

**Universidad de las Ciencias Informáticas
FACULTAD 6**



Título: Desarrollo de una herramienta para la planificación y control del presupuesto en el Centro GEYSED.

Trabajo de Diploma para optar por el título de
Ingeniero Informático

Autores: Elizabeth Salazar Laffita.

Fabián Guerra Padrón.

Tutor: Ing. Héctor René Sánchez Falero.

Co-Tutor: Sandor Escobar Ruiz

Junio, 2013

Yo conozco tus obras; he aquí, he puesto delante de ti una puerta abierta, la cual nadie puede cerrar; porque aunque tienes poca fuerza, has guardado mi palabra, y no has negado mi nombre.

Apoc 3.8

Declaración de Autoría

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

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

Elizabeth Salazar L.

Fabián Guerra P.

Héctor R Sánchez F.

Firma del Autor

Firma del Autor

Firma del Tutor

Datos de Contacto

Síntesis del Tutor:

Nombre y apellidos: Héctor René Sánchez Falero.

Correo electrónico: hramírez@uci.cu

Categoría científica: Trabajador docente.

Profesión: Ingeniero en Ciencias Informáticas.

Síntesis del Co-Tutor:

Nombre y apellidos: Sandor Escobar Ruíz.

Correo electrónico: sescobar@uci.cu

Categoría científica: Trabajador no docente

Profesión: Ingeniero en Ciencias Informáticas.

Agradecimientos

Desde el principio de este curso declaré "Mi tesis en las manos del DIOS TODOPODEROSO!!! AMEN" y hoy, después de tantos días, después de haber pasado momentos de prueba, puedo decir: Gracias Dios, gracias porque ciertamente Tú has estado conmigo, gracias porque he visto Tu mano obrando a mi favor, gracias porque cuando no podía Tú me ayudaste, gracias porque cuando nadie estaba Tú me acompañaste, gracias porque eres Tú el que levanta mi cabeza. Gracias Jesús, gracias por ser mi mejor amigo y mi Gran Amor, separada de ti nada puedo hacer, gracias porque aunque vino el enemigo como río Tú levantaste bandera, gracias porque eres mi alto refugio, mi fortaleza, mi escudo y mi libertador. Gracias Espíritu Santo, por ser mi Consolador, por enseñarme, por guiarme y por ayudarme a tomar las decisiones correctas.

A mis padres también doy gracias, ustedes dos son las personas que más amo en toda la tierra. Gracias por estar conmigo en todo momento, porque nunca me han dado la espalda. Papi y Mami, gracias por cuidarme durante estos 23 años, por darme todo lo que he necesitado, gracias por darme aun lo que no merecía. Hoy quiero decirles que los amo mucho, más de lo que se imaginan, y si lo imaginan los amo mucho más. Papi, tú me has enseñado mucho, ¿qué hija no admira a su padre? Yo te admiro, y te doy gracias por tus consejos, por tu fidelidad, por amarme y querer lo mejor para mí, yo quiero cuidarte y que te apoyes en mí, ¿qué sentido tendría el deporte sin ti?, te amo y siempre seré "Tu Puti". Mami, gracias por todo lo que has hecho por mí, por quitarte lo tuyo para dármelo a mí, gracias por tu paciencia

Agradecimientos

y por la educación que me has dado, ¿qué sería de mi inglés sin ti?, te amo mami, siempre seré "Tu Lizi".

Hermano de mi alma, Rafe, si supieras cuánto te amo. Tú eres el regalo que Dios me ha dado. Cada día yo quiero tu bien y tu felicidad. Gracias por contar conmigo, por llorar conmigo, por ser mi amigo. Gracias por estar a mi lado, por escuchar música de la radio junto a mí ¿te acuerdas? Te amo mucho, siempre seré "Tu Poto".

A mi tía Lulín, a Omar y a Yodi les doy muchas gracias, porque las puertas de su casa y de sus vidas siempre han estado abiertas para mí, no tengo palabras para expresar mi gratitud, gracias por amarme y siempre estar cerca de mí. Tía Lulín, gracias por mostrarme al Dios verdadero y por no cesar de orar por mí. Gracias doy a mi tía Memé, a Luis y a Lili, ustedes también son muy importantes para mí. A mi tía Ada también doy gracias por ser atenta conmigo y a mi tía Mirella por tenerme presente. A mi abuelas Frita y Georgina, que aunque ya no estén siempre recordaré todo el amor que me dieron.

Gracias doy a mis amistades especiales del pre, Arianna, Reinier, Barrera, Jorlis, Raysel, Joenis y Dainier, quienes a pesar de estos 5 años en los cuales nos hemos visto solo en vacaciones, se han mantenido fieles y han sabido quererme y cuidar nuestra amistad. Ustedes son un tesoro para mí, los quiero mucho a todos.

Mis amistades de la UCI no se quedan atrás, Frank, Charlesbo, Mark, los llevo en mi corazón, gracias a ustedes por los momentos que pasamos juntos.

Agradecimientos

Al pueblo de la UCI, mis hermanos en Cristo, muchas gracias, llegaron en el momento indicado y se han convertido en mi familia, en mi mano derecha, han sido una bendición de Dios para mi vida. A Sailormoon, a Reinier Martínez y a Yupi, mis agradecimientos, por todo, cada tiempo juntos, cada comida, cada detalle, cada palabra, cada excursión, cada enseñanza; gracias doy a Dios por haberles conocido, los quiero mucho y lo que pueda decir no es suficiente para expresar todo mi amor por ustedes. A Mechi gracias por brindarme tu amistad, por las pijamadas, por tu apoyo. Al tío José, gracias por ayudarme para que pudiera salir bien.

Gracias a los hermanos de Murgas y a la Pastora Elizabeth por el cariño y la atención que me brindaron. Y a mis queridos hermanos de la Sua Iglesia Bautista, muchas gracias por sus oraciones.

Gracias a Juanita, mi vecina predilecta, gracias por estar siempre al pendiente de mí, por extenderme tu mano amiga.

A mi primer tutor Roexcy, gracias; y al último, Héctor René quien ha dado el frente para que este trabajo saliese adelante.

A Gretchen por responder a todas mis preguntas.

Agradecimientos

Una vez más, gracias Dios, sin Ti no hubiese llegado hasta aquí. A Ti sea toda la Gloria por los siglos de los siglos. Amén.

Elizabeth Salazar Laffita.

Agradecimientos

A todos los que han contribuido, de una forma u otra, a mi formación como profesional, especialmente:

A toda mi familia por el apoyo incondicional y la confianza que han depositado en mí, especialmente a mis padres, por el amor que me han dado en todo momento.

A todos mis buenos amigos, por pasar tan agradables y a veces no tan agradables momentos, pero siempre juntos, y por hacerme saber que en cualquier momento puedo contar con ellos, al igual que ellos conmigo.

A mi tutor, por la ayuda en todo momento.

A mi cotutor, por aguantarme cada vez que me ponía malcriado y no lo dejaba trabajar.

A todas estas personas especiales para mí, quiero darle las gracias por brindarme toda su ayuda, apoyo y comprensión para poder lograr hacer mi sueño realidad.

Fabián Guerra Padrón.

Dedicatoria

Dedico este trabajo a mi Dios Eterno y a mis padres, Teresa y Rafael.

Elizabeth Salazar Laffita.

Principalmente, con mucho cariño, a mis padres, Esther y Juan Carlos, por ser las personas que además de darme la vida, me han guiado y ayudado a llegar hasta aquí y que sé que están orgullosos de mí y eso me hace la persona más feliz del mundo.

A aquellos que actuando desde las sombras contribuyeron a sostenerme en todo lo que he logrado.

A mi abuelos, en especial a mis abuelas Olga y Martha que me ha malcriado muchísimo, y a mis abuelos Machito y Roberto, que aunque no estén físicamente se que siempre estarán conmigo.

A mis hermanos en especial a Karla, porque a pesar de las discusiones constantes la quiero con la vida.

A mis tíos, principalmente a Adrian, por quererme tanto y preocuparse por mí.

A mis primos, especialmente a mi primo Ruslan que más que un primo es mi hermano y siempre ha estado ahí para mí en todo momento.

A mis hermanos que no son de sangre pero que los quiero como si lo fueran: Adrian, por estar desde el círculo cuidando mis pasos, a Michael por demostrarme que siempre que se quiere se puede, a Sergio mi hermano en la vida, mi enemigo en el deporte.

A Sandor por ayudarme siempre incondicionalmente en todo.

A mis amistades José, Triana, Alexander, Pancho, Aliu, Reiner, Chao, Leandro, Ilen, Ortiz, Arlet, en fin, a todas las personas que me han aguantado y apoyado durante estos años.

A todos los presentes.

Con ustedes comparto mi triunfo, ya que a ustedes se los debo.

Fabián Guerra Padrón.

Resumen

En el Centro de Geoinformática y Señales Digitales (GEYSED), el asesor económico operaba con una gran cantidad de documentos, lo que hacía el trabajo riesgoso, al existir una alta probabilidad de que se cometieran errores. A raíz de esto, la entrega del plan presupuestario anual no se realizaba en tiempo y el control de los gastos reales solamente se efectuaba a nivel de centro, sin detallar las necesidades de cada proyecto. Teniendo en cuenta estas dificultades, se realizó un estudio acerca de las herramientas informáticas que se han desplegado en Cuba y en el mundo y se vinculan al campo de acción, el cual arrojó como resultado el descarte de las mismas por cuanto no daban solución al problema existente. El objetivo de este Trabajo de Diploma es elaborar un sistema informático que permita la planificación y el control del presupuesto en el Centro GEYSED. Para el desarrollo del mismo se escogió la metodología de desarrollo, *eXtreme Programming* (XP), pues promueve una constante interacción con el cliente y se emplearon las siguientes tecnologías: lenguaje de programación, C++ y Sistema Gestor de Base de Datos (SGBD), PostgreSQL.

Palabras Claves: control, gastos, planificación, presupuesto.

Abstract

In Center of Geoinformatics and Digital Signals (CGDS), the economic adviser was operating with a huge amount of documents, and the work was risky because it was easy make a lot of mistakes. Because of that, it not delivered on time the annual budgetary plan and the real expenses control was made only at center level, without the details of necessary of the projects. Having in mind these difficulties, it made a study about informatics tools that were deployed in Cuba and around the world. As result, these tools were discarded. The objective of this work is to develop an informatics system for planning and control budgets in CGDS's Center. For its development was chosen methodology eXtreme Programming (XP), because it promotes a constant interaction with the client. Also, it used the next technologies: programming language, C++; and System Manager Database, PostgreSQL.

Keywords Used: budget, control, expenses, planning.

Tabla de Contenidos

INTRODUCCIÓN.....1

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA SOBRE LOS PROCESOS DE PLANIFICACIÓN Y CONTROL DEL PRESUPUESTO.....5

 Introducción5

 1.1 Presupuesto.....5

 1.2 Herramientas informáticas existentes7

 1.3 Proceso de Desarrollo de Software.....11

 1.4 Herramientas para el desarrollo del sistema.....13

 1.5 Conclusiones del Capítulo.....17

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA PARA LA PLANIFICACIÓN Y EL CONTROL DEL PRESUPUESTO.....18

 Introducción18

 2.1 Descripción del Sistema18

 2.2 Personas que interactúan con el sistema18

 2.3 Requisitos del sistema.....19

 2.4 Fase de Planificación.....20

 2.5 Arquitectura Candidata23

 2.6 Conclusiones del Capítulo.....25

CAPÍTULO 3. PLANIFICACIÓN Y DISEÑO DEL SISTEMA PARA LA PLANIFICACIÓN Y EL CONTROL DEL PRESUPUESTO.....26

 Introducción26

 3.1 Estimación de esfuerzos por Historia de Usuario26

 3.2 Plan de Iteraciones.....27

 3.3 Plan de entrega28

Tabla de Contenidos

3.4 Fase de diseño	29
3.5 Conclusiones del Capítulo	33
CAPÍTULO 4. IMPLEMENTACIÓN Y PRUEBA DEL SISTEMA PARA LA PLANIFICACIÓN Y EL CONTROL DEL PRESUPUESTO.....	34
Introducción	34
4.1 Estándares de Codificación.....	34
4.2 Tareas de programación	34
4.2 Patrones de Diseño	38
4.3 Pruebas	40
CONCLUSIONES GENERALES	50
REFERENCIAS BIBLIOGRÁFICAS	52
BIBLIOGRAFÍA.....	56
GLOSARIO	62

Índice de Tablas

Tabla 1.Herramientas existentes	10
Tabla 2. Personal que interactúa con el sistema	18
Tabla 3. Historia de Usuario.	21
Tabla 4. Representación de la Historia de Usuario No.3	22
Tabla 5. Representación de la Historia de Usuario No.6	22
Tabla 6. Representación de la Historia de Usuario No.8	23
Tabla 7. Representación de la Historia de Usuario No.9	23
Tabla 8. Estimación de esfuerzo por Historia de Usuario	27
Tabla 9. Plan de duración de las iteraciones.....	28
Tabla 10. Plan de Entrega	29
Tabla 11. Plan de duración de entrega	29
Tabla 12. Clase Principal.....	30
Tabla 13. Clase Conexión	30
Tabla 14. Clase Fórmula	31
Tabla 15. Clase Graficar.....	31
Tabla 16. Clase Exportar	31
Tabla 17. Clase Buscar	31
Tabla 18. Clase Cambiar_Pass.....	31
Tabla 19. Descripción de los campos de una Tarea de programación	35
Tabla 20. Tareas de programación por cada Historia de Usuario	36
Tabla 21. Descripción de la Tarea de programación HU6_1	37
Tabla 22. Descripción de la Tarea de programación HU6_2	37
Tabla 23. Descripción de la Tarea de programación HU10_1	38
Tabla 24. Descripción de la Tarea de programación HU13_1	38
Tabla 29. Descripción de los campos de un Caso de Prueba	45

Índice de Tablas

Tabla 30. Descripción del Caso de Prueba de Aceptación 6	46
Tabla 31. Descripción del Caso de Prueba de Aceptación 7	46

Índice de Figuras

Figura 1. Codificación de la funcionalidad graficar.42

Figura 2. Grafo de Flujo asociado a la funcionalidad graficar.....43

Figura 3.Resultado de las pruebas en la 1ra Iteración47

Figura 4.Resultado de las pruebas en la 2da Iteración48

Figura 5.Resultados generales de las pruebas.....48

Introducción

La economía mundial del siglo XXI se encuentra en una situación de crisis, considerada la más grave de la historia moderna según los autores del libro “La crisis económica global, la Gran Depresión del Siglo XXI”. El endeudamiento de las instituciones financieras y el crecimiento acelerado en los precios del PIB (Producto Interno Bruto), son muestra de los efectos de este fenómeno. En medio de este entorno inestable y competitivo, las empresas en todo el mundo intentan sobrevivir, pretendiendo acomodarse al contexto económico actual. (Chossudovsky, 2010) Por esta razón el buen manejo del presupuesto se convierte en un elemento a tomar en cuenta a la hora de fomentar el desarrollo de la actividad económica en una organización, en un período de tiempo determinado. (Montoya, y otros, 2013)

A través de la planificación del presupuesto se estiman una serie de recursos para un período determinado y se implementan planes de acción de forma monetaria. Mientras que mediante el control presupuestario se pueden realizar análisis detallados de los gastos reales en los que se ha incurrido, es decir, con el control se vela por lo planificado. (Guillermo, 2012)

Cuba, desde el primero de Enero de 1959 ha tratado de buscar un modelo económico eficiente. Por tal razón, a partir del año 1996 comienzan a realizarse transformaciones en el sector empresarial con la aplicación del Perfeccionamiento Empresarial, adecuándolo a la realidad de la economía socialista y sin dejar a un lado el anhelo de estar a la altura de las empresas de éxito en el mundo. Los principios generales del Perfeccionamiento Empresarial escritos en el Decreto Ley No.187 del año 1998, exigen que la empresa presupuestada cubana cubra sus gastos con sus ingresos y elabore sus planes anuales, tareas que dependen principalmente de una correcta planificación y control del presupuesto. (Dra Zaldívar, 2007)

Como consecuencia de la implantación del Perfeccionamiento Empresarial, y su planteamiento de que la innovación tecnológica debe estar presente como un elemento básico de desarrollo, las organizaciones empresariales y presupuestadas cubanas se encuentran transitando por un proceso de autocontrol o control interno. El cual se ha visto apoyado por el auge que ha tomado la informática en la actualidad. Posibilitando el diseño de herramientas informáticas que permitan la planificación y el control del presupuesto, las cuales contribuyen con la eliminación de la desorganización que conduce al derroche, el robo y la negligencia. (Dra Zaldívar, 2007)

La Universidad de las Ciencias Informáticas (UCI), es una entidad presupuestada perteneciente al Ministerio de Educación Superior (MES), la cual pretende servir de soporte a la industria cubana del software. La UCI en este sentido puede apoyar a los procesos económicos en instituciones del país que lo requieran así como en su infraestructura empresarial. En la facultad 6 de dicha universidad se encuentra el Centro de Geoinformática y Señales Digitales (GEYSED), el cual presenta como misión el desarrollo de productos y soluciones informáticas en el campo del procesamiento de Señales Digitales y la Geoinformática.

El asesor económico del Centro GEYSED es el encargado de llevar a cabo la planificación del presupuesto de gasto del centro y el control del mismo de forma mensual, trimestral y anual. Esta tarea se ha vuelto muy compleja pues se lleva a cabo de forma manual, mediante el uso excesivo del Microsoft Excel. Esto dificulta el trabajo llevado a cabo durante los procesos de planificación y control, porque se utiliza una gran cantidad de documentos para luego brindar una única información. Por esta razón no se ha realizado en tiempo la entrega de dicha información al Vicedecano de Administración de la facultad 6 y al Departamento de Planificación de la Universidad. Además, con frecuencia se cometen errores, lo cual hace que los datos no sean reales del todo. Igualmente el asesor económico necesita realizar la ejecución del presupuesto de gasto a nivel de proyecto, con el objetivo de conocer cómo incide cada proyecto en la ejecución real del presupuesto, debido a que si hay una variación positiva (ejecución) o negativa (sobre-ejecución) en los diferentes elementos del gasto, se podría identificar qué o quién es el que lo está provocando.

Por las razones expuestas anteriormente el **problema a resolver** queda planteado de la siguiente manera: ¿Cómo gestionar los procesos de planificación y control del presupuesto en el Centro GEYSED?

El **objeto de estudio** lo constituye, los procesos de planificación y control del presupuesto.

El **campo de acción** se enmarca en la planificación y control del presupuesto en el Centro GEYSED.

Donde se plantea como **objetivo general**: desarrollar una herramienta que facilite la planificación y control del presupuesto en el Centro GEYSED.

La idea a defender queda definida en el siguiente planteamiento: Con la implementación de la herramienta que facilite la planificación y control del presupuesto en el Centro GEYSED, se garantiza la entrega a tiempo de los reportes del Plan Anual y del Control de los Gastos Reales al Vicedecano de Administración de la facultad 6 y al Departamento de Planificación de la Universidad.

Para darle cumplimiento al objetivo general se proponen las siguientes **tareas de investigación**:

- Caracterizar el proceso de planificación y control del presupuesto.
- Caracterizar a nivel nacional e internacional otras herramientas para la planificación y control del presupuesto.
- Caracterizar los métodos, herramientas y procedimientos más factibles para el desarrollo de la aplicación.
- Desarrollar una herramienta que garantice el proceso de planificación y control del presupuesto en GEYSED.

Se emplean los métodos científicos de la investigación: Teóricos y Empíricos. De los Métodos Teóricos se emplearon:

- El **Analítico-Sintético**: Este método se emplea para precisar las variables y el procesamiento de los datos obtenidos en los diferentes instrumentos aplicados (entrevista de tipo informal) en correspondencia con el objetivo propuesto (Hernández, y otros, 2011). Su uso posibilitó la realización del estudio teórico de la investigación, el análisis acerca de los procesos de planificación y control del presupuesto permitiendo extraer sus elementos más importantes.
- El **Histórico-Lógico**: Se utiliza para conocer los antecedentes y elementos de la investigación referidos a los procesos de planificación y control y la determinación de las tendencias actuales sobre la temática que se estudia (Hernández, y otros, 2011). Su empleo permitió investigar acerca de las herramientas informáticas existentes en Cuba y en el mundo que gestionan presupuestos.
- La **Modelación**: Es el método mediante el cual se crean abstracciones con el objetivo de explicar la realidad (Hernández, y otros, 2011) Se utilizó en la modelación de la base de datos mediante el lenguaje unificado de modelado (UML).

De los Métodos Empíricos se utilizó:

- La **Entrevista**: Se realiza para conocer los diferentes puntos de vista sobre el tema de la investigación y saber datos específicos del centro (Hernández, y otros, 2011), con el fin de obtener información valiosa que posibilite el avance eficiente de la investigación y el logro de sus objetivos. El tipo de entrevista utilizada es: informal, individual y no estructurada. Las preguntas realizadas al asesor económico del Centro GEYSED se encuentran en el Anexo 2 de la presente investigación.

El presente trabajo está estructurado en cuatro capítulos:

Capítulo 1. Fundamentación Teórica sobre los procesos de Planificación y Control del Presupuesto: Este capítulo aborda la fundamentación teórica que sirve de base a la investigación del problema planteado. Se realiza un estudio del estado del arte del tema asociado, a nivel nacional e internacional. Queda plasmada una síntesis de las herramientas existentes referentes al problema identificado y se describen la metodología y tecnologías en las que se apoya la solución del problema.

Capítulo 2. Características del sistema para la Planificación y el Control del Presupuesto: En este capítulo se definen las características del sistema, los requisitos funcionales y no funcionales. Se realiza la fase de Planificación, mostrando las historias de usuario. La arquitectura de software queda definida en este capítulo.

Capítulo 3. Planificación y Diseño del Sistema para la Planificación y el Control del Presupuesto: En este capítulo se realiza la documentación concerniente a las fases de planificación y diseño. Se realiza la estimación del esfuerzo por historia de usuario, se confecciona el plan de iteraciones y el plan de duración de las iteraciones, el plan de entregas, así como el plan de duración de las entregas. También quedan elaboradas las Tarjetas CRC.

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto: Este capítulo contiene las fases de desarrollo y prueba según la metodología a utilizar. Se muestran las tareas a desarrollar por cada historia de usuario y el resultado de las pruebas realizadas al sistema, además de los estándares de programación utilizados.

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

Introducción

En el presente capítulo se describen las principales características del presupuesto como aspectos fundamentales para el desarrollo de la investigación. Se analizan algunas soluciones informáticas, en el ámbito nacional e internacional, que gestionan presupuestos. Además se describen las características de la metodología de desarrollo, lenguaje de programación, entorno de desarrollo integrado, lenguaje de modelado y herramienta Case, como elementos propuestos para lograr un desarrollo exitoso de la herramienta para la planificación y el control del presupuesto en el Centro GEYSED.

1.1 Presupuesto

El presupuesto como instrumento clave de la gestión empresarial llega a ser tan significativo que hasta el momento se desconoce que otro tipo de medio los puede suplir. Por consiguiente si en una organización estos no funcionan o no se obtienen los resultados esperados se debe replantear la forma en que se realizan. Tal es su importancia que muchos autores han conceptualizado sobre el tema.

Un presupuesto es un plan integrador que se expresa en términos financieros con respecto a las operaciones y recursos que forma parte de una empresa para un período determinado, con el fin de lograr los objetivos fijados por la gerencia general. (Burbani, 2005)

El presupuesto es un plan de acción dirigido a cumplir una meta prevista, expresada en valores y términos financieros que, debe cumplirse en determinado tiempo y bajo ciertas condiciones previstas. (Leiseca, 2010)

Luego de haber estudiado los conceptos definidos por otros autores, referentes al presupuesto, los autores del presente trabajo consideran que, el presupuesto es un instrumento utilizado para la planificación y el control de recursos con fines económicos. Puede realizarse con periodicidad y consta de objetivos, los cuales permiten tener anticipadamente todo lo necesario para la elaboración y ejecución del mismo.

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

Objetivos de los presupuestos

La elaboración del presupuesto persigue objetivos tales como (Burbani, 2005):

- Planear sistemáticamente todas las actividades que la empresa debe desarrollar en dinero y volúmenes, en un período determinado.
- Controlar el manejo de ingresos y egresos, medir los resultados cuantitativos y cualitativos y fijar responsabilidades en las diferentes áreas de la empresa para lograr el cumplimiento de las metas previstas.
- Coordinar los diferentes centros de costo para que se asegure la marcha de la compañía en forma integral.

Planificación del presupuesto

Es la primera etapa del proceso administrativo y está muy ligada a las demás etapas (organización, dirección y control). Es un proceso de toma de decisiones, las cuales deben ser anticipadas e interrelacionadas. Esta determina los objetivos a cumplir y lo que se debe hacer para alcanzarlos. También establece a dónde se quiere llegar, qué se debe hacer, cómo, cuándo y en qué orden. Este proceder permite el incremento de la eficiencia y elimina en cierta medida la improvisación.(Welsch, 2005)

Planificación es el proceso de desarrollar objetivos empresariales y elegir un futuro curso de acción para lograrlos. Comprende a) establecer los objetivos de la empresa, b) elegir un curso de acción para alcanzar los objetivos, c) iniciar las actividades necesarias para traducir los planes en acciones d) re planear sobre la marcha para corregir deficiencias existentes.(Welsch, 2005)

El objetivo fundamental de la planificación empresarial es la elaboración del sistema de planes económicos. Planificar el presupuesto, como parte de este proceso, permite cuantificar los objetivos y metas de la actividad productiva, económica y financiera de la empresa.

Control del presupuesto

Todo proceso de planificación requiere un mecanismo de control para que sea efectivo, ya que es el esfuerzo sistemático para hacer coincidir los resultados con los objetivos propuestos. Controlar es el proceso de asegurar el desempeño eficiente para alcanzar los objetivos de la empresa. Implica: a)

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

establecer metas y normas, b) comparar el desempeño medido contra las metas y normas establecidas, c) reforzar los aciertos y corregir las fallas. (Welsch, 2005)

El control presupuestario es el proceso de descubrir qué es lo que se está haciendo, comparando los resultados con los datos presupuestados correspondientes para verificar los logros o remediar las diferencias, y comprobar las desviaciones existentes. Es indispensable para controlar la gestión de la empresa y anticiparse a los problemas que se pueden producir. Consiste en la comparación de los ingresos y gastos reales con los que se evaluaron y determinaron en la planificación presupuestaria, a fin de establecer y analizar las causas que puedan haber motivado la aparición de las diferencias entre estos datos. (Welsch, 2005)

1.2 Herramientas informáticas existentes

Disímiles son las aplicaciones informáticas que presentan como objetivo fundamental procesar la información contable, debido a la necesidad que tienen las entidades de tener a su alcance toda la información de manera organizada para el instante en que sea necesaria. Una búsqueda desarrollada por los autores arrojó que, en el plano nacional e internacional existen herramientas que se encargan, entre otras funciones, de facilitar la planificación y el control del presupuesto, por ejemplo:

PRESTO: Es una herramienta internacional desplegada por la empresa desarrolladora de productos informáticos Soft. Es la más utilizada para realizar presupuestos de edificación y obra civil y empresas constructoras. Entre sus atractivos se encuentra que: (De Benito, y otros, 2007)

- Permite realizar fácilmente presupuestos y ofertas reutilizando información de proyectos anteriores o bases de datos de precios.
- Es un sistema diseñado para Windows 2000\NT\XP o Superior.

Presenta como recursos avanzados los siguientes: (De Benito, y otros, 2007)

- Edición simultanea de varios presupuestos o bases de datos de precios.
- Uso de arrastrar y soltar, copiar y pegar para traspasar información entre ventanas, entre presupuestos o con otros programas, como hojas de cálculo y editores de textos.
- Utilidades para copiar, pegar, renombrar, mover, eliminar, comprimir, descomprimir y enviar un presupuesto por correo electrónico sin abandonar Presto.

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

Esta herramienta a pesar de contar con funcionalidades presupuestales no se ajusta a los requisitos necesarios para resolver el problema de la planificación y el control del presupuesto en el Centro GEYSED. Primeramente porque está diseñada para efectuar presupuestos de edificación y obra civil, lo cual está muy distante de los objetivos que se persiguen con la elaboración del presente trabajo. No cuenta con algún mecanismo que permita la creación de gráficos para realizar análisis. Sus requisitos de instalación no cumplen con la política del país de migrar a Software Libre, esto hace que se incurra en un gasto elevado en pagos de licencia de software. Presto garantiza la planificación del presupuesto pero no permite que se realice el control del mismo, funcionalidad que debe estar presente en el sistema que se desea implementar.

VERSAT-SARASOLA: Es un Software Integrado creado en Cuba, orientado a la Gestión Empresarial y Presupuestaria. Presenta como característica que está diseñado para utilizarse en cualquier tipo de entidad. Permite llevar el control y el registro contable de todos los hechos económicos que se originan en las empresas. (Martín, 2005). Elimina la dependencia del fabricante extranjero al tener la base de trabajo y los programadores en el país.(Porteiro, y otros, 2007)

Versat Sarasola está formado por 12 módulos o subsistemas, entre los cuales se encuentra el módulo de Finanzas, Caja y Banco. Este es el subsistema más abarcador del Versat Sarasola pues está conformado por cinco actividades: Caja, Banco, Cobros y Pagos, Crédito Bancario y Otras Operaciones Financieras. Es en esta última actividad que se encuentra el manejo de los presupuestos. Sin embargo las funcionalidades que realiza no se necesitan para planificar y controlar el presupuesto en el centro GEYSED de la facultad 6. Dichas funcionalidades se mencionan a continuación:

- Planificar los pagos a efectuar que generan salida de efectivo.
- Planificar los cobros a efectuar que generan entrada de efectivos.
- Se realiza un documento que recoge los aportes del Presupuesto del Estado, como por ejemplo, el pago de la seguridad social.(VERSAT, 2012)

Además, en el módulo Costos y Procesos se realiza la actividad Elementos de Gastos, en donde se definen los elementos de gastos que se utilizarán en el sistema. Para lo cual se encuentran las opciones

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

Adicionar elemento y Eliminar elemento, pero al ejecutar esta última se borran todas las áreas asociadas. Considerándose de esta manera, que este subsistema cumple parcialmente con uno de los requisitos presentes en la herramienta a implementar en el centro GEYSED. (VERSAT, 2012) Por la información antes mencionada, se considera que este sistema no se corresponde con las necesidades presentadas por el asesor económico del Centro GEYSED para el desarrollo de una herramienta que permita la planificación y control del presupuesto.

SISTEMA DE CONTROL DE SOLICITUDES AL PRESUPUESTO: Sistema informático desarrollado en la UCI para controlar el flujo de trabajo relacionado con las solicitudes de compra y ejecución del presupuesto de la Vice-Rectoría Económica. Para llevarlo a cabo se utilizó como lenguaje de programación C#.NET y como gestor de base de datos SQL Server 2000. El mismo consta de dos módulos, el primero recoge todo el flujo de trabajo que se realiza en el área de Planificación y Estadísticas, y el segundo recoge todo el proceso que se realiza en el área de Finanzas. Su principal objetivo es la informatización de todos los procesos que realiza la VR Económica y dentro de ellos se determinó con mayor prioridad el control presupuestario y financiero de las solicitudes que se presentan en el comité de compras. A continuación se presentan las funcionalidades que el sistema proporciona:

- Inserción de solicitudes al sistema alertando cuando no existe presupuesto para procesar dicha solicitud.
- Adiciona cuentas bancarias así como el presupuesto para cada una de ellas y el saldo inicial en banco con que inician el año.
- Adiciona todos los elementos de gastos.
- Adiciona todos los especialistas que participan en el comité de compras.
- Gestiona el estado de la solicitud (enviada al rector, no aprobada, aprobada, enviada a finanzas).
- Inserta cheques asociándolos a la solicitud correspondiente.
- Presenta reportes para facilitar el análisis y procesamiento de la información existente en el sistema.

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

La herramienta efectúa el control de presupuestos, pero no lo planifica, razón que fortalece la decisión de no emplearla en el Centro GEYSED. Además realiza funcionalidades que el cliente del presente trabajo de diploma no necesita.

En la siguiente tabla se relacionan algunos aspectos que fueron decisivos para aprobar la creación de una herramienta informática en el Centro GEYSED que planifique y controle el presupuesto.

Parámetros	Presto	VERSAT-SARASOLA	Sistema de Control de Solicitudes al Presupuesto
Planifica presupuesto.	Sí	Sí	No
Controla presupuesto.	No	No	Sí
Realizar gráficos para realizar análisis.	No	No	No
Exportar a PDF.	No	Sí	No
Insertar elementos del gasto.	Sí	Sí	Sí
Eliminar elementos del gasto.	Sí	Sí	No
Realizar el presupuesto de gasto	Sí	Sí	Sí

Tabla 1.Herramientas existentes

Por todo lo antes mencionado el Centro de Geoinformática y Señales Digitales (GEYSED) decidió implementar un sistema que cumpliera con los requisitos mostrados en la tabla anterior. Esta herramienta debe lograr la ejecución del presupuesto de gasto en el centro y en cada uno de sus proyectos así como

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

los elementos mostrados en la tabla anterior. Además se considera que sería una solución confiable y concreta que cubra las necesidades del cliente. Para su correcta elaboración se estudiarán las metodologías existentes, el lenguaje de modelado y la herramienta CASE a utilizar, y otros elementos de gran importancia para su desarrollo.

1.3 Proceso de Desarrollo de Software

Un proceso del software es un conjunto de actividades y resultados asociados que producen un producto de software. Estas actividades son llevadas a cabo por los ingenieros de software. (Sommerville, 2005) Un proceso de desarrollo del software debería ser capaz de evolucionar durante muchos años. Durante esta evolución debería limitar su alcance, a las realidades que permitan las tecnologías, herramientas, personas y patrones de organización. (Jacobson, y otros, 2000)

El proceso de desarrollo requiere de un conjunto de conceptos, una metodología y un lenguaje propio. Su objetivo es plantear las actividades que proporcionen el desarrollo de un sistema informático, manteniendo la calidad y fijando un plazo de tiempo y un costo predecible.

Metodología de Desarrollo de Software

Una metodología de desarrollo es un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar nuevos software. Las metodologías de desarrollo surgen para guiar a las personas en el desarrollo del software, de manera que sepan qué hacer en cada momento y cómo alcanzar un producto de alta calidad. En la actualidad existen diferentes metodologías, todas con el objetivo de elevar la calidad del software, y se agrupan en dos grupos, las tradicionales y las ágiles. (Carvajal, 2008)

Las metodologías tradicionales ponen mucho énfasis sobre el plan de proyecto, en tenerlo todo bien especificado antes de comenzar, seguir fielmente el camino planificado y documentar exhaustivamente todo lo realizado. Las metodologías ágiles, por el contrario, ponen énfasis en entregar buen código al cliente. En obtener resultados que satisfagan al cliente, adaptándose a sus siempre cambiantes necesidades. Aunque esto no quiere decir que las metodologías ágiles abandonen totalmente herramientas y procedimientos ya establecidos, ni que renuncien a la disciplina. Tan solo significa que los

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

emplean de forma flexible. Lo principal es obtener buenos resultados, no el hacer trabajos perfectos. (Carvajal, 2008)

Principales ideas de las metodologías ágiles: (Carvajal, 2008)

- El cliente está en todo momento colaborando en el proyecto.
- Es más importante la capacidad de respuesta ante un cambio realizado que el seguimiento estricto de un plan.
- Se hace más importante crear un producto de software que funcione que escribir mucha documentación.
- Se da mayor valor al desarrollo incremental del software con iteraciones cortas.

XP (eXtreme Programming)

Es una metodología ágil que se centra básicamente en satisfacer las necesidades del cliente para lograr un producto de buena calidad en poco tiempo. Busca simplificar el desarrollo del software y reducir el costo del proyecto. Se requiere un grupo pequeño de trabajo (entre 2 y 15 personas) que puede ir aumentando conforme sea necesario. Se realizan pruebas todo el tiempo y las iteraciones suelen ser más cortas que en otros métodos. (Extreme Programming, 2009)

Promueve el trabajo en equipo al potenciar las relaciones interpersonales como clave para el éxito en el desarrollo del software. Presenta roles tales como: (Extreme Programming, 2009)

- Programador: es el responsable de construir el sistema y de tomar decisiones técnicas. En XP los programadores diseñan, programan y realizan las pruebas.
- Cliente: determina qué construir y cuándo, establece las pruebas funcionales.
- Encargado de pruebas (Tester): ayuda al cliente con las pruebas funcionales y se encarga de ejecutarlas con regularidad.
- Encargado de seguimiento (Tracker): es quien realiza el seguimiento del proceso de cada iteración y conserva datos históricos.
- Entrenador (Coach): es el responsable del proceso global y a medida que el equipo madura tiende a estar en un segundo plano.

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

Para cumplir el objetivo de este trabajo de diploma la metodología seleccionada es XP. Primero porque es una metodología ágil y por consiguiente optimiza el tiempo y evita la demora en la entrega del producto. También porque se toma en cuenta la participación del cliente desde el primer momento en que comienza el proyecto y hasta el final del mismo. La selección de esta metodología se basa además en que ella responde a las necesidades del cliente y los cambios que éstos puedan hacer a su pedido. Por último se está en presencia de un equipo de trabajo pequeño y el cliente necesita el producto lo antes posible.

Lenguaje Unificado de Modelado (UML)

El Lenguaje Unificado de Modelado es un lenguaje estándar que puede utilizarse para visualizar, especificar, construir y documentar los artefactos de un sistema de software. Actualmente es el lenguaje de modelado más conocido y utilizado, pues permite la modelación de sistemas con tecnología orientada a objetos. Ofrece una amplia variedad de diagramas, lo que hace que este lenguaje sea el predilecto de muchos. UML no es un lenguaje de programación visual, pero sus modelos pueden conectarse de forma directa a una gran variedad de lenguajes de programación. (Booch, 2007)

Otras de las propiedades de UML como lenguaje de modelado son: (Booch, 2007)

- Reemplaza a decenas de notaciones empleadas con otros lenguajes.
- Modela estructuras complejas.

UML no está limitado al modelado de software. Es lo suficientemente expresivo para modelar sistemas que no son software, como flujos de trabajo en el sistema jurídico, estructura y comportamiento de un sistema de vigilancia médica de un enfermo, y el diseño de hardware. (Booch, 2007)

1.4 Herramientas para el desarrollo del sistema

Herramienta CASE: Visual Paradigm 8.0

Las herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Ordenador) son aplicaciones informáticas que presentan como objetivo apoyar una o más fases del ciclo de vida del desarrollo de sistemas. Están destinadas a aumentar la productividad en el desarrollo de software, mediante la reducción del coste de las mismas en términos de tiempo y dinero. Estas herramientas sirven de gran ayuda en el desarrollo del ciclo de vida del software, en tareas como el

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores, entre otras. (software, 2007)

El desarrollo de la presente propuesta de solución incluirá el uso de la herramienta Visual Paradigm en su versión 8.0. Esta es una poderosa herramienta CASE y la más sencilla para usar el lenguaje de modelado UML, proporcionando a los desarrolladores una plataforma que les permite diseñar un producto con calidad. Además está diseñada para soportar una gama de lenguajes de programación, entre ellos C++. (software, 2007) Visual Paradigm 8.0 permite construir diagramas de diversos tipos, utilizándose durante el desarrollo del presente trabajo en la modelación de la base de datos. (Extensión de Visual Paradigm for UML para el desarrollo dirigido por modelos de aplicaciones de gestión de información, 2012)

Lenguaje de programación: C++

La programación orientada a objetos es un modelo de programación, con su teoría y su metodología. Es otra forma de descomponer problemas, este nuevo método se llama descomposición en objetos, se fija no en lo que hay que hacer en el problema, sino en cuál es el escenario real del mismo, y luego se intenta simular ese escenario en el programa. (Izquierdo, 2007)

Un lenguaje de programación es aquel elemento dentro de la informática que permite crear programas, mediante un conjunto de instrucciones, operadores y reglas de sintaxis que pone a disposición del programador, para que este pueda comunicarse con los dispositivos hardware y software existentes. (Arias, 2008). En los últimos tiempos han surgido variedades de estos lenguajes, cada vez con más facilidades para el desarrollador.

Entre los lenguajes más usados se encuentra C++, el cual es un lenguaje imperativo orientado a objetos derivado del C. C++ es un lenguaje de programación muy extenso y complejo, pero ha experimentado a pesar de todo un gran éxito desde su creación. Muchos sistemas operativos, compiladores e intérpretes han sido escritos en C++. (Jiménez, 2011)

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

Se escoge este lenguaje de programación para el desarrollo de este trabajo porque es muy eficiente, siendo uno de los lenguajes más rápidos, permitiendo que la aplicación a implementar trabaje con rapidez. Además es el lenguaje nativo del entorno de desarrollo QT Creator, que se analiza a continuación.

Ambiente de Desarrollo Integrado (IDE): QT Creator 5.0.1

QT Creator es el entorno de desarrollo seleccionado para desarrollar la presente investigación, pues representa para programadores con mínima experiencia un buen punto de partida para programar en C++, además de poseer un IDE bastante amigable. QT Creator es un poderoso entorno de desarrollo integrado para C++, el cual incluye herramientas de diseño de Interfaces y debugging de C++. (Programming, 2012) Otras de las características que estimulan la utilización de este IDE es que su uso es gratuito, permite que el programador pueda ver y modificar el código fuente, y además utilizarlo sin restricciones de licencia.

Sistema Gestor de BD (SGBD): PostgreSQL 9.1

Un Sistema Gestor de Base de Datos (SGBD) se define como el conjunto de programas que administran y gestionan la información contenida en una base de datos. (Àlvarez, 2007) Tiene como objetivo servir de interfaz entre la base de datos, el usuario y las aplicaciones.

Presenta características tales como: (Àlvarez, 2007)

- Independencia: se puede modificar el esquema (físico o lógico) de la base de datos sin necesidad de realizar cambios en las aplicaciones que se sirven de ella.
- Seguridad: los SGBD deben garantizar que la información almacenada en la base de datos se encuentre segura ante cualquier amenaza.
- Integridad: garantizar la protección de los datos ante cualquier urgencia, dígame fallos de hardware o cualquier otra circunstancia capaz de corromper la información almacenada.

El Sistema Gestor de Base de Datos Relacionales Orientadas a Objetos conocido como PostgreSQL cuenta ya con más de 15 años de desarrollo activo. Se selecciona este SGBD para el desarrollo del presente trabajo pues PostgreSQL se ejecuta en la mayoría de los sistemas operativos más utilizados en el mundo. Además, es el gestor de base de datos de código abierto más avanzado hoy en día, soporta

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

casi toda la sintaxis SQL y cuenta también con un amplio conjunto de enlaces con lenguajes de programación, incluyendo C++. (Pecos, 2012) Además PostgreSQL está enfocado en mantener la integridad de los datos y tiene un planificador de consultas muy sofisticado que es capaz de unir grandes cantidades de tablas eficientemente. (PostgreSQL, 2013)

Capítulo 1: Fundamentación Teórica sobre los Procesos de Planificación y Control del Presupuesto

1.5 Conclusiones del Capítulo

Como resultado del estudio realizado concerniente al tema de presupuestos, sus objetivos y sus etapas de planificación y control, se logró una mayor comprensión del tema en cuestión y conocer con más precisión las tareas que el sistema debe desarrollar. Durante el análisis de algunas soluciones informáticas existentes en el mundo, en Cuba y en la UCI, asociadas al contenido de este trabajo, se aclara que no cumplen con los requisitos planteados por el asesor económico del Centro GEYSED. Con la selección de la metodología XP como guía para el desarrollo del trabajo, queda demostrado que se dará respuesta a las necesidades del cliente en el menor tiempo posible. Otra de las conclusiones obtenidas fue que, el uso de las tecnologías mencionadas favorece la rapidez de la aplicación y la seguridad de sus datos.

Capítulo 2. Características del Sistema para la Planificación y el Control del Presupuesto

Capítulo 2. Características del Sistema para la Planificación y el Control del Presupuesto

Introducción

En este capítulo se describe de forma general el sistema a desarrollar, así como los requisitos funcionales y no funcionales del mismo, las personas que se relacionan con el sistema y cómo interactúan con la aplicación, y se hace una descripción de las características de la arquitectura en capas, la cual se utilizará en el software. También se da inicio a la fase de Planificación de la metodología XP.

2.1 Descripción del Sistema

El sistema que se desea desarrollar da cumplimiento a la necesidad del asesor económico del Centro GEYSED de planificar y controlar el presupuesto haciendo uso de una herramienta informática que genere los reportes asociados a estas tareas de manera automática. Teniendo en cuenta lo planteado en la situación problemática, se desea darle solución a dicho problema implementando un software capaz de realizar el presupuesto de gasto. El mismo podrá ser utilizado por el asesor económico para realizar el control de los gastos en los que ha incurrido el centro a modo general y cada uno de sus proyectos de manera específica, así como el análisis de los mismos a través de gráficos.

2.2 Personas que interactúan con el sistema

Los clientes que interactúan con este sistema son aquellos que obtienen un resultado del mismo.

Personal que interactúa con el sistema	Justificación
Asesor Económico de GEYSED	El cliente interactúa directamente con el sistema en la elaboración del presupuesto.

Tabla 2. Personal que interactúa con el sistema

Capítulo 2. Características del Sistema para la Planificación y el Control del Presupuesto

2.3 Requisitos del sistema

Los requisitos funcionales son capacidades o condiciones que el sistema debe cumplir y los requisitos no funcionales son propiedades o cualidades que el producto debe tener, son las características que hacen al producto atractivo, usable, rápido, y/o confiable. (Pressman, 2012) De acuerdo a estas definiciones, se mencionan a continuación los requisitos que debe cumplir la herramienta que se desea implementar en el Centro GEYSED.

Requisitos Funcionales

- R1. Autenticar usuario:** Esta funcionalidad permite que el usuario se autentique, con su respectivo usuario y contraseña.
- R2. Cambiar contraseña:** Esta funcionalidad permite al usuario cambiar su contraseña.
- R3. Insertar nuevo elemento de gasto:** El sistema permite la inserción de un nuevo elemento de gasto.
- R4. Eliminar elemento de gasto:** El sistema permite que se elimine el elemento de gasto seleccionado.
- R5. Modificar elemento de gasto:** El sistema permite al usuario seleccionar un elemento de gasto y modificarlo.
- R6. Generar reporte del presupuesto de gasto:** Permite generar el reporte del presupuesto de gasto a partir de la información adquirida en cada elemento de gasto del sistema.
- R7. Exportar reporte a PDF:** El sistema permite exportar los reportes en formato PDF.
- R8. Generar reporte del control de los gastos reales del centro:** Permite generar el reporte del control de los gastos reales en MN, CUC y Moneda Total del centro a partir de la información adquirida en cada elemento de gasto del sistema.
- R9. Generar reporte del control de los gastos reales de los proyectos:** Permite generar el reporte del control de los gastos reales en MN, CUC y Moneda Total de los proyectos a partir de la información adquirida en cada elemento de gasto del sistema.
- R10. Crear gráfico para realizar análisis:** Esta funcionalidad posibilita que la información seleccionada por el usuario sea guardada en forma de gráfico.
- R11. Guardar gráfico:** Esta funcionalidad permite que el gráfico creado sea luego guardado en la dirección indicada por el usuario.
- R12. Cargar gráfico:** El sistema permite que, una vez guardado un gráfico, este pueda ser cargado accediendo a la dirección en donde se encuentra.

Capítulo 2. Características del Sistema para la Planificación y el Control del Presupuesto

R13. Buscar elementos de gasto: El sistema permite buscar un elemento de gasto y lo muestra.

Requisitos no Funcionales

Diseño e Implementación:

- Se utilizará el lenguaje de programación: C++.
- Se utilizará Visual Paradigm como Herramienta CASE para modelar la base de datos y el Entorno de Desarrollo Integrado QT Creator.

Usabilidad:

- El sistema contará con una interfaz sencilla para que el usuario pueda utilizarla sin necesidad de poseer las habilidades de un informático.

Hardware:

- El sistema se ejecutará en una computadora con una memoria RAM de 512 Mb o más.

Seguridad:

- La aplicación permitirá que el usuario se autentique.
- Para proteger los datos se hará uso de la seguridad que proporciona el PostgreSQL.

Software:

- El Gestor de Base de Datos a utilizar es PostgreSQL.

2.4 Fase de Planificación

La metodología de desarrollo XP comienza con su fase de planificación. Durante esta etapa se realiza el proceso de identificación y desarrollo de las historias de usuario, así como la familiarización del equipo de trabajo con las tecnologías y herramientas seleccionadas para el desarrollo del software.

Historia de Usuarios (HU)

Representan una breve descripción del comportamiento del sistema, emplean terminología del cliente sin lenguaje técnico. Las Historias de Usuario tienen tres aspectos:

- Tarjeta: En ella se almacena suficiente información para identificar y detallar la historia.
- Conversación: Cliente y programadores discuten la historia para ampliar los detalles.
- Prueba de Aceptación: Permite confirmar que la historia ha sido implementada correctamente.

Capítulo 2. Características del Sistema para la Planificación y el Control del Presupuesto

A continuación se muestra la tarjeta Historia de Usuario, en la tabla 3.

Historia de Usuario	
Número:	Usuario:
Nombre Historia:	
Prioridad en el negocio: (Alta/Media/Baja)	Riesgo en desarrollo: (Alto/Medio/Bajo)
Puntos estimados: (semanas)	Iteración asignada:
Descripción:	
Observaciones:	

Tabla 3. Historia de Usuario.

Campos de la Tarjeta Historia de Usuarios

Número: Índice de la Historia de Usuario. Es un número único que se le asigna a cada HU con el fin de lograr una mejor organización de estas.

Nombre historia: Nombre de la HU.

Descripción: Se describe el requisito que da origen a la HU. Debe ser corta, precisa y dejar claro qué es lo que se desea hacer.

Observaciones: Aspectos que se deben tener en cuenta para implementar la HU. Precondiciones o poscondiciones que se deben tener presentes al implementar la HU.

A continuación muestran algunas de las historias de usuario del presente trabajo, las restantes se encuentran descritas en el Anexo 3.

Historia de Usuario	
Número:1	Usuario: Asesor económico
Nombre Historia: Insertar nuevo elemento de gasto	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio

Capítulo 2. Características del Sistema para la Planificación y el Control del Presupuesto

Puntos estimados: 2 semanas	Iteración asignada: 1
Descripción: El sistema inserta un nuevo elemento de gasto, al usuario seleccionar la opción Insertar fila. Luego de esto se puede ingresar la información correspondiente a dicho elemento de gasto.	
Observaciones: El usuario debe estar autenticado.	

Tabla 4. Representación de la Historia de Usuario No.1

Historia de Usuario	
Número: 2	Usuario: Asesor económico
Nombre Historia: Generar reporte del presupuesto de gasto	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1 semana	Iteración asignada: 1
Descripción: Luego de que el usuario inserte los elementos de gasto, el sistema automáticamente genera el reporte del presupuesto de gasto, valiéndose de fórmulas preestablecidas. La aplicación también permite que el usuario pueda realizar cambios en estas fórmulas, a través de la opción Calcular.	
Observaciones: El usuario debe estar autenticado. Deben haberse insertado todos los elementos de gasto.	

Tabla 5. Representación de la Historia de Usuario No.2

Historia de Usuario	
Número: 3	Usuario: Asesor económico
Nombre Historia: Generar reporte del control de los gastos reales del centro	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1 semana	Iteración asignada: 1
Descripción: Luego de que el usuario inserte en el sistema los elementos de gasto, el sistema automáticamente genera el reporte del control de los gastos reales en MN, CUC y Moneda Total del	

Capítulo 2. Características del Sistema para la Planificación y el Control del Presupuesto

centro, valiéndose de fórmulas preestablecidas. La aplicación también permite que el usuario pueda realizar cambios en estas fórmulas a través de la opción Calcular.
Observaciones: El usuario debe estar autenticado. Deben haberse insertado todos los elementos de gasto.

Tabla 6. Representación de la Historia de Usuario No.3

Historia de Usuario	
Número: 4	Usuario: Asesor económico
Nombre Historia: Generar reporte del control de los gastos reales de los proyectos	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1 semana	Iteración asignada: 1
Descripción: Luego de que el usuario inserte en el sistema los elementos de gasto, el sistema automáticamente genera el reporte del control de los gastos reales en MN, CUC y Moneda Total de los proyectos, valiéndose de fórmulas preestablecidas. La aplicación también permite que el usuario pueda realizar cambios en estas fórmulas a través de la opción Calcular.	
Observaciones: El usuario debe estar autenticado. Deben haberse insertado todos los elementos de gasto.	

Tabla 7. Representación de la Historia de Usuario No.4

2.5 Arquitectura Candidata

Uno de los pasos fundamentales en el proceso de desarrollo de software es ciertamente la definición de la arquitectura del software, la cual constituye el diseño de más alto nivel de la organización de un sistema, programa o aplicación. Es de gran importancia ya que la manera en que se estructura un sistema tiene un impacto directo sobre la capacidad de este para satisfacer lo que se conoce como los atributos de calidad del sistema. Ejemplos de atributos de calidad son el desempeño, que tiene que ver con el tiempo de respuesta del sistema a las peticiones que se le hacen; la usabilidad, que tiene que ver con qué tan sencillo les resulta a los usuarios realizar operaciones con el sistema.(Cervantes, 2010)

Capítulo 2. Características del Sistema para la Planificación y el Control del Presupuesto

Estas arquitecturas están definidas muchas veces por el tipo de tecnología a la cual se enfrenta un programador o grupo de programadores, por lo cual algunos tipos de arquitectura son más recomendables que otras para ciertos tipos de tecnologías. Para el desarrollo de esta herramienta se escogió la arquitectura en tres capas, que se describe a continuación.

Arquitectura en tres capas

Para resolver los problemas derivados de utilizar aplicaciones compactas, se ha generalizado la división de las aplicaciones en capas que normalmente serán tres. Se selecciona la arquitectura en tres capas para el desarrollo de la presente investigación porque, una capa servirá para el acceso a los datos, una capa para centralizar la lógica de negocio y por ultimo una capa de presentación para facilitar al usuario el uso del sistema. La separación de roles en tres capas hace más fácil reemplazar o modificar una capa sin afectar las restantes. (Sánchez, 2004)

Capa de Presentación: Es la interfaz con el usuario, se limita a gestionar todos aquellos aspectos relacionados con la lógica de presentación de la aplicación. Por ejemplo, comprobación de datos de entrada y formatos de salida. (Systems, 2006)

Capa Lógica de Negocio: Proporciona la funcionalidad y contiene la lógica de negocio (aplicación). Constituye el puente entre la primera y la tercera capa, o sea recibe la entrada de la capa anterior, interactúa con los servicios de datos para ejecutar las operaciones y envía el resultado procesado a la capa de presentación. (Systems, 2006)

Capa de Acceso a Datos: Incluye el acceso a la Base de Datos que contiene las entidades que se manejan en el negocio y el acceso a los datos almacenados. (Systems, 2006)

Capítulo 2. Características del Sistema para la Planificación y el Control del Presupuesto

2.6 Conclusiones del Capítulo

Con el levantamiento de los requisitos funcionales y no funcionales se logró analizar detalladamente las condiciones que el sistema debe cumplir y las cualidades que debe tener. La elaboración de las Historias de Usuario permitió una mayor comprensión en el equipo de desarrollo, de las necesidades del cliente, perfeccionando las características que debe cumplir el software y cumpliendo el objetivo trazado desde el inicio del trabajo. La selección de la arquitectura en tres capas como estructura de la aplicación, redujo la dependencia entre las capas y garantizó que el mantenimiento y el soporte de la misma fuese más eficiente, producto a la separación de roles.

Capítulo 3. Planificación y Diseño del Sistema para la Planificación y el Control del Presupuesto

Capítulo 3. Planificación y Diseño del Sistema para la Planificación y el Control del Presupuesto

Introducción

En el capítulo anterior se inicia la fase de planificación propuesta por la metodología XP, con la confección de las historias de usuario. El presente capítulo da continuidad a esta fase y realiza una descripción referente a la fase de diseño.

3.1 Estimación de esfuerzos por Historia de Usuario

Para realizar la estimación del esfuerzo por historia de usuario se hace necesario tener en cuenta que estas deben ser programadas en un tiempo entre una y tres semanas. De lo contrario, si es superior a tres semanas, se divide en dos o más historias. Si es menor de una semana, se combina con otra historia.

Se realizó una estimación para cada una de las historias de usuario identificadas, obteniendo los resultados que se aprecian en la tabla 8. Además, se considera que algunas historias de usuario se realizan en un período inferior a una semana, por lo que se decide lo siguiente:

- Las HU Autenticar usuario y Cambiar contraseña se combinan con la HU Insertar nuevo elemento de gasto.
- La HU Exportar reporte a PDF se combina con la HU Generar reporte del presupuesto de gasto.
- Las HU Guardar gráfico y Cargar gráfico se combinan con la HU Crear gráfico para realizar análisis.

Historia de Usuario	Puntos de Estimación
Autenticar usuario	1/2
Cambiar contraseña	1/2
Insertar nuevo elemento de gasto	2
Eliminar elemento de gasto	1
Modificar elemento de gasto	1
Generar reporte del presupuesto de gasto	1
Exportar reporte a PDF	1/2

Capítulo 3. Planificación y Diseño del Sistema para la Planificación y el Control del Presupuesto

Generar reporte del control de los gastos reales del centro	1
Generar reporte del control de los gastos reales de los proyectos	1
Crear gráfico para realizar análisis	1
Guardar gráfico	1/2
Cargar gráfico	1/2
Buscar elementos de gasto	1

Tabla 8. Estimación de esfuerzo por Historia de Usuario

3.2 Plan de Iteraciones

A continuación se realiza el plan de iteraciones, conocido también como planificación de la etapa de implementación del sistema. Consiste en seleccionar las historias de usuario que se implementan en cada iteración y su respectivo tiempo de duración. Se decide realizar el sistema en dos iteraciones.

Iteración 1

En esta iteración se implementan las HU que tienen alta prioridad. Al finalizar se contará con las funcionalidades descritas en las HU 1, 2, 3, 4, 5, 6, 7, 8 y 9: Autenticar usuario, Cambiar contraseña, Insertar nuevo elemento de gasto, Eliminar elemento de gasto, Modificar elemento de gasto, Generar reporte del presupuesto de gasto, Exportar reporte a PDF, Generar reporte del control de los gastos reales del centro y Generar reporte del control de los gastos reales de los proyectos.

Iteración 2

En esta iteración se implementan las funcionalidades restantes, las cuales tienen prioridad media. Al terminar se tendrán implementadas las historias de usuario 10, 11, 12 y 13: Creación de gráficos para realizar análisis, Guardar gráfico, Cargar gráfico y Buscar elementos de gasto.

Plan de duración de las iteraciones

Iteraciones	Historias de Usuario en orden de implementación	Duración de las iteraciones (semanas)
Iteración 1	1 Autenticar usuario	

Capítulo 3. Planificación y Diseño del Sistema para la Planificación y el Control del Presupuesto

	<ul style="list-style-type: none"> 2 Cambiar contraseña 3 Insertar nuevo elemento de gasto 4 Eliminar elemento de gasto 5 Modificar elemento de gasto 6 Generar reporte del presupuesto de gasto 7 Exportar reporte a PDF 8 Generar reporte del control de los gastos reales del centro 9 Generar reporte del control de los gastos reales de los proyectos 	7
Iteración 2	<ul style="list-style-type: none"> 1 Creación de gráficos para realizar análisis 2 Guardar gráfico 3 Cargar gráfico 4 Buscar elementos de gasto 	2

Tabla 9. Plan de duración de las iteraciones

3.3 Plan de entrega

El plan de entrega es el compromiso del equipo de desarrollo con los clientes, creado a partir de las historias de usuario. Se establece qué HU se agrupan para conformar una entrega y el orden de las mismas. En este plan se decide realizar dos entregas, las cuales se reflejan en la tabla que se muestra a continuación.

Entregas	Historia de Usuarios
Entrega 1	<ul style="list-style-type: none"> Autenticar usuario Cambiar contraseña Insertar nuevo elemento de gasto Eliminar elemento de gasto Modificar elemento de gasto Generar reporte del presupuesto de gasto
Entrega 2	<ul style="list-style-type: none"> Exportar reporte a PDF Generar reporte del control de los gastos reales del centro

Capítulo 3. Planificación y Diseño del Sistema para la Planificación y el Control del Presupuesto

	<p>Generar reporte del control de los gastos reales de los proyectos</p> <p>Creación de gráficos para realizar análisis</p> <p>Guardar gráfico</p> <p>Cargar gráfico</p> <p>Buscar elementos de gasto</p>
--	---

Tabla 10. Plan de Entrega

Como resultado del plan de entrega se crea el plan de duración de cada una de ellas, el cual se muestra en la Tabla 11. Por tal razón se realizan varias versiones en las fechas que se estiman, hasta que se logre finalizar el producto.

Entregas	Iteración 1 4ta semana de Abril	Iteración 2 2da semana de Junio
1	Versión 0.1	
2		Versión 0.2

Tabla 11. Plan de duración de entrega

3.4 Fase de diseño

La fase de diseño establece que este debe ser revisado y mejorado de manera continua, mientras se añaden funcionalidades al sistema. (Molpeceres, 2003) En esta fase, XP se rige por una serie de recomendaciones, las cuales se muestran a continuación:

- Realizar diseños simples.
- Usar Tarjetas CRC (Cargo o clase, Responsabilidad y Colaboración) para las sesiones de diseño.
- No agregar funcionalidad extra.

Diseños Simples

La metodología XP se encuentra enfocada en conseguir diseños simples y sencillos, evitando posibles complicaciones. Es decir, se debe diseñar la solución más simple que pueda funcionar. (Molpeceres, 2003)

Capítulo 3. Planificación y Diseño del Sistema para la Planificación y el Control del Presupuesto

A continuación se muestran los principios que se han tenido en cuenta al realizar el diseño de la interfaz de usuario de la herramienta que permita la planificación y control del presupuesto en el Centro GEYSED:

- La información mostrada se distribuye correctamente en el software.
- Las opciones generales del sistema se mantienen en el mismo lugar para evitar la desorientación del asesor económico.
- Los colores que se utilizan en el software brindan una vista clara y agradable.

Teniendo en cuenta lo antes expuesto se muestra la interfaz de usuario en los Anexos, en el Anexo 1: Imagen de la Interfaz del Sistema.

Tarjetas CRC (Cargo o clase, Responsabilidad y Colaboración)

Las Tarjetas CRC persiguen como objetivo, facilitar la comunicación entre el cliente y los desarrolladores y además documentar los resultados. (Molpeceres, 2003) En el caso del software que se desea desarrollar, se muestran a continuación las Tarjetas CRC realizadas para cada una de las clases de la aplicación.

Tarjeta CRC de la Clase Principal

Responsabilidades	Clases relacionadas
Contiene los métodos que permiten el funcionamiento de la aplicación en general.	Conexión, Fórmula, Graficar, Exportar, Buscar y Cambiar_Pass.

Tabla 12. Clase Principal

Tarjeta CRC de la Clase Conexión

Responsabilidades	Clases relacionadas
En esta clase se declaran los atributos y métodos que permiten la conexión a la Base de Datos.	Principal.

Tabla 13. Clase Conexión

Tarjeta CRC de la Clase Fórmula

Responsabilidades	Clases relacionadas

Capítulo 3. Planificación y Diseño del Sistema para la Planificación y el Control del Presupuesto

En esta clase se declaran los atributos y métodos que permiten, de ser necesario, crear nuevas fórmulas y eliminarlas.	Principal.
--	------------

Tabla 14. Clase Fórmula

Tarjeta CRC de la Clase Graficar

Responsabilidades	Clases relacionadas
Esta clase permite la creación de gráficos para realizar análisis.	Principal.

Tabla 15. Clase Graficar

Tarjeta CRC de la Clase Exportar

Responsabilidades	Clases relacionadas
Esta clase permite la acción de exportar a PDF.	Principal.

Tabla 16. Clase Exportar

Tarjeta CRC de la Clase Buscar

Responsabilidades	Clases relacionadas
En esta clase se realizan las funcionalidades que permiten buscar un elemento de gasto en el sistema.	Principal.

Tabla 17. Clase Buscar

Tarjeta CRC de la Clase Cambiar_Pass

Responsabilidades	Clases relacionadas
En esta clase se realizan las funcionalidades que permiten cambiar la contraseña de acceso al sistema.	Principal.

Tabla 18. Clase Cambiar_Pass

Capítulo 3. Planificación y Diseño del Sistema para la Planificación y el Control del Presupuesto

Funcionalidad Extra

La metodología XP recomienda no añadir funcionalidad extra al programa aunque se piense que en un futuro será utilizada. Solo el 10% de la misma es utilizada, lo que implica que el desarrollo de funcionalidad extra es un desperdicio de tiempo y recursos. (Molpeceres, 2003)

A la hora de realizar el software se tuvo en cuenta este principio, pues no se realizó una historia de usuario en una iteración que no fuera la correspondiente. Además, durante el desarrollo de una tarea, no comenzó la implementación de otra historia de usuario hasta tanto no se culminase la anterior.

Capítulo 3. Planificación y Diseño del Sistema para la Planificación y el Control del Presupuesto

3.5 Conclusiones del Capítulo

Con el cierre de la fase de Planificación se obtiene el siguiente resultado, el tiempo de duración estimado para cada iteración y el plan de entrega del producto, agilizan el desarrollo del sistema permitiendo la entrega final en la fecha fijada. Mediante la elaboración de las Tarjetas CRC, durante la fase de Diseño, se garantizó que el cliente estuviera al tanto junto los desarrolladores, del progreso de la solución, lo que facilitó la comunicación entre ambos. Al conseguir un diseño simple de la interfaz, se garantiza el trabajo del cliente sin necesidad de que este posea conocimientos de informática.

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

Introducción

En este capítulo se describen los artefactos generados durante las fases de implementación y prueba, que consiste en el desarrollo del sistema y las pruebas realizadas al mismo. Cada historia de usuario a implementar se divide en diferentes tareas de programación y consecutivamente se elaboran los casos de prueba.

4.1 Estándares de Codificación

Los estilos o estándares de codificación son pautas de programación que están enfocadas en la estructura y apariencia física del programa, facilitando la comprensión del código por terceras personas. Su uso permite reflejar un estilo armonioso, como si todo el código hubiese sido escrito por un solo programador. Si se aplica de manera continua un estándar de codificación y posteriormente se realizan revisiones del código, aumentan las posibilidades de que el proyecto de software se convierta en un sistema fácil de comprender y de mantener. (Hristov, 2007) A continuación se define el estilo de codificación utilizado en la aplicación:

- Para nombrar las variables se seguirá la regla de escribirlas con letras minúsculas y se usará el separador línea abajo “_” entre las palabras. Ejemplo: pass_old.
- En el caso de las clases, sus nombres se escriben con letras minúsculas y se usará el separador línea abajo “_” entre las palabras. Ejemplo: cambiar_pass.
- Para el nombre de los métodos, las palabras continuas deben comenzar cada una con mayúscula. Ejemplo: CrearTablas ().
- Para comentar el código se utiliza el carácter “//” al principio de una línea.
- Las llaves se usarán poniendo la llave inicial en una línea para ella sola, y en su respectiva columna la llave final, también en una línea.

4.2 Tareas de programación

Las historias de usuario son descompuestas en tareas de programación. Estas son actividades que los programadores conocen que el sistema debe hacer. Deben ser estimables y poder ser implementadas

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

entre uno y tres días. Cada tarea de programación será comprobada mediante los casos de prueba y no resulta necesaria su comprensión por parte del cliente. (Beck, 1999) A continuación se muestra la tarjeta Tarea de programación, en la tabla 19.

Tarea de programación	
Número tarea:	Número historia:
Nombre tarea:	
Tipo de tarea: Desarrollo / Corrección / Mejora / Otra(especificar)	Puntos estimados:
Descripción:	

Tabla 19. Descripción de los campos de una Tarea de programación

Campos de la tarjeta Tarea de programación

Número historia: Número de la historia de usuario a la que corresponde la tarea de programación.

Número tarea: Número único asignado a la tarea de programación que pertenece a una HU determinada.

Nombre tarea: Nombre de la tarea de programación. Debe describir lo que se realizará y no ser muy extenso.

Tipo de tarea: Informa el tipo de tarea a realizar. Las tareas pueden ser:

- Desarrollo: tarea que se realizará por primera vez.
- Corrección: tarea que se realiza a partir de una anterior que no se efectuó correctamente, es decir, el resultado de los casos de prueba correspondientes no fue satisfactorio.
- Mejora: tarea a la cual se le incorporan nuevos requerimientos, por lo tanto deberá ser modificada y enfrentar nuevamente los casos de prueba.
- Otra: tarea que no corresponde con ninguna de las anteriormente descritas.

Puntos estimados: tiempo estimado en días que durará la implementación de la tarea de programación.

Descripción: se describe de manera breve y precisa lo que se desea realizar.

Historia de Usuario	Tareas de programación
Autenticar usuario.	1. Validar los datos introducidos: usuario y contraseña.

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

Cambiar Contraseña	<ol style="list-style-type: none"> 1. Validar que la vieja contraseña introducida sea correcta. 2. Sustituir la vieja contraseña por la nueva contraseña introducida.
Insertar nuevo elemento de gasto	<ol style="list-style-type: none"> 1. Opción para agregar elementos de gasto en el sistema.
Eliminar elemento de gasto	<ol style="list-style-type: none"> 1. Opción para eliminar el elemento de gasto seleccionado.
Modificar elemento de gasto	<ol style="list-style-type: none"> 1. Opción para modificar el elemento de gasto seleccionado.
Generar reporte del presupuesto de gasto	<ol style="list-style-type: none"> 1. Realizar el cálculo de cada elemento de gasto mediante la utilización de fórmulas. 2. Actualizar el reporte si sucede algún cambio en un elemento de gasto.
Exportar reporte a PDF	<ol style="list-style-type: none"> 1. Opción para exportar a PDF.
Generar reporte del control de los gastos reales del centro	<ol style="list-style-type: none"> 1. Realizar el cálculo de cada uno de los gastos reales del centro mediante el uso de fórmulas. 2. Actualizar el reporte en caso de cambio en algún elemento de gasto.
Generar reporte del control de los gastos reales de los proyectos	<ol style="list-style-type: none"> 1. Realizar el cálculo de cada uno de los gastos reales de los proyectos mediante el uso de fórmulas. 2. Actualizar el reporte en caso de cambio en algún elemento de gasto.
Crear gráficos para realizar análisis	<ol style="list-style-type: none"> 1. Opción para crear el gráfico.
Guardar gráfico	<ol style="list-style-type: none"> 1. Opción para guardar el gráfico.
Cargar gráfico	<ol style="list-style-type: none"> 1. Opción para cargar el gráfico.
Buscar elementos de gasto	<ol style="list-style-type: none"> 1. Opción para buscar el elemento de gasto.

Tabla 20. Tareas de programación por cada Historia de Usuario

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

Descripción de las Tareas de programación por cada Historia de Usuario

Iteración 1

Durante la primera iteración se desarrollaron nueve tareas de programación, las cuales se encuentran descritas en el Anexo 4, a continuación muestran solamente dos de ellas.

Tarea de programación	
Número tarea: HU6_1	Número historia: 1
Nombre tarea: Realizar el cálculo de cada elemento de gasto mediante la utilización de fórmulas.	
Tipo de tarea: Desarrollo	Puntos estimados: 2 días
Descripción: El sistema realiza el cálculo de los elementos de gasto mediante fórmulas asignadas y presenta la opción Calcular que permite modificar dichas fórmulas o agregar otras nuevas.	

Tabla 21. Descripción de la Tarea de programación HU6_1

Tarea de programación	
Número tarea: HU6_2	Número historia: 2
Nombre tarea: Actualizar el reporte si sucede algún cambio en un elemento de gasto.	
Tipo de tarea: Desarrollo	Puntos estimados: 2 días
Descripción: Si el usuario realiza un cambio en algún elemento de gasto, el reporte automáticamente queda actualizado conforme al cambio realizado.	

Tabla 22. Descripción de la Tarea de programación HU6_2

Iteración 2

Durante la segunda iteración se desarrollaron cuatro tareas de programación, las cuales se encuentran descritas en el Anexo 4, a continuación se muestran dos de ellas.

Tarea de programación	
Número tarea: HU10_1	Número historia: 1
Nombre tarea: Opción para crear el gráfico.	

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

Tipo de tarea: Desarrollo	Puntos estimados: 3 días
Descripción: El usuario selecciona la información que necesita graficar en el reporte y da clic en la opción Graficar.	

Tabla 23. Descripción de la Tarea de programación HU10_1

Tarea de programación	
Número tarea: HU13_1	Número historia: 1
Nombre tarea: Opción para buscar el elemento de gasto	
Tipo de tarea: Desarrollo	Puntos estimados: 5 días
Descripción: El usuario selecciona la opción Buscar en el sistema y escribe el nombre del elemento de gasto a buscar.	

Tabla 24. Descripción de la Tarea de programación HU13_1

4.2 Patrones de Diseño

Uno de los pasos importantes a la hora de desarrollar un proyecto de software es identificar qué patrones pueden ser utilizados. Un patrón es una solución estándar para un problema de programación. (Larman, 2005)

Un patrón de diseño define un esquema de refinamiento de los subsistemas o componentes dentro de un sistema, o las relaciones entre estos. Este describe una estructura común y recurrente de componentes interrelacionados, que resuelve un problema general de diseño dentro de un contexto particular. Se definen dos grupos de patrones, los GRASP (Patrones generales de software para asignar responsabilidades) y los Gof (*Gang of Four*) (Larman, 2005). A continuación se describen los patrones de diseño aplicados durante el desarrollo del producto.

Patrones generales de software para asignar responsabilidades (GRASP)

Los patrones GRASP constituyen el fundamento de cómo se diseñará el sistema. (Larman, 2005) Durante el diseño del sistema se emplearon los siguientes patrones GRASP:

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

Experto: El principio de este patrón es asignar una responsabilidad al experto en información: la clase que cuenta con la información necesaria para cumplir la responsabilidad. Se utiliza más que cualquier otro al asignar responsabilidades y conserva el encapsulamiento ya que los objetos se valen de su propia información para hacer lo que se les pide. Esto soporta un bajo acoplamiento, lo que favorece al hecho de tener sistemas más robustos y de fácil mantenimiento. La aplicación de este patrón también trae como beneficio que el comportamiento se distribuya entre las clases que tienen la información requerida alentando con ello definiciones de clases más sencillas, más fáciles de mantener y comprender. (Larman, 2005) El uso de este patrón se evidencia en la definición de las clases de acuerdo con las funcionalidades que deben realizar a partir de la información que manejan. Específicamente las clases Buscar, Cambiar_Pass, Conexión, Exportar, Fórmula, Graficar y Principal.

Creador: Este patrón guía la asignación de responsabilidades relacionadas con la creación de objetos, actividad que está presente todo el tiempo en los sistemas orientados a objetos. El principal objetivo de este patrón es identificar la clase que debe llevar a cabo la creación de objetos de otra clase, teniendo en cuenta la conexión entre estos, pues la clase responsable de esta tarea debe agregar, contener, y registrar las instancias de estos objetos, así como utilizarlos. (Larman, 2005) Este patrón se utiliza en la clase Principal.

Bajo acoplamiento: La aplicación del bajo acoplamiento significa que una clase no depende de muchas otras, lo que trae como beneficios que al realizarse cambios en otras clases no se ocasionen cambios locales, y además que esta no sea difícil de entender cuando esté aislada y que sea fácil de reutilizar puesto que es independiente. (Larman, 2005) Este patrón está presente en la clase Principal.

Alta cohesión: La implementación de este patrón garantiza que las clases con responsabilidades estrechamente relacionadas no realicen un trabajo excesivo. Esta relación es fundamental para, mejorar la claridad y facilidad con que se entiende el diseño, simplificar el mantenimiento y mejoras en las funcionalidades, generar un bajo acoplamiento y contar con mayor capacidad de reutilización. (Larman, 2005) El uso de este patrón se hace evidente en la clase Principal.

Controlador: Identifica el encargado de atender los eventos del sistema. El controlador no debe ser saturado con la asignación de demasiadas responsabilidades, ha de delegar a otros objetos el trabajo a

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

realizarse en tanto coordina la actividad. (Larman, 2005) El uso de este patrón está presente en la clase Principal.

Patrones GoF

Los patrones GoF se dividen en tres categorías:

- Creacionales: Resuelven problemas relativos a la creación de objetos.
- Estructurales: Resuelven problemas relativos a la composición de objetos.
- Comportamiento: Resuelven problemas relativos a la interacción entre objetos. (Cáceres, 2009)

El patrón GoF empleado en esta solución es:

Observador: Es un patrón de diseño de software en el cual un objeto mantiene la lista de todos sus dependientes, llamados observadores, a los cuales notifica automáticamente cuando algún estado o evento cambia, usualmente llamando alguno de sus métodos. Los widgets que provee el IDE QT Creator poseen signals y slot predefinidos, y heredan de la clase QWidget, que a su vez hereda de la clase QObject. Se define una señal/signal como un aviso que un objeto puede emitir cuando le ocurre algo (un cambio de estado, también denominado evento), un slot es un método de un objeto que puede ser llamado cuando se genere una señal. Cualquier objeto en el que se desee implementar señales y slots debe heredar de la clase QObject, que es la base de todos los objetos del QT Creator. (Kohen, 2010)

4.3 Pruebas

XP propone la realización de dos tipos de pruebas al sistema, las pruebas unitarias y las pruebas de aceptación. Las pruebas de aceptación no son más que pruebas escritas desde la perspectiva del cliente y las pruebas unitarias están escritas desde la perspectiva del programador. El objetivo es que el cliente conozca cuando una HU está lista. (Beck, 1999)

Las pruebas unitarias son las encargadas de verificar el código, mientras que las pruebas de aceptación están destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida. Para cada historia de usuario se establecen las pruebas de aceptación. Los clientes son responsables de verificar que los resultados de estas pruebas sean correctos. Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. (Joskowicz, 2008)

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

Generalmente las pruebas al sistema son desarrolladas por los programadores para verificar que el sistema se comporta del modo esperado, por lo que pueden encajar dentro de la definición de pruebas unitarias que XP propone. Además tienen como objetivo verificar que el sistema cumple los requisitos establecidos por el usuario, por lo que también pueden encajar dentro de la categoría de pruebas de aceptación. Estas últimas son más importantes que las pruebas unitarias pues significan la satisfacción del cliente con el producto y el final de una iteración y el comienzo de la siguiente.

Una historia de usuario puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. El objetivo final de estas es garantizar que los requisitos hayan sido cumplidos y que el sistema sea aceptable. (Beck, 1999)

Pruebas Unitarias

Las pruebas unitarias se realizan de forma manual o automatizada, con el objetivo de demostrar que lo implementado es lo que se pensaba al principio. (Beck, 2000) Para la ejecución de esta prueba se utilizó la técnica del camino básico. Esta permite obtener una medida de la complejidad lógica de un diseño y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución. Los casos de prueba obtenidos del conjunto básico garantizan que durante la prueba se ejecute por lo menos una vez cada sentencia del programa. (Pressman, 2012)

La técnica del camino básico consiste en derivar casos de prueba a partir de un conjunto dado de caminos independientes. Los casos de prueba definen un conjunto de entradas de pruebas, ejecución de condiciones y resultados esperados. Un camino independiente es cualquier camino del programa que introduce una nueva condición. Para obtener el conjunto de caminos independientes, se construye el grafo de flujos (representa el flujo de control lógico) asociado al código y se calcula su complejidad ciclomática, que es una métrica del software que proporciona una medición cuantitativa de la complejidad lógica del programa. Luego, se identifican los nodos y según las instrucciones serán las aristas que los conecten.

El procedimiento es el siguiente:

- A partir del diseño o del código fuente, se dibuja el grafo asociado.

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

- Se calcula la complejidad ciclomática del grafo de flujo.
- Se determina el conjunto básico de caminos independientes.
- Se preparan los casos de prueba que obliguen a la ejecución de cada camino del conjunto básico.

La complejidad ciclomática se puede calcular de tres maneras:

- El número de regiones del grafo de flujo coincide con la complejidad ciclomática.
- La complejidad ciclomática, $V(G)$, de un grafo de flujo G se define como $V(G) = A - N + 2$, donde A es el número de aristas y N el número de nodos.
- La complejidad ciclomática, $V(G)$, de un grafo de flujo G también se define como $V(G) = P + 1$, donde P es el número de nodos predicado contenidos en el grafo de flujo G .

A continuación se muestra un ejemplo de la prueba realizada a la funcionalidad graficar de la clase Graficar.

Funcionalidad graficar

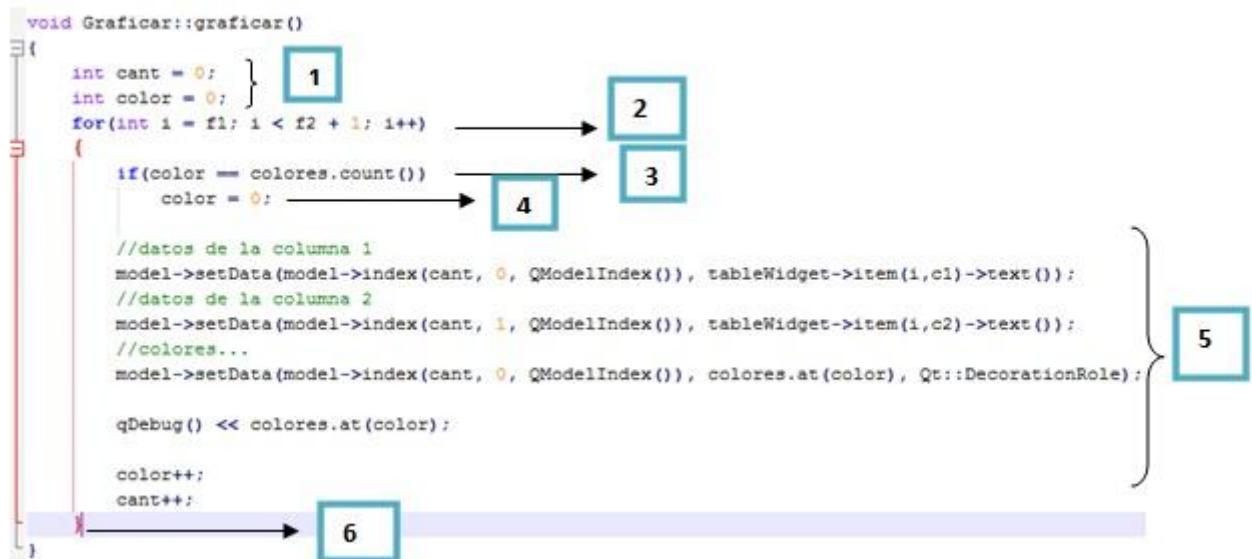


Figura 1. Codificación de la funcionalidad graficar.

Grafo de Flujo

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

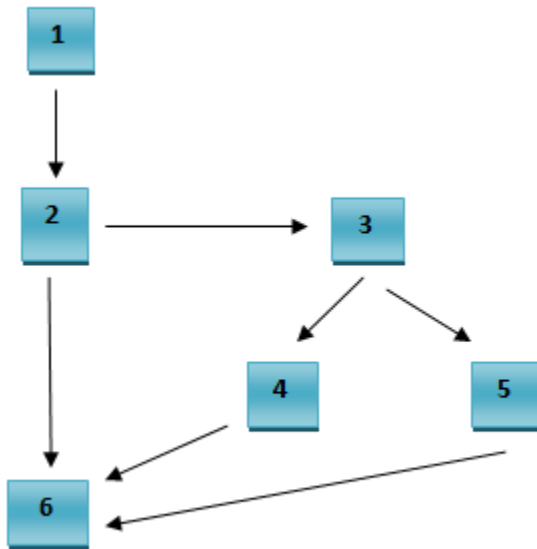


Figura 2. Grafo de Flujo asociado a la funcionalidad graficar.

La complejidad ciclomática del grafo es 3, y este es el límite superior para el número de caminos independientes y un valor límite superior para el número de pruebas que se deben diseñar y ejecutar para garantizar que se cubren todas las sentencias del programa.

No.	Camino
1	1-2-3-4-6
2	1-2-3-5-6
3	1-2-6

Tabla 25. Caminos básicos de la funcionalidad graficar.

A continuación se muestran los casos de prueba para cada uno de los caminos.

Caso de prueba para el camino básico No.1	
Camino	1-2-3-4-6
Descripción	Una vez declaradas las variables necesarias para el desarrollo de la funcionalidad, se

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

	recorren las filas seleccionadas para graficar y se verifica que no estén vacías. En caso de no estarlo, se distribuyen los colores y se grafica.
Entrada	Ninguna
Resultado esperado	Se logra crear el gráfico.
Resultado obtenido	Se logró crear el gráfico.

Tabla 26. Caso de prueba para el camino básico No.1

Caso de prueba para el camino básico No.2	
Camino	1-2-3-5-6
Descripción	Una vez declaradas las variables necesarias para el desarrollo de la funcionalidad, se recorren las filas seleccionadas para graficar y se verifica que no estén vacías. En caso de estarlo, no se grafica.
Entrada	Ninguna
Resultado esperado	No se logra crear el gráfico.
Resultado obtenido	No se logró crear el gráfico.

Tabla 27. Caso de prueba para el camino básico No.2

Caso de prueba para el camino básico No.3	
Camino	1-2-6
Descripción	Una vez declaradas las variables necesarias para el desarrollo de la funcionalidad, se recorren las filas seleccionadas para graficar. En caso de que no haya filas no se grafica.
Entrada	Ninguna
Resultado esperado	No se logra crear el gráfico.
Resultado obtenido	No se logró crear el gráfico.

Tabla 28. Caso de prueba para el camino básico No.3

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

Una vez terminados estos casos de prueba se obtiene como resultado que todas las sentencias del programa se ejecutarán al menos una vez.

Pruebas de Aceptación

En la siguiente tabla se muestra la tarjeta Prueba de Aceptación.

Caso de Prueba de Aceptación	
Código:	Historia de Usuario (No y Nombre):
Nombre:	
Descripción:	
Condiciones de ejecución:	
Entradas / Paso de ejecución:	
Resultado esperado:	
Evaluación de la prueba:	

Tabla 29. Descripción de los campos de un Caso de Prueba

Campos de la tarjeta Caso de Prueba

Historia de Usuario (No y Nombre): Número y nombre de la historia de usuario a la que corresponde.

Código: Número asignado a cada caso de prueba perteneciente a una historia de usuario determinada.

Nombre: Nombre del caso de prueba. Debe describir lo que se quiere comprobar.

Descripción: Descripción de lo que se desea probar, debe ser corta y precisa.

Condiciones de ejecución: Condiciones a tener en cuenta para ejecutarse el caso de prueba.

Entradas / Paso de ejecución: Entradas al caso de prueba en caso de necesitarlas.

Resultado esperado: Descripción breve de lo que debe suceder.

Evaluación de la prueba: Se evalúa si el caso de prueba tuvo éxito o no.

A continuación se muestran las pruebas realizadas a la solución propuesta.

Casos de Prueba del Sistema

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

A continuación se muestran dos de las pruebas de aceptación realizadas a la solución propuesta. Para observar las restantes, consultar el Anexo 5.

Caso de Prueba de Aceptación	
Código: 1	Historia de Usuario (No y Nombre): HU #6 Generar reporte del presupuesto de gasto.
Nombre: Mostrar el reporte del presupuesto de gasto.	
Descripción: Prueba para la funcionalidad generar reporte del presupuesto de gasto.	
Condiciones de ejecución: El usuario tiene que estar autenticado. Se deben haber insertado los elementos de gasto en el sistema.	
Entradas / Paso de ejecución: Se genera automáticamente el reporte del presupuesto de gasto.	
Resultado esperado: Se muestra el reporte del presupuesto de gasto correctamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 30. Descripción del Caso de Prueba de Aceptación 1

Caso de Prueba de Aceptación	
Código: 2	Historia de Usuario (No y Nombre): HU #7 Exportar reporte a PDF.
Nombre: Comprobar que se exporte el reporte a PDF.	
Descripción: Prueba para la funcionalidad exportar a PDF.	
Condiciones de ejecución: El usuario tiene que estar autenticado.	
Entradas / Paso de ejecución: Se intenta exportar el reporte a PDF.	
Resultado esperado: Se exporta el reporte al formato PDF.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 31. Descripción del Caso de Prueba de Aceptación 2

Resultados de las pruebas.

Al realizar las pruebas de aceptación durante la primera iteración se obtuvo un 31% de no conformidades, implicadas cuatro Historias de Usuario, las cuales se muestran a continuación.

- **HU Insertar nuevo elemento de gasto:** Los formularios que contienen los elementos de gasto no se cierran. Al insertar un elemento de gasto se coloca debajo del valor Total.

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

- **HU Generar reporte del presupuesto de gasto:** Las fórmulas no están registradas.
- **HU Generar reporte del control de los gastos reales en MN, CUC y Moneda Total del centro:** Las fórmulas no están registradas.
- **HU Generar reporte del control de los gastos reales en MN, CUC y Moneda Total de los proyectos:** Las fórmulas no están registradas.

Para una mejor visibilidad de los resultados de las pruebas en la primera iteración, se muestra el siguiente gráfico:

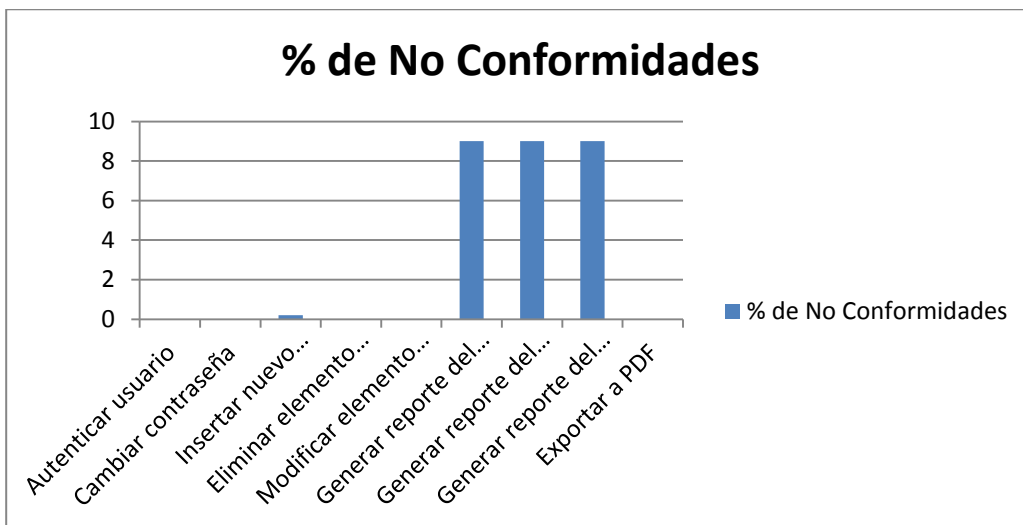


Figura 3. Resultado de las pruebas en la 1ra Iteración

Al realizar las pruebas de aceptación durante la segunda iteración se obtuvo un 3% de no conformidades, implicadas dos Historias de Usuario, las cuales se muestran a continuación.

- **HU Crear gráfico para realizar análisis:** No existe diversidad de colores en el gráfico y no se muestra el mensaje que confirme la creación del gráfico.
- **HU Guardar gráfico:** No se muestra el mensaje que confirme que el gráfico se guardó.

Para una mejor visibilidad de los resultados de las pruebas al finalizar la segunda iteración, se muestra el siguiente gráfico:

Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

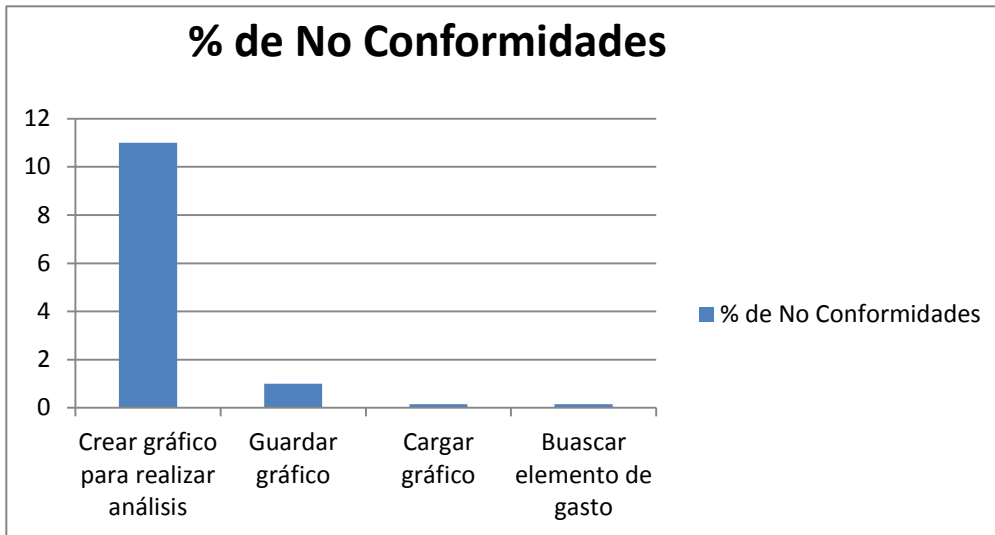
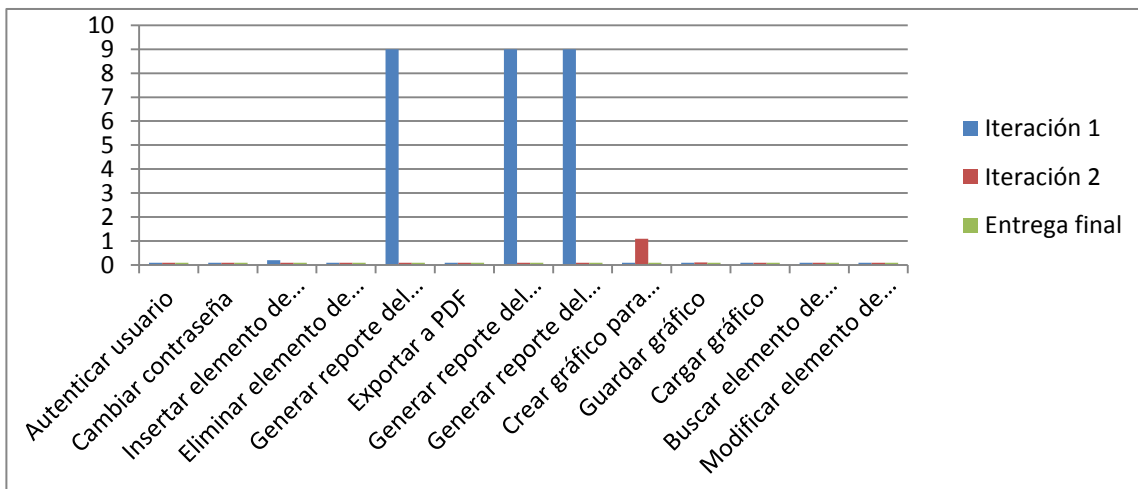


Figura 5. Resultados generales de las pruebas.

Al realizar la entrega final del producto todas las no conformidades nombradas anteriormente fueron corregidas. En el siguiente gráfico se muestra a modo general el resultado de las pruebas.



Capítulo 4. Implementación y Prueba del Sistema para la Planificación y el Control del Presupuesto

4.4 Conclusiones del Capítulo

Con la descripción de los artefactos generados en las fases de Implementación y Prueba del sistema, se concluye que, el desglose de las historias de usuario en tareas de programación garantizó que la implementación de las mismas fuera más entendible por parte de los desarrolladores. El uso de estándares de codificación posibilitó un estilo propio, permitiendo la comprensión y el mantenimiento de la herramienta. Tras las pruebas realizadas se aseguró la mejora continua de la herramienta, con un total de 6 no conformidades entre la 1ra y la 2da iteración y un 66% de satisfacción; todas las no conformidades fueron resueltas al realizarse la entrega final del producto, garantizándose un 100% de satisfacción.

Conclusiones Generales

Con la culminación del presente Trabajo de Diploma se puede concluir que:

- El estudio realizado a profundidad en el estado del arte permitió seleccionar las herramientas, tecnologías y metodología adecuadas para guiar el ciclo de vida del proyecto.
- La decisión de utilizar XP como metodología de desarrollo permitió la existencia de una colaboración entre el cliente y el equipo de desarrollo evitando desacuerdos innecesarios.
- Los resultados arrojados por las pruebas realizadas muestran el perfeccionamiento de las funcionalidades en cada iteración.
- Se le da cumplimiento al objetivo general planteado al inicio del trabajo, el cual consistió en el desarrollo de una herramienta que facilite la planificación y control del presupuesto en el Centro GEYSED.

Recomendaciones

- Extender la solución al centro DATEC para que sea utilizada la herramienta que planifica y controla el presupuesto por su asesor económico.
- Perfeccionar las funcionalidades creadas para lograr una mayor satisfacción por parte del cliente.

Referencias Bibliográficas

Àlvarez, Sara. 2007. desarrolloweb.com. [En línea] 31 de Julio de 2007. [Citado el: 11 de Diciembre de 2012.] <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.

Arias, Marin Marvin, David. 2008. EditBoard.com. *EditBoard.com*. [En línea] 16 de Octubre de 2008. [Citado el: 11 de Diciembre de 2012.] <http://catedraprogramacion.foroactivos.net/t83-definicion-de-lenguaje-de-programacion-tipos-ejemplos>.

Beck, Kent. 1999. *Extreme Programming Explained. Embrace Change*. s.l. : s.l.:Addison-Wesley Professional, 1999. ISBN: 9780201616415.

Beck, Kent y Fowler, Martin. 2000. *Planning Extreme Programming*. s.l. : s.l.: First Edition ed, 2000.

Booch, Grady Rumbaugh James Jacobson Ivar. 2007. *El Lenguaje Unificado de Modelado. Manual de Referencia*. s.l. : Prentice Hall, 2007.

Burbani, Ruiz Jorge E. 2005. *Presupuestos: enfoque de gestión, planeación y control de recursos*. Bogota, Colombia : NOMOS, 2005. ISBN 958410359-8.

Carvajal, José Carlos Riola. 2008. *Metodologías Ágiles: Herramientas y Modelos de Desarrollo para aplicaciones Java EE como metodología empresarial. Tesis final de Máster*. Barcelona, España : Máster en Tecnologías de la Información, 2008.

Cervantes, Humberto. 2010. SG Software Guru. *SG Software Guru*. [En línea] UAM-Iztapalapa, 2010. [Citado el: 27 de Febrero de 2013.] <http://sg.com.mx/content/view/922>.

Chossudovsky, Michel y Gavin, Andrew Marsh. 2010. *La crisis económica global, la Gran Depresión del Siglo XXI*. Montreal, Canadá : Global Research, 2010.

Referencias Bibliográficas

De Benito, Rodolfo Aragón y Sánchez, Ana Jesús Granda. 2007. *Manual de Presto 4ta Edición*. Madrid, España : MCGraw-Hill/Interamericana de España S.A., 2007. ISBN: 9788448156206.

Dra Zaldívar, Martha Puig. 2007. *Las transformaciones en el sistema empresarial a partir de los 90 y la búsqueda de la eficiencia y la competitividad*. Ciudad de La Habana : Facultad de Economía UH, 2007.

Extensión de Visual Paradigm for UML para el desarrollo dirigido por modelos de aplicaciones de gestión de información. Cabrera, González Lianet. 2012. No.10, La Habana, Cuba : Serie Científica de la Universidad de las Ciencias Informáticas, 2012, Vol. Vol.5. ISSN:/RNPS:.

Extreme Programming, : A gentle introduction. 2009. Agile Process. *Agile Process*. [En línea] Copyrigh(c), 28 de Septiembre de 2009. <http://www.extremeprogramming.org/>.

Guillermo, Gretchen Hernández. 2012. Planificación y Control del presupuesto. *Entrevista Informal*. Ciudad de La Habana : Personal, Diciembre de 2012.

Hernández, Rolando Alfredo León y Coello, Sayda González. 2011. *El Proceso de Investigación Científica*. Ciudad de La Habana : Editorial Universitaria, 2011. ISBN 978-959-16-1307-3.

Hristov, Alexander. 2007. *Manual de Estilo de Programación*. s.l. : PLANETALIA Formación y Consultoría, 2007.

Izquierdo, Luis R. 2007. [En línea] 20 de Mayo de 2007. [Citado el: 11 de Diciembre de 2012.] <http://luis.izqui.org/resources/ProgOrientadaObjetos.pdf>.

Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000. *El Proceso Unificado del Desarrollo de Software*. s.l. : PEARSON EDUCACION,SA Madrid, 2000. 84-7829-036-2.

Jiménez, E.M. 2011. *Ingeniería de Software Ágil*. 2011. ISBN: 978-1-4477-5566-1.

Referencias Bibliográficas

Jiménez, Millán Adolfo. 2011. Zator Systems. [En línea] 28 de Abril de 2011. [Citado el: 11 de Diciembre de 2012.] www.zator.com/Cpp/E1_2.htm.

Joskowicz, José. 2008. *Reglas y Prácticas en eXtreme Programming*. Canada : Department of Computer Science, Univeristy of Manitoba, 2008.

Kohen, Eli I. Gómez. 2010. *Estudio de las prestaciones de una red Ad-Hoc Wi-Fi*. España : Arquitectura de Computadoras, 2010.

Larman, Craig. 2005. *UML y Patrones Introducción al análisis y diseño orientado a objetos*. s.l.: PRENTICE HALL, México,1999, 2005. ISBN: 970-17-0261-1.

Leiseca, Hernández Esteban. 2010. *Sistema de Planificacion del Presupuesto del Departamento de Transporte de la Unidad de Aseguramiento Aduanal*. Ciudad de La Habana : s.n., 2010.

Martín, Eduardo Luis. 2005. El Economista. [En línea] 2005. [Citado el: 3 de Diciembre de 2012.] <http://www.eleconomista.cubaweb.cu/2005/edicionimpresa/premioeconomiamiguel.html>.

Molpeceres, Alberto. 2003. [En línea] 15 de Diciembre de 2003. [Citado el: 7 de Diciembre de 2012.] <http://www.willydev.netdescargasarticulosgeneralcualxpfddrup.pdf>.

Montoya, Fernando Cerio, Sambeat, Andrés Vicién y Fabra, Oscar Rodríguez. 2013. *Impacto de la crisis internacional en las NNUU: Ajustes en el presupuesto ordinario de la organización*. España : EMAD Instituto Español de Estudios Estratégicos, 2013.

Pecos, Martínez Daniel. 2012. [En línea] 2012. [Citado el: 11 de Diciembre de 2012.] http://www.danielpecos.com/docs/mysql_postgres/x15.html.

Porteiro, Marisel Sosa y Cobo Morales, Pedro H. 2007. La Revista del empresario cubano. [En línea] 2007. [Citado el: 7 de Diciembre de 2012.] http://www.betsime.disaic.cu/secciones/eco_enemar_07.htm#2.

Referencias Bibliográficas

PostgreSQL, Professional. 2013. 2ndQuadrant Professional PostgreSQL. *2ndQuadrant Professional PostgreSQL*. [En línea] 2ndQuadrant Ecuador CDLA, 2013. [Citado el: 30 de Mayo de 2013.] <http://www.2ndquadrant.com/es/postgresql-vs-mysql/>.

Pressman, Roger S. 2012. *Ingeniería del Software. Un enfoque práctico*. 2012.

Programming, C++. 2012. C++ Programming. [En línea] 4 de Abril de 2012. [Citado el: 11 de Diciembre de 2012.] <http://xhrist14n.wordpress.com/2012/04/04/qt-programando-en-c/>.

Sánchez, González Carlos. 2004. *Proyecto de fin de carrera de Ingeniería Informática*. Universidad de Coruña, Facultad de Informática : Departamento de Tecnologías de la Información y las Comunicaciones: s.n., 2004.

software. 2007. software.com.ar targetware. [En línea] 2007. [Citado el: 11 de Diciembre de 2012.] <http://www.software.com.ar/visual-paradigm-para-uml.html>.

Sommerville, Ian. 2005. *Ingeniería del software*. s.l. : PEARSON EDUCACION , SA Madrid, 2005. 84-7829-074-5.

Systems, Eastern Software. 2006. Managinf: servicios y software ERP. *Managinf: servicios y software ERP*. [En línea] 8 de Septiembre de 2006. [Citado el: 28 de Febrero de 2013.] <http://www.managinf.com/arquitectura.pdf>.

VERSAT, Software. 2012. *Manual del Usuario y de Explotación VERSAT SARASOLA*. Santa Clara, Villa Clara, Cuba : TEICOSOFT MINAZ, 2012.

Welsch, Glenn A. 2005. *Presupuestos: planificacion y control*. Mexico : Pearson Educacion, 2005. 970-26-0551-2.

Bibliografía

Alemany, Garrido Salvador. 2009. *Introducción a Qt. Programación gráfica en C++ con Qt4*. s.l. : fiv.upv, 2009.

Àlvarez, Sara. 2007. desarrolloweb.com. [En línea] 31 de Julio de 2007. [Citado el: 11 de Diciembre de 2012.] <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.

Americati. 2006. Americati. [En línea] 11 de Noviembre de 2006. [Citado el: 11 de Diciembre de 2012.] http://www.americati.com/doc/ventajas_c.pdf.

Arias, Marin Marvin, David. 2008. EditBoard.com. *EditBoard.com*. [En línea] 16 de Octubre de 2008. [Citado el: 11 de Diciembre de 2012.] <http://catedraprogramacion.foroactivos.net/t83-definicion-de-lenguaje-de-programacion-tipos-ejemplos>.

Beck, Kent. 1999. *Extreme Programming Explained. Embrace Change*. s.l. : s.l.:Addison-Wesley Professional, 1999. ISBN: 9780201616415.

Beck, Kent y Fowler, Martin. 2000. *Planning Extreme Programming*. s.l. : s.l.: First Edition ed, 2000.

Booch, Grady Rumbaugh James Jacobson Ivar. 2007. *El Lenguaje Unificado de Modelado. Manual de Referencia*. s.l. : Prentice Hall, 2007.

Burbani, Ruiz Jorge E. 2005. *Presupuestos: enfoque de gestión, planeación y control de recursos*. Bogota, Colombia : NOMOS, 2005. ISBN 958410359-8.

Cáceres, Tello Jesús. 2009. *Patrones de diseño: ejemplo de aplicación en los Generative Learning Object*. Universidad de Alcalá Dpto. Ciencias de la Computación : RED. Revista de Educación a Distancia, 2009.

Carrillo, Pérez Isaías. 2008. *Metodologías del desarrollo del software*. 2008.

Carvajal, José Carlos Riola. 2008. *Metodologías Ágiles: Herramientas y Modelos de Desarrollo para aplicaciones Java EE como metodología empresarial. Tesis final de Máster.* Barcelona, España : Máster en Tecnologías de la Información, 2008.

Cervantes, Humberto. 2010. SG Software Guru. *SG Software Guru.* [En línea] UAM-Iztapalapa, 2010. [Citado el: 27 de Febrero de 2013.] <http://sg.com.mx/content/view/922>.

Chossudovsky, Michel y Gavin, Andrew Marsh. 2010. *La crisis económica global, la Gran Depresión del Siglo XXI.* Montreal, Canadá : Global Research, 2010.

Cubano, La Revista del Empresario. 2009. Perfeccionamiento Empresarial. C.Habana : Disaic Casa Consultora, 2009. Vol. No.2.

De Benito, Rodolfo Aragón y Sánchez, Ana Jesús Granda. 2007. *Manual de Presto 4ta Edición.* Madrid, España : MCGraw-Hill/Interamericana de España S.A., 2007. ISBN: 9788448156206.

Dra Zaldívar, Martha Puig. 2007. *Las transformaciones en el sistema empresarial a partir de los 90 y la búsqueda de la eficiencia y la competitividad.* Ciudad de La Habana : Facultad de Economía UH, 2007.

ERP-Spain.com. 2006. [En línea] 24 de Mayo de 2006. [Citado el: 3 de Diciembre de 2012.] <http://www.erp-spain.com/articulo/25152/>.

Extensión de Visual Paradigm for UML para el desarrollo dirigido por modelos de aplicaciones de gestión de información. Cabrera, González Lianet. 2012. No.10, La Habana, Cuba : Serie Científica de la Universidad de las Ciencias Informáticas, 2012, Vol. Vol.5. ISSN:/RNPS:.

Extreme Programming, : A gentle introduction. 2009. Agile Process. *Agile Process.* [En línea] Copyrigh(c), 28 de Septiembre de 2009. <http://www.extremeprogramming.org/>.

Fernández, Lanvin Daniel. Definición de una arquitectura software para el diseño de aplicaciones . *Definición de una arquitectura software para el diseño de aplicaciones* . [En línea] [Citado el: 28 de Febrero de 2013.] <http://pdimagenes.iespana.es/carlos/ArquitecturaJ2EE.PDF>.

Gaceta Oficial No.013 Extraordinaria. Justicia, Gaceta Oficial de la República de Cuba. Ministerio de. 2011. La Habana : Contraloría General de la República, 2011. ISSN 1682-7511.

Greiner, Oliver. *Importancia del control presupuestario en el contexto de la implementación estratégica: reflexiones y resultados empíricos*. España : Horváth & Partners.

Gudynas, Eduardo. Enero, 2009. *La primera crisis global del siglo XXI. Miradas y reflexiones*. Montevideo, Uruguay : CLAES D3E Centro Latino Americano de Ecología Social, Enero, 2009.

Guillermo, Gretchen Hernández. 2012. Planificación y Control del presupuesto. *Entrevista Informal*. Ciudad de La Habana : Personal, Diciembre de 2012.

Hernández, Rolando Alfredo León y Coello, Sayda González. 2011. *El Proceso de Investigación Científica*. Ciudad de La Habana : Editorial Universitaria, 2011. ISBN 978-959-16-1307-3.

Hristov, Alexander. 2007. *Manual de Estilo de Programación*. s.l. : PLANETALIA Formación y Consultoría, 2007.

Izquierdo, Luis R. 2007. [En línea] 20 de Mayo de 2007. [Citado el: 11 de Diciembre de 2012.] <http://luis.izqui.org/resources/ProgOrientadaObjetos.pdf>.

Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000. *El Proceso Unificado del Desarrollo de Software*. s.l. : PEARSON EDUCACION,SA Madrid, 2000. 84-7829-036-2.

Jiménez, E.M. 2011. *Ingeniería de Software Ágil*. 2011. ISBN: 978-1-4477-5566-1.

Jiménez, Millán Adolfo. 2011. Zator Systems. [En línea] 28 de Abril de 2011. [Citado el: 11 de Diciembre de 2012.] www.zator.com/Cpp/E1_2.htm.

Joskowicz, José. 2008. *Reglas y Prácticas en eXtreme Programming*. Canada : Department of Computer Science, Univeristy of Manitoba, 2008.

Kohen, Eli I. Gómez. 2010. *Estudio de las prestaciones de una red Ad-Hoc Wi-Fi*. España : Arquitectura de Computadoras, 2010.

Larman, Craig. 2005. *UML y Patrones Introducción al análisis y diseño orientado a objetos*. s.l. : PRENTICE HALL, México, 1999, 2005. ISBN: 970-17-0261-1.

Leiseca, Hernández Esteban. 2010. *Sistema de Planificacion del Presupuesto del Departamento de Transporte de la Unidad de Aseguramiento Aduanal*. Ciudad de La Habana : s.n., 2010.

Linux, Atelier. 2009. Geeks & Linux Atelier. [En línea] 15 de Mayo de 2009. [Citado el: 11 de Diciembre de 2012.] <http://www.glatelier.org/2009/05/qt-creator-desarrollando-aplicaciones-rapidamente/>.

Martín, Eduardo Luis. 2005. El Economista. [En línea] 2005. [Citado el: 3 de Diciembre de 2012.] <http://www.eleconomista.cubaweb.cu/2005/edicionimpresa/premioeconomiamiguel.html>.

Mistral, Universidad Gabriela. 2012. Buenas Tareas. [En línea] Septiembre de 2012. <http://www.buenastareas.com/ensayos/Investigaci%C3%B3n-De-La-Aplicaci%C3%B3n/5231101.html>.

Molpeceres, Alberto. 2003. [En línea] 15 de Diciembre de 2003. [Citado el: 7 de Diciembre de 2012.] <http://www.willydev.netdescargasarticulosgeneralcualxpfddrup.pdf>.

Montoya, Fernando Cerio, Sambeat, Andrés Vicién y Fabra, Oscar Rodríguez. 2013. *Impacto de la crisis internacional en las NNUU: Ajustes en el presupuesto ordinario de la organización*. España : EMAD Instituto Español de Estudios Estratégicos, 2013.

Murua, Olalde Juan. 2004. [En línea] 7 de Diciembre de 2004. [Citado el: 7 de Diciembre de 2012.] http://www.octobirp.es/material%20para%20curso%20METODOLOGIAS/_Metodologias%20Desarrollo%20Software.doc.

Pecos, Martínez Daniel. 2012. [En línea] 2012. [Citado el: 11 de Diciembre de 2012.] http://www.danielpecos.com/docs/mysql_postgres/x15.html.

Porteiro, Marisel Sosa y Cobo Morales, Pedro H. 2007. La Revista del empresario cubano. [En línea] 2007. [Citado el: 7 de Diciembre de 2012.] http://www.betsime.disaic.cu/secciones/eco_enemar_07.htm#2.

PostgreSQL, Professional. 2013. 2ndQuadrant Professional PostgreSQL. *2ndQuadrant Professional PostgreSQL*. [En línea] 2ndQuadrant Ecuador CDLA, 2013. [Citado el: 30 de Mayo de 2013.] <http://www.2ndquadrant.com/es/postgresql-vs-mysql/>.

Pressman, Roger S. 2012. *Ingeniería del Software. Un enfoque práctico*. 2012.

Programming, C++. 2012. C++ Programming. [En línea] 4 de Abril de 2012. [Citado el: 11 de Diciembre de 2012.] <http://xhrist14n.wordpress.com/2012/04/04/qt-programando-en-c/>.

Sánchez, González Carlos. 2004. *Proyecto de fin de carrera de Ingeniería Informática*. Universidad de Coruña, Facultad de Informática : Departamento de Tecnologías de la Información y las Comunicaciones: s.n., 2004.

software. 2007. software.com.ar targetware. [En línea] 2007. [Citado el: 11 de Diciembre de 2012.] <http://www.software.com.ar/visual-paradigm-para-uml.html>.

Software, Presto. 2013. Presupuestos. *Presupuestos*. [En línea] Soft, 2013. [Citado el: 2013 de Febrero de 28.] <http://www.prestosoftware.cl/>.

Sommerville, Ian. 2005. *Ingeniería del software*. s.l. : PEARSON EDUCACION , SA Madrid, 2005. 84-7829-074-5.

Systems, Eastern Software. 2006. Managinf: servicios y software ERP. *Managinf: servicios y software ERP*. [En línea] 8 de Septiembre de 2006. [Citado el: 28 de Febrero de 2013.] <http://www.managinf.com/arquitectura.pdf>.

VERSAT, Software. 2012. *Manual del Usuario y de Explotación VERSAT SARASOLA*. Santa Clara, Villa Clara, Cuba : TEICOSOFT MINAZ, 2012.

Web, Aplicaciones basadas en. Aplicaciones basadas en Web. *Aplicaciones basadas en Web*. [En línea] <http://www.janium.com/page2/page1/page6/page7/page7.html>.

Welsch, Glenn A. 2005. *Presupuestos: planificacion y control*. Mexico : Pearson Educacion, 2005. 970-26-0551-2.

Glosario

Array: arreglo, conjunto de variables en donde cada una de ellas puede ser referenciada utilizando su posición relativa, es decir su ubicación en relación con el primer de dicho conjunto.

Centros de costo: es una división que genera costos para la organización, pero solo indirectamente le añaden beneficio o utilidad. Ejemplos típicos de esto son los departamentos.

Control Interno: es el proceso integrado a las operaciones de una empresa, con un enfoque de mejoramiento continuo, efectuado por la dirección y el personal. Contribuye a prever y limitar los riesgos internos y externos.

Ejecución: ejecución real del presupuesto en el rango del valor planificado.

Empresa: es concebida como un conjunto de factores humanos, materiales, técnicos y financieros, ordenados según las normas cierta estructura organizativa, y relacionados para el cumplimiento de determinados objetivos.

Gasto: egreso o salida de dinero que una persona o empresa debe pagar para un artículo o por un servicio.

Gestión empresarial: proceso de planificar, organizar, ejecutar y evaluar una empresa.

Multi-versión: técnica avanzada para mejorar las prestaciones de una base de datos en un entorno multiusuario.

Organización y Dirección: etapas que permiten asignar recursos y poner en marcha lo planificado, con el fin de alcanzar los objetivos propuestos.

Perfeccionamiento Empresarial: es un sistema de dirección y gestión empresarial cuyo objetivo central es incrementar al máximo la eficiencia y la competitividad de la empresa estatal socialista sobre la base de otorgarles las facultades y atribuciones necesarias para una correcta administración.

Sobre-ejecución: ejecución real del presupuesto por encima del valor planificado.