

PRÁCTICA EMPRESARIAL EN VISIÓN INGENIERÍA
ACTIVIDADES DE APOYO EN EL DESARROLLO DE LOS PRODUCTOS
SOFTWARE DE LA EMPRESA VISIÓN INGENIERÍA

CAROLINA SEPULVEDA ACEVEDO

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECHANICAS
ESCUELA DE INGENIERÍA DE SISTEMAS
BUCARAMANGA
2006

PRÁCTICA EMPRESARIAL EN VISIÓN INGENIERÍA
ACTIVIDADES DE APOYO EN EL DESARROLLO DE LOS PRODUCTOS
SOFTWARE DE LA EMPRESA VISIÓN INGENIERÍA

PRACTICANTE EMPRESARIAL:
CAROLINA SEPULVEDA ACEVEDO
CÓDIGO 1921926

Este proyecto es presentado como requisito para optar al título de
Ingeniero de Sistemas e Informática

DIRECTOR – FERNANDO ANTONIO ROJAS MORALES
TUTOR – GERARDO ARMANDO GARAVITO FLÓREZ

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECHANICAS
ESCUELA DE INGENIERÍA DE SISTEMAS
BUCARAMANGA
2006

AGRADECIMIENTOS

En primer lugar a Dios que me dió vida y salud para terminar la carrera.

A mis padres Hermes y Alba Luz, a mi hermana Paola Andrea porque siempre me han apoyado y en especial a mi mejor amigo, mi hermano Jonnathan que siempre ha confiado en mi y quien ha sido la inspiración de mi trabajo.

A todas aquellas personas que han confiado en mis capacidades como estudiante, profesional y amiga.

Al Ingeniero Gerardo Armando Garavito Gerente de Visión Ingeniería, quien me abrió las puertas de su empresa para realizar la práctica empresarial.

TITULO: ACTIVIDADES DE APOYO EN EL DESARROLLO DE LOS PRODUCTOS SOFTWARE DE LA EMPRESA VISION INGENIERIA.*

AUTOR: CAROLINA SEPULVEDA ACEVEDO**

PALABRAS CLAVE: ORACLE, ACCESS, VISUAL BASIC, ASP

DESCRIPCIÓN

El presente trabajo de grado en modalidad práctica empresarial, tiene como objetivo principal Realizar actividades de apoyo en el desarrollo, ajustes y pruebas de opciones específicas de los productos software de la empresa Visión Ingeniería. Dicho trabajo de apoyo se efectuó sobre los sistemas OPTIMIZADOR (Optimizador para Condiciones de Operación en Estaciones de Bombeo de la VIT - ICP) y SCCPL (proyecto de Consultoría para el desarrollo de nuevas funcionalidades del sistema SCCPL). Sistemas desarrollados sobre una base de datos Oracle 9i con Herramientas de desarrollo VISUAL BASIC 6.0 y ASP y los cuales son propiedad de la Empresa ECOPETROL – ICP, INSTITUTO COLOMBIANO DE PETRÓLEO. El proceso de apoyo consistió en desarrollar y ajustar reportes tanto aquellos que se generaban en VISUAL BASIC como los que se generaban en ASP para ser integrados con herramientas Office. Desarrollo y ajustes en Formas ya existentes en los sistemas OPTIMIZADOR y SCCPL CONVENIOS. Pruebas y documentación a las diferentes opciones que son ajustadas o desarrolladas para su implantación en los sitios donde actualmente utilizan dichos sistemas.

Los desarrollos y ajustes elaborados por el practicante fueron revisados por personal de la Empresa para verificar su correcto funcionamiento antes de ser implantados en las aplicaciones.

* Trabajo de Grado

** Escuela de Ingeniería de Sistemas e Informática.

Director: Ing. Fernando Antonio Rojas Morales

Tutor Empresarial: Ing. Gerardo Armando Garavito Florez

TITLE: ACTIVITIES OF SUPPORT IN THE DEVELOPMENT OF THE PRODUCTS SOFTWARE OF THE VISION INGENIERIA COMPANY *

AUTHOR: CAROLINA SEPULVEDA ACEVEDO **

KEY WORDS: ORACLE, ACCESS, VISUAL BASIC, ASP

DESCRIPTION

The present work degree in enterprise practical modality, has like primary target support to the workings of the company VISION in the development, adjustments and tests of precise options of the products software of the Vision Ingenieria Company.

This work of support took place on systems OPTIMIZADOR (Optimizador para Condiciones de Operación en Estaciones de Bombeo de la VIT - ICP) and SCCPL (Proyecto de Consultoría para el desarrollo de nuevas funcionalidades del sistema SCCPL). These systems were developed on a data base Oracle 9i, with Development tools VISUAL BASIC 6.0 y ASP and which are property of the ECOPETROL – ICP (INSTITUTO COLOMBIANO DE PETRÓLEO) Company. The support process consisted of developing and as much fitting to reports those that were generated in VISUAL BASIC like which they were generated in ASP to be integrated with Office tools. Development and adjustments in already existing Forms on some Tools and creation of new screens required for systems OPTIMIZADOR and SCCPL CONVENIOS. Tests and documentation for the different options that are fit or developed for their implantation in the sites where at the moment use these systems.

The developments and adjustments elaborated by the student in practice were reviewed by personnel of the Company to verify their correct operation before being implanted in the applications.

* Graduation Work

** School of Engineering of Systems and Computer science.

Director: Ing. Fernando Antonio Rojas Morales

Tutor: Ing. Gerardo Armando Garavito Florez

TABLA DE CONTENIDO

	Pág.
INTRODUCCIÓN	9
1. DESCRIPCIÓN DEL PROYECTO.	10
1.1. OBJETIVO GENERAL	10
1.2. OBJETIVOS ESPECÍFICOS	10
1.3. DEFINICIÓN DEL PROBLEMA	11
1.4. DESCRIPCIÓN DE OBJETIVOS	11
1.5. IMPACTO Y VIABILIDAD	11
2. METODOLOGÍA Y PLAN DE TRABAJO	12
2.1. METODOLOGÍA	12
3. PRESENTACIÓN DE LA EMPRESA	23
3.1. DESCRIPCIÓN DE LA EMPRESA	23
3.2. RESEÑA HISTÓRICA	23
3.3. MISIÓN	24
3.4. VISIÓN	24
4. EL SISTEMA OPTIMIZADOR	25
5. EL SISTEMA SCCPL	26
6. PSP (PERSONAL SOFTWARE PROCESS)	26
7. TSP (TEAM SOFTWARE PROCESS)	35
7.1. ¿QUE ES TSP?	35
7.2. ROLES	35
7.3. PROCESO	36
7.4. PLANIFICACION	37
7.5. SEGUIMIENTO	37

7.6	BENEFICIOS	38
7.7	APLICACIÓN DEL PSP	38
8.	PHVA	40
9.	DETALLADO DE ACTIVIDADES	41
10.	TIEMPO POR TIPO DE TAREA EN CADA DIA DE LA SEMANA	59
11.	TIEMPO POR PROCESO	61
	CONCLUSIONES	62
	BIBLIOGRAFIA	64

LISTA DE TABLAS

	Pág.
Tabla 1. Fases del PHVA	40
Tabla 2. Detallado de Actividades	58
Tabla 3. Tiempos por Tarea en cada día de la semana	59
Tabla 4. Tiempos por proceso	61

LISTA DE FIGURAS

	Pág.
Figura 1. Proceso del PSP	36
Figura 2. Planificación del PSP	37
Figura 3. Formato de registro de tiempos	38
Figura 4. Gráfico de Tiempos por Tarea en cada día de la semana	60
Figura 5. Gráfico de Tiempos por proceso	61

GLOSARIO

ORACLE: es un sistema de administración de base de datos (o RDBMS por el acrónimo en inglés de Relational Data Base Management System), fabricado por Oracle Corporation.

ASP: Es el lenguaje con el que se definen las páginas Web. Básicamente se trata de un conjunto de etiquetas que sirven para definir la forma en la que se presenta el texto y otros elementos de la página.

RUP (Rational Unified Process): es un proceso de desarrollo de software: “conjunto de actividades necesarias para transformar los requisitos del usuario en un sistema software”. RUP es fruto de los aportes de un gran número de investigadores y empresas de desarrollo de programas que busca asegurar la producción de software de alta calidad, satisfaciendo las necesidades del cliente, y con arreglo a un plan y presupuesto predecibles.

PSP (Personal Software Process): El Personal Software Process, conocido por sus siglas como PSP, es una metodología de reciente creación, proveniente del Instituto de Ingeniería del Software (SEI). PSP es una alternativa dirigida a los ingenieros de sistemas, que les permite mejorar la forma en la que construyen software. Considerando aspectos como la planeación, calidad, estimación de costos y productividad.

INTRODUCCIÓN

La empresa Visión Ingeniería es una empresa de consultoría en el área de bases de datos relacionales, desarrollos en Intranet e Internet, sistemas de información geográfica y procesos de documentación técnica e ingeniería de reversa.

VISIÓN ofrece soluciones informáticas y de gestión tecnológica con parámetros de calidad, siguiendo estándares nacionales e internacionales, asegurando resultados con excelencia, efectividad y eficiencia mediante la asimilación, adaptación e innovación tecnológica.

La practica empresarial como herramienta de adaptación para el estudiante a su medio laboral, se constituye en una valiosa herramienta tanto para la universidad como proveedora de profesionales capacitados, como para el estudiante en su formación profesional, ya que nos permite culminar la formación que nos ofrece la universidad enfrentándonos a un ambiente laboral y a las responsabilidades que esto representa.

Las labores de pruebas, afinamientos en procesos y ajustes menores a formas y reportes consumen gran cantidad de recursos en cuanto a personal y tiempo se refieren, estas tareas cortas son el espacio ideal para que el nuevo ingeniero comience a desarrollar su capacidad de trabajo en un ambiente laboral y permiten a la empresa valorar el desempeño del nuevo profesional, brindando la oportunidad a la empresa de contar con personal capacitado y al estudiante de obtener un puesto de trabajo.

1. DESCRIPCIÓN DEL PROYECTO.

La práctica empresarial en la empresa Visión Ingeniería consiste en dar apoyo a las labores de la firma en el desarrollo, ajustes y pruebas de opciones puntuales de los sistemas que poseen, con el fin de Afianzar los conocimientos en bases de datos y herramientas cliente – servidor, aplicando los conocimientos de Ingeniería de Software en casos prácticos.

Dicho apoyo se concentrará en los sistemas OPTIMIZADOR y SCCPL-CONVENIOS sistemas de propiedad de la empresa y los cuales requieren de labores de pruebas y ajustes menores para su correcto funcionamiento, para estas labores se contará con el apoyo del personal de la empresa que revisará los ajustes que se hagan por parte del practicante a los sistemas antes mencionados.

1.1. OBJETIVO GENERAL

Dar apoyo a las labores de la firma VISION en el desarrollo, ajustes y pruebas de opciones puntuales de los sistemas que poseen, con el fin de Afianzar los conocimientos en bases de datos y herramientas cliente – servidor, aplicando los conocimientos de Ingeniería de Software en casos prácticos.

1.2. OBJETIVOS ESPECÍFICOS

- Desarrollar, ajustar y validar opciones puntuales asociadas a proyectos software, por medio de herramientas Microsoft.
- Realizar pruebas a los diferentes proyectos software como parte del proceso de control de calidad del producto final que se entrega a los clientes.
- Apoyar en el manejo de prototipos para la evaluación de facilidades de control de versiones del software.
- Apoyar la documentación técnica y de usuario de las opciones desarrolladas por la empresa.

1.3. Definición del problema

Actualmente, la empresa Visión Ltda. desarrolla proyectos de ingeniería de software, en los cuales se realizan actividades puntuales como el desarrollo de opciones de valor agregado para sus clientes y la realización de pruebas y ajustes que permitan a la empresa garantizar que sus productos cumplen con los estándares de calidad y los requisitos establecidos. Estas actividades son de carácter operativo y repetitivo, y requieren de una mayor dedicación de tiempo por parte de los ingenieros de la empresa, el cual puede ser ocupado en la generación de otros proyectos software. En estos términos, la empresa busca colocar estudiantes de ingeniería de sistemas en práctica, que no generen un alto impacto financiero y desarrollen estas actividades de apoyo que son necesarias tanto para los productos ofrecidos a los clientes externos como internos.

1.4 Descripción de objetivos

Acompañando los procesos orientados a la producción de software de Visión Ltda., se espera apoyar a la empresa en actividades puntuales de algunos de los proyectos que esta desarrollando. Las actividades son: Desarrollar, ajustar y validar opciones asociadas a proyectos realizados por la empresa con herramientas Microsoft, realizar pruebas en los diferentes proyectos software como parte del proceso de control de calidad del producto final que se entrega a los clientes, apoyar la documentación técnica y de usuario de opciones desarrolladas por la firma y manejar prototipos de evaluación de facilidades de control de versiones; aplicando las metodologías y lineamientos establecidos por la empresa.

1.5 Impacto y Viabilidad

Con este proyecto se pretende lograr el apoyo en actividades que son repetitivas y que requieren tiempo excesivo a los ingenieros de la empresa. Lograr la formación de recurso humano que pueda ser vinculado posteriormente a la empresa, y apoyar actividades de mejoramiento continuo orientadas a optimizar los procesos de producción de software de la empresa.

La empresa VISION, cuenta con los recursos computacionales, el espacio físico asignado y con profesionales certificados en Oracle y Microsoft con experiencia comprobada, que permiten a la empresa tener al estudiante en práctica empresarial. Además la empresa cuenta en la actualidad con proyectos tanto para clientes externos como clientes internos, que requieren de apoyo en desarrollo y pruebas de opciones puntuales y que permitan a la empresa generar un valor agregado a sus productos.

2. METODOLOGÍA Y PLAN DE TRABAJO¹

2.1 Metodología

Rational Unified Process (RUP)

El Proceso Unificado de Desarrollo de Software (RUP - Rational Unified Process) es el proceso de desarrollo de software creado por Rational Rose – IBM. Comprende el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software.

RUP es un marco genérico que puede especializarse para una variedad de tipos de sistemas, diferentes áreas de aplicación, tipos de organizaciones, niveles de aptitud y diferentes tamaños de proyectos.

RUP está basado en componentes. El software esta formado por componentes software interconectados a través de interfaces.

RUP está dirigido por casos de uso, centrado en la arquitectura, y es iterativo e incremental.

Dirigido por Casos de Uso

Un caso de uso es un fragmento de funcionalidad del sistema que proporciona un resultado de valor a un usuario. Los casos de uso modelan los requerimientos funcionales del sistema.

Todos los casos de uso juntos constituyen el modelo de casos de uso.

Los casos de uso también guían el proceso de desarrollo (diseño, implementación y prueba). Basándose en los casos de uso los desarrolladores crean una serie de modelos de diseño e implementación que llevan a cabo los casos de uso. De este modo los casos de uso no solo inician el proceso de desarrollo sino que le proporcionan un hilo conductor, avanza a través de una serie de flujos de trabajo que parten de los caso de uso.

¹ Tomado de El Proceso Unificado de Desarrollo de Software (Gustavo Torossi)

Centrado en la Arquitectura

La arquitectura de un sistema software se describe mediante diferentes vistas del sistema en construcción.

El concepto de arquitectura software incluye los aspectos estáticos y dinámicos más significativos del sistema.

La arquitectura es una vista del diseño completo con las características más importantes resaltadas, dejando los detalles de lado.

Arquitectura: Conjunto de decisiones significativas acerca de la organización de un sistema software, la selección de los elementos estructurales a partir de los cuales se compone el sistema, las interfaces entre ellos, su comportamiento, sus colaboraciones, y su composición.

Los casos de uso y la arquitectura están profundamente relacionados. Los casos de uso deben encajar en la arquitectura, y a su vez la arquitectura debe permitir el desarrollo de todos los casos de uso requeridos, actualmente y a futuro.

El arquitecto desarrolla la forma o arquitectura a partir de la comprensión de un conjunto reducido de casos de uso fundamentales o críticos (usualmente no mas del 10 % del total). En forma resumida, podemos decir que el arquitecto:

- Crea un esquema en borrador de la arquitectura comenzando por la parte no específica de los casos de uso (por ejemplo la plataforma) pero con una comprensión general de los casos de uso fundamentales.
- A continuación, trabaja con un conjunto de casos de uso claves o fundamentales. Cada caso de uso es especificado en detalle y realizado en términos de subsistemas, clases, y componentes.
- A medida que los casos de uso se especifican y maduran, se descubre más de la arquitectura, y esto a su vez lleva a la maduración de más casos de uso.

Este proceso continúa hasta que se considere que la arquitectura es estable.

Iterativo e Incremental

Es práctico dividir el esfuerzo de desarrollo de un proyecto de software en partes más pequeñas o mini proyectos. Cada mini proyecto es una iteración que resulta en un incremento.

Las iteraciones hacen referencia a pasos en el flujo de trabajo, y los incrementos a crecimientos en el producto.

Las iteraciones deben estar controladas. Esto significa que deben seleccionarse y ejecutarse de una forma planificada.

Los desarrolladores basan la selección de lo que implementarán en cada iteración en dos cosas: el conjunto de casos de uso que amplían la funcionalidad, y en los riesgos más importantes que deben mitigarse.

En cada iteración los desarrolladores identifican y especifican los casos de uso relevantes, crean un diseño utilizando la arquitectura seleccionada como guía, para implementar dichos casos de uso. Si la iteración cumple sus objetivos, se continúa con la próxima. Sino deben revisarse las decisiones previas y probar un nuevo enfoque.

Beneficios

- La iteración controlada reduce el riesgo a los costes de un solo incremento.
- Reduce el riesgo de retrasos en el calendario atacando los riesgos más importantes primero.
- Acelera el desarrollo. Los trabajadores trabajan de manera más eficiente al obtener resultados a corto plazo.
- Tiene un enfoque más realista al reconocer que los requisitos no pueden definirse completamente al principio.

El ciclo de vida de la Iteración Controlada es un modelo emergente y posee ventajas por su flexibilidad y modularidad, proporcionando a los desarrolladores control en sus productos.

Ciclo de Vida del Proceso Unificado

El Proceso Unificado se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema. Cada ciclo constituye una versión del sistema.

Fases

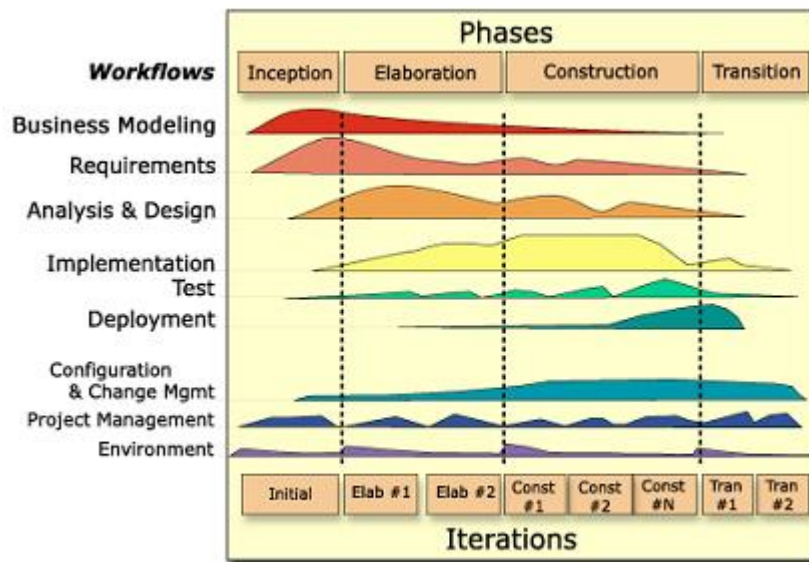
Cada ciclo consta de cuatro fases: inicio, elaboración, construcción, y transición.

Inicio: Durante la fase de inicio se desarrolla una descripción del producto final, y se presenta el análisis del negocio.

Elaboración: Durante la fase de elaboración se especifican en detalle la mayoría de los casos de uso del producto y se diseña la arquitectura.

Construcción: Durante la fase de construcción se crea el producto. La línea base de la arquitectura crece hasta convertirse en el sistema completo.

Transición: La fase de transición cubre el período durante el cual el producto se convierte en la versión beta y luego se hace el lanzamiento de este.



Fases e Iteraciones de la Metodología RUP

Cada fase se subdivide en iteraciones. En cada iteración se desarrolla en secuencia un conjunto de disciplinas o flujos de trabajos.

Disciplinas

Cada disciplina es un conjunto de actividades relacionadas (flujos de trabajo) vinculadas a un área específica dentro del proyecto total. Las más importantes son:

Requerimientos, Análisis, Diseño, Codificación, y Prueba.

Cada disciplina está asociada con un conjunto de modelos que se desarrollan. Estos modelos están compuestos por artefactos. Los artefactos más importantes son los modelos que cada disciplina realiza: modelo de casos de uso, modelo de diseño, modelo de implementación, y modelo de prueba.

El Proceso Unificado consiste en una serie de disciplinas o flujos de trabajo que van desde los requisitos hasta las pruebas. Los flujos de trabajo desarrollan modelos desde el modelo de casos de uso hasta el modelo de pruebas.

Disciplina	Modelos
Requisitos	Modelo de Casos de Uso
Análisis	Modelo de Análisis
Diseño	Modelo de Diseño - Modelo de Despliegue
Implementación	Modelo de Implementación
Prueba	Modelo de Prueba

Hitos

Cada fase finaliza con un hito. Cada hito se determina por la disponibilidad de un conjunto de artefactos, es decir un conjunto de modelos o documentos que han sido desarrollados hasta alcanzar un estado predefinido.

Al final se obtiene un conjunto de datos a partir del seguimiento del tiempo y esfuerzo consumidos en cada fase. Estos datos son útiles para las estimaciones en futuros proyectos.

Fase de Inicio

Durante la fase de inicio se desarrolla una descripción del producto final, y se presenta el análisis del negocio. Esta fase responde las siguientes preguntas:

- ¿Cuáles son las principales funciones del sistema para los usuarios más importantes?
- ¿Cómo podría ser la mejor arquitectura del sistema?
- ¿Cuál es el plan del proyecto y cuanto costará desarrollar el producto?

En esta fase se identifican y priorizan los riesgos mas importantes.

El objetivo de esta fase es ayudar al equipo de proyecto a decidir cuales son los verdaderos objetivos del proyecto. Las iteraciones exploran diferentes soluciones posibles, y diferentes arquitecturas posibles.

Puede que todo el trabajo físico realizado en esta fase sea descartado. Lo único que normalmente sobrevive a la fase de inicio es el incremento del conocimiento en el equipo.

Los artefactos que típicamente sobreviven a esta fase son:

- Un enunciado de los mayores requerimientos planteados generalmente como casos de uso.
- Un boceto inicial de la arquitectura.
- Una descripción de los objetivos del proyecto.
- Una versión muy preliminar del plan del proyecto.
- Un modelo del negocio.

La fase de inicio finaliza con el Hito de Objetivos del Ciclo de Vida.

Este hito es alcanzado cuando el equipo de proyectos y los stakeholders llegan a un acuerdo sobre:

- Cuál es el conjunto de necesidades del negocio, y que conjunto de funciones satisfacen estas necesidades.
- Una planificación preliminar de iteraciones.
- Una arquitectura preliminar.

Debe poder responderse las siguientes cuestiones:

- ¿Se ha determinado con claridad el ámbito del sistema? ¿Se ha determinado lo que va a estar dentro del sistema y fuera del sistema?
- ¿Se ha llegado a un acuerdo con todas las personas involucradas (stakeholders) sobre los requisitos funcionales del sistema?
- ¿Se vislumbra una arquitectura que pueda soportar estas características?
- ¿Se identifican los riesgos críticos? ¿Se prevé forma de mitigarlos?
- ¿El uso del producto justifica la relación costo-beneficio?
- ¿Es factible para su organización llevar adelante el proyecto?
- ¿Están los inversores de acuerdo con los objetivos?

Fase de Elaboración

Durante la fase de elaboración se especifican en detalle la mayoría de los casos de uso del producto y se diseña la arquitectura.

Las iteraciones en la fase de elaboración:

- Establecen una firme comprensión del problema a solucionar.
- Establece la fundación arquitectural para el software.
- Establece un plan detallado para las siguientes iteraciones.
- Elimina los mayores riesgos.

El resultado de esta fase es la línea base de la arquitectura.

En esta fase se construyen típicamente los siguientes artefactos:

- El cuerpo básico del sw en la forma de un prototipo arquitectural.
- Casos de prueba
- La mayoría de los casos de uso (80%) que describen la funcionalidad del sistema.
- Un plan detallado para las siguientes iteraciones.

La fase de elaboración finaliza con el hito de la Arquitectura del Ciclo de Vida.

Este hito se alcanza cuando el equipo de desarrollo y los stakeholders llegan a un acuerdo sobre:

- Los casos de uso que describen la funcionalidad del sistema.
- La línea base de la arquitectura
- Los mayores riesgos han sido mitigados
- El plan del proyecto

Al alcanzar este hito debe poder responderse a preguntas como:

- ¿Se ha creado una línea base de la arquitectura? ¿Es adaptable y robusta? ¿Puede evolucionar?
- ¿Se han identificado y mitigado los riesgos más graves?
- ¿Se ha desarrollado un plan del proyecto hasta el nivel necesario para respaldar una agenda, costes, y calidad realistas?
- ¿Proporciona el proyecto, una adecuada recuperación de la inversión?
- ¿Se ha obtenido la aprobación de los inversores?

Fase de Construcción

Durante la fase de construcción se crea el producto. La línea base de la arquitectura crece hasta convertirse en el sistema completo.

Al final de esta fase, el producto contiene todos los casos de uso implementados, sin embargo puede que no este libre de defectos.

Los artefactos producidos durante esta fase son:

- El sistema software
- Los casos de prueba
- Los manuales de usuario

La fase de construcción finaliza con el hito de Capacidad Operativa Inicial.

Este hito se alcanza cuando el equipo de desarrollo y los stakeholders llegan a un acuerdo sobre:

- El producto es estable para ser usado
- El producto provee alguna funcionalidad de valor
- Todas las partes están listas para comenzar la transición

Fase de Transición

La fase de transición cubre el período durante el cual el producto se convierte en la versión beta.

Las iteraciones en esta fase continúan agregando características al sw. Sin embargo las características se agregan a un sistema que el usuario se encuentra utilizando activamente.

Los artefactos construidos en esta fase son los mismos que en la fase de construcción. El equipo se encuentra ocupado fundamentalmente en corregir y extender la funcionalidad del sistema desarrollado en la fase anterior.

La fase de transición finaliza con el hito de Lanzamiento del Producto.

Este hito se alcanza cuando el equipo de desarrollo y los stakeholders llegan a un acuerdo sobre:

- Se han alcanzado los objetivos fijados en la fase de Inicio.
- El usuario está satisfecho.

El Proceso Unificado utiliza el Lenguaje Unificado de Modelado (UML) para la construcción de artefactos que materializan el producto software.

Unified Modeling Language (UML)

El Lenguaje de Modelamiento Unificado (UML - Unified Modeling Language) es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables.

UML es una herramienta usada en la descripción y construcción de software reconocida por la industria como estándar, mientras que el RUP es una metodología de IBM Rational para el desarrollo y construcción de software basado íntegramente en UML como soporte a la metodología.

Capability Maturity Model Integration (CMMI)

CMMI es una fusión de modelos de mejora de procesos para ingeniería de sistemas, ingeniería del software, desarrollo de productos integrados y adquisición del software.

CMMI fue creado por el SEI (instituto de ingeniería del software) con el fin de unir la gran cantidad de modelos creados por el SEI y otras organizaciones a lo largo de los años.

Es uno de los modelos más utilizados en la industria del software, disminuye o elimina el retrabajo, aumenta la fiabilidad en la predicción de costos, aumenta el reuso de productos y procesos, disminuye costos debido a múltiples evaluaciones y programas de mejoras de proceso.

Los procesos para desarrollar software a gran escala, pueden ser muy grandes y complejos. Pueden ser difíciles de definir e incluso mucho más difíciles de introducir. Por esta razón fue que el Software Engineer Institute (SEI) de la universidad Carnegie-Mellon desarrolló un ambiente de trabajo (framework) de madurez de procesos de software. Este framework es una forma ordenada para las organizaciones de determinar las capacidades de sus procesos actuales y establecer prioridades en su mejora. Se hace a través del establecimiento de 5 niveles de prioridad progresivos logrando procesos de capacidades más maduros. Por cada nivel se han definido los principios elementales o KPAs (Key Process Areas) que proveen las metas y ejemplifican las prácticas a llevar a cabo.

CMM (Capability Maturity Model) ha sido revisado y refinado por varios especialistas y representan el mejor juicio actual y el método más efectivo para conseguir los objetivos de cada nivel de madurez.

PSP: Proceso personal de software

El proceso personal de software (PSP, Personal Software Process) es un proceso de auto mejoramiento diseñado para ayudar a controlar, administrar y mejorar la forma en que se trabaja individualmente. Está estructurado por formularios, guías y procedimientos para desarrollar software.

Si es usado apropiadamente, brinda los datos históricos necesarios para trabajar mejor y lograr que los elementos rutinarios del trabajo sean más predecibles y eficientes.

Usando PSP, se pueden construir programas de más de 10.000 líneas de código (LOC), sin embargo, hay dos problemas típicos en los grandes programas. Primero, mientras se crece en tamaño, también lo hace el tiempo y el esfuerzo requerido. Esto puede ser un problema particular si sólo existe un ingeniero en el proyecto. Segundo, la mayoría de los ingenieros tienen problemas en la visualización de todas las facetas importantes de un programa, incluso cuando su tamaño es moderado. Existen muchos detalles e interrelaciones que deben tenerse en cuenta, muchas dependencias lógicas, interacciones en el tiempo o condiciones excepcionales. Una de las formas más poderosas de resolver estos problemas es el proceso de software del equipo (Team Software Process, TSP).

TSP: Proceso de software del equipo

Una definición acertada de un equipo es la dada por Dyer (Jean L. Dyer. "Team Research and team training: a-state-of-the-art review". Human Factors Reviews, The Human Factors Society, Inc, 1984.), donde un equipo consiste en al menos dos personas que trabajan para lograr una meta/ objetivo/ misión común, donde cada persona tiene asignado un rol específico o funciones específicas que desarrollar, y donde el completamiento de la misión requiere alguna forma de dependencia entre los miembros del equipo.

El proceso del equipo de software (TSP, Team Software Process) es un proceso que al igual que el PSP, está basado en el modelo CMM, TSP está diseñado para ayudar a controlar, administrar y mejorar la forma en que trabaja un equipo de software. Al igual que PSP, está estructurado por formularios, guías y procedimientos para desarrollar software.

Las actividades definidas para la consecución de los objetivos trazados en esta práctica empresarial, se han planteado según el proceso de producción definido por la empresa.

Estudio de Procesos de La Empresa.

- Conocimiento de la empresa.
- Recopilación bibliográfica.

Apoyo en desarrollo, ajuste y validación de opciones puntuales.

- Análisis de las opciones a desarrollar.
- Análisis de las opciones a ajustar.
- Desarrollo, ajuste y validación de opciones.
- Documentación técnica y de usuario

Apoyo en pruebas de sistemas que posee la empresa.

- Ejecución de pruebas funcionales y de operatividad.
- Documentación de resultados de las pruebas.

Apoyo en el manejo de prototipos de evaluación de facilidades de control de versiones.

- Estudio y Análisis de los prototipos de evaluación utilizados.
- Ejecución de pruebas funcionales y de operatividad.
- Documentación de resultados.

3. PRESENTACIÓN DE LA EMPRESA

3.1 Descripción de la Empresa

VISIÓN INGENIERIA, ubicada en la Carrera 34 No. 48-06 Of.401, es una empresa de consultoría en el área de bases de datos relacionales, desarrollos en Intranet e Internet, sistemas de información geográfica y procesos de documentación técnica e ingeniería de reversa.

VISIÓN INGENIERIA ofrece soluciones informáticas y de gestión tecnológica con parámetros de calidad, siguiendo estándares nacionales e internacionales, asegurando resultados con excelencia, efectividad y eficiencia mediante la asimilación, adaptación e innovación tecnológica. Prueba de esto ha sido la utilización de estándares metodológicos y comerciales mundialmente reconocidos para la elaboración de los productos entregados.

VISION INGENIERIA certificada ISO 9001:2000, está dedicada a suministrar soluciones informáticas en diferentes tipos de empresa. Cuenta con un sistema de gestión de calidad (SGC) guiado por dicha norma.

3.2 Reseña Histórica

VISION INGENIERIA nace como respuesta a la necesidad de servicios de consultoría en Sistemas de Información para Santander; se constituye como sociedad por escritura pública el 28 de Marzo de 1994, en la Notaría Sexta del Circuito de Bucaramanga, otorgándole una vigencia hasta el 6 de Enero de 2008; inscrita en la Cámara de Comercio de Bucaramanga el 13 de Abril de 1994 bajo el número 22331 del libro 9, con Matrícula Mercantil No. 05-045020-03, con identificación Tributaria No. 800.225.066-4 y conformada en la actualidad por el Ing. de Sistemas GERARDO ARMANDO GARAVITO FLOREZ y la Sra. MARIA FERNANDA RODRIGUEZ DIAZ 95% y del 05% respectivamente.

VISION INGENIERIA dentro de sus servicios, desarrolla labores de consultoría en el área de bases de datos relacionales, desarrollos en Intranet e Internet, Sistemas de información Geográfica y procesos de documentación técnica e ingeniería de reversa.

Inicia sus labores prestando servicios a ECOPETROL - ICP, extendiendo posteriormente la prestación de sus servicios a otras entidades estatales como Corporación Autónoma Regional de la Defensa de la Meseta de Bucaramanga –

CDMB, Alcaldía de Bucaramanga, Gobernación de Santander y entidades privadas o mixtas.

Durante estos últimos años, VISION INGENIERIA ha buscado consolidarse a nivel financiero y de recurso humano, manteniendo un equipo de trabajo y manejando un detallado y planeado flujo de caja.

VISIÓN INGENIERIA, hace parte de la red mundial de alianzas de ORACLE a través del Programa Oracle Partner Network como distribuidores autorizados para Licenciamiento y Educación de ORACLE, lo que ha permitido el reconocimiento como expertos en el desarrollo de Sistemas de Información basados en herramientas ORACLE.

Actualmente, se encuentra en un proceso para comercializar Sistemas de Información propios, como el Sistema de Gestión Corporativa Integral que recopila diferentes Sistemas que facilitan toda la labor organizacional tales como: Planeación de Proyectos, Contratación, Desarrollo Humano, Almacén, Sistema Inteligente de Mercadeo, Sistema de Consulta Gerencial y Sistema Financiero Integrado, con el fin de consolidarse como líder regional.

VISIÓN INGENIERIA, sigue y seguirá trabajando constante y disciplinadamente para continuar su proceso de posicionamiento en el área de informática.

3.3 Misión

Proveer soluciones informáticas y de gestión tecnológica con parámetros de Calidad, siguiendo estándares nacionales e internacionales, con un recurso humano capacitado, calificado y comprometido; garantizando resultados excelentes, efectivos y eficientes que generen valor agregado a nuestros clientes, mediante la asimilación, adaptación e innovación tecnológica, que aseguren la sostenibilidad y el desarrollo de la empresa.

3.4 Visión

Para el año 2008 VISIÓN INGENIERIA, tendrá presencia y establecimiento de sus marcas a nivel nacional e internacional, fortaleciendo su posicionamiento en estos mercados.

4 EL SISTEMA OPTIMIZADOR

Derechos de Autor y Sistema Propiedad de ECOPETROL – ICP, INSTITUTO COLOMBIANO DE PETRÓLEO.

Uno de los proyectos en los que se encuentra trabajando actualmente la empresa Visión Ltda., es la migración de la base de datos desde Ms Access a Oracle 9i R2 del Optimizador para Condiciones de Operación en Estaciones de Bombeo de la VIT - ICP (Vicepresidencia de Transporte de ECOPETROL – Instituto Colombiano del Petróleo) llamado OPTIMIZADOR.

Esta aplicación fue desarrollada en Visual Basic 6.0 y su base de datos se encuentra en Access 97. Uno de los objetivos de su mantenimiento es convertir su sistema monousuario a un ambiente Cliente –Servidor, tomando como motor de base de datos ORACLE.

Por otra parte, esta aplicación fue realizada por ingenieros químicos del ICP, por lo tanto no tienen la estructura que debe poseer un desarrollo software.

Inicialmente, los ingenieros de desarrollo de Visión encargados de este proyecto, analizaron la parte operativa de la aplicación para entender el funcionamiento de los procesos que en ella se manejan, revisaron el código para entender la manera como se programaron las diferentes opciones y el manejo que se le da a los datos. Posteriormente, elaboraron el Modelo Entidad – Relación preliminar, revisaron los usuarios de la aplicación así como los derechos que tiene cada uno de ellos dentro la misma; establecieron el Formato para el control de los cambios, realizados por los nuevos Desarrollos y el Soporte a la aplicación que se encuentra en producción, construyeron la nueva base de datos en ORACLE, realizaron la migración de datos y seguidamente se iniciaron los ajustes de la aplicación para que corriera con la nueva base de datos.

La aplicación esta compuesta por 5 módulos que contienen la configuración, parametrización, administración, optimización y ayuda.

Compuesta por 118 formularios donde el más extenso contiene 785 sentencias que acceden a la base de datos, 12 módulos y 30 módulos de clase. Para un total de 4974 sentencias por ajustar.

5. EL SISTEMA SCCPL

Derechos de Autor y Sistema Propiedad de ECOPETROL – ICP, INSTITUTO COLOMBIANO DE PETRÓLEO.

Durante el apoyo al proyecto de Consultoría para el desarrollo de nuevas funcionalidades del sistema SCCPL para el Instituto Colombiano del Petróleo de ECOPETROL S.A. en Piedecuesta Santander, se realizaron actividades de ajustes, documentación y capacitación del módulo SCCPL-CONVENIOS, el cual administra la gestión de los convenios realizados por ECOPETROL - ICP con sus socios.

No se presentan las aplicaciones en este documento para respetar los derechos de autor y propiedad de los sistemas.

6. PSP (Personal Software Process)

PSP amplía el proceso de mejora a la gente que realiza el trabajo de desarrollo de software.

PSP se concentra en las prácticas de trabajo de los ingenieros en una forma individual. El principio detrás de PSP es éste, sirve para producir software de calidad, cada ingeniero debe trabajar en la necesidad de realizar trabajo de calidad. PSP se diseñó para ayudar a profesionales del software para que utilicen constantemente prácticas sanas de ingeniería de software.

Así mismo les enseña a cómo planear y darle un seguimiento a su trabajo, a utilizar un proceso bien definido y medido, a establecer metas medibles, y finalmente a la utilización del rastreo constante para alcanzar dichas metas. PSP les demuestra a los ingenieros a cómo manejar la calidad desde el principio del trabajo, a cómo analizar los resultados de cada trabajo, y a cómo utilizar los resultados para mejorar el proceso del proyecto siguiente.

Principios del PSP

El diseño de PSP se basa en los siguientes principios de planeación y de calidad

- Cada ingeniero es esencialmente diferente; para ser más precisos, los ingenieros deben planear su trabajo y basar sus planes en sus propios datos personales.
- Para mejorar constantemente su funcionamiento, los ingenieros deben utilizar personalmente procesos bien definidos y medidos.

- Para desarrollar productos de calidad, los ingenieros deben sentirse personalmente comprometidos con la calidad de sus productos.
- Cuesta menos encontrar y arreglar errores en la etapa inicial del proyecto que encontrarlos en las etapas subsecuentes.
- Es más eficiente prevenir defectos que encontrarlos y arreglarlos.
- La manera correcta de hacer las cosas es siempre la manera más rápida y más barata de hacer un trabajo.

Para hacer un trabajo de ingeniería de software de la manera correcta, los ingenieros deben planear de la mejor manera su trabajo antes de comenzar y deben utilizar un proceso bien definido para realizar de la mejor manera la planeación del trabajo.

Para que los desarrolladores lleguen a entender su funcionamiento de manera personal, deben medir el tiempo que pasan en cada proceso, los defectos que inyectan y remueven de cada proyecto y finalmente medir los diferentes tamaños de los productos que llegan a producir.

Para producir constantemente productos de calidad, los ingenieros deben planear, medir y rastrear constantemente la calidad del producto y deben centrarse en la calidad desde el principio de un trabajo.

Finalmente, deben analizar los resultados de cada trabajo y utilizar estos resultados para mejorar sus procesos personales.

Niveles del PSP

PSP tiene un marco de proceso de evolución similar al que tiene el Capability Maturity Model (CMM) o Modelo de Capacidad de Madurez. PSP trata parcialmente 12 de las 18 KPA's definidas en el CMM.

Las KPA's son las áreas de procesos clave o Key Process Áreas por su significado en inglés, estas áreas ayudan a guiar a los programadores a que exista un mejoramiento notable en el proceso de software.

En CMM un nivel de madurez sólo se alcanza si se logran cumplir todas las KPA's que exige cada nivel. Sin embargo PSP solamente cubre de manera parcial estas KPA's debido a que es un complemento de CMM y no depende uno del otro en ningún sentido por lo que es considerado como material de apoyo.

Como se ha visto anteriormente el Instituto de la Ingeniería del Software (SEI) ha desarrollado el proceso personal del software para definir y reparar la holgura que existe entre el modelo de la madurez de la capacidad y el individuo. Por lo tanto es ideal utilizarlo junto con CMM pero no es obligatorio ya que es un proceso y no un modelo como lo es CMM.

Para desarrollar software de alta calidad, cada componente individual también debe de contar con la más alta calidad posible. La estrategia total de PSP es cerciorarse de que todos los componentes individuales se desarrollen con la más alta calidad. PSP logra esto proporcionando un marco de proceso personal ya definido que el programador puede utilizar. Este marco es:

- Desarrollar un plan para cada proyecto y/o componente.
- Registrar su tiempo de desarrollo.
- Registrar sus defectos
- Conservar sus datos en informes del proyecto
- Utilizar sus datos para planear los proyectos y/o los componentes futuros.
- Analizar sus datos para desarrollar sus procesos con más calidad para mejorar su funcionamiento.

El proceso personal de software fue diseñado para ayudar y guiar a los ingenieros de software a realizar bien su trabajo y haciendo esto pueden llegar a cubrir las KPA's requeridas. PSP también muestra como aplicar métodos avanzados de ingeniería a sus proyectos y/o deberes diarios. Asimismo provee métodos de estimación y de planeación muy bien detallados que son necesarios para dar un seguimiento a su trabajo.

La disciplina del PSP provee un marco estructurado para desarrollar habilidades personales y métodos que se necesitarán más adelante para ir forjando al ingeniero de software. Es importante que la calidad del software desarrollado abarque hasta el más mínimo detalle, por muy pequeño que éste sea, ya que si no se hace así, puede dañar el sistema entero.

Es importante para las personas o empresas que quieran implementar PSP saber que deben de cumplir con todas las KPA's para que avancen de la mejor manera posible al siguiente nivel.

Cabe recalcar que se puede "personalizar" el proceso agregando o removiendo tareas conforme a las exigencias de cada persona o empresa. Esto quiere decir que por lo mismo de que PSP es un proceso y no un modelo, se puede amoldar a las necesidades del programador.

Ahora bien que si lo que se quiere hacer es que el proceso de PSP sea usado para cumplir con el modelo de capacidad de madurez, entonces se deben de tomar en cuenta los puntos que exige CMM de PSP.

En cada nivel de CMM existen KPA's que pueden ser cumplidas siguiendo el proceso personal de software, de hecho esa sería la mejor manera de cumplir con las exigencias de CMM.

Existen resultados que demuestran la eficiencia y una reducción de tiempo considerable cuando se usan estos procesos diseñados para alcanzar los niveles de CMM más rápidamente que si se utilizaran otros "camino".

PSP se trabaja en la actualidad con la versión actualizada más reciente de CMM: CMMI (Capability and Maturity Model Integration), sucesor del modelo original CMM.

Hasta ahora se ha visto que PSP tiene sus niveles internos y que también cuenta con niveles que se deben cumplir junto a su modelo por el cual fue creado, CMM. El programador y el proyecto determinan si se debe utilizar PSP en su totalidad o sólo parcialmente.

También se ha visto la gran dinamicidad que puede llegar a tener PSP con respecto a cualquier proyecto que esté en desarrollo. Desgraciadamente éste atributo no es bien aprovechado o en dado caso no se hace de manera correcta la decisión sobre que puntos tomar de PSP y cuales no, si conviene hacer uso de todos los puntos que sugiere el proceso o solamente unos cuantos.

PSP 0 Y PSP 0.1

Esta última fase es en la que se compara el desempeño final con el desempeño planeado; también se registran los datos que tienen que ver con el proceso para que finalmente los datos finales se pongan en el reporte final.

Todo estas fases están diseñadas de esta manera porque el objetivo principal del nivel inicial de PSP (PSP 0) es el de proveer un marco bien definido.

Para realizar la recolección de todos los datos que intervienen en el proyecto es muy importante este primer paso ya que proporciona un panorama general de qué métodos utilizar para "atacar" el problema. Es de gran importancia que se entienda claramente este concepto porque representa los fundamentos de los demás niveles.

Dado que PSP tiene un número de métodos que generalmente los ingenieros no practican, estos métodos se introducen en una serie de siete versiones del proceso completo. Estas versiones se etiquetan desde PSP 0 hasta PSP 3, y cada versión tiene un sistema similar de registros, de formatos, scripts, y estándares. Los scripts que intervienen en el proceso definen los pasos para cada parte del proceso.

Los registros y los formatos proporcionan las plantillas para almacenar los datos y los estándares ayudan a dirigir a los ingenieros durante todo el proyecto que desarrollan, desde el principio hasta el final de éste.

Sin embargo también hay que notar que si el deseo del usuario es el de adaptar este nivel a los hábitos de programación personales, entonces éste tiene que modificar los scripts, formatos y registros para que se adapten a sus necesidades.

PSP 0 provee una estructura bien definida para tareas pequeñas y la medición de éstas, posteriormente se puede optar por continuar a los niveles siguientes. Como ya se ha visto el primer paso en el proceso de PSP es la planeación. Para este paso existe un script de planeación que ayuda a dirigir este trabajo en todos sus niveles y también es necesario llevar un resumen del plan para registrar todos los datos de esta fase de planeación.

De esta forma los ingenieros registran cada dato que interviene en el desarrollo del programa, es decir, registran su tiempo en cada proceso y los datos de los defectos que se encuentran durante todo el proceso de desarrollo e inclusive desde la planeación. Al final del trabajo, durante la fase de postmortem (P.M.), miden el tamaño del programa, e incorporan estos datos en el formato del resumen del proyecto.

Cuando todos estos formatos están hechos, entonces los ingenieros entregan el producto acabado junto con el formulario lleno del resumen del proyecto.

Requisitos

Todas las fases que se emplean en PSP requieren de un script para cada etapa del ciclo de vida de cualquier programa que se desarrolla.

Estas etapas, que sólo el Proceso Personal de Software (PSP), utiliza son:

1. Planeación.
2. Desarrollo.
3. Postmortem.

Todas estas etapas forman parte de un proceso integral y a su vez estas etapas tienen sus requerimientos que le exigen al programador. De manera general todo se engloba en el script del proceso que ayuda al programador a desarrollar programas modulares y que provee de un panorama general de las etapas de planeación, desarrollo y postmortem.

El script de planeación tiene como entrada los requerimientos del programa así como la estimación de los recursos que se emplean en éste. El script de desarrollo le exige al programador los requerimientos de cada fase de desarrollo que toma en cuenta PSP (Diseño, Codificación, Compilación y Pruebas) para que al final se cuente con un programa bien probado y a prueba de errores. El script de Postmortem pide al programador todos los defectos que se encontraron durante la realización del proyecto y también exige el tiempo final que se utilizó durante la realización del programa, esto para que exista un historial final que ayude a tener un margen de comparación para utilizarlo posteriormente y evitar caer en los mismos errores.

Como en todo proceso de ingeniería de software, los ingenieros comienzan planeando el trabajo con el mayor detalle posible y algunas veces los ingenieros solamente cuentan con una declaración única de los requisitos del proyecto. Si es así, entonces dicha declaración debe ser la base para el plan original.

La exactitud, la estimación y el plan son influenciados en su mayoría por el conocimiento general que tienen los ingenieros respecto al trabajo que se hará. El diseño conceptual comienza cuando los ingenieros primero definen cómo el

producto debe ser diseñado para que después éste sea construido, sin embargo, en esta etapa del proceso es muy temprano realizar un diseño del producto completo desde la fase del planeamiento.

Los desarrolladores tienen que estar concientes de que deben realizar un diseño conceptual que se adapte lo más fácilmente posible al proyecto. Ésta conjetura es esencial porque cada vez que los ingenieros tengan dudas respecto al proyecto, bastaría mirar este diseño para tener bien plantadas las bases de lo que se está creando. Más adelante, durante la fase del diseño, los ingenieros examinan las alternativas del diseño y al final todas esas variantes finales producen lo que se conoce como el diseño del producto completo.

La estimación del tamaño y de los recursos del producto debe ser moderada por los equipos o individuos que intervienen en el desarrollo del proyecto. Sin embargo, para los ingenieros que desarrollan software de manera individual, esta correlación tiene resultados generalmente altos.

Por lo tanto, PSP comienza a estimar los tamaños de los productos que los ingenieros desarrollan personalmente. PSP se basa en el tamaño y en los datos de la productividad de cada ingeniero y con estos datos estima el tiempo requerido para hacer el trabajo.

PSP realiza las estimaciones, tanto del tamaño del programa como de los recursos del mismo, con un método que se creó para estos fines y que tiene por nombre PROBE (PROxy Based Estimating) por sus siglas en inglés, traducido al español se entiende como Estimación basada en la evaluación. La evaluación se aplica a todos los objetos que se encuentran en el diseño conceptual.

La metodología de PROBE consiste en que los ingenieros deben determinar primero los objetos que se requieren para construir el producto descrito en el diseño conceptual. Después se determina el tipo probable de los métodos que se emplean en el programa y el número de métodos que cada objeto necesita. También se puede hacer uso de referencias de datos históricos sobre los tamaños de objetos similares que se han desarrollado previamente y que al mismo tiempo utilizan el cálculo de la regresión lineal para determinar el tamaño total del producto acabado.

Puesto que el tamaño del objeto está en función del estilo de programación, el método PROBE demuestra a los ingenieros cómo se deben utilizar los datos sobre los programas que ellos han desarrollado personalmente para que después se generen los distintos tipos del tamaño de cada programa y este dato esté presente en todo momento para su uso personal cada vez que se desee. Una vez que se

estiman los tamaños de los objetos, se utiliza la regresión lineal para estimar la cantidad total de código o (LOC) Lines Of Code por sus siglas en inglés, entiéndase como líneas de código fuente, que planean desarrollar.

Para utilizar la regresión lineal, los ingenieros deben realizar una comparación de los datos históricos contra el resultado estimado del tamaño del programa actual, esta comparación se debe hacer por lo menos con tres programas anteriores.

El método PROBE también utiliza la regresión lineal para estimar los recursos que se emplea en el desarrollo completo.

Una vez más esta estimación se basa en el tamaño estimado del programa contra los datos reales del esfuerzo con por lo menos tres proyectos anteriores. Los datos deben demostrar una correlación razonable entre el tamaño del programa y el tiempo de desarrollo, PSP requiere que el resultado de esta correlación sea de por lo menos 0.5.

Una vez que se ha estimado el tiempo total que se empleará para el trabajo, los ingenieros deben apoyarse en sus datos históricos para estimar el tiempo necesario que cada fase del trabajo tomará. Por medio de los porcentajes que se obtienen en el campo del formato de registro de tiempo, los ingenieros tienen que asignar su tiempo de desarrollo total estimado a las fases de planeamiento, diseño, revisión de diseño, código, revisión de código, compilación, pruebas y finalmente post-mortem.

Cuando estos porcentajes han sido calculados, los ingenieros ahora cuentan con una estimación más real para el tamaño del programa, el tiempo de desarrollo total y el tiempo requerido para cada fase del desarrollo.

Una vez que los ingenieros sepan el tiempo requerido para cada fase del proceso, deben estimar el tiempo que le dedicarán al trabajo cada día o cada semana, según sus preferencias.

Con esa información, los ingenieros distribuyen su tiempo que le dedicarán a cada tarea y al mismo tiempo le asignan las horas de programación que tienen disponibles para cada programa.

Finalmente en la fase de desarrollo del producto, los ingenieros no deben hacer más que el trabajo de programación.

Mientras que este trabajo normalmente no se considera parte del proceso de planeamiento, los ingenieros deben utilizar los datos que arroje este proceso para hacer los planes futuros y que tienen mayor importancia.

Después de terminar cada trabajo, los ingenieros deben hacer un análisis post-mortem del trabajo. En el post-mortem, los desarrolladores actualizan el resumen del plan del proyecto con datos reales, también calculan cualquier dato que se calcule en el desempeño final y al final repasan qué tan bien realizaron su proyecto contra el plan definido desde el principio del proyecto.

Como paso final del planeamiento, los ingenieros deben actualizar sus bases de datos históricas con los datos del tamaño y de la productividad final alcanzada. Durante esta fase, los ingenieros también examinan cualquier posibilidad de mejora que pueda tener el proceso y de ser necesario le hacen los ajustes necesarios al proceso completo o solamente a una parte de éste.

Los desarrolladores también tienen que analizar los defectos encontrados en la fase de compilación y en la fase de pruebas para que actualicen sus listas de comprobación personales que surgen de las revisiones hechas en cada proceso para que al final esto les facilite encontrar y arreglar defectos similares que se encuentren en proyectos futuros.

Lo anteriormente mencionado depende de la habilidad individual de los ingenieros o grupal en caso de que se trate de un proyecto grande.

La clave para aprender la metodología del proceso personal de software (PSP), está en leer y analizar cada uno de los datos del trabajo que se desarrolla y lo que en realidad estos datos “dicen” acerca del desempeño personal de cada programador.

También interviene el método de estimación PROBE y muestra los distintos recursos que exigen cada una de las fases desde el diseño conceptual hasta el producto final.

7. TSP (Team Software Process)

7.1 ¿Qué es TSP?

- TSP sirve para construir y guiar equipos interdisciplinarios
- Proporciona
 - un proceso definido para construir el equipo
 - un marco para el trabajo en equipo
 - un ambiente de gestión para soportarlo
- Diseñado para equipos de desarrollo y mantenimiento de entre 2 a 20 ingenieros
- Incluye
 - un proceso completamente definido para el trabajo en equipo
 - roles definidos para los miembros del equipo
 - un proceso estructurado para el lanzamiento y seguimiento
 - una herramienta para soportar el trabajo del equipo y del ingeniero

7.2 ROLES

- Los roles distribuyen la gestión entre los ingenieros
- estos roles definen las responsabilidades para gestionar el entorno de trabajo
- los miembros eligen sus roles durante el lanzamiento del equipo
- los roles estándar cubren:
 - planificación
 - proceso
 - calidad
 - soporte
 - interfaz con el usuario
 - diseño
 - implementación
 - prueba

7.3 PROCESO

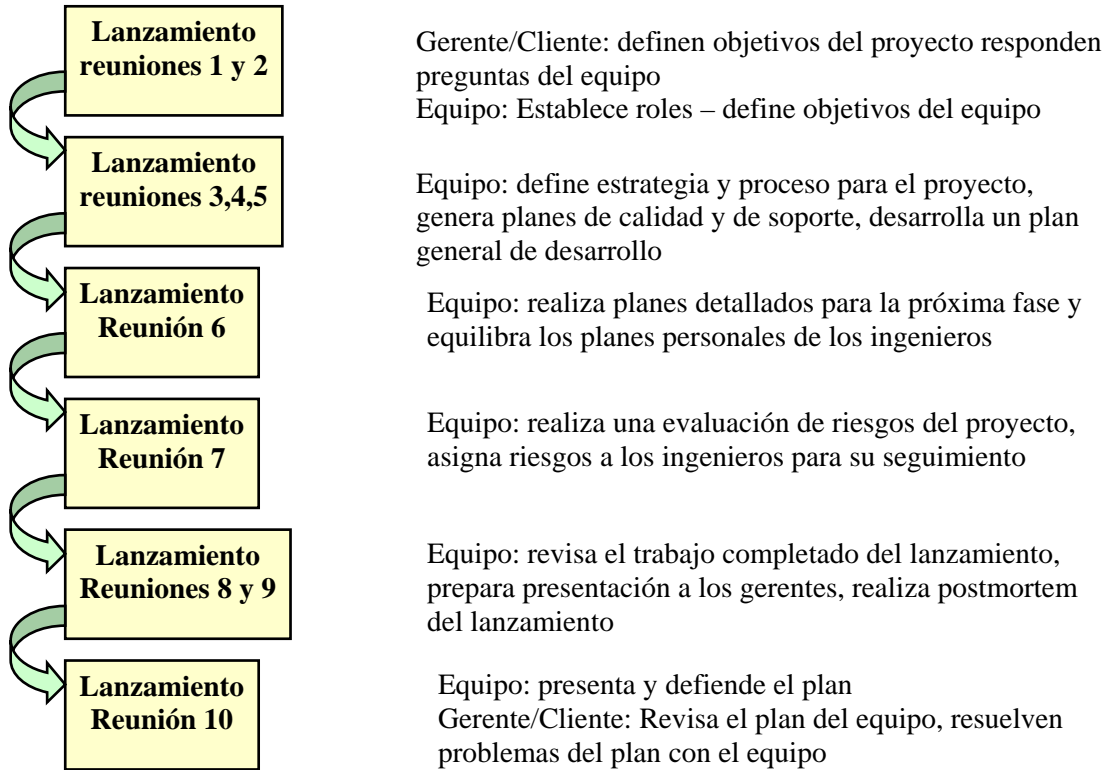


Figura 1- Proceso del PSP

7.4 PLANIFICACION

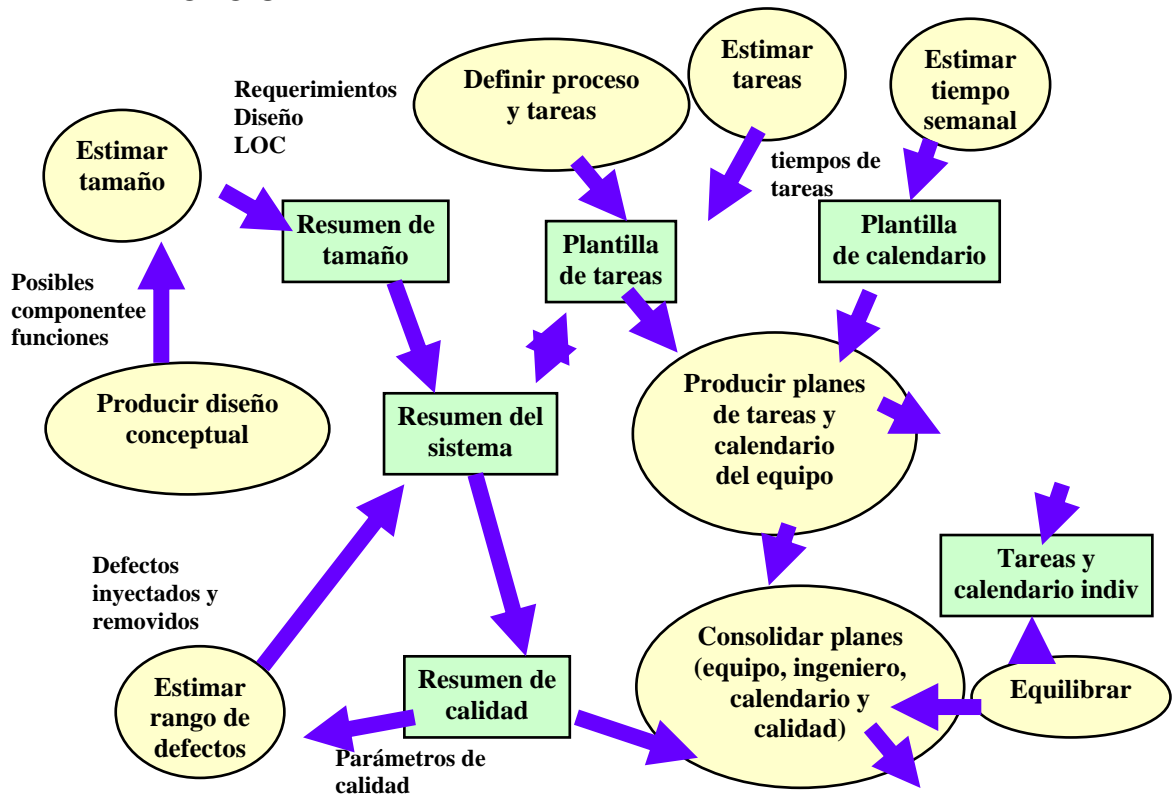


Figura 2- Planificación del PSP

7.5 SEGUIMIENTO

- los planes del equipo e individuales son la base para un adecuado seguimiento
- los miembros del equipo regularmente reevalúan los riesgos y consideran formas de mitigarlos
- en reuniones semanales los ingenieros
 - informan estado de las tareas
 - revisan los riesgos clave
 - re-equilibran la carga de trabajo
 - el equipo produce informes semanales precisos del estado para la gerencia

7.6 BENEFICIOS (según Humphrey)

- equipos consistentemente cumplen los compromisos
 - un equipo terminó 6 semanas antes
 - un equipo terminó en fecha pero incluyó 25% más de funcionalidad que lo planificado
 - equipos más productivos
 - equipos duplicaron su productividad
 - defectos en el producto reducidos sustancialmente
 - proyecto sin defectos reportados en el 1er.año
 - mejores diseños
 - en un proyecto no se detectaron defectos de diseño en pruebas
 - en otro proyecto el diseño fue seleccionado para una nueva familia de productos
- Beneficios de TSP (según Humphrey)
- en el personal
 - a los ingenieros les gusta trabajar con este enfoque
 - reducción en la rotación del personal

7.7. APLICACIÓN DEL PSP.

En cuanto a la aplicación de PSP como propuesta para la Empresa la siguiente plantilla de registro de tiempo que propone el PSP, podemos notar que información plantea PSP como importante para manejar el proceso de desarrollo de un producto, aterrizándola bajo el esquema PHVA que maneja la empresa en su sistema de gestión de la calidad.

Formato del Registro de Tiempo

Estudiante	_____	Fecha	_____
Instructor	_____	Programa #	_____

Fecha	Inicio	Trmino	Tiempo de Interrupción	Tiempo Delta	Fase	Comentarios

Figura 3- Formato de registro de tiempos

Podemos notar que la platilla esta incluida casi en su totalidad en el sistema de gestión de la calidad que maneja la empresa, sistema del cual se extrae la información que se muestra a continuación como parte del seguimiento de actividades del practicante dentro de la empresa.

8. PHVA

El mantenimiento y mejora continua del desempeño de los procesos del S.G.C. Se puede lograr aplicando el concepto del **ciclo PHVA** en todos los niveles dentro de la organización. **El ciclo PHVA** (Planificar, **H**acer, **V**erificar y **A**ctuar) esta íntimamente asociado con la planificación, implementación, control y mejora continua de todos los procesos de la organización.

La norma ISO 9001 se encuentra estructurada bajo el esquema del ciclo de mejoramiento el cual consiste en PLANEAR las actividades de cada proceso o las tareas que realizamos en el día a día, HACER o ejecutar los procesos, VERIFICAR que los procesos los estamos haciendo de acuerdo a lo planeado, en este punto se pueden tomar mediciones o hacer seguimientos y por último ACTUAR es decir cuando no cumplimos lo planeado o no ejecutamos las tareas de una forma adecuada se debe tomar las acciones pertinentes para dar solución o mejorar con respecto a las falencia encontradas.

Según la norma ISO 9001:2000 se establecen las siguientes definiciones para estos términos:

PLANEAR	Establecer los objetivos y procesos necesarios para conseguir resultados De acuerdo con los requisitos del cliente y las políticas de la organización.
HACER	Hacer lo planificado. (Implementar los procesos)
VERIFICAR	Realizar el seguimiento y la medición de los procesos y productos respecto a las políticas, los objetivos y los requisitos del producto, e informar sobre los resultados.
ACTUAR	Si la verificación arroja resultados que indiquen que hay que llevar a cabo acciones, se toman las acciones para mejorar continuamente el desempeño de los procesos.

Tabla 1- Fases del PHVA

9. DETALLADO DE ACTIVIDADES

En la siguiente tabla se muestra el detallado de actividades desarrolladas por el practicante en la empresa Visión Ingeniería

TAREA	INICIO	HORAS	TIPO_TAREA
Ajustes a la forma frmEliSistemas	23/01/2006 8:45	3.5006	CORRECCIONES
Validación de ajustes a la forma frmEliSistemas	23/01/2006 14:45	.7501	CORRECCIONES
Ajustes a la forma frmEliSistemas	24/01/2006 8:30	4.0006	DESARROLLO
Depuración y Pruebas a la forma frmEliSistemas	24/01/2006 14:30	3.0005	DESARROLLO
Revisión de la forma frmConfiguradorSistemas	30/01/2006 9:30	.25	DESARROLLO
Ajustes a la función EliminarSistema en la forma frmConfiguradorSistemas	01/02/2006 8:20	.9168	PRUEBAS
Ajustes a la función activarydesactivarestaciones en la forma frmConfiguradorSistemas	01/02/2006 14:45	.7501	DESARROLLO
Ajustes a la función activarydesactivarestaciones en la forma frmConfiguradorSistemas	01/02/2006 16:00	.3334	DESARROLLO
Ajustes a la función asactual_click en la forma frmConfiguradorSistemas	03/02/2006 14:35	1.8336	DESARROLLO
Ajustes a la función descargarycargar en la forma frmConfiguradorSistemas	07/02/2006 9:22	2.3004	DESARROLLO
Ajustes a la función actualizarestacion en la forma frmConfiguradorSistemas	08/02/2006 10:00	2.5004	DESARROLLO
Ajustes a la función agregarestacion en la forma frmConfiguradorSistemas	08/02/2006 14:45	2.7504	DESARROLLO
Ajustes a la función agregarestacionbombas en la forma frmConfiguradorSistemas	08/02/2006 17:30	1.0002	PRUEBAS
Ajustes a la función agregarestacionbombas en la forma frmConfiguradorSistemas	10/02/2006 8:45	.7501	ANALISIS-DISEÑO
Ajustes a la función cargararchivo en	10/02/2006	3.0005	DESARROLLO

TAREA	INICIO	HORAS	TIPO_TAREA
la forma frmConfiguradorSistemas	9:30		
Ajustes a la función cargarestacion en la forma frmConfiguradorSistemas	10/02/2006 14:45	2.7504	DESARROLLO
Ajustes a la función cargartrayectoconfigurado en la forma frmConfiguradorSistemas	10/02/2006 17:30	1.0002	PRUEBAS
Ajustes a la función cargartrayectoconfigurado en la forma frmConfiguradorSistemas	13/02/2006 8:45	3.7506	PRUEBAS
Ajustes a la función compilacionestacion en la forma frmConfiguradorSistemas	13/02/2006 14:34	.4334	CORRECCIONES
Ajustes a la función generaciondearchivo en la forma frmConfiguradorSistemas	13/02/2006 15:00	3.5006	PRUEBAS
Ajustes a la función generaciontablasesquemas en la forma frmConfiguradorSistemas	14/02/2006 8:34	2.4337	CORRECCIONES
Ajustes a la función generaciontablasrepartos en la forma frmConfiguradorSistemas	14/02/2006 11:00	1.5002	PRUEBAS
Ajustes a la función generaciontablasesquemasniveles en la forma frmConfiguradorSistemas	14/02/2006 14:45	.7501	PRUEBAS
Ajustes a la función niveles en la forma frmConfiguradorSistemas	15/02/2006 8:30	4.0006	CORRECCIONES
Ajustes a la función prod cargado en la forma frmConfiguradorSistemas	16/02/2006 9:00	3.5006	DESARROLLO
Ajustes a la función rutinadegesquemas en la forma frmConfiguradorSistemas	16/02/2006 14:45	1.2502	PRUEBAS
Ajustes a la función rutinadegesquemas en la forma frmConfiguradorSistemas	17/02/2006 14:50	1.1669	DESARROLLO
Ajustes a la función rutinadedespacho en la forma frmConfiguradorSistemas	17/02/2006 16:05	.5834	PRUEBAS
Ajustes a la función rutinadedespacho en la forma frmConfiguradorSistemas	20/02/2006 9:00	3.5006	DESARROLLO
Ajustes a la función	20/02/2006	.7501	PRUEBAS

TAREA	INICIO	HORAS	TIPO_TAREA
generartablascorridadinamica en la forma frmConfiguradorSistemas	14:45		
Ajustes a la función generartablascorridadinamica en la forma frmConfiguradorSistemas	21/02/2006 14:50	2.9171	CORRECCIONES
Depuración a la forma frmEliSistemas	22/02/2006 14:49	3.6839	DESARROLLO
Validación y pruebas a la forma frmEliSistemas	23/02/2006 9:00	3.5006	ANALISIS-DISEÑO
Ajustes a la forma frmEliSistemas	23/02/2006 14:30	4.0006	ANALISIS-DISEÑO
Depuración a la forma frmConfiguradorSistemas	24/02/2006 9:00	3.5006	ANALISIS-DISEÑO
Validación y Pruebas a la forma frmConfiguradorSistemas	24/02/2006 14:49	3.6006	DESARROLLO
Ajustes a la forma frmConfiguradorSistemas	27/02/2006 9:00	1.0002	PRUEBAS
Validación y Pruebas a la forma frmConfiguradorSistemas	27/02/2006 10:10	2.4171	DESARROLLO
Ajustes a la forma frmConfiguradorSistemas	27/02/2006 14:30	2.0837	DESARROLLO
Ajustes a la forma frmConfiguradorSistemas	28/02/2006 8:15	4.1673	DESARROLLO
Validación y Pruebas a la forma frmConfiguradorSistemas	28/02/2006 14:45	3.5839	ANALISIS-DISEÑO
Actualización de formas básicas para la configuración de sistemas	01/03/2006 08:10	3.0838	DESARROLLO
Actualización de formas específicas de configuración de sistemas	01/03/2006 11:30	.7501	DESARROLLO
Validación de las formas de configuración de sistemas	01/03/2006 12:15	.3334	PRUEBAS
Análisis del funcionamiento de las formas anexas de configuración de sistemas	01/03/2006 14:30	1.8003	ANALISIS-DISEÑO
Ajustes a la forma de configuración de sistemas	01/03/2006 16:18	.4501	DESARROLLO
Ajustes a formas anexas: frmRenSistema	01/03/2006 16:45	.15	DESARROLLO
Ajustes a formas anexas: frmConfigurarEstaciones	01/03/2006 16:55	.4167	DESARROLLO
Análisis del funcionamiento de las	01/03/2006	1.0002	ANALISIS-

TAREA	INICIO	HORAS	TIPO_TAREA
formas anexas referentes a bombas reciprocantes	17:30		DISEÑO
Análisis del funcionamiento de las formas anexas referentes a bombas centrifugas	02/03/2006 08:20	.6668	ANALISIS-DISEÑO
Análisis del funcionamiento de las formas anexas referentes a filtoturbinas	02/03/2006 09:00	.3334	ANALISIS-DISEÑO
Análisis del funcionamiento de las formas anexas referentes a filtros	02/03/2006 09:20	.0833	ANALISIS-DISEÑO
Ajustes a formas anexas	02/03/2006 09:25	3.0838	DESARROLLO
Ajustes a formas anexas	02/03/2006 14:30	1.7503	DESARROLLO
Análisis del funcionamiento de las formas anexas referentes a resistencias	02/03/2006 16:30	2.0003	ANALISIS-DISEÑO
Análisis del funcionamiento de las formas anexas referentes a bypasses	03/03/2006 08:15	4.2507	ANALISIS-DISEÑO
Ajustes a formas anexas	03/03/2006 14:30	4.0006	DESARROLLO
Análisis del funcionamiento de las formas anexas referentes a tramos	04/03/2006 08:30	4.0006	ANALISIS-DISEÑO
Análisis del funcionamiento de las formas anexas referentes a topografía	06/03/2006 08:10	.3334	ANALISIS-DISEÑO
Ajustes a formas anexas	06/03/2006 08:45	3.7506	DESARROLLO
Ajustes a formas anexas	06/03/2006 14:30	4.0006	DESARROLLO
Análisis del funcionamiento de las formas anexas referentes a sistema	07/03/2006 08:10	.3334	ANALISIS-DISEÑO
Ajustes a formas anexas referentes a sistema	07/03/2006 08:30	2.1837	DESARROLLO
Ajustes a formas anexas referentes a sistema	07/03/2006 10:41	.2242	DESARROLLO
Ajustes a formas anexas referentes a sistema	07/03/2006 10:55	1.5836	DESARROLLO
Validación y prueba de formas anexas	07/03/2006 14:30	1.0002	PRUEBAS
Análisis del funcionamiento de las	07/03/2006	3.0005	ANALISIS-

TAREA	INICIO	HORAS	TIPO_TAREA
formas anexas: frmDefyConfTramo	15:30		DISEÑO
Ajustes a formas anexas: frmDefyConfTramo	08/03/2006 08:00	2.0003	DESARROLLO
Corrección a formas anexas: frmDefyConfTramo	08/03/2006 10:00	.7501	CORRECCIONES
Análisis del funcionamiento de las formas anexas: frmdesplieguepccd	08/03/2006 10:45	.5001	ANALISIS- DISEÑO
Ajustes a formas anexas: frmdesplieguepccd	08/03/2006 11:15	1.2502	DESARROLLO
Análisis del funcionamiento de las formas anexas: frmentregascd	08/03/2006 14:45	2.5837	ANALISIS- DISEÑO
Análisis del funcionamiento de las formas anexas: frmEsquemasSistemas	08/03/2006 17:20	1.1669	ANALISIS- DISEÑO
Ajustes a formas anexas: frmEsquemasSistemas	09/03/2006 08:10	.3334	DESARROLLO
Validación y prueba de formas anexas	09/03/2006 08:30	.5001	PRUEBAS
Corrección de formas anexas	09/03/2006 09:00	1.0835	CORRECCIONES
Validación y prueba de formas anexas	09/03/2006 10:05	.6668	PRUEBAS
Ajustes a formas anexas: frmGraficasSeguimiento	09/03/2006 10:45	1.2502	DESARROLLO
Validación y prueba de formas anexas	09/03/2006 12:00	.5001	PRUEBAS
Corrección de formas anexas	09/03/2006 14:30	1.5002	CORRECCIONES
Validación y prueba de formas anexas	09/03/2006 16:00	1.0002	PRUEBAS
Análisis del funcionamiento de las formas anexas: frmHabporcentajes	09/03/2006 17:00	1.0002	ANALISIS- DISEÑO
Análisis del funcionamiento de las formas anexas: Incrementomensual	09/03/2006 18:00	.5001	ANALISIS- DISEÑO
Análisis del funcionamiento de las formas anexas:	10/03/2006 08:05	1.7472	ANALISIS- DISEÑO

TAREA	INICIO	HORAS	TIPO_TAREA
frmInforaestaciones			
Ajustes a formas anexas	10/03/2006 09:50	.1667	DESARROLLO
Análisis del funcionamiento de las formas anexas: frmkwh	10/03/2006 10:00	1.0002	ANALISIS-DISEÑO
Validación y prueba de formas anexas	10/03/2006 11:00	1.5002	PRUEBAS
Análisis del funcionamiento de las formas anexas: frmprminfestaciones	10/03/2006 14:30	1.5002	ANALISIS-DISEÑO
Ajustes a formas anexas	10/03/2006 16:00	1.5002	DESARROLLO
Análisis del funcionamiento de las formas anexas: frmRCDEDatos	10/03/2006 17:30	1.0002	ANALISIS-DISEÑO
Ajustes a formas anexas: frmkwh	11/03/2006 08:30	2.9171	DESARROLLO
Ajustes a formas anexas: frmprminfestaciones	11/03/2006 11:25	.0833	DESARROLLO
Ajustes a formas anexas: frmRCDEDatos	11/03/2006 11:30	1.0002	DESARROLLO
Ajustes a formas anexas: frmHabporcentajes	13/03/2006 08:26	1.9695	DESARROLLO
Ajustes a formas anexas: Incrementomensual	13/03/2006 10:25	2.0837	DESARROLLO
Ajustes a formas anexas: frmInforaestaciones	13/03/2006 14:45	1.2502	DESARROLLO
Análisis del funcionamiento de las formas anexas: frmTendcostorsqgestaciones	13/03/2006 16:00	.5001	ANALISIS-DISEÑO
Ajustes a formas anexas: frmTendcostorsqgestaciones	13/03/2006 16:30	.5001	DESARROLLO
Ajustes a formas anexas: frmTendcostorsqgestaciones	14/03/2006 08:20	1.6669	DESARROLLO
Validación y prueba de formas anexas	14/03/2006 10:00	.5001	PRUEBAS
Análisis del funcionamiento de las formas anexas: frmTendestaciones	14/03/2006 10:30	.7501	ANALISIS-DISEÑO
Ajustes a formas anexas:	14/03/2006	.0833	DESARROLLO

TAREA	INICIO	HORAS	TIPO_TAREA
frmTendestaciones	11:15		
Análisis del funcionamiento de las formas anexas: frmValorenBolsaConf	14/03/2006 11:25	.5834	ANALISIS-DISEÑO
Ajustes a formas anexas: frmValorenBolsaConf	14/03/2006 12:00	.5001	DESARROLLO
Ajustes a formas anexas: frmTendestaciones	14/03/2006 14:30	1.0002	DESARROLLO
Ajustes a formas anexas: frmTendcostorsqgestaciones	14/03/2006 16:00	1.4169	DESARROLLO
Ajustes a formas: frmAsignacionVolumenes	14/03/2006 17:25	.3334	DESARROLLO
Ajustes a formas: frmCDinamica	14/03/2006 17:45	.7501	DESARROLLO
Ajustes a formas: frmconfiguradorestaciones	15/03/2006 08:30	4.0006	DESARROLLO
Ajustes a formas: frmconfigurarestaciones	15/03/2006 14:30	2.8338	DESARROLLO
Ajustes a formas: frmEsquemas	15/03/2006 17:45	.7501	DESARROLLO
Ajustes a formas: frmOpOperaci	16/03/2006 08:15	3.7506	DESARROLLO
Ajustes a formas: frmProgramacion	16/03/2006 14:30	4.0006	DESARROLLO
Ajustes a formas: frmUsuarios	17/03/2006 08:30	4.0006	DESARROLLO
Análisis del funcionamiento de las formas	17/03/2006 14:30	1.0002	ANALISIS-DISEÑO
Ajustes a formas: frmAdministradorInformes	17/03/2006 15:30	3.0005	DESARROLLO
Ajustes a formas: frmComparativosScada	18/03/2006 08:30	.6668	DESARROLLO
Ajustes a formas: frmConfiguradorSistemas	18/03/2006 09:10	1.5836	DESARROLLO
Ajustes a formas: frmInforaestaciones	18/03/2006 10:45	1.7503	DESARROLLO
Ajustes a formas: frmtendcostovsqgestaciones	21/03/2006 08:15	4.2507	DESARROLLO
Ajustes a formas: frmTendUnidades	21/03/2006 14:30	4.0006	DESARROLLO
Ajustes a formas: frmTendestaciones	22/03/2006	.7501	DESARROLLO

TAREA	INICIO	HORAS	TIPO_TAREA
	08:15		
Ajustes a formas: RCDEDatos	22/03/2006 10:00	2.5004	DESARROLLO
Ajustes a formas: frmCriteriosdeOptimizacion	22/03/2006 14:30	.5001	DESARROLLO
Ajustes a formas: frmEliSistemas	22/03/2006 15:15	1.7503	DESARROLLO
Ajustes a formas: frmEntregascd	22/03/2006 17:15	1.2502	DESARROLLO
Depuración y pruebas de formas	23/03/2006 08:15	1.2502	PRUEBAS
Depuración y pruebas de formas	23/03/2006 09:30	2.0003	PRUEBAS
Análisis del funcionamiento de las formas	23/03/2006 11:30	.25	ANALISIS-DISEÑO
Ajustes a formas: frmCEstacionaria	23/03/2006 11:45	.25	DESARROLLO
Ajustes a formas: frmdatosScada	23/03/2006 12:00	.5001	DESARROLLO
Depuración y pruebas de formas	23/03/2006 14:30	2.0003	PRUEBAS
Corrección de las formas	23/03/2006 16:30	.1667	CORRECCIONES
Ajustes a formas: frmopcionesinforme	23/03/2006 16:55	.3334	DESARROLLO
Ajustes a formas: frminfparoscd	23/03/2006 17:15	1.2502	DESARROLLO
Ajustes a formas: frmparosenttiempo	24/03/2006 08:30	2.4926	DESARROLLO
Análisis del funcionamiento de las formas	24/03/2006 11:00	.7501	ANALISIS-DISEÑO
Ajustes a formas: frmrepartosHap	24/03/2006 11:45	.7501	DESARROLLO
Ajustes a formas: frmparosenvolumen	24/03/2006 14:30	2.5004	DESARROLLO
Ajustes a formas: frmProgramacion	24/03/2006 17:00	1.0002	DESARROLLO
Ajustes a formas: frmllenadolineacd	24/03/2006 18:00	1.3335	DESARROLLO
Ajustes a formas: frmGraficasSeguimiento	25/03/2006 09:30	1.0002	DESARROLLO

TAREA	INICIO	HORAS	TIPO_TAREA
Depuración y pruebas de formas	25/03/2006 10:30	2.0003	PRUEBAS
Análisis del funcionamiento de las formas	27/03/2006 08:00	.25	ANALISIS-DISENO
Análisis del funcionamiento de las formas	27/03/2006 08:15	.25	ANALISIS-DISENO
Ajustes a formas: frmdesplieguepccd	27/03/2006 08:30	.5001	DESARROLLO
Depuración y pruebas de formas	27/03/2006 09:00	.1667	PRUEBAS
Análisis del funcionamiento de las formas	27/03/2006 09:20	.7501	ANALISIS-DISENO
Análisis del funcionamiento de las formas	27/03/2006 10:05	1.5002	ANALISIS-DISENO
Corrección de formas	27/03/2006 11:35	.25	CORRECCIONES
Ajustes a formas: frmcorrimientolinea	27/03/2006 11:50	.6668	DESARROLLO
Ajustes a formas: frmllenadodelineacd	27/03/2006 14:30	2.8338	DESARROLLO
Corrección de formas	27/03/2006 17:20	.5001	CORRECCIONES
Ajustes a formas: frmRepartoByD	27/03/2006 17:50	.6668	DESARROLLO
Ajustes a formas: frmcondarranque	28/03/2006 08:15	4.2507	DESARROLLO
Ajustes a formas: MDIForm1	28/03/2006 14:30	1.817	DESARROLLO
Análisis del funcionamiento de las formas	28/03/2006 16:19	.6804	ANALISIS-DISENO
Ajustes a formas: frmElogin	28/03/2006 17:00	1.5002	DESARROLLO
Ajustes a formas: frmUsuarios	29/03/2006 08:15	3.2505	DESARROLLO
Análisis del funcionamiento de las formas	29/03/2006 11:30	.25	ANALISIS-DISENO
Ajustes a formas: frmDefSistemas	29/03/2006 11:45	.7501	DESARROLLO
Ajustes a formas: frmparosenvolumen	29/03/2006 14:30	2.167	DESARROLLO
Ajustes a formas:	29/03/2006	1.0835	DESARROLLO

TAREA	INICIO	HORAS	TIPO_TAREA
frmConfiguradorSistemas	16:40		
Ajustes a formas: frmdatabasicoconf	29/03/2006 17:45	.7501	DESARROLLO
Análisis del funcionamiento de las formas	30/03/2006 08:20	.6668	ANALISIS-DISEÑO
Ajustes a formas: frmCTramos	30/03/2006 09:00	3.5006	DESARROLLO
Ajustes a formas: frmDespCostosKwhconf	30/03/2006 14:30	1.5002	DESARROLLO
Ajustes a formas: frmCodigoHap	30/03/2006 16:00	1.0002	DESARROLLO
Ajustes a formas: frmconfigurarEstaciones	30/03/2006 17:20	1.1669	DESARROLLO
Ajustes a formas: frmCnrCrConf	31/03/2006 08:30	1.0002	DESARROLLO
Ajustes a formas: frmTopologia	31/03/2006 09:35	2.8338	DESARROLLO
Ajustes a formas: frmRenSistema	31/03/2006 14:45	1.7503	DESARROLLO
Ajustes a formas: frmValoresBolsaConf	31/03/2006 16:50	1.6669	DESARROLLO
Análisis de funciones para ejecutar Corrida Estacionaria	01/04/2006 08:40	.4167	ANALISIS-DISEÑO
Ajustes en frmCEstacionaria	01/04/2006 09:05	3.4172	DESARROLLO
Ajustes en frmCEstacionaria	03/04/2006 08:15	.25	DESARROLLO
Ajustes en función infoExcel	03/04/2006 10:45	1.7503	DESARROLLO
Ajustes en frmCEstacionaria	03/04/2006 14:45	2.2454	DESARROLLO
Corrección a función infoExcel de frmCEstacionaria	03/04/2006 17:20	1.1669	CORRECCIONES
Ajustes en frmCDinamica	04/04/2006 08:30	.7501	DESARROLLO
Ajustes en frmCEstacionaria	04/04/2006 09:20	3.1672	DESARROLLO
Ajustes en frmCDinamica	04/04/2006 14:40	3.8339	DESARROLLO
Corrección a formLoad de frmCDinamica	05/04/2006 08:15	1.0002	CORRECCIONES

TAREA	INICIO	HORAS	TIPO_TAREA
Ajustes en frmCEstacionaria	05/04/2006 09:15	3.2505	DESARROLLO
Ajustes en frmCDinamica	05/04/2006 14:40	1.3335	DESARROLLO
Análisis de funciones para ejecutar Corrida Dinamica	05/04/2006 16:00	.25	ANALISIS-DISEÑO
Ajustes en frmCDinamica	05/04/2006 16:30	.5001	DESARROLLO
Análisis de funciones Scada	05/04/2006 17:00	.5001	ANALISIS-DISEÑO
Ajustes en frmComparativosScada	05/04/2006 17:45	.25	DESARROLLO
Ajustes en frmComparativosScadaSistema	05/04/2006 18:10	.3334	DESARROLLO
Análisis de funciones Scada	06/04/2006 08:17	.3834	ANALISIS-DISEÑO
Ajustes en frmComparativosScada	06/04/2006 08:45	1.0835	CORRECCIONES
Ajustes en frmComparativosScadaSistema	06/04/2006 10:00	1.5002	CORRECCIONES
Análisis de nuevas funciones Corrida Estacionaria	06/04/2006 11:30	.7501	ANALISIS-DISEÑO
Análisis de nuevas funciones Corrida Estacionaria	06/04/2006 12:20	.1	ANALISIS-DISEÑO
Ajustes en frmComparativosScada	06/04/2006 14:35	1.1669	DESARROLLO
Ajustes en frmComparativosScadaSistema	06/04/2006 15:45	.5001	DESARROLLO
Ajustes en frmAdministradorInformes	06/04/2006 16:20	.3334	DESARROLLO
Ajustes en frmGraficasSeguimiento	06/04/2006 16:50	.8335	DESARROLLO
Ajustes en frmCDinamica	06/04/2006 17:50	.6668	DESARROLLO
Ajustes en frmCDinamica	07/04/2006 08:20	.8335	DESARROLLO
Ajustes en frmCEstacionaria	07/04/2006 09:15	.6668	DESARROLLO
Ajustes para nuevas funcionalidades	07/04/2006 10:00	.25	CORRECCIONES
Ajustes en frmEliSistemas	07/04/2006	2.0837	DESARROLLO

TAREA	INICIO	HORAS	TIPO_TAREA
	10:15		
Análisis para desarrollar reportes en ASP	07/04/2006 12:20	.1667	ANALISIS-DISEÑO
Ajustes en frmAdministradorInformes	07/04/2006 14:30	2.0003	DESARROLLO
Pruebas de conexión desde ASP	07/04/2006 16:30	.25	PRUEBAS
Ajustes en frmAdministradorInformes	07/04/2006 17:00	1.5002	DESARROLLO
Análisis para desarrollar reportes en ASP	08/04/2006 08:30	.1667	ANALISIS-DISEÑO
Desarrollo de página inf_corridaD.asp	08/04/2006 08:43	.7685	DESARROLLO
Desarrollo de página inf_corridaD.asp	08/04/2006 09:42	1.8003	DESARROLLO
Pruebas de páginas ASP	08/04/2006 11:30	.25	PRUEBAS
Corrección por pruebas	08/04/2006 11:45	.25	CORRECCIONES
Análisis para desarrollar reportes en ASP	17/04/2006 08:19	.0839	ANALISIS-DISEÑO
Desarrollo de página inf_EstacionesD.asp	17/04/2006 08:25	.4167	DESARROLLO
Desarrollo de página inf_Scada.asp	17/04/2006 08:58	1.7714	DESARROLLO
Ajustes a páginas ASP	17/04/2006 10:45	.25	CORRECCIONES
Desarrollo de página inf_ScadaAIO.asp	17/04/2006 11:02	.3695	DESARROLLO
Ajustes a inf_Scada.asp	17/04/2006 11:30	1.0002	CORRECCIONES
Ajustes a inf_ScadaAIO.asp	17/04/2006 14:30	2.167	DESARROLLO
Análisis a funcionamiento de reportes ASP	17/04/2006 16:40	.0833	ANALISIS-DISEÑO
Ajustes a inf_Scada.asp	17/04/2006 16:50	1.1669	DESARROLLO
Ajustes a inf_ScadaAIO.asp	17/04/2006 18:00	.5001	DESARROLLO
Ajustes a inf_corridaD.asp	17/04/2006 18:30	.5001	DESARROLLO

TAREA	INICIO	HORAS	TIPO_TAREA
Ajustes a inf_corridaD.asp	18/04/2006 08:30	1.5002	DESARROLLO
Ajustes a inf_corridaE.asp	18/04/2006 10:15	.9168	DESARROLLO
Ajustes a inf_EstacionesD.asp	18/04/2006 11:15	1.2502	DESARROLLO
Ajustes por cambios en la aplicación	18/04/2006 14:35	.5001	DESARROLLO
Ajustes por cambios en la aplicación	18/04/2006 15:10	.25	DESARROLLO
Ajustes por cambios en la aplicación	18/04/2006 15:30	.5001	DESARROLLO
Ajustes por cambios en la aplicación	18/04/2006 16:05	.9168	DESARROLLO
Ajustes por cambios en la aplicación	18/04/2006 17:00	.3334	DESARROLLO
Análisis cambios en reportes ASP	18/04/2006 17:20	.25	ANALISIS- DISEÑO
Ajustes a inf_corridaE.asp	18/04/2006 17:40	.8335	DESARROLLO
Ajustes a inf_corridaD.asp	19/04/2006 08:10	.25	DESARROLLO
Ajustes a inf_EstacionesD.asp	19/04/2006 08:25	2.3337	DESARROLLO
Análisis cambios en reportes ASP	19/04/2006 10:45	.5001	ANALISIS- DISEÑO
Ajustes a inf_Scada.asp	19/04/2006 11:15	.5001	DESARROLLO
Ajustes a inf_ScadaAIO.asp	19/04/2006 11:45	.6168	DESARROLLO
Ajustes a inf_EstacionesD.asp	19/04/2006 14:50	.6668	DESARROLLO
Pruebas a páginas ASP	19/04/2006 15:30	.25	PRUEBAS
Ajustes a páginas ASP	19/04/2006 15:50	.0833	DESARROLLO
Análisis nuevos ajustes a la aplicación	19/04/2006 16:00	2.5004	ANALISIS- DISEÑO
Ajustes a la aplicación optimizador	20/04/2006 08:30	4.0006	DESARROLLO
Análisis para nuevos ajustes en la	20/04/2006	4.0006	ANALISIS-

TAREA	INICIO	HORAS	TIPO_TAREA
aplicación	14:30		DISEÑO
Ajustes a formas: frmAsignacionVolumenes	21/04/2006 08:10	2.8338	DESARROLLO
Ajustes a formas: frmCDinamica	21/04/2006 11:00	1.5002	DESARROLLO
Ajustes a formas: frmconfiguradorestaciones	21/04/2006 17:15	1.2502	DESARROLLO
Ajustes a formas: frmconfigurarestaciones	24/04/2006 08:10	1.8336	DESARROLLO
Ajustes a formas: frmEsquemas	24/04/2006 10:00	2.5004	DESARROLLO
Ajustes a formas: frmOpOperaci	24/04/2006 14:35	1.0168	DESARROLLO
Ajustes a formas: frmProgramacion	24/04/2006 15:38	2.8671	DESARROLLO
Ajustes a formas: frmUsuarios	24/04/2006 18:35	.6668	DESARROLLO
Ajustes a formas: frmAdministradorInformes	25/04/2006 08:05	2.4171	DESARROLLO
Ajustes a formas: frmComparativosScada	25/04/2006 10:45	.7501	DESARROLLO
Ajustes a formas: frmConfiguradorSistemas	25/04/2006 11:30	.7501	DESARROLLO
Ajustes a formas: frmInforaestaciones	25/04/2006 14:55	1.0835	DESARROLLO
Análisis para nuevos ajustes en la aplicación	25/04/2006 16:10	.3334	ANALISIS- DISEÑO
Análisis para nuevos ajustes en la aplicación	25/04/2006 16:30	1.4169	ANALISIS- DISEÑO
Ajustes en frmCDinamica	25/04/2006 18:00	.5001	DESARROLLO
Ajustes en frmCEstacionaria	26/04/2006 08:15	.9168	DESARROLLO
Pruebas a las formas ajustadas	26/04/2006 09:10	.1667	PRUEBAS
Corrección según pruebas	26/04/2006 09:20	.6668	CORRECCIONES
Ajustes en frmComparativosScada	26/04/2006 10:00	2.5004	DESARROLLO
Ajustes en frmComparativosScadaSistema	26/04/2006 14:35	2.9171	DESARROLLO

TAREA	INICIO	HORAS	TIPO_TAREA
Ajustes en frmAdministradorInformes	26/04/2006 17:30	1.7503	DESARROLLO
Ajustes en frmGraficasSeguimiento	27/04/2006 08:10	.9168	DESARROLLO
Pruebas a las formas ajustadas	27/04/2006 09:12	.3	PRUEBAS
Ajustes en frmComparativosScada	27/04/2006 09:40	1.0835	DESARROLLO
Ajustes en frmComparativosScadaSistema	27/04/2006 10:50	.8335	DESARROLLO
Pruebas a las formas ajustadas	27/04/2006 11:40	.1667	PRUEBAS
Ajustes en frmComparativosScada	27/04/2006 11:55	.5834	DESARROLLO
Ajustes en frmComparativosScadaSistema	27/04/2006 14:35	3.9173	DESARROLLO
Ajustes en frmAdministradorInformes	28/04/2006 08:05	3.9173	DESARROLLO
Pruebas a las formas ajustadas	28/04/2006 12:00	.5001	PRUEBAS
Ajustes en frmAdministradorInformes	28/04/2006 14:35	3.9173	DESARROLLO
Análisis de la aplicación estudiada con anterioridad	02/05/2006 08:15	.7501	PRUEBAS
Arreglo de Reportes	02/05/2006 09:05	.9168	DESARROLLO
Arreglo de Reportes	02/05/2006 10:05	1.1669	DESARROLLO
Arreglo de Reportes	02/05/2006 11:20	1.0835	DESARROLLO
Análisis de funcionalidad	02/05/2006 14:45	.1667	ANALISIS- DISEÑO
Arreglo de Reportes	02/05/2006 14:55	.5834	DESARROLLO
Arreglo de Reportes	02/05/2006 16:00	1.0002	DESARROLLO
Arreglo de Reportes	02/05/2006 17:00	.5001	DESARROLLO
Validación de Filtros en la opción Reportes	02/05/2006 18:00	.5001	PRUEBAS
Arreglo de Reportes	02/05/2006	.5001	DESARROLLO

TAREA	INICIO	HORAS	TIPO_TAREA
	18:30		
Análisis de funcionalidad	03/05/2006 08:15	.1667	ANALISIS- DISEÑO
Arreglo en la opción Registro de Convenios	03/05/2006 08:25	.3334	DESARROLLO
Arreglo en la opción Registro de Convenios	03/05/2006 08:50	2.167	DESARROLLO
Arreglo en la opción Registro de Convenios	03/05/2006 11:20	.8335	DESARROLLO
Arreglo en la opción Registro de Convenios	03/05/2006 12:15	.1667	DESARROLLO
Arreglo en la opción Registro de Convenios	03/05/2006 14:50	1.4169	DESARROLLO
Arreglo en la opción Registro de Convenios	03/05/2006 18:00	.5001	DESARROLLO
Arreglo en la opción Registro de Convenios	04/05/2006 08:10	4.334	DESARROLLO
Arreglo en la opción Registro de Convenios	04/05/2006 14:40	.8335	DESARROLLO
Revisión reporte Convenios por año	04/05/2006 15:40	.5001	CORRECCIONES
Ajuste en Opción Registro Convenios años anteriores	04/05/2006 16:15	.9168	DESARROLLO
Ajuste en Opción Registro Convenios años anteriores	05/05/2006 08:10	1.0835	DESARROLLO
Modificación en Opción Registro Convenios años anteriores	05/05/2006 09:25	.5834	CORRECCIONES
Ajuste en Opción Registro Convenios años anteriores	05/05/2006 10:10	.5834	DESARROLLO
Modificación en Opción Registro Convenios años anteriores	05/05/2006 10:45	1.7503	CORRECCIONES
Modificación en Presentación de Carpetas en Registro	05/05/2006 14:45	3.7506	CORRECCIONES
Ajustes en configuración de Roles	08/05/2006 08:15	.25	DESARROLLO
Ajustes en Impuesto de Timbre	08/05/2006 08:35	.25	DESARROLLO
Ajustes en Objeto del Convenio	08/05/2006 08:50	1.3836	DESARROLLO
Análisis de funcionalidad de garantías	08/05/2006 10:14	.5092	ANALISIS- DISEÑO

TAREA	INICIO	HORAS	TIPO TAREA
Arreglo de campos de póliza	08/05/2006 10:50	.5001	DESARROLLO
Arreglo de opción Nuevo de Socios	08/05/2006 11:20	.1667	ANALISIS- DISENO
Arreglos de Impuesto de Timbre	08/05/2006 11:30	.25	DESARROLLO
Corrección en checklist para desembolsos	08/05/2006 11:50	.6668	CORRECCIONES
Corrección en checklist para desembolsos	08/05/2006 15:00	1.7503	CORRECCIONES
Revisión de Checklist para desembolsos	08/05/2006 16:50	1.6669	ANALISIS- DISENO
Arreglos en funciones con personas activas	09/05/2006 08:15	2.2504	DESARROLLO
Arreglo en datos de documento de cobro	09/05/2006 10:30	.8335	DESARROLLO
Estudio de funcionalidad en desembolsos	09/05/2006 11:22	.1334	ANALISIS- DISENO
Ajuste en checklist de desembolsos	09/05/2006 11:30	1.0002	DESARROLLO
Ajuste en checklist de desembolsos	09/05/2006 14:54	.7668	DESARROLLO
Ajuste en checklist de desembolsos	09/05/2006 15:45	.25	DESARROLLO
Arreglos en Convenios tesis de grado	09/05/2006 16:20	.25	DESARROLLO
Arreglos en Agregar Personas	09/05/2006 16:40	1.8336	DESARROLLO
Arreglos en grabar garantías	10/05/2006 08:15	1.7503	DESARROLLO
Revisión tipos nivel académico	10/05/2006 10:10	.8335	PRUEBAS
Arreglos en Ejecución	10/05/2006 11:02	.9652	DESARROLLO
Arreglos en Desembolsos	10/05/2006 12:03	.4401	DESARROLLO
Análisis de opción Entregables	10/05/2006 15:45	.7501	ANALISIS- DISENO
Arreglos en Entregables	10/05/2006 16:30	1.0002	DESARROLLO
Arreglo de error en adicionar socio	10/05/2006	1.5836	DESARROLLO

TAREA	INICIO	HORAS	TIPO_TAREA
	17:35		
Revisión de error en adicionar socio	11/05/2006 08:15	.3334	PRUEBAS
Análisis de opción Crear Personas	11/05/2006 08:32	.1167	ANALISIS- DISEÑO
Arreglos en opción Crear Personas	11/05/2006 08:45	.7457	DESARROLLO
Arreglo de error al crear personas	11/05/2006 09:50	.5334	DESARROLLO
Arreglo de error al guardar garantía	11/05/2006 10:30	2.0003	DESARROLLO
Arreglo de error al eliminar garantía	11/05/2006 14:45	3.7506	DESARROLLO
Arreglo de error al guardar aporte	12/05/2006 08:20	.3334	DESARROLLO
Arreglo de error al eliminar aporte	12/05/2006 08:45	.6668	DESARROLLO
Arreglo de error al generar reporte	12/05/2006 09:30	.7501	DESARROLLO
Arreglo de error al guardar convenio	12/05/2006 10:15	.25	DESARROLLO
Arreglo de error al eliminar garantía	12/05/2006 10:32	.2167	DESARROLLO
Arreglo de error en visualización de fechas	12/05/2006 10:50	.1667	DESARROLLO
Ajustes en tipos de datos	12/05/2006 11:02	.4667	DESARROLLO
Ajustes en mensajes de error	12/05/2006 11:30	1.0002	DESARROLLO
Ajustes en mensajes de validación	12/05/2006 16:20	.9168	DESARROLLO

Tabla 2- Detallado de Actividades

10. TIEMPO POR TIPO DE TAREA EN CADA DÍA DE LA SEMANA

Tiempo en horas dedicado por el practicante en cada día de la semana a los diferentes tipos de tareas

Que se ven en un proceso de desarrollo de software

TIPO_TAREA	DIA	HORAS
PRUEBAS	LUNES	9.1681
DESARROLLO	LUNES	49.1292
CORRECCIONES	LUNES	10.2683
ANÁLISIS-DISEÑO	LUNES	6.094
PRUEBAS	JUEVES	9.9683
DESARROLLO	JUEVES	55.121
CORRECCIONES	JUEVES	5.8343
ANÁLISIS-DISEÑO	JUEVES	18.3529
PRUEBAS	MARTES	5.0008
DESARROLLO	MARTES	72.2191
CORRECCIONES	MARTES	5.3509
ANÁLISIS-DISEÑO	MARTES	11.2321
PRUEBAS	SÁBADO	2.2504
DESARROLLO	SÁBADO	14.9874
CORRECCIONES	SÁBADO	.25
ANÁLISIS-DISEÑO	SÁBADO	4.5841
PRUEBAS	VIERNES	3.8339
DESARROLLO	VIERNES	67.2863
CORRECCIONES	VIERNES	6.3343
ANÁLISIS-DISEÑO	VIERNES	15.6661
PRUEBAS	MIÉRCOLES	3.5006
DESARROLLO	MIÉRCOLES	69.8162
CORRECCIONES	MIÉRCOLES	6.4177
ANÁLISIS-DISEÑO	MIÉRCOLES	11.9686

Tabla 3- *Tiempos por Tarea en cada día de la semana*

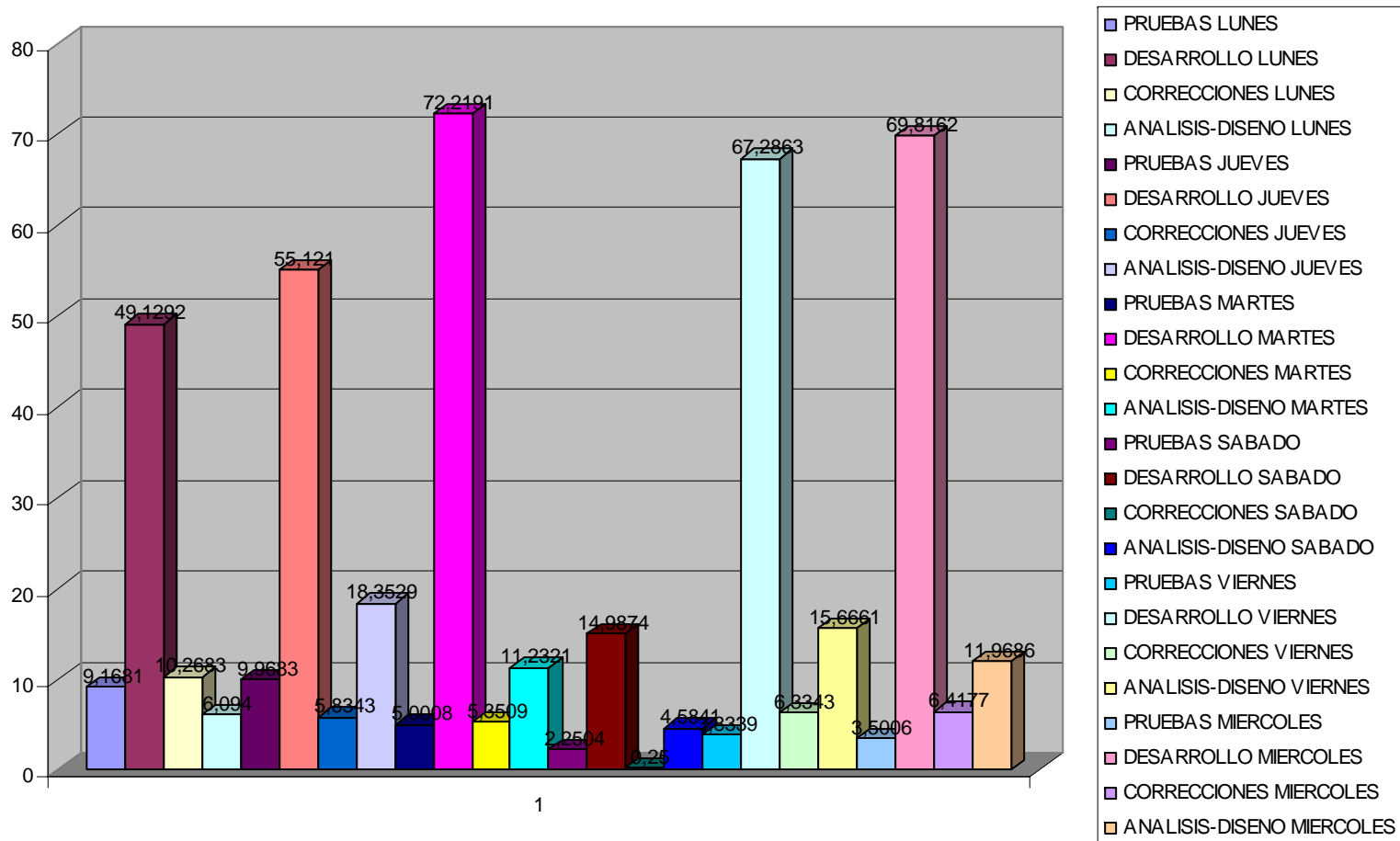


Figura 4- Gráfico de Tiempos por Tarea en cada día de la semana

11. TIEMPO POR PROCESO

PROCESO	HORAS
MEDICIÓN Y MEJORA	16,8735
PERSONAL	9,3348
PLANEACIÓN	21,3904
PRODUCCIÓN	417,0359

Tabla 4- *Tiempos por proceso*

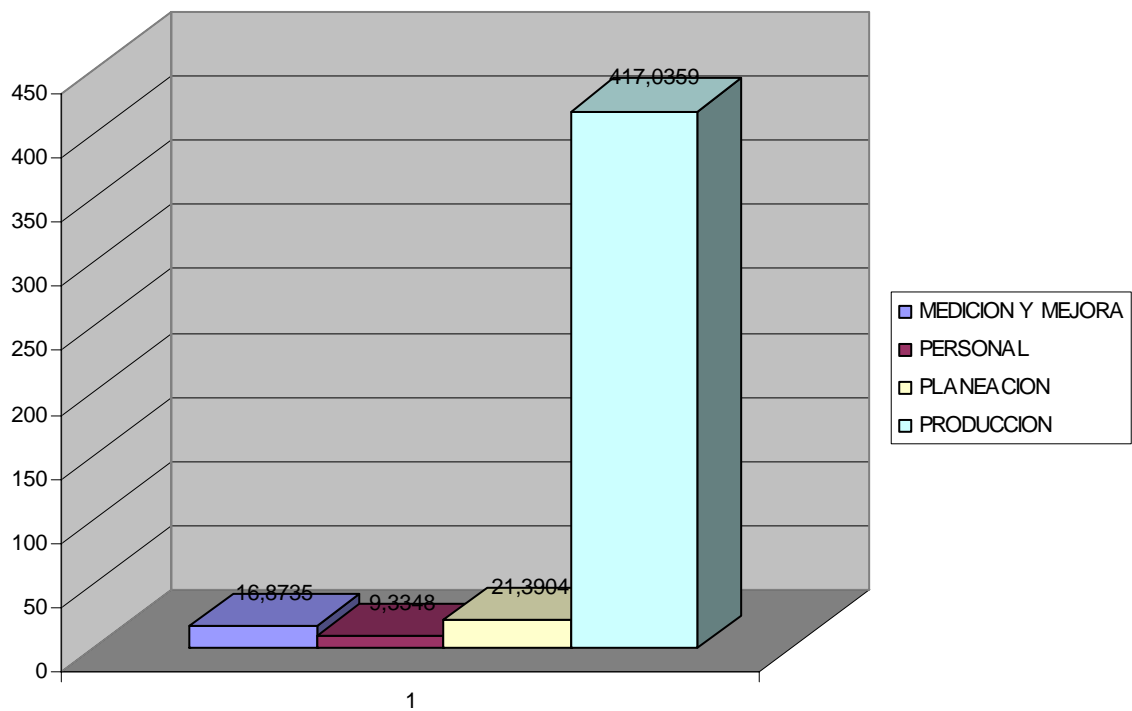


Figura 5 – *Gráfico de Tiempos por proceso*

CONCLUSIONES

La utilización de la metodología RUP durante la práctica empresarial facilitó el control del proceso de desarrollo del software; además la aplicación de conceptos de ingeniería de software aprendidos durante el ejercicio de la carrera y la realización de esta práctica empresarial permite la construcción de software de alta calidad de una forma productiva.

Es necesario resaltar la utilización de la metodología RUP en la realización de productos software de la empresa, en donde se trabaja por iteraciones, se hacen ciclos de programación orientados a entregas cortas de productos y se mantiene mucha interacción con el usuario final para asegurar que los requerimientos se están manejando de manera adecuada.

Por otra parte, con el fin de tener a la mano un marco de trabajo estructurado para desarrollar las habilidades personales y los métodos que se necesitarán como Ingenieros de sistemas se tomaron conceptos de la disciplina PSP (Personal Software Process). La medición y las métricas ayudaron a entender tanto el proceso técnico que se utiliza para desarrollar un producto como el propio producto.

Se presentó la metodología PSP a consideración de la empresa, así como los beneficios y ventajas que podría ofrecer dicha metodología al proceso de desarrollo de software. Además se manifestó por parte de la Empresa que se estaba elaborando un software para el control y seguimiento del proceso de desarrollo de software con estas metodologías.

La empresa propuso considerar los criterios de PSP para medición de desempeño, para generar indicadores de gestión de los estudiantes en práctica que apoyan proyectos de desarrollo y posteriormente se pudo observar que la plantilla de registro de tiempos fue incluida casi en su totalidad en el sistema de gestión de la calidad de la Empresa.

En cuanto a la información de tiempo por proceso, podemos observar que debido a que el tipo de actividades para las cuales se solicitó al estudiante en práctica, como lo es el apoyo y mantenimiento de productos ya desarrollados. La mayor parte del tiempo del practicante se destina a al proceso de producción, viéndose una diferencia significativa con respecto a los procesos de planeación, medición y mejora. Debido en gran parte a que dichos procesos se ven en mayor parte en el proceso de creación de un software nuevo, más que en el mantenimiento del mismo.

También me gustaría mostrar como conclusión lo importante que es la modalidad de práctica empresarial como alternativa de proyecto de grado, pues brinda al estudiante la oportunidad de enfrentarse a casos reales de desarrollo y por ende adquirir una experiencia real en el campo del desarrollo, brindándole unas mayores oportunidades al momento de salir a conseguir un puesto de trabajo, y ampliado su apreciación en cuanto a lo que es un ambiente de trabajo y las responsabilidades que genera el ser parte de un proyecto dentro de una empresa.

BIBLIOGRAFIA

<http://www.programatium.com/01Manuales/Lenguajes/visualbasic.htm>

<http://www.microsoft.com/spanish/msdn/vs2005/default.msp>

<http://www.asptutor.com/sql/>

http://www.ingenieria.cl/escuelas/informatica/apuntes_curso_uml/Documentos%20uml.pdf

Object-Oriented Project Management with UML.
MURRAY R., Cantor

Microsoft Visual Basic 6.0
Manual del Programador
Microsoft Corporation, Editorial McGraw-Hill

Planeación de Proyectos – Un enfoque para la Ingeniería de Sistemas e Informática. Bucaramanga, Colombia. 2004.
GOMEZ FLOREZ, Luis Carlos