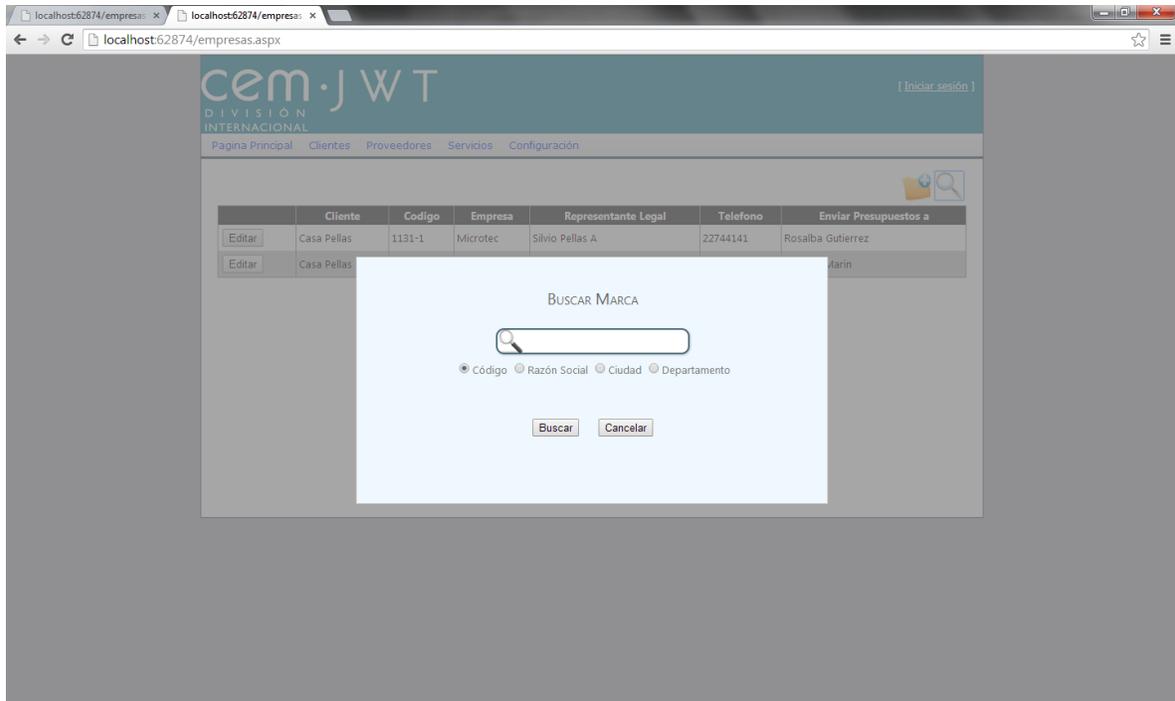


Imagen 35. Pantalla buscar y filtrar Marca

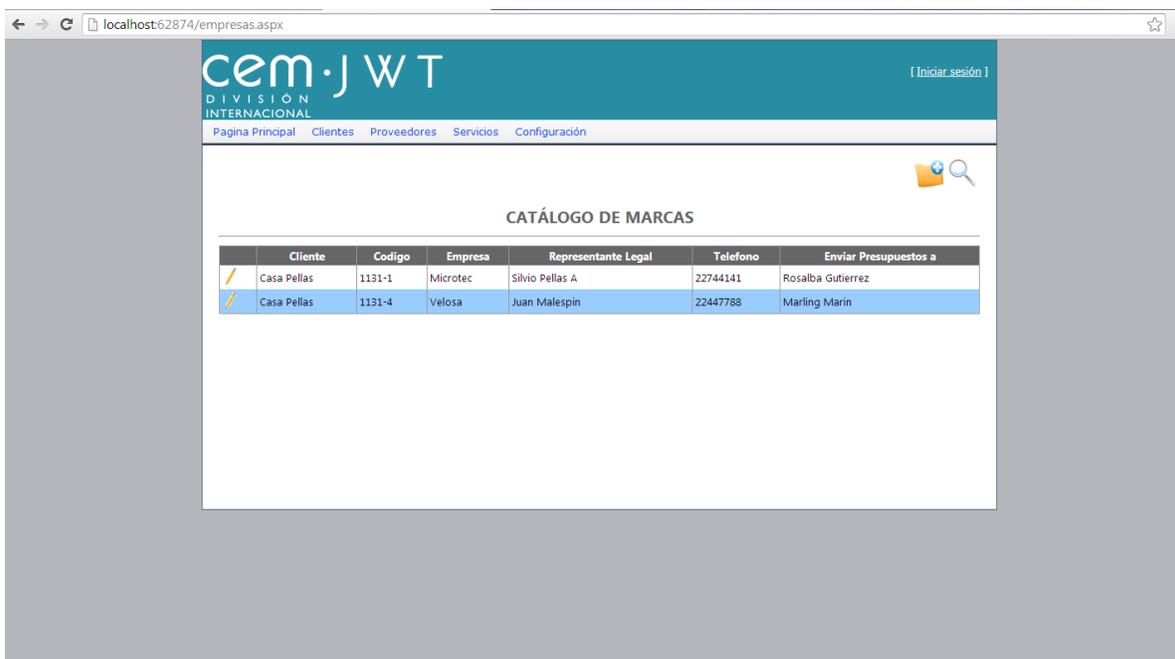


Imagen 36. Pantalla catálogo de Marcas

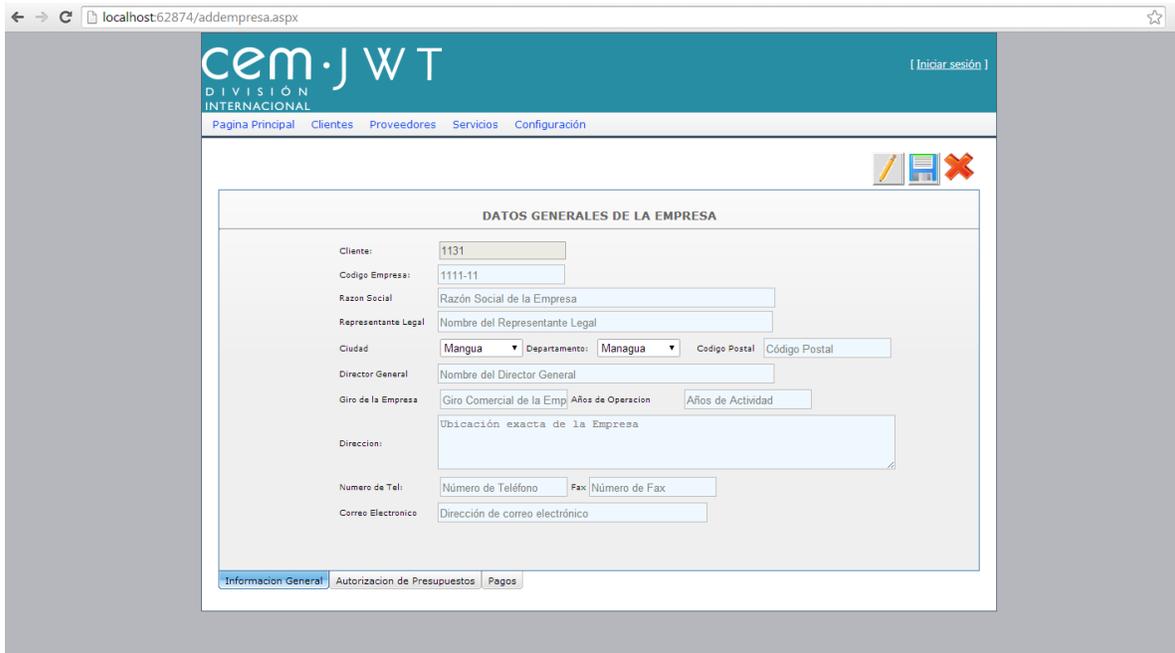


Imagen 37. Pantalla crear nueva Marca

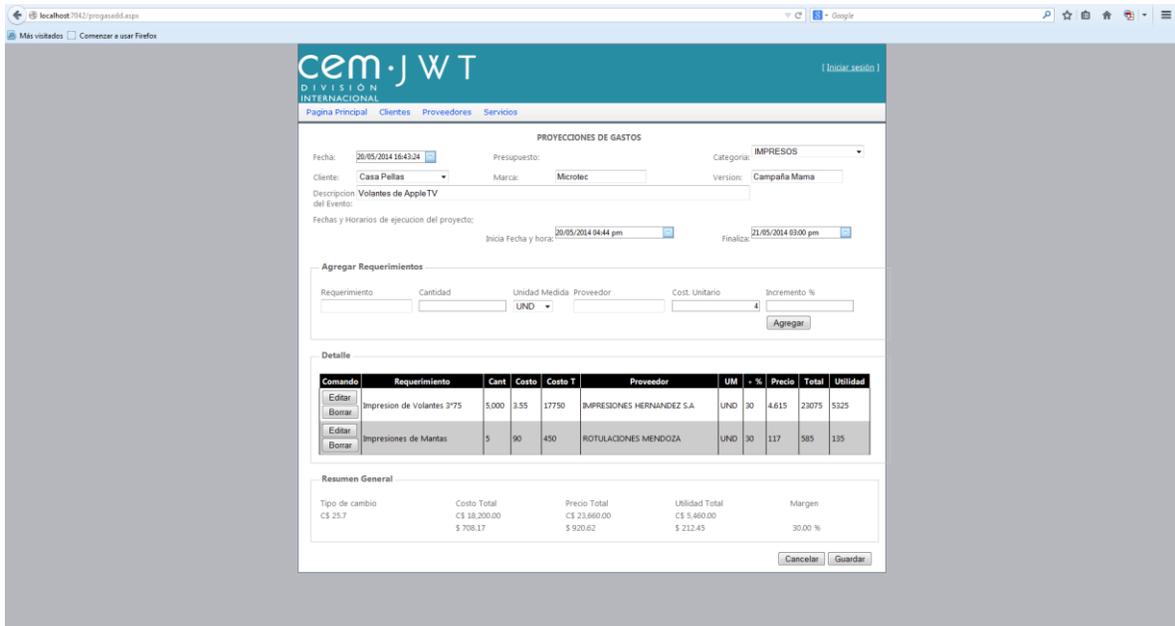


Imagen 38. Pantalla crear nueva Proyección de Gastos

**Resultados.**

Como resultado de la entrega de la segunda iteración, el cliente quedó satisfecho con la funcionalidad de los módulos desarrollados, además se realizaron los cambios visuales necesarios en todos formularios, dichos cambios fueron solicitados por el cliente en la primera iteración.

### 8.3. Tercera Iteración

Para la última iteración se ha desarrollado el módulo de Presupuestos los cuales son generados a partir de una proyección de gastos; también se ha diseñado y codificado la administración de usuarios del sistema, para esto se han utilizado y aplicado las herramientas que destaca la metodología XP.

A continuación en las tablas 14, 15, 16 y en las imágenes 39-41 se muestra un resumen de las herramientas utilizadas en esta iteración y por último en las imágenes 42 y 43 se presentan las capturas de pantalla de los módulos desarrollados. En la sección de Resultados del documento se detalla cada una de estas herramientas, en el Resultado 1 las Historias de usuarios, en el Resultado 2 las tareas de ingeniería y en el Resultado 3 las pruebas de aceptación.

Tabla 14. Historias de usuario de la tercera iteración

Número	Nombre
8	Administración de Usuarios
9	Creación de Presupuestos
10	Aprobación de Presupuestos
11	Generación de reportes Presupuestos.

Tabla 15. Tareas de ingeniería de la tercera iteración

Numero de Tarea	Número de Historia	Nombre de la Tarea
24	8	Diseño de la Interfaz de Administración de Usuarios
25	8	Adaptación de la BD para Administración de Usuarios
26	8	Codificación de Administración de Usuarios
27	9	Diseño de interfaz de la Creación de Presupuestos
28	9	Adaptación de la BD para los Presupuestos
29	9	Codificación del módulo de Creación de Presupuestos
30	9	Diseño de la Interfaz de Catálogo de Presupuestos
31	10	Diseño de la Interfaz de Aprobación de Presupuestos
32	10	Adaptación de la BD para estado de los Presupuestos
33	11	Diseño de la Interfaz de los Reportes de Presupuestos
34	11	Codificación del módulo de Reporte de Presupuestos

Tabla 16. Pruebas de aceptación de la tercera iteración

Numero de Prueba	Número de Historia	Nombre de la Prueba
11	8	Crear usuarios
12	8	Editar usuarios
13	9	Crear Presupuesto
14	10	Aprobar Presupuestos
15	11	Generar reportes de Presupuestos
16	11	Exportar reportes de Presupuestos

Bosquejos

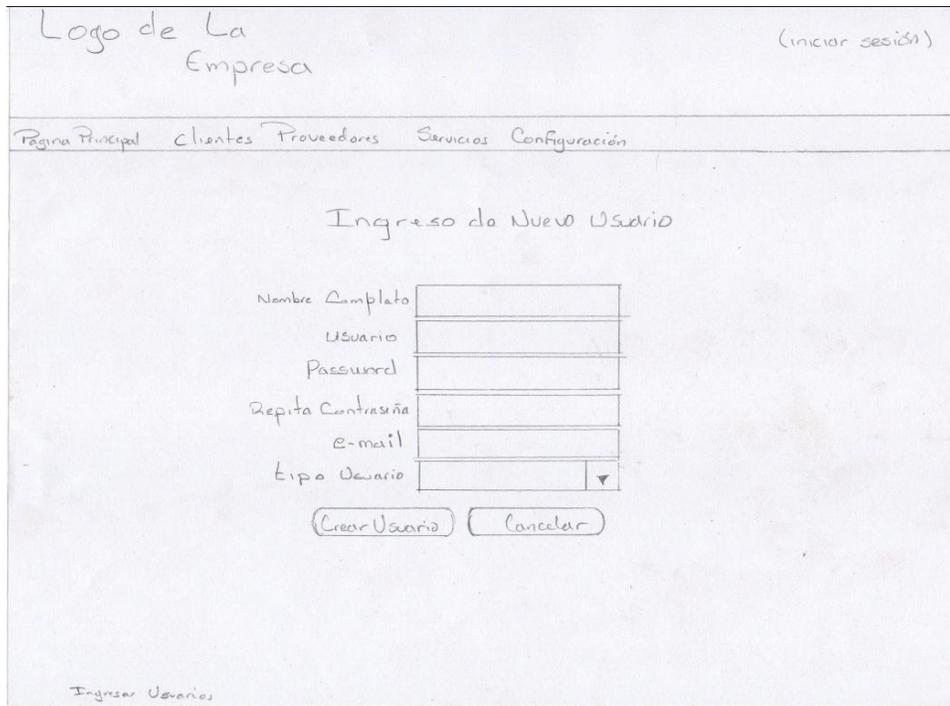


Imagen 39. Bosquejo de la pantalla Administración de usuarios

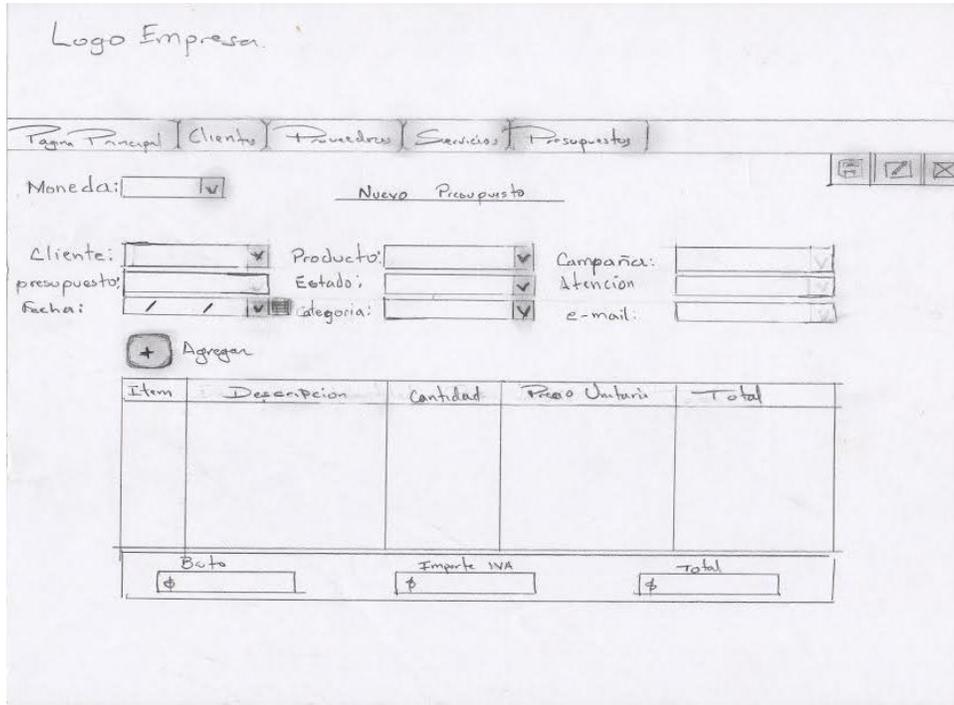


Imagen 40. Bosquejo de la pantalla nuevo Presupuesto

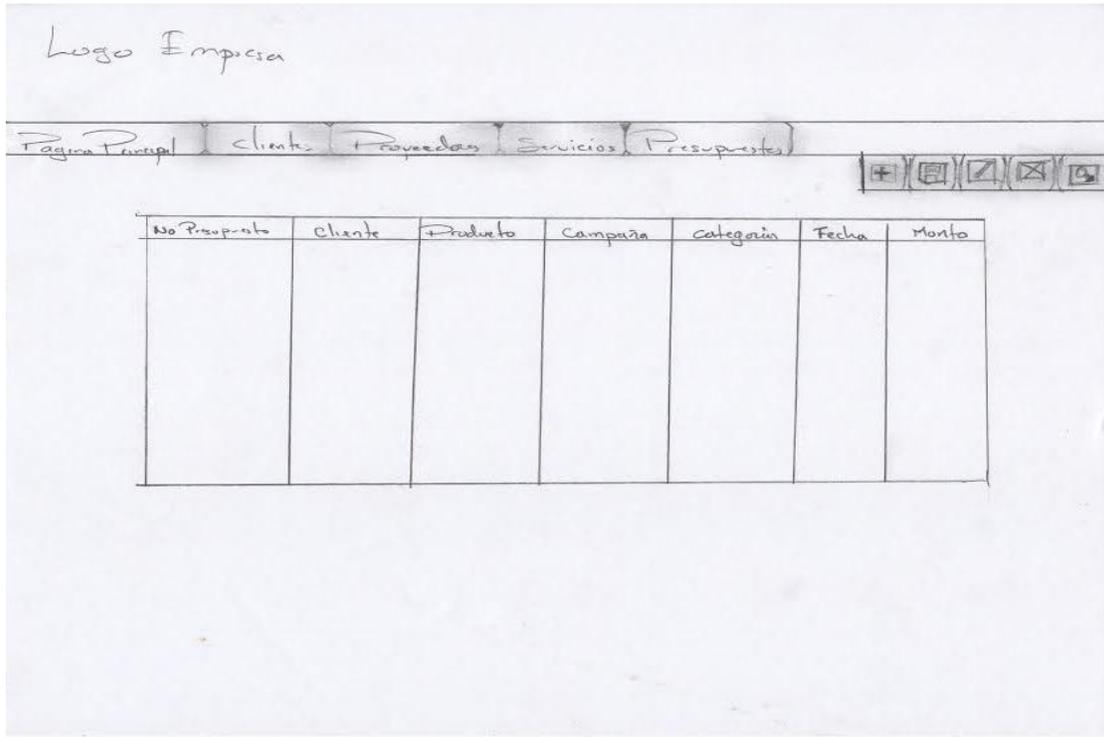


Imagen 41. Bosquejo de la pantalla catálogo de Presupuestos

Capturas de Pantalla

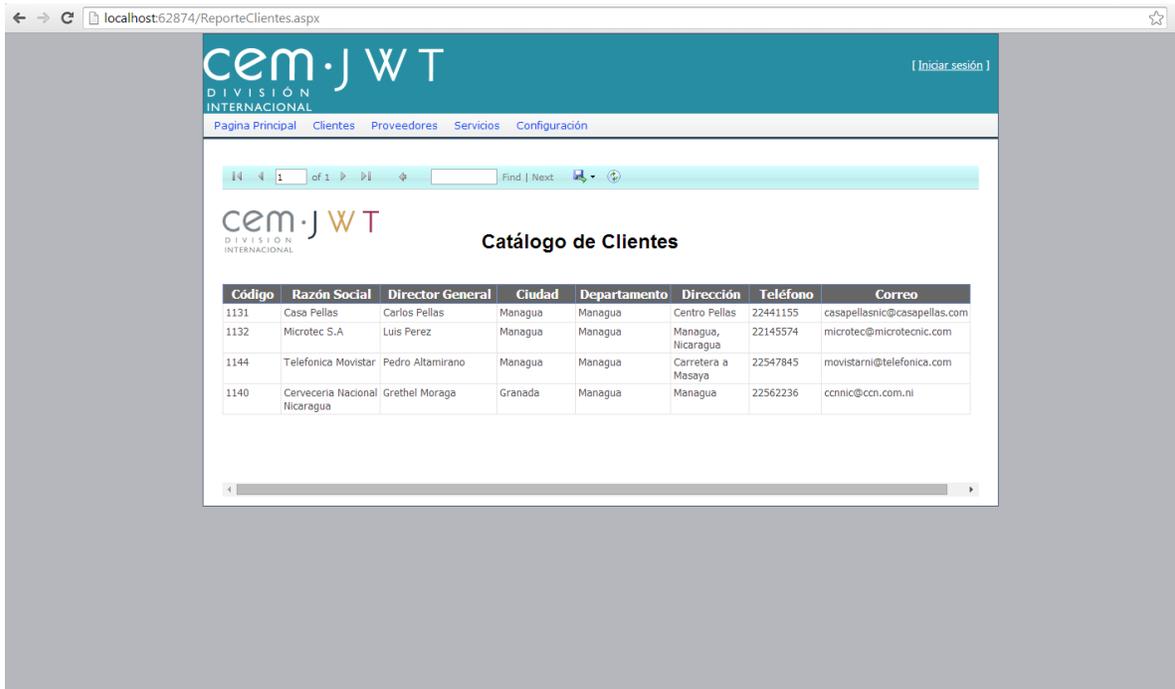


Imagen 42. Pantalla reporte de catálogo de Clientes

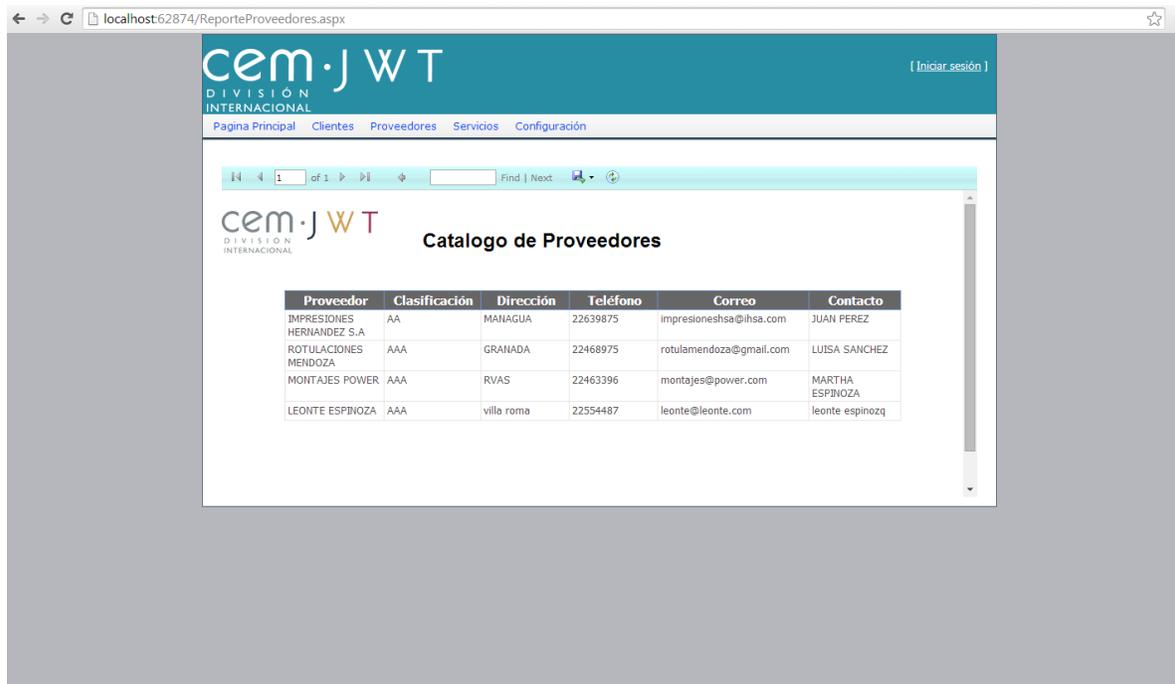


Imagen 43. Pantalla reporte de catálogo de Proveedores

## **Resultados**

Como resultado de la entrega de la última iteración del proyecto, el cliente quedó satisfecho con la funcionalidad e integración de los módulos desarrollados, ya que con esta última entrega se contará con los reportes necesarios para gerencia los cuales vendrán a agilizar el análisis de la información y la toma de decisiones.

## 9. CONCLUSIONES

Podemos asegurar que las técnicas de la metodología de programación extrema aplicadas a la creación del sistema de gestión de presupuestos fueron parte esencial del desarrollo de este, porque nos permitió como equipo plantearnos una forma ordenada de trabajo así como también permitió al cliente satisfacer sus necesidades.

Las ventajas obtenidas de utilizar la metodología XP en el desarrollo de este sistema son muy evidentes ya que en un tiempo considerablemente corto se pudo concluir con la entrega del proyecto, otras de las ventajas de la metodología XP observadas es el alto grado de adaptabilidad y sencillez tanto en su aprendizaje como en su aplicación lo que reduce los costos de implantación en el equipo de desarrollo.

Como desventaja en nuestra experiencia existió dificultad al aplicar la técnica de programación en parejas ya que no se trata solo de sentar a dos programadores frente a una computadora, se necesita definir una estrategia clara y roles específicos que debe desempeñar cada uno, lo que conlleva un proceso de capacitación y entrenamiento, lo cual no se pudo realizar al inicio por el factor tiempo y distancia porque los programadores trabajaban en distintas compañías; en nuestro caso particular pudimos experimentar el trabajo en parejas incorporando como colaborador de la agencia a un miembro del equipo para la tercera iteración del proyecto, como resultado de esto se pudo apreciar que la programación en pareja ayuda en gran medida al desarrollo de código organizado y libre de errores, al mismo tiempo sirve para enriquecer el conocimiento de los programadores a través del intercambios de experiencias e ideas durante este proceso.

Debido a todo lo anterior podemos afirmar que no siempre puede aplicarse esta técnica ya esta puede aplicarse plenamente cuando el grupo de desarrollo pueda estar comunicado (ya sea en la misma instalación, a través de videoconferencias, video-chat o asistencias remota) sin interrupción durante el proceso.

En fin, una vez en uso el sistema de gestión de presupuestos se automatizó los procesos manuales que realizaba el departamento de operaciones de la agencia, proporcionándole al usuario una forma más amigable de trabajar con los presupuestos, lo que dio como resultado la optimización de los tiempos de emisión de los documentos, el control del flujo de información de estos con las diferentes áreas de la agencia, un mejor manejo del historial de la información de los clientes, proveedores, departamentos y servicios. Se mejoró la calidad de los informes entregados a la Gerencia.

Con la implantación del sistema de gestión de presupuestos se logró crear la transparencia del proceso en la emisión de presupuestos que es uno de los puntos que más valora la Gerencia, así como también se logró controlar que solo el departamento de operaciones pueda realizar este tipo de documentos y por último se logró brindar al departamento de operaciones una herramienta que le ayude a darle el debido seguimiento a los clientes y proveedores.

## 10. BIBLIOGRAFIA

Abrahamsson, P., Salo, O., Ronkainen, J., Warsta, J. (2002). Agile software development methods Review and analysis.VTT Publications.

Beck, K.(1999).Extreme Programming Explained. Embrace Change. (1999). Pearson Education, 1999.Traducido al español como: Una explicación de la programación extrema. Aceptar el cambio", Addison Wesley, 2000.

Cockbun, A., Williams, L. (2000). The Costs and Benefits of Pair Programming.Humans and Technology Technical Report.

Cockbun, A. (2001). Agile Software Development.Addison-Wesley.

Highsmith J., Orr K. (2002). Adaptive Software Development: A Collaborative Approach to Managing Complex Systems. Dorset House.

Highsmith, J. (2002). Agile Software Development Ecosystems.Addison-Wesley.

Jeffries, R., Anderson, A., Hendrickson, C. (2001). Extreme Programming Installed". Addison-Wesley.

Newkirk, J., Martin R.C. (2001). Extreme Programming in Practice.Addison-Wesley.

Poppendieck M., Poppendieck T. (2003). Lean Software Development: An Agile Toolkit for Software Development Managers. Addison Wesley.

Schwaber K., Beedle M., Martin R.C. (2001). Agile Software Development with SCRUM.Prentice Hall.

Stapleton J. (1997). DSDM Dynamic Systems Development Method: The Method in Practice. Addison-Wesley.

# 11. RESULTADOS

## 11.1. RESULTADO 1: Historias de usuario

Tabla 17. Historia de usuario Acceso al sistema

<b>Historia de Usuario</b>	
Número: 1	Usuario: Operador
Nombre historia: Acceso al sistema	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Engels Obregón	
Descripción: Los operadores del sistema tendrán un usuario y clave única con la que podrán ingresar al sistema.	
Observaciones:	

Tabla 18. Historia de usuario Creación de Proveedores

<b>Historia de Usuario</b>	
Número: 2	Usuario: administrador
Nombre historia: Creación de proveedores	
Prioridad en negocio: alta	Riesgo en desarrollo: media
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Engels Obregón	
Descripción: solo el administrador del sistema podrá crear los proveedores que tendrá el sistema. De igual forma estos deberán estar guardados en la bitácora.	
Observaciones:	

Tabla 19. Historia de usuario Creación de clientes

Historia de Usuario	
Número: 3	Usuario: administrador
Nombre historia: creación de clientes	
Prioridad en negocio: alta	Riesgo en desarrollo: media
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Marco Fonseca	
Descripción: solo el administrador del sistema podrá crear los clientes que tendrá el sistema. De igual forma estos deberán estar guardados en la bitácora.	
Observaciones:	

Tabla 20. Historia de usuario Creación de productos y servicios

Historia de Usuario	
Número: 4	Usuario: administrador
Nombre historia: Creación de servicios	
Prioridad en negocio: media	Riesgo en desarrollo: alta
Puntos estimados: 6	Iteración asignada: 1
Programador responsable: Marco Fonseca	
Descripción: El administrador será el encargado de realizar y grabar los productos y servicios de los diferentes proveedores. Los datos de los productos y servicios que necesitamos serán suministrados en una lista enviada por correo.	
Observaciones:	

Tabla 21. Historia de usuario Creación de Proyección de Gastos

<b>Historia de Usuario</b>	
Número: 5	Usuario: Operador
Nombre historia: Creación de la Proyección de Gastos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 10	Iteración asignada: 2
Programador responsable: Marco Fonseca	
Descripción: El operador del sistema creará las Proyecciones de gastos para clientes según los registros de los proveedores.	
Observaciones:	

Tabla 22. Historia de usuario Creación de búsquedas y Filtros

<b>Historia de Usuario</b>	
Número: 6	Usuario: Operador
Nombre historia: Creación de Búsquedas y filtros.	
Prioridad en negocio: Media	Riesgo en desarrollo: medio
Puntos estimados: 6	Iteración asignada: 2
Programador responsable: Engels Obregón	
Descripción: El operador será capaz de realizar búsquedas en los diferentes formularios del sistema, el sistema debe contar con filtros para realizar dichas búsquedas.	
Observaciones:	

Tabla 23. Historia de usuario Creación de Marca

<b>Historia de Usuario</b>	
Número: 7	Usuario: administrador
Nombre historia: Creación de Marca	
Prioridad en negocio: Alta	Riesgo en desarrollo: media
Puntos estimados: 2	Iteración asignada: 2
Programador responsable: Marco Fonseca	
Descripción: solo el administrador del sistema podrá crear los clientes que tendrá el sistema. De igual forma estos deberán estar guardados en la bitácora.	
Observaciones:	

Tabla 24. Historia de usuario Administración de usuarios

<b>Historia de Usuario</b>	
Número: 8	Usuario: Administrador
Nombre historia: Administración de Usuarios	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 5	Iteración asignada: 3
Programador responsable: Engels Obregón	
Descripción: El Administrador se encarga de ingresar nuevos usuarios y definir a su vez que tipo de usuario o rol al que pertenecen.	
Observaciones:	

Tabla 25. Historia de usuario Creación de Presupuestos

<b>Historia de Usuario</b>	
Número: 9	Usuario: Operador
Nombre historia: Creación de Presupuestos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 5	Iteración asignada: 3
Programador responsable: Marco Fonseca	
Descripción: El Operador podrá generar presupuestos para el cliente a partir de una proyección de gastos.	
Observaciones: Los Presupuestos deben estar enlazados a una proyección de gastos.	

Tabla 26. Historia de usuario Aprobación de Presupuestos

<b>Historia de Usuario</b>	
Número: 10	Usuario: Administrador
Nombre historia: Aprobación de Presupuestos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 8	Iteración asignada: 3
Programador responsable: Marco Fonseca	
Descripción: El Administrador podrá definir el estado del presupuesto como aprobado una vez que el cliente haya dado el visto bueno.	
Observaciones:	

Tabla 27. Historia de usuario Generación de Reportes de Presupuestos

<b>Historia de Usuario</b>	
Número: 11	Usuario: Operador
Nombre historia: Generación de reportes de Presupuestos.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 10	Iteración asignada: 3
Programador responsable: Engels Obregón	
Descripción: El Operador podrá generar reportes de todos los presupuestos emitidos, el sistema debe contar con filtros de fecha, clientes, estado, etc.	
Observaciones:	

## 11.2. RESULTADO 2: Tareas de ingeniería

Tabla 28. Tarea de ingeniería 1 para Historia de usuario 1

<b>Tarea de ingeniería.</b>			
Número tarea:	1	Número historia:	1
Nombre tarea:	Diseño de interfaz del acceso al sistema		
Tipo de tarea :	Desarrollo	Puntos estimados:	0.5
Fecha inicio:	16/12/2013	Fecha fin:	17/12/2013
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizara el diseño de la interfaz en el cual los usuarios pondrán el nombre de usuario y su contraseña.                 </div>		

Tabla 29. Tarea de ingeniería 2 para Historia de usuario 1

<b>Tarea de ingeniería.</b>			
Número tarea:	2	Número historia:	1
Nombre tarea:	Validación de usuarios en la BD		
Tipo de tarea :	Desarrollo	Puntos estimados:	0.5
Fecha inicio:	18/12/2013	Fecha fin:	19/12/2013
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizara el script de registro de usuario para que se pueda realizar la verificación de que los usuarios que están tratando de acceder al sistema sean los que estén registrados en el mismo. Todas las claves estarán encriptadas a través de md5.                 </div>		

Tabla 30. Tarea de ingeniería 3 para Historia de usuario 1

<b>Tarea de ingeniería.</b>			
Número tarea:	3	Número historia:	1
Nombre tarea:	Adaptación de la Base de datos para los usuarios		
Tipo de tarea :	Desarrollo	Puntos estimados:	1
Fecha inicio:	20/12/2013	Fecha fin:	21/12/2013
Programador responsable:	Engels Obregón		
Descripción:	<p>Se realizara la adaptación de la base de datos del sistema para que pueda contener los registros necesarios para las verificaciones de los usuarios. Tales como nombre, login, password, fecha de creación de usuario.</p>		

Tabla 31. Tarea de ingeniería 1 para Historia de Usuario 2

<b>Tarea de ingeniería.</b>			
Número tarea:	4	Número historia:	2
Nombre tarea:	Diseñar interfaz de usuario para la creación de proveedores		
Tipo de tarea :	Desarrollo	Puntos estimados:	1
Fecha inicio:	23/12/2013	Fecha fin:	24/12/2013
Programador responsable:	Engels Obregón		
Descripción:	<p>Se necesita que el administrador u operador pueda crear los proveedores asociados a los productos que brinda la agencia. La interfaz deberá de contener los campos comunes como nombre, razón social, RUC, categoría, dirección, contacto, teléfono y correo electrónico</p>		

Tabla 32. Tarea de ingeniería 2 para Historia de usuario 2

<b>Tarea de ingeniería.</b>			
Número tarea:	5	Número historia:	2
Nombre tarea:	Diseño de la base de datos de Proveedores		
Tipo de tarea :	Desarrollo	Puntos estimados:	0.5
Fecha inicio:	24/12/2013	Fecha fin:	24/12/2013
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se requiere acondicionar la base de datos para que registre los proveedores y que su código sea su llave primaria la cual deberá de ser única. Deberá de contener los campos requeridos y la asociación de los productos y servicios en la tabla de productos.                 </div>		

Tabla 33. Tarea de ingeniería 3 para Historia de usuario 2

<b>Tarea de ingeniería.</b>			
Número tarea:	6	Número historia:	2
Nombre tarea:	Grabar en BD proveedores		
Tipo de tarea :	Desarrollo	Puntos estimados:	1.5
Fecha inicio:	26/12/2013	Fecha fin:	27/12/2013
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se debe de guardar en la base de datos los datos correspondientes al proveedor en cuestión, datos generales, categoría, contactos.                 </div>		

Tabla 34. Tarea de ingeniería 1 para historia de usuario 3

<b>Tarea de ingeniería.</b>			
Número tarea:	7	Número historia:	3
Nombre tarea:	Creación de Interfaz usuario para la creación de clientes		
Tipo de tarea :	Desarrollo	Puntos estimados:	1
Fecha inicio:	29/12/2013	Fecha fin:	30/12/2013
Programador responsable:	Marco Fonseca		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se necesita crear la interfaz de usuario para poder la grabar a los clientes referentes del sistema.                 </div>		

Tabla 35. Tarea de ingeniería 2 para historia de usuario 3

<b>Tarea de ingeniería.</b>			
Número tarea:	8	Número historia:	3
Nombre tarea:	Diseño de base de datos de clientes		
Tipo de tarea :	Desarrollo	Puntos estimados:	1
Fecha inicio:	30/12/2013	Fecha fin:	31/12/2013
Programador responsable:	Marco Fonseca		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se debe de guardar en la base de datos los datos correspondientes al cliente como datos generales, categoría, contactos.                 </div>		

Tabla 36. Tarea de ingeniería 3 para historia de usuario 3

<b>Tarea de ingeniería.</b>			
Número tarea:	9	Número historia:	3
Nombre tarea:	Grabación de clientes		
Tipo de tarea :	Desarrollo	Puntos estimados:	1
Fecha inicio:	02/01/2014	Fecha fin:	04/01/2014
Programador responsable:	Marco Fonseca		
Descripción:	<p>Se necesita crear un campo clave en clientes que tiene que ser código el cual va determinar el nombre de cliente y este debe estar relacionado a una marca del cliente. De igual forma todos los datos de la marcas del cliente están relacionados a los datos generales del cliente.</p>		

Tabla 37. Tarea de ingeniería 1 para Historia de usuario 4

<b>Tarea de ingeniería.</b>			
Número tarea:	10	Número historia:	4
Nombre tarea:	Registro de productos impresos de la agencia		
Tipo de tarea :	Desarrollo	Puntos estimados:	1
Fecha inicio:	04/01/2014	Fecha fin:	06/01/2014
Programador responsable:	Marco Fonseca		
Descripción:	<p>Se registra todos los datos de los productos como el nombre, categoría, precio, material, proveedor que lo suministra.</p>		

Tabla 38. Tarea de ingeniería 2 para Historia de usuario 4

<b>Tarea de ingeniería.</b>			
Número tarea:	11	Número historia:	4
Nombre tarea:	Escoger automáticamente el proveedor del producto		
Tipo de tarea :	Desarrollo	Puntos estimados:	0.5
Fecha inicio:	06/01/2014	Fecha fin:	07/01/2014
Programador responsable:	Marco Fonseca		
Descripción:	<p>Cada vez que se ingrese cualquier producto automáticamente el sistema deberá de suministrar la lista de los proveedores disponibles para escoger que proveedor que ofrece ese producto.</p>		

Tabla 39. Tarea de ingeniería 3 para Historia de usuario 4

<b>Tarea de ingeniería.</b>			
Número tarea:	12	Número historia:	4
Nombre tarea:	Actualizar la comisión de la agencia		
Tipo de tarea :	Desarrollo	Puntos estimados:	0.5
Fecha inicio:	07/01/2014	Fecha fin:	08/01/2014
Programador responsable:	Marco Fonseca		
Descripción:	<p>Se debe de generar al precio del proveedor automáticamente la comisión de la agencia que será del 23 % sobre el valor del precio sugerido por el proveedor.</p>		

Tabla 40. Tarea de ingeniería 4 para Historia de usuario 4

<b>Tarea de ingeniería.</b>			
Número tarea:	13	Número historia:	4
Nombre tarea:	Adaptación de la Base de Datos para la grabación de productos		
Tipo de tarea :	Desarrollo	Puntos estimados:	1
Fecha inicio:	08/01/2014	Fecha fin:	09/01/2014
Programador responsable:	Marco Fonseca		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se debe de adaptar la base de datos para realizar la inserción de los productos que serán la lista de servicios que conformaran los presupuestos.                 </div>		

Tabla 41. Tarea de ingeniería 5 para Historia de usuario 4

<b>Tarea de ingeniería.</b>			
Número tarea:	14	Número historia:	4
Nombre tarea:	Guardar en la Base de Datos productos		
Tipo de tarea :	Desarrollo	Puntos estimados:	1
Fecha inicio:	09/01/2014	Fecha fin:	10/01/2014
Programador responsable:	Marco Fonseca		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Una vez ingresados los datos de los productos se realizara el guardado en la base de datos del sistema.                 </div>		

Tabla 42. Tarea de ingeniería 1 para Historia de usuario 5

<b>Tarea de ingeniería.</b>			
Número tarea:	15	Número historia:	5
Nombre tarea:	Diseño de interfaz de Creación de una nueva Proyección de Gastos		
Tipo de tarea :	Desarrollo	Puntos estimados:	0.5
Fecha inicio:	10/02/2014	Fecha fin:	10/02/2014
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizara el diseño de la interfaz del ingreso de nuevas proyecciones de gastos en el cual los usuarios ingresarán los datos correspondientes a cada Proyección.                 </div>		

Tabla 43. Tarea de ingeniería 2 para Historia de usuario 5

<b>Tarea de ingeniería.</b>			
Número tarea:	16	Número historia:	5
Nombre tarea:	Adaptación de la BD para Proyecciones de Gastos		
Tipo de tarea :	Desarrollo	Puntos estimados:	1
Fecha inicio:	11/02/2014	Fecha fin:	11/02/2014
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se requiere acondicionar la base de datos para que registre las proyecciones de gastos y que su código sea su llave primaria la cual deberá de ser única. Deberá de contener los campos requeridos para que toda la información sea almacenada.                 </div>		

Tabla 44. Tarea de ingeniería 3 para Historia de usuario 5

<b>Tarea de ingeniería.</b>			
Número tarea:	17	Número historia:	5
Nombre tarea:	Codificación del módulo de Proyección de Gastos		
Tipo de tarea :	Desarrollo	Puntos estimados:	15
Fecha inicio:	17/02/2014	Fecha fin:	
Programador responsable:	Marco Fonseca		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizará todos los métodos y funciones necesarias para garantizar el buen funcionamiento de todo el módulo de Proyección de Gastos.                 </div>		

Tabla 45. Tarea de ingeniería 4 para Historia de usuario 5

<b>Tarea de ingeniería.</b>			
Número tarea:	18	Número historia:	5
Nombre tarea:	Diseño de la Interfaz de Catálogo de Proyecciones de Gastos		
Tipo de tarea :	Desarrollo	Puntos estimados:	1
Fecha inicio:	20/02/2014	Fecha fin:	
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizará el diseño de la Interfaz del Catálogo de Proyecciones de Gastos, acá deberá de existir un control que permita editar o eliminar los presupuestos que hayan sido registrados.                 </div>		

Tabla 46. Tarea de ingeniería 1 para Historia de usuario 6

<b>Tarea de ingeniería.</b>			
Número tarea:	19	Número historia:	6
Nombre tarea:	Diseño de la Interfaz de Búsquedas.		
Tipo de tarea :	Desarrollo	Puntos estimados:	5
Fecha inicio:	25/02/2014	Fecha fin: 27/02/2014	
Programador responsable:	Engels Obregón		
Descripción:	Se realizará el diseño de la Interfaz de las Búsquedas y filtros en cada formulario.		

Tabla 47. Tarea de ingeniería 2 para Historia de usuario 6

<b>Tarea de ingeniería.</b>			
Número tarea:	20	Número historia:	6
Nombre tarea:	Definir las consultas a la Base de datos para realizar las búsquedas.		
Tipo de tarea :	Desarrollo	Puntos estimados:	5
Fecha inicio:	28/02/2014	Fecha fin: 03/03/2014	
Programador responsable:	Engels Obregón		
Descripción:	Se estructuraran las consultas en la base de datos para poder mostrar la información buscada en el sistema		

Tabla 48. Tarea de ingeniería 1 para Historia de usuario 7

<b>Tarea de ingeniería.</b>			
Número tarea:	21	Número historia:	7
Nombre tarea:	Diseño de la Interfaz de Creación de Marcas de los clientes.		
Tipo de tarea :	Desarrollo	Puntos estimados:	5
Fecha inicio:	05/03/2014	Fecha fin: 06/03/2014	
Programador responsable:	Marco Fonseca		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizará el diseño de la Interfaz del ingreso de Marcas de los clientes.                 </div>		

Tabla 49. Tarea de ingeniería 2 para Historia de usuario 7

<b>Tarea de ingeniería.</b>			
Número tarea:	22	Número historia:	7
Nombre tarea:	Adaptación de la Base de Datos para el ingreso de las marcas		
Tipo de tarea :	Desarrollo	Puntos estimados:	5
Fecha inicio:	07/03/2014	Fecha fin: 08/03/2014	
Programador responsable:	Marco Fonseca		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se requiere acondicionar la base de datos para que registre las marcas que tienen los clientes. Deberá de contener los campos requeridos para que toda la información sea almacenada.                 </div>		

Tabla 50. Tarea de ingeniería 3 para Historia de usuario 7

<b>Tarea de ingeniería.</b>			
Número tarea:	23	Número historia:	7
Nombre tarea:	Diseño de la Interfaz de Catálogo de Marcas		
Tipo de tarea :	Desarrollo	Puntos estimados:	1
Fecha inicio:	10/03/2014	Fecha fin:	15/03/2014
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizará el diseño de la Interfaz del Catálogo de Proyecciones de Gastos, acá deberá de existir un control que permita editar o eliminar los presupuestos que hayan sido registrados.                 </div>		

Tabla 51. Tarea de ingeniería 1 para Historia de usuario 8

<b>Tarea de ingeniería.</b>			
Número tarea:	24	Número historia:	8
Nombre tarea:	Diseño de la Interfaz de Administración de Usuarios		
Tipo de tarea :	Desarrollo	Puntos estimados:	5
Fecha inicio:	16/03/2014	Fecha fin:	12/03/2014
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizará el diseño de la Interfaz de Administración de Usuarios, acá deberán llenarse los campos correspondientes con la información personal de cada usuario a dar de alta.                 </div>		

Tabla 52. Tarea de ingeniería 2 para Historia de usuario 8

<b>Tarea de ingeniería.</b>			
Número tarea:	25	Número historia:	8
Nombre tarea:	Adaptación de la BD para Administración de Usuarios		
Tipo de tarea :	Desarrollo	Puntos estimados:	5
Fecha inicio:	21/03/2014	Fecha fin:	30/03/2014
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;"> <p>Se requiere acondicionar la base de datos para que registre el ingreso de nuevos usuarios y que su código sea su llave primaria la cual deberá de ser única. Deberá de contener los campos requeridos para que toda la información sea almacenada.</p> </div>		

Tabla 53. Tarea de ingeniería 3 para Historia de usuario 8

<b>Tarea de ingeniería.</b>			
Número tarea:	26	Número historia:	8
Nombre tarea:	Codificación de Administración de Usuarios		
Tipo de tarea :	Desarrollo	Puntos estimados:	15
Fecha inicio:	31/03/2014	Fecha fin:	05/04/2014
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;"> <p>Se realizará todos los métodos y funciones necesarias para garantizar el buen funcionamiento de Administración de usuarios.</p> </div>		

Tabla 54. Tarea de ingeniería 1 para Historia de usuario 9

<b>Tarea de ingeniería.</b>			
Número tarea:	27	Número historia:	9
Nombre tarea:	Diseño de interfaz de la Creación de Presupuestos		
Tipo de tarea :	Desarrollo	Puntos estimados:	8
Fecha inicio:		Fecha fin:06/04/2014	10/04/2014
Programador responsable:	Marco Fonseca		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizara el diseño de la interfaz del ingreso de Presupuestos que dependerán de una proyección de gastos ya creada.                 </div>		

Tabla 55. Tarea de ingeniería 2 para Historia de usuario 9

<b>Tarea de ingeniería.</b>			
Número tarea:	28	Número historia:	9
Nombre tarea:	Adaptación de la BD para los Presupuestos		
Tipo de tarea :	Desarrollo	Puntos estimados:	5
Fecha inicio:	11/04/2014	Fecha fin:	13/04/2014
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se requiere acondicionar la base de datos para que registre los Presupuestos creados y que su código sea su llave primaria la cual deberá de ser única. Deberá de contener los campos requeridos para que toda la información sea almacenada.                 </div>		

Tabla 56. Tarea de ingeniería 3 para Historia de usuario 9

<b>Tarea de ingeniería.</b>			
Número tarea:	29	Número historia:	9
Nombre tarea:	Codificación del módulo de Creación de Presupuestos		
Tipo de tarea :	Desarrollo	Puntos estimados:	10
Fecha inicio:	14/04/2014	Fecha fin:	25/04/2014
Programador responsable:	Marco Fonseca		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizará todos los métodos y funciones necesarias para garantizar el buen funcionamiento de todo el módulo de Creación de Presupuestos.                 </div>		

Tabla 57. Tarea de ingeniería 4 para Historia de usuario 9

<b>Tarea de ingeniería.</b>			
Número tarea:	30	Número historia:	9
Nombre tarea:	Diseño de la Interfaz de Catálogo de Presupuestos.		
Tipo de tarea :	Desarrollo	Puntos estimados:	9
Fecha inicio:	26/04/2014	Fecha fin:	28/04/2014
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizará el diseño de la Interfaz del Catálogo de Presupuestos acá deberá de existir un control que permita editar o eliminar los presupuestos que hayan sido registrados.                 </div>		

Tabla 58. Tarea de ingeniería 1 para Historia de usuario 10

<b>Tarea de ingeniería.</b>			
Número tarea:	31	Número historia:	10
Nombre tarea:	Diseño de la Interfaz de Aprobación de Presupuestos		
Tipo de tarea :	Desarrollo	Puntos estimados:	9
Fecha inicio:	29/04/2014	Fecha fin:	30/04/2014
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizará el diseño de la Interfaz de Aprobación de Presupuestos acá deberá de existir un control que permita editar estado de los presupuestos que hayan sido registrados.                 </div>		

Tabla 59. Tarea de ingeniería 2 para Historia de usuario 10

<b>Tarea de ingeniería.</b>			
Número tarea:	32	Número historia:	10
Nombre tarea:	Adaptación de la BD para estado de los Presupuestos		
Tipo de tarea :	Desarrollo	Puntos estimados:	5
Fecha inicio:	02/05/2014	Fecha fin:	03/05/2014
Programador responsable:	Marco Fonseca		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se requiere acondicionar la tabla de presupuestos en la base de datos para que registre el estado de los Presupuestos creados.                 </div>		

Tabla 60. Tarea de ingeniería 1 para Historia de usuario 11

<b>Tarea de ingeniería.</b>			
Número tarea:	33	Número historia:	11
Nombre tarea:	Diseño de la Interfaz de Generación de Reportes de Presupuestos.		
Tipo de tarea :	Desarrollo	Puntos estimados:	5
Fecha inicio:	04/05/2014	Fecha fin:	10/05/2014
Programador responsable:	Engels Obregón		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se realizará el diseño de la Interfaz gráfica del reporte de Presupuestos acá deberá de existir un control que permita exportar el reporte a archivos PDF y Excel.                 </div>		

Tabla 61. Tarea de ingeniería 2 para Historia de usuario 11

<b>Tarea de ingeniería.</b>			
Número tarea:	34	Número historia:	11
Nombre tarea:	Codificación del módulo de Reporte de Presupuestos		
Tipo de tarea :	Desarrollo	Puntos estimados:	5
Fecha inicio:	11/05/2014	Fecha fin:	20/05/2014
Programador responsable:	Marco Fonseca		
Descripción:	<div style="border: 1px solid black; padding: 5px;">                     Se deberá realizar los métodos y consultas necesarias para que los reportes puedan ser generados en pantalla y ser exportados o impresos.                 </div>		

### 11.3. RESULTADO 3: Pruebas de Aceptación.

Tabla 62. Caso de prueba acceso al sistema

Caso de Prueba	
Numero Caso de Prueba: 1	Numero Historia de Usuario: 1
Nombre Caso de Prueba: Acceso al Sistema	
Descripción: 1- Llenar cada uno de los campos respectivos del usuario. 2- Presionamos el botón Iniciar Sesión.	
Condiciones de Ejecución: Cada usuario debe de contar con un usuario y una contraseña previamente configurados para poder tener acceso al sistema.	
Entradas: 1- Usuario. 2- Contraseña	
Resultado Esperado: El acceso al sistema	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 63. Caso de prueba creación de Proveedor

Caso de Prueba	
Numero Caso de Prueba: 2	Numero Historia de Usuario: 2
Nombre Caso de Prueba: Creación de Proveedor	
Descripción: <ol style="list-style-type: none"> <li>1. Hacer clic en el botón añadir.</li> <li>2. Llenar todos los campos correspondientes a la información general del Proveedor.</li> <li>3. Hacer clic en la pestaña Contacto.</li> <li>4. Llenar cada uno de los campos correspondientes a la información del contacto.</li> <li>5. Hacer clic en la pestaña categorías.</li> <li>6. Seleccionar las categorías del Proveedor.</li> </ol>	
Condiciones de Ejecución: El operador deberá llenar los campos solicitados en cada pestaña antes de pasar al siguiente formulario.	
Entradas: 1- Información general y clasificación del Proveedor. 2- Información general del contacto con el proveedor. 3- Categorías.	
Resultado Esperado: Ingreso del Proveedor, Ingreso del Contacto con el Proveedor.	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 64. Caso de prueba edición de Proveedor

Caso de Prueba	
Numero Caso de Prueba: 3	Numero Historia de Usuario: 2
Nombre Caso de Prueba: Edición de proveedor	
Descripción: <ol style="list-style-type: none"> <li>1- Hacer clic en el botón Editar del registro que se desea editar.</li> <li>2- Modificar los campos correspondientes a la información general del Proveedor que se desean editar.</li> <li>3- Hacer clic en la pestaña Contacto.</li> <li>4- Modificar los campos que se deseen editar del contacto.</li> <li>5- Hacer clic en la pestaña Categorías.</li> <li>6- Modificar o agregar una categoría al proveedor.</li> </ol>	
Condiciones de Ejecución: Una vez presionado el botón editar del registro seleccionado deberá aparecer el formulario con la información del registro seleccionado.	
Entradas:	
Resultado Esperado: Edición del Proveedor, Edición del Contacto con el Proveedor.	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 65. Caso de prueba creación de Clientes

Caso de Prueba	
Numero Caso de Prueba: 4	Numero Historia de Usuario: 3
Nombre Caso de Prueba: Creación de Clientes.	
Descripción: <ol style="list-style-type: none"> <li>1- Hacer clic en el botón añadir.</li> <li>2- Llenar todos los campos correspondientes a la información general del Cliente.</li> <li>3- Hacer clic en la pestaña Referencia.</li> <li>4- Llenar cada uno de los campos correspondientes a la información de las referencias.</li> <li>5- Hacer clic en la pestaña Autorización de Presupuesto.</li> <li>6- Llenar los campos correspondientes a la persona que autoriza el presupuesto.</li> <li>7- Hacer clic en la pestaña pagos.</li> <li>8- Llenar los campos correspondientes a la persona que autoriza y realiza el pago.</li> </ol>	
Condiciones de Ejecución: El operador deberá llenar los campos solicitados en cada formulario antes de pasar a la siguiente pestaña.	
Entradas: <ol style="list-style-type: none"> <li>1- Información general del cliente.</li> <li>2- Información de quien autoriza el presupuesto.</li> <li>3- Referencias del cliente.</li> <li>4- Información de quien realiza los pagos.</li> </ol>	
Resultado Esperado: Ingreso de cliente y sus referencias así como el ingreso de los datos de quien autoriza y realiza los pagos.	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 66. Caso de prueba edición de Clientes

Caso de Prueba	
Numero Caso de Prueba: 5	Numero Historia de Usuario: 3
Nombre Caso de Prueba: Edición de Clientes.	
Descripción: 1- Hacer clic en el botón Editar a la par del registro del Cliente. 2- Editar los campos que se deseen correspondientes a la información general del Cliente. 3- Hacer clic en la pestaña Referencia. 4- Editar los campos que se deseen correspondientes a la información de las referencias. 5- Hacer clic en la pestaña Autorización de Presupuesto. 6- Editar los campos que se deseen correspondientes a la persona que autoriza el presupuesto. 7- Hacer clic en la pestaña pagos. 8- Editar los campos que se deseen correspondientes a la persona que autoriza y realiza el pago.	
Condiciones de Ejecución: El operador deberá llenar los campos solicitados en cada formulario antes de pasar a la siguiente pestaña.	
Entradas:	
Resultado Esperado: Nueva información de cliente y sus referencias así como la modificación de los datos de quien autoriza y realiza los pagos.	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 67. Caso de prueba creación de Servicios

Caso de Prueba	
Numero Caso de Prueba: 6	Numero Historia de Usuario: 4
Nombre Caso de Prueba: Creación de Servicios	
Descripción: 1- Llenar cada uno de los campos respectivos al usuario. 2- presionamos el botón agregar.	
Condiciones de Ejecución: Deben haber categorías creadas para poder crear o agregar servicios a estas.	
Entradas: 1- servicio	
Resultado Esperado: Servicio agregado	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 68. Caso de prueba creación de nueva Proyección de gastos

<b>Caso de Prueba</b>	
Numero Caso de Prueba: 7	Numero Historia de Usuario: 5
Nombre Caso de Prueba: Proyección de Gastos	
Descripción: 1- Llenar cada uno de los campos requeridos 2- Hacer clic en el botón Agregar 3- Verificar lista de detalle	
Condiciones de Ejecución: Deben existir previamente creados los proveedores, Clientes y operadores de esta sección del sistema.	
Entradas: 1- Datos generales de la proyección. 2- Requerimientos.	
Resultado Esperado: Creación de Proyección de gastos	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 69. Caso de prueba búsquedas en los formularios

<b>Caso de Prueba</b>	
Numero Caso de Prueba: 8	Numero Historia de Usuario: 6
Nombre Caso de Prueba: Realizar búsquedas en los formularios	
Descripción: 1- Seleccionar uno de los criterios de búsqueda 2- Hacer clic en el botón Buscar 3- Verificar lista de detalle	
Condiciones de Ejecución: Deben existir previamente creados los datos a buscar.	
Entradas: 1- Criterio de búsqueda.	
Resultado Esperado: Búsqueda solicitada.	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 70. Caso de prueba creación de Marcas

Caso de Prueba	
Numero Caso de Prueba: 9	Numero Historia de Usuario: 7
Nombre Caso de Prueba: Creación de Marcas	
Descripción: 1- Seleccionar el Cliente al cual se le agregará una Marca. 2- Llenar todos los campos correspondientes a la información general de la Marca. 3- Hacer clic en la pestaña Referencia. 4- Llenar cada uno de los campos correspondientes a la información de las referencias. 5- Hacer clic en la pestaña Autorización de Presupuesto. 6- Llenar los campos correspondientes a la persona que autoriza el presupuesto. 7- Hacer clic en la pestaña pagos. 8- Llenar los campos correspondientes a la persona que autoriza y realiza el pago.	
Condiciones de Ejecución: El operador deberá llenar los campos solicitados en cada formulario antes de pasar a la siguiente pestaña.	
Entradas: 1- Información general de la Marca. 2- Información de quien autoriza el presupuesto. 3- Referencias de la Marca. 4- Información de quien realiza los pagos.	
Resultado Esperado: Ingreso de Marca y sus referencias así como el ingreso de los datos de quien autoriza y realiza los pagos.	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 71. Caso de prueba edición de Marcas

Caso de Prueba	
Numero Caso de Prueba: 10	Numero Historia de Usuario: 7
Nombre Caso de Prueba: Edición de Marca	
Descripción: 1- Hacer clic en el botón Editar a la par del registro de la Marca. 2- Editar los campos que se deseen correspondientes a la información general de la Marca. 3- Clic en el botón guardar.	
Condiciones de Ejecución: El operador deberá llenar los campos solicitados en cada formulario antes de pasar a la siguiente pestaña.	
Entradas:	
Resultado Esperado: Nueva información de la Marca y sus referencias así como la modificación de los datos de quien autoriza y realiza los pagos.	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 72. Caso de prueba creación de usuarios

<b>Caso de Prueba</b>	
Numero Caso de Prueba: 11	Numero Historia de Usuario: 8
Nombre Caso de Prueba: Creación de Usuarios	
Descripción: 1- Llenar cada uno de los campos requeridos 2- Presionar el botón Crear Usuario	
Condiciones de Ejecución: Esta acción solo puede ser realizada por el Administrador del sistema.	
Entradas: Datos del Usuario y tipo de usuario.	
Resultado Esperado: Creación de Usuario	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 73. Caso de prueba edición de usuarios

<b>Caso de Prueba</b>	
Numero Caso de Prueba: 12	Numero Historia de Usuario: 8
Nombre Caso de Prueba: Edición de usuarios	
Descripción: 1- Seleccionar el usuario a editar. 2- Editar los campos necesarios.	
Condiciones de Ejecución: Esta acción solo puede ser realizada por el Administrador del sistema.	
Entradas: Datos del Usuario.	
Resultado Esperado: Edición de Usuario	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 74. Caso de prueba creación de presupuestos

<b>Caso de Prueba</b>	
Numero Caso de Prueba: 13	Numero Historia de Usuario: 9
Nombre Caso de Prueba: Crear Presupuesto	
Descripción: 1- Seleccionar la proyección de gastos correspondiente al presupuesto 2- Hacer clic en el botón Crear Presupuesto. 3- Verificar lista de detalle	
Condiciones de Ejecución: Deben existir previamente creadas las proyecciones de gastos.	
Entradas: 1- Datos generales de la proyección.	
Resultado Esperado: Creación del Presupuesto.	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 75. Caso de prueba aprobación de presupuestos

<b>Caso de Prueba</b>	
Numero Caso de Prueba: 14	Numero Historia de Usuario: 10
Nombre Caso de Prueba: Aprobar Presupuestos	
Descripción: 1- Seleccionar en el master el presupuesto a ser aprobado. 2- Hacer clic en el botón Aprobar presupuesto. 3- Verificar lista de detalle del presupuesto.	
Condiciones de Ejecución: Deben existir previamente creados los presupuestos.	
Entradas: 1- Presupuesto	
Resultado Esperado: Aprobación del Presupuesto.	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 76. Caso de prueba generación de reportes de presupuestos

<b>Caso de Prueba</b>	
Numero Caso de Prueba: 15	Numero Historia de Usuario: 11
Nombre Caso de Prueba: Generar reportes de Presupuestos	
Descripción: 1- Seleccionar en el master el presupuesto del que se requiera el reporte. 2- Hacer clic en el botón ver reporte.	
Condiciones de Ejecución: Deben existir previamente creados los presupuestos.	
Entradas: 1- Presupuesto	
Resultado Esperado: Reporte del presupuesto personalizado en pantalla.	
Evaluación: La prueba se concluyó satisfactoriamente.	

Tabla 77. Caso de prueba exportación de reportes de presupuestos

<b>Caso de Prueba</b>	
Numero Caso de Prueba: 16	Numero Historia de Usuario: 11
Nombre Caso de Prueba: Exportar reportes de Presupuestos.	
Descripción: 1- Una vez generado el reporte hacer clic en el botón guardar como y seleccionar el formato al que se desea exportar el reporte.	
Condiciones de Ejecución: Debe haberse previamente generado el reporte del presupuesto.	
Entradas: 1- Presupuesto	
Resultado Esperado: Reporte del presupuesto exportado a un archivo Excel o Pdf.	
Evaluación: La prueba se concluyó satisfactoriamente.	

## 11.4. RESULTADO 4: Tarjetas CRC

Tabla 77. Tarjeta CRC Ejecutivo

<b>Ejecutivo</b>	
<b>Responsabilidad</b>	<b>Colaboración</b>
Mostrar información de los ejecutivos	
Mostrar listado de los ejecutivos	

Tabla 78. Tarjeta CRC Cliente

<b>Cliente</b>	
<b>Responsabilidad</b>	<b>Colaboración</b>
Verificar datos del cliente	
Verificar datos del ejecutivo	Ejecutivo
Verificar marcas del cliente	Marca
Asignar ID al cliente	

Tabla 79. Tarjeta CRC Marca

<b>Marca</b>	
<b>Responsabilidad</b>	<b>Colaboración</b>
Verificar datos de marca	
Verificar datos del cliente	Cliente
Asignar ID a la marca	

Tabla 80. Tarjeta CRC Departamento

<b>Departamento</b>	
<b>Responsabilidad</b>	<b>Colaboración</b>
Mostrar información de los departamentos	
Mostrar listado de los departamentos	
Asignar rubros al departamento	Rubro
Asignar ID al departamento	

Tabla 81. Tarjeta CRC Rubro

<b>Rubro</b>	
<b>Responsabilidad</b>	<b>Colaboración</b>
Mostrar información de Rubro	
Mostrar listado de Rubros	

Tabla 82. Tarjeta CRC Proveedores

<b>Proveedores</b>	
<b>Responsabilidad</b>	<b>Colaboración</b>
Mostrar información de los proveedores	
Mostrar listado de proveedores	

Tabla 83. Tarjeta CRC Proyección de gastos

<b>Proyección de Gastos</b>	
<b>Responsabilidad</b>	<b>Colaboración</b>
Asignar proyección de gasto a cliente	Cliente
Asignar Departamento a la proyección de gastos	Departamento
Asignar Ejecutivo a cliente	Ejecutivo
Asignar ID a Proyección de gasto	
Asignar detalle a proyección de gasto	
Mostrar proyección de Gastos	

Tabla 84. Tarjeta CRC Proyección de Categoría

<b>Categoría</b>	
<b>Responsabilidad</b>	<b>Colaboración</b>
Mostrar información de las categorías	
Mostrar listado de las categorías	

Tabla 85. Tarjeta CRC Proyección de Presupuesto

<b>Presupuesto</b>	
<b>Responsabilidad</b>	<b>Colaboración</b>
Recuperar información de Proyección de Gasto	Proyección de Gasto
Calcular monto de presupuesto	
Asignar No de Presupuesto	
Mostrar el Presupuesto	

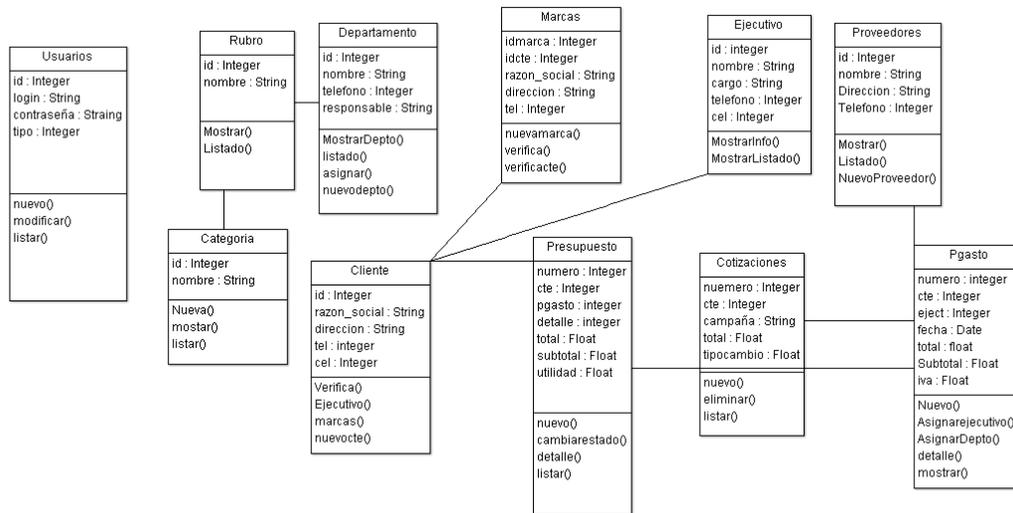
Tabla 86. Tarjeta CRC Cotizaciones

<b>Cotizaciones</b>	
<b>Responsabilidad</b>	<b>Colaboración</b>
Recuperar información de Proyección de Gasto	Proyección de Gasto
Calcular monto de la cotización	
Asignar No. De Cotización	
Mostrar Cotización	

Tabla 87. Tarjeta CRC Usuario

<b>Usuario</b>	
<b>Responsabilidad</b>	<b>Colaboración</b>
Mostrar información de los usuarios	
Mostrar listado de los usuarios	

### 11.5. RESULTADO 5: Diagrama de clases del Sistema de Gestión de Presupuestos.



### 11.6. RESULTADO 6: Bitácora de reuniones

Día	Lunes 23 de Septiembre del 2013
Horario	6:00PM - 7:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Reunión para planear entrevista y propuesta de sistema
Objetivo	Definir propuesta de proyecto para presentar a la gerencia administrativa de CEM Comunicaciones S.A.
Resultado	Conseguir de parte de la gerencia administrativa la aprobación y propuesta de sistema de acorde a su necesidad
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón

Día	Viernes 27 de Septiembre del 2013
Horario	1:00PM - 4:00PM
Lugar de encuentro	CEM.JWT División Internacional.
Actividades realizadas	Reunión con gerencia de administrativa
Objetivo	Determinar el alcance del sistema
Resultado	Estudio y manejo de las operaciones que se usan actualmente para generar presupuestos lo que conlleva a crear una idea más clara de cómo automatizar dichas operaciones.
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón, Jairo Narváez

Día	Sábado 28 de Septiembre del 2013
Horario	2:00PM - 5:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Sesión de seminario de graduación
Objetivo	Definir objetivos, nombre del proyecto, planteamiento del problema, metodología ágil de desarrollo de SF
Resultado	Metodología a utilizar (Programación Extrema), objetivos, Nombre, planteamiento del problema.
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón, Ainoa Calero

## METODOLOGÍAS ÁGILES DEL DESARROLLO DE SOFTWARE

---

Día	Jueves 03 de Octubre del 2013
Horario	6:00PM - 8:00PM
Lugar de encuentro	CEM.JWT División Internacional.
Actividades realizadas	Reunión con gerencia de administrativa
Objetivo	Planear las tareas de usuarios y requisitos
Resultado	Tareas de usuarios
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón, Jairo Narváez

Día	Sábado 5 de Octubre del 2013
Horario	2:00PM - 4:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Sesión de seminario de graduación.
Objetivo	Presentar los avances al tutor
Resultado	Corrección de las debilidades en lo trabajado previamente
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón, Ainoa Calero

Día	Jueves 10 de Octubre del 2013
Horario	6:00PM - 8:00PM
Lugar de encuentro	Casa de Marcos Fonseca
Actividades realizadas	Análisis de bosquejos, historias de usuarios, tareas de ingeniería.
Objetivo	Investigar la metodología en su aplicación y llevar a cabo los procesos siguientes al análisis
Resultado	Presentación de asignación para la sesión de clases del día sábado
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón

Día	Sábado 12 de Octubre del 2013
Horario	2:00PM - 4:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Sesión de seminario de graduación.
Objetivo	Presentar a la tutora Ainoa Calero avance del documento

## **METODOLOGÍAS ÁGILES DEL DESARROLLO DE SOFTWARE**

---

---

Resultado	Planificación para siguiente sesión de reunión
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón, Ainoa Calero

Día	Jueves 17 de Octubre del 2013
Horario	6:00PM - 8:00PM
Lugar de encuentro	Casa de Marcos Fonseca
Actividades realizadas	Desarrollo, plantillas y pantallas
Objetivo	Diseñar las pantallas mediante los bosquejos
Resultado	Presentación de asignación para la sesión de clases del día sábado
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón.

Día	Miércoles 23 de Octubre del 2013
Horario	2:00PM - 4:00PM
Lugar de encuentro	Casa de Engels Obregón
Actividades realizadas	Desarrollo de funcionalidad de inicio de sesión y clientes.
Objetivo	Culminar el inicio de sesión y clientes
Resultado	Aplicación de la metodología ágil de desarrollo XP mediante la programación en parejas, tomando en cuentas los roles.
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón, Ainoa Calero

Día	Lunes 04 de Noviembre del 2013
Horario	6:00PM - 8:00PM
Lugar de encuentro	Casa de Marcos Fonseca
Actividades realizadas	Pruebas de funcionamiento del sistema.
Objetivo	Encontrar fallas en sistema para corrección
Resultado	Modulo Clientes
Participantes	Marco Fonseca, Engels Obregón

Día	Sábado 9 de noviembre del 2013
Horario	2:00PM - 4:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Sesión de seminario de graduación.

## **METODOLOGÍAS ÁGILES DEL DESARROLLO DE SOFTWARE**

---

---

Objetivo	Presentar a tutora los avances y correcciones en el módulo clientes.
Resultado	Corrección de modulo clientes y propuestas para siguiente modulo
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón, Ainoa Calero

Día	Viernes 15 de Noviembre del 2013
Horario	6:00PM - 8:00PM
Lugar de encuentro	Casa de Leonte Espinoza
Actividades realizadas	Pruebas del módulo Clientes
Objetivo	Culminar el modulo clientes con las pruebas del tester para la presentación a tutora.
Resultado	Presentación de asignación para la sesión de clases del día sábado
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón

Día	Sábado 16 de Noviembre del 2013
Horario	2:00PM - 4:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Sesión de seminario de graduación.
Objetivo	Entrega del módulo Clientes concluido
Resultado	Entrega del módulo clientes e inicio del módulo proveedores
Participantes	Marco Fonseca, Leonte Espinoza, Engels Obregón, Ainoa Calero

Día	Jueves 21 de Noviembre del 2013
Horario	6:00PM - 8:00PM
Lugar de encuentro	Casa de Engels Obregón
Actividades realizadas	Desarrollo del módulo Proveedores
Objetivo	Avanzar con el modulo proveedores
Resultado	Modulo proveedores (Borrador)
Participantes	Marco Fonseca, Engels Obregón

Día	Sábado 23 de Noviembre del 2013
Horario	2:00PM - 4:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Sesión de seminario de graduación.
Objetivo	Presentar a tutora avance de código y funcionamiento de modulo proveedores para obtener ideas
Resultado	Corrección de modulo proveedores
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza, Ainoa Calero

Día	Miércoles 04 de Diciembre del 2013
Horario	6:00PM - 8:00PM
Lugar de encuentro	Casa de Engels Obregón
Actividades realizadas	Desarrollo del módulo Proveedores
Objetivo	Avanzar con el modulo proveedores
Resultado	Modulo proveedores culminado
Participantes	Marco Fonseca, Engels Obregón

Día	Sábado 14 de Diciembre del 2013
Horario	1:00PM - 4:00PM
Lugar de encuentro	Casa de Leonte Espinoza
Actividades realizadas	Pruebas al módulo de proveedores.
Objetivo	Culminar pruebas planteadas en la metodología para dar por finiquitado el modulo proveedores
Resultado	Culminación del módulo proveedores
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza

Día	Viernes 03 de Enero del 2014
Horario	6:00PM - 8:00PM
Lugar de encuentro	Casa de Marco Fonseca

## **METODOLOGÍAS ÁGILES DEL DESARROLLO DE SOFTWARE**

---

---

Actividades realizadas	Creación del módulo servicios
Objetivo	Tomar nota de ideas sobre creación del módulo servicios.
Resultado	Inicio del módulo servicios
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza

Día	Sábado 4 de Enero del 2014
Horario	3:00PM - 7:00PM
Lugar de encuentro	Casa de Marco Fonseca
Actividades realizadas	Creación del módulo Servicios
Objetivo	Diseñar interfaz y codificar el modulo servicios.
Resultado	Modulo Servicios en Previo
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza

Día	Jueves 09 de Enero del 2014
Horario	6:00PM - 8:00PM
Lugar de encuentro	Casa de Leonte Espinoza
Actividades realizadas	Revisión de modulo Servicios
Objetivo	Culminar Con el desarrollo del módulo Servicios
Resultado	Modulo servicios terminado
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza

Día	Sábado 11 de Enero del 2014
Horario	2:00PM - 4:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Sesión de seminario de graduación.
Objetivo	Presentar a tutora avance de código y funcionamiento de modulo Servicios
Resultado	Corrección de modulo servicios
Participantes	Marco Fonseca, Engels Obregón, Ainoa Calero

## METODOLOGÍAS ÁGILES DEL DESARROLLO DE SOFTWARE

Día	Martes 14 de Enero del 2014
Horario	6:00PM - 8:00PM
Lugar de encuentro	Casa de Leonte Espinoza
Actividades realizadas	Pruebas según metodología del módulo Servicios
Objetivo	Revisar el modulo Servicios
Resultado	Modulo Servicios culminado
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza

Día	Viernes 17 de Enero del 2014
Horario	6:00PM - 8:00PM
Lugar de encuentro	Metro centro - Managua
Actividades realizadas	Planificación de segunda iteración
Objetivo	Presentar a tutora avance de código y funcionamiento de modulo proveedores para obtener ideas
Resultado	Corrección de modulo proveedores
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza

Día	Sábado 18 de Enero del 2014
Horario	2:00PM - 4:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Presentación de Primera Iteración del Proyecto a tutora
Objetivo	Presentar a tutora Primera iteración del Proyecto
Resultado	Primera iteración culminada
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza, Ainoa Calero

Día	Miércoles 22 de Enero del 2014
Horario	6:00PM - 7:30PM
Lugar de encuentro	Restaurante Floridita – Plaza Inter
Actividades realizadas	Creación de bosquejos, historias de usuarios.
Objetivo	Empezar con segunda iteración
Resultado	Documentación necesaria para la empezar siguiente parte
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza, Jairo Narváez

## METODOLOGÍAS ÁGILES DEL DESARROLLO DE SOFTWARE

---

Día	Jueves 30 de Enero del 2014
Horario	2:00PM - 4:00PM
Lugar de encuentro	CEM.JWT División Internacional.
Actividades realizadas	Aceptación de la Primera Parte del Proyecto e inicio de la segunda parte (Retroalimentación)
Objetivo	Presentar a gerencia administrativa Primera Iteración e iniciar la segunda.
Resultado	Primera iteración culminada, segunda iniciada
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza, Jairo Narváez

Día	Viernes 7 de Febrero del 2014
Horario	6:00PM - 7:30PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Diseño de pantallas y formularios para la segunda iteración
Objetivo	Desarrollar la segunda iteración
Resultado	Pantallas de la segunda iteración
Participantes	Marco Fonseca, Engels Obregón

Día	Sábado 8 de Febrero del 2014
Horario	2:00PM - 4:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Presentación a tutora del avance
Objetivo	Mostrar a Tutora avance, para obtener correcciones
Resultado	Retroalimentación
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza, Ainoa Calero

Día	Jueves 20 de Febrero del 2014
Horario	6:00PM - 7:30PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Codificación de la segunda iteración
Objetivo	Desarrollar la segunda iteración

## **METODOLOGÍAS ÁGILES DEL DESARROLLO DE SOFTWARE**

---

Resultado	Funcionalidad en modulo presupuestos
Participantes	Marco Fonseca, Engels Obregón

Día	Martes 11 de Marzo del 2014
Horario	5:00PM - 8:00PM
Lugar de encuentro	Casa de Marco Fonseca
Actividades realizadas	Desarrollo de segunda iteración
Objetivo	Creación de Cotizaciones
Resultado	Creación de cotizaciones
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza, Ainoa Calero

Día	Viernes 14 de Marzo del 2014
Horario	6:00PM - 7:30PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Codificación de Cotizaciones
Objetivo	Desarrollar la segunda iteración
Resultado	Funcionalidad de Cotizaciones
Participantes	Marco Fonseca, Engels Obregón

Día	Sábado 15 de Marzo del 2014
Horario	2:00PM - 4:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Presentación a tutora del avance
Objetivo	Mostrar a Tutora avance, para obtener correcciones
Resultado	Retroalimentación
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza, Ainoa Calero

Día	Viernes 4 de Abril del 2014
Horario	6:00PM - 7:30PM
Lugar de encuentro	UNAN-Managua

## **METODOLOGÍAS ÁGILES DEL DESARROLLO DE SOFTWARE**

---

---

Actividades realizadas	Conclusión de Segunda Iteración
Objetivo	Realizar entrega de Segunda Iteración a cliente
Resultado	Conclusión de Segunda iteración
Participantes	Marco Fonseca, Engels Obregón

Día	Sábado 5 de Abril del 2014
Horario	2:00PM - 4:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Creación de presentación para exposición
Objetivo	Crear la presentación para la Presentación final del proyecto
Resultado	Preliminar de presentación de exposición
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza

Día	Sábado 12 de Abril del 2014
Horario	2:00PM - 4:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Presentación de diapositivas a tutora
Objetivo	Retroalimentar al equipo correcciones para presentación
Resultado	Mejora de presentación
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza, Ainoa Calero

Día	Lunes 14 de Abril del 2014
Horario	5:00PM - 8:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Planeación de Tercera Iteración
Objetivo	Planear tercera fase del proyecto
Resultado	Lluvia de ideas para tercera iteración
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza

## METODOLOGÍAS ÁGILES DEL DESARROLLO DE SOFTWARE

---

Día	Miércoles 16 de Abril del 2014
Horario	5:00PM - 8:00PM
Lugar de encuentro	CEM.JWT División Internacional.
Actividades realizadas	Creación de Historias de Usuarios
Objetivo	Crear Historias de usuarios
Resultado	Requisitos planteados por el cliente
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza, Jairo Narváez

Día	Viernes 18 de Abril del 2014
Horario	5:00PM - 8:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Creación de actividades (Tareas de ingeniería)
Objetivo	Desarrollar actividades para segunda iteración
Resultado	Actividades de tercera iteración
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza

Día	Lunes 21 de Abril del 2014
Horario	5:00PM - 8:00PM
Lugar de encuentro	CEM.JWT División Internacional.
Actividades realizadas	Creación de Historias de pantallas para tercera iteración
Objetivo	Crear Pantallas para tercera iteración
Resultado	Pantallas de tercera iteración
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza, Jairo Narváez

Día	Viernes 25 de Abril del 2014
Horario	5:00PM - 8:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Codificación de administración de usuarios
Objetivo	Codificar Administración de Usuarios
Resultado	Funcionalidad de Administración de Usuarios
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza

## METODOLOGÍAS ÁGILES DEL DESARROLLO DE SOFTWARE

---

Día	Viernes 2 de Mayo del 2014
Horario	5:00PM - 8:00PM
Lugar de encuentro	Casa De Marco Fonseca
Actividades realizadas	Codificación de Proyección de Gastos.
Objetivo	Codificar la proyección de gastos.
Resultado	Proyección de gastos
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza.

Día	Viernes 9 de Mayo del 2014
Horario	5:00PM - 8:00PM
Lugar de encuentro	UNAN-Managua
Actividades realizadas	Revisión de tercera iteración
Objetivo	Revisar funcionalidad de tercera iteración
Resultado	Corrección de puntos de interés en tercera iteración
Participantes	Marco Fonseca, Engels Obregón, Leonte Espinoza

Día	Lunes 12 de Mayo del 2014
Horario	5:00PM - 8:00PM
Lugar de encuentro	Casa De Marco Fonseca
Actividades realizadas	Codificación de reportes
Objetivo	Codificar de reportes
Resultado	Reportes principales del sistema
Participantes	Marco Fonseca

Día	Viernes 16 de Mayo del 2014
Horario	5:00PM - 8:00PM
Lugar de encuentro	Casa de Marco Fonseca
Actividades realizadas	Revisión de Reportes
Objetivo	Revisar funcionalidad de reportes
Resultado	Corrección de puntos de interés en reportes
Participantes	Marco Fonseca, Engels Obregón.