

Universidad Popular Autónoma del Estado de Puebla
Centro Interdisciplinario de Posgrados
Investigación y Consultoría
Departamento de Ingeniería
Doctorado en Planeación Estratégica y Dirección
de Tecnología

Título de la investigación

**Modelo de indicadores para evaluar la complejidad de proyectos de
software**

Tesis que para obtener el Grado de Doctor
en Planeación Estratégica y Dirección de Tecnología

Presenta

Alejandro Fuentes Penna

Puebla, México.

2011



UPAEP – Secretaría General

Dirección General de Apoyos Académicos

Dirección del Centro de Recursos para el Aprendizaje y la Investigación.

Biblioteca Central - **Karol Wojtyla**

Tesis Digitales Restricciones de uso:

DERECHOS RESERVADOS ©

PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de textos, imágenes, gráficas, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente de donde la obtuvo mencionando el autor o autores involucrados en el documento.

Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

DEDICATORIA

Este trabajo está dedicado a todas las personas que me han apoyado...

A todas aquellas personas que en toda mi vida me han dado ánimos para continuar aún en la adversidad del tiempo...

A todos mis amigos, familiares y conocidos, porque de cada uno de ustedes he aprendido...

En especial, le dedico todo mi trabajo a mi esposa, porque ella está a mi lado a cada momento, en las buenas, en las malas y en las peores...

AGRADECIMIENTOS

A mi esposa, que sin su apoyo no podría ver cumplidas mis metas, porque es ella mi complemento... Admiro la fortaleza que refleja y su gran amor que me inspira para lograr cada paso, que, por pequeño que sea, se reflejan en los resultados que forman parte de mis triunfos, sus triunfos, nuestros triunfos...

A mis padres, porque creen en mí y con su apoyo he salido de diversas circunstancias y en gran parte, gracias a ustedes, hoy puedo ver alcanzada una meta más, ya que siempre me han impulsado en los momentos más difíciles, y porque el orgullo que sienten por mí, fue lo que me hizo ir hasta el final.

A mis hermanas, sobrinas, tíos, primos, abuelos y amigos, en general, a toda mi familia, por fomentar en mí el deseo de superación y el anhelo de triunfo en la vida.

Mil palabras no bastarían para agradecer su apoyo, su paciencia, y sus consejos en los momentos difíciles. Gracias a todos.

ÍNDICE GENERAL

RESUMEN

ABSTRACT

INTRODUCCIÓN

CAPÍTULO 1. PROPÓSITO Y ORGANIZACIÓN 1

CAPÍTULO 2. MARCO TEÓRICO 7

CAPÍTULO 3. MODELO DE INDICADORES PARA EVALUAR LA
COMPLEJIDAD DE PROYECTOS DE SOFTWARE..... 55

CAPÍTULO 4. EXPERIMENTACIÓN Y RESULTADOS 72

CAPÍTULO 5 ANÁLISIS Y DISCUSIÓN 95

6. REFERENCIAS 98

ANEXOS..... 108

ÍNDICE DE FIGURAS

Figura 1 Estructura de título	6
Figura 2 Grupo de procesos para la administración de proyectos en PMI.	10
Figura 3. Funciones y procesos representativos de la administración de proyectos	11
Figura 4 Estructura de ITIL	16
Figura 5 Diagrama de categorías de procesos de MoProSoft.....	19
Figura 6. Flujo de trabajo de la alta dirección.	20
Figura 7. Diagrama de flujo de trabajo de la gestión de proceso.	22
Figura 8. Flujo de trabajo de la gestión de recursos.	24
Figura 9. Diagrama de flujo de la gestión de recursos humanos y ambiente de trabajo.	26
Figura 10. Diagrama de flujo de trabajo de la gestión de bienes, servicios e infraestructura. ...	27
Figura 11. Diagrama de flujo de trabajo de la gestión del conocimiento de la organización....	29
Figura 12. Diagrama de flujo de trabajo de la administración de proyectos específicos.	32
Figura 13. Ejemplo de clasificación de productos de software	34
Figura 14. Relación gráfica entre tiempo y costo para proyectos con PERT/CPM	42
Figura 15. Secuencia de actividades en un proyecto	45
Figura 16. Taxonomía de PSP.....	46
Figura 17. Etapas para el desarrollo de software.....	72
Figura 18. Análisis de complejidad.	73
Figura 19. Índice “t” relacionado con los trabajadores que participarán en el proyecto.	74
Figura 20. Conjunto de datos de trabajadores	74
Figura 21. Índice “a” relacionado con las clases de actividades del proyecto.	74
Figura 22. Conjunto de datos de las clases de actividades.	75
Figura 23. Nivel de habilidad del trabajador para el desarrollo de actividades por clase.	75
Figura 24. Nivel de habilidad de los trabajadores definidos en el conjunto de datos de trabajadores.....	75
Figura 25. Número de actividades que podrá desempeñar cada trabajador en el proyecto actual.	76
Figura 26. Asignación del número máximo de actividades por trabajador.	76
Figura 27. Tiempo que le dedicará cada trabajador al proyecto actual.	76
Figura 28. Asignación del tiempo total que le dedicará cada trabajador al proyecto actual.	76
Figura 29. Parámetro relacionado con el número total de actividades por clase.....	77
Figura 30. Asignación del número de actividades por clase del proyecto actual.....	77
Figura 31. Parámetro relacionado con el costo unitario de las actividades por clase y por trabajador.	77
Figura 32. Asignación del costo de una actividad por clase y por trabajador considerando aspectos indirectos como: salario, nivel de habilidad, entre otros aspectos.	78
Figura 33. Parámetro relacionado con el tiempo promedio que se tiene en el desarrollo de una actividad por clase.	78
Figura 34. Tiempo promedio que le dedica un trabajador a una actividad por clase.	78
Figura 35. Parámetro relacionado con el número de errores promedio que tiene cada trabajador en el desarrollo de actividades por clase.	78

Figura 36. Asignación del número promedio de errores que tiene un trabajador en el desarrollo de actividades por clase.	79
Figura 37. Variable donde se guardará la asignación del tiempo que le dedicará cada trabajador a las diferentes clases de actividades con base en el número máximo de actividades por trabajador definidas en $TotalTiempoTrabajador(t)$	79
Figura 38. Resultado obtenido del modelo como resultado de la prueba inicial.....	80
Figura 39. Variable relacionada con el costo total del desarrollo del proyecto a partir de minimizar el costo con base en los costos unitarios por actividad.	80
Figura 40. Variable que vincula las actividades con los trabajadores.	80
Figura 41. Variable relacionada con el total de tiempo del proyecto a partir de minimizar el tiempo con base en el tiempo total que le dedica cada trabajador al proyecto y el número máximo de actividades.	80
Figura 42. Variable que minimiza el número total de errores con base en el parámetro $ErroresTrabajador(t,a)$	80
Figura 43. Variable donde se asignan las actividades con base en el nivel de habilidad que tiene cada trabajador para el desarrollo de actividades por clase. El objetivo es maximizar el nivel total de habilidades.	81
Figura 44. La asignación del número de actividades por trabajador debe ser menor ó igual al número máximo de actividades que dicho trabajador desarrollará en el proyecto actual.	81
Figura 45. Un trabajador no puede ser asignado más horas de las que dispone para el proyecto actual.....	81
Figura 46. Función matemática para minimizar el costo total del proyecto a partir de los parámetros: $TotalActividadesTrabajador(t,a)$, $TotalActividadesClase(a)$ y $CostoUnitarioActividad(t,a)$; y a partir de las funciones: $DesarrolloActividades(t,a)$ y $CostoTotalDesarrollo$	82
Figura 47. Función matemática para minimizar el tiempo total para el desarrollo del proyecto actual a partir de los parámetros: $TotalTiempoTrabajador(t)$ y $TiempoUnitarioActividad(t,a)$; y a partir de la función $TiempoTotalDesarrollo$	82
Figura 48. Función matemática para minimizar el número total de errores a partir del parámetro $ErroresTrabajador(t,a)$; y a partir de la variable $ErroresTotalesDesarrollo$	82
Figura 49. Función matemática para maximizar la asignación del nivel de habilidades en el desarrollo de las actividades a partir del parámetro $NivelHabilidadesTrabajador(a,t)$ y de la variable $AsignarHabilidades$	82
Figura 50. Declaración de la función principal a partir del algoritmo CPLEX.....	83
Figura 51. Interfaz generada a partir de la prueba inicial.....	83

ÍNDICE DE TABLAS

Tabla 1 Procesos de administración de proyectos en grupos de procesos y áreas de conocimiento	13
Tabla 2. Niveles de KPA.....	37
Tabla 3. Áreas de proceso en CMMI.....	39
Tabla 4. Algoritmos que resuelven algunas variantes de PSP.....	53
Tabla 5. Función objetivo.....	55
Tabla 6. Tabla de índices de la dimensión del proyecto.....	57
Tabla 7. Tabla de indicadores.....	59
Tabla 8. Variables de decisión.....	61
Tabla 9. Restricciones.....	61
Tabla 10. Funciones matemáticas.....	62
Tabla 11. Resultado de 25 instancias.....	86
Tabla 12. Datos para establecer la complejidad	88
Tabla 13. Análisis de las instancias 3 y 2.	90
Tabla 14. Asignación de trabajadores por clase de actividad.....	90
Tabla 15 Proyectos donde participan los trabajadores	92

RESUMEN

La presente investigación está enfocada en determinar la complejidad de un proyecto de desarrollo de software en la etapa de definición de requerimientos y generar una asignación de recursos humanos con base en las habilidades de cada trabajador en el desarrollo de las actividades del proyecto. El resultado permitirá generar una estimación del costo total y del tiempo total del proyecto partiendo de la asignación de habilidades al personal con base en resultados de proyectos anteriores.

ABSTRACT

This research is focused on determine the software project development complexity at the requirements definition step. On this research we generate a resources assignment based on workers skills, workers development time, activities cost per worker and the companies capacity on available resources. With the results, the project administrator will has an initial workload and an estimated project initial total cost.

INTRODUCCIÓN

La Planeación Estratégica (Newell, 2010) se puede definir como el proceso por el cual se ordenan los objetivos y las acciones en el tiempo; se basa en la selección de medios, objetivos y metas relacionados con la problemática a resolver desde un punto de vista del tiempo, costo y alcance de ésta. Dentro de la planeación se fija un estado inicial y un estado al que se quiere llegar en el transcurso del tiempo. Durante este proceso, se realiza una planeación inicial y ésta se ve afectada por el comportamiento de los recursos, los resultados que se obtienen y las actividades no planeadas; por ende, se requiere realizar un replanteamiento de dicha planeación conforme a los recursos actuales, el alcance y el tiempo restante.

Actualmente, las Pequeñas y Medianas Empresas (PYMES) conforman el 95% (INEGI, 2000) de las empresas en México, lo que conlleva que aquellas empresas que se dedican al desarrollo de software dedicadas a la producción regional, se encuentren en el rango de 1 a 20 trabajadores en promedio. Las PYMES dedicadas al desarrollo de software tienen algunas limitantes en cuanto a que no son atendidas las capacidades instaladas, las etapas de planeación y análisis de requerimientos en la medida que se requiere para el desarrollo de proyectos de software, dando lugar a un incremento en horas de corrección y modificación del software. El proceso de desarrollo de software consiste en 4 etapas fundamentales con base en el esquema en cascada (Molita et al, 2006): análisis, diseño, desarrollo e implementación. A partir de estas etapas básicas, se han desarrollado diferentes modelos tales como:

- Espiral.
- Prototipos.
- Modular.
- Entre otros.

La etapa de análisis consiste en definir los requerimientos y las características que tendrá el proyecto de software de acuerdo a las necesidades del cliente y de los usuarios; sin embargo, en esta etapa inicial no se contemplan aspectos como: planeación estratégica, asignación de recursos, determinación de las etapas de desarrollo, fechas parciales de entrega, entre otros puntos. De esta

manera, sólo se establece el costo total del proyecto y la entrega final del mismo, lo que conlleva la falta del seguimiento de las actividades y una asignación ineficaz de los recursos.

CAPÍTULO 1. PROPÓSITO Y ORGANIZACIÓN

Objetivos de la investigación

Objetivo general

- Desarrollar un modelo de indicadores que permitan determinar la complejidad de proyectos de desarrollo de software.

Objetivos particulares

- Analizar las características que son evaluadas en el desarrollo de proyectos de software estableciendo un conjunto de indicadores que permitan evaluar la complejidad de los proyectos de desarrollo de software.
- Desarrollar una metodología para identificar la complejidad de los proyectos de desarrollo de software con base en los recursos empleados mediante técnicas de optimización combinatoria para establecer una función óptima.
- Desarrollar una metodología para determinar los niveles de soporte en PYMES mediante ITIL (IT Infrastructure Library) relacionados con el desarrollo de software.

Planteamiento del problema

El Problema de programación de proyectos (PSP por sus siglas en inglés ‘Project Scheduling Problem’) de software consiste en determinar la asignación de actividades a los integrantes del equipo de desarrollo de forma equilibrada durante el proyecto. Para ello, es necesario determinar las características del proyecto de software en la etapa de planeación con base en el costo, alcance y tiempo empleado:

- Costo: Involucra a las personas como un conjunto de habilidades, salario y un horario asignado.
- Alcance: Identifica las características globales que tendrá el proyecto, estableciéndolo en tiempo y recursos.
- Tiempo: es la línea de tiempo que se determina para llevar a cabo las etapas del proyecto de software.

Si durante el desarrollo del proyecto de software hay modificaciones en alguno de estos aspectos, la alteración en los otros 2 aspectos es directamente proporcional a los cambios realizados. Existen actividades significativas que no son contempladas en la planeación como las modificaciones ó correcciones que se realizan de forma esporádica durante el desarrollo del proyecto de software. Por ende, la etapa de planeación de proyectos de software no se realiza con base en la experiencia del líder de proyecto.

En la mayoría de las empresas mexicanas la estimación de costo, alcance y tiempo empleados se realiza con base en la experiencia del director general ó dueño para determinar los salarios y la asignación de actividades de forma subjetiva sin llevar un seguimiento en el uso del tiempo laboral.

Actualmente, creemos que no existe una metodología ó modelo que permita determinar la complejidad de un proyecto de software a partir de la asignación de actividades, del alcance, costo y tiempo de desarrollo. Generalmente, la asignación de actividades tales como modificaciones ó correcciones al proyecto no es contemplada en el tiempo estimado para el proyecto. El desarrollo de métricas ha facilitado la evaluación de secciones de un proyecto de software de forma aislada tales como: métricas técnicas, de calidad, de productividad, orientadas a la persona, orientadas al tamaño y orientadas a la función; sin embargo, dichas métricas son sólo un referente para determinar la eficiencia promedio pero no determinan la eficiencia con base en las características de cada proyecto, donde el alcance y el tiempo influyen significativamente en el indicador de eficiencia.

Hipótesis

Si se determina inicialmente la complejidad de los proyectos de desarrollo de software a partir de una función objetivo, entonces se podrán reasignar o calendarizar de manera óptima los recursos disponibles para los proyectos con base en el comportamiento o capacidades de los participantes, las actividades y el presupuesto disponible del proyecto.

Justificación de la investigación

El Proceso de gestión para la creación de un Sistema o software se basa principalmente en la estimación y asignación de recursos con cierto grado de incertidumbre. Actualmente existen técnicas para estimar el costo y tiempo en el desarrollo de software siendo esta actividad la base de la planificación de un proyecto; tal es el caso de Microsoft Project cuya función principal es estimar el tiempo que duran las actividades y con la asignación de recursos por actividad. Dentro de los factores que determinan la planeación de un proyecto se encuentran: alcance, recursos y tiempo, principalmente; adicional a estos factores, el tamaño del proyecto y la complejidad del mismo influyen de manera directa en el proceso de desarrollo de software. Adyacente al desarrollo de cada proyecto, la disponibilidad de información histórica es otro elemento que determina el riesgo de la estimación con base en el análisis de los factores antes mencionados a partir del comportamiento de éstos.

La planeación efectiva de un proyecto de desarrollo de software depende de la planeación detallada, anticipando problemas que puedan surgir y preparando soluciones tentativas a ellos. La asignación de recursos durante la etapa de la planeación de un proyecto de desarrollo de software, sin embargo, no cuenta con un indicador que permita evaluar la complejidad del mismo con base en el alcance, el tiempo de desarrollo y el costo para determinar un equilibrio entre estos factores ni se contempla el comportamiento histórico en el desarrollo de proyectos para determinar con mayor exactitud la combinación de dichos factores y el comportamiento de éstos en el transcurso del desarrollo del proyecto.

En Ruiz-Vanoye et al (2011), se hace mención que la complejidad computacional contiene diversos elementos tales como las clases de complejidad de los problemas (P, NP, NP-completo y NP-completo, y otros), la complejidad de los algoritmos (es una manera de clasificar qué tan eficiente es un algoritmo mediante el tiempo de ejecución para resolver un problema con la entrada del peor caso), la complejidad de los casos (que son las medidas de la complejidad computacional para determinar el conjunto de las instancias de problemas), y otros elementos. La teoría de la Complejidad Computacional de Instancias (CPI) es una medida de la complejidad computacional (tiempo y espacio) para determinar el conjunto de las

instancias de problemas. La CPI podría ser llamada como la complejidad computacional de instancias o la complejidad de los casos. La Complejidad computacional de los casos es una medida de la complejidad computacional relacionada con los casos individuales (la especificación de determinados valores en los parámetros de un problema (Garey y Johnson, 1979)) de una cadena x con respecto a un conjunto A y en un plazo de tiempo determinado t (Knuth, 1976).

Con la determinación de la complejidad de las instancias (planteamiento inicial del proyecto en la etapa de análisis de requerimientos), permitirá al líder del proyecto, identificar la posibilidad de llevar a cabo un proyecto con base en los recursos que pueden emplearse en el desarrollo de dicho proyecto.

Alcances y limitaciones

Características de la investigación:

- Área: proyectos de software.
- Enfoque del modelo de investigación: relacional.
- Tipo de investigación: exploratoria.
- Enfoque de investigación: deductivo.
- Estrategia de investigación: teórica – empírica.
- Métodos de recolección de datos: instancias PSP Software.
- Objeto de estudio: indicadores estratégicos para determinar la complejidad de proyectos de desarrollo de software.

Viabilidad de la investigación

El fundamento para el desarrollo del presente trabajo de investigación se basa en los siguientes supuestos:

Para el desarrollo de proyectos de software se requiere realizar una planeación que involucre tiempos de entrega, costos, actividades y tiempos parciales considerando los microciclos en la producción de software. Actualmente no existe una metodología, modelo ó método que permita definir de manera inicial la complejidad para optimizar los recursos involucrados en el

proceso de desarrollo de software y realizar revisiones periódicas sobre el balance del tiempo de desarrollo y tiempo dedicado a actividades que se relacionan de forma indirecta con éste ó con actividades del área de informática no relacionadas con programación.

Impacto esperado

- **Desarrollo humano sustentable:** La presente investigación está enfocada a determinar la complejidad del proyecto y ofrecer una sugerencia con respecto al personal requerido para cada proyecto con base en sus habilidades relacionadas con las actividades del proyecto. Este resultado permitirá generar una aproximación de los recursos que serán empleados en el desarrollo de cada proyecto, partiendo de una estimación derivada de las características del personal, de los resultados en proyectos anteriores y de la complejidad determinada a partir del modelo propuesto en la presente investigación.
- **Proveer Información confiable, oportuna y veraz:** Proveer información confiable, oportuna y veraz basada en el resultado de la aplicación del modelo propuesto en la presente investigación con datos obtenidos de instancias desarrolladas basadas en la función objetivo para obtener la combinación óptima de las variables.
- **Optimización de recursos:** Con base en el desarrollo de la presente investigación, generar un modelo para determinar la complejidad del proyecto y establecer la correlación entre los indicadores propuestos para minimizar la desviación de costo, tiempo de desarrollo y alcance planeados.

Organización del Estudio

Las secciones principales en el documento de tesis se presentan en la figura 1:

1. Propósito y organización.
2. Marco teórico
3. Metodología de investigación
4. Experimentación y análisis de resultados
5. Conclusiones y trabajos futuros

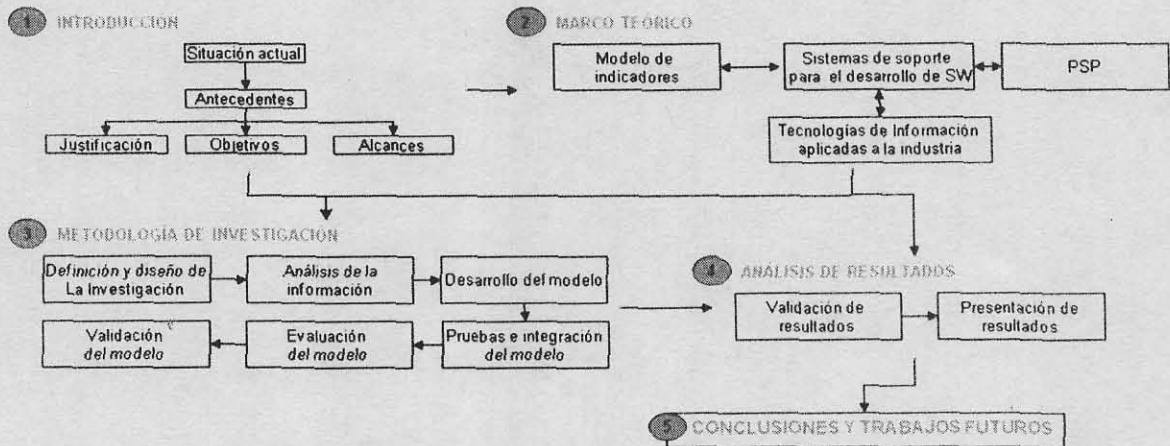


Figura 1 Estructura de título

Fuente: Elaboración propia

NOTA:

- 1 – Introducción:** situación actual y antecedentes relacionados con el problema a analizar en el presente trabajo y, a partir de esta información la generación de la justificación, objetivos y alcance de la investigación,
- 2 – Marco teórico:** antecedentes teóricos para definir el modelo de investigación propuesto.
- 3 – Metodología de la investigación:** presenta el diseño de la investigación, análisis de datos y desarrollo, validación, evaluación y pruebas e integración del modelo.
- 4 – Experimentación y análisis de resultados:** análisis de resultados mediante el uso de técnicas estadísticas y/o de inteligencia artificial.
- 5 – Conclusiones y Trabajos futuros:** conclusiones obtenidas en relación al análisis de los resultados, la aportación y los trabajos futuros relacionados con el presente trabajo.

CAPÍTULO 2. MARCO TEÓRICO

2.1 Antecedentes

2.2 Modelo de Indicadores

2.2.1 Planeación estratégica

El término planeación estratégica se ha enfocado a describir un conjunto de actividades y procesos. Las características de la planeación estratégica en el sentido de (Keller, 1983), destaca lo siguiente:

- La toma de decisiones estratégica en cualquier institución es llevada a cabo por los dirigentes como agentes activos en el proceso histórico de cada institución.
- La planeación estratégica considera el entorno cambiante externo y orienta el desarrollo institucional.
- Se centra en las decisiones, análisis, previsiones y metas.
- Se centra en el destino de la organización a partir del análisis histórico del comportamiento institucional y del entorno empresarial.

En términos de (Keller et al, 1988) se detallan 5 principios relacionados con planeación estratégica:

- Análisis del entorno: Información externa que afecta a la institución y definir las oportunidades y amenazas del entorno de la institución
- Análisis del ámbito interno para identificar fortalezas y debilidades.
- Declaración de la misión que justifica la razón de ser de la empresa
- Objetivos y acciones estratégicas institucionales.
- Objetivos y estrategias de las unidades
- Presupuesto
- Control

2.2.2 Indicadores

Los indicadores son una medida del efecto de la operación del sistema sobre el descriptor (Torquebiau, 1989); si el sistema es sostenible, tiene un efecto positivo sobre el descriptor (características significativas de un elemento de acuerdo con los principales atributos de sostenibilidad de un sistema determinado), caso contrario, un efecto negativo. Por otra parte, (Adriaanse, 1993) distingue tres funciones principales de los indicadores: "...simplificación, cuantificación y comunicación. Por lo general, los indicadores simplifican para hacer cuantificables los fenómenos complejos."

Los indicadores están integrados por una o varias variables que son elementos de un función para el cálculo del indicador; datos que son valores que integran a las variables definidas para dicho indicador; estadísticas que presentan los datos en una forma ordenada, tanto de datos de las variables como datos de salida de cada indicador; e índices que son la relación entre los datos de una variable en diferentes puntos de tiempo.

(Torquebiau, 1990) establece como norma para la definición de indicadores algunos elementos:

- Interpretación del significado del indicador.
- Definir qué, cómo y cuándo medir.
- Definir los insumos requeridos para el cálculo.
- Establecer las limitaciones del indicador.
- Interpretar los resultados de los valores del indicador considerando las limitaciones que tiene.
- Una clasificación del impacto positivo o negativo (alto, medio ó bajo).
- Una definición de los valores extremos del indicador.

(Ruitenbeek, 1991) identifica cinco fundamentos para la generación de indicadores como soporte de decisiones:

- El indicador estándar es útil para propósitos descriptivos y está focalizado a requerimientos a corto plazo.
- El indicador a seleccionar debe informar como afecta a su entorno.
- Aplicaciones proyectivas; los indicadores deben describir el pasado y permitir realizar proyecciones del comportamiento de los datos.

El uso de indicadores tiene como objetivo principal describir aspectos importantes sobre el proceso de toma de decisiones. Sin embargo, dichos indicadores deben definirse en función del tipo de decisión que se tomará. A partir de los datos generados de los indicadores, se puede analizar la situación actual para identificar los puntos críticos.

2.2.3 Modelo de indicadores

Ante la necesidad de definir un sistema de indicadores, es necesario definir las variables principales y las bases para establecer los indicadores. El análisis de la aplicación de indicadores en el tema de desarrollo de proyectos de software permite identificar las variables principales, las cuales se orientan a recursos humanos, materiales y económicos (costo), principalmente. A partir de la definición de variables, es necesario desarrollar un esquema para definir los indicadores.

Los pasos principales para la definición de indicadores se describen en (De Camino, 1991):

- El primer paso para la definición de indicadores es identificar las categorías de análisis desde el punto de vista del desarrollo de proyectos de software. Las categorías identificadas son las siguientes: base de recursos, operación, recursos de entrada y recursos de salida.
- Como segundo paso, definir los elementos que integran las categorías de análisis.
- Definir descriptores e indicadores, las condiciones para cumplir con los indicadores y llevar a cabo el análisis de los casos particulares para cada indicador.
- Definir los criterios para la definición de indicadores y el rango de datos por indicador.

2.3 Sistemas de soporte para el desarrollo de proyectos de software

2.3.1 PMBOK (Project Management Body of Knowledge)

(Newell, 2010) describe la administración de proyectos como la aplicación de conocimiento, habilidades y técnicas aplicadas a los proyectos en un orden específico para cumplir con las necesidades de las personas involucradas (stakeholders) cuyo comportamiento es temporal (inicio – fin) con el objetivo de generar un producto o servicio. El inicio del proyecto es determinado por la entrega de la documentación inicial denominado “Project charter”, cuyo objetivo es reunir los costos y expectativas iniciales del mismo. El fin del proyecto es determinado en el momento en que se han cumplido todos los objetivos y se han entregado los productos o servicios relacionados con dicho proyecto; sin embargo, en ocasiones los proyectos no se terminan por diferentes causas. En la figura 2 se representan los grupos de procesos de la administración de proyectos con base en PMI.



Figura 2 Grupo de procesos para la administración de proyectos en PMI.

(Cleland and Ireland, 2007) presenta una relación de funciones ó procesos representativos de la administración de proyectos (Figura 3), cuya división se basa en la planeación organización y en la planeación que debe ser considerada por equipos de trabajo:

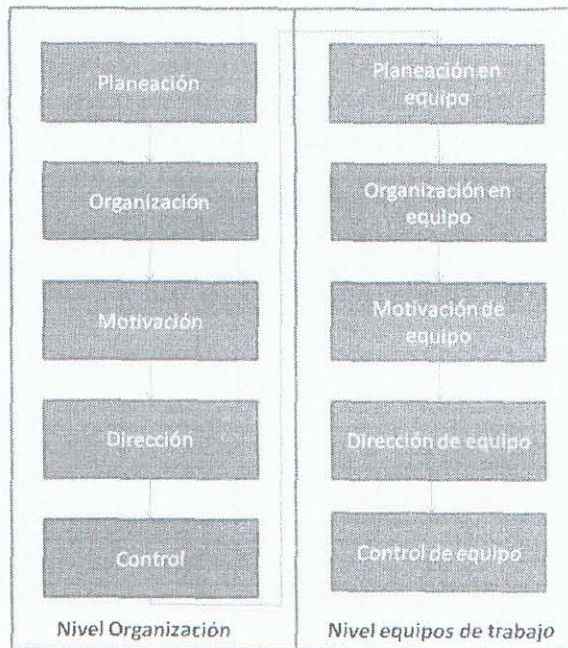


Figura 3. Funciones y procesos representativos de la administración de proyectos

Project Management Institute (PMI), propone una estructura para el desarrollo de proyectos que consiste en las siguientes etapas con base en PMI (2003) y presenta el mapeo de los procesos organizándolos por área de conocimiento mostrado en la tabla 1:

- **Inicialización del proyecto:** La inicialización de un proyecto parte de la necesidad, de una problemática ó de un requerimiento de negocio, considerados en PMI las demandas de mercado, necesidades de negocio, solicitudes del cliente, avances tecnológicos, requerimientos legal ó necesidades sociales, que conforman el tema principal del proyecto. En esta etapa es conveniente definir las características del proyecto relacionado con la necesidad detectada cuyas características se basan en el “Esquema de desarrollo de la descripción del proyecto”, provisto en (PMI, 2003). Para la autorización formal del proyecto PMI establece el formato correspondiente a la cédula del proyecto donde se especifica: el nombre del proyecto, la sinopsis, propósito, necesidad del negocio, la descripción de los entregables y productos, los supuestos, riesgos y alcances y limitaciones, los recursos disponibles y requeridos, el enfoque del proyecto, los medios de comunicación y reporte, los criterios de aceptación del

proyecto por parte de los patrocinadores y la administración de cambios. La cédula del proyecto deber ser firmada por el administrador del proyecto y el patrocinador.

- Planeación del proyecto: La metodología para planeación de un proyecto es un enfoque estructurado empleado para guiar al equipo del proyecto durante el desarrollo del plan del proyecto (documento para administrar la ejecución del proyecto), donde se definen: la cédula del proyecto, descripción de la estrategia ó enfoque de administración del proyecto, objetivos y entregables, WBS (por sus siglas en inglés Work Breakdown Structure), estimación de costos, periodo de desarrollo de proyecto y por actividad, asignación de responsabilidades a través del WBS, medición del desempeño, hitos principales y fechas de entrega, plan de gestión de riesgos, la administración de la calidad, la administración de los recursos humanos, la administración de la comunicación del proyecto y la administración de las contrataciones. Las plantillas para la planeación de proyecto se proponen en (PMI, 2003).
- Ejecución del proyecto: la ejecución del proyecto está integrada procesos clave y procesos auxiliares: ejecución del plan del proyecto (desarrollo de las actividades incluidas en el proyecto), aseguramiento de la calidad (evaluación del desempeño general del proyecto para asegurar que se cumplan con los estándares de calidad establecidos), equipo de desarrollo (asignación del personal con base en sus competencias/habilidades para cumplir con el desempeño del proyecto), distribución de la información (establecer los medios de difusión de la información para su disponibilidad, en tiempo y forma, por parte de los stakeholders del proyecto), selección de fuentes y administración de la contratación.
- Control del proyecto: en la etapa de monitoreo, se identifican las variaciones que se tienen en el desarrollo del proyecto con respecto a la planeación del mismo. El control del proyecto incluye acciones preventivas como anticipación a los posibles problemas y las acciones correctivas a los problemas detectados. Los procesos que se consideran son: control integrado de cambios (coordinación de cambios durante el desarrollo del proyecto), verificación del alcance (aceptación formal del alcance del proyecto y de la conclusión de las etapas del mismo), control de la programación (control de cambios en la programación del proyecto), Control de costos (control de cambios en el

presupuesto), control de calidad (monitoreo de los resultados del proyecto para determinar si cumplen con los estándares de calidad e identificar formas para eliminar las causas de desempeño insatisfactorio); reporte de desempeño (colecta y difusión de información relacionada con el desempeño a través de: reportes de estatus y medición del alcance), y control y monitoreo de riesgos (identificar los riesgos, monitoreo de éstos, aseguramiento de la ejecución del plan de riesgos y evaluación de la efectividad en la reducción de riesgos).

- Cierre del proyecto: en esta etapa se lleva a cabo la revisión de la documentación del proyecto para analizar el desempeño del proyecto incluyendo la documentación de planeación que estableció el marco de referencia para medir el desempeño. Al final de esta etapa, se debe confirmar que el proyecto ha cumplido con los requerimientos de todos los clientes, es decir se ha llevado a cabo la aceptación formal de los resultados, entregables y requerimientos por parte del comité de liberación a través del formato de aceptación y cierre del proyecto.

Para Snyder (2010), PMI está integrado por 42 procesos distribuidos en las áreas de conocimiento que conforman el desarrollo de proyectos con base en esta metodología y propone una descripción de dichos procesos.

Tabla 1 Procesos de administración de proyectos en grupos de procesos y áreas de conocimiento

Grupo de procesos : Área de conocimiento	Inicialización	Planeación	Ejecución	Control	Cierre
Administración de la integración del proyecto		Desarrollo del plan del proyecto	Ejecución del plan del proyecto	Control integral de cambios	
Administración del alcance del proyecto	Inicialización	Alcance del proyecto Definición del alcance		Verificar alcance Control de cambios en el alcance	
Administración del tiempo del proyecto		Definición de las actividades Definición de la secuencia		Control de la programación	

		Estimación de la duración de las actividades Desarrollo de la programación de actividades			
Administración del costo del proyecto		Planeación de recursos Estimación del costo Presupuesto del costo		Control de costos	
Administración de la calidad del proyecto		Planeación de la calidad	Aseguramiento de la calidad	Control de la calidad	
Administración de recursos humanos del proyecto		Planeación organizacional Contratación del staff	Desarrollo del equipo		
Administración de las comunicaciones del proyecto		Planeación de la comunicación	Distribución de la información	Reportes de desempeño	Cierre administrativo
Administración de riesgo del proyecto		Planeación de la administración del riesgo Identificación del riesgo Análisis cuantitativo del riesgo Planeación de la respuesta ante el riesgo		Monitoreo y control del riesgo	
Gestión de adquisiciones		Planeación de adquisiciones Solicitud de adquisiciones	Solicitudes Selección de fuentes Administración de contratos		Cierre de contrataciones

2.3.2. IT Infraestructura Library (ITIL)

ITIL se presenta como una “buena práctica” basado en el ciclo de vida del servicio y en los conceptos relacionados con “servicio” y “valor” (Van-Bon et al, 2008). Se basa en el concepto de transición de servicio (Van-Bon et al, 2008) que comprende los siguientes pasos:

- Planificación y soporte de la transición: los recursos se planean y coordinan para cumplir con las especificaciones de diseño del servicio.

- Gestión de cambios: Garantiza que los cambios se aplican de forma controlada.
- Gestión de la configuración y activos de servicio (SACM): gestiona los activos de servicio y elementos de configuración (CI) para dar soporte a los demás procesos.
- Gestión de entregas y despliegues: concentrado en construir, probar y desplegar los servicios especificados en el diseño del servicio, y en garantizar que el cliente/usuario puedan utilizar el servicio de manera adecuada.
- Validación y pruebas de servicio: las pruebas garantizan que los servicios nuevos ó modificados se “ajusten al propósito y uso” predeterminados.
- Evaluación: verificar el rendimiento del servicio.
- Gestión del conocimiento: mejorar la calidad de la toma de decisiones (de la dirección) garantizando la disponibilidad de la información segura y fiable durante el ciclo de vida del servicio.

La operación del servicio es responsable de que se ejecuten los procesos que optimizan los costes y la calidad del servicio en el ciclo de vida de la gestión del servicio. La operación del servicio tiene que contribuir a que el cliente (negocio) logre sus objetivos, y debe garantizar el funcionamiento eficaz de los componentes que dan soporte al servicio. Los procesos de operación del servicio son:

- Gestión de eventos: supervisa todos los eventos que se producen en la infraestructura de TI con el fin de monitorizar el rendimiento.
- Gestión de incidencias: se concentra en restaurar el fallo del servicio lo antes posible para los usuarios, de manera que su impacto sobre el negocio sea el mínimo.
- Gestión de problema: Incluye todas las actividades necesarias para diagnosticar las causas subyacentes a las incidencias y para encontrar una solución a dichas incidencias.
- Gestión de peticiones: Se encarga del tratamiento de peticiones de servicio de los usuarios, proporcionando un canal de solicitud, información y ejecución de la petición.
- Gestión de accesos: Permite utilizar el servicio a los usuarios autorizados y limita el acceso a los usuarios sin autorización.

- Monitorización y control: se basa en un ciclo de continuo de monitorización, reporte y actuación.
- Operaciones de TI: actividades para gestionar la infraestructura de TI.
- Gestión de instalaciones y centros de datos: la gestión del entorno físico de las operaciones de TI, normalmente en centros de proceso de datos.

Como se explica en Fuentes-Penna et al (2010), ITIL ofrece una guía de las mejores prácticas para la administración de servicios de las Tecnologías de Información (TI) y las interacciones entre diversos procesos para el soporte de servicios. ITIL propone una solución para la administración de las TI como niveles de servicio: incidentes y problemas. En Fuentes (2010), se expone la descripción de la figura 4 con base en la metodología de soporte de servicio, donde los incidentes y problemas que se presentan a nivel TI, afectan las operaciones del negocio por su estrecha interdependencia.

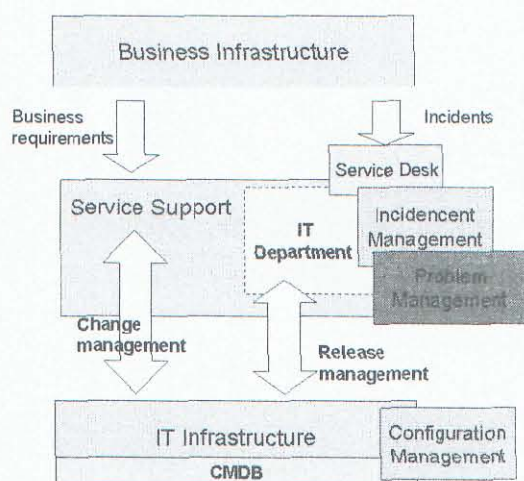


Figura 4 Estructura de ITIL

2.3.3. Moprosoft

MoProSoft (Modelo de Procesos para la Industria del Software) es el modelo de procesos para la industria mexicana de Software diseñado para medir la capacidad de los procesos que siguen las empresas y para garantizar una calidad constante en los desarrollos y

mantenimiento de software. Se basa principalmente en estándares internacionales como: ISO 9000, ISO 15504, SW-CMM y CMM-I.

MoProSoft es un modelo de calidad que permite a las empresas de desarrollo de software, el acceso a las prácticas de Ingeniería de Software de clase mundial. La adopción de MoProSoft habilita la obtención de un certificado ISO 9000.

El propósito de MoProSoft, con base en Itera (2010), es apoyar a las organizaciones en la estandarización de sus prácticas, en la evaluación de su efectividad y en la integración de la mejora continua. Los procesos abarcan las responsabilidades asociadas a la estructura de una organización que son: la Alta Dirección, Gestión y Operación.

El desarrollo y mantenimiento de software se lleva a cabo a través de una serie de actividades realizadas por equipos de trabajo. La Ingeniería de Software se ha dedicado a identificar las mejores prácticas para realizar estas actividades recopilando las experiencias exitosas de la industria de software a nivel mundial. Estas prácticas se han organizado por áreas de aplicación, y se han dado a conocer como áreas clave de procesos, en caso de CMM, o como procesos de software en ISO/IEC 15504. El modelo apoya a las organizaciones en la estandarización de sus prácticas, en la evaluación de su efectividad y en la integración de la mejora continua.

Las prácticas de planeación, seguimiento y evaluación se incluyeron en todos los procesos de gestión y administración; por su parte los objetivos, los indicadores, las mediciones y las metas cuantitativas fueron incorporados de manera congruente y práctica en todos los procesos; las verificaciones, validaciones y pruebas están incluidas, de manera explícita, dentro de las actividades de los procesos; y existe una base de conocimientos que resguarda todos los documentos y productos generados.

Con base en Moprosoft (2006), los requerimientos de una empresa para poder aplicar el modelo de procesos de MoProSoft son:

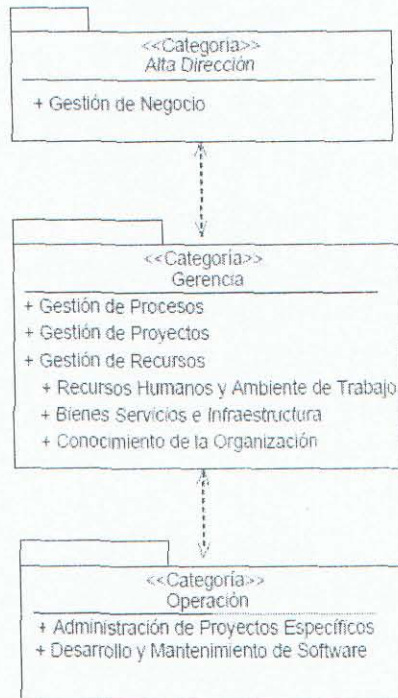


Figura 5 Diagrama de categorías de procesos de MoProSoft

Alta dirección: son los procesos relacionados con la gestión de negocios (misión, visión, valores, objetivos de la organización, definición de los procesos con indicadores y metas, definir la estructura organizacional y estrategias de recursos, base del conocimiento de la organización, presupuesto, periodicidad de valoración del plan estratégico, plan de comunicación con los clientes, cartera de proyectos, entre otros). Provee los lineamientos para los procesos corporativos y recopila retroalimentación de la información generada en toda la organización. Los indicadores que contempla esta categoría son: el desempeño de los indicadores de los objetivos del plan estratégico, conocimiento del plan estratégico por parte de los miembros de la organización y propuestas de mejora en función de los reportes de valoración. En la figura se presenta el flujo de trabajo de la alta dirección relacionado con sus actividades principales

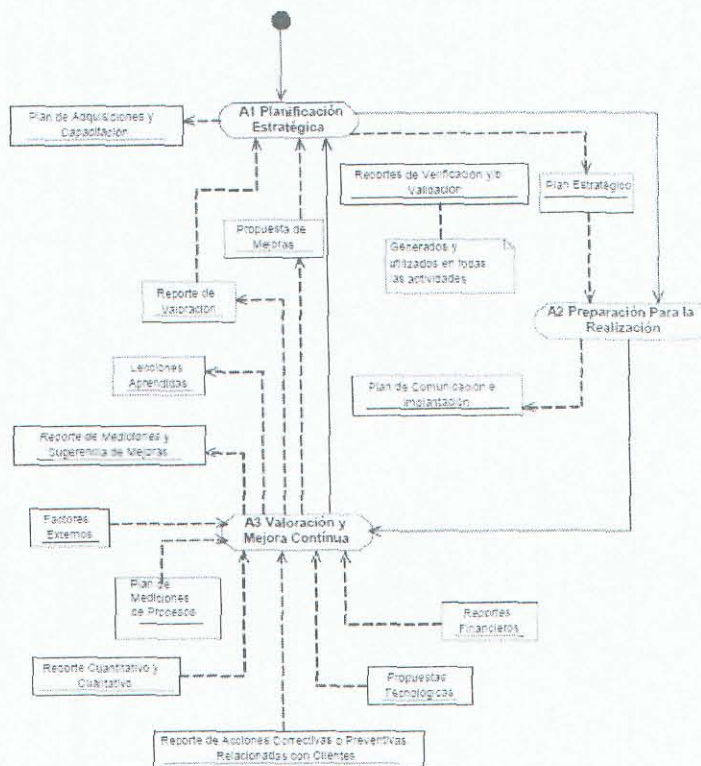


Figura 6. Flujo de trabajo de la alta dirección.

- Dirección ó gerencia: Esta categoría se orienta hacia las prácticas de gestión de los procesos, de los proyectos y de los recursos (constituido por los subprocesos de Recursos Humanos y Ambiente de Trabajo, Bienes Servicios e Infraestructura y Conocimiento de la Organización), basándose en los lineamientos propuestos por la alta dirección en el plan estratégico. Provee los elementos para el funcionamiento de los procesos en la categoría de operaciones, recibe y evalúa la información de esta categoría y comunica los resultados a la alta dirección.
 - o El proceso de Gestión de Procesos (figura 7) contempla los siguientes indicadores: plan de procesos con los procesos requeridos identificados en el plan estratégico; Actividades de definición, implantación y mejora de los procesos conforme a lo establecido en el plan de procesos; los miembros de la organización conocen los procesos correspondientes y trabajan en función de éstos; documentación y actualización de los procesos de la organización; Plan de mejora definido en función de las sugerencias de mejora; los procesos

cumplen con las metas cuantitativas; y la entrega del reporte cuantitativo y cualitativo es entregado en las fechas definidas a la gestión de negocios. se compone de las siguientes actividades:

- Planificación de procesos. A partir de los procesos requeridos identificados en el plan estratégico, en el Plan de Mejora y en el Plan de acciones, la planificación de procesos establece o actualiza un plan de procesos que contiene: La definición de elementos de procesos con base en los modelos de procesos de referencia; calendario para establecer o mejorar procesos relacionando actividades y responsables, plan de adquisiciones y capacitación, solicitudes de personal capacitado, proveedores, infraestructura y herramientas, así como requerimientos de capacitación; plan de evaluación de procesos, que incluye evaluaciones internas y externas; plan de mediciones de procesos en el cual se especifican los tipos de mediciones, la periodicidad, la responsabilidad; y plan de manejo de riesgos de procesos que contiene la identificación y evaluación de riesgos, así como los planes de contención y de contingencia correspondientes.
- Preparación a la implantación. Consiste en realizar: la asignación de los responsables de procesos; documentación o actualización de la documentación de los procesos de la organización de acuerdo a la definición de elementos de procesos establecida; capacitación a los miembros de la organización en los procesos, de acuerdo al plan de adquisiciones y capacitación; y la implantación de los procesos en proyectos piloto, en caso de considerarse conveniente.
- Evaluación y control. Las tareas correspondientes son: seguimiento a las actividades del plan de procesos; recolección de reportes de mediciones y sugerencias de mejora, generando el reporte cuantitativo y Cualitativo que se entrega al responsable de gestión de negocio; el análisis de las sugerencias de mejora contribuye a la generación del plan de mejora; ejecución del Plan de evaluación, con la finalidad de

verificar la implantación de los procesos, recopilando los hallazgos y oportunidades de mejora documentados en el reporte de evaluación, en el plan de acciones que den respuesta a los hallazgos complementados con el plan de mejora de acuerdo a las oportunidades detectadas; seguimiento al plan de acciones; supervisión y control de los riesgos identificados en el plan de manejo de riesgos; e identificación y documentación de las lecciones aprendidas.

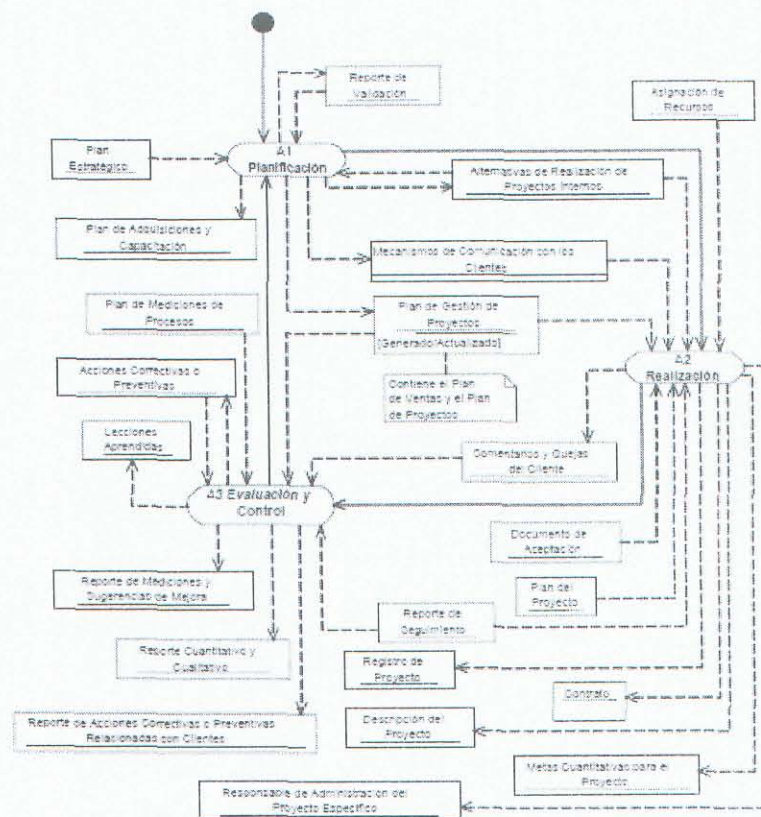


Figura 7. Diagrama de flujo de trabajo de la gestión de proceso.

- Gestión de recursos (figura 8): su propósito es conseguir y dotar a la organización de recursos humanos, infraestructura, ambiente de trabajos, proveedores y generar la base de conocimiento de la organización. Los indicadores que se contemplan el a gestión de recursos son: grado de

satisfacción de los responsables de los procesos y proyectos con respecto a la disponibilidad de los recursos solicitados; grado de satisfacción de los responsables de los procesos y proyectos con respecto a la calidad de los recursos entregados; relación entre gasto presupuestado y gasto real de los recursos entregados; grado de satisfacción de los usuarios de la base de conocimiento de la organización; y entrega de propuestas tecnológicas al responsable de la gestión de negocio en las fechas establecidas. Las actividades que rigen a este proceso son:

- Planificación de Recursos: se establece a partir del plan estratégico y plan de adquisiciones y capacitación de los procesos y proyectos. Como resultado se obtienen: el plan operativo de recursos humanos y ambiente de trabajo, plan operativo de bienes, servicios e infraestructura y el plan operativo de conocimiento de la Organización.
- Seguimiento y Control: Se da seguimiento a la ejecución de los planes operativos de cada uno de los subprocesos considerando el reporte de recursos humanos disponibles, capacitación y ambiente de Trabajo, el reporte de bienes, servicios e infraestructura y el reporte del estado de la base de conocimiento. Con base en los reportes antes mencionados, se genera el reporte cuantitativo y cualitativo que incluye información sobre recursos disponibles y adquiridos de acuerdo al plan de comunicación e implantación. Y, con base en plan de mediciones de procesos se genera el reporte de mediciones y sugerencias de mejora.
- Investigación de Tendencias Tecnológicas: Se lleva a cabo en función del plan estratégico, para realizar un análisis prospectivo y de viabilidad dirigido al grupo directivo. Como resultado se obtienen propuestas tecnológicas.



- 24

criterios; elaboración del plan de capacitación; elaboración de los formularios para la evaluación de desempeño y para la encuesta sobre el ambiente de trabajo.

- Instrumentación: selección, asignación y aceptación de los recursos humanos (como resultado se tiene la Asignación de Recursos); capacitación de recursos humanos de acuerdo a las necesidades actuales y futuras de los procesos y proyectos (como resultado se deberá generar el reporte de capacitación); evaluación periódica del desempeño de los recursos humanos asignados a los procesos y proyectos; y evaluación del ambiente de trabajo (como resultado se genera el reporte de ambiente de trabajo).
 - Generación de Reportes: reporte de recursos humanos disponibles, capacitación y ambiente de trabajo; y reporte de mediciones y sugerencias de mejora.
- Bienes, servicios e infraestructura (figura 10): proporciona los proveedores de bienes, servicios e infraestructura que satisfagan los requisitos de adquisición de los procesos y proyectos. Los indicadores que se contemplan son: grado de satisfacción de los solicitantes de bienes y servicios recibidos para conocer la eficacia en la selección de los proveedores; y grado de cumplimiento de las actividades de mantenimiento planificadas. Las actividades que se contemplan en este proceso son:
- Preparación: revisión del plan operativo de bienes, servicios e infraestructura y de acciones correctivas; definición de criterios; elaboración del plan de mantenimiento; y obtención de la solicitud de bienes o servicios.

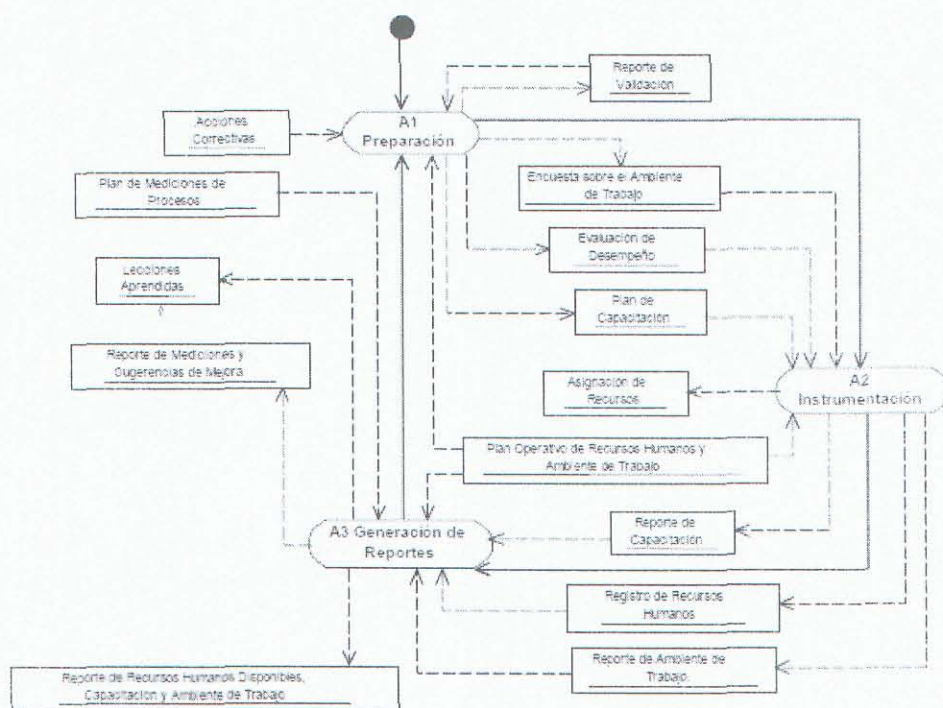


Figura 9. Diagrama de flujo de la gestión de recursos humanos y ambiente de trabajo.

- Instrumentación: selección de proveedores y adquisición de bienes y servicios (como resultado se obtiene el registro de bienes o servicios y la actualización del catálogo de proveedores); evaluación periódica de la satisfacción del solicitante, por el bien o servicio adquirido y su registro en el catálogo de proveedores; y mantenimiento de la infraestructura.
- Generación de reportes: reporte de bienes, servicios e infraestructura; y reporte de mediciones y sugerencias de mejora.

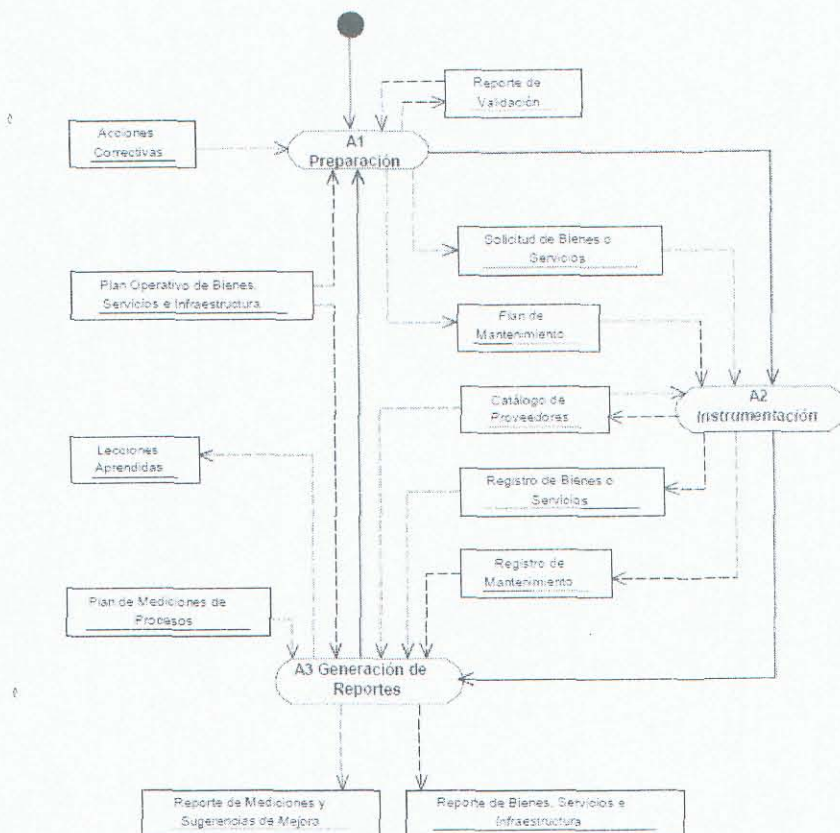


Figura 10. Diagrama de flujo de trabajo de la gestión de bienes, servicios e infraestructura.

- Conocimiento de la organización (figura 11): su objetivo es mantener disponible y administrar la base de conocimiento que contiene la información y los productos generados por la organización. Los indicadores contemplados son: Grado de satisfacción de los usuarios con respecto a los mecanismos de alimentación, modificación y mantenimiento de la Base de Conocimiento; grado de satisfacción de los usuarios con respecto a los mecanismos de consulta de la Base de Conocimiento; y grado de satisfacción de los usuarios con respecto a los mecanismos de control de acceso y respaldo de la Base de Conocimiento. En función del plan operativo de conocimiento de la organización y acciones correctivas de gestión de recursos se realizan las siguientes actividades:

- Planificación: establecer el plan de administración de la base de Conocimiento que contenga la descripción de actividades para la

definición o modificación del modelo conceptual de la Base de Conocimiento (BC), usuarios y sus requerimientos, así como los mecanismos de operación, mantenimiento, verificación, validación en función de los requerimientos de los usuarios.

- Alimentación, consulta, mantenimiento y respaldo para cada tipo de repositorio.
- Realización: Establecimiento del diseño de la BC de la organización, está constituido por el modelo conceptual, incluyendo su metamodelo, y por los mecanismos de operación. En función de los requerimientos de los procesos, la Base de Conocimiento está compuesta por los siguientes repositorios:
 - Negocio: documentación utilizada y generada en el proceso de gestión de negocio.
 - Procesos: documentación utilizada y generada en el proceso de gestión de procesos.
 - Proyectos: documentación utilizada y generada en el proceso de gestión de proyectos y administración de proyectos específicos.
 - Desarrollo y mantenimiento: productos de software generados en el proceso de desarrollo y Mantenimiento de software.
 - Recursos: documentación utilizada y generada en el proceso de gestión de recursos.
 - Recursos humanos: documentación utilizada y generada en el subproceso de recursos humanos y ambiente de trabajo.
 - Bienes adquiridos y proveedores: documentación utilizada y generada en el subproceso de Bienes, Servicios e Infraestructura.
 - Documentación BC: documentación utilizada y generada acerca de su estructura, contenido y operación.
 - Conocimiento tecnológico (terminología, conceptos, metodologías).

- Bibliotecas de reuso.

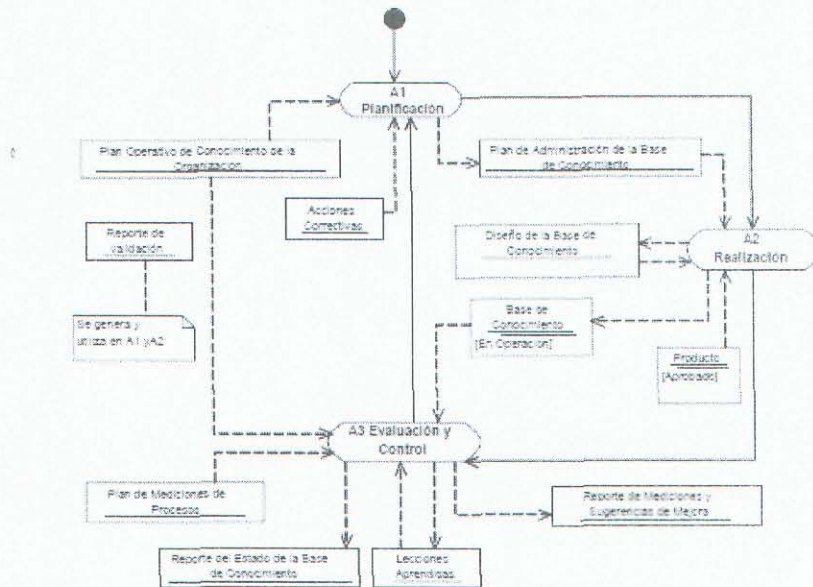


Figura 11. Diagrama de flujo de trabajo de la gestión del conocimiento de la organización.

- Operaciones: contempla las prácticas de proyectos de desarrollo de software y mantenimiento. Esta categoría desarrolla sus actividades acorde a los elementos provistos por la dirección, genera información relacionada con los productos desarrollados y provee los productos terminados.
 - Administración de proyectos específicos (figura 12): Tiene como objetivo establecer y llevar a cabo sistemáticamente las actividades que permitan cumplir con los objetivos de un proyecto en tiempo y costos esperados. Los indicadores que se contemplan son: El plan del proyecto y el plan de desarrollo contemplan a los objetivos establecidos en la descripción del proyecto y a las metas cuantitativas para el proyecto; las actividades del proyecto se realizan conforme a lo establecido en el plan del proyecto y en el plan de desarrollo; el tiempo y costo real están acordes con lo estimado; las reuniones de avance del proyecto se realizan conforme a lo acordado con el

cliente; y el mecanismo de recepción y análisis se aplica a todas solicitudes de cambios. Las actividades que se contemplan en este proceso son:

- **Planificación:** conjunto de actividades cuya finalidad es obtener y mantener el plan del proyecto y el plan de desarrollo que regirán al proyecto específico, con base en la Descripción del Proyecto. Para la generación de este plan se realizan las siguientes tareas: Definir el proceso específico con base en la descripción del proyecto y el proceso de desarrollo y mantenimiento de software de la organización o con base en el acuerdo con el Cliente; definir el protocolo de entrega con el cliente; definir ciclos y actividades con base en la descripción del proyecto y en el proceso específico; determinar el tiempo estimado para cada actividad, considerando las Metas Cuantitativas para el Proyecto; elaborar el plan de adquisiciones y capacitación para obtener recursos humanos capacitados y adquirir materiales, equipo y herramientas para llevar a cabo el proyecto; establecer el equipo de trabajo que realizará el proyecto; establecer el calendario de las actividades; calcular el costo estimado del proyecto; definir el plan de manejo de riesgos; documentar el plan del proyecto; documentar el plan de desarrollo; y formalizar el inicio de un nuevo ciclo del proyecto.
- **Realización:** Consiste en llevar a cabo las actividades del plan del proyecto de acuerdo a: acordar las tareas del equipo de trabajo con el responsable de desarrollo y mantenimiento de software; acordar la distribución de la información al equipo de trabajo; revisar con el responsable de desarrollo y mantenimiento de software la descripción del producto, el equipo de trabajo y el calendario; revisar el cumplimiento del plan de adquisiciones y capacitación; administrar subcontratos; recolectar los reportes de actividades, reportes de mediciones y sugerencias de mejora y productos de trabajo; revisar el registro de rastreo en función de los productos de trabajo recolectados; revisar los productos terminados durante el proyecto; recibir y analizar

las solicitudes de cambios del cliente; realizar reuniones con el equipo de trabajo y con el cliente para reportar el avance del proyecto y tomar acuerdos; • Evaluación y Control: Consiste en asegurar que se cumplan los objetivos del proyecto. Se supervisa y evalúa el progreso para identificar desviaciones y realizar acciones correctivas, cuando sea necesario. Dentro de esta actividad se realizan las siguientes tareas: evaluar el cumplimiento del plan del proyecto y plan de desarrollo; analizar y controlar los riesgos; y generar el reporte de seguimiento del proyecto. Como resultado de estas actividades se tiene el plan del proyecto y el plan de desarrollo actualizados.

- Cierre: Consiste en entregar los productos de acuerdo a un Protocolo de Entrega y dar por concluido el ciclo o proyecto. Como resultado se obtiene el documento de aceptación del cliente. Se realizan las siguientes tareas: formalizar la terminación del proyecto o de un ciclo; llevar a cabo el cierre del contrato con subcontratistas; y generar el reporte de mediciones y sugerencias de mejora.



- En la **Definición general** del proceso se identifica su nombre, categoría a la que pertenece, propósito, descripción general de sus actividades, objetivos, indicadores, metas cuantitativas, responsabilidad y autoridad, subprocesos en caso de tenerlos.

procesos relacionados, entradas, salidas, productos internos y referencias bibliográficas.

- En las **Prácticas** se identifican los roles involucrados en el proceso y la capacitación requerida, se describen las actividades en detalle, asociándolas a los objetivos del proceso, se presenta un diagrama de flujo de trabajo, se describen las verificaciones y validaciones requeridas, se listan los productos que se incorporan a la base de conocimiento, se identifican los recursos de infraestructura necesarios para apoyar las actividades, se establecen las mediciones del proceso, así como las prácticas para la capacitación, manejo de situaciones excepcionales y uso de lecciones aprendidas.
- En las **Guías** de ajuste se sugieren modificaciones al proceso que no deben afectar los objetivos del mismo.

A su vez, Moprosoft identifica diferentes roles que intervienen en el desarrollo de proyectos, de los cuales los roles principales son:

- Cliente: Es la persona ó personas que solicitan un producto de software y financia el desarrollo ó mantenimiento.
- Usuario: Son las personas que utilizarán el producto generado.
- Grupo directivo: grupo de personas que dirigen la organización ubicados en los niveles de alta dirección y dirección ó gerencia.
- Responsables de proceso: Encargado de realizar las prácticas de un proceso y del cumplimiento de los objetivos.
- Involucrados: Otros roles con habilidades requeridas para la ejecución de actividades ó tareas específicas. Por ejemplo: analista, programador, entre otros.

Los productos de software desarrollados con base en Moprosoft se clasifican de manera general, como: especificación de requerimientos, análisis, diseño, software, prueba, entre otros. Esta clasificación (figura 13) debe ser especializada de acuerdo a las necesidades detectadas, tal es el caso de productos específicos como: manual de usuario, reportes de pruebas, componentes de software, sistemas compuestos de subsistemas, entre otros. En la figura se muestra un ejemplo de clasificación de productos de software.

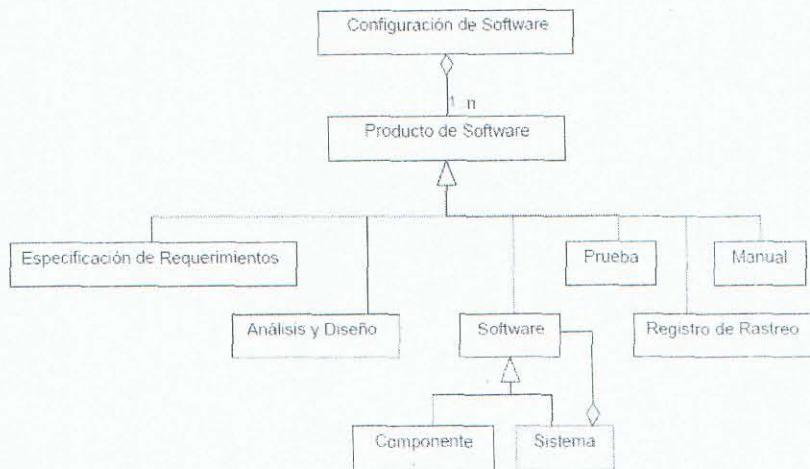


Figura 13. Ejemplo de clasificación de productos de software

Requerimientos para el uso del modelo Moprosoft

- Cada organización debe generar los procesos tomando en cuenta lo siguiente:
 - Definir las metas cuantitativas de acuerdo a las estrategias de la organización
 - Identificar y definir las características de los roles y productos (entradas, salidas e internos).
 - Definir estándares de documentación para cada producto.
 - Definir los recursos de infraestructura de cada proceso.
 - Definir las métricas de evaluación para cada proceso.
 - Para el caso del proceso de desarrollo de software y mantenimiento, definir técnicas ó métodos específicos para las actividades propias de este proceso.
 - Establecer la correspondencia entre los procesos de la organización con el modelo de Moprosoft para identificar coincidencias y discrepancias (con ello, ajustar las discrepancias para lograr la cobertura completa del modelo de Moprosoft).
 - Establecer la estrategia de implantación de los procesos definidos.

2.3.4 RUP

En (Kroll, 2003), se expone la metodología denominada “Rational Unified Process – RUP” como un marco de desarrollo para el proceso de ingeniería de software; es considerado como un compendio de mejores prácticas que involucran la asignación y gestión de actividades y responsabilidades en una organización dedicada al desarrollo de software. Aplicando esta metodología, los equipos de desarrollo de software pueden producir software de alta calidad que cumpla con las necesidades de los usuarios finales mediante un horario y presupuesto predecible.

Las características principales de RUP son:

- Enfoque de desarrollo de software iterativo incremental donde cada iteración puede involucrar una ó más disciplinas de desarrollo (requerimientos, análisis, diseño, implementación, evaluación, etc.); a su vez, cada iteración contiene un conjunto de objetivos y produce una implementación parcial del producto final.
- RUP se centra en el desarrollo de la arquitectura de software en etapas tempranas del desarrollo del proyecto.
- Se conduce mediante casos de uso que definen las características principales del proyecto de desarrollo de software.
- Contiene guías detalladas, ejemplos y plantillas para todas las etapas del ciclo de desarrollo de software.
- Es considerado un proceso de desarrollo de software bien definido (well-defined) y bien estructurado (well-structured).
- Provee una estructura bien definida para el ciclo de vida de un proyecto de desarrollo de software articulando los hitos esenciales y los puntos para la toma de decisiones.

Principios subyacentes del enfoque RUP

El núcleo de RUP está conformado por varios principios que soportan el desarrollo iterativo. Los principios descritos en (Kroll, 2003) son:

- Se enfoca a cubrir el mayor número de riesgos al inicio de proyecto y de forma continua: riesgos de negocio, riesgos técnicos ó riesgos relacionados con un proyecto.

- Asegura la aportación de valor al cliente mediante la documentación de requerimientos de tal forma que sean entendibles por el cliente y trabaja de forma directa con dichos requerimientos en las etapas de diseño, implementación y pruebas para asegurar el cumplimiento de éstos.
- Gestión de cambios en etapas tempranas del desarrollo de software.
- Basado en la configuración de la arquitectura en etapas tempranas del proyecto.
- Construcción del sistema con base en componentes.
- Trabajo colaborativo.

2.3.5 CMMI

El modelo Capability Maturity Model (CMM) ó también denominado CMM-SW fue desarrollado como marco de referencia para la evaluación y mejora de procesos de software en su primera versión en el año 1991 (Paulk, 1993). En (CMMI, 2002) se hace referencia a los elementos de CMM para el desarrollo de procesos eficaces basados en los conceptos desarrollados por: Crosby, Deming, Juran y Humphrey. En el año 2000, CMM-SW fue actualizado hacia el modelo CMMI (Capability Maturity Model Integration).

Los 5 niveles de madurez

CMM organiza la madurez de un proceso de software en 5 niveles (CMMI, 2000), donde se establecen escalas de medida de la capacidad de los procesos y se fijan unos objetivos que ayudan a la organización a priorizar los esfuerzos dedicados a la mejora de estos procesos:

- Nivel 1 ó inicial: los procesos no están definidos aunque algunos de ellos se realicen de forma controlada. El trabajo se realiza de forma impredecible. El éxito depende de las capacidades y esfuerzos dedicados a la mejora de estos procesos.
- Nivel 2 ó repetible: se establecen las actividades básicas de gestión de proyectos, de forma que pueden observarse y reproducirse las prácticas que han tenido éxito en proyectos anteriores.
- Nivel 3 ó Definido: Se documentan y estandarizan tanto los procesos de desarrollo y mantenimiento de software, como la gestión. Todos los proyectos usan una versión de los procesos integrada en la organización.

- Nivel 4 ó Gestionado: se establece un programa amplio y detallado de medidas para los procesos y los productos de software. Se recopilan y analizan los datos de todos los proyectos, formando una base de datos cuantitativa como apoyo a la evaluación y estimación de proyectos.
- Nivel 5 ó en optimización: la organización cuenta con procesos de mejora continua de todos los procesos, empleando los datos de los proyectos y sus resultados, generando nuevas ideas e innovación en el uso de métodos y herramientas.

Cada nivel de madurez está organizado en Áreas Clave de Proceso KPA (Key Process Area) que representan un grupo de prácticas o actividades relacionadas para alcanzar un conjunto de objetivos que permiten mejorar la capacidad ó madurez del proceso. Los niveles de KPA se presentan en la Tabla 2:

Tabla 2. Niveles de KPA.

Nivel	Áreas Clave de Proceso
1. Inicial	No hay áreas clave de proceso definidas.
2. Repetible	Gestión de la configuración del software. Aseguramiento de la calidad de software. Gestión de acuerdos y contratos con proveedores. Planificación, seguimiento y control de procesos. Gestión de requisitos.
3. Definido	Revisión detallada de los procesos. Coordinación dentro de los grupos de trabajo. Ingeniería del producto de software. Gestión integrada del proyecto. Programa de formación. Definición del proceso organizativo. Enfoque hacia los procesos organizativos.
4. Gestionado	Gestión de la calidad de software. Gestión cuantitativa de proyectos.
5. En optimización	Gestión del cambio en los procesos. Gestión de los cambios tecnológicos. Prevención de defectos.

A su vez, los KPA están organizados en Características Comunes CF (Common Features), es decir, atributos que muestran si la implementación e institucionalización de un KPA ha sido efectiva, repetible y perdurable. Las 5 características de los KPA se pueden definir como:

- Compromiso. Conjunto de acciones que la organización debe realizar para asegurar que el proceso sea repetible y duradero. Se relacionan con las políticas de la organización y el liderazgo de la dirección.
- Capacidad. Precondiciones que deben darse en un proyecto ó en la organización para implementar los procesos de software. Afecta de forma directa a los recursos, a la estructura y a la formación.
- Actividades. Roles y procedimientos relacionados con la planeación y el seguimiento del trabajo para implementar los KPA.
- Medidas y análisis. Incluyen ejemplos de las medidas para determinar el estado y la eficacia de las actividades relacionadas.
- Verificación. Describen los pasos para asegurar que las actividades se realizan con base en los procesos establecidos.

CMMI

Surgió como la necesidad de estandarizar procesos derivados de diferentes modelos que presentaban diferentes arquitecturas, enfoques, contenidos y aplicaciones. CMMI contiene un conjunto de productos y métodos de evaluación para los diferentes modelos adaptables que conforman CMMI. CMMI tiene un enfoque disciplinado para mejorar los procesos de la organización, ayudando a establecer los objetivos de mejora y las prioridades, proporcionando guías para implementar procesos de calidad así como un marco de referencia para la realización de las evaluaciones (CMMI, 2002).

CMMI con CMM tiene pocas diferencias, siendo la más relevante que CMMI se puede adaptar a diferentes procesos (Ahem et al., 2001). Cabe destacar que CMMI incluye nuevos procesos, mejores prácticas clave derivadas de la aplicación de CMM, se añadió la conformación de objetivos genéricos por área de proceso. En el nivel 2 se agregó una sección

para medición y análisis que agrupan las prácticas de diferentes procesos contemplados en CMM en esta categoría. En el nivel 3, se desagregó el proceso de ingeniería del producto conformando los procesos de desarrollo de requisitos, soluciones técnicas, integración del producto, verificación y validación; a su vez, se agregó el área denominada análisis de decisiones y resolución. En el nivel 4 se reestructuró y las áreas de gestión cuantitativa de procesos y gestión de la calidad del software se han convertido en gestión cuantitativa del proyecto y rendimiento o realización del proceso organizacional, respectivamente. Para el nivel 5, se fusionaron las áreas de gestión de los cambios tecnológicos y gestión del cambio en los procesos en una única área: innovación organizacional y despliegue. El área de prevención de defectos se reestructuró y renombró como análisis casual y resolución.

El modelo CMMI es muy complejo, retomando las partes principales se pueden enumerar:

- Áreas de proceso: CMMI Identifica 24 áreas de procesos que son relevantes para la capacidad y la mejora del proceso de software (Tabla 3).
- Metas: CMMI se enfoca a la definición de metas específicas asociadas a cada área de proceso que definen el estado deseable de cada tarea, y metas genéricas asociadas con la institucionalización de buenas prácticas.
- Prácticas: en CMMI se consideran descripciones de vías para conseguir las metas.

Tabla 3. Áreas de proceso en CMMI

Categoría	Área de proceso
Gestión del proceso	Definición de procesos organizacionales Centrar la atención en procesos organizacionales Aprendizaje organizacional Rendimiento de los procesos organizacionales Desarrollo e innovación organizacional
Gestión del proyecto	Planificación del proyecto Control y seguimiento del proyecto Gestión de acuerdos con los proveedores Gestión de la integración del proyecto Gestión de riesgos Integración del equipo

	Gestión cuantitativa del proyecto
Ingeniería	Gestión de requerimientos Desarrollo de requerimientos Soluciones técnicas Integración del producto Verificación Validación
Soporte	Gestión de configuraciones Gestión de calidad del proceso y del producto Análisis y mediciones Análisis y toma de decisiones Entorno organizacional para integración Análisis y resolución casual

2.3.6 PERT-CPM

PERT/CPM es una técnica que representa los sistemas como una red de eventos (Izar, 1996) que facilitan la apreciación y comprensión de la estructura y el funcionamiento de sistemas como proyectos, modelos, métodos y esquemas.

PERT involucra técnicas de revisión y evaluación de problemas basándose en un enfoque probabilístico. Por otra parte CPM se enfoca a definir el camino crítico con un enfoque determinístico.

PERT ha sido empleado en proyectos de investigación y desarrollo donde no existen precedentes en cuanto al tiempo de desarrollo de las actividades, por lo que es necesario estimar la duración de dichas actividades para efectuar la programación correspondiente. Se supone que el tiempo de las actividades debe seguir un comportamiento que pueda ser descrito por alguna distribución de probabilidad. El tiempo para cada actividad se debe calcular estimando el tiempo óptimo, el tiempo pesimista y el tiempo probable – sin considerar información referente a proyectos anteriores – mediante la siguiente fórmula:

$$t_c = (t_o + 4t_m + t_p) / 6 \quad (1)$$

Donde:

T_c Tiempo esperado para la actividad

T_o Tiempo optimista de la actividad

T_m Tiempo más probable para la actividad

T_p tiempo pesimista para la actividad

Con estas ponderaciones, se busca encontrar, de forma subjetiva, el tiempo esperado por actividad; sin embargo, cuando los intervalos de valor entre tiempos optimista y pesimista son grandes, disminuye la confianza en el uso del tiempo esperado estimado por la fórmula (1). Con base en ello, se parte de dos suposiciones: los tiempos de las actividades son estadísticamente independientes y el tiempo del proyecto sigue una curva de distribución normal de probabilidad.

Relación entre tiempo y costo para proyectos con PERT/CPM

CPM, el método del camino crítico, trata la relación entre costo y tiempo de los proyectos, siendo complemento a PERT en la estimación de costo y tiempo. CPM considera que la mayoría de las actividades puede reducirse hasta cierto punto en relación a tiempo con la asignación de recursos adicionales: mayor número de personas, empleo de equipos y maquinaria adicionales, pago extra a contratistas, entre otros, lo que repercutiría en el incremento de costo del proyecto.

PERT/CPM sólo puede ser empleada en proyectos cuya relación costo-tiempo es inversamente proporcional, siendo el punto normal de costo mínimo en un tiempo estimado normal y otro punto de urgencia donde el tiempo se reduce pero el costo se orienta hacia el punto de urgencia (figura 14). A su vez, en PERT/CPM, se deben estimar los costos contemplando un costo de actividad a tiempo de urgencia y un segundo valor de costo de la actividad a tiempo normal; estos valores son estimados de manera subjetiva por parte del administrador de proyectos.

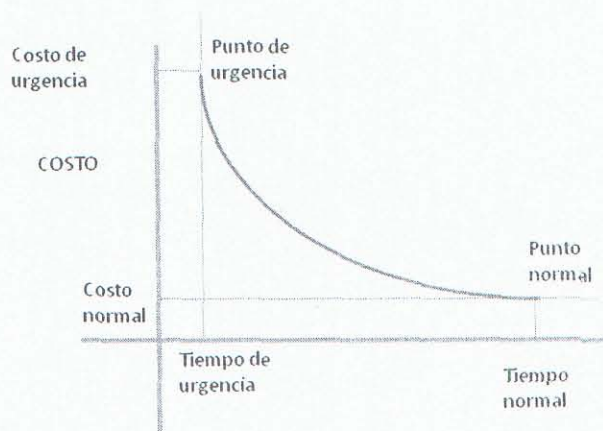


Figura 14. Relación gráfica entre tiempo y costo para proyectos con PERT/CPM

Para la estimación del costo con respecto al tiempo, se emplea la fórmula (2):

$$s_{tc} = (Cu - Cn) / t_{ah} \quad (2)$$

Donde:

Cu = Costo de la actividad a tiempo de urgencia

Cn = Costo de la actividad a tiempo normal

t_{ah} = Tiempo de ahorro dado por la fórmula: $t_{ah} = t_n - t_u$

La fórmula s_{tc} se debe calcular por cada actividad que corresponde al proyecto que se está analizando. Con ello, tanto el tiempo como el costo sólo se estiman.

Las desventajas contempladas en el modelo de PERT/CPM se enlistan a continuación:

- No contempla la actualización de las estimaciones en el momento de realizar modificaciones en la ejecución de actividades, lo que implica la generación de un nuevo modelo PERT/CPM.
- Se tienen varios formatos que se hicieron como complemento al modelo de PERT/CPM, lo que hace confuso la distinción entre el modelo original y el actualizado.
- No permite representar proyectos con cierto grado de complejidad.

- No ofrece un modelo donde contemple condiciones para el análisis de opciones.
- Es considerado un modelo basado en la técnica de prueba y error.
- No permite la visualización de la relación entre las actividades cuanto se tiene un gran número de éstas.

2.4 Project Scheduling Problem (PSP)

La gestión de proyectos es la aplicación de los conocimientos, habilidades, herramientas y técnicas a las actividades de los proyectos para que cumplan o excedan las necesidades y expectativas de un proyecto, tales como: alcance, tiempo, costo y requerimientos de calidad, requerimientos identificados (necesidades) y requerimientos no identificados (expectativas). (Ruiz-Vanoye y otros, 2010).

De acuerdo a (Atkinson, 1999), la gestión de proyectos se considera como la planificación, seguimiento y control de todos los aspectos relacionados con un proyecto y la motivación de todos quienes participan en ella para lograr los objetivos del proyecto en relación al tiempo y costo especificados, la calidad y el rendimiento de dicho proyecto.

En el proceso de gestión de proyectos se lleva a cabo la aplicación de conocimientos, habilidades, herramientas y técnicas para la definición de las actividades de los proyectos para cumplir las necesidades y expectativas de un proyecto, tales como: alcance, tiempo y costo.

Con base en la descripción de Kimms (2001), un proyecto es un esfuerzo temporal que, a través de un conjunto de recursos, se busca satisfacer objetivos específicos en un periodo de tiempo determinado. Los siguientes atributos apoyan en la definición de un proyecto:

- Objetivo único y bien definido.
- Temporalidad. Un proyecto debe tener definido un inicio y fin con base en las características de las actividades a desarrollarse.
- Desarrollo colaborativo progresivo. Un equipo de trabajo, al inicio del proyecto, debe definir los requerimientos y el alcance del mismo. Por consecuencia, la actualización el estatus del proyecto se debe realizar conforme pasa el tiempo. Y por último, La

asignación de actividades se debe llevar a cabo con base en las características del personal involucrado.

- Recursos: los recursos incluyen personal, hardware, software, recursos financieros, insumos, recursos renovables, recursos no renovables, entre otros, que deben emplearse de manera óptima para cumplir con los objetivos del proyecto. Se sugiere el desarrollo de actividades bajo una arquitectura empresarial en procesos, los cuales involucran diferentes áreas de la misma empresa para compartir los recursos destinados al proyecto.
- Definición de los clientes y selección de patrocinadores: Todos los proyectos cuentan con partes interesadas (stakeholders) en el desarrollo del proyecto, donde algunos toman el rol de patrocinio, quien usualmente, provee la dirección y el financiamiento.
- Grado de incertidumbre: Factores externos e internos involucrados en la definición del alcance, tiempo y costos del proyecto, generan un grado de incertidumbre cuya consecuencia se refleja en la variación que se tiene durante el desarrollo del mismo.
- Restricciones: cada proyecto tiene diferentes restricciones debido a su alcance (características del producto o servicio), costo (estimación y seguimiento de costos) y tiempo (asignación del tiempo por actividad, tiempo total y el seguimiento de los tiempos acumulados considerando las variaciones en relación a la planeación), principalmente (nombrado en administración de proyectos como la triple restricción).

La administración de proyectos, de acuerdo a Demeulemesteer (2002), involucra principalmente la planeación, la programación y el control de las actividades de un proyecto para lograr los objetivos relacionados con un desempeño óptimo, costo y tiempo en función de un alcance determinado empleando fuentes de forma efectiva y eficiente:

- La fase de planeación involucra el listado de actividades que deben ser desarrolladas para completar el proyecto y la definición de los requerimientos generales de diferentes fuentes estimando duración y costo.
- La programación es la etapa que consiste en trazar las actividades del proyecto en cierto orden en relación al tiempo y al desempeño esperado; en esta etapa, los recursos son calculados por etapa a lo largo del tiempo estimado para cada actividad.

- La etapa de control se focaliza en la diferencia entre la programación y el desempeño que se tiene de las actividades en tiempo real, una vez que inicia el proyecto.

2.4.1 Descripción

Kelley (1963) propone el concepto de Critical Path Method (CPM) – actualmente Project Scheduling Problem (PSP) – como una técnica de modelado de proyectos que incluye el listado de actividades requeridas para desarrollar un proyecto, la duración en tiempo de cada actividad (que en su conjunto es el tiempo total del proyecto) y la dependencia entre actividades; utilizando estos valores, CPM estima las diferentes secuencias de actividades ($\text{inicio}_{a1} - \text{fin}_{a1} - \text{inicio}_{a2}$) calculando el camino más largo con el tiempo más corto para la terminación del proyecto:

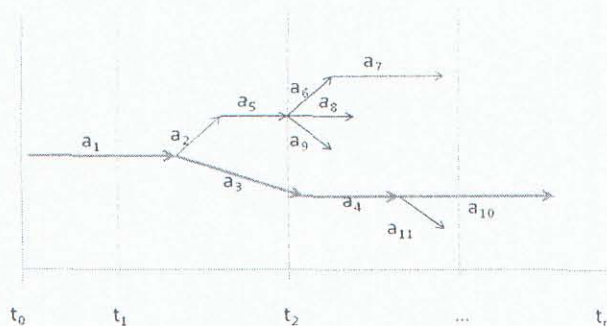


Figura 15. Secuencia de actividades en un proyecto

En la figura 15, se muestran las diferentes secuencias de actividades y en color rojo se presenta la ruta crítica en relación a la dependencia entre actividades.

Con base en la definición de PSP que presenta Ruiz-Vanoye y otros (2010), las siglas PSP son consideradas como un nombre genérico para toda clase de problemas en los que es necesario programar, de manera óptima el tiempo, costo y los recursos de los proyectos.

2.4.2 Clasificación de modelos basados en PSP

Ruiz-Vanoye y otros (2010), clasifica a PSP (Figura 16) de acuerdo al objetivo de optimización mediante el cual fueron creados, por ejemplo PSP de manera general busca optimizar: tiempo, costo del proyecto y la utilización de los recursos.

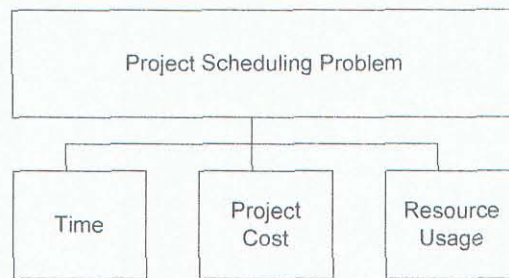


Figura 16. Taxonomía de PSP.

Los parámetros definidos en la figura 16 se clasifican de la siguiente manera:

1. Tiempo: La duración total del proyecto se puede determinar a partir de la sumatoria de los tiempos de cada actividad y puede ser determinado por los parámetros de tiempo, donde el tiempo de inicio hace referencia al evento que inicializa la actividad, y el tiempo de término que especifica el tiempo de vencimiento propuesto. El tiempo máximo del proyecto se estima a partir de la ruta crítica;
2. Costo del proyecto: La determinación del costo se basa principalmente en la determinación de recursos humanos, materiales y financieros que estarán involucrados en el desarrollo del proyecto.
3. Recursos: Los recursos han sido catalogados por diferentes autores con base en las características de éstos:
 - a. Slowinski (1980) y Weglarz (1980) definen los recursos como renovables (equipo, mano de obra y equipamiento), no renovables (representados principalmente por dinero) y recursos limitados (dinero, materias primas, cartuchos, entre otros).

- b. Blazewicz et al (1983 y 1986) clasifica los recursos como priorizables y no priorizables con base en su uso en el desarrollo de actividades del proyecto.
- c. Schewchuk and Chang (1995) y Böttcher et al. (1999) definen a los recursos reutilizables como aquellos recursos que tienen un deterioro por su uso en el desarrollo de actividades del proyecto y, por consecuencia, tienen periodos de indisponibilidad, afectando el desarrollo del proyecto.
- d. Los recursos dedicados (Bianco et al., 1998), pueden ser asignados a una actividad en un periodo de tiempo determinado.
- e. Boer (1998), define a los recursos espaciales como aquellos empleados en el desarrollo de un grupo de actividades hasta la conclusión de dicho grupo de actividades; ejemplo: oficinas, cubículos, contenedores, entre otros.
- f. Los recursos complementarios son definidos por Artigues and Roubellat (2001) como los recursos empleados para la creación de otros recursos y que no se emplean como soporte para las actividades. Por ejemplo, un grupo de trabajadores que desarrollan ó ensamblan equipo especializado ó computadoras.
- g. Los recursos acumulativos son definidos por Neumann et al. (2002) como la generalización de los recursos no renovables; es decir, son aquellos recursos empleados en un periodo de tiempo hasta el límite de su uso, por ejemplo, el dinero, las instalaciones de almacenamiento que tienen un límite de almacenamiento, materias primas, entre otros.
- h. Nerón (2002), define a los recursos multi-habilidades como los recursos que se pueden emplear en diferentes actividades. Cada recurso cuenta con más habilidad y, por ende, se pueden emplear en diferentes actividades, pero una a la vez. Tal es el caso de los trabajadores que pueden desarrollar diferentes actividades, los equipos de cómputo configurables para diferentes actividades, entre otros.
- i. Los recursos sincronizados descritos por Schwindt and Trautmann (2003), hacen referencia a los recursos que se deben sincronizar para el inicio de un conjunto de actividades simultáneas.

- j. Los recursos asignables de acuerdo a Schwindt and Trautmann (2003), como los recursos que requieren ser configurados antes de iniciar la actividad a la que fueron destinados.
- k. Recursos adyacentes con base en la definición de Duin and Van Der Sluis, (2006) y Paulus and Hurink (2006), tienen una localización física y contribuyen de forma indirecta en el desarrollo de actividades; por ejemplo, los mostradores en un aeropuerto, escritorios para el desarrollo de software, entre otros.
- l. Los recursos de transición (Neumann et al., 2006), son aquellos que requieren de una configuración inicial y que va cambiando conforme es empleado en diferentes actividades.

Las actividades se definen como un conjunto de operaciones ó tareas propias de una persona u organización. Las actividades se caracterizan por: requerimientos (demanda de recursos, solicitud de recursos), personal involucrado, modelo de procesamiento de actividades y actividades predecesoras restrictivas. Los parámetros relativos a actividades de acuerdo a (Weglarz et al., 2010) son:

- Requerimientos necesarios para ejecutar la actividad.
- Parámetros de tiempo: se puede considerar el tiempo de desarrollo de la actividad, la fecha de entrega, ó algún parámetro relativo al tiempo que se contemple para el desarrollo de las actividades del proyecto.
- Parámetros de peso: se relacionan con el peso (costo ó recompensa) que se tiene por ejecutar una actividad.
- Tiempo de configuración. Tiempo relacionado con la configuración de los requerimientos para poder ejecutar una actividad.
- Priorización de las actividades.

2.4.3 Variantes de PSP

Las principales variantes de PSP se pueden clasificar en:

- Resource Constraint Project Scheduling Problem (RCPSP): consiste en organizar un conjunto de actividades, en un tiempo determinado con recursos limitados (Pritsker et al., 1969) catalogados como renovables y no renovables asignados en el transcurso del proyecto. RCPSP, a su vez, tiene variantes clasificadas como: Single-Mode Resource-Constrained Project Scheduling Problem or SMRCPSP (Demeulemeester and Herroelen, 1992), high-school timetabling (Brucker and Knust, 2001; Schaerf, 1999b), University Course Timetabling (Brucker and Knust, 2001; Schaerf, 1999a), audit-scheduling problem (Brucker and Schumacher, 1999), entre otras variantes que son descritas en Brucker et al. (1999a) y Herroelen et al. (1998). El modelo matemático se presenta a continuación:

$$\text{Min } S_{n+1} \quad (3)$$

$$\text{subject to: } S_j - S_i \geq d_j, \quad (i, j) \in A \quad (4)$$

$$\sum_{P(t)} r_{ik} \leq R_k, \quad t = 1, \dots, T; \quad k = 1, \dots, K \quad (5)$$

$$S_i \geq 0 \quad \forall i \quad (6)$$

Donde:

S tiempo de inicio;

n número de actividades en el proyecto;

n+1 actividad subsecuente;

d_j es la duración de la actividad j;

i, j actividades que pertenecen al conjunto A

k recursos;

K número de tipos de recursos;

t periodo de tiempo;

r unidades de recursos;

R recursos requeridos por el conjunto de actividades;

T el tiempo total del proyecto.

- Multi-Mode Resource-Constrained Project Scheduling Problem or MRCPSPP (Talbot, 1982). MRCPSPP involucra la selección de la combinación de tiempo/recursos en una actividad para minimizar el desarrollo del proyecto. MRCPSPP es considerado un

problema NP completo en Kolisch and Drexl, (1997). Algunas variables de MRCPSP son: Multi-mode resource-constrained project scheduling problem with discounted cash flows or MRCPSPCF (Brucker et al., 1999; Ulusoy, 2001), multi-skill project scheduling problem or MSPSP (Bellenguez and Néron, 2005; Bellenguez-Morineau, 2008), entre otros. El modelo matemático (Talbot, 1982) se presenta a continuación:

$$\text{Minimize } \sum_{t=EF_{n+1}}^{LF_{n+1}} tx_{n+1,m,t} \quad (7)$$

$$\text{subject to } \sum_{m=1}^{|M_j|} \sum_{t=EF_j}^{LF_j} x_{jmt} = 1 \text{ for } j = 0, \dots, n+1 \quad (8)$$

$$\sum_{m=1}^{|M_j|} \sum_{t=EF_j}^{LF_j} tx_{jmt} \leq \sum_{m=1}^{|M_j|} \sum_{t=EF_j}^{LF_j} tx_{jmt} - d_{jm} \text{ for all } (A_i, A_j) \in P \quad (9)$$

$$\sum_{j=1}^n \sum_{m=1}^{|M_j|} \sum_{q=\max\{t+d_m-1, LF_j\}}^{\min\{t+d_m-1, LF_j\}} r_{jmk}^p x_{jmq} \leq R_k^p \text{ for } k = 1, \dots, R; t = 1, \dots, H \quad (10)$$

$$\sum_{j=1}^n \sum_{m=1}^{|M_j|} \sum_{t=EF_j}^{LF_j} r_{jml}^v \leq R_k^v \text{ for } l = 1, \dots, N \quad (11)$$

$$x_{jmt} \in \{0,1\} \text{ for } i = 0, \dots, n+1; m \in M_j; t = EF_j, \dots, LF_j \quad (12)$$

Donde:

R_k^p (R_k^v) número de unidades disponibles de recursos (renovables o no renovables) del proyecto;

r_{jmk}^p número de recursos renovables ($k=1\dots N$) y no renovables ($l=1\dots n$) requeridos para la actividad A_j ;

d_{jm} es la duración de la actividad A_j ;

- Construction Project Scheduling Problem, construction PSP or CPSP (Adeli and Karim, 1997), consiste en seleccionar recursos clasificados en fuerza de trabajo y maquinaria, y las actividades en proyectos de construcción, considerando condiciones particulares tales como: métodos organizacionales y tecnología, disponibilidad de recursos, calendarización de actividades, entre otros) donde el método de evaluación se enfoca en 3 conceptos: duración, costo y calidad. El modelo matemático se presenta a continuación:

$$\text{Minimize } f_{N+1} \quad (13)$$

$$f_j - d_j \geq f_i \text{ and } f_i = 0, \forall (i, j) \in H \quad (14)$$

$$Ch\{d'_j \leq d_j \leq d_j^*\} \geq \beta_j, j \in UD \quad (15)$$

$$\sum_{i \in S_t} r_{ikt} \leq R_{kt}, k \in DR \quad (16)$$

$$Pos\left\{\sum_{i \in S_t} r_{ikt} \leq a_{kt}\right\} \geq \alpha_{kt}, k \in FR \quad (17)$$

$$Pr\left\{\sum_{i \in S_t} r_{ikt} \leq b_{kt}(\xi)\right\} \geq p_{kt}, k \in SR, \text{ for } t = 1, 2, \dots, f_N \quad (18)$$

Donde:

f tiempo final estimado del proyecto;

$N+1$ final estimado de una actividad;

N número de actividades del proyecto;

F_j tiempo para completar la actividad j ;

H es un par de actividades con una relación fin-inicio;

β_j es la probabilidad ó posibilidad estimada;

i actividad;

k recursos de una actividad;

K número de tipos de recursos;

t periodo de tiempo;

r unidades de recursos;

r_{ikt} cantidad de recursos tipo k que son requeridos por la actividad i en el tiempo t ;

R recursos requeridos par aun conjunto de actividades;

R_{kt} Total de recursos tipo k que son requeridos por la actividad i en el tiempo t ;

a_{kt} recursos totals disponibles del tipo k en el tiempo t ;

$b_{kt}(\xi)$ es la disponibilidad total de recursos aleatorios tipo k en el tiempo t ;

ch , Pos posibilidad de eventos difusos;

Pr probabilidad de eventos aleatorios;

SR conjunto de recursos requeridos por la administración del proyecto;

FR conjunto de recursos disponibles a disposición del administrador de proyectos como un número difuso relacionado con el tiempo;

DR conjunto de recursos disponibles para el administrador del proyecto planeados en el tiempo t ;

UD conjunto de actividades con una duración aleatoria UD_1 ó difusa UD_2 tal que $UD = UD_1 + UD_2$.

- Project Scheduling Problem for Software Development, software PSP or PSPSD (Huang, 2009). PSPSD consiste en la asignación de recursos humanos a varias actividades en el desarrollo de software de acuerdo a sus habilidades, mientras se mantiene el esfuerzo, los gastos y la calendarización al mínimo (Gonsalves and Itoh, 2010). Las variants de PSPSD son: Fuzzy project scheduling system for software development (Hapke et al., 1994), Time depend software Project Scheduling Problem (Möhring, 2002), A project scheduling problem with labour constraints and time-dependent activities requirements (Drezet, 2007), Project Scheduling Problem for Software Development with Random Fuzzy Activity Duration Times (Huang, 2009), y Fuzzy resource-constrained project scheduling problem for software development (Wang and Huang, 2010). El modelo matemático se presenta a continuación:

$$\text{minimize } f(x) = f_1 + f_2 \quad (19)$$

$$f_1 = \sum_{j=1}^m Pc_j \quad (20)$$

$$f_2 = \sum_{j=1}^m (Q_{Tj} + P_{Tj}) \quad (21)$$

$$F_{T(j-1)} < S_{Tj} \quad \forall j \quad (22)$$

$$P_j(t) \Rightarrow \tau_i(t) \neq P_j(t) \Rightarrow \tau_k(t) \quad \forall i, j, k \quad (23)$$

Donde:

- Pc costo por unidad de tiempo del personal j^{th} ;
- m número de personas asignadas al proyecto;
- f_1 costo total de desarrollo;
- P_{Tj} tiempo de proceso de la actividad j^{th} ;
- Q_{Tj} tiempo restante de la actividad j^{th} de la disponibilidad de recursos;
- f_2 duración actual del proyecto;
- S_{Tj} tiempo de inicio de la actividad j^{th} ;
- F_{Tj} Tiempo final de la actividad j^{th} .

2.4.4 Algoritmos empleados para resolver PSP

En la tabla 4 se presenta un concentrado de algoritmos (Ruiz-Vanoye y otros, 2010) que resuelven las variantes de PSP.

Tabla 4. Algoritmos que resuelven algunas variantes de PSP

Algoritmos / métodos	RCPSP y variantes	MRCPSPP y variantes	Construction PSP y variantes	Software PSP y variantes
Tabu Search	1	5	-	-
Simulated Annealing	2	6	9	-
Genetic Algorithm	3	7	10	11
Others	4	8	-	12

CAPÍTULO 3. MODELO DE INDICADORES PARA EVALUAR LA COMPLEJIDAD DE PROYECTOS DE SOFTWARE

3.1 Definiciones

Programación lineal

Un modelo de programación lineal considera que las variables de decisión tienen un comportamiento lineal, tanto en la función objetivo (Hernández, 2006) como en las restricciones del problema. La función objetivo se puede orientar a minimizar ó maximizar la combinación de las variables; una función objetivo se puede expresar como en la tabla 5:

Tabla 5. Función objetivo.

Función objetivo	Minimize $c^T x$
Restricciones:	Subject to $Ax = b$
Variable de decisión X en un rango de $l \dots u$	$l \leq x \leq u$

CPLEX

En AIMMS web (2011), CPLEX es un método para la solución de problemas de programación lineal, de programación entera mixta y programación cuadrática. Está diseñado para resolver problemas extensos con múltiples combinaciones cuya solución es considerada exponencial. Concentra diferentes algoritmos para resolver problemas de programación lineal: algoritmo simplex primal y dual, algoritmo de barrera, el algoritmo de red, matrices dispersas, entre otros.

3.2 Componentes iniciales del problema

El objetivo general de la presente tesis se orienta a desarrollar un modelo de indicadores para el análisis de la complejidad en etapas tempranas de la planeación de proyectos de desarrollo de software.

Cada proyecto conlleva la administración de recursos empleados en la asignación de actividades y de personal con un conjunto de habilidades para el desarrollo de dichas actividades, donde el costo total de un proyecto se genera a partir de la combinación óptima de tiempo, costo y alcance a partir de los recursos destinados al desarrollo de cada proyecto.

3.3 Múltiple demanda de escenarios

Cada proyecto tiene una combinación exponencial de los recursos, donde cada escenario involucra recursos humanos, actividades, tiempo de desarrollo del proyecto, distribución de tiempos por actividad/persona, asignación del personal con base en sus habilidades, asignación del personal con base en el costo por actividad, entre otros factores que se involucran de forma directa e indirecta con la planeación de un proyecto de desarrollo de software.

Para el caso particular de esta tesis, se desarrolla un escenario para obtener el grado de complejidad del proyecto con base en determinar el dominio de las actividades por parte del personal (habilidades), tiempo estimado de cada trabajador en el desarrollo de una actividad por clase (tiempo por actividad) y el número de errores promedio que tiene cada trabajador en el desarrollo de las actividades por clase (errores por trabajador).

3.4 Descripción del modelo

AIMSS (Ortega y Sánchez, 2007 y Heerink, 2010) es una herramienta para ambiente construir aplicaciones de investigación de operaciones con optimización combinatoria basándose en la programación lineal, que es una técnica matemática utilizada en la solución de diversos problemas relacionados con la gestión, desde la programación, selección de medios, planificación financiera para el presupuesto de capital, el transporte y muchos otros, con la característica especial de programación lineal que esperar siempre a maximizar o minimizar cierta cantidad proveniente de una función objetivo lineal.

Para el modelado de la presente tesis, se tienen los siguientes conceptos:

- S – Índices que se relacionan con el conjunto de datos que serán empleados.
- P – Parámetros que se relacionan con los índices.

- V – Variables que serán empleadas en las diferentes operaciones.
- C – Restricciones que se tienen en relación a los datos del modelo.
- Mp – Programa matemático que ejecuta las funciones objetivo.

Los índices, en la presente tesis, son un listado de datos asociados que permitirán hacer los cálculos relacionados con la función objetivo. En la tabla 6 se muestran los índices (S) relacionados con las dimensiones del problema con base en los indicadores seleccionados para el modelo propuesto:

Tabla 6. Tabla de índices de la dimensión del proyecto.

Índice	Descripción
T	Trabajadores que se involucran en el desarrollo del proyecto
A	Actividades que serán desarrolladas en el proyecto
E	Número de errores estimados que tiene cada trabajador en el desarrollo de las actividades por clase del proyecto actual
C	Costo promedio por actividad de las diferentes clases
P	Proyecto que será considerado en el escenario actual
R	Recursos que serán empleados en el escenario actual
Req	Número de requerimientos determinados que serán cubiertos por las actividades del proyecto
Clase	Clase – conjunto de actividades relacionadas entre sí.
S	Grado de habilidad que tiene un trabajador en el desarrollo de las actividades por clase.

3.4.1 Criterios para determinar el dominio de datos en indicadores

Los indicadores están definidos por un rango de datos cuyo límite depende, principalmente, de las características de la empresa. A continuación se exponen los dominios de los indicadores:

- Nivel de habilidad: Este indicador está determinado en una escala de 1 a 10, donde cada trabajador puede tener desde un nivel básico 1 hasta un nivel experto por cada clase de actividades. Este dominio estará determinado por la capacitación que ha

tenido el trabajador en relación a las clases de actividades (requerimientos, análisis, diseño, etc.) y del tiempo que ha trabajado desarrollando actividades de dichas clases:

- Nivel 1: recién egresado sin experiencia.
- Nivel 2: recién egresado con el desarrollo de actividades escolares relativas a cada clase.
- Nivel 3: recién egresado que ha laborado en actividades de cada clase.
- Nivel 4: Egresado del área con un año de experiencia en el desarrollo de actividades por clase.
- Nivel 5: Trabajador con 2 a 3 años de experiencia en el desarrollo de actividades por clase.
- Nivel 6: Trabajador con 5 años de experiencia en el desarrollo de actividades por clase.
- Nivel 7: Trabajador con un año de experiencia en el desarrollo de proyectos de desarrollo de software y con más de 3 años de experiencia en el desarrollo de actividades por clase.
- Nivel 8: Trabajador con 2 a 3 años de experiencia en el desarrollo de proyectos de desarrollo de software y con más de 5 años de experiencia en el desarrollo de actividades por clase.
- Nivel 9: Trabajador con más de 5 años de experiencia en el desarrollo de proyectos de desarrollo de software y más de 5 años de experiencia en el desarrollo de actividades por clase.
- Nivel 10: Trabajador que haya participado en la administración y desarrollo de proyectos de software ERP, con más de 5 años de experiencia en el desarrollo de proyectos de software y más de 5 años de experiencia en el desarrollo de actividades por clase.
- Nivel de errores: Este indicador estará valorado por parte de la empresa en relación a una ponderación sobre los errores que han sido detectados en proyectos anteriores donde ha participado el trabajador. En este caso, el rango de datos varía de 1 (ponderación que se emplea cuando el trabajador comete pocos errores) a 10 (ponderación máxima si un trabajador comete muchos errores).

- Total de Actividades del trabajador: es la asignación máxima de actividades que le podrá dedicar cada trabajador al proyecto actual.
- Tiempo total del trabajador: Es el tiempo máximo que le podrá dedicar el trabajador al proyecto actual. Se ha considerado como tiempo base la hora.
- Costo Unitario por actividad: El costo unitario por actividad se estima con el costo por hora que cobra un trabajador al realizar una actividad por clase.
- Tiempo Unitario por Actividad: Es el número aproximado en horas que se tarda en realizar cada trabajador destinado al proyecto actual una actividad por clase.

Los indicadores (tabla 7) son empleados en la presente tesis para asignar datos a los índices y vincularlos con las variables.

Tabla 7. Tabla de indicadores.

Parámetro	Descripción
NivelHabilidadesTrabajador(t,a)	Nivel que tiene cada trabajador para realizar las diferentes clases de actividades
TotalActividadesTrabajador(t)	El número de actividades que cada trabajador le dedicará al desarrollo del proyecto en una unidad de tiempo definida por el administrador del proyecto
TotalTiempoTrabajador(t)	El tiempo que cada trabajador le dedicará al desarrollo del proyecto en una unidad de tiempo definida por el administrador del proyecto
TotalActividadesClase(a)	Actividades por clase que serán desarrolladas en el proyecto durante la unidad de tiempo definida por el administrador del proyecto
CostoUnitarioActividad(t,a)	Costo unitario que tiene cada actividad por clase con base en el nivel de habilidad que tiene cada trabajador. Es decir, es la relación costo-desarrollo de cada actividad-trabajador.
TiempoUnitarioActividad(t,a)	Tiempo estimado que se tarda cada trabajador en desarrollar una actividad por clase.

ErroresTrabajador(t,a)	Número de errores promedio que tiene cada trabajador al desarrollar cada clase de actividad.

3.4.2 Descripción de variables del proyecto

Las variables definidas en el presente proyecto son utilizadas en el software AIMMS:

- Desarrollo de Actividades: Esta variable se basa en una matriz, donde en el eje de las X se emplean las clases de actividades, en el eje de las Y se emplean los trabajadores, y la intersección (X,Y) determina el número de actividades que desarrollará cada trabajador por clase de actividades.
- Costo Total de Desarrollo: En esta variable se almacena el costo total que tendrá el proyecto con base en la función objetivo y en la interacción entre los diferentes indicadores con base en el método CPLEX.
- Tiempo total de Desarrollo: En esta variable se almacena el tiempo total que tendrá el proyecto con base en la función objetivo.
- Errores totales del proyecto: Es una variable auxiliar donde, por cada iteración, se almacena la ponderación total de errores de acuerdo a los trabajadores elegidos por medio del método CPLEX.
- Asignar habilidades: Variable auxiliar donde se almacena la suma del nivel de habilidad que tienen los trabajadores seleccionados por el método CPLEX.
- Restricción de trabajadores: Variable auxiliar donde se compara el número de actividades que desarrollará el trabajador en el proyecto contra el número máximo de actividades que le podrá dedicar el trabajador al proyecto.
- Restricción de actividades: Variable auxiliar almacena la suma del número de actividades asignadas por trabajador que deberá ser exactamente igual al número requerido de actividades por clase.

En las variables de decisión (tabla 8) se almacenan los resultados de las funciones matemáticas empleadas en el modelo propuesto.

Tabla 8. Variables de decisión.

Variable	Descripción
DesarrolloActividades(t,a)	Asignación de actividades totales del proyecto a los diferentes trabajadores
CostoTotalDesarrollo	Estimación del costo total del proyecto con base en el costo por actividad y su asignación por trabajador
DesarrolloTiempo(t,a)	Estimación del tiempo unitario por actividad con base en el tiempo promedio de desarrollo por clase de actividad que emplea cada trabajador
TiempoTotalDesarrollo	Estimación del tiempo total del proyecto considerando el parámetro TiempoUnitarioActividad y la variable DesarrolloActividades(t,a).
ErroresTotalesDesarrollo	Variable que estima los errores totales del proyecto con base en el parámetro ErroresTrabajador(t,a) por las actividades asignadas a cada trabajador en la variable DesarrolloActividades(t,a).

Las restricciones contempladas en este modelo (tabla 9), se relacionan con los aspectos que pueden causar conflicto en la asignación de actividades y en el tiempo que un trabajador le puede dedicar al desarrollo de las diferentes actividades.

Tabla 9. Restricciones.

Restricciones	Descripción
RestriccionTrabajadores(t)	La asignación del número de actividades (DesarrolloActividades(t,a)) debe ser menor al número de actividades que el trabajador desarrollará en el escenario del proyecto actual

	(TotalActividadesTrabajador(t))
RestricciónActividades(a)	El número total de actividades del proyecto (DesarrolloActividades(t,a)) deberá ser menor ó igual a la capacidad total de la organización (TotalActividadesClase(a)).

3.4.3 Descripción de las funciones matemáticas

Las funciones matemáticas (tabla 10) del modelo son empleadas para definir las funciones objetivos relacionadas con maximizar ó minimizar los recursos.

Tabla 10. Funciones matemáticas

Función matemática	Descripción
PlanMenorCosto	Minimizar el costo del proyecto
PlanMenorTiempo	Minimizar el tiempo del proyecto
PlanMenorErrores	Minimizar el número de errores en el desarrollo del proyecto
AsignarHabilidades(t,a)	Maximizar el nivel de habilidad del proyecto

Las funciones matemáticas son empleadas para asignar los resultados a las diferentes variables:

- Minimizar costo: Mediante el método CPLEX, se busca reducir el costo total del proyecto con base en la asignación de actividades de acuerdo al costo unitario por cada actividad y por trabajador.
- Minimizar tiempo: Con la metodología propuesta, se busca reducir el tiempo total del proyecto con base en la asignación de actividades de acuerdo al tiempo unitario por actividad y por trabajador.

- Minimizar errores: Con la combinación de trabajadores, actividades y errores, se busca obtener la ponderación mínima considerando las ponderaciones de errores por trabajador.
- Maximizar habilidades: Con esta función matemática, se busca asignar a los trabajadores para el desarrollo de actividades buscando que con la selección de trabajadores se maximice el nivel de habilidad.

3.5 Función objetivo

El objetivo principal es minimizar el tiempo y costo total del proyecto con base en la ponderación de los costos, tiempo y alcance del escenario propuesto; la fórmula (23) presenta la sumatoria de los costos y del tiempo del escenario:

$$\text{Minimizar } z = \sum_{s1} \sum_{s2} \sum_s P_{s1} r_{s2} t_s \quad (23)$$

Donde:

Tt = Tiempo total del proyecto

t_{ai} = Tiempo de la actividad i

A = Actividades que pertenecen al proyecto

p_i = proyecto i

P = Proyectos que se desarrollan actualmente

a_i = Actividad i

t_j = Tiempo que se tarda en desarrollar el empleado j la actividad i

t_a = Tiempo total de la actividad i

La función objetivo propuesta cumple con los siguientes principios:

- Minimizar el tiempo de desarrollo de un proyecto con base en el tiempo que se estima que un trabajador puede desarrollar diferentes actividades
- Minimizar el costo del proyecto con base en las habilidades del personal para el desarrollo de las actividades.

- Minimizar el tiempo de desarrollo con base en la asignación de actividades al personal de acuerdo a la relación tiempo de desarrollo – número de errores.

3.6 Formulación del modelo matemático de optimización propuesto

El modelo matemático desarrollado para el presente proyecto, consiste de una series de pasos para determinar la complejidad del software con base en las características iniciales de los recursos disponibles y de aquellos recursos destinados al proyecto.

1. Determinar la cantidad de requerimientos de software a desarrollar.

$$\forall p \in P \exists \sum_{req=1}^n req \quad (24)$$

$$\forall a_i \in A \exists clase_j \exists \sum_{req=1}^{N_i} req \quad (25)$$

Donde:

P= Conjunto de proyectos

p= Proyecto actual

req = Conjunto de requerimientos del proyecto

A = Conjunto de actividades del proyecto

a_i = Actividad i

2. Determinar el número de actividades que serán desarrolladas en el proyecto que deberán cumplir con los requerimientos establecidos en el paso anterior.

- listado de clases de actividades
- listado general de actividades
- asignación de actividades del proyecto en cada clase

$$\forall a_i \in A \exists clase_j \in Clase \quad (26)$$

$$\forall p_i \in P \exists A_j = \{a_1, a_2 \dots a_n\} \mid a \in Clase_k \quad (27)$$

Donde:

A= Conjunto de actividades

a_i= Actividad i

Clase = Conjunto de clases del proyecto donde Clase={Análisis, Diseño, Programación, Pruebas, Implementación, Evaluación}

Clase_j=Clase j

P = Proyecto actual

3. Asignación de costos promedio por actividad/trabajador con base en las clases determinadas y en el salario de cada trabajador:

$$\forall a_i \in A \exists ct_j \mid \sum_{k=1}^n ct_k = C_{TP} \quad (28)$$

Donde:

A= Conjunto de actividades

a_i= Actividad i

C_{TP}=Costo total del proyecto actual

ct_k= Costo por actividad k

ct_j= costo por clase

4. Determinar el nivel de habilidad que tiene cada trabajador en el desarrollo de actividades de cada clase.

$$\exists s_e \in S \mid s_e \rightarrow Clase_j \wedge 1 \leq s_e \leq 10 \quad (29)$$

Donde:

s_e= Nivel de habilidad del Trabajador e

S = Conjunto de habilidades requeridas por la empresa

Clase_j = Conjunto de clases (Análisis, diseño, programación, ...)

5. Determinar el tiempo promedio que un trabajador emplea en el desarrollo de una actividad de cada clase en una unidad de tiempo determinada por el administrador de proyectos.

$$\forall a_i \in A \exists t_j \in e_j \mid t_j \leq t_a \quad (30)$$

Donde:

A= Conjunto de actividades

a_i= Actividad i

t_j= Tiempo que se tarda un trabajador j en realizar la actividad

e_j= Empleado j

t_a= tiempo total que le dedicará el trabajador al proyecto actual

6. Determinar el número de errores promedio que un trabajador tiene en cada clase en una unidad de tiempo determinada por el administrador de proyectos.

$$\forall e \in E \exists \sum_{j=1}^n errores_promedio_{clase_j} \wedge clase_j \in Clase \quad (31)$$

Donde:

E= Trabajadores que estarán en el proyecto actual

e= trabajador

errores_promedio_{clasej}= ponderación de errores cometidos en la clase j por el trabajador en proyectos

anteriores

Clase = Conjuntó de clases

clase_j = Clase j

7. Determinar el tiempo total que le dedicará cada trabajador al proyecto a analizar.

$$\forall e_i \in E \exists t_j \in T_c \mid t_j \leq T_c \quad (32)$$

Donde:

T_e= Tiempo total del empleado

t_j = tiempo j que debe ser menor o igual al tiempo total del empleado

E = Conjunto de trabajadores que participarán en el proyecto

e_i = Trabajador i

8. Determinar el número máximo de actividades que desarrollará el trabajador en el proyecto a analizar.

$$\forall e_i \in E \exists \sum_{k=1}^n a \in A \quad (33)$$

Donde:

A = Conjunto de actividades

a_i = Actividad i

E = Conjunto de trabajadores que participarán en el proyecto

e_i = Trabajador i

9. Asignación de las actividades a los empleados involucrados en el proyecto con base en sus habilidades y menor número de errores.

$$e \in E \exists S_i = \{s_1, s_2, \dots, s_n\} \mid s_n \in Sp \quad (34)$$

$$\exists S_i = \{s_1, s_2, \dots, s_n\} \mid (S_i \in S_p) \text{ and } (S_i = a_j) \quad (35)$$

$$\forall s_i \in S_p \exists \text{ clase}_j \in \text{Clases} \quad (36)$$

$$\forall (a_i \in A_p) \text{ and } (a_i \in \text{clase}_j) \exists S_p = \{s_1, s_2, \dots, s_n\} \in \text{clase}_j \mid (e_i \in E_p) \text{ donde } (S_p = S_e) \quad (37)$$

$$\forall (a_i \in A_p) \text{ and } (a_i \in \text{clase}_j) \exists e_i \in E \wedge \sum_{i=1}^{T_i e} \text{errores} \leq e_j \in E \wedge \sum_{j=1}^{T_j e} \text{errores} \quad (38)$$

Donde:

A = Conjunto de actividades

a_j = Actividad j

Clase = Conjunto de clases del proyecto donde Clase={Análisis, Diseño, Programación, Pruebas,

Implementación, Evaluación}

Clase_j=Clase j

S_i = Conjunto de habilidades del trabajador i

S_p= Conjunto de habilidades requeridas para el desarrollo del proyecto

s₁, s₂, s_n= habilidades

E = Conjunto de trabajadores del proyecto

e_i = Trabajador i

errores = ponderación de errores del trabajador i

e_j= ponderación mínima de error que se debe contemplar para el presente proyecto

10. Determinar el costo total del proyecto

$$\text{Min} \sum_{a=1}^n \sum_{e=1}^m c_{(e,a)} * Cplex(e,a) \quad (39)$$

Donde:

c_(e,a)=Reducir el costo del proyecto actual

11. Determinar el tiempo total para el desarrollo del proyecto

$$\text{Min} \sum_{a=1}^n \sum_{e=1}^m t_{(e,a)} * Cplex_t(e,a) \quad (40)$$

Donde:

t_(e,a)= Reducir el tiempo del proyecto actual

12. Determinar la asignación de actividades a los empleados con base en el número de errores promedio que tiene cada trabajador en el desarrollo de actividades por clase.

$$\text{Min} \sum_{clase=1}^n errores_promedio_{(clase)} * Cplex_errores(t,clase) \quad (41)$$

Donde:

Errores_promedio_(e,a)=Reducir la ponderación total del proyecto actual

13. Determinar la asignación de las actividades con base en el nivel de dominio que tienen los empleados en las diferentes clases de actividades.

$$Max \sum_{e=1}^m \sum_{clase=1}^n habilidades_{(e,clase)} * Cplex_habilidades(e,clase) \quad (42)$$

Donde:

$habilidades_{(e,a)}$ = Maximizar la ponderación de habilidades de los trabajadores del proyecto actual

14. Determinar la complejidad del proyecto

$$C = \left(\sum_{i=1}^n a_p / A \right) * 3 + \left(\sum_{i=1}^n req_p / A \right) + \left(\sum_{i=1}^n e_p / E \right) * 2 + \left(\left(\sum_{i=1}^n a_{cu} / n \right) / c_{(e,a)} \right) + \%f + \left(\sum_{i=1}^n c_{(e,a)} / t_{(e,a)} \right) * 2 \quad (43)$$

Donde: CAMBIAR SUM A_{CU}

a_p = Maximizar la ponderación de habilidades de los trabajadores del proyecto actual

A = Conjunto de actividades del proyecto actual

req_p = conjunto de requerimientos del proyecto

e_p = Número de trabajadores seleccionados de acuerdo a la función objetivo.

E = Número de empleados que se disponen para el proyecto actual

a_{cu} = Promedio de actividades del proyecto

n = Número total de actividades del proyecto

$c_{(e,a)}$ = Costo unitario por actividad a del trabajador e

$\%f$ = porcentaje de la fuerza laboral empleada, determinada de la división del número de trabajadores seleccionados entre el total de trabajadores destinados al proyecto.

$t_{(e,a)}$ = tiempo unitario del trabajador e para realizar la actividad a

Comúnmente, la complejidad inherente al desarrollo de software se deriva de 4 elementos (Booch, 1994):

- Complejidad del dominio del problema. Se relaciona con la facilidad ó dificultad relacionada con el uso del software desarrollado. Este elemento se emplea en la etapa de implementación y evaluación del software a través del uso del mismo por parte de los usuarios y clientes.
- La dificultad de gestionar el proceso de desarrollo. Es una ponderación que se le asigna a un proyecto con base en el número de personas que integran el equipo de trabajo y de la cantidad de código que se genera en las etapas de programación y pruebas. Este elemento se emplea en las etapas de desarrollo de software (análisis, diseño, desarrollo e implementación).
- El detalle que se puede alcanzar a través del software. Se enfoca en determinar la necesidad del desarrollo de software a partir de la reutilización de código, la selección y empleo de estándares. Este elemento se emplea en las etapas de diseño y programación.
- El problema de caracterizar el comportamiento de sistemas discretos. Cada estado debe contener las variables, sus valores, direcciones en tiempo de ejecución, entren otros; donde cada estado debe afectar lo mínimo en otras partes del sistema, cuya transición entre estados debe ser determinista. Este elemento se emplea en la etapa de diseño del software.

Para la presente investigación, se define la complejidad de software como la ponderación derivada de los recursos disponibles (recursos humanos, disponibilidad de tiempo) entre los requerimientos del proyecto de software (costo, tiempo y alcance), a partir del análisis de los siguientes indicadores:

- a) Cantidad de requerimientos: número de requerimientos que se definen en la etapa de análisis de requerimientos.
- b) Cantidad de actividades: Planteamiento inicial del número total de actividades propuestas para el desarrollo del proyecto de software.
- c) Actividades por clase: clasificación de las actividades en las clases propuestas: análisis, diseño, programación, evaluación, implementación, etc.

- d) tiempo total estimado del proyecto: Tiempo generado a partir del modelo PSPSWD generado en AIMMS.
- e) costo total estimado del proyecto: Costo total estimado a partir del modelo PSPSWD generado en AIMMS.
- f) Número de trabajadores involucrados: Número de trabajadores estimado a partir del modelo PSPSWD generado en AIMMS.
- g) Esfuerzo Horas/Hombre determinado por el porcentaje de trabajadores empleados en el proyecto: Porcentaje de la fuerza laboral empleada en el proyecto.
- h) La estimación del índice de complejidad (fórmula 43), es la razón estimada directamente proporcional a la sumatoria de los indicadores e inversamente proporcional al inverso del producto de los mismos.

CAPÍTULO 4. EXPERIMENTACIÓN Y RESULTADOS

Las etapas consideradas en la presente tesis para el desarrollo de proyectos de software se muestra en la figura 17, donde se considera como punto de partida el análisis de requerimientos que se determinan a partir de las necesidades de los clientes, de los usuarios, entre otros roles relacionados con el producto de software a generar.

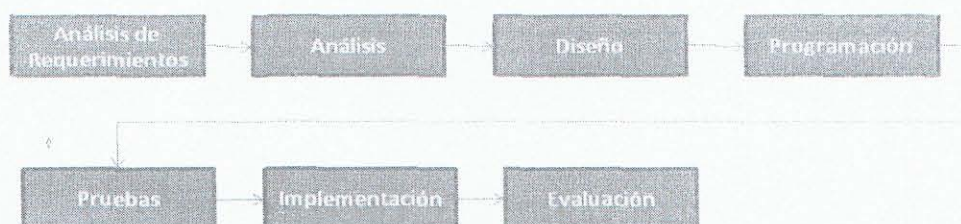


Figura 17. Etapas para el desarrollo de software

El modelo de indicadores para evaluar la complejidad en proyectos de software, será implementado entre las etapas de análisis de requerimientos y análisis de software (figura 18). Al concluir la primera etapa, se obtiene un listado de requerimientos que deberán ser cubiertos por un número estimado de actividades pertenecientes a las clases de actividades: análisis de software, diseño de software, programación, pruebas de software, implementación y evaluación. A partir de esta asignación, el presente modelo tendrá como resultado:

- Una asignación de empleados con base en el número de actividades por clase; el número máximo de actividades que cada trabajador le dedicará al proyecto, grado de dominio del empleado en el desarrollo de actividades por clase, nivel de dominio que tiene cada trabajador en el desarrollo de actividades por clase y el número de errores promedio que tiene un trabajador al desarrollar las actividades por clase;
- Una asignación del costo total del proyecto con base en el costo por actividad que tiene cada trabajador, al costo unitario de cada actividad/trabajador;
- Una asignación del tiempo total del proyecto con base en el tiempo por actividad que tiene cada trabajador, el tiempo que le dedicará cada trabajador al proyecto, el tiempo unitario para el desarrollo de actividades por clase y trabajador; y

- Determinar el nivel de complejidad que tiene el proyecto analizado con base en las características de tiempo, costo y alcance del proyecto, considerando recursos humanos, costos y número de actividades.

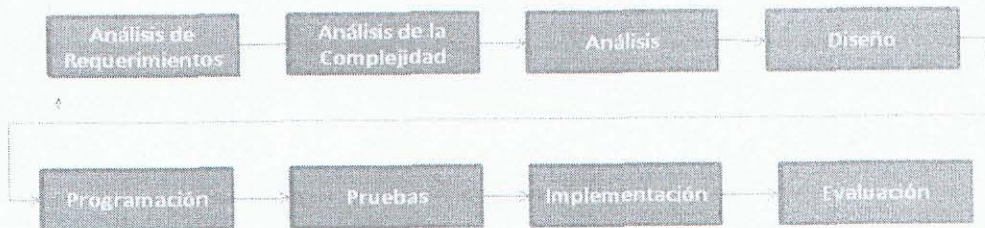


Figura 18. Análisis de complejidad.

4.1 Representación del modelo matemático en la herramienta AIMMS 3.10

Entorno de trabajo

El equipo de cómputo y software empleados en la comprobación de las instancias son:

1. Sistema operativo: Windows vista home basic service pack 2 de 32 bits.
2. Computadora: Dell inspiron INSP1440, Intel Core duo T6500 2.1 GHz, memoria RAM 4 GB.
3. AIMMS ver 3.10 SU1 (build 3.10.5.16043).

Nota: Las pruebas se aplicaron únicamente con el programa AIMMS en la ejecución de las instancias.

Prueba inicial

En la primera instancia empleada para el modelado de la solución se generaron las siguientes gráficas:

1. Registro de conjuntos de datos e índices que definen la profundidad y dimensión del modelo proporcionando un mecanismo para agrupar los parámetros, variables y restricciones:
 - a. Clases de actividades.

b. Trabajadores que componen los recursos humanos de la empresa.

trabajadores

Type: Set

Identifier: trabajadores

Index domain: ☒

Subset of: ☒

Text: ☒

Index: ☒

Parameter: ☒

Property: ☒

Order by: ☒

Definition: ☒

Initial data: ☒

Figura 19. Índice “t” relacionado con los trabajadores que participarán en el proyecto.

trabajadores [Data Page] trabajadores

+ | x |

t1

t2

t3

t4

t5

t6

t7

t8

t9

t10

Figura 20. Conjunto de datos de trabajadores

actividades

Type: Set

Identifier: actividades

Index domain: ☒

Subset of: ☒

Text: ☒

Index: ☒

Parameter: ☒

Property: ☒

Order by: ☒

Definition: ☒

Initial data: ☒

Figura 21. Índice “a” relacionado con las clases de actividades del proyecto.

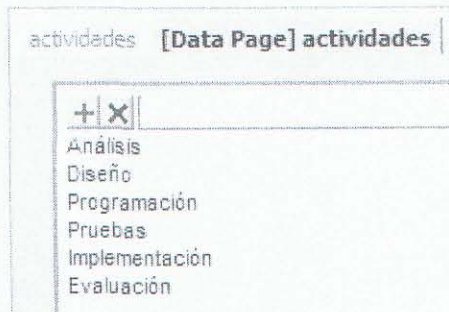


Figura 22. Conjunto de datos de las clases de actividades.

2. Registro de parámetros. En AIMSS son definidos como una cantidad conocida que contiene un valor numérico ó de tipo cadena de texto:



Figura 23. Nivel de habilidad del trabajador para el desarrollo de actividades por clase.

NivelHabilidadesTrabajador [Data Page] NivelHabilidadesTr...

	t	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
Análisis		5	9	6	9	8	7	6	9	6	8
Diseño		9	6	8	9	6	5	4	7	9	3
Programación		6	7	9	5	7	8	9	8	7	6
Pruebas		6	5	4	7	9	8	6	6	9	9
Implementación		7	9	6	3	9	4	8	6	8	6
Evaluación		8	4	9	8	7	6	5	5	9	9

Figura 24. Nivel de habilidad de los trabajadores definidos en el conjunto de datos de trabajadores

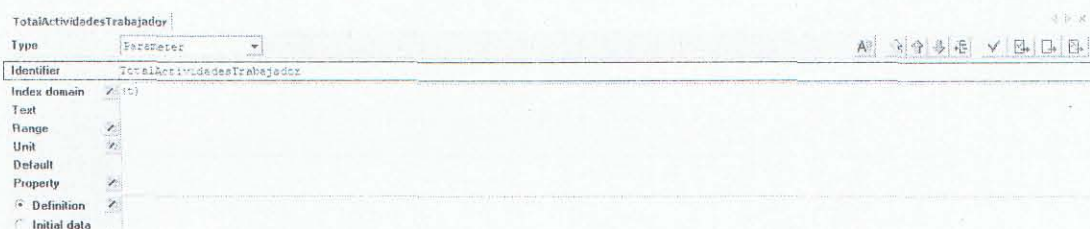


Figura 25. Número de actividades que podrá desempeñar cada trabajador en el proyecto actual.

TotalActividadesTrabajador [Data Page] TotalActividadesTr...

t	
t1	20
t2	20
t3	20
t4	20
t5	20
t6	20
t7	20
t8	20
t9	20
t10	20

Figura 26. Asignación del número máximo de actividades por trabajador.

TotalTiempoTrabajador

Type Parameter

Identifier TotalTiempoTrabajador

Index domain ☒ (t)

Test ☒

Range ☒

Unit ☒

Default ☒

Property ☒

Definition ☒

Initial data ☐

Figura 27. Tiempo que le dedicará cada trabajador al proyecto actual.

TotalTiempoTrabajador [Data Page] TotalTiempoTrabaj...

t	
t1	100
t2	100
t3	100
t4	100
t5	100
t6	100
t7	100
t8	100
t9	100
t10	100

Figura 28. Asignación del tiempo total que le dedicará cada trabajador al proyecto actual.

TotalActividadesClase

Type: Parameter

Identifier: TotalActividadesClase

Index domain: a

Test: /

Range: /

Unit: /

Default: /

Property: /

Definition: /

Initial data: /

Figura 29. Parámetro relacionado con el número total de actividades por clase.

TotalActividadesClase [Data Page] TotalActividadesCl...

a	
Análisis	20
Diseño	30
Programación	20
Pruebas	10
Implementación	10
Evaluación	10

Figura 30. Asignación del número de actividades por clase del proyecto actual.

CostoUnitarioActividad

Type: Parameter

Identifier: CostoUnitarioActividad

Index domain: (t, a)

Test: /

Range: /

Unit: /

Default: /

Property: /

Definition: /

Initial data: /

Figura 31. Parámetro relacionado con el costo unitario de las actividades por clase y por trabajador.

CostoUnitarioActividad [Data Page] CostoUnitarioActiv...

	a	Análisis	Diseño	Programación	Pruebas	Implementación	Evaluación
t							
t1		1000	2000	3000	1000	500	500
t2		1000	2000	3000	1000	500	500
t3		2000	3000	1000	500	1000	500
t4		2000	3000	1000	500	1000	500
t5		3000	1000	2000	500	500	1000
t6		3000	1000	2000	500	500	1000
t7		500	1000	500	3000	2000	500
t8		500	1000	500	3000	2000	500
t9		1000	500	500	2000	3000	1000
t10		1000	500	500	2000	3000	1000

Figura 32. Asignación del costo de una actividad por clase y por trabajador considerando aspectos indirectos como: salario, nivel de habilidad, entre otros aspectos.

TimepoUnitarioActividad

Type: Parameter

Identifier: TimepoUnitarioActividad

Index domain: (t, s)

Text: []

Range: []

Unit: []

Default: []

Property: []

Definition: []

Initial data: []

Figura 33. Parámetro relacionado con el tiempo promedio que se tiene en el desarrollo de una actividad por clase.

TimepoUnitarioActividad [Data Page] TimepoUnitarioAct...

	a	Análisis	Diseño	Programación	Pruebas	Implementación	Evaluación
t							
t1		10	15	10	20	5	3
t2		8	10	12	10	14	4
t3		15	25	20	10	15	5
t4		10	23	15	12	20	6
t5		25	20	25	15	15	7
t6		23	12	15	3	3	8
t7		3	8	10	10	25	9
t8		5	10	4	30	20	12
t9		10	10	5	15	10	14
t10		7	5	20	20	12	4

Figura 34. Tiempo promedio que le dedica un trabajador a una actividad por clase.

ErroresTrabajador

Type: Parameter

Identifier: ErroresTrabajador

Index domain: (t, s)

Text: []

Range: []

Unit: []

Default: []

Property: []

Definition: []

Initial data: []

Figura 35. Parámetro relacionado con el número de errores promedio que tiene cada trabajador en el desarrollo de actividades por clase.

	a	Análisis	Diseño	Programación	Pruebas	Implementación	Evaluación
t							
t1		3	3	2	1	3	3
t2		2	4	1	2	2	2
t3		3	2	1	5		4
t4		2	3	4	3	1	1
t5		1	6	5	3	1	2
t6		1	4	2	4	5	6
t7		5	5	6	2	6	9
t8		2	8	5	3	4	2
t9		5	7	3	2	3	1
t10		2	6	8	1	2	5

Figura 36. Asignación del número promedio de errores que tiene un trabajador en el desarrollo de actividades por clase.

- Registro de variables. En AIMMS, las variables son definidas como cantidades desconocidas que serán calculadas a partir de las funciones matemáticas y de los procesos de maximizar ó minimizar las funciones objetivo del modelo:

DesarrolloActividades

Type: Variable

Identifier: DesarrolloActividades

Index domain: ☒ (t, A)

Text: ☐

Range: ☒ nonnegative

Unit: ☒

Default: ☐

Property: ☒

Nonvar status: ☒

Definition: ☐

Figura 37. Variable donde se guardará la asignación del tiempo que le dedicará cada trabajador a las diferentes clases de actividades con base en el número máximo de actividades por trabajador definidas en TotalTiempoTrabajador(t).

	a	Análisis		Diseño		Programación		Pruebas		Implementación		Evaluación	
	Suffix	Level	basic	Level	basic	Level	basic	Level	basic	Level	basic	Level	basic
	t												
t1			NonBasic		10 Basic		NonBasic		10 Basic		NonBasic		NonBasic
t2			NonBasic		NonBasic		20 Basic		NonBasic		NonBasic		NonBasic
t3			NonBasic		10 Basic		NonBasic		NonBasic		10 Basic		NonBasic
t4			NonBasic		10 Basic		NonBasic		NonBasic		NonBasic		NonBasic
t5			Basic		NonBasic		NonBasic		NonBasic		NonBasic		NonBasic
t6			20 Basic		NonBasic		NonBasic		NonBasic		NonBasic		NonBasic
t7			NonBasic		NonBasic		NonBasic		NonBasic		NonBasic		NonBasic
t8			NonBasic		NonBasic		NonBasic		NonBasic		NonBasic		NonBasic
t9			NonBasic		NonBasic		NonBasic		NonBasic		NonBasic		10 Basic
t10			NonBasic		NonBasic		NonBasic		NonBasic		NonBasic		NonBasic

Figura 38. Resultado obtenido del modelo como resultado de la prueba inicial

CostoTotalDesarrollo	
Type	Variable
Identifier	CostoTotalDesarrollo
Index domain	☒
Test	
Range	☒ free
Unit	☒
Default	
Property	☒
Nonvar status	☒
Definition	$\text{sum}[(t,a), \text{CostoUnitarioActividad}(t,a) * \text{DesarrolloActividades}(t,a)]$

Figura 39. Variable relacionada con el costo total del desarrollo del proyecto a partir de minimizar el costo con base en los costos unitarios por actividad.

DesarrolloTiempo	
Type	Variable
Identifier	DesarrolloTiempo
Index domain	☒ (t,a)
Test	
Range	☒ free
Unit	☒
Default	
Property	☒
Nonvar status	☒ TiempoUnitarioActividad(t,a)
Definition	

Figura 40. Variable que vincula las actividades con los trabajadores.

TiempoTotalDesarrollo	
Type	Variable
Identifier	TiempoTotalDesarrollo
Index domain	☒
Test	
Range	☒ free
Unit	☒
Default	
Property	☒
Nonvar status	☒
Definition	$\text{sum}[(t,a), \text{TiempoUnitarioActividad}(t,a) * \text{DesarrolloActividades}(t,a)]$

Figura 41. Variable relacionada con el total de tiempo del proyecto a partir de minimizar el tiempo con base en el tiempo total que le dedica cada trabajador al proyecto y el número máximo de actividades.

ErroresTotalesDesarrollo	
Type	Variable
Identifier	ErroresTotalesDesarrollo
Index domain	☒
Test	
Range	☒ free
Unit	☒
Default	
Property	☒
Nonvar status	☒
Definition	$\text{sum}[(t,a), \text{ErroresTrabajador}(t,a) * \text{DesarrolloActividades}(t,a)]$

Figura 42. Variable que minimiza el número total de errores con base en el parámetro $\text{ErroresTrabajador}(t,a)$.

AsignarHabilidades	
Type	Variable
Identifier	AsignarHabilidades
Index domain	☒
Text	☒
Range	☒
Unit	☒
Default	☒
Property	☒
Nonvar status	☒
Definition	$\sum_{s \in S} (r, s) \cdot \text{NivelHabilidadesTrabajador}(s, s) \cdot \text{DesarrolloActividades}(t, s)$

Figura 43. Variable donde se asignan las actividades con base en el nivel de habilidad que tiene cada trabajador para el desarrollo de actividades por clase. El objetivo es maximizar el nivel total de habilidades.

4. Restricciones. Las restricciones conforman el mecanismo principal para especificar las restricciones sobre el rango de valores que pueden tomar las variables:

RestriccionTrabajadores	
Type	Constraint
Identifier	RestriccionTrabajadores
Index domain	☒ (t)
Text	☒
Unit	☒
Property	☒
Definition	$\text{sum}[(s), \text{DesarrolloActividades}(t, s)] \leq \text{TotalActividadesTrabajador}(t)$

Figura 44. La asignación del número de actividades por trabajador debe ser menor ó igual al número máximo de actividades que dicho trabajador desarrollará en el proyecto actual.

RestriccionActividades	
Type	Constraint
Identifier	RestriccionActividades
Index domain	☒ (t)
Text	☒
Unit	☒
Property	☒
Definition	$\text{sum}[(t), \text{DesarrolloActividades}(t, s)] \leq \text{TotalActividadesClase}(s)$

Figura 45. Un trabajador no puede ser asignado más horas de las que dispone para el proyecto actual.

5. Funciones matemáticas. El objetivo principal, consiste en encontrar la solución óptima que se asume en la función objetivo (maximizar o minimizar). Constan de un conjunto de datos indeterminados, una colección de restricciones especificadas y una función objetivo a ser optimizada.

PlanMenorCosto	
Type	Mathematical Progr
Identifier	PlanMenorCosto
Objective	☒ CostoTotalDesarrollo
Direction	☒ minimize
Constraints	☒
Variables	☒
Text	☒
Type	☒
Violation penalty	
Comment	

Figura 46. Función matemática para minimizar el costo total del proyecto a partir de los parámetros: $TotalActividadesTrabajador(t,a)$, $TotalActividadesClase(a)$ y $CostoUnitarioActividad(t,a)$; y a partir de las funciones: $DesarrolloActividades(t,a)$ y $CostoTotalDesarrollo$.

PlanMenorTiempo	
Type	Mathematical Progr
Identifier	PlanMenorTiempo
Objective	☒ TiempoTotalDesarrollo
Direction	☒ minimize
Constraints	☒
Variables	☒
Text	☒
Type	☒
Violation penalty	
Comment	

Figura 47. Función matemática para minimizar el tiempo total para el desarrollo del proyecto actual a partir de los parámetros: $TotalTiempoTrabajador(t)$ y $TiempoUnitarioActividad(t,a)$; y a partir de la función $TiempoTotalDesarrollo$.

PlanMenorErrores	
Type	Mathematical Progr
Identifier	PlanMenorErrores
Objective	☒ ErroresTotalesDesarrollo
Direction	☒ minimize
Constraints	☒
Variables	☒
Text	☒
Type	☒
Violation penalty	
Comment	

Figura 48. Función matemática para minimizar el número total de errores a partir del parámetro $ErroresTrabajador(t,a)$; y a partir de la variable $ErroresTotalesDesarrollo$.

HabilidadesActividad	
Type	Mathematical Progr
Identifier	HabilidadesActividad
Objective	☒ AsignarHabilidades
Direction	☒ maximize
Constraints	☒
Variables	☒
Text	☒
Type	☒
Violation penalty	
Comment	

Figura 49. Función matemática para maximizar la asignación del nivel de habilidades en el desarrollo de las actividades a partir del parámetro $NivelHabilidadesTrabajador(a,t)$ y de la variable $AsignarHabilidades$.

6. Ejecución principal. Función principal donde se establecen las declaraciones necesarias para ejecutar la aplicación a partir de la función objetivo.

Figure 50 shows the main execution procedure in AIMMS. The procedure is named `MainExecution` and is located in the `MainExecution` package. The body of the procedure is as follows:

```

Procedure
MainExecution
Body
  solve PlanMenorCosto;
  solve PlanMenorTiempo;
  solve PlanMenorErrores;
  if ((PlanMenorCosto.ProgramStatus <> "Optimal") and (PlanMenorTiempo.ProgramStatus <= "Optimal")) then
    if ((PlanMenorErrores.ProgramStatus <= "Optimal") and (HabilidadesActividad.ProgramStatus <= "Optimal")) then
      empty DesarrolloActividades, CostoTotalDesarrollo, TiempoTotalDesarrollo;
    end if;
  end if;
end;
  
```

Figura 50. Declaración de la función principal a partir del algoritmo CPLEX.

7. Interfaz de salida. AIMMS define un conjunto de gráficas y tablas para representar los datos recopilados en los parámetros y los datos generados en las variables (Figura 51).

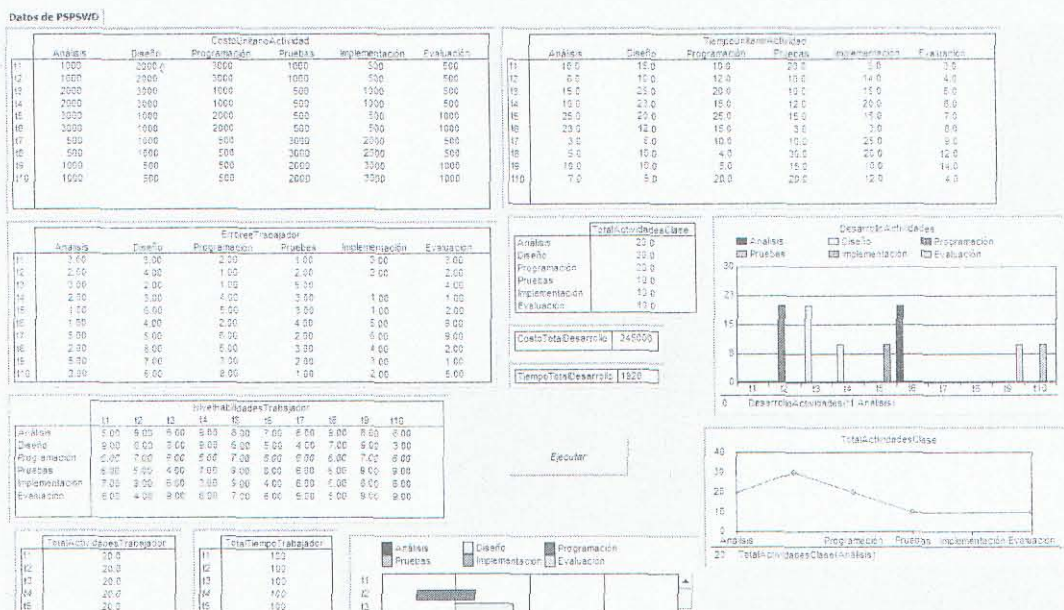


Figura 51. Interfaz generada a partir de la prueba inicial.

Análisis de resultados

Para la obtención de los resultados, se realizaron las siguientes actividades:

- Análisis de un proyecto por instancia (sección proyecto actual) con base en las características de los recursos requeridos para el desarrollo de dicho proyecto.

- Selección y Clasificación de indicadores en el desarrollo de los proyectos.
- Metodología para identificar la complejidad de los proyectos con base en los recursos empleados y en indicadores de desarrollo de software.
- Modelo para el análisis de indicadores para la toma de decisiones y predicción de escenarios en el desarrollo de proyectos de software.

Se realizaron 25 instancias (tabla 11) donde se estableció la complejidad de proyectos de software (fórmula 43) a partir del modelo matemático de optimización, con base en datos aleatorios, donde se calcula el tiempo total y el costo total del proyecto con base en combinaciones (iteraciones) entre los recursos con que cuenta la empresa para cada proyecto; las restricciones relacionadas con costo actividad/trabajador; minimizar el costo total del proyecto; minimizar el tiempo total; minimizar el número promedio de errores; y maximizar el uso de los recursos humanos con mejores características.

La complejidad está determinada por un valor ponderado entre 1 y 10, donde 1 es el valor mínimo, es decir, en este caso, la empresa cuenta con recursos excesivos para cubrir las necesidades planteadas en el análisis de requerimientos; y 10 el valor máximo de complejidad donde los requerimientos identificados en el proyecto a analizar deberán ser cubiertos por al menos el 90% del total de los recursos de la empresa destinados al proyecto, dando un margen del 10% para realizar combinaciones en la asignación de recursos.

En la tabla 11 se presentan los resultados obtenidos:

- El número de instancia representa las instancias empleadas (anexos);
- El indicador de complejidad es el índice desarrollado para determinar la complejidad del proyecto de acuerdo a la ponderación entre los recursos requeridos para desarrollar el proyecto y los recursos destinados para resolverlo;
- El número de iteraciones, es el número de posibles soluciones que se realizaron mediante el método CPLEX para determinar la iteración que cumple con la función objetivo;

- La memoria empleada, es el tamaño de memoria requerida para poder ejecutar el método CPLEX con el software AIMSS para determinar la combinación óptima de recursos;
- Costo total de desarrollo y Tiempo total de desarrollo, son los datos óptimos obtenidos a partir de la función objetivo, considerando minimizar tanto tiempo total como costo total del proyecto, entre otros factores explicados anteriormente.

Para una instancia con grado de complejidad 10, los indicadores deberían tener los siguientes valores:

- Emplear el 90% de la fuerza laboral;
- El indicador que representa el número de requerimientos entre el total de actividades se aproxima a 1;
- La ponderación del esfuerzo horas/hombre determinado por el responsable del proyecto es del 90%;
- El número de actividades para este proyecto entre el total de actividades que se pueden desarrollar es del 90%;
- La relación del tiempo total entre el costo total es superior a 1;
- Para la selección de los trabajadores no se realizaron suficientes combinaciones para asignarlos con el número mínimo de errores ni con base al grado de habilidad en el desarrollo de cada actividad.

En la tabla 12, se muestran las 25 instancias ordenadas por el índice de complejidad. En esta tabla se muestra el comportamiento del modelo donde los diferentes indicadores afectan de forma directa los resultados: índice de complejidad, monto total del proyecto y costo total del proyecto. Por ejemplo, en las instancias 3 y 2, se tienen recursos similares: mismo número de actividades de análisis, diseño, programación, pruebas, implementación y evaluación, y la misma ponderación del número de trabajadores seleccionados en el proyecto a partir de la función objetivo entre el número de trabajadores destinados a dicho proyecto (son 6 trabajadores entre un total de 10: la ponderación es del 60%). Estas instancias varían en la ponderación del indicador de actividades solicitadas entre el total de actividades que pueden

desarrollar los trabajadores involucrados: para la instancia 3 se tiene una ponderación del 0.6, es decir, que se deberá cubrir el 60% del total de actividades y para la instancia 2, se tiene una ponderación del 40%; por otra parte, en la instancia 3 se emplea el 10% de la fuerza laboral total de la empresa y para el caso de la instancia 2 se emplea el 20% de dicha fuerza laboral.

Tabla 11. Resultado de 25 instancias.

No. instancia	Complejidad	Número de Iteraciones	Memoria empleada	Costo total de desarrollo	Tiempo total de desarrollo
1	4.3	31	69 Mb	245,000.00	1920
2	4.1	23	68.2 Mb	1'690,000.00	19330
3	3.7	28	69 Mb	190'000,000.00	2'129,000
4	5.8	30	69.7 Mb	6,150'000,000.00	69'800,000
5	6.2	36	69.6 Mb	60,500'000,000.00	6,670'00,000
6	4.8	17	69.8 Mb	6,050'000,000.00	66'700,000
7	4.3	17	69.9 Mb	86,000'000,000.00	47'600,000
8	4.6	17	70 Mb	86,000'000,000.00	47,600'000,000
9	7.4	32	68.1 Mb	854,400'000,000.00	796,200'000,000
10	6.4	26	69.3 Mb	854,400'000,000.00	796,200'000,000
11	5.6	39	69.6 Mb	909,200'000,000.00	837,000'000,000
12	6.6	47	69.9 Mb	909,200'000,000.00	837,000'000,000
13	7.0	65	70.0 Mb	954,000'000,000.00	882,600'000,000
14	7.2	52	70.3 Mb	963,000'000,000.00	942,600'000,000
15	6.8	36	70.7 Mb	998,200'000,000.00	976,200'000,000
16	7.3	42	70.8 Mb	1'060,200'000,000.00	1'000,000'000,000
17	7.6	95	71.2 Mb	1'132,200'000,000.00	999,400'000,000
18	6.8	59	71.4 Mb	1'255,000'000,000.00	1'114,000'000,000
19	7.3	44	71.5 Mb	1'223,400'000,000.00	1'175,400'000,000
20	7.5	68	71.6 Mb	1'229,400'000,000.00	1'149,400'000,000
21	7.5	64	72.0 Mb	1'342,000'000,000.00	1'363,600'000,000
22	6.1	67	63.2 Mb	1'301,000'000,000.00	1'249,600'000,000
23	6.9	76	63.7 Mb	1'368,000'000,000.00	1'188,000'000,000
24	7.4	77	63.8 Mb	1'571,200'000,000.00	1'604,000'000,000

25	7.5	94	64 Mb	1'584,200'000,000.00	1'698,800'000,000
----	-----	----	-------	----------------------	-------------------

Por ejemplo, la instancia 25, se tienen 164'000,000 de actividades posibles que pueden desarrollar 50 trabajadores (Anexo – instancia 25) y se necesitan cubrir 141'200,000.00 (86% del total de actividades que se pueden desarrollar). Con base en las características de los recursos empleados en esta instancia, se determinó que el costo total del proyecto es de 1'584,200'000,000.00 y llevará un tiempo total en horas de 1'698,800'000,000, a partir de la combinación de los recursos disponibles vs los recursos solicitados para el desarrollo del proyecto, se determinó que tendrá una complejidad del 7.5 considerando que se emplearán:

- El 98% de la fuerza laboral;
- El indicador que representa el número de requerimientos entre el total de actividades es de 0.3;
- La ponderación del esfuerzo horas/hombre determinado por el responsable del proyecto es del 27%;
- El número de actividades para este proyecto entre el total de actividades que se pueden desarrollar es del 86%.
- La relación del tiempo total entre el costo total es de 1.07;
- La selección de los trabajadores se llevó a cabo en función del número mínimo de errores y de aquellos trabajadores que presentan un alto grado de habilidad en el desarrollo de cada actividad.

Tabla 12. Datos para establecer la complejidad

No	Complejidad	No. de requerimientos	Análisis	Diseño	Programación	Pruebas	Implementación	Evaluación	Total Actividades solicitadas	Total Actividades Trabajadores	No. Trabajadores empresa	No. Trabajadores involucrados	Esfuerzo hrs/hombre	Monto total	Tiempo total
3	3.7	10.000	15.000	15.000	14.000	14.000	14.000	7.000	129.000	214.000	10	6	10	190.000.000.00	2.129.000
2	4.1	5.000	15.000	15.000	14.000	14.000	14.000	7.000	1.200	3.000	10	6	20	1.690.000.00	19.330.00
7	4.3	100.000	236.000	182.000	202.000	266.000	221.000	255.000	40.000.000	60.000.000	15	6	95	86.000.000.000.00	47.600.000
1	4.3	3.000	15.000	15.000	14.000	14.000	14.000	7.000	100	200	10	5	30	245.000.00	1.920.00
8	4.6	123.000	236.000	182.000	202.000	266.000	221.000	255.000	40.000.000	60.000.000	15	5	30	86.000.000.000.00	47.600.000.000
6	4.8	50.000	34.000	34.000	33.000	33.000	33.000	12.000	40.000.000	60.000.000	15	10	90	6.050.000.000.00	66.700.000
11	5.6	345.000	331.000	272.000	292.000	366.000	304.000	360.000	51.000.000	87.000.000	24	17	12	909.200.000.000.00	837.000.000.000
4	5.8	12.000	15.000	15.000	14.000	14.000	14.000	7.000	4.000.000	4.400.000	10	10	50	6.150.000.000.00	69.800.000
22	6.1	250.000	577.000	588.000	613.000	682.000	572.000	665.000	103.000.000	164.000.000	50	36	37	1.301.000.000.000.00	1.249.600.000.000
5	6.2	20.000	15.000	15.000	14.000	14.000	14.000	7.000	40.000.000	44.000.000	10	10	80	60.500.000.000.00	6.670.000.000
10	6.4	320.000	322.000	263.000	283.000	357.000	295.000	340.000	51.000.000	79.000.000	22	17	56	854.400.000.000.00	796.200.000.000
12	6.6	356.700	331.000	272.000	292.000	366.000	304.000	360.000	65.000.000	87.000.000	24	18	50	909.200.000.000.00	837.000.000.000
18	6.8	4.200	420.000	406.000	441.000	500.000	414.000	485.000	90.000.000	121.000.000	35	30	56	1.255.000.000.000.00	1.114.000.000.000
15	6.8	200.000	388.000	344.000	379.000	448.000	369.000	435.000	76.000.000	108.000.000	30	24	65	998.200.000.000.00	976.200.000.000
23	6.9	324.000	581.000	598.000	640.000	725.000	644.000	290.000	133.000.000	164.000.000	50	41	54	1.368.000.000.000.00	1.188.000.000.000
13	7.0	401.000	331.000	272.000	292.000	366.000	304.000	360.000	72.000.000	87.000.000	24	21	43	954.000.000.000.00	882.600.000.000
14	7.2	415.000	288.000	308.000	316.000	335.000	316.000	138.000	76.000.000	87.000.000	24	23	23	963.000.000.000.00	942.600.000.000
16	7.3	300.000	349.000	372.000	384.000	439.000	397.000	180.000	81.000.000	108.000.000	30	27	87	1.060.200.000.000.00	1.000.000.000.000
19	7.3	100	405.000	426.000	448.000	513.000	458.000	210.000	93.000.000	121.000.000	35	30	89	1.223.400.000.000.00	1.175.400.000.000
9	7.4	200.000	292.000	253.000	263.000	332.000	270.000	330.000	66.000.000	77.000.000	21	21	43	854.400.000.000.00	796.200.000.000
24	7.4	500.000	581.000	598.000	640.000	725.000	644.000	290.000	147.000.000	164.000.000	50	47	33	1.571.200.000.000.00	1.604.000.000.000
20	7.5	800.000	405.000	426.000	448.000	513.000	458.000	210.000	103.000.000	121.000.000	35	32	70	1.229.400.000.000.00	1.149.400.000.000

25	7.5	500.000	581.000	598.000	640.000	725.000	644.000	290.000	141.200.000	164.000.000	50	50	27	1.584.200.000.000.00	1.698.800.000.000
21	7.5	400.000	405.000	426.000	448.000	513.000	458.000	210.000	103.000.000	121.000.000	35	35	44	1.342.000.000.000.00	1.363.600.000.000
17	7.6	25.000	349.000	372.000	384.000	439.000	397.000	180.000	90.000.000	108.000.000	30	28	97	1.132.200.000.000.00	999.400.000.000

En la tabla 13 se presenta el concentrado que refleja las diferencias entre las instancias 2 y 3, donde se considera la instancia 3 con una complejidad del 3.7 porque se está empleando únicamente el 10% de la fuerza laboral total de la empresa; para la instancia 2, se emplea el 20% de la fuerza laboral total de la empresa; es este factor el principal indicador para la diferencia de complejidad entre ambas instancias.

Tabla 13. Análisis de las instancias 3 y 2.

No	Complejidad	Ponderación de actividades	Esfuerzo hrs/hombre	Monto total	Tiempo total
3	3.7	0.6	0.1	190,000,000.00	2,129.000
2	4.1	0.4	0.2	1,690,000.00	19,330.00

Para la asignación de los trabajadores (t) a cada clase de actividades, se considera la combinación de los siguientes aspectos de acuerdo a la función objetivo:

- Nivel de habilidad por clase de actividad.
- Tiempo que dispone el trabajador para el desarrollo del proyecto actual.
- Número de actividades que puede realizar el trabajador en el proyecto actual.
- Ponderación de errores por clase de actividad.
- Costo por actividades por clase de cada trabajador.
- Tiempo por actividad por clase de cada trabajador.

Con el método CPLEX, se llevan a cabo las diferentes combinaciones de los trabajadores destinados al proyecto con base en los aspectos mencionados anteriormente, para cumplir con la función objetivo relativa a maximizar el nivel de desarrollo del proyecto, es decir, que sean seleccionados los trabajadores que se mejor se desempeñen en las diferentes clases de actividades con el mejor costo, el menor tiempo, seleccionando los trabajadores con mayor nivel de habilidad y con la ponderación menor relativa a errores (tabla 14).

Tabla 14. Asignación de trabajadores por clase de actividad.

No	Análisis	Diseño	Programación	Pruebas	Implementación	Evaluación
----	----------	--------	--------------	---------	----------------	------------

1	t6	t3, t4	t2	t10	t5	t9
2	t6	t3	t2, t6	t1, t6, t10	t5	t4, t9
3	t5	t1, t3, t4, t6	t2	t9, t10	t5	t9
4	t6	t3	t6	t1, t2, t7, t8, t10	t3, t4, t5	t4, t8
5	t6	t3, t6	t2	t1, t2, t6, t8, t10	t3, t5, t7	t4, t8, t9
6	t6	t3	t3	t1, t2, t6, t8, t10	t3, t4, t5, t7	t4, t8, t9
7	t5	t15	t2	t10, t15	t3	t4
8	t5	t15	t2	t10, t15	t3	t4
9	t5, t8	t14, t15, t20	t2, t21	t1, t6, t8, t9, t10, t11, t16, t17, t19	t3, t4, t7	t12, t13, t18, t19
10	t6, t8	t14, t15, t20	t2, t3	t1, t8, t9, t10, t15, t16, t21, t22	t3, t4, t5, t7, t14	t11, t14
11	t6, t8	t14, t15, t20	t2	t1, t8, t9, t10, t15, t16, t21, t22	t3, t4, t5, t7, t14	t11, t14
12	t6, t8	t15, t21, t24	t2	t1, t8, t9, t10, t14, t16, t22	t3, t4, t5, t7, t14	t11, t12
13	t6, t8	t11, t20, t21, t24	t2, t15, t20	t1, t8, t9, t10, t11, t14, t16, t22	t3, t4, t5, t7, t11	t11, t12, t13, t17
14	t6, t8	t20, t21, t24	t2, t11, t15, t17	t1, t8, t9, t10, t11, t14, t16, t22	t3, t4, t5, t7, t11	t12, t17, t18, t19, t23
15	t6, t8	t17, t20	t2, t14, t15	t1, t7, t8, t9, t10, t11, t16, t22	t3, t4, t5, t11, t21, t24	t12, t13, t18, t19, t23
16	t6, t8, t30	t14, t15, t20	t2, t3, t26, t27	t1, t9, t6, t8, t9, t10, t15, t16, t21, t22, t25	t3, t4, t5, t7, t28, t29	t17, t18, t19, t23, t24
17	t6, t8, t30	t20, t24	t2, t3, t26, t27	t1, t6, t8, t9, t10, t14, t15, t16, t21, t22, t25	t3, t4, t5, t7, t14, t28, t29	t11, t12, t13, t18, t19
18	t6, t8	t11, t15, t17, t20, t21, t24	t2, t11, t26, t27, t30	t1, t6, t8, t9, t10, t14, t16, t22, t25	t3, t4 t5, t7, t21, t28, t29	t12, t13, t17, t18, t19, t23
19	t6, t8, t29, t30, t35	t14, t15, t20, t21, t28, t32	t2, t3, t26, t27, t31	t1, t6, t8, t9, t10, t16, t22, t25	t3, t4 t5, t7, t28, t33, t34	t17, t18, t19, t20, t23
20	t6, t8, t30, t34, t35	t14, t15, t20, t21, t28, t32	t2, t3, t26, t27, t31	t1, t6, t8, t9, t10, t16, t22, t25	t3, t4 t5, t7, t28, t29, t33	t11, t13, t17, t18, t19, t20, t23
21	t5, t6, t8, t29,	t15, t17, t20, t21, t24	t2, t3, t14, t15, t26, t27, t31, t32	t1, t8, t9, t10, t14, t16, t22, t25	t4, t7, t28, t29, t33, t34	t11, t12, t13, t18, t19,

	t30,t35					t23
22	t6,t8,t30, t34,t35	t14,t15,t20, t21,t24	t2,t3,t6,t14,t26, t27,t31,t32	t1,t7,t8,t9,t10, t11,t16,t22,t25	t3,t4,t5,t28,t29,t33	t12,t13,t17, t18,t19,t23
23	t9,t29,t34, t35	t15,t21,t39, t44,t45,t48	t2,t6,t14,t26, t27,t31,t32	t1,t7,t8,t9,t10, t16,t46,t49	t3,t4,t5,t17,t18,t19, t20,t28,t29,t33,t40	t36,t37,t42, t43
24	t5, t6,t8,t29, t30	t17,t20,t21, t24,t38,t39, t41,t44,t45, t48	t2,t3,t11,t14, t15,t17, t26,t27,t30,t31, t32,t35	t1,t9,t10,t16, t22,t25,t40,t46, t49	t3,t4,t7,t28,t33,t34, t47	t19, t23,t36,t37,t 41,t42,t43
25	t5, t6,t8,t29, t30	t17,t20,t21, t24,t38,t39, t41,t44,t45, t48	t2,t3,t11,t14, t15,t17, t26,t27,t30,t31, t32,t35	t1,t9,t10,t16, t22,t25,t40,t46, t49	t3,t4,t7,t28,t33,t34, t47	t19, t23,t36,t37,t 41,t42,t43

Asignación de responsabilidades

Con base en los datos obtenidos a partir de la función objetivo, en la tabla 15 se presenta el concentrado de actividades por clase donde han sido asignados los trabajadores con base en la función objetivo. A partir de la tabla quince se determina que el trabajador t6 puede ser el responsable del grupo de trabajadores que realicen actividades de análisis, considerando que dicho trabajador tiene mayor participación en esta clase; el trabajador t20, a su vez, puede ser el responsable de la clase de diseño; el trabajador t2 responsable de la clase de programación; el trabajador t8 responsable del área de pruebas; el trabajador t3 responsable del área de implementación y el trabajador t23 responsable del área de evaluación.

Tabla 15 Proyectos donde participan los trabajadores

Trabajador	Análisis	Diseño	Programación	Pruebas	Implementación	Evaluación
t1		1		21		
t2			23	3		
t3		6	9		21	
t4		2		7	19	6
t5	4				18	
t6	20	2	3	4		
t7				4	15	
t8	16			18		3
t9				18		5
t10				25		

t11		1	4	5	3	7
t12						9
t13						8
t14		7	6	6	4	2
t15		13	6	6		
t16				17		
t17 ^e		5	2	1		7
t18						10
t19				1		12
t20		15	1			2
t21		10	1	4	2	
t22				15		
t23						10
t24		9			1	1
t25				9		
t26			10			
t27			10			
t28		2			10	
t29	5				7	
t30	8		3			
t31			7			
t32		2	5			
t33					7	
t34	3				3	
t35	4		2			
t36						3
t37						3
t38		2				
t39		3				
t40				3		
t41		1			1	2
t42						3
t43						3
t44 ^e		3				
t45		3				
t46				3		
t47					2	
t48		3				

t49				3		
-----	--	--	--	---	--	--

Las tablas 14 y 15, presentan la forma en que se asignaron las actividades a los trabajadores que participan en los diferentes proyectos. Con esta clasificación, se puede observar el comportamiento de dichos trabajadores de acuerdo a su desempeño dentro de los diferentes proyectos. En estas tablas se puede observar que el trabajador t6 puede ser responsable del área de análisis por su participación en diferentes proyectos dentro de las 25 instancias; por otra parte, para ser responsable de la clase de diseño, se puede considerar al trabajador t20; el trabajador t2, puede ser considerado como responsable de la clase de programación; el trabajador t10 puede ser considerado responsable del área de pruebas; el trabajador t3, responsable del área de implementación; y por último, el trabajador t23, responsable de la clase de evaluación.

CAPÍTULO 5 ANÁLISIS Y DISCUSIÓN

5.1 Contribuciones originales

- Se desarrolló un modelo de optimización combinatoria con base en el paradigma Software Scheduling Problem (PSP) para determinar el costo total y el tiempo total de un proyecto con base en los recursos disponibles y en las características de éstos para dicho proyecto.
- Se desarrolló un modelo para identificar la complejidad de los proyectos con base en los recursos empleados y el impacto socio-económico.
- Modelo de análisis de indicadores para la toma de decisiones y predicción de escenarios en proyectos de desarrollo de software.
- Se minimizó la desviación del cumplimiento de requisitos para el desarrollo de proyectos de software considerando tiempo y alcance lineal.
- Se Optimizó el tiempo de desarrollo de un proyecto de software asignando las actividades a los participantes mediante el número mínimo de errores y el máximo nivel de habilidad por clase de actividad.

5.2 Conclusiones

Las metodologías descritas en el capítulo 2 (Moprosoft, CMMI, ITIL, RUP) se aplican en etapas avanzadas sin considerar las características iniciales del proyecto ó la capacidad instalada de la empresa en cuanto a recursos y partiendo de un supuesto basado en la experiencia del administrador del proyecto.

Con esta propuesta, la determinación de la complejidad de un proyecto de software en etapas tempranas, proporcionará al líder de proyecto información relacionada con la posibilidad de desarrollar el proyecto con base en los recursos iniciales destinados, y proveer una asignación inicial para validar cargas de trabajo (tiempo estimado por actividad, número de errores promedio por actividad y nivel de habilidad en relación a cada clase de actividad), una proyección inicial del costo total estimado a partir de la ponderación en proyectos anteriores y determinar la duración total del proyecto con base en los tiempos estimados por trabajador.

5.3 Trabajos futuros

El enfoque orientado a determinar la complejidad en proyectos de software, se puede aplicar en diferentes tipos de proyectos, partiendo del supuesto que cualquier proyecto tiene 3 características fundamentales: tiempo, costo y alcance.

Por otra parte, es conveniente mencionar que los proyectos en empresas parten de un supuesto económico que diferencia para proyectos de investigación donde los supuestos iniciales varían con respecto al objetivo de dicha investigación. Por ende, se puede considerar como trabajo futuro, orientar la determinación de la complejidad en etapas iniciales a proyectos de investigación con los requerimientos iniciales relativos a este campo.

6. REFERENCIAS

- Abdomerovic, 2001 Abdomerovic Muhamed. Brainstorming The PMBOK® Guide. The complete reference for relating and chronologically sequencing process inputs and outputs. ISBN: 0-9745796-2
- Adeli and Karim, 1997 Adeli, H. and Karim, A. (1997). Scheduling/cost optimization and neural dynamics model for construction. *Journal of Construction Engineering and Management*, 123: 450-458.
- Ahem et al., 2001 Ahem D. H. et al, CMMI Distilled. MA: Addison-Wesley. 2001
- AIMMS Web, 2010 Optimization Software for Mathematical Programming. <http://www.aimms.com/>. Última visita: 04 de Febrero de 2011.
- Alba and Chicano, 2007 Alba, E. and Chicano, J.F. (2007). Software project management with GAs. *Information Sciences*, 177(11):Pages 2380-2401
- Alcaraz and Maroto, 2001 Alcaraz, J. and Maroto, C. (2001). A robust genetic algorithm for resource allocation in project scheduling. *Annals of Operations Research*, 102: 83-109.
- Alcaraz et al., 2003 Alcaraz, J., Maroto, C., and Ruiz, R. (2003). Solving the multi-mode resource-constrained project scheduling problem with genetic algorithms. *Journal of the Operational Research Society*, 54: 614-626.
- Alcaraz et al., 2004 Alcaraz, J., Maroto, C. and Ruiz, R. (2004). Improving the performance of genetic algorithms for RCPS problem. *Proceedings of the Ninth International Workshop on Project Management and Scheduling*, Nancy 2004, 40-43.
- Anderson et al, 2004 Anderson, D. Sweeney D. & Williams, T. (2004) Estadística para administración y economía (8ª. ed.). México: Thomson.
- Artigues and Roubellat, 2001 Artigues, C. and Roubellat, F. (2001). A Petri net model and a general method for on and off-line multi-resource job shop scheduling with setup times. *International Journal of Production Economics*, 74 (1-3): 63-75.
- Atkinson, 1999 Atkinson, R. (1999). Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria. *International Journal of Project Management*, 17(6): 337-342.
- Baar et al., 1998 Baar, T., Brucker, P. and Knust, S. (1998). Tabu-search algorithms and lower bounds for resource-constrained scheduling problem. In: Voss, S., Martello, S., Osman, I., Roucairol, C. (Eds.). *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*. Kluwer Academic, 1-18.
- Baptiste et al., 1999 Baptiste, P., Le Pape, C. and Nuijten, W. (1999). Satisfiability tests and time-bound adjustments for cumulative scheduling problems. *Annals of Operations Research*, 92: 305-333.
- Bellenguez and Néron, 2005 Bellenguez, O. and Néron, E. (2005). Lower bounds for the multi-skill project scheduling problem with hierarchical level of skills. In: Burke, E.K., Trick, M. (Eds.), *Lecture Notes in Computer Science* (3616): *Practice and Theory of Automated Timetabling V*. Springer, Berlin, 229-243.
- Bellenguez-Morineau, 2008 Bellenguez-Morineau, O. (2008). Methods to solve multi-skill project scheduling problem. *4OR: A Quarterly Journal of Operations Research*, 6 (1): 85-88.
- Bianco et al., 1998 Bianco, L., Dell'Olmo, P. and Speranza, M.G. (1998). Heuristics for multi-mode scheduling problems with dedicated resources. *European Journal of*

- Operational Research*, 107 (2): 260-271.
- Bouleimen and Lecocq, 2003 Bouleimen, K. and Lecocq, H. (2003). A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode version. *European Journal of Operational Research*, 149: 268-281.
- Blazewicz et al., 1983 Blazewicz, J., Lenstra, J.K. and Rinnooy Kan, A.H.G. (1983). Scheduling subject to resource constraints: Classification and complexity. *Discrete Applied Mathematics*, 5 (1): 11-24.
- Blazewicz et al., 1986 Blazewicz, J., Drabowski, M. and Weglarz, J. (1986). Scheduling multiprocessor tasks to minimize schedule length. *IEEE Transactions on Computers*, 35:389-393
- Boctor, 1996 Boctor, F.F. (1996). Resource-constrained project scheduling by simulated annealing. *International Journal in Production Research*, 34: 2335-2351.
- Boctor, 1996a Boctor, F. (1996a). An adaption of the simulated annealing for solving resource-constrained project scheduling problems. *International Journal of Production Research*, 34: 2335-2351.
- Boer, 1998 De Boer, R. (1998). Resource-constrained multi-project management, a hierarchical decision support system. Ph.D. Thesis, *University of Twente*.
- Böttcher et al., 1999 Böttcher, J., Drexl, A., Kolisch, R. and Salewski, F. (1999). Project scheduling under partially renewable resource constraints. *Management Science*, 45 (4): 543-559.
- Bouleimen and Lecocq, 2003^{*} Bouleimen, M. and Lecocq, H. (2003). A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple version. *European Journal of Operational Research*, 149: 268-281.
- Brucker et al., 2001 Brucker, P. and Knust, S. (2001). Resource-constrained project scheduling and timetabling. In: Bruke, E., Erben, W. (Eds.), *Practice and Theory of Automated Timetabling III (PATAT III)*. *Lecture Notes in Computer Science*, 2079: 277-293.
- Brucker et al., 1999a Brucker, P. and Schumacher, D. (1999). A new tabu search procedure for audit-scheduling problem. *Journal of Scheduling*, 2: 157-173.
- Brucker et al., 1999 Brucker, P., Drexl, A., Mohring, R. and Neumann, K. (1999). Resource-constrained project scheduling: Notation, classification, models and methods. *European Journal of Operational Research*, 112: 3-41.
- Budd and Harris, 2009 Budd Leslie and Harris Lisa. Routledge e-bussiness. Managing or Governing? Edit: Taylor and Francis Group. First published 2009. ISBN: 0-203-88388-8.
- Chang et al., 2008 Chang, C.K., Jiang, H-y., Di, Y., Zhu, D. and Ge, Y.(2008). Time-line based model for software project scheduling with genetic algorithms. *Information and Software Technology*, 50(11): 1142-1154.
- Chen and Shahandashtia, 2009 Chen, P-H. and Shahandashtia, S.M.(2009). Hybrid of genetic algorithm and simulated annealing for multiple project scheduling with multiple resource constraints. *Automation in Construction*, 18(4): 434-443.
- Cho and Kim, 1997 Cho, J.-H. and Kim, Y.-D. (1997). A simulated annealing algorithm for resource-constrained project scheduling problems. *Journal of the Operational Research Society*, 48: 735-744.
- Cleland and Ireland, 2007) Cleland David I. and Ireland Lewis R. Project Management. Strategic design and implementation. Fifth edition. ISBN: 0-07-147160-X. 2007. Edit: McGraw-Hill.
- CMMI, 2002 Capability Maturity Model Integration (CMMI) Versión 1.1 (2002) CMMI for Software Development (CMMI-SW 1.1) Software Engineering Institute

- CMU/SEI 2002-TR-28.
- Cobit, 1998 IT Governance Institute. IT Assurance guide. Using Cobit. ISBN: 1-933284-74-9. Printed in the United States of America.
- Dayanand and Padman, 1993 Dayanand, N. and Padman, R. (1993). Payments in projects: A contractor's model. Working Paper 93-71, The Heinz School, *Carnegie Mellon University*, Pittsburgh, PA
- Dayanand and Padman, 1995 Dayanand, N. and Padman, R. (1995). Project contracts and payment schedules: the client's problem. Working Paper 95-23, The Heinz School, *Carnegie Mellon University*, Pittsburgh, PA
- Dayanand and Padman, 1997 Dayanand, N. and Padman, R. (1997). On modelling payments in projects. *Journal of the Operational Research Society*, 48: 906-918.
- De P. et al., 1995 De, P., James Dunne, E., Ghosh, J.B. and Wells, C.E. (1995). The discrete time-cost tradeoff problem revisited. *European Journal of Operational Research*, 81(2): 225-238
- De Reyck et al., 1998 De Reyck, B., Demeulemeester, E. and Herroelen, W. (1998). Local search methods for the discrete time/resource trade-off problem in project networks. *Naval Research Logistic Quarterly*, 45: 553-578.
- Demeulemeester and Herroelen, 1992 Demeulemeester, E. and Herroelen, W. (1992). A branch-and-bound procedure for multiple resource-constrained project scheduling problem. *Management Science*, 38(12): 1803-1818.
- Demeulemeester et al., 2000 Demeulemeester, E., De Reyck, B. and Herroelen, W. (2000). The discrete time/resource trade-off problem in project networks: a branch and bound approach. *IIE Transaction*, 32(11): 1059-1069.
- Demeulemeester and Herroelen, 2002 Demeulemeester, E. and Herroelen, W. (2002). *Project Scheduling: A Research Handbook*. Kluwer Academic Publishers, Boston.
- Demeulemeester, 2002 Demeulemeester, E. Project Scheduling. A research handbook. Demeulemeester, Erick L. and Herroelen, Willy S. Kluwer's International Series. ISBN 1-40207-051-9. 2002
- Doersch and Patterson, 1977 Doersch, E.W. and Patterson, J.H. (1977). Scheduling a project to maximize its present value: A zero-one programming approach. *Management Science*, 23 (8): 882-889.
- Drezet, 2007 Drezet, L.E. and Billaut, J.C. (2008). A project scheduling problem with labor constraints and time-dependent activities requirements. *International Journal of Production Economics*, 112(1) 217-225.
- Duin and Van Der Sluis, 2006 Duin, C.W. and Van Der Sluis, E. (2006). On the complexity of adjacent resource scheduling. *Journal of Scheduling*, 9 (1): 49-62.
- Eshtehardian et al., 2009 Eshtehardian, E., Afsharb, A. and Abbasnias, R. (2009). Fuzzy-based MOGA approach to stochastic time-cost trade-off problem. *Automation in Construction*, 18(5): 692-701
- Escap, 2010 What is good governance?. Economic and Social Commission for Asia and de Pacific. Última visita: 05 de Octubre de 2010.
<http://www.unescap.org/pdd/prs/ProjectActivities/Ongoing/gg/governance.asp>
- Escap2, 2010 About Governance. Economic and Social Commission for Asia and de Pacific. Última visita: 05 de Octubre de 2010.
<http://www.cegd.eu/Pages/AboutGovernance.aspx>
- Fuentes-Penna et al, 2010 Alejandro Fuentes-Penna, Ocotlán Díaz-Parra, José C. Zavala-Díaz, Jorge A. Ruiz-Vanoye, Juan C. Olivares-Rojas: Guideline of Identification and Track on Support's level on Mexican Very Small Enterprises (MVSE). International

- Journal of Combinatorial Optimization Problems and Informatics, Vol. 1, No. 1, pp. 50-55, May-Aug (2010).
- Garson, 2006 Garson David. Public Information Technology and E-governance: Managing the virtual state. ISBN 0-7637-3468-3. Edit: Jones and Bartlett Publishers Canada.
- Ghazanfari, 2007 Ghazanfari, M., Shahanaghi, K. and Yousefli, A. (2008). An Application of Possibility Goal Programming to the Time-Cost Trade off Problem. Department of Industrial Engineering, Iran University of Science and Technology, Tehram, Iran. *Journal of Uncertain Systems*, 2 (1): 22-30.
- Gonsalves and Itoh, 2010 Gonsalves, T. and Itoh, K. (2010). Multi-Objective Optimization for Software Development Projects. *International MultiConference of Engineers and Computer Scientists 2010, IMECS 2010*, Vol. 1, March 17-19, Hong Kong.
- Grinold, 1972 Grinold, R.C. (1972). The payment scheduling problem. *Naval Research Logistics*, 19.
- Gutiérrez, 2007 Gutiérrez Franc, E., La Torre-Zurita, F. and Mejía-Delgadillo, G. (2007). A genetic algorithm for the resource constrained project scheduling problem (RCPSP). *19th International Conference On Production Research – ICPR 19*. July 29 - August 2, 2007, Valparaíso, Chile
- Hapke et al., 1994 Hapke, M., Jaszkiwicz, A. and Slowinska, R. (1994). Fuzzy project scheduling system for software development. *Fuzzy Sets and Systems*, 67(1): 101-117.
- Hartmann and Kolisch 2000 Hartmann, S. and Kolisch, R. (2000). Experimental evaluation of state-of-the-art heuristics for the resource-constrained project scheduling problem. *European Journal of Operational Research*, 127: 394-407.
- Hartmann, 1998 Hartmann, S. (1998). A competitive genetic algorithm for resource-constrained project scheduling. *Naval Research Logistics*, 456: 733-750.
- Hartmann, 2001 Hartmann, S. (2001). Project scheduling with multiple modes: A genetic algorithm. *Annals of Operations Research*, 102: 111-135.
- He and Xua, 2008 He, Z. and Xua, Y. (2008). Multi-mode project payment scheduling problem (MPPSP) with bonus-penalty structure. *European Journal of Operational Research*, 189(3): 1191-1207.
- Heerink, 2010 Heering, Koos. AIMMS Tutorial. Marzo 2010. Edit: Paragon Decision Technology B. V. All rights reserved.
- Hegazy, 1999 Hegazy, T. (1999). Optimization of construction time–cost trade-off analysis: using genetic algorithms. *Canadian Journal of Civil Engineering*, 26(6):685–697.
- Hernández, 2006 Hernández-López, Leonor. Predicción y optimización de emisores y consumo mediante redes neuronales en motores diesel. Universidad Politécnica de Valencia. 2006. ISBN-10: 84-291-4708-X. ISBN-13: 978-84-291-4708-7.
- Hernández, 2006 Hernández Sampieri, R., Fernández Collado, C. y Baptista, P. (2006). Metodología de la investigación (4ª ed.). México: Mc Graw-Hill.
- Herroelen et al., 1997 Herroelen, W.S., Dommelen, P. and Demeulemeester, E.L. (1997). Project network models with discounted cash flows: A guided tour through recent developments. *European Journal of Operational Research*, 100: 97-121.
- Herroelen et al., 1998 Herroelen, W., Reyck, B. and Demeulemeester, E. (1998). Resource-constrained project scheduling: A survey of recent developments. *Computers and Operations Research*, 4: 279-302.
- Hindi et al., 2002 Hindi, K.S., Yang, H. and Fleszar, K. (2002). An evolutionary algorithm for resource-constrained project scheduling. *IEEE Transactions on Evolutionary*

- Computation*, 6: 512-518.
- Huang, 2009 Huang, W., Ding, L., Wen, B. and Cao, B. (2009). Project Scheduling Problem for Software Development with Random Fuzzy Activity Duration Times. *Advances in Neural Networks – ISNN 2009*, 5552: 60-69.
- Isaac, 1995 Isaac, Stephen y Michael, William (1995). Handbook in Research and Evaluation, for education and the Behavioral Sciences (3er. ed.), EUA. American Psychological Association. (n. d.). Electronic references. Retrieved June 12, 2002, from <http://www.apastyle.org/elecsource.html>
- ISACA, 2010 COBIT the comprehensive IT governance framework that addresses every aspect of IT and integrates all of the main global IT standards. ISACA – Information Systems Audit and Control Association.
- Itera, 2010 Itera IT & business process. <http://www.iteraprocess.com/>. Última visita: 27 de septiembre de 2010.
- ITIL, 2005 Introduction to ITIL. The key to Managing IT Services. ISBN: 0 11 330017 5. 2005. Published by TSO (The Stationery Office).
- Izar, 1996 Izar Landeta Juan M. Fundamentos de investigación de operaciones para administración. 19 de julio de 1996. Editorial: Universitaria potosina. ISBN: 968-7674-01-6.
- James et al., 1990 James M. Antill and Ronald W. Woodhead. Path methods in construction practice. Fourth edition. 1990. Copyright by John Wiley & Sons, Inc. ISBN 0-471-62057-2
- Jarboui et al., 2008 Jarboui, B., Damak, N., Siarry, P. and Rebai, A. (2008). A combinatorial particle swarm optimization for solving multi-mode resource-constrained project scheduling problems. *Applied Mathematics and Computation*, 195: 299-308.
- Jozefowska et al., 2001 Jozefowska, J., Mika, M., Rozycki, R., Waligora, G. and Weglarz, J. (2001). Simulated annealing for multi-mode resource-constrained project scheduling. *Annals of Operations Research*, 102: 137-155.
- Julin, 2001 Julin, H. and Zhong-Ping, W. and Guochun, T. (2004). Construction project scheduling problem with uncertain resource constraints. Research Report School of Mathematics and Statistics, *Wuhan University*.
- Kask, 2009 Kask, Oliver and Asbjørn Eide. Draft report on the notions of “good governance” and “good administration”. CDL 2009. Venice Commission. Council of Europe. Strasbourg, 21 September 2009. Study no. 470 / 2008.
- Kelley et al., 1961 Kelley, J.E. (1961). Critical-Path Planning and Scheduling: Mathematical Basis. *Operations Research*, 9(3): 296-320.
- Kelly, 1963 Kelly, J.E. (1963). *The Critical-path Method: Resources Planning and Scheduling*. Prentice-Hall, Englewood Cliffs, N.J.
- Kimms, 2001⁶ Kimms, A. (2001). *Mathematical Programming and Financial Objectives for Scheduling Projects*, Publisher: Kluwer, Boston.
- Kohlmorgen et al., 1999 Kohlmorgen, U., Schmeck, H. and Haase, K. (1999). Experiences with fine-grained parallel genetic algorithms. *Annals of Operations Research*, 90: 203-219.
- Kolisch and Drexl, 1997 Kolisch, R. and Drexl, A. (1997). Local Search for Nonpreemptive Multi-Mode Resource-Constrained Project Scheduling. *IIE Transactions*, 29(11): 987-999.
- Kolisch and Hartmann, 1999 Kolisch, R. and Hartmann, S. (1999). Heuristic algorithms for solving resource-constrained project scheduling problem: Classification and computation analysis. In: Weglarz, J. (Ed.), *Project Scheduling: Recent*

- Models, Algorithms and Applications*. Kluwer Academic Publisher, Boston, 147-178.
- Kolisch and Hartmann, 2006 Kolisch, R. and Hartmann, S. (2006). Experimental investigation of heuristics for resource-constrained project scheduling: An update. *European Journal of Operational Research*, 174: 23-37.
- Kolisch and Padman, 2001 Kolisch, R. and Padman, R. (2001). An integrated survey of deterministic project scheduling. *OMEGA International Journal of Management Science*, 29 (3): 249-272.
- Kroll, 2003 Kroll Per and Kruchten Philippe. The Rational Unified Process Made Easy. Editorial: Addison-Wesley. 2003. ISBN 0-321-16609-4
- Le et al., 2009 Le, H.Q. and Rüppel, U. (2009). Multi-site Construction Project Scheduling considering resource moving time in developing countries. *18th International Conference on the Application of Computer Science and Mathematics in Architecture and Civil Engineering*. K. Gürlebeck and C. Könke (eds.), Weimar, Germany.
- Leu et al., 2001 Leu, S-S., Chen, A-T. and Yang, C-H. (2001). A genetic algorithm-based fuzzy optimal model for construction. *International Journal of Project Management*, 19: 47-58.
- Leu et al, 2001a Leu, S.S., Chen, A.T. and Yang, C.H. (2001). A GA-based fuzzy optimal model for construction time–cost trade-off, *International Journal of Project Management*, 19(1): 47-58.
- Liu et al., 2005 Liu, T., Liu, M., Zhang, Y.-B. and Zhang, L.(2005). Hybrid genetic algorithm based on synthetical level of resource conflict for complex construction project scheduling problem. *International Conference on Machine Learning and Cybernetics*, 9: 5699 - 5703.
- Lova et al., 2009 Lova, A., Tormos, P., Cervantes, M. and Barber, F. (2009). An efficient hybrid genetic algorithm for scheduling projects with resource constraints and multiple execution modes. *International Journal of Production Economics*, 117(2): 302-316.
- Martín, 2004 Martín García, Elena y Valeira Reina Gerardo. Sistemas evolutivos y selección de indicadores. Universidad de Sevilla. Google books. Impreso en España. 2004. Secretariado de publicaciones.
- Mika et al., 2006 Mika, M., Waligóra, G. and Weglarz, J. (2006). Modelling setup times in project scheduling. In: Józefowska, J., Weglarz, J. (Eds.), *Perspectives in Modern Project Scheduling*. Springer, Berlin, 131-163.
- Mika et al., 2008 Mika, M., Waligóra, G. and Węglarz, J. (2008). Tabu search for multi-mode resource-constrained project scheduling with schedule-dependent setup times. *European Journal of Operational Research*, 187(3): 1238-1250.
- Mendes et al., 2009 Mendes, J.J.M., Goncalves, J.F. and Resende, M.G.C. (2009). A random key based genetic algorithm for the resource constrained project scheduling problem. *Computers and Operations Research*, 36(1): 92-109.
- Mingozi et al., 1998 Mingozi, A., Maniezzo, V., Ricciardelli, S. and Bianco, L. (1998). An exact algorithm for project scheduling with resource constraints based on a new mathematical formulation. *Management Science*, 44: 714-729.
- Möhring, 2002 Möhring, R.H., Schulz, A.S., Stork, F. and Uetz, M. (2002). Solving Project Scheduling Problems by Minimum cut computations. Working paper 4231-02. MIT Sloan School of Management.
- Moprosoft, 2006 Oktaba, Hanna. Moprosoft. Software Industry Process Model Version 1.3.2. April, 2006. All rights reserved Ministry of Economy México.

- Moprosoft, 2010 Comunidad Moprosoft – Modelo de procesos para la industria del Software. <http://www.comunidadmoprosoft.org.mx/>. Última visita: 27 de Septiembre de 2010.
- Mori and Tseng, 1997 Mori, M., Tseng, C. (1997). A genetic algorithm for multi-mode resource constrained project scheduling problem. *European Journal of Operational Research*, 100: 134-141.
- Néron, 2002 Néron, E. (2002). Lower bounds for the multi-skill project scheduling problem. *Proceedings of the Eighth International Workshop on Project Management and Scheduling (PMS2002)*, Valencia, Spain, 274-277.
- Neumann et al., 2002 Neumann, K., Schwindt, C. and Zimmermann, J. (2002). *Project Scheduling with Time Windows and Scarce Resources: Temporal and Resource-Constrained Project Scheduling with Regular and Nonregular Objective Functions*. Springer, Berlin.
- Neumann et al., 2006 Neumann, K., Schwindt, C. and Zimmermann, J. (2006). Resource-constrained project scheduling with time windows. Józefowska, J., Weglarz, J. (Eds.), *Perspectives in Modern Project Scheduling*. Springer, Berlin, 375-407.
- Newell, 2010 Michael W. Newell, Preparing for the Project Management Professional Certification Exam. Project Management Professional – PMP. Third edition.
- Nonobe and Ibaraki, 2002 Nonobe, K. and Ibaraki, T. (2002). Formulation and tabu search algorithm for the resource constrained project scheduling problem. Ribeiro, C.C., Hansen, P. (Eds.), *Essays and Surveys in Metaheuristics*. Kluwer Academic Publishers, 557-588.
- Okot, 2000 Rogers W'O Okot-Uma. Electronic governance: Re-inventing Good Governance. Commonwealth Secretariat London. 2000.
- Oktaba et al., 2005 Oktaba Hanna, Alquicira-Esquivel Claudia, Su-Ramos Angélica, Martínez-Martínez Alfonso, Quintanilla-Osorio Gloria, Ruvalcaba-López Mara, López-Lira-Hinojo Francisco, Rivera-López María E., Orozco-Mendoza María J., Fernández-Ordóñez Yolanda and Flores Lemus Miguel A. Modelo de Procesos para la industria del software por niveles de capacidad de procesos versión 1.3. Agosto 2005. Secretaría de Economía – México.
- Ortega and Sánchez, 2007 Ortega Mier, Miguel and García, Sánchez Álvaro. AIMMS Tutorial. Noviembre de 2007. Universidad Politécnica de Madrid. ISBN: 978-84-612-1155-5.
- Özdamar, 1999 Özdamar, L. (1999). A genetic algorithm approach to a general category project scheduling problem. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 29: 44-59.
- PA, 2010 Partnership Agreement between the States of the African, Caribbean and Pacific group of States on the one part, and the European Community and its member States on the another part.
- Pacheco, 2006 Pacheco, A. y Cruz, M. A. (2006). Metodología crítica de la investigación. México: CECSA.
- Panganiban, 2004 Panganiban, Rik. E-democracy and the United Nations: Using Information Communications Technologies to Increase Access to Information and Participation within the UN System. Center for United Nations Reform Education. A contribution to the World Summit on the Information Society by the Center for United Nations Reform Education.
- Patterson et al., 1989 Patterson, J.H., Slowinski, R., Talbot, F.B. and Weglarz, J. (1989). Chapter 1: An algorithm for a general class of precedence and resource constrained

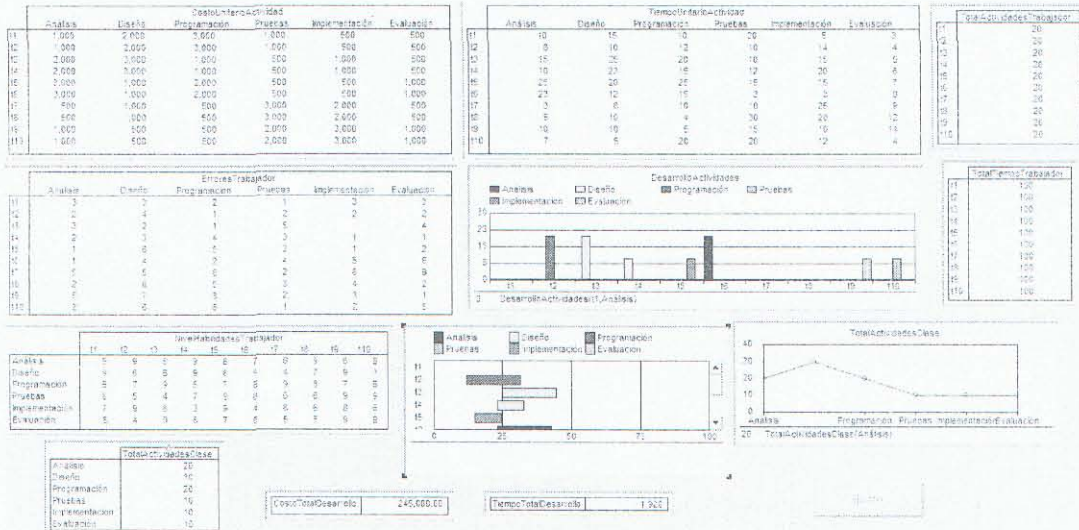
- scheduling problems. Sowinski, R., Weglarz, J. (Eds.), *Advances in project scheduling*. Elsevier, Amsterdam.
- Paulk, 1993 Paulk, M. C.; Curtis, B.; Chrissis, M. B. and Bush, M. Key Practices of the Capability Maturity Model, Version 1.1 Software Engineering Institute CMU/SEI-93-TR-25.
- Paulus and Hurink, 2006 Paulus, J.J. and Hurink, J. (2006). Adjacent-resource scheduling: why spatial resources are so hard to incorporate. *Electronic Notes in Discrete Mathematics*, 25: 113-116.
- Pinson et al., 1994 Pinson, E., Prins, C. and Rullier, F. (1994). Using tabu search for solving the resource-constrained project scheduling problem. *Proceedings of the Fourth International Workshop on Project Management and Scheduling*, Leuven, Belgium, 102-106.
- PMI, 2003 The PMI® Compendium of Project Management Practices. Author: Project Management Institute. All rights reserved.
- Pritsker et al, 1969 Pritsker, A.A.B., Watters, L.J. and Wolfe, P.M. (1969). Multiproject scheduling with limited resources: A zero-one programming approach. *Management Science*, 16: 93-107.
- Ranjbar and Kianfar, 2007 Ranjbar, M.R. and Kianfar, F. (2007). Solving the discrete time/resource trade-off problem in project scheduling with genetic algorithms. *Applied Mathematics and Computation*, 191(2): 451-456.
- Ranjbar et al., 2008 Ranjbar, M., De Reyck, B. and Kianfar, F. (2008). A hybrid scatter-search for the discrete time/resource trade-off problem in project scheduling. *European Journal of Operational Research* 193 (1), 35-48.
- Rojas, 2005 Rojas Soriano, R. (2005). Guía para realizar investigaciones sociales (40ª. ed.). México: Plaza y Valdés.
- Russell, 1970 Russell, A.H. (1970). Cash Flows in Networks. *Management Science*, 6(5): 357-372.
- Ruiz-Vanoye et al, 2010 Jorge A. Ruiz-Vanoye, Ocotlán Díaz-Parra, José C. Zavala-Díaz, Alejandro Fuentes-Penna, Juan C. Olivares-Rojas. A survey of Project Scheduling Problems (PSP). Proceedings of the 15th Annual International Conference on Industrial Engineering Theory, Applications and Practice, pp. 460-472, Mexico City, Mexico October 17-20, (2010) ISBN 97809652558-6-8.
- Ruiz-Vanoye et al, 2010 (2) Jorge A. Ruiz-Vanoye, Ocotlán Díaz-Parra, José C. Zavala-Díaz, Alejandro Fuentes-Penna, Juan C. Olivares-Rojas: Models, resources and activities of Project Scheduling Problems. International journal of industrial engineering-theory applications and practice. Vol. x, N. y, pp.xy-yx (2011). ISSN: 1943-670X.
- Ruiz-Vanoye et al, 2011 Jorge A. Ruiz-Vanoye, Sebastian Pathiyamattom-Joseph, Miguel A. Fernández-Medina, Alejandro Fuentes-Penna, Ocotlán Díaz-Parra. Project Scheduling Problem for Software Development Library - PSPSWDLIB. Revista de Ciências da Computação, No. 5. Accepted
- Ruiz-Vanoye et al, 2011 (2) Jorge A. Ruiz-Vanoye, Ocotlán Díaz-Parra. An Overview of the Theory of Instances Computational Complexity. International Journal of Combinatorial Optimization Problems and Informatics, Vol. 2, No. 2. pp. 37-43, May-Aug (2011).
[http://ijcopi.org/ojs/index.php?journal=ijcopi&page=article&op=view&path\[\]=60](http://ijcopi.org/ojs/index.php?journal=ijcopi&page=article&op=view&path[]=60)
- Schaerf, 1999a Schaerf, A. (1999a). A survey of automated timetabling. *Artificial Intelligence Review*, 13: 87-127.

- Schaerf, 1999b Schaerf, A. (1999b). Local search techniques for large high school timetabling problems. *IEEE Transactions on Systems, Man and Cybernetics, Part A: systems and human*, 29(4): 368-377.
- Schwindt and Trautmann, 2003 Schwindt, C. and Trautmann, N. (2003). Scheduling the production of rolling ingots: industrial context, model, and solution method. *International Transactions in Operational Research*, 10 (6): 547-563.
- Shewchuk and Chang, 1995 Shewchuk, J.P. and Chang, T.C. (1995). Resource-constrained job scheduling with recyclable resources. *European Journal of Operational Research*, 81(2): 364-375.
- Shue, 2008 Shue, L-Y. (2008). A generalized heuristic learning approach to Project Scheduling Problems with Resource Constraints. *Journal of the Chinese Institute of Industrial Engineers*, 25(3): 204-214.
- Schwalbe, 2009 Schwalbe, Kathy. Information Technology. Project Management 6e. Course technology. CENGAGE Learning. 2009. ISBN-13 978-0-324-78692-7. ISBN-10 0-324-78692-.
- Siemens, 1971 Siemens, N. (1971). A simple CPM Time-Cost Tradeoff Algorithm. *Management Science*, 17(6): 354-363.
- Slowinski, 1980 Slowinski, R. (1980). Two approaches to problems of resource allocation among project activities – a comparative study. *Journal of the Operational Research Society*, 31(8): 711-723.
- Slowinski et al., 1994 Slowinski, R., Soniewicki, B. and Weglarz, J. (1994). DSS for multiobjective project scheduling. *European Journal of Operational Research*, 79: 220-229.
- Stackpole, 2010 Stackpole Cynthia Snyder. A user's manual to the PMBOK® guide. Edit: Wiley. Copyright 2010 by John Wiley and Sons, Inc. ISBN: 978-0-470-58489-7.
- Stuttgart, 2001 Stuttgart, Michael Reiss. E-business: basics and challenges. Photogrammetric Week '01. Dr. Fritsch & R. Spiller, Eds.
- Talbot, 1982 Talbot, F.B. (1982). Resource-constrained project scheduling with time-resource trade-offs: the nonpreemptive case. *Management Science*, 28(10): 1197-1210.
- Tchao and Martins, 2008 Tchao, C. and Martins, S.L. (2008). Hybrid Heuristics for Multi-mode Resource-Constrained Project Scheduling. Learning and Intelligent Optimization. *Lecture Notes in Computer Science*, 5313:234-242.
- Tiwari et al., 2009 Tiwari, V., Patterson, J.H. and Mabert, V.A. (2009). Scheduling projects with heterogeneous resources to meet time and quality objectives. *European Journal of Operational Research*, 193(3): 780-790.
- Tormos et al., 2001 Tormos, P. and Lova, A. (2001). A Competitive Heuristic Solution Technique for Resource-Constrained Project Scheduling. *Annals of Operations Research*, 102: 65-81.
- Turner and Simister, 2000 Turner John Rodney and Simister Stephen J. Handbook of project management. Edit: Gower Publishing, 2000. Third edition. ISBN 0 566 08128 5.
- Tuya et al, 2007 Tuya Javier, Ramos-Román Isabel y Dolado-Cosín Javier. Técnicas cuantitativas para la gestión de la ingeniería de software. ISBN: 978-84-9745-204-5. Editorial: Netbiblio.
- Ulosoy, 2001 Ulosoy, G. (2001). Four payment models for the multi-mode resource constrained project scheduling problem with discounted cash flows. *Annals of Operations Research*, 102: 237-261.
- Valls et al., 2003 Valls, V., Ballestín, F. and Quintanilla, M.S. (2003). A hybrid genetic

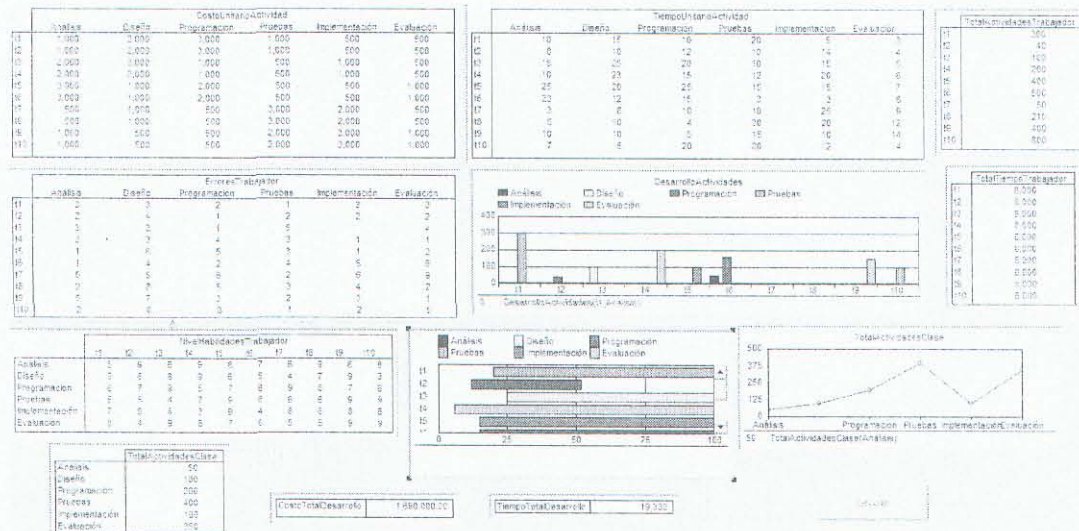
- algorithm for the RCPSP. Technical Report, Department of Statistics and Operations Research, *University of Valencia*.
- Valls et al., 2008 Valls, V., Ballestín, F. and Quintanilla, M.S. (2008). A hybrid genetic algorithm for the resource-constrained project scheduling problem. *European Journal of Operational Research*, 185: 495-50.
- Van-Bon et al, 2008 Van-Bon Jan, De-Jong Arjen, Kolthof Axel, Pieper Mike, Tjassing Ruby, Van-der-Veen Annelies and Verheijen Tienieke. Gestión de Servicios de TI basada en ITIL® V3. Guía de Bolsillo. ITSM Library. ISBN: 978 90 8753 106 5. Primera edición, primera impresión, Diciembre 2008. Van Haren Publishing.
- Van-Bon et al 2, 2008 Van-Bon Jan, De-Jong Arjen, Kolthof Axel, Pieper Mike, Tjassing Ruby, Van-der-Veen Annelies and Verheijen Tienieke. Fundamentos de la gestión de servicios de TI en ITIL. ITSM Library. ISBN: 978 90 8753 060 0. 3ª edición, primera reimpresión, enero 2008.
- Van Peteghem and Vanhoucke, 2010 Van Peteghem, V. and Vanhoucke, M. (2010). A genetic algorithm for the preemptive and non-preemptive multi-mode resource-constrained project scheduling problem. *European Journal of Operational Research*, 201(2): 409-418.
- Wang and Huang, 2010 Wang, X. and Huang, W. (2010). Fuzzy resource-constrained project scheduling problem for software development. *Wuhan University Journal of Natural Sciences*, 15 (1): 25-30.
- Weglarz, 1980 Weglarz, J. (1980). On certain models of resources allocation problems. *Kybernetes*, 9(1), 61-66.
- Zhang et al., 2006 Zhang, H., Tam, C. and Li, H. (2006). Multimode project scheduling based on particle swarm optimization. *Computer-Aided Civil and Infrastructure Engineering*, 21: 93-103.

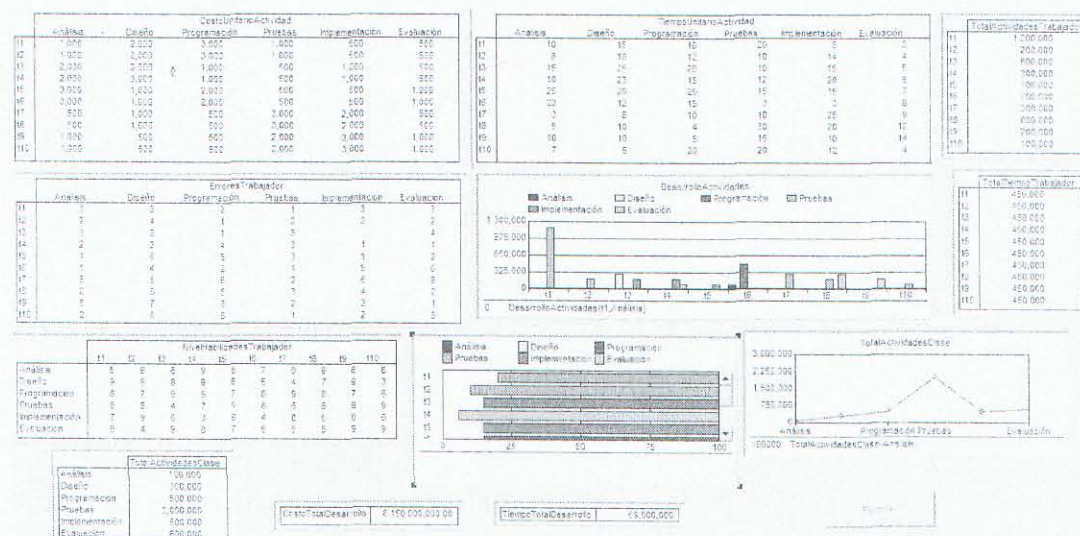
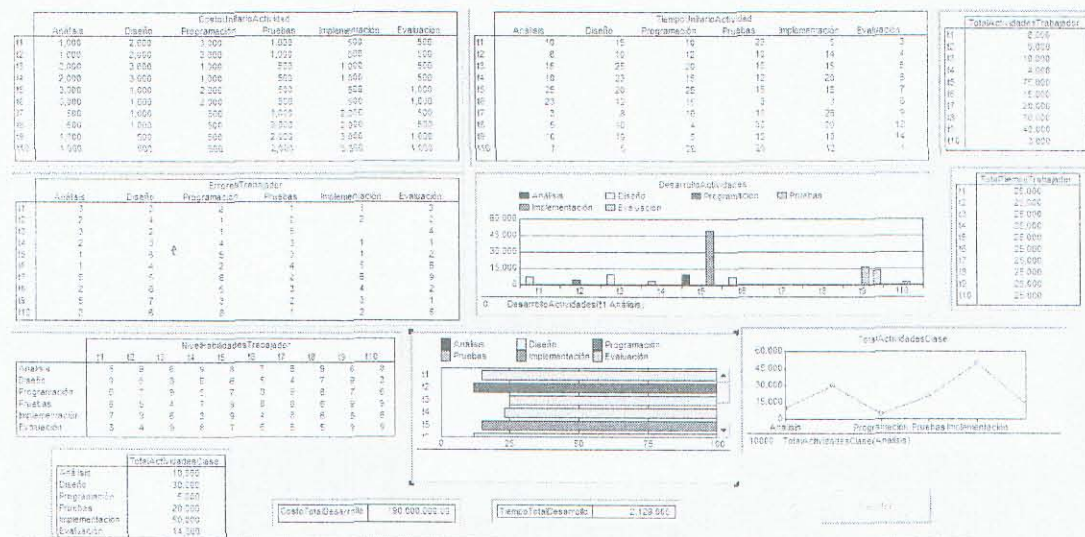
ANEXOS

ANEXO – INSTANCIA 1



ANEXO – INSTANCIA 2





ANEXO – INSTANCIA 5

	Cuentas por Cobrar					
	A-Activos	Pasivos	Programación	Prestos	Programación	Evaluación
1	1.000	2.000	3.000	1.000	500	500
2	1.000	2.000	3.000	1.000	500	500
3	2.000	3.000	4.000	1.000	1.000	500
4	2.000	3.000	4.000	1.000	1.000	500
5	2.000	1.000	2.000	500	500	1.000
6	3.000	1.000	2.000	500	500	1.000
7	500	1.000	500	3.000	2.000	500
8	500	1.000	500	3.000	2.000	500
9	1.000	500	500	2.000	3.000	1.000
10	1.000	500	500	2.000	3.000	1.000

	Terdapat Interaksi/No					
	Analisis	Dasar	Programasi	Pesutan	Implementasi	Evaluasi
12	10	10	10	20	5	7
13	8	10	12	10	14	4
14	15	10	20	10	11	5
15	20	15	15	10	20	5
16	25	20	20	15	10	7
17	23	12	15	5	7	8
18	5	8	16	10	28	6
19	8	10	4	10	10	10
20	10	10	5	15	10	14
21	7	5	20	10	10	4

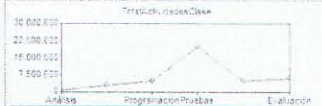
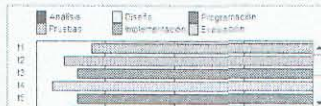
	Total stages tested for
11	12,000,000
12	2,000,000
13	5,000,000
14	3,000,000
15	1,000,000
16	7,000,000
17	3,000,000
18	5,000,000
19	2,000,000
110	1,000,000

	Etiopatogenesi				
	Analisis	Diagnos	Programa	Pelaksanaan	Evaluasi
1.1	1	3	1	2	3
1.2	2	4	1	2	2
1.3	2	2	1	2	4
1.4	1	5	1	1	1
1.5	1	4	2	3	6
1.6	1	5	2	4	5
1.7	3	3	1	3	2
1.8	3	1	3	3	1
1.9	3	6	3	1	2



	Total Temporal Cost (s)
12	4 500.000
13	3 500.000
14	4 375.000
15	4 500.000
16	4 500.000
17	4 500.000
18	4 500.000
19	4 500.000
20	2 750.000

	Nivel de Capacidade Tecnológica									
	1	2	3	4	5	6	7	8	9	10
Atividade	6	12	18	24	30	36	42	48	54	60
Diária	6	12	18	24	30	36	42	48	54	60
Programação	6	7	8	9	10	11	12	13	14	15
Prática	2	4	6	8	10	12	14	16	18	20
Implementação	7	9	11	13	15	17	19	21	23	25
Evaluation	5	6	7	8	9	10	11	12	13	14



	Total Charges to CSAs
Analysis	1,000,000
Design	2,000,000
Programming	3,000,000
Release	20,000,000
Implementation	5,000,000
Evaluation	5,000,000

Costo total Desempeño	60.500.000.000,00
-----------------------	-------------------

TiempoTotalDesarrollo	857.000.000
-----------------------	-------------

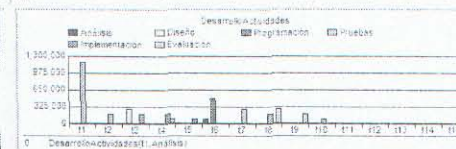
ANEXO – INSTANCIA 6

	Ambas	Doble	Cuadrado	Cuadrilíneo	Atividade	Ampliação	Extensão
11	1.000	2.000	2.000	1.000	500	500	
12	1.000	2.000	2.000	1.000	500	500	
13	2.000	2.000	1.700	500	1.000	200	
14	2.000	2.000	1.600	1.000	1.000	0	
15	2.000	1.000	2.000	500	500	1.000	
16	3.000	1.000	2.000	500	500	1.000	
17	500	1.000	1.000	2.000	2.000	500	
18	500	1.000	500	1.000	2.000	500	
19	500	500	500	2.000	1.000	1.000	
20	500	500	500	500	500	500	

	Temas de las actividades				
	Grupos	Procesamiento	Algebra	Implementación	Evaluación
11	10	15	15	20	5
12	8	10	12	10	14
13	15	25	25	15	15
14	10	25	15	5	20
15	20	30	27	15	15
16	23	12	15	3	7
17	5	6	15	10	25
18	5	10	4	30	20
19	15	10	5	15	15
20	5	5	25	12	15

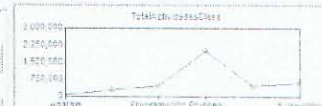
	Total Goodness (points)
11	120,000
12	200,000
13	500,000
14	800,000
15	120,000
16	700,000
17	500,000
18	800,000
19	200,000

	Analysis	Exact	Regression	Partial	Implementable	Extension
1	2	3	2	1	2	2
2	2	4	2	1	2	2
12	3	0	1	5	4	4
13	1	5	4	3	1	2
15	1	6	5	3	1	2
16	1	4	2	4	5	6
17	5	5	6	2	5	8
18	2	8	5	3	4	2
19	5	7	2	2	3	1



	Salario - Abogado
11	450.000
12	450.000
13	450.000
14	450.000
15	450.000
16	450.000
17	450.000
18	450.000
19	450.000
20	450.000

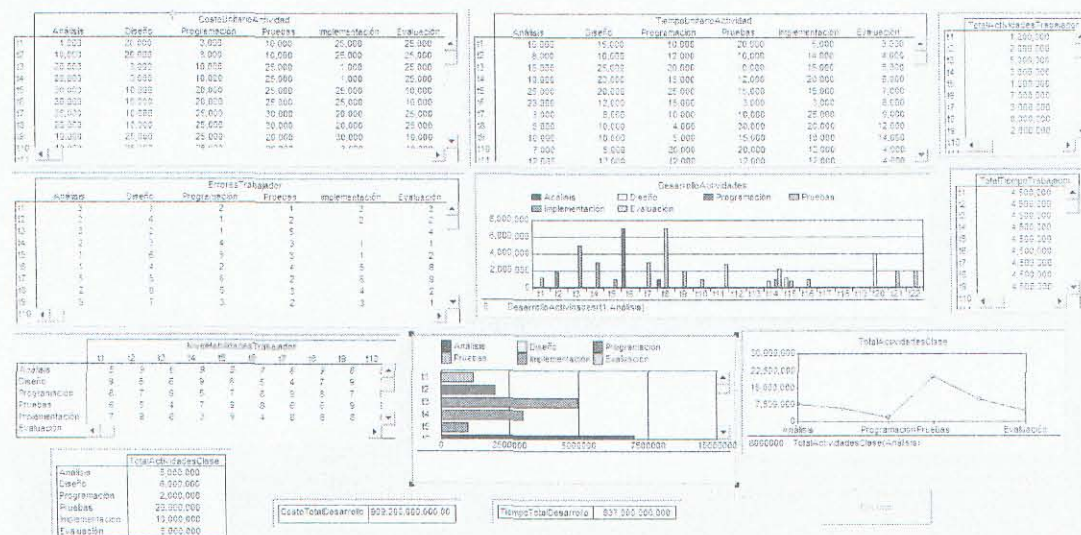
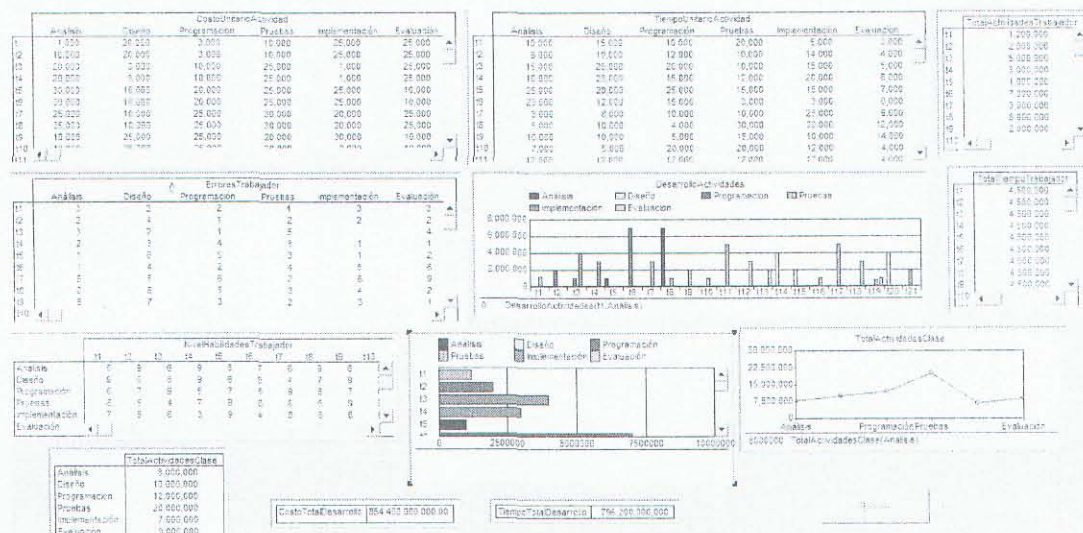
	11	12	13	14	15	16	17	18	19	20
Analysis	1	2	3	4	5	6	7	8	9	10
Design	1	2	3	4	5	6	7	8	9	10
Development	1	2	3	4	5	6	7	8	9	10
Deployment	1	2	3	4	5	6	7	8	9	10
Implementation	1	2	3	4	5	6	7	8	9	10
Validation	1	2	3	4	5	6	7	8	9	10



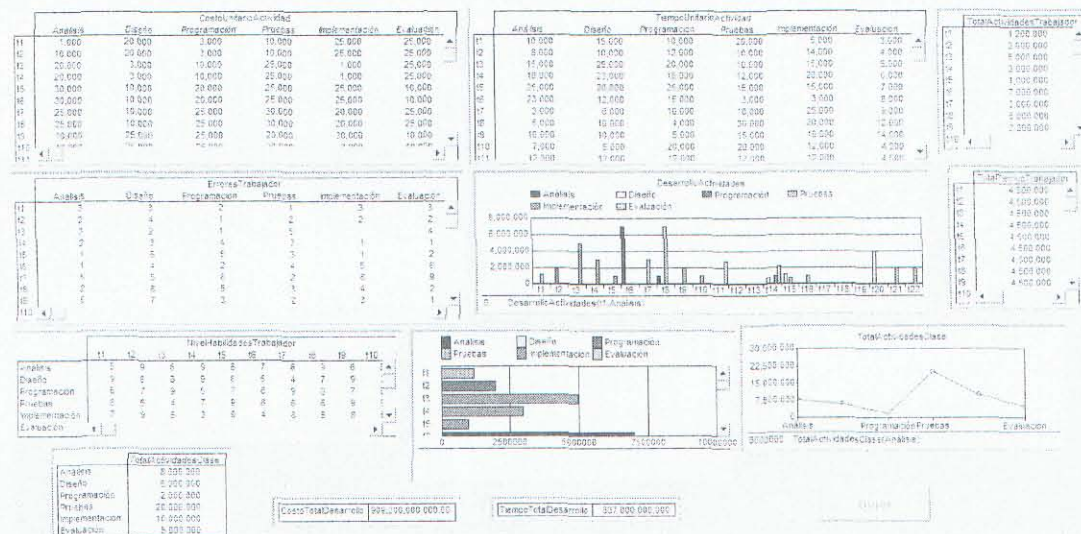
Total Project Budget	
Analysis	100,000
Design	300,000
Programming	500,000
Testing	2,000,000
Implementation	500,000
Post-implementation	200,000

Costs Total Expenditure	5 150 000 000 Gp
-------------------------	------------------

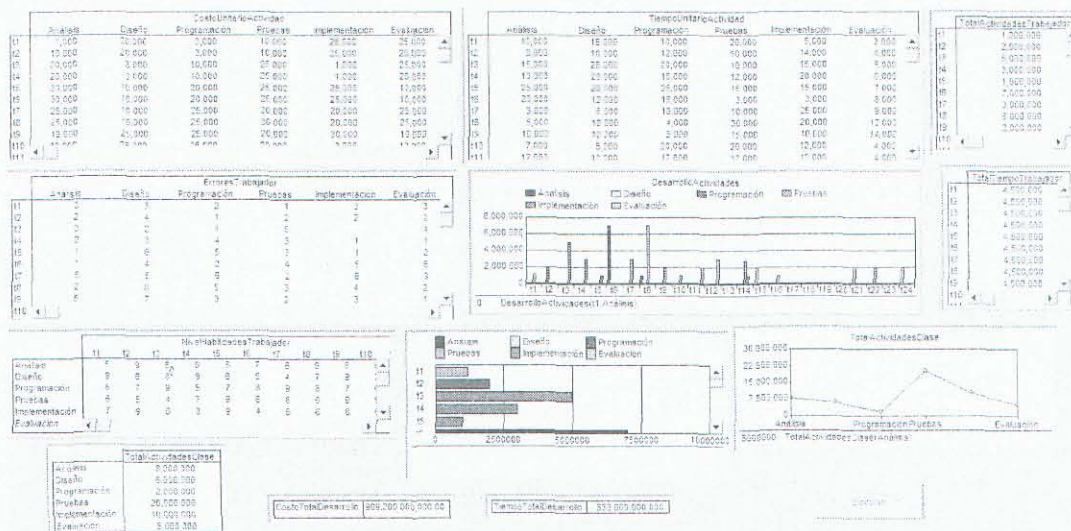
Tuition/Tuition Waiver	\$2,330.00
------------------------	------------



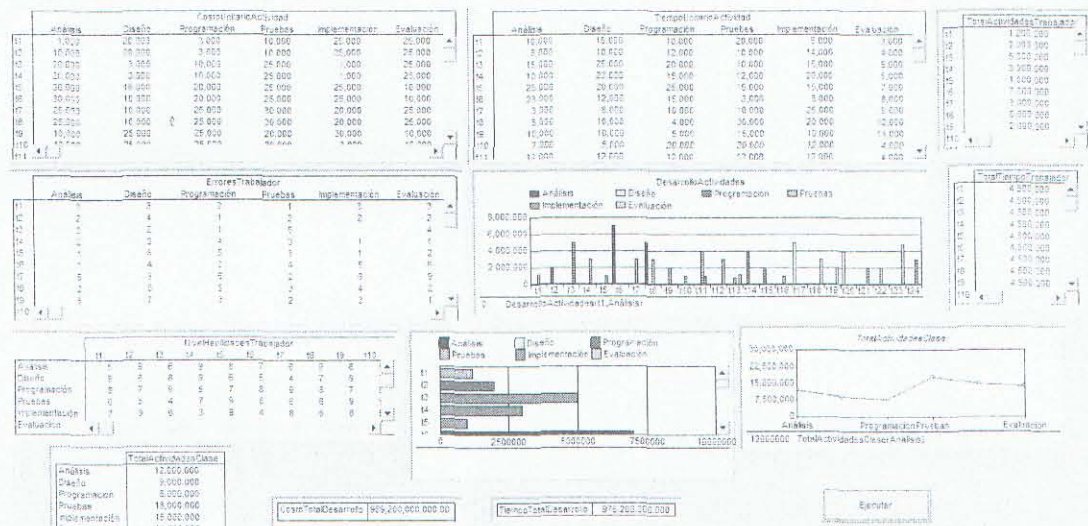
ANEXO – INSTANCIA 11



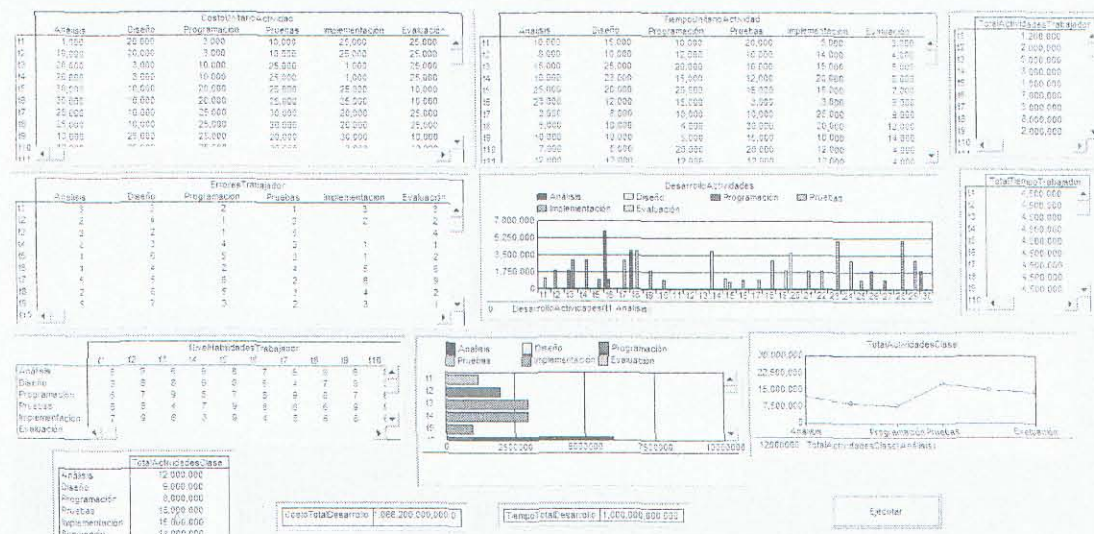
ANEXO – INSTANCIA 12



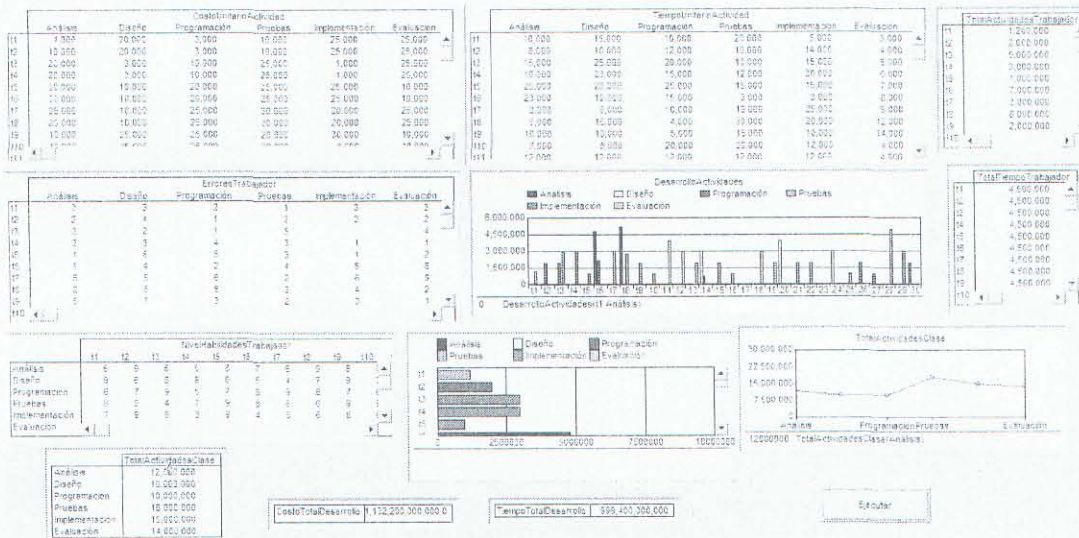
ANEXO - INSTANCIA 15



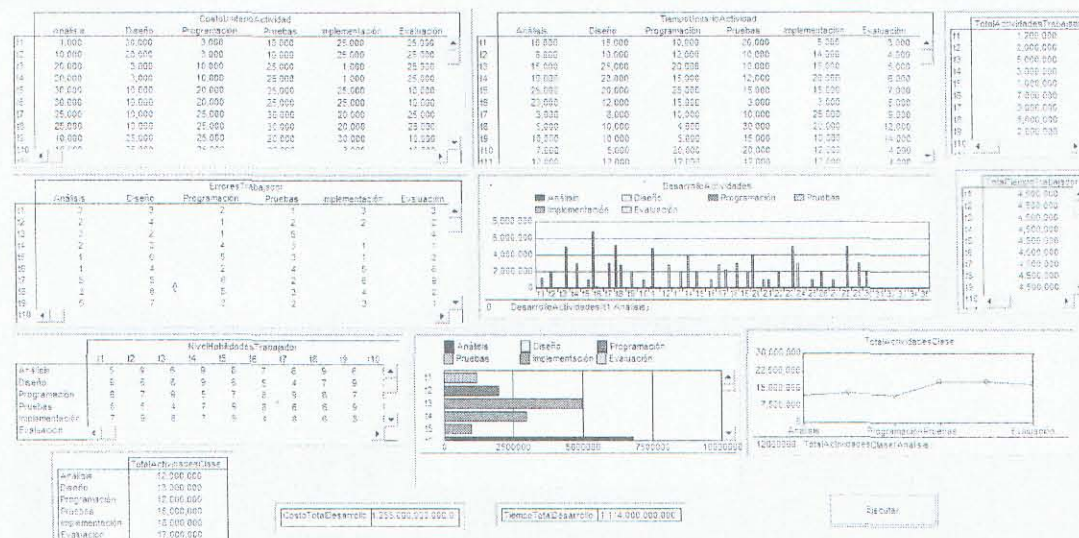
ANEXO – INSTANCIA 16



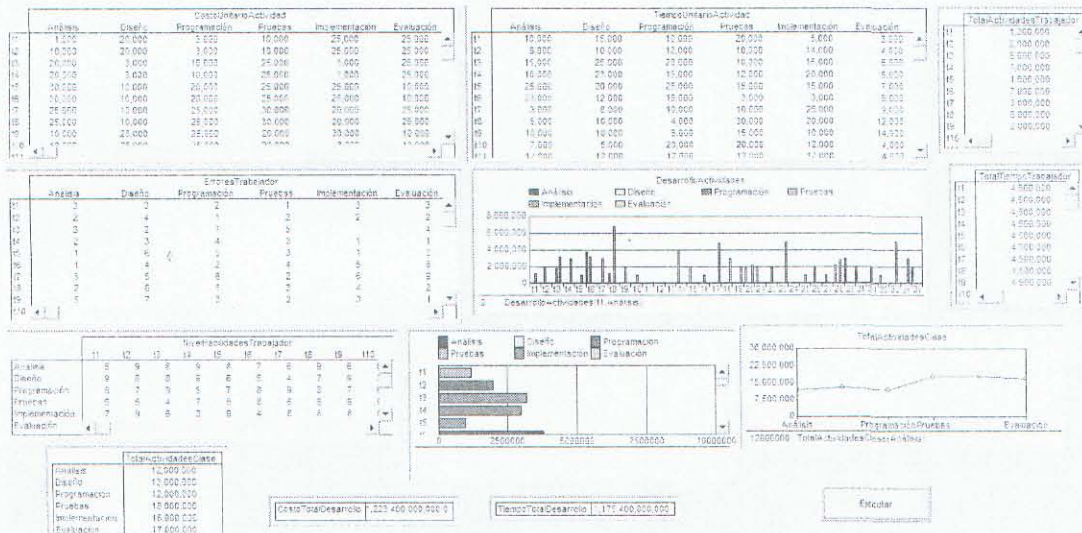
ANEXO – INSTANCIA 17



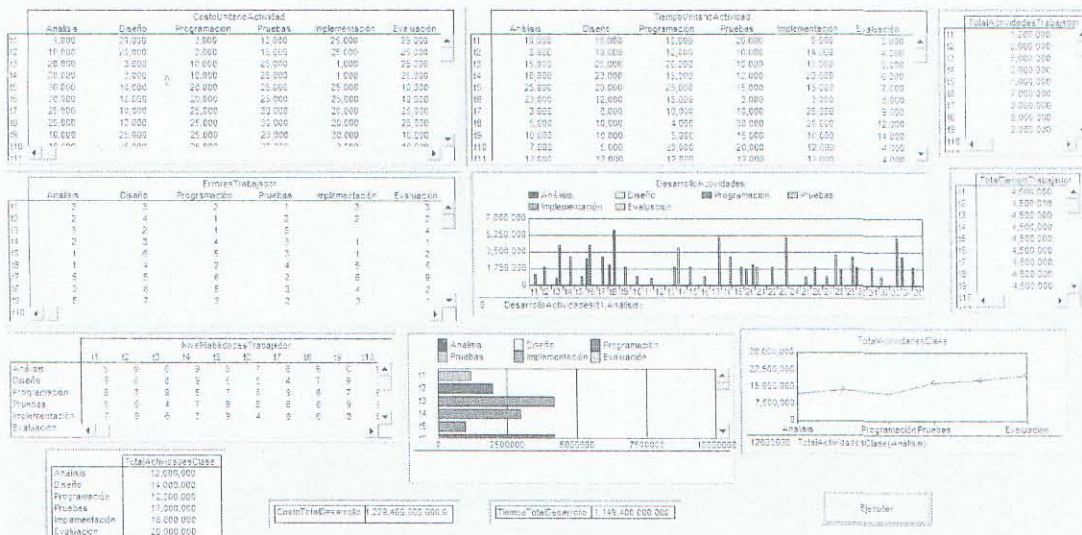
ANEXO – INSTANCIA 18

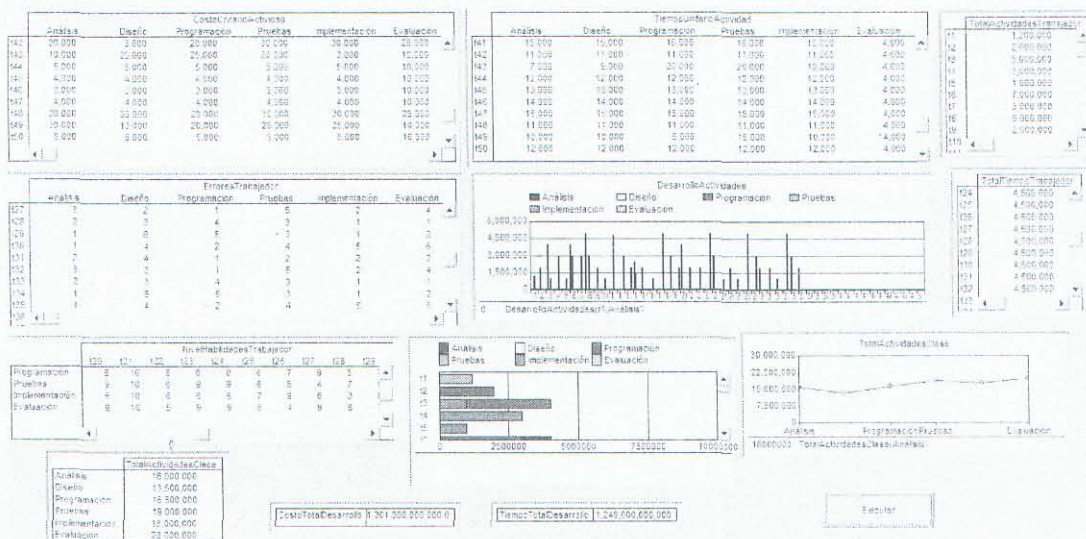
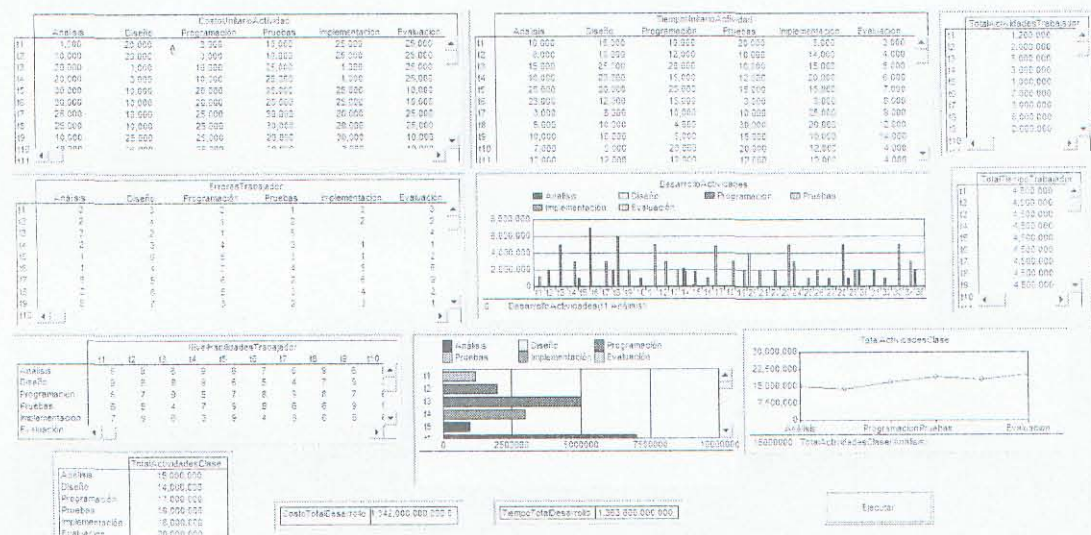


ANEXO – INSTANCIA 19

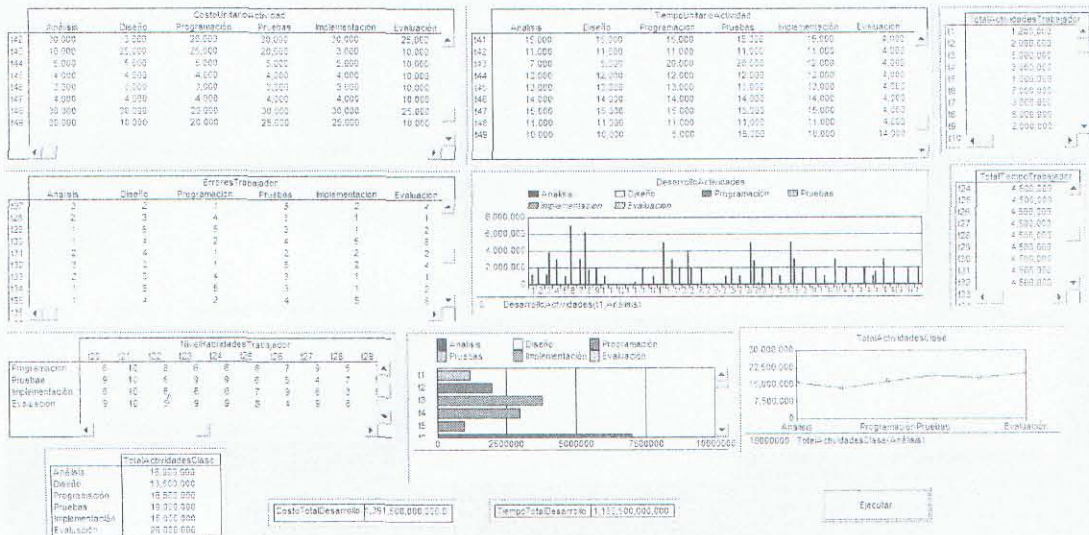


ANEXO – INSTANCIA 20





ANEXO – INSTANCIA 23



ANEXO – INSTANCIA 24

