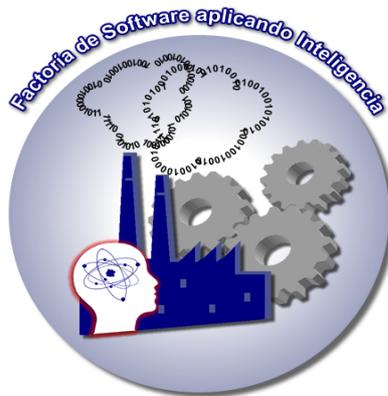


**Universidad de las Ciencias Informáticas**

**Facultad 4 y 8**



**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE  
INGENIERO EN CIENCIAS INFORMÁTICAS**



Entidad Proceso del Modelo de Factoría de Software aplicando  
Inteligencia

**Autores: Zaylí Noda García**

**Ariagna González Landeiro**

**Tutor: Ing. Yaimí Trujillo Casañola**

Ciudad de la Habana, 2007

“Año 49 del Triunfo de la Revolución”

Curso 2006-2007

## DECLARACIÓN DE AUTORÍA

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

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

---

Firma del Autor  
Zaylí Noda García

---

Firma del Autor  
Ariagna González Landeiro

---

Firma del Tutor  
Ing. Yaimí Trujillo Casañola

## DATOS DE CONTACTO

Nombre y Apellidos: Ing. Yaimí Trujillo Casañola.

Fecha de nacimiento: 20 de Octubre de 1981

Institución: Universidad de las Ciencias Informáticas,

País: Cuba.

Ciudadanía: Cubana.

Carnet de Identidad: 81102008414

Correo: Yaimí@uci.cu.

Situación laboral: Profesora Instructora, Departamento de la Especialidad.

Institución: Universidad de las Ciencias Informáticas.

Dirección: Carretera San Antonio de los Baños, Torrens, Municipio Boyeros, Ciudad de La Habana, Cuba, Código postal 19370.

### Currículo

Ingeniera Informática del Instituto Superior Politécnico José Antonio Echeverría desde Julio del 2004. Al graduarse pasa a ser profesora de la Universidad de las Ciencias Informáticas, en la Disciplina de Ingeniería y Gestión de Software. Obtiene la categoría de Instructora en octubre del 2005 y actualmente está en el proceso de categorización optando por la categoría de Asistente. Ha impartido asignaturas como Introducción a la Programación, Programación I, Práctica Profesional, Interpretación de UML, Introducción al Enfoque de Factoría de Software, Ingeniería de Software I y II entre otras. Ha desarrollado proyectos de investigación, tiene 6 artículos publicados en memorias de eventos científicos y Sitios Web. Ha ejercido como tutora de varias tesis de grado, así como ponente y miembro de tribunales. Se encuentra cursando la maestría de Gestión de Proyectos Informáticos. Ha participado como ponente en eventos científicos nacionales e internacionales de la rama. Ha trabajado en proyectos productivos como Programadora, Analista, Diseñadora y Líder de proyecto obteniendo software utilizados en empresas del país y productos para la exportación.

***"¿Por qué esta magnífica tecnología científica, que ahorra trabajo y nos hace la vida más fácil, nos aporta tan poca felicidad? La respuesta es ésta, simplemente: porque aún no hemos aprendido a usarla con tino."***

***Albert Einstein***

## **AGRADECIMIENTOS**

Deseamos agradecerle primeramente a la Revolución Cubana que nos ha dado la alegría de poder hacer posible el sueño de estudiar. A nuestra tutora Yaimí que en todo momento no ha apoyado. A todos los profesores que durante cinco años contribuyeron a nuestra formación integral como profesional.

A la Universidad de Ciencias Informáticas (UCI), por habernos formado no solo como profesionales sino como personas humildes y honestas. A todos nuestros compañeros de estudios que siempre nos brindaron su amistad y su apoyo tanto en las buenas como en las malas.

A nuestros padres por darnos la fuerza que necesitamos y el apoyo incondicional a cada momento de nuestras vidas, por ser nuestros soles y nuestro ejemplo a cada paso que damos.

En fin agradecemos a todos aquellos familiares y amigos que siempre están ahí a nuestro lado para lo que sea.

## **DEDICATORIA**

Dedico este trabajo a mi mamá, a mi papá y a mi hermana que han sido mi gran apoyo, a mi esposo que es mi vida y mi luz, a mis abuelos maternos que han sido como mis padres, a mis abuelos paternos que desde el cielo siempre me han dado su bendición y que sé que de una forma u otra siempre me han acompañado en los momentos importantes de mi vida. A todos mis tíos que siempre me han mimado muchísimo, en especial a mis tíos Mercy y Kike que más que mis tíos son mis hermanos, a mi tío Luis y Alberto, a mis primos todos, a mi ahijada que llena de alegría mis días, a mis queridos suegros que siempre me han querido muchísimo y me han dado su apoyo incondicional, en fin a todos mis amigos y a todos mis familiares.

Zaylí

Dedico mi tesis especialmente a mi mamá por ser mi apoyo, mi guía y mi razón de ser durante toda mi vida, a mi papá y a mi hermano que también me han apoyado cuando los he necesitado, a mis abuelos, a los que están hoy para disfrutar de este día y a los que no, porque todos han sabido mimarme y darme todo el cariño del mundo para poder seguir adelante, y en especial a mi abuelo Benito que su mayor deseo hubiese sido verme convertida en toda una profesional, a toda mi familia en general, a mis vecinos por apoyar y cuidar a mi mamá en tantos años que ha estado sola. Se la dedico a todas las personas que me han apoyado y han sido mis compañeros y amigos durante estos cursos, dígame Alionuska, Rubén, Lidice, Carlito, Pepito, Madelyn, Yoan, Yohandy, y Sabina entre otros, a Daimara por soportar mis indiferencias y soportarme por más de 10 años, a mi compañera de tesis que ha tenido que aguantar mis desaparecidas. En fin a todos los que me han apoyado, guiado y ayudado a formarme durante toda mi vida de estudiante, sin dejar de mencionar a todas las personas maravillosas con las que he compartido en estos últimos meses en la Facultad Regional de Artemisa.

Ariagna

## **RESUMEN**

En la Universidad de Ciencias Informáticas (UCI) en los últimos años se ha observado un desarrollo vertiginoso de la producción de software, cobrando esta gran importancia dentro de la economía nacional, esto ha llevado a la creación de varias estrategias y procedimientos para acometer esta misión de manera factible, pero a pesar de todos los esfuerzos en torno a este tema, todavía existen deficiencias en la definición de los procesos en cada uno de los proyectos, en los que no se tienen en cuenta los requisitos principales de un software como son la calidad y las necesidades del cliente. El objetivo de esta investigación es definir la Entidad Proceso del Modelo de Factoría aplicando Inteligencia para la Universidad de las Ciencias Informáticas, para ello se utilizó la metodología de la investigación científica, teniendo como hipótesis que si se realiza un estudio del proceso de desarrollo de software en la UCI y de los modelos de producción existentes en el mundo se obtendrá la definición de una Entidad Proceso del Modelo de Factoría aplicando Inteligencia. Esta Entidad incorporada a dicho modelo, puede contribuir a la mejora de los procesos de desarrollo y a una producción en la que se destaquen la eficiencia y calidad, puesto que su definición esta basada en una metodología por líneas de producción y con el uso de estándares internacionales. Se espera que este proceso sea aplicable a todo desarrollo de software, siendo flexible, adaptable, y luego de aplicado sea repetible y mejorable continuamente dadas las condiciones de cualquier proyecto o equipo de desarrollo.

**Palabras Claves:** Factoría de Software, Proceso de Desarrollo de Software.

## ÍNDICE

INTRODUCCIÓN .....	1
Capítulo 1: Introducción al enfoque de Factoría y al Proceso de Desarrollo de Software .....	5
1.1.    Introducción.....	5
1.2.    Factoría de Software .....	5
1.2.1.    Consideraciones generales sobre Factoría de Software.....	7
1.3.    Modelos de Factoría de Software.....	8
1.3.1.    Modelo basado en la norma ISO 9001 y CMM.....	8
1.3.2.    Modelo Eureka .....	10
1.3.3.    Modelo Clasificadorio .....	12
1.3.4.    Modelo propuesto por Basili .....	13
1.3.5.    Modelo Replicable .....	15
1.3.6.    Modelo de Factoría de Software aplicando Inteligencia .....	16
1.3.7.    Consideraciones sobre los modelos de factoría de software estudiados .....	18
1.4.    Proceso de desarrollo de Software .....	19
1.4.1.    Mapa de Proceso.....	21
1.5.    Gestión de Proceso .....	22
1.6.    Metodologías de desarrollo de Software.....	24
1.7.    Modelos y estándares de calidad.....	25
1.7.1.    Modelo de Madurez de Capacidades Integrado (CMMI).....	25
1.7.1.1.    Método de Evaluación CMMI (SCAMPI).....	27
1.7.2.    ISO (International Standard Organization) (Organización Internacional De Normalización).....	28
1.7.3.    Comparación entre ISO 15504 y CMM.....	29
1.8.    Experiencia Internacional. ....	29
1.9.    Conclusión .....	33
Capítulo 2: Definición de la Entidad Proceso.....	34
2.1.    Introducción .....	34
2.2.    Universidad de las Ciencias Informáticas (UCI). ....	34
2.3.    Métodos, y procedimientos utilizados. ....	36
2.3.1.    Entrevista # 1 .....	37

2.3.2. Entrevista # 2 .....	37
2.3.3. Encuesta .....	38
2.4. Entidad Proceso del Modelo de Factoría de Software aplicando Inteligencia. ....	39
2.4.1. Descripción del proceso 201 Organizar trabajo.....	42
2.4.2. Descripción del proceso 202 Definir el proceso de desarrollo de software. ....	43
2.4.3. Descripción del proceso 203 Controlar el proyecto .....	46
2.4.4. Estructura organizacional.....	48
2.4.5. Personas necesarias para el proceso.....	50
2.4.6. Procesos de soporte.....	51
2.5. Conclusiones. ....	53
Capítulo 3: Aplicación práctica .....	54
3.1. Introducción .....	54
3.2. Análisis de los resultados de la aplicación de las encuestas y entrevistas.....	54
3.3. Aplicación Práctica .....	56
3.3.1. Caso de estudio.....	56
3.3.2. Proceso “Modelamiento del negocio”.....	57
3.3.3. Organización.....	61
3.3.4. Repositorio.....	61
3.3.5. Bases tecnológicas.....	63
3.3.6. Elementos del Proceso.....	63
3.4. Conclusiones .....	69
CONCLUSIONES .....	70
RECOMENDACIONES.....	71
REFERENCIAS BIBLIOGRÁFICAS .....	72
ANEXOS .....	75
GLOSARIO DE TÉRMINOS.....	116
GLOSARIO DE SIGLAS .....	119

### INTRODUCCIÓN

En la actualidad el mundo se dirige hacia un nuevo modelo económico basado en el conocimiento, los innumerables avances en las tecnologías han contribuido a que el conocimiento sea considerado como un nuevo recurso, generador de importantes ventajas competitivas.

La gestión del conocimiento cobra, por tanto, una nueva dimensión: está entrando en la Categoría de Sociología de la Información, donde, su principal función es que una empresa u organización no deba pasar dos veces por un mismo proceso para resolver de nuevo el mismo problema, sino que ya disponga de mecanismos para abordarlo utilizando información guardada sobre situaciones previas.

Esto conjuntamente con el desarrollo de las Tecnologías de la Informática y las Comunicaciones (TIC) ha traído consigo que la Industria del Software se encuentre inmersa en grandes cambios, pues se requieren software cada vez más grandes y complejos con mayor implicación de personas en la producción, todo esto conduce a la necesidad de implantar nuevas metodologías y modelos, así como la reformulación de nuevos conceptos para optimizar los procesos de desarrollo de software.

Cuba no se encuentra ajena al desarrollo de la informática a nivel mundial, en los últimos años ha impulsado la producción de software nacional, pero aún no cuenta con toda la experiencia requerida para afrontar este gran reto. Por lo cual el Gobierno Cubano tiene como una de sus tareas reorganizar la Industria Cubana del Software, cualificando los recursos humanos necesarios para informatizar la sociedad, y así poder insertarse en el mercado de software a nivel mundial.

Dada esta gran necesidad se crea en el año 2002 la Universidad de las Ciencias Informáticas (UCI), primera universidad surgida al calor de la Batalla de Ideas, la cual pretende ser la vanguardia del desarrollo de las empresas de software en Cuba y llevar la informatización a todos los sectores de la sociedad: Salud, Educación, Cultura, Deporte, Turismo, Prensa, etc. Regir y propiciar un avance tecnológico que permita convertir la industria del software en un renglón fundamental de la economía cubana, pues en ella, juega un papel fundamental la relación estudio-trabajo, donde el estudiante y el profesor están vinculados a la producción de software.

Actualmente se ha percibido en varios proyectos productivos de la UCI que existen deficiencias en la definición de proceso, no se lleva a cabo un proceso planificado, ni un flujo definido de proceso, por consiguiente, no se cumple con las fechas de entrega previstas, lo que promueve desconfianza por parte

de los clientes. Sí se sigue una metodología pero no se concretan las actividades, el flujo de procesos, ni los roles, afectándose la eficiencia, la calidad, y el tiempo de desarrollo de un producto.

Esto trae consigo que no exista documentación completa de los productos, que el trabajo en equipo no esté organizado, y que no se conozcan cuales son exactamente las necesidades finales del cliente.

La solución de todos estos problemas esta enfocada hacia una producción estructurada, que ofrezca servicios personalizados, encaminada a incrementar la calidad del producto. Las factorías de software, tienen como uno de sus objetivos, industrializar el proceso de desarrollo de software, a través de procedimientos, normas y métodos definidos que cuenten con las herramientas necesarias para su implantación.

Las factorías se perfilan, en la unión del conocimiento y la metodología, en la que se acumule todo lo desarrollado, lo que permite conseguir altos porcentajes de reutilización. La industrialización del proceso de software facilita la evaluación, medición y control del proceso, y con ello, su mejora y adaptación al cambio, no sólo en el análisis de los procesos internos, sino en la investigación de nuevas tecnologías, herramientas y métodos. La ardua tarea consiste en aportar un modelo para la mejora del desarrollo de software, definiendo una serie de prácticas técnicas y de gestión que deben adoptarse en los proyectos informáticos para asegurar el cumplimiento de los plazos, los presupuestos y las exigencias de calidad de los clientes.

Los modelos de factorías existentes han surgido debido a las necesidades que se han presentado a nivel mundial, pero hasta el momento solo el Modelo de Factoría aplicando Inteligencia, cumple con las condiciones socioeconómicas de Cuba, pero su Entidad Proceso no posee todas las especificidades necesarias, por lo que se tratar de definir un proceso que de solución a los problemas que hoy presenta la producción de software.

Para cumplir las especificidades de un proceso de desarrollo en la actualidad se pudiera tener, como nuevo paradigma, que las entidades sean gestionadas siguiendo el *enfoque de proceso* o más comúnmente conocido como *gestión por proceso*, que es una necesidad imperativa para las organizaciones que buscan éxito y competitividad a través de la calidad.

La gestión por proceso contribuye a que se desarrollen modelos del proceso de desarrollo de software que permitan conocer el proyecto a profundidad y los posibles errores de las decisiones tomadas sobre la globalidad del mismo, y así garantizar su mejora continua, aspectos que son esenciales para tener una buena planificación y control del proyecto.

Con el fin de contribuir a solucionar los problemas en la definición de los procesos en los proyectos de desarrollo de software en la UCI, se define utilizar el modelo de factoría diseñado por Yaimí Trujillo Casañola, aportándole a este modelo la definición de la Entidad Proceso de Desarrollo la cual es importante en toda factoría de software, puesto que contribuye con la obtención de un producto final con calidad.

Luego del análisis e investigación del proceso actual de desarrollo de los proyectos productivos en la UCI, este trabajo se traza como **problema científico** que “las deficiencias en la definición de los procesos de desarrollo de software afectan la organización de la producción”. De aquí que el **objeto de estudio** lo constituya el proceso productivo en la Universidad de las Ciencias Informáticas (UCI). Todo ello encaminado a cumplir el **objetivo general** de definir la Entidad Proceso de la propuesta de Modelo de Factoría aplicando Inteligencia. El **campo de acción** abarca el proceso de desarrollo de software.

Basado en el problema y como vía para alcanzar el objetivo propuesto, se formuló la **hipótesis**: Si se realiza un estudio del proceso de desarrollo de software en la UCI y de los modelos de producción existentes en el mundo se obtendrá la definición de una Entidad Proceso del Modelo de Factoría aplicando Inteligencia

Como orientación para el cumplimiento de los objetivos planteados se elaboraron y cumplieron las siguientes **tareas específicas de investigación**:

- Caracterizar el enfoque y el modelo de factoría.
- Evaluar los diferentes modelos de factorías, mediante una comparación entre los mismos.
- Caracterizar los procesos de desarrollo de software y la gestión de proceso.
- Identificar el impacto que han tenido las diferentes empresas al llevar el desarrollo de los productos mediante la Factoría de Software.
- Caracterizar el entorno de trabajo.
- Determinar las metodologías de desarrollo de software y los estándares de calidad aplicables a la Entidad Proceso del modelo de factoría de software que se propone.
- Definir las características y conceptos necesarios para la Entidad Proceso.
- Definir la Entidad Proceso.

El presente documento se estructura en tres capítulos:

El Capítulo 1 Introducción al enfoque de Factoría y al proceso de desarrollo de software, describe detalladamente el enfoque y los modelos de factoría, se mencionan los impactos que ha tenido a nivel

mundial la aplicación de la factoría, se definen los conceptos más importantes que pueden ser útiles para definir el proceso y se realiza un breve explicación a cerca de los estándares de calidad.

Capítulo 2: Definición de la Entidad Proceso. Tiene como objetivo caracterizar la situación de la producción de software en la UCI, proponer la Entidad Proceso del Modelo de Factoría de Software aplicando Inteligencia, así como describir los métodos y procedimientos utilizados para llevar acabo la investigación.

Capítulo 3: Aplicación Práctica, se presenta la Entidad Proceso, se plantean los pasos a seguir para la implantación del modelo, y se analizan los resultados obtenidos en los diferentes métodos y procedimientos.

## Capítulo 1: Introducción al enfoque de Factoría y al Proceso de Desarrollo de Software

### 1.1. Introducción

En el presente capítulo se brinda una visión general de los aspectos relacionados con el enfoque de factoría, y los conceptos necesarios para el estudio y planteamiento de la Entidad Proceso para este enfoque como son: ¿Qué es una Factoría de Software y cuales son sus objetivos? Se plantean algunos estándares de calidad, se mencionan los impactos a nivel mundial de la aplicación de la factoría y los modelos de factoría más usados, así como los conceptos relacionados con la Entidad Proceso los cuales son fundamentales para brindar la solución.

### 1.2. Factoría de Software

Se denomina factoría, de forma genérica, a cualquier tipo de fábrica o industria, es decir, a cualquier tipo de instalación en la cual se produce la transformación de materias primas o productos semiterminados en otros productos, bien para otras industrias, bien para su uso o consumo final. Por extensión se está aplicando esta palabra para designar determinadas actividades en las cuales no se produce consumo y transformación de materias y que tienen como objeto final la obtención de productos intangibles: factoría de comunicación, factoría de cine, factoría de software.

El término factoría de software fue utilizado por primera vez en la década del 60 en Japón. Pero varias empresas asociaron el término al mero desarrollo de software. A continuación se enuncian varios conceptos de factoría de software dado por distintos autores:

Cusumano en 1989 plantea que: “Una empresa productora de software que no responda a características como: producción de software en gran escala, estandarización de tareas, estandarización del control, división del trabajo, mecanización y automatización, no puede ser considerada una factoría de software.”(CUSUMANO 1989.)

En 1992 Cantone dice que:”Una factoría de software debe, para ser flexible, ser capaz de producir varios tipos de productos; llevar a cabo conceptos de Ingeniería de Software (metodología, herramientas, gestión

de la configuración), y también ser capaz de estudiar, diseñar, implementar y mejorar sus sistemas y procesos.” (CANTONE 1992)

Cusumano y Cantone plantean que las factorías de software buscan industrializar el proceso de desarrollo aplicando siempre las buenas prácticas de la ingeniería de software incluyendo la mejora continua de procesos.

Basili en 1992, plantea que: “Una organización con características de factoría de software debe poseer una estructura de construcción de software basada en componentes. Los componentes utilizados en la construcción del software pueden ser desarrollados por una unidad de producción de componentes (factoría de componentes). La factoría de componentes es la base para la implementación de una factoría de software.”(BASILI 1992)

Fernstrom en ese mismo año expresa que:”Una organización fabril para el desarrollo de software debe tener claro el asunto del “software único”, es decir, todo software es único, pero algunas partes de ellos se pueden repetir en varios proyectos. El proceso industrial debe contener el desarrollo, almacenamiento y montaje de partes reutilizables en un producto.”(FERNSTROM 1992.)

Según Basili y Fernstrom la Factoría de software no es solo industrializar el proceso sino que se debe de utilizar también la producción de software basada en componentes, para permitir un alto uso de componentes reutilizables, lo que permite disminuir el tiempo de desarrollo y elevar la calidad.

En el año 2001 Li señala que:”Una factoría de software debe poseer un conjunto de herramientas estandarizadas para la construcción de software, bases históricas para ser usadas en la dirección de proyectos, y principalmente, poseer un alto grado de reutilización de código en el proceso de desarrollo de un determinado software, apoyado en una base de componentes reutilizables.” (LI 2001.)

Ya Li no solo habla de la reutilización del código, y de industrializar el proceso, sino que plantea la necesidad de estandarizar el proceso de desarrollo y utilizar bases históricas para estimar recurso y costo basados en experiencias reales.

Por su parte Fernández y Teixeira expresa que: “Una factoría de software es una organización con procesos estructurados, controlados y mejorados de forma continua, considerando principios de Ingeniería Industrial, orientados a dar respuesta a múltiples demandas de distintas naturaleza y alcance. Dirigida a la creación de productos de software, conforme a los requerimientos documentados de los usuarios y clientes, de la forma más productiva y económica posible.” (FERNANDES 2004)

Según Fernández una factoría de software puede tener varios dominios de actuación en dependencia de las fases de desarrollo del software en las que opere, desde un proyecto de software completo, hasta un proyecto físico o solo codificación del software.

### **1.2.1. Consideraciones generales sobre Factoría de Software**

Estos autores enumeran una serie de características inherentes a una estructura fabril para software:

1. Proceso definido y estandarizado para el desarrollo de software.
2. Estimación de costos y tiempo basados en el conocimiento real de la capacidad productiva, mediante métodos de obtención basados en datos históricos.
3. Planificación y control de la producción
4. Control y almacenamiento en bibliotecas de componentes de software (documentos, código, métodos, etc.)
5. La producción de software debe estar fuertemente basada en métodos y técnicas estandarizadas.
6. Producción a gran escala.

El proceso de desarrollo de software ya se inclina hacia la producción sistematizada que ofrezca servicios y prestaciones diferenciadas centradas en obtener un producto de mayor calidad, es decir ya la sociedad de la información se dirige hacia la industrialización de este proceso, es decir a la factoría de software, facilitando la adaptación al cambio, la reducción del trabajo y el control del proceso de desarrollo.

Basado en las definiciones anteriormente analizadas este trabajo asume factoría de software como una organización estructurada creada para el desarrollo de software, donde se apliquen los conceptos de ingeniería (metodologías, reutilización de componentes software, automatización de los procesos de construcción, soporte y gestión, el uso de los estándares, métricas de tiempos, costo y errores, producción a gran escala, alta productividad, etc.) a la producción de software.

Entre los principales objetivos trazados por una factoría de software están:

1. Industrializar el desarrollo de sistemas de software.
2. Producción de software a gran escala.
3. Lograr una alta productividad en el desarrollo de software.
4. Establecer líneas de producción.
5. Mejora continua de los procesos.
6. Estimación de costos y plazos extremadamente precisa.

7. Reducción de los costos de producción.
8. Lograr un buen control de la calidad.
9. Especializar al profesional en una tarea específica del proceso, concentrando sus esfuerzos en dicha tarea.

### **1.3. Modelos de Factoría de Software**

Los modelos de Factoría de Software son la forma en que se han llevado a la práctica el enfoque de factoría de software por distintas empresas y entidades que los han adoptado. Se describen una selección de los modelos de Factoría de Software más representativos encontrados en la bibliografía consultada, los mismos servirán como base de la investigación a partir de los elementos más importantes identificados en cada uno de ellos. Se abordarán cinco modelos, los mismos son:

- Modelo basado en la norma ISO 9001 y CMM.
- Modelo Eureka.
- Modelo Clasificadorio.
- Modelo propuesto por Basili.
- Modelo Replicable.

#### **1.3.1. Modelo basado en la norma ISO 9001 y CMM**

El Modelo basado en la Norma ISO 9001 y CMMI es aplicable a cualquier Factoría de Software; pueden adaptarse las entidades que componen el mismo de acuerdo a las características y necesidades del entorno. En este modelo se hace una división de los elementos fundamentales de una Factoría de Software en cinco entidades bien definidas las cuales son:

- Técnicas: Comprende el contexto de las técnicas que sirven de soporte al proceso de desarrollo, técnicas para la reutilización de software, para el desarrollo basado en componentes y otras técnicas utilizadas por la factoría.
- Proceso: Representa el proceso de desarrollo de software, los flujos de trabajo y actividades que componen el mismo.
- Trabajadores involucrados: Personas que actúan directamente en el desarrollo de software.

- Gestión de la factoría: Define la estructura organizacional de la Factoría de Software, el proceso fabril y la gestión de calidad.
- Activos del proceso, herramientas y componentes de código: Entiéndase como activos del proceso modelos, patrones, algoritmos utilizados como artefactos en el proceso. Los activos del proceso también pueden ser denominados como componentes de infraestructura, componentes de valor en el proceso.

La arquitectura propuesta por el modelo se puede observar a continuación en la figura 1.

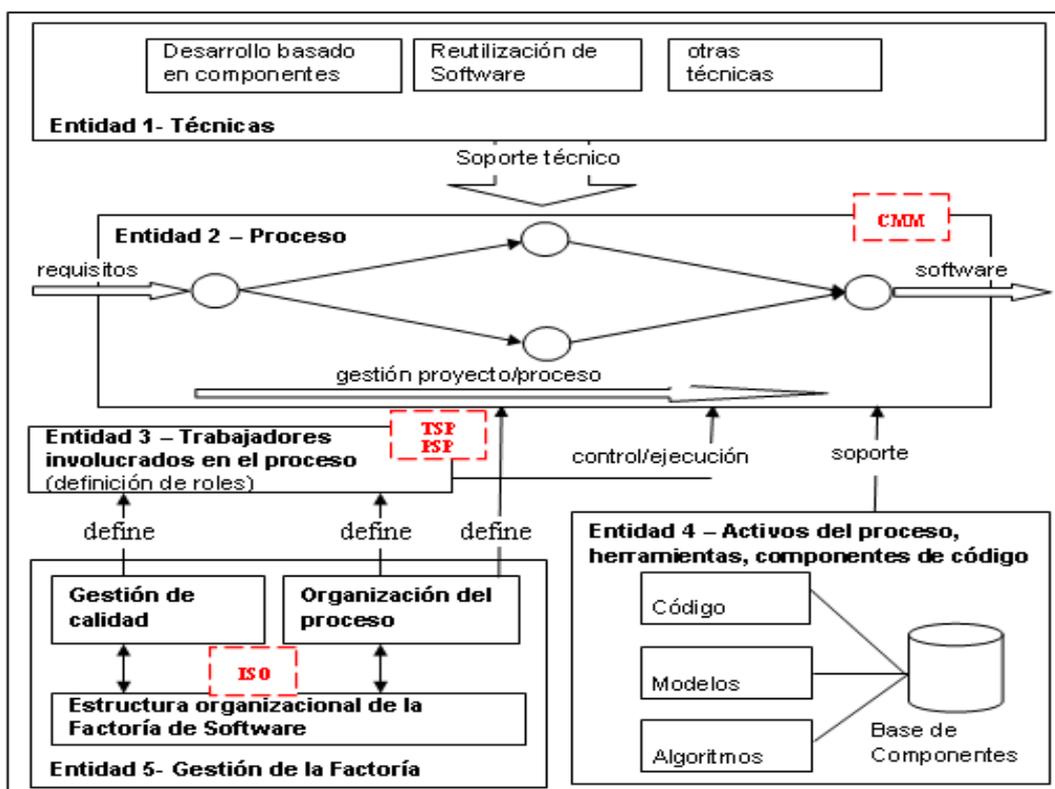


Figura 1.- Arquitectura del Modelo basado en la norma ISO 9001 y CMM

“En la misma se observa que la entidad Técnicas provee el soporte técnico y conceptual para la definición del proceso. Este es guiado por el estándar de calidad CMM, los requisitos de calidad para la organización de la factoría son definidos por la norma ISO 9001. El modelo toma la norma ISO 9001 como un estándar utilizado en el contexto industrial cuyo enfoque está en el sistema de calidad organizacional, propone un conjunto de principios probados para mejorar la calidad final del producto mediante mejoras en la

organización de la empresa. CMM es designado para la industria del software, de este modo las áreas claves proveen detalles importantes para la evaluación y mejora del proceso de desarrollo, su propósito es guiar a las organizaciones en la selección de estrategias de mejora determinando la madurez del proceso actual e identificando los puntos importantes que se deben estudiar y trabajar para mejorar tanto el proceso como la calidad del software.”(HOZ 2005)

“La entidad Gestión de la Factoría define, a través de las sub-entidades Gestión de Calidad y Organización del Modelo de Proceso, los trabajadores involucrados en el proceso de desarrollo de software y sus roles. La sub-entidad Organización del Modelo de Proceso define características y organización del proceso de desarrollo de software. Los Trabajadores Involucrados son guiados por los modelos PSP (Personal Software Process) y TSP (Team Software Process). Los activos del proceso, las herramientas y los componentes de código dan soporte al proceso de desarrollo de software.”(HOZ 2005)

Lo más relevante del modelo presentado anteriormente es que define objetivamente y describe las entidades que forman el modelo y las relaciones que se establecen entre ellas, así como la aplicación de normas y técnicas de calidad usadas hoy en el mundo del software, aunque CMM es un estándar que se abolió en diciembre del 2005 por lo que una de las mejoras que se le propone es el uso de CMMI, además no desglosa las actividades, objetivos y características de cada una de las entidades que lo conforman.

### **1.3.2. Modelo Eureka**

La Iniciativa intergubernamental EUREKA ha movilizó a la comunidad europea para innovar. Creada en 1985, esta Red descentralizada permite a las pequeñas, medianas y grandes empresas, centros de investigación, universidades y administraciones nacionales reunir fuerzas para llevar a cabo proyectos de investigación y desarrollo (I+D) cercano al mercado a través de proyectos de colaboración transnacional. Uno de los resultados más relevantes es el proyecto Eureka Software Factory.

“El modelo Eureka surgió como el proyecto Eureka Software Factory. El objetivo del proyecto es crear un mercado para productos CASE. En el mismo participan un conjunto de compañías europeas; tales compañías actúan en las siguientes áreas: manufactura de computadoras, instituciones de investigación, producción de herramientas CASE y desarrollo de sistemas.” (HOZ 2005)

El modelo está compuesto de los elementos: proceso, reglas, herramientas e información, personas y computadoras, ver figura 2.

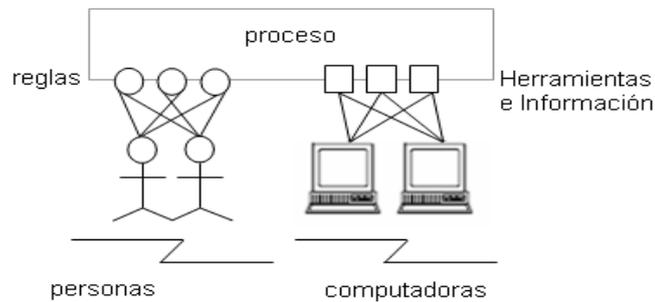


Figura 2.- Arquitectura del Modelo Eureka

“El proceso de desarrollo está compuesto por reglas, las que son definidas por las personas involucradas en el ambiente de desarrollo de software y constituyen patrones a seguir, algoritmos, métodos de desarrollo de software. Las herramientas e información almacenadas, soportan la automatización del proceso de desarrollo.” (HOZ 2005).

El modelo propone un proceso de desarrollo de software distribuido, siguiendo el enfoque software bus. Lo que permite que a través de las reglas se puedan unir los componentes realizados por distintos equipos para formar el producto (ver Anexo 1).

En el Anexo 1 se aprecia que utilizando este modelo los componentes pueden ser desarrollados por diversas Factorías de Software en localidades diferentes, y su conexión se realiza de acuerdo a reglas establecidas. Esto permite además que cada área se pueda especializar en diferentes líneas de producción y que las factorías que usen ese modelo puedan compartir y reutilizar los componentes para la construcción del software. Además se clasifican los componentes en los de servicios y los de interfaz, reflejándose un desarrollo de software por capas.

El aporte de este modelo está en el desarrollo distribuido de software, da una visión de cómo se puede llevar la construcción de un producto software entre diferentes factorías, y después realizar la unión de los componentes elaborados por cada una para formar el producto final todo esto soportado por sistemas automatizados y un conjunto de reglas que establecen las condiciones para cada uno de los artefactos que se van obteniendo. Pero este modelo no enuncia el cómo se organiza la producción, el proceso y los desarrolladores, cómo gestionar un proyecto, cómo se define el proceso y no usa estándares de calidad.

### 1.3.3. Modelo Clasificadorio

El Modelo Clasificadorio propuesto por Fernández y Teixeira está dirigido a clasificar las factorías de acuerdo al alcance o ámbito de funcionamiento que tienen a lo largo del proceso de desarrollo de software. Una Fábrica de Software puede ser clasificada en:

- Factoría de Proyectos Ampliada.
- Factoría de Proyectos de Software.
- Factoría de Proyectos Físicos.
- Factoría de Programas.

En la figura 3 se puede observar la clasificación según el alcance del proyecto.

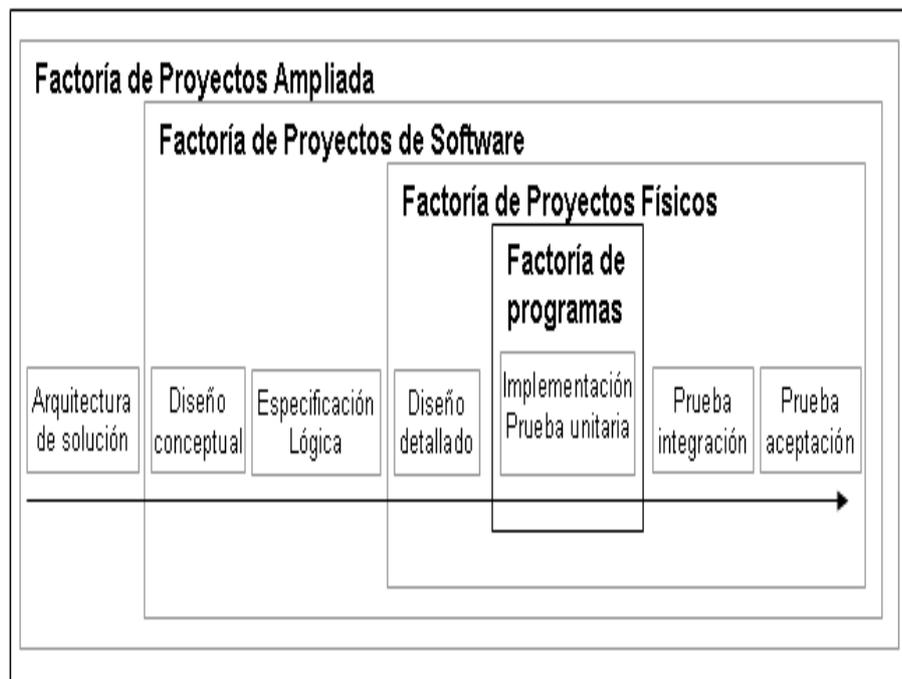


Figura 3.- Arquitectura del Modelo Clasificadorio

“Una Factoría de Proyectos Ampliada comprende el concepto de arquitectura de solución. La arquitectura de solución es una etapa anterior al diseño conceptual del software, la cual se ocupa en proyectar una solución en la que el software está formado por los componentes más significativos arquitectónicamente, se definen los principios que orientan el diseño y evolución del software. La arquitectura de solución puede

contener, además del software, definición de procesos, definición de equipamiento, infraestructura de redes, plataforma de desarrollo, patrones a seguir.

La Factoría de Proyectos de Software abarca todo el ciclo de vida sistémico para la realización del software, correspondiente al análisis, diseño, implementación, prueba e implantación. En este tipo de factorías se tiene un conocimiento al detalle del negocio a automatizar.

La Factoría de Proyectos Físicos se abstrae del enfoque sistémico del software, se dedica al diseño, implementación y prueba. No se tiene un pleno conocimiento del negocio.

La Factoría de Programas, considerada la menor de las entidades, tiene como objetivos desarrollar componentes de código para la construcción del software. Esta factoría no se preocupa del contexto sistémico ni del diseño, se ocupa de producir código según las especificaciones del diseño. Posee como entrada la especificación del diseño de una parte del software y su salida es un componente de código que formará parte del software a desarrollar.” (FERNANDES 2004)

Este modelo aporta una clasificación para las factorías en dependencia de la magnitud del proyecto, da una visión de cómo puede ser clasificado en correspondencia a las fases del proceso de desarrollo que se lleven a cabo, así mismo da la idea de que en la factoría se realiza un ciclo de vida de un producto o parte de él y cómo se puede ir avanzando en este enfoque, ya que en un futuro se pudiera pasar a una factoría de mayor o menor alcance. Este modelo solo se enmarca en el proceso, no define las demás áreas involucradas en la producción, la gestión del proyecto, la organización de la producción y de los desarrolladores, el uso de herramientas para la automatización de los procesos de construcción, soporte y gestión, entre otras.

#### **1.3.4. Modelo propuesto por Basili**

El presente modelo se plantea que un área de producción con características de Factoría debe estar basada en componentes y que los mismos pueden ser desarrollados por una factoría de componentes. O sea, propone dividir el área de producción en dos subáreas: la de producción de software y la de componentes.

Como muestra la figura 4, el modelo se divide en organización basada en proyectos de software (producción de software), y factoría de componentes (unidad de producción de componentes).

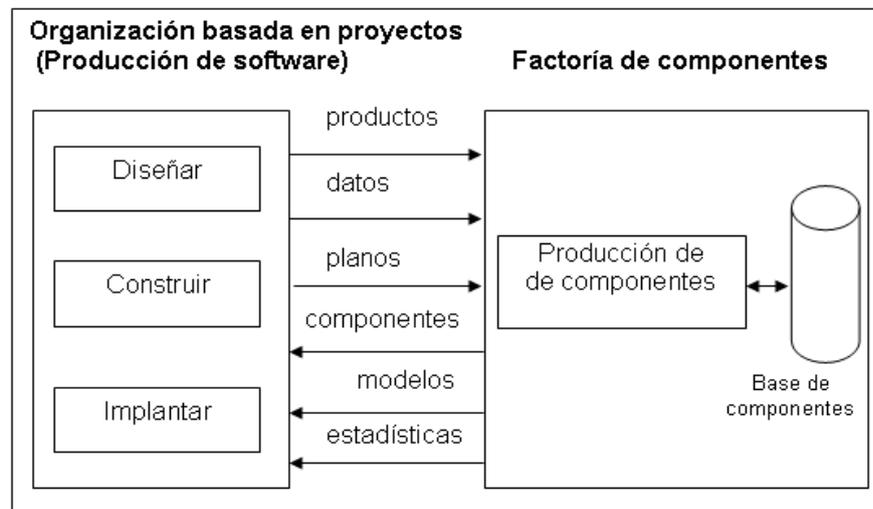


Figura 4.- Arquitectura del Modelo propuesto por Basili

“La Organización basada en proyectos realiza las solicitudes de productos (componentes para la construcción del software), de datos (estadística para la estimación de costos y plazos) y de planos (modelos, métodos para el análisis y diseño de software) a la factoría de componentes. La factoría de componentes posee una base de componentes reutilizables, de la cual se apoya para dar respuesta a las solicitudes hechas por la unidad de producción de software. En respuesta a la solicitud la organización basada en proyectos recibe los modelos y componentes para la construcción del software, además de estadísticas y datos históricos que se encuentran en la base de componentes.”(HOZ 2005).

“Comparando el presente modelo con la idea de línea de producción y montaje del proceso industrial, es posible notar también una fuerte presencia de intercambio e integración de piezas, en el caso de la línea de producción de software, esas piezas son denominadas componentes de software.” (HOZ 2005).

Este modelo se basa en la división de la factoría en dos unidades aumentando la eficiencia y especialización en la producción, y también se enfoca en la reutilización durante el desarrollo; para esto se propone tener una base de componentes reutilizables. Pero se encuentra incompleto pues solo describe la relación entre las dos subáreas y los procesos que se ejecutan entre ellas, aunque se pueden adaptar a las características de una determinada Factoría de Software, pero solo se enmarca en el área de producción dejando sin definir las demás áreas involucradas en la producción, la gestión del proyecto, la organización de la producción y de los desarrolladores, el uso de herramientas para la automatización de los procesos de construcción, soporte y gestión, entre otras.

### 1.3.5. Modelo Replicable

El modelo a describir fue concebido para ser aplicado a cualquier factoría. Es fácil de adaptar a cualquier entorno, a las necesidades y recursos de una factoría. Reúne en él las características más importantes de los modelos anteriores, es el que más detallado se encuentra en la bibliografía consultada.

“Este modelo plantea que una factoría de software debe poseer:

- Un modelo de organización de la producción.
- Una unidad de producción de componentes y una unidad de producción de software.
- Tanto la unidad de producción de componentes como la de software poseen un proceso.
- El proceso es guiado por un modelo de calidad de software.
- El proceso es compuesto de actividades que son compuestas de tareas.
- Las tareas utilizan los componentes, y estos son clasificados en infraestructura (o activos del proceso) y código.
- Las tareas usan un conjunto de herramientas para la automatización de las mismas.
- Por último el proceso puede ser aplicado al desarrollo de software o al desarrollo de un componente.” (HOZ 2005).

El modelo basa su descripción en determinar las relaciones entre conceptos a diferencia del Modelo basado en la norma ISO 9001 y CMM que lo realiza entre entidades, ver figura 5.

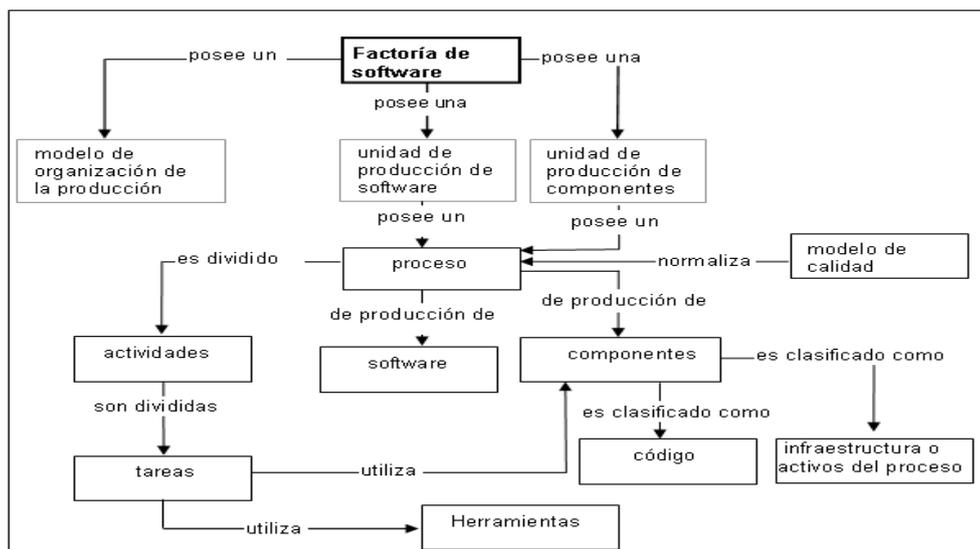


Figura 5.- Arquitectura del Modelo Replicable

En él se define la organización de la producción dividiendo la misma en cinco áreas y define las actividades que intervienen en cada una de ellas, los objetivos de los mismos y la relación que existe entre ellas. Estas son:

- Área de producción de análisis de sistema o modelado de negocio.
- Área de producción de diseño de software.
- Área de construcción de software.
- Área de producción de componentes de infraestructura o activos del proceso.
- Área de producción de componentes de código.

“Donde la unión de las áreas de análisis de sistemas, diseño de software y construcción de software forman el ámbito de negocio del modelo de producción. El ámbito de negocio incluye la interacción entre el cliente y la fábrica de software.

Las áreas de producción de componentes de infraestructura y componentes de código forman el ámbito interno del modelo, el que es transparente a los ojos del cliente de la factoría. Este ámbito es el responsable de los subproductos creados, componentes para la construcción del sistema.”(HOZ 2005).

Describe además las propuestas de roles a asignar para cada una de las actividades aunque no los organiza estructuralmente. Describe además las técnicas y herramientas a utilizar. Este modelo define mejor el proceso de producción pero no se enmarca en el uso de alguna de las metodologías estandarizadas, ni utiliza estándares de calidad, presenta el área de producción dividiéndolas como la propone Basili aportándole más actividades. Aún deja incompleto la gestión de proyecto.

### **1.3.6. Modelo de Factoría de Software aplicando Inteligencia**

“Este modelo cuenta con seis entidades, las cuales son:

- Bases tecnológicas: Comprende el contexto de las bases tecnológicas y herramientas, las técnicas y mecanismos para construir, soportar y gestionar el proceso de desarrollo.
- Proceso: Comprende el conjunto de actividades que conforman el flujo de trabajo, el cual depende de la metodología que se utilice para guiar el desarrollo del proyecto.
- Personas: Comprende el capital humano involucrado con el proceso de desarrollo de software, la estructura organizativa y los roles que ocupan, está dividida en dos áreas: Gestores de la Factoría y Grupo de desarrollo.

- Repositorio de componentes: Comprende el almacenamiento y gestión de los activos del proceso y componentes de código. Entiéndase como activos del proceso formularios, documentos, patrones, algoritmos utilizados como artefactos en el proceso. Los activos del proceso también pueden ser denominados como componentes de infraestructura, componentes de valor en el proceso.
- Gestión de la Factoría: Comprende todas las áreas de la gestión de proyecto. Presenta las áreas de proceso de gestión, organización del proceso, estructura organizacional y gestión de la calidad.
- Inteligencia: Comprende los métodos que permitan la orientación estratégica de la factoría con el uso de herramientas de Vigilancia Tecnológica, Inteligencia Empresarial, Prospectiva. Presenta dos áreas: la interna de inteligencia organizacional y la externa de inteligencia empresarial.”(CASAÑOLA 2007)

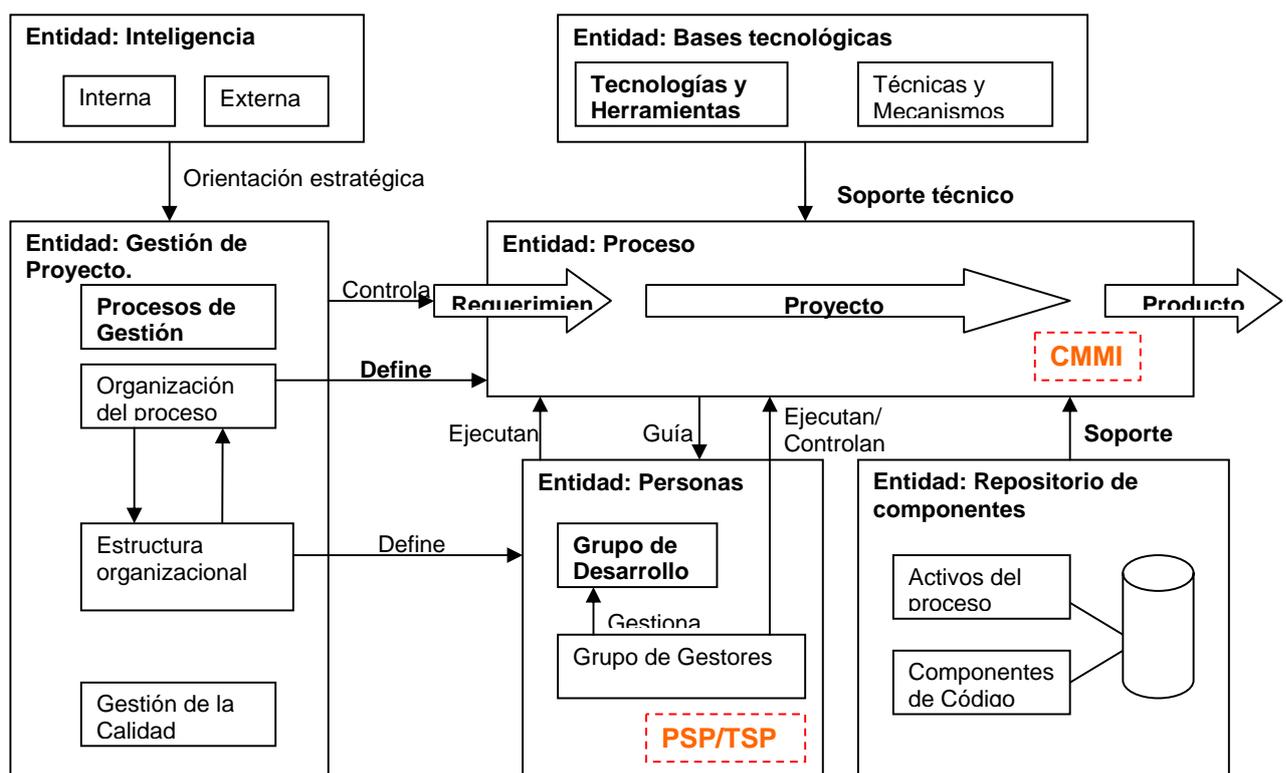


Figura 6.- Arquitectura del Modelo de Factoría aplicando Inteligencia

Como muestra la figura 6 el modelo se basa en que la entrada de un proyecto son los requerimientos y el resultado final es un producto, que toma forma durante su desarrollo gracias a la intervención de las personas representadas por la entidad Personas, utilizando PSP y TSP para la planificación personal y en equipo. El equipo de desarrollo lo forman las personas involucradas directamente en el proceso, el de gestores comprende el equipo de dirección de la misma, encargados del control y gestión del grupo de desarrollo. Los cuales son quienes ejecutan las actividades o flujos de trabajo, a su vez son guiados por el proceso de desarrollo de software, representado en el modelo mediante la entidad Proceso que es regulado por CMMI ya que éste es un modelo de calidad integrado para la industria del software que provee áreas y practicas importantes para el desarrollo y evaluación del proceso de desarrollo y la gestión de proyectos.

El proceso es automatizado y soportado por diversas tecnologías y herramientas, técnicas y mecanismos representados en la entidad Bases tecnológicas. La reutilización tiene efectos muy positivos en el desarrollo de software, entre estos efectos están el aumento en la productividad y calidad así como la reducción del tiempo de desarrollo, para dar soporte al proceso en este sentido la factoría cuenta con una base de componentes reutilizables, representada en la entidad Repositorio de Componentes. Todo esto es gestionado desde la entidad Gestión de la Factoría la que tiene la responsabilidad de definir el proceso y la estructura organizacional utilizando una metodología e ISO respectivamente.

Este modelo engloba lo indispensable en cuanto a producción de software se refiere, es el modelo ideal para la Industria Cubana del Software, pero aún no cuenta con una definición completa de la entidad proceso, pues en él se describe lo fundamental sin llegar a plantear como y mediante que herramientas se puede plantear un proceso.

### **1.3.7. Consideraciones sobre los modelos de factoría de software estudiados**

Después de estudiados los modelos, todos presentan elementos a considerar y por definir que fueron mencionados en el análisis de cada uno de ellos. Se puede concluir que el modelo más factible debe tener en cuenta que:

- La producción basada en componentes donde exista un área de producción de software y otra de componentes eleva la productividad y la eficiencia de los trabajadores, disminuye el riesgo y eleva la confiabilidad del producto obtenido.

- La entidad Repositorio de componentes permite la gestión óptima de los componentes de código y los activos del proceso.
- El uso de estándares como CMMI, PSP y TSP, ISO 15504 elevan la calidad del proceso de desarrollo y del producto.
- La definición de reglas que permitan la coordinación de cada una de las personas que intervienen en el proceso y el ensamblaje de cada uno de los componentes favorece el trabajo distribuido, eleva la productividad y permite la especialización de las personas involucradas en la producción.
- La clasificación de la factorías según el alcance da la idea del proceso que se realiza en ella y hacia donde se puede ir avanzando en el enfoque, ya que en un futuro se pudiera pasar a una factoría de mayor o menor alcance y de la relación que pudiera existir entre factorías de diferentes alcances.
- La mayoría de los modelos no enfatizan en la definición del proceso, ni definen los elementos indispensables en un proceso

Todas estas características se observan en el Modelo de Factoría Aplicando Inteligencia, modelo que se diseño específicamente para la UCI, teniendo en cuenta las ventajas y desventajas de los estudiados anteriormente, es por ello que el proceso que se definirá deberá ser aplicado en esta propuesta, para lograr la producción que se desea.

#### **1.4. Proceso de desarrollo de Software**

El término proceso de desarrollo, ha pasado a ser el factor determinante en la producción de software, puesto que define el proyecto en su totalidad. A continuación se muestran varios conceptos de proceso de desarrollo dados por diferentes autores:

Pressman expresa en 1992 que: “La capa de proceso es el fundamento de la Ingeniería de Software (IS). El proceso es la unión que mantiene juntas las capas de tecnología y que permite un desarrollo racional y oportuno de la IS. Las áreas claves del proceso forman la base del control de gestión de proyectos del software y establecen el contexto en el que se aplican los métodos técnicos, se producen resultados del trabajo, se establecen hitos, se asegura la calidad y se gestiona el cambio de manera adecuada.” (PRESSMAN 1992)

Según Pressman el proceso de desarrollo de software es fundamental en la IS, es el uso de técnicas y métodos para producir software, y mediante él se asegura la calidad con la que se obtendrá el producto final.

Fuggetta en 1995 dice que: "El proceso de software es un conjunto de personas, estructuras de organización, reglas, políticas, actividades con procedimientos, componentes de software, metodologías y herramientas utilizadas o creadas específicamente para conceptualizar, desarrollar, ofrecer un servicio, innovar y extender un producto de software."(FUGGETTA 1995)

En 1998 Isabel Ramos plantea que: "El proceso de desarrollo de software puede definirse como un conjunto de herramientas, métodos y prácticas que se emplean para producir software."(ISABEL RAMOS ROMAN 1998)

Según Fuggetta e Isabel el proceso como el conjunto de actividades como definir, crear y ofrecer un producto de software que serán realizadas por las personas involucradas, mediante un conjunto de herramientas, componentes de software, metodologías.

Jacobson en ese mismo año plantea: "El proceso de desarrollo de software es aquel en que las necesidades del usuario son traducidas en requerimientos de software, estos requerimientos transformados en diseño y el diseño implementado en código, el código es probado, documentado y certificado para su uso operativo."(I JACOBSON 1998)

Juan Carlos en su tesis en el 2003 señala que: "un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos del usuario en un sistema de software." (ROSALES 2003)

Jacobson y Juan Carlos coinciden en que el proceso de desarrollo de software es el conjunto actividades que están presentes durante la creación de un producto, desde que se reciben las necesidades del usuario hasta obtener el software final.

Según plantea Letelier en el año 2004: "No existe un proceso de software universal. Las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigen que el proceso sea configurable."(LETELIER 2004)

Ya Letelier aclara que todo proceso posee un ciclo de vida que esta constituido por un número de actividades, métodos y herramientas, pero que no se puede definir un proceso universal, ya que cada proyecto posee sus características particulares, exigiendo que el proceso sea configurable.

En el 2006 Blaya dice que: un proceso es un “Conjunto de actuaciones, decisiones, actividades y tareas que se encadenan de forma secuencial y ordenada para conseguir un resultado que satisfaga plenamente los requerimientos del cliente al que va dirigido”.(BLAYA 2006)

Blaya especifica un poco más que Letelier pues precisa que todo proceso esta constituido por un conjunto de actividades (levantamiento de requisitos, diseño de los requisitos, implementación, documentación y prueba del producto), que se realizan de forma ordenada satisfaciendo siempre las necesidades y requerimientos de los clientes finales.

Basado en las definiciones anteriores este trabajo define el proceso de desarrollo como aquel en el cual se toman las necesidades o requisitos que persigue el cliente, y se definen a partir de una metodología diferentes actividades, tareas, roles y herramientas que serán empleados para construir el producto final. Para ello se debe tener en cuenta que el proceso sea flexible a las condiciones existentes permitiendo obtener un producto eficiente, seguro y al menor coste posible.

### **1.4.1. Mapa de Proceso**

Un mecanismo de gran utilidad para la representación gráfica de los procesos de trabajo es el mapa de proceso, el cual contribuye a hacer visible el trabajo que se lleva a cabo en una unidad de una forma distinta a la que ordinariamente se conoce. A través de este tipo de gráfica se pueden observar fácilmente las tareas o pasos que a menudo pasan desapercibidos, y que sin embargo, afectan positiva o negativamente el resultado final del trabajo.

Este tipo de mapa sirve de gran ayuda para toda organización productora de software pues según Cordero “los mapas de procesos son útiles para:

- Conocer cómo se llevan a cabo los trabajos actualmente
- Analizar los pasos del proceso para reducir el ciclo de tiempo o aumentar la calidad
- Utilizar el proceso actual como punto de partida para llevar a cabo proyectos de mejoramiento del proceso
- Orientar a nuevos empleados
- Desarrollar formas alternas de realizar el trabajo en momentos críticos
- Evaluar, establecer o fortalecer los indicadores o medidas de resultados”(CORDERO 2007)

Un mapa de procesos es la representación gráfica de uno o más procesos que contribuyen de forma significativa al logro de un resultado. La simbología utilizada en la confección del Mapa de proceso debe

ser solo la de rectángulos y flechas direccionales (ver anexo 2), pues con ello se facilita comprender y entender, a simple vista, el contenido de un proceso.

De acuerdo con Rodríguez "el Mapa de proceso debe ser compartido por todos los miembros de la organización. Esto compromete a todos los integrantes de la entidad con el cumplimiento de la razón de ser de la misma."(RODRIGUEZ 2006 )

El mapa de proceso es un artefacto que se debe confeccionar antes de comenzar a definir cualquier tipo de proceso (ya sea de desarrollo de software o no) pues define y representa claramente cada subproceso que se llevará a cabo durante todo el ciclo de vida del proyecto y se mantendrá en constante actualización hasta la culminación del proceso de desarrollo.

## **1.5. Gestión de Proceso**

Actualmente, las organizaciones, independientemente de su tamaño, han de hacer frente a mercados competitivos en los que han de conseguir la satisfacción de sus clientes con la eficiencia económica de sus actividades, un proceso de desarrollo eficiente minimiza las fallas que se puedan presentar. La tecnología continúa avanzando y actualizándose cada día, por lo que la producción debe estar monitorizada por la Gestión de Proceso que es la encargada de percibir la organización como un sistema interrelacionado de procesos que contribuyen conjuntamente a incrementar la satisfacción del cliente.

En el año 2005 Adolfo expone que: "La Gestión por Procesos es un sistema de trabajo enfocado a perseguir la mejora continua del funcionamiento de las actividades de una organización, mediante: la identificación, la selección, la descripción, la documentación y la mejora de los procesos. Todas las actividades o secuencias de actividades que se desarrollan en el Servicio constituyen un proceso, y como tal hay que gestionarlas".(GARCIA 2005)

Adolfo plantea que la Gestión por Proceso esta orientada fundamentalmente a la mejora continua del proceso con el objetivo de mantener la calidad del mismo, ya dirigiendo la gestión directamente hacia todas las actividades que se deben desarrollar para conseguir un proceso con calidad.

En el año 2006 Blaya señala que: "La Gestión de Procesos determina qué procesos necesitan ser mejorados o rediseñados, establece prioridades y provee de un contexto para iniciar y mantener planes de mejora que permitan alcanzar objetivos establecidos. Hace posible la comprensión del modo en que están configurados los procesos de negocio, de sus fortalezas y debilidades."(BLAYA 2006)

Oscar Barrios plantea en ese mismo año que: “La Gestión es el proceso mediante el cual el directivo o equipo directivo determinan las acciones a seguir (planificación), según los objetivos institucionales, necesidades detectadas, cambios deseados, nuevas acciones solicitadas, implementación de cambios necesarios, y la forma como se realizarán estas acciones (estrategias, acción) y los resultados que se lograrán.”(RIOS 2006)

Según Blaya y Oscar Barrios la Gestión de Proceso define los pasos a seguir mediante la ejecución de un proyecto, verifica que si se necesita mejorar o rediseñar alguna acción, estableciendo un proceso de mejora constante de acuerdo a las necesidades de los cliente y a los resultados que se desean obtener.

En el año 2006 el consultor José Manuel Jiménez plantea que: “La Gestión de Procesos percibe la organización como un sistema interrelacionado de procesos que contribuyen conjuntamente a incrementar la satisfacción del cliente.”(VALENTIN 2006)

Según un artículo publicado en la biblioteca digital “BibliotecaNet” en el 2006: “La Gestión por Procesos es la forma de gestionar toda la organización basándose en los Procesos, entendiendo estos como una secuencia de actividades orientadas a generar un valor añadido sobre una entrada para conseguir un resultado, y una salida que a su vez satisfaga los requerimientos del Cliente.”(BIBLIOTECANET 2006)

Manuel Jiménez concuerda con el artículo de la BibliotecaNet en que la gestión de proceso está basada en la mejora de los procesos de la organización, ya que los procesos son un conjunto de actividades fuertemente relacionadas que contribuyen a satisfacer las necesidades o requerimientos del cliente final.

Ya en el 2007 Payán concuerda con que: “En forma general se define la Gestión por Procesos como el despliegue y desarrollo de las funciones administrativas de Planear, Organizar, Dirigir Ejecutar y Controlar los Procesos Organizacionales y Funcionales de una Empresa en busca de su crecimiento y sostenibilidad en concordancia con el direccionamiento establecido (misión, visión, valores, principios, políticas y objetivos corporativos).”(PAYAN 2007)

Payán ya abarca un poco más pues define la Gestión por Proceso como un conjunto de procesos o subprocesos de mejora (planificación, organizar el proceso, buena dirección, control) que deben de estar fuertemente relacionados con la mision, los principios y objetivos de la organización para mantener o conseguir la calidad que precisa el desarrollo para así obtener como resultado un proceso eficiente y con calidad.

Por todas las definiciones anteriores este trabajo define la Gestión de Proceso como un conjunto de actividades dentro de las que destacan la planificación, el control, la verificación, la mejora continua y

documentación de todos los procesos, todo esto para lograr la calidad y mejora continua del proceso de desarrollo.

## 1.6. Metodologías de desarrollo de Software

Como plantea Barzanallana "las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software."(BARZANALLANA 2007)

Las metodologías de desarrollo de software surgieron a raíz de la necesidad de controlar y documentar proyectos cada vez más complejos, impulsadas principalmente por instituciones económicamente importantes y con requisitos de seguridad y fiabilidad en sus sistemas sumamente estrictos.

Las metodologías actuales suponen un enfoque integral del problema. En particular son fundamentales en la reducción de costos y plazos, así como la calidad del producto final. Estas tecnologías constituyen la denominada "Ingeniería del Software", que se puede definir como el tratamiento sistemático de todas las fases del ciclo de vida del software.

Actualmente las metodologías se clasifican en metodologías Fuerte o Establecidas y Ligeras o Ágiles. Las primeras no se han caracterizado por su popularidad. Las segundas reaccionan a la burocracia.

Según Jorge Fernández, "las metodologías ágiles, tienen como común denominador un modelo de desarrollo incremental para producir tempranamente pequeñas entregas en ciclos rápidos, y predisposición para el cambio y la adaptación continua; según sea la conformidad o no de lo producido, y las modificaciones propuestas por los usuarios. Estas metodologías por lo general se centran en desarrollar productos funcionales más que en conseguir una buena documentación."(GONZALEZ 2006)

Aunque esto es general para todas las metodologías ágiles, cada una de ellas posee características propias.

Las metodologías tradicionales según Labrin "imponen una disciplina de trabajo sobre el proceso de desarrollo de software, con el objetivo de asegurar que el software que se obtenga satisfaga los requerimientos del usuario y reúna estándares aceptables de calidad. El trabajo de planificación es riguroso, aún cuando en la práctica muchas veces estas planificaciones no se respetan."(LABRIN 2005)

Estas propuestas pueden ser efectivas y necesarias en un gran número de proyectos, pero también pueden presentar problemas en otros muchos.

En el desarrollo de software plantea Zarzuela que “no se debe imponer un estándar a partir de algún método particular, ni definir un equipo completo de métodos aceptables. Dado que no existe una metodología universal para hacer frente con éxito a cualquier proyecto de desarrollo de software.”(ZARZUELA 2006)

Toda metodología debe ser adaptada al contexto del proyecto (recursos técnicos y humanos, tiempo de desarrollo, tipo de sistema, etc.). Se considera que a partir de las propuestas cada proyecto debe examinar sus necesidades, elegir un método, así como definir y justificar la selección dentro de la definición de los requerimientos del software.

## **1.7. Modelos y estándares de calidad**

“Los Estándares de Calidad son aquellos que permiten definir un conjunto de criterios de desarrollo que guían la forma en que se aplica la Ingeniería del Software.” (SCALONE 2006)

En este epígrafe se presenta una selección de los estándares de calidad más representativos encontrados en la bibliografía consultada, los cuales por sus características y ventajas son aplicables al proceso. Los mismos son reconocidos mundialmente por diferentes empresas y entidades que lo han adoptado. Se procede a continuación a analizar diferentes estándares que han impactado internacionalmente.

### **1.7.1. Modelo de Madurez de Capacidades Integrado (CMMI)**

El modelo de madurez de capacidades Integrado (CMMI) fue concebido para determinar y mejorar la capacidad de los procesos de las organizaciones, al punto de que desarrollen productos de calidad de manera consistente y predecible.

Según plantea Blando “en diciembre del 2000, el Instituto de Ingeniería de Software (SEI) publicó el CMMI, con el objetivo de realizar algunas mejoras respecto al Modelo de Madurez de Capacidades de Software (SW-CMM) e integrarlo con el Modelo de Madurez de Capacidades en la Ingeniería de Sistemas (SE-CMM) y el Modelo de Madurez de Capacidades para el Desarrollo Integrado de productos (IPD-CMM), que pasaban a ser considerados como obsoletos.”(BLANDO 2005)

En la Enciclopedia Libre (Wikipedia) se resalta que “el modelo CMMI contiene dos representaciones:

- Continua: Establece un conjunto predefinido de áreas de proceso para establecer la ruta para la mejora en la organización descrita como niveles de madurez.
- Escalonada: Permite a la organización seleccionar un conjunto de áreas de proceso específicas y hacer mejoras en relación con los niveles de capacidad.”(WIKIPEDIA 2007)

Estas representaciones son equivalentes, y cada organización puede optar por adoptar la que se adapte a sus características y prioridades de mejora.

### **Representación Continua**

En esta visión o “representación” la organización mostrará la representación de nivel de capacidad de cada una de las áreas de proceso del modelo.

Para la representación continua existen 6 niveles de capacidad de los procesos, estos niveles definidos por CMMI para medir la capacidad de los procesos según plantea Carmelo son:

- “0.- *Incompleto*: El proceso no se realiza, o no se consiguen sus objetivos.
- 1.- *Ejecutado*: El proceso se ejecuta y se logra su objetivo.
- 2.- *Gestionado*: Además de ejecutarse, el proceso se planifica, se revisa y se evalúa para comprobar que cumple los requisitos.
- 3.- *Definido*: Además de ser un proceso gestionado se ajusta a la política de procesos que existe en la organización, alineada con las directivas de la empresa.
- 4.- *Cuantitativamente gestionado*: Además de ser un proceso definido se controla utilizando técnicas cuantitativas.
- 5.- *Optimizado*: Además de ser un proceso cuantitativamente gestionado, de forma sistemática se revisa y modifica o cambia para adaptarlo a los objetivos del negocio. “(PEREZ 2004)

### **Representación Escalonada**

Esta representación utiliza niveles de madurez, los cuales se aplican a la madurez de la organización en su conjunto. Hay 5 niveles de madurez numerados de 1 a 5. Cada nivel de madurez abarca un conjunto de áreas de proceso predefinidas.

Para el CMMI representación escalonada los componentes según el “CMMI Club de Programadores” son:

- “Cinco Niveles de Madurez, que soportan y guían el proceso de mejora
- Cuatro Áreas de Procesos, que agrupan los procesos claves indicando cuales áreas deben ser implementadas para alcanzar cada nivel de madurez.

- Dentro de cada área, son identificados los objetivos y prácticas específicos del área, en total son 25 Prácticas Claves.
- Dentro de cada área, se utilizan cuatro aspectos para organizar los objetivos y prácticas genéricas.”(PROCESIX 2004)

### **1.7.1.1. Método de Evaluación CMMI (SCAMPI)**

El SEI ha desarrollado también un nuevo método de evaluación de las organizaciones según CMMI denominado Método Estándar de Evaluación de CMMI para Mejora de Procesos (SCAMPI). Este método es aplicable a un rango amplio de modos de uso de la evaluación, incluyendo la mejora del proceso interno y la determinación de la capacidad externa. Generalmente el grupo evaluador está conformado por integrantes de la misma empresa que es evaluada.

Carmelo aclara que SCAMPI permite:

- “Comprender mejor el nivel de competencia en ingeniería de una organización, identificando los puntos fuertes y débiles de sus procesos actuales.
- Relacionar esos puntos fuertes y débiles con el modelo CMMI.
- Priorizar planes de mejora.
- Centrarse en las mejoras más importantes que haya que acometer según el nivel de madurez de la organización y de los recursos de que disponga.
- Obtener para la organización su clasificación en uno de los niveles del modelo.
- Identificar riesgos de desarrollo y adquisición relativos a las limitaciones de la organización.”(PEREZ 2004)

Utilizar en toda organización el modelo CMMI y método de evaluación SCAMPI constituye una gran ventaja para el proceso de desarrollo además el uso de éstos puede dar soporte a la gestión de las evaluaciones de la norma ISO/IEC 15504.

### 1.7.2. ISO (International Standard Organization) (Organización Internacional De Normalización)

La Organización Internacional para la Estandarización (ISO) produce normas internacionales industriales y comerciales.

#### ISO/IEC 15504

En enero de 1993 la comisión ISO/IEC aprobó un programa de trabajo para el desarrollo de un modelo que fuera la base de un futuro estándar internacional para la evaluación de los procesos del ciclo de vida del software. Este trabajo recibió el nombre de proyecto SPICE (Software Process Improvement and Capability dEtermination). Es un emergente estándar internacional de evaluación y determinación de la capacidad y mejora continua de procesos de ingeniería del software, con la filosofía de desarrollar un conjunto de medidas de capacidad estructuradas para todos los procesos del ciclo de vida y para todos los participantes.

Según Juan Palacios el ISO/IEC 15504 comprende:

- “Evaluación de procesos
- Mejora de procesos
- Determinación de capacidad
- Alineación con ISO/IEC 12207 Information Technology Software Life Cycle Processes
- Compatibilidad y es equivalencia con el modelo CMMI.”(PALACIO 2004)

Este modelo tiene una arquitectura basada en dos dimensiones:

**Dimensión de proceso:** Caracterizada por las declaraciones del propósito de un proceso, que son objetivos esenciales mensurables de un proceso.

**Dimensión de capacidad de proceso:** Caracterizada por una serie de atributos de proceso, aplicables a cualquier proceso, que representan características mensurables necesarias para gestionar un proceso y mejorar su capacidad.

El desarrollo de ISO 15504 ha proporcionado al mundo de la informática otro estándar de evaluación de procesos que en los últimos tiempos ha sido uno de los más probados a nivel mundial.

### 1.7.3. Comparación entre ISO 15504 y CMM

Claramente hay una correlación fuerte entre ISO 15504 y el CMMI, a continuación a modo de resumen, se presenta un cuadro comparativo con las principales características de cada modelo realizado por Manuel de la Villa en (VILLA 2005)

Tabla 1. Comparación entre CMMI e ISO 15504.

	<b>CMMI</b>	<b>ISO 15504</b>
Ámbito de aplicación	Software y Sistemas	Software y Sistemas
En su favor	El de mayor prestigio	Más consensuado y probado
En su contra	Difícil de entender, mayor inversión, prescriptivo	Difícil en capacidad, complejo para evaluar
Procesos	Estructura propia	Delega en ISO 12207, por mayor aplicabilidad
Validación	Encuestas satisfacción y casos de estudio	'Trials' y esfuerzo empírico
Objetivo	Mejora del proceso, determinación capacidad contratista	Valoración del proceso y guía para la mejora.
Representación	Continua y por etapas	Continua (por etapas a nivel de proceso)
Técnicas análisis	Cuestionarios de evaluación	Varios
Método para mejora de procesos	IDEAL, mapa guiado	SPICE 4ª Parte

### 1.8. Experiencia Internacional.

A lo largo de esta investigación a escala mundial se observa cómo se han desarrollado varias empresas aplicando el modelo de factoría. Puede parecer para muchos una forma novedosa de producir software o cualquier otro producto pero la verdad es que la primera experiencia que se conoce data del año 1969 en Japón. A continuación se pretende dar a conocer un resumen de cómo marcha la aplicación de este modelo.

“La primera compañía en el mundo que nombró una organización del software como una factoría fue Hitachi en 1969, la segunda fue implantada entre los años 1975-1976 bajo el nombre de Corporación de Desarrollo de Sistemas, por un líder americano en el campo del software personalizado.”(IVAN AAEN 1997).

Según Cusumano “una estampida de factorías de software siguió después en particular en Japón donde se cuenta con los ejemplos como: NEC, Toshiba y Fujitsu que lanzaron sus propios esfuerzos de factoría entre 1976-1977”.(CUSUMANO 1989.)

De acuerdo con el Dr. Yoshihiro Matsumoto quien fue creador de la Toshiba Software Factory “en 1977, con una plantilla de aproximadamente 2 300 desarrolladores de software y trabajadores. La factoría se especializó en la producción de sistemas de software de aplicación industrial en tiempo real para los dominios como: generación de energía eléctrica, transmisión y distribución de energía, fundición de acero, automatización de la factoría, la robótica, el control de tráfico, etc. (MATSUMOTO 2002)

En Filipinas opera una factoría de software internacional llamada “GlueCode Software”, además opera en los EE.UU. Esta compañía proporciona “contratos de software para los servicios manufacturados industriales que son utilizados por empresas u organizaciones y compañías de productos de software, en ventas de tecnología a empresas importantes.”

Otra muestra de compañías que aplican el modelo de factoría, lo tenemos en la Aberdeen Software, quien según una publicación que vio la luz en el mes de julio del año 2006, apunta “que desarrollan aplicaciones de software que usan las mejores prácticas de la ingeniería como el diseño orientado a objetos, modelado de UML y Arquitectura Orientada a Servicios (SOA), para proporcionar soluciones de calidad altas.”(MARTINEZ 2006). Trabajan y colaboran con sus clientes durante todo el ciclo de vida del proyecto. Para este propósito se crea un equipo de profesionales adaptados a las necesidades y realidades de cada cliente. Hacen uso intensivo de softwares que diseñan las herramientas para la dirección del proyecto, en el control de versiones, para aumentar la productividad y la calidad del producto final.

El “Grupo Gesfor” es una multinacional de capital 100% español, fundada en 1985 y dedicada a la consultoría y servicios en el sector de las Tecnologías de la Información. “El grupo cuenta con una plantilla de 1.250 empleados en todo el mundo y, en 2005, realizó un volumen de negocio de 52,4 millones de €.”(BLANCO 2006). Las necesidades que llevaron a la adopción del enfoque de Factoría de software por esta multinacional son las siguientes: crear líneas de negocio con mayor proyección de crecimiento para la compañía, para así tener mayor crecimiento económico; crear productos de alta productividad y calidad;

lograr una mayor flexibilidad, para así aportar soluciones personalizadas a cada cliente, respetando las aplicaciones existentes e integrándolas a los nuevos cambios estratégicos; lograr una mayor productividad adelantándose a las necesidades de sus clientes, logrando de esta forma un mayor mercado e Innovación, creando un valor en el servicio al cliente y potenciando la actividad de investigación más desarrollo, más innovación (I+D+i); y para lograr una Gestión del cambio, adaptándose e implantando nuevas formas de hacer con mayor calidad y en el tiempo requerido.

Matchmind (MM) es una empresa dedicada a servicios profesionales de consultoría de gestión y tecnología de la información. La cual tiene como base de producción factorías de software distribuidas. Según unas publicaciones "Ya están operativas tres factorías de software de Matchmind, en Madrid, Ávila y Sevilla, con un total de 180 profesionales. En abril entrará en funcionamiento otra más en Segovia, donde actualmente están reclutando al equipo humano que trabajará en él. Los planes que maneja la compañía es abrir en tres años un total de diez centros similares repartidos por toda España: dos más en 2006, otros dos en 2007 y otro par de instalaciones adicionales en 2008."(VALENTIN 2006).

MM ha triplicado el número de puestos de trabajos, duplicado el número de clientes, ha supuesto un crecimiento por encima del 40 % y tiene ingresos de 50 millones de euros.

ENDESA es un ejemplo de cliente que utiliza los servicios de diferentes factorías. Realiza contratos utilizando el modelo de Factoría de Software de la Empresa, estableciendo alianzas con cuatro proveedores líderes en esta industria, cada uno con una especialización. Estas compañías se encargarán de diseñar y desarrollar el software que ENDESA requiera para sus diferentes áreas de negocio: Accenture, para los Sistemas Comerciales; Capgemini para los Sistemas de Distribución; Indra, para los Sistemas de Generación y Mercado Eléctrico; e IBM para los Sistemas de Gestión Interna. Los contratos, se firmarán por un período de cinco años para cada una de las cuatro Factorías de Software, que integrarán tanto el desarrollo como el mantenimiento de los sistemas. Esto permitirá un ahorro de costos cercano al 25%, lo que significa más de 52 millones de euros en el período establecido, lo que representa un ahorro anual de 10,4 millones de euros.

Accenture es una compañía global de consultoría de gestión, servicios tecnológicos y outsourcing. Los centros de desarrollo de Accenture, repartidos por todo el mundo, son pioneros en la industrialización del software. El Centro de Desarrollo de CORITEL pertenece al Grupo Accenture que cuenta con dos ubicaciones, una en Málaga y otra en Madrid él ha alcanzado el nivel de madurez 5 en la certificación

CMMI, primera empresa española en alcanzar esta categoría, tras el obtenido en Octubre de 2003 como CMM 5.

El Centro de Desarrollo dispone de tres líneas de producción diferenciadas, una especializada en el sector de las soluciones catalogadas como ERP (Enterprise Resource Planning) y CRM (Customer Relationship Management), donde se incluyen los mayores fabricantes de software de gestión para empresas. La segunda línea es la especializada en aplicaciones de Internet -con las tecnologías Java y Microsoft más avanzadas- y la utilización de herramientas de integración de aplicaciones. La tercera línea, por último, tiene que ver con el desarrollo de aplicaciones de gestión para grandes sistemas mainframe en entornos críticos.

Otros ejemplos es la Factoría de software Kingston, Calgary software factory, CERTANT, Argentina Software Factory, Vates S.A.

El concepto de factoría de software es un modelo de servicios apoyado en una estrategia metodológica que ha madurado por más de veinte años en el mundo de la informática. Esta forma de trabajo, que es una analogía de los procesos de producción industriales, permite reducir considerablemente la incertidumbre en el desarrollo y mantenimiento de proyectos de software y ofrece una serie de beneficios directos sobre la confiabilidad y nivel de satisfacción de los productos entregados, proporcionando una mayor claridad presupuestaria y un calendario acotado de los proyectos e incidencias asociadas a las aplicaciones de software de las empresas. Se aprecia que su implantación ha reportado beneficios entre los que sobresalen:

- Proceso de desarrollo de acuerdo a las necesidades del cliente, enfocado principalmente a la calidad y la mejora continua.
- Especialización: creación de grupos especializados adaptados a las necesidades del momento.
- Flexibilidad: adaptación de la factoría a cada cliente.
- Disponibilidad de librerías de componentes para la construcción rápida de soluciones.
- Costos y tiempos de desarrollo ajustados a las estimaciones.
- Compromiso de calidad.

En la aplicación de este modelo por las distintas empresas que se han visto parece que hay un denominador común y es la misión que tiene cada una de ellas: Proporcionar soluciones software de la tecnología más avanzada, adaptadas a los cambios y avances tecnológicos, siempre con una inmejorable relación calidad / precio. Proporcionar a los clientes soluciones flexibles y adaptables a las necesidades

específicas de cada proceso de negocio. Implantar metodologías para optimizar la producción de componentes en volumen y asegurar altos niveles de calidad con resultados óptimos en rendimiento y plazos. Garantizar la máxima calidad de los productos a través de una rigurosa metodología de trabajo y de herramientas que permiten la Gestión Auditable y Colaborativa de Proyectos.

## **1.9. Conclusión**

En este capítulo se han definido conceptos fundamentales a conocer para la investigación; y ejemplos de experiencias de la aplicación del enfoque de factoría en el mundo , después de analizar las definiciones abordadas fue posible percatarse de que las Factorías de Software son las encargadas de industrializar el desarrollo de software, y que el Proceso de Desarrollo de Software es el fundamento de la Ingeniería de Software, es decir es la entidad que define una producción eficaz y eficiente basada en los requisitos del cliente.

La selección de los modelos descritos anteriormente responde a que se han encontrado en los mismos un grupo de características fundamentales que debe tener una factoría según las definiciones de factoría de software y el proceso se selecciona a partir de las características y necesidades de la producción de software sin destacar ningún proyecto en específico.

Después de analizar algunos estándares de calidad como el CMMI y la norma ISO 15504, se llega a la conclusión de que para cumplir las expectativas de un proceso con un alto nivel de calidad, se debe implementar el estándar de calidad CMMI porque se enfoca no solo al proceso y a la evaluación de este, sino, que además se enfatiza en la mejora continua del proceso, todo lo anterior siguiendo los pasos de la gestión de proceso, la aplicación de este estándar se llevará a cabo mediante la representación continua que se aborda anteriormente en este capítulo, utilizando el método de evaluación SCAMPI para conocer que se ha hecho y en que estado se encuentra el proceso

Por todo lo anterior se evidencia la necesidad de crear un modelo y un proceso para definir y establecer un proceso de desarrollo en cada uno de los proyectos para asegurar el éxito del modelo de factoría aplicando inteligencia, modelo que contiene agrupados todos los factores positivos de los que le anteceden y que además, fue definido a partir de la experiencia y necesidades de la producción de software cubana.

Este capítulo en general estudia todos los conceptos relacionados con el campo de acción y el objeto de estudio para así, sentar las bases que definen el objetivo que persigue esta investigación.

## Capítulo 2: Definición de la Entidad Proceso

### 2.1. Introducción

En este capítulo se describe la Universidad de Ciencias Informáticas (UCI), los diferentes métodos y procedimientos que se utilizaron para llevar a cabo la investigación, así como la descripción de la Entidad Proceso que se propone utilizar en el Modelo de Factoría de Software aplicando Inteligencia.

### 2.2. Universidad de las Ciencias Informáticas (UCI).

La universidad genera conocimiento pero es muy difícil que lo aplique, por lo que se aleja de los problemas de la producción conllevando a que las investigaciones no respondan a las necesidades de las empresas. Por otra parte a las empresas les es muy difícil investigar pues tienen que depender de un mercado cada vez más exigente en tiempo, costo y calidad.

La búsqueda de soluciones ha encontrado la solución en la vinculación Universidad-Empresa, esta es una alianza estratégica de intercambio donde la primera obtiene la facilidad de aplicar sus investigaciones y de vincular sus estudiantes y profesores al mundo empresarial y de funcionar como una entidad empresarial; la segunda recibe el conocimiento y la innovación constante que generan las universidades. Sobre la base de este nuevo concepto de universidad productiva, la universidad ha logrado una fuerte vinculación Estudio-Trabajo y Universidad-Industria.

En la UCI la producción es un problema social, político, y económico, el cien por ciento de los estudiantes y profesores se deben vincular a la producción participando en proyectos de alto valor tanto para el mercado nacional, como internacional, se plasma la concepción de que la docencia se realice desde la producción y que los estudiantes deben estar vinculados desde los primeros años a proyectos productivos.

La UCI tiene como misión:

- Formar profesionales, comprometidos con su Patria, altamente calificados en la rama de la informática.

- Producir software y servicios informáticos, a partir de la vinculación estudio-trabajo como modelo de formación.

La Universidad presenta una fuerte formación Docente – Productiva: Las facultades se especializan según perfiles y se vinculan a proyectos productivos. Aprendizaje desde la producción: Planes de estudio diseñados de manera flexible y con posibilidades de cambio. Uso creciente de la Teleformación: Se utilizan teleclases y se montan los contenidos en sitios Web lo que permite el autoaprendizaje.

Este nuevo concepto proporciona grandes perspectivas para el desarrollo de la Industria Cubana del Software, y su impacto se siente ya diferentes sectores de la sociedad.

Para organizar este proceso de formación – investigación – producción se creó en la UCI una Infraestructura Productiva (IP) que dirige los proyectos de toda la universidad subordinada a la vicerrectora primera. La IP presenta una dirección de calidad que es la responsable de verificar si los productos pueden ser entregados al cliente o debe perfeccionarse para que cumpla con los requisitos de calidad establecidos a su inicio.

Los proyectos productivos de cada facultad se organizan y se llevan a cabo en la misma bajo la supervisión del decano y el vicedecano de producción. Los líderes de proyectos son designados por la facultad, los demás miembros son los profesores y estudiantes que se captan según las necesidades que exige el proyecto, además participan especialistas de diferentes empresas, Centro de Investigación y otras universidades del país. Las herramientas, metodologías y los roles que va desempeñar cada miembro del equipo lo decidirá el líder del proyecto conjuntamente con el vicedecano de producción.

Los proyectos nacionales que acomete la universidad resultan de demandas conciliadas con los Organismos de la Administración Central del Estado (OACE). La exportación de Software y brindar servicios informáticos es el objetivo central de la UCI, actividad que hace algún tiempo realiza con varios sectores de la sociedad y con varios países.

En ella existen muchos proyectos en los que el desarrollo de software no se realiza de la mejor manera, presentándose deficiencias en la definición de los flujos de procesos, roles y responsabilidades, atrasándose la entrega de los productos, disminuyendo la calidad, y la eficiencia del mismo. La planificación del trabajo no es la mejor, en la mayoría de los casos no se siguen estándares y metodologías establecidos en la Ingeniería de Software. Esto empeorará con el aumento de la fuerza de trabajo y de la demanda del cliente, ocasionando que el desarrollador se sienta desorientado y no sepa

qué hacer en cada momento ni a quién dirigirse, llevando a la desorganización de la producción afectando la productividad.

### **2.3. Métodos, y procedimientos utilizados.**

El método científico de investigación es la forma de abordar la realidad, de estudiar la naturaleza, la sociedad y el pensamiento, con el propósito de descubrir su esencia y sus relaciones.(LEON 2002)

En la investigación como métodos de investigación se han utilizado los Métodos teóricos: Histórico lógico, Hipotético deductivo, Sistémico.

Los métodos teóricos permiten comprender el fenómeno que se estudia, su evolución, elaborar la hipótesis y proponer las mejoras a los problemas que se identificaron.

En la investigación se planteó el problema como un todo, se enfocaron los principales problemas existentes en la producción de software, el enfoque de factoría y sus modelos desde un enfoque histórico lógico, en la primera parte de esta investigación se desarrolló un estudio del estado del arte del enfoque y de los procesos de desarrollo de software; se analizaron los aspectos positivos y negativos de cada uno de los modelos ya existentes, y las tendencias en la resolución de esta problemática.

Es importante destacar que la revisión bibliográfica constituyó un método importante para la concepción del proyecto de investigación y para tomar los conocimientos relacionados con la temática.

La investigación sigue además un método hipotético deductivo ya que a partir del problema se plantearon los objetivos específicos y la hipótesis que en el transcurso de la investigación son resueltas siguiendo métodos fundamentados.

Los métodos empíricos permiten describir y explicar las características del fenómeno en estudio.

Los métodos particulares dentro de los empíricos se utilizaron con el propósito de recoger los datos necesarios para identificar la problemática e identificar las causas de las mismas, así como determinar la magnitud de su influencia.

Las encuestas y las entrevistas fueron técnicas muy valiosas para obtener claridad de la situación en que se encuentra la organización con respecto a la producción, y para establecer los flujos de trabajo necesarios para la lógica del modelo, medir el alcance y la importancia que posee el tema. Captar la información cualitativa y cuantitativa del fenómeno, conocer los criterios sobre la forma en que se organiza y se lleva a cabo la producción de software en la UCI, así como las posibles soluciones que se proponen

en la investigación, para ello se entrevistaron y encuestaron personas involucradas en la producción de software en la UCI y fuera de ella, con cierto grado científico.

### **2.3.1. Entrevista # 1**

Esta actividad la población a estudiar fue el personal involucrado desde distintos puntos de vistas con la producción de la UCI, como por ejemplo: directivos institucionales, o de las organizaciones, desarrolladores, profesores de cada una de las especialidades y a otros de que de una forma u otra se relacionan con este proceso. La unidad de estudio el proceso de producción en los proyectos de producción de software de la UCI.

Los proyectos productivos a investigar fueron seleccionados por la técnica de muestreo no probabilística, muestreo intencional para obtener la mayor representación de proyectos e información posible, de acuerdo con los intereses que se persiguen en la investigación.

Se seleccionaron para entrevistar 10 directivos institucionales, 3 vicedecanos de producción, 10 Líderes de proyectos, 10 dirigentes de las organizaciones políticas y de masas, desarrolladores de 30 proyectos de las diferentes líneas de producción de la Universidad, teniendo en cuenta la cantidad de proyectos de la universidad la muestra para la investigación es mayor del 20%.

Esta entrevista se realizó con el objetivo de identificar el grado de conocimiento de los involucrados de la situación problémica y del problema, validar la propuesta de solución, tomar elementos y opiniones valiosas a tener en cuenta en la solución. En el Anexo 3 se puede observar el diseño de la entrevista.

### **2.3.2. Entrevista # 2**

Para la realización de la segunda entrevista la población a estudiar fueron especialistas en producción de software externos a la UCI y la unidad de estudio el proceso de producción en los proyectos de producción de software. La selección se realizó con la técnica de muestreo no probabilística, muestreo intencional. Se seleccionaron personas con grado científico y que participan en eventos nacionales e internacionales que han estado vinculados a la universidad o a la industria.

Se seleccionaron 20 especialistas vinculados a la producción de software de diferentes empresas, entidades y universidades donde se desarrolla software, 10 especialistas en el uso de la inteligencia de las casas consultoras DISAIC y Biomundi.

Se realizó con el objetivo de validar los conceptos utilizados en la investigación, validar la propuesta de solución y recopilar elementos a tener en cuenta en la solución.

En los Anexos 4 y 5 se pueden ver los diseños de los modelos de la entrevista aplicada a los especialistas en la producción de software y en el uso de la inteligencia respectivamente.

### 2.3.3. Encuesta

La población a estudiar fueron los miembros de los proyectos de producción de software de la UCI y la unidad de estudio el proceso de producción en los proyectos de producción de software de la UCI. La selección se realizó con la técnica de muestreo no probabilística, muestreo intencional para poder obtener la mayor representatividad e información posible, de acuerdo con los intereses de la investigación que fue entrevistar a los líderes de proyecto, planificadores y desarrolladores de proyectos de todas las líneas de producción y de todas las facultades, en distintas fases del desarrollo.

Se seleccionaron 32 proyectos que teniendo en cuenta la cantidad de proyectos de la universidad, la muestra para la investigación es mayor del 20%.

Se realizó con el objetivo de identificar cuantitativamente los problemas que se habían identificado, el grado de conocimiento de los involucrados de la situación problemática y del problema, así como su percepción.

En la encuesta se evaluaron los indicadores de la variable del proceso de desarrollo, los mismos fueron:

Indicador	Sub-indicador
Organización del proceso y las personas	Definición de Roles y Responsabilidades
	Definición del flujo de trabajo
Gestión de proyecto	Planificación del proyecto
	Uso de PSP y TSP
	Gestión de tiempo
	Gestión de costo
	Gestión de recursos
Definición de las Bases Tecnológicas	Establecimiento de la revisión y control del proyecto
	Definición de la línea de producción
	Definición de los estándares a utilizar

	Repositorio de Componentes
	Definición de la tecnología a usar
Comunicación con el Cliente	Modelación de las funcionalidades del proyecto a desarrollar
	Aceptación de la documentación por parte del cliente

A la hora de elaborar la encuesta se combinaron los tipos de preguntas.

La mayoría fueron semicerradas pues se tiene el interés de conocer la información cuantitativa pero también de saber la opinión del tema, así como involucrar y motivar a los encuestados en la solución. Se utilizaron además preguntas cerradas, directas e indirectas y de control.

En el Anexos 6 se puede ver el modelo de la encuesta aplicada.

## 2.4. Entidad Proceso del Modelo de Factoría de Software aplicando Inteligencia.

El desarrollo de software actual está siendo orientado por estándares y metodologías que se adaptan a las condiciones de las organizaciones. Mantener bien organizado y localizado cada elemento de la metodología es una tarea muy importante. En áreas de producción pensadas para elaborar sistemas diferentes esto se vuelve una necesidad vital. Ejecutar el proceso productivo para el desarrollo del producto, es el objetivo fundamental de esta entidad.

En el desarrollo de software no se debe imponer un estándar a partir de algún método particular, ni definir un equipo completo de métodos aceptables. Dado que no existe una metodología universal para hacer frente con éxito a cualquier proyecto de desarrollo de software. Toda metodología debe ser adaptada al contexto del proyecto (recursos técnicos y humanos, tiempo de desarrollo, tipo de sistema, etc.) Se considera que a partir de las propuestas cada proyecto debe examinar sus necesidades, elegir un método, así como definir y justificar la selección dentro de la definición de los requerimientos del software.

El proceso debe estar definido por el área organización del proceso de la entidad gestión de proyecto y controlado desde esta. El líder del proyecto es el encargado de dirigir y guiar al equipo en su conjunto durante el desarrollo, controlar que cada integrante cumpla con su trabajo en el tiempo establecido además de atender sus necesidades.

Particularmente persigue ejecutar los flujos de trabajo basados en las metodologías, las cuales sintetizan ese gran proceso a los aspectos claves que pueden ser útiles en el desempeño de la factoría, así como los elementos que deben garantizar un avance estable. En la figura 7 se representan los principales elementos a definir dada una metodología y la relación entre ellos.

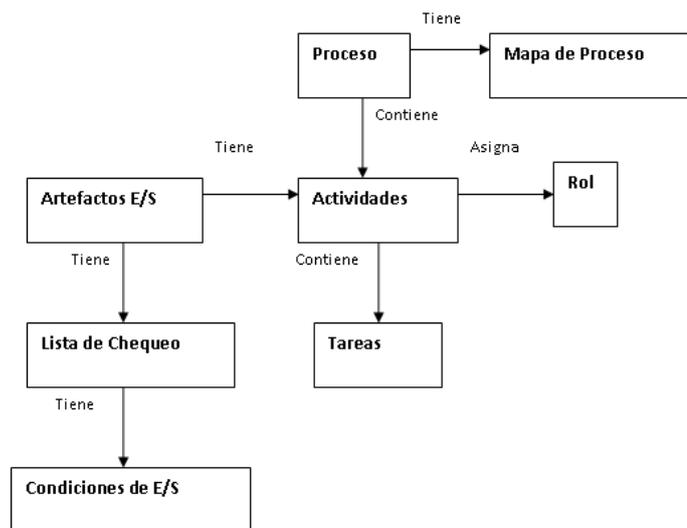


Figura 7.- Representación gráfica de los elementos del proceso.

El proceso debe tener un mapa de proceso que describa el flujo. El proceso está compuesto por actividades y estas son asignadas a un rol, además debe tener tareas para ejecutarlas y artefactos de entrada y salidas los cuales presentan una lista de chequeo y condiciones de entrada y salida. Los artefactos terminados deben cumplir las especificaciones de calidad que se le propone en la lista de chequeo y en las condiciones de entrada/salida. Mediante la cual se especifican una serie de puntos a evaluar. En todas las fases del desarrollo se deben tener en cuenta estos puntos en función de que al finalizar cada actividad haya que hacer el menor número de correcciones posibles.

La Entidad Proceso influye en la definición del resto de las entidades, definido el proceso a partir del tipo de proyecto, tiempo de duración, alcance y metodología: se identifican las personas y roles, las bases tecnológicas, las características de los elementos a almacenar en el repositorio, la gestión de proyecto que se debe establecer y las necesidades de gestión del conocimiento e inteligencia empresarial.



### 2.4.1. Descripción del proceso 201 Organizar trabajo

Con este proceso la entidad podrá planificar las tareas o actividades que enfrentará, para así poder darle seguimiento y tener constancia de aquello que no se pudo hacer y su motivo, como esta distribuido este proceso se puede observar claramente en el anexo 13.

**Definir artefactos de entrada:**

*Lista de objetivos:* lista de objetivos trazados por la entidad.

*Lista de tareas:* lista de tareas asignadas a la entidad.

**Definir artefactos de salida:**

*Informe de auditoria:* documento donde se reflejan los objetivos, el plan y los resultados de la auditoria.

*Plan de trabajo:* documento donde se reflejan la planificación de las actividades, responsables, fechas, aseguramientos y participantes.

**Definir las actividades:**

*Planificar tareas:* esta actividad persigue planificar las tareas en la entidad con la cual se buscará organizarla. Las principales tareas son

- a) Analizar los objetivos: analiza los objetivos fundamentales de la entidad y organiza actividades para su cumplimiento.
- b) Analizar de las tareas asignadas: analiza las tareas que le han sido asignadas a la entidad y organiza su ejecución.
- c) Asignar responsables: asigna las tareas a los miembros de la entidad.

*Controlar plan de trabajo:* esta actividad verifica sistemáticamente las tareas aprobadas en la planificación, permitiendo irle dando cumplimiento y tener constancia de quiénes han cumplido y quiénes no lo han podido hacer con sus respectivas razones. Las principales tareas son:

- a) Asegurar las actividades: involucra a todos las personas que deben tributar al cumplimiento de las tareas y el estado de sus responsabilidades para su aseguramiento.
- b) Chequear estado de las actividades: analiza el estado de preparación de cada actividad y la viabilidad de su éxito en el periodo que corresponde.

- c) Actualizar el plan: verifica el cumplimiento y actualiza los cambios en función de las incidencias.
- d) Auditar el sistema de trabajo: audita el sistema de trabajo dan la cobertura de revisar el sistema y la planificación de la organización, para en alguna medida poder luego establecer una evaluación cualitativa y cuantitativa de la misma.

Para mayor comprensión sobre estas actividades ver anexo 7.

### **2.4.2. Descripción del proceso 202 Definir el proceso de desarrollo de software.**

Con este proceso la unidad organización del proceso de la entidad gestión de proyecto define todo el proceso de desarrollo de software, para entender este proceso consulte el anexo 14.

#### **Definir artefactos de entrada:**

*Documento de descripción del proyecto:* Documento que refleja la descripción del negocio, los resultados esperados, la visión de la solución informática, las principales necesidades en orden de prioridad y las características básicas que desea en el producto.

*Lista de proyectos ejecutados:* es la lista de los proyectos ejecutados hasta ese momento con la descripción de cada uno de ellos, conteniendo la descripción del negocio, resultados esperado y obtenido por el cliente, objetivos, banco de problemas, riesgo, ubicación de los componentes, y tecnología utilizada.

*Descripción de la Clasificación de la Factorías:* Contiene la descripción de las clasificaciones de las factorías de acuerdo a su alcance

*Documentación de descripción de la metodología:* describe los elementos de la metodología responde a las preguntas ¿Cuándo? ¿Cómo? ¿Que? ¿Quién?

#### **Definir artefactos de salida:**

*Documento de descripción del proceso:* contiene la descripción de los flujos de actividades, las tareas, los artefactos, los roles y las listas de chequeo que se combinaban para desarrollar el proyecto.

*Plan de gestión de riesgo:* provee una visión global de los riesgos que se puedan presentar en el proyecto determina las estrategias para resolverlos.

**Definir las actividades:**

*Identificar la línea de producción:* la actividad identifica la línea de producción a partir de la visión de la solución informática, de las principales necesidades del cliente y las características básicas del producto a obtener.

- a) Solicitar documento de descripción del proyecto: solicita a los clientes la descripción del negocio, los resultados esperados, la visión de la solución informática, las principales necesidades en orden de prioridad y las características básicas que desea en el producto.
- b) Identificar línea de producción: a partir del análisis del documento de descripción del proyecto se identifica por las características de la solución y de las restricciones la línea de producción a la que se ajusta.
- c) Comparar el proyecto con otros existentes anteriormente: a partir de la información sobre los proyectos realizados anteriormente se analizan los puntos en común, buscando elementos reutilizables. Si el proyecto es similar a los existentes se analizan los resultados obtenidos en el proyecto.

*Identificar clasificación de factoría para el proyecto:* define a partir de la descripción del proyecto y la descripción de la clasificación según el alcance, el tipo de factoría que debe asumir el proyecto (factoría de implementación de software, factoría de diseño e implementación de software, factoría de modelado, diseño e implementación o factoría de ciclo completo).

- a) Analizar el producto a obtener: a partir de la descripción de la información de entrada y de los resultados a obtener en el proyecto determinar los flujos de trabajo a ejecutar para el desarrollo del producto (modelo del negocio, captura de requisitos, análisis y diseño, implementación, prueba y despliegue).
- b) Identificar clasificación de factoría para el proyecto: a partir de la descripción del proyecto y la descripción de la clasificación según el alcance se identifica el tipo de factoría que debe asumir el proyecto (factoría de implementación de software, factoría de diseño e implementación de software, factoría de modelado, diseño e implementación o factoría de ciclo completo).

*Definir el proceso:* Esta actividad determina el flujo de procesos a ejecutar para obtener el producto deseado. Las principales tareas son:

- a) Determinar la metodología: se determina que metodología de desarrollo de software se utilizará para guiar el proceso de desarrollo.
- b) Determinar las actividades: establece la secuencia de actividades necesarias para obtener el producto esperado y la descripción.
- c) Determinar las tareas: determinan los pasos a seguir dentro de cada actividad, es decir es el desglose de cada actividad en tareas más simples.
- d) Determinar artefactos: define los artefactos de entrada y salida de cada actividad.
- e) Determinar los roles: asigna a cada actividad que se realice el rol que sea el responsable de su cumplimiento.
- f) Caracterizar el rol: se define las competencias para desempeñar cada rol (habilidades, sistemas de conocimiento y valores).
- g) Estimar los recursos: se realizan las estimaciones de tiempo, costo y recursos humanos que requiere el proyecto.
- h) Organizar grupo de desarrollo: define la cantidad de personas por roles necesita para trabajar.
- i) Establecer la lista de chequeo: determina las condiciones de cada uno de los artefactos que se van generando, y contiene también las condiciones de entrada y salida de cada uno, la cual se va verificando sistemáticamente.
- j) Identificar bases tecnológicas: se determinan los mecanismos y técnicas, herramientas y tecnología necesarias para soportar, construir y gestionar el proceso.
- k) Publicar información del proceso: se le comunica a cada una de las entidades de soporte la información requerida para el aseguramiento del proyecto.

*Identificar riesgos del proceso de desarrollo de software:* Esta actividad se debe evaluar continuamente, para prever las insuficiencias en el proyecto.

- a) Priorizar riesgos que afecten al proceso: se elabora la lista de riesgos y se organizan por orden de prioridad.
- b) Establecer plan de gestión de riesgos: confecciona el plan de mitigación y de contingencia para enfrentar los posibles riesgos.

Un resumen de todas las actividades de este proceso se puede observar en el anexo 8.

### **2.4.3. Descripción del proceso 203 Controlar el proyecto**

Es el proceso mediante el cual se establece y controla la evolución del proyecto, el cumplimiento del cronograma de trabajo y el funcionamiento de equipo de proyecto con el objetivo de determinar las debilidades y fortalezas, las amenazas y oportunidades para mejorar continuamente el proceso definido y el desenvolvimiento del proyecto.(Ver anexo 15)

#### **Definir artefactos de entrada:**

*Documento de descripción del proyecto:* Documento que refleja la descripción del negocio, los resultados esperados, la visión de la solución informática, las principales necesidades en orden de prioridad y las características básicas que desea en el producto.

*Cronograma del proyecto:* Documento que recoge la hora, fecha, responsables del plan de actividades y sistema de trabajo del proyecto.

*Cronograma individual:* Documento que recoge la hora, fecha, responsables del plan de actividades y sistema de trabajo de cada uno del proyecto.

*Documentación del proyecto:* Recopilación de todos los artefactos que se han generado durante el desarrollo del proceso.

*Producto:* son todos los componentes que se van obteniendo.

*Documentación del producto:* son todos los artefactos que se van realizando desde que comienza a realizarse el producto hasta que lo obtenemos.

*Banco de problemas:* recoge todos los problemas que han existido durante el desarrollo del proyecto, y las vías posibles para solucionarlos o mejorarlos.

*Plantilla del proyecto:* Guarda todos los datos de las personas que significaron algún rol dentro del proyecto, incluyendo sus responsabilidades y cumplimiento de las mismas.

**Definir artefactos de salida:**

*Acta de entrega:* consta de la fecha, hora y lugar donde se entregó la definición de proceso, y las opiniones de cada una de las partes.

*Listado de las no conformidades:* documento que recoge todos los defectos o errores encontrados en cada una de las revisiones o controles al proyecto.

*Matriz DAFO:* documento que refleja el análisis de las debilidades, amenazas, fortalezas y oportunidades de un tema.

*Documento cierre del proyecto:* refleja si el producto está en condiciones de cierre y los elementos más significativos del proyecto a tener en cuenta en futuros proyectos entre ellos el banco de problemas y el banco de soluciones.

*Plantilla del personal que se libera:* listado de desarrolladores que no son necesarios en la etapa posterior del proyecto para llevar a cabo los procesos postventa.

*Descripción de mejora del proyecto:* listado de desarrolladores que no son necesarios en la etapa posterior del proyecto para llevar a cabo los procesos postventa.

**Definir las actividades:**

*Entregar al equipo de proyecto la definición de proceso:* Mediante esta actividad el equipo de desarrollo tendrá toda la definición de cómo debe realizar el proceso de desarrollo de software.

- a) Entregar al equipo de proyecto la descripción del proceso: mediante el cual se le hace entrega al equipo de desarrollo el documento de descripción del proceso.
- b) Discutir el flujo de actividades: en ella se analiza con el equipo de proyecto el cómo se va a cometer el proyecto durante todo su desarrollo, cada desarrollador debe identificarse con las actividades a realizar.
- c) Redefinir el proceso: a partir del análisis con el equipo de proyecto pueden surgir sugerencias las cuales deben ser evaluadas y pueden implicar cambios en la definición.

*Controlar el proyecto:* se controla la evolución del proyecto, el cumplimiento del cronograma de trabajo y el funcionamiento de equipo de proyecto.

- a) Analizar evolución del proyecto: determina si el proceso se ejecuta de la manera establecida, cumpliendo los cronogramas previstos y con la efectividad del equipo de proyecto requerida.

- b) Evaluar el proceso: se realizan inspecciones técnicas al proyecto, orientando a la documentación y evolución de los componentes.
- c) Analizar DAFO: analizar las principales fortalezas y debilidades, oportunidades y amenazas, que presenta el proyecto y toma decisiones para la mejora del proceso.

*Cerrar Proyecto:* se analiza si el proyecto cumplió los objetivos previstos y obtuvo el producto esperado por el cliente.

- a) Entregar el producto final: el equipo de proyecto debe entregar la documentación y el producto.
- b) Analizar la entrega al cliente: se evalúa si el producto cumple con los requerimientos del cliente y tienen la calidad para realizar su liberación.
- c) Verificar almacenamiento del proyecto: verifica que el repositorio de componentes contenga todos los productos o artefactos obtenidos hasta el momento, para que puedan ser reutilizable o seguir usándose en otra fase del proyecto, se almacena el producto final como componente terminado.
- d) Analizar el cierre del proyecto: se realiza un análisis del proceso desarrollado por el proyecto, de los problemas presentados.
- e) Liberar el personal: se libera el personal del proyecto que no se requiere para los servicios post venta de soporte y mantenimiento.
- f) Proponer mejoras: se evalúa la viabilidad de las mejoras que se le puede realizar al producto para reiniciar un nuevo proceso.(Ver anexo 9)

#### **2.4.4. Estructura organizacional.**

Una de las tareas para definir todo proceso es la estimación, la cual posee como una de sus funciones definir la cantidad de personas que interviene en el proceso, para este grupo de personas se define una estructura organizacional.

La estructura organizacional es definida por el área estructura organizacional de la entidad gestión de proyecto teniendo en cuenta varios factores que influyen tanto positivos como negativamente en la estructura a aplicar. La misma depende de las características de la Factoría, el número de personas que

compondrá el equipo, la preparación que posean sus integrantes, la dificultad de las tareas asignadas al mismo.

Pressman en el libro "Ingeniería de Software. Un enfoque Práctico" hace un análisis de varios organigramas para equipos de desarrollo de software, estos son: Descentralizado Democrático (DD), Descentralizado Controlado (DC) y Centralizado Controlado (CC) los cuales se pueden analizar considerando sus características.

Algunos factores que influyen significativamente en la estructura son: la dificultad del problema a resolver, el tamaño del proyecto, el tiempo de vida del equipo, el grado de complejidad del proyecto, la calidad requerida y fiabilidad del sistema que se desea construir, el grado de comunicación requerida entre otros.

La estructura centralizada puede realizar las tareas más rápidamente, pero debe ser aplicada a la solución de problemas sencillos en los cuales el equipo domina el problema a resolver, las tecnologías y herramientas a utilizar. En los inicios teniendo en cuenta el estado en que se encuentra una Factoría esta no sería la estructura más adecuada para aplicar pues no existe una basta experiencia en el uso de las tecnologías y herramientas con que se trabajan, en un futuro no muy lejano cuando el equipo esté un poco más familiarizado con las mismas puede optarse por aplicar esta estructura de equipo en el desarrollo.

Los grupos descentralizados generan más y mejores soluciones que los individuales, por lo que estos equipos tienen más probabilidades de éxito en la resolución de problemas complejos. Partiendo de que el rendimiento de un equipo es inversamente proporcional a la cantidad de comunicación que se debe entablar se puede plantear que los proyectos muy grandes son guiados mejor por grupos con estructura CC o DC, donde se puedan formar subgrupos fácilmente. Cuando la modularidad es relativamente baja se aplica mejor el organigrama DD y cuando es posible una modularidad alta funcionan bien los organigramas CC o DC, además de que producen menos defectos que los equipos DD. Hay que tener en cuenta que los equipos descentralizados requieren más tiempo para completar un proyecto que los centralizados y al mismo tiempo son mejores cuando se precisa una gran cantidad de comunicación.

Al conformar la estructura de los equipos de desarrollo debe lograrse la integración de las personas en grupos pequeños, esto posibilita un mejor desempeño del grupo de trabajo y aumenta la cohesión del mismo. Un grupo pequeño debe poseer de 8 a 10 personas para lograr una buena interacción entre los integrantes y que cada persona pueda comunicarse con los demás cara a cara. Cuando los integrantes empiezan a comunicarse abiertamente entre sí, esto da lugar a mayor confianza e interacción dentro del

grupo, también las discusiones comienzan a centrarse más específicamente en las tareas de resolución de problemas y en el desarrollo de estrategias para cumplir las tareas.

### **2.4.5. Personas necesarias para el proceso.**

Durante todo el proceso de desarrollo de software, las personas juegan un rol determinante. La entidad de proceso, dentro de la factoría constituye una estructura organizativa donde cada persona implicada ocupa un rol determinado en dependencia de sus habilidades, conocimientos y valores, para mayor información consultar la tabla de resumen de roles en el anexo 10.

En esta entidad se definen los especialistas que se necesitan para los respectivos procesos que en ella se llevan a cabo.

*Especialista en ingeniería software:* Su responsabilidad es guiar, organizar, dirigir todo el proceso de definición del proceso de desarrollo de software. Él coordina el trabajo del especialista en procesos, el especialista en planificación y el contador para lograr una organización del trabajo. También controla y sigue la evolución del proyecto hasta su terminación y entrega. Debe conocer temas de Ingeniería de software y técnicas de organización de la producción y de dirección, siendo una persona modesta, sencilla, responsable y teniendo cierta ética profesional.

*Especialista en procesos:* Tiene como responsabilidad evaluar, establecer y describir el proceso de desarrollo de software, debe ser comunicativo, locuaz, y sobre todo creativo, y tener buena capacidad de análisis. Conocer sobre procesos de software, y si puede ser o tener conocimiento sobre la ingeniería industrial mejor.

*Especialista en planificación:* es responsable de guiar y evaluar a los planificadores en función de que lleven a cabo el proyecto de la manera establecida. Estima las personas que se necesitan para llevar a cabo el proyecto. Debe tener valores tales como, modestia, sencillez, profesionalidad, responsabilidad, ética profesional. Y debe tener conocimiento sobre procesos de software, e ingeniería industrial. Debe dominar técnicas de estimación de tiempo y personas.

*Contador:* será responsable de estimar y controlar el costo del proceso de desarrollo. Debe tener conocimiento sobre temas de contabilidad y economía, dominar técnicas de estimaron de costo, siendo Comunicativo, locuaz, creativo, y teniendo capacidad de análisis.

## 2.4.6. Procesos de soporte

### Organización estructural

La estructura organizacional es definida por el área estructura organizacional de la entidad gestión de proyecto acá son muy pocos los roles pero el especialista en ingeniería de software debe coordinar las tareas de los especialistas de proceso, el especialista en planificación y el contador, garantizando buena eficiencia en el trabajo. La propuesta de estructura organizacional se puede ver en la figura 9.



Figura 9.- Estructura organizacional

### Repositorio

En la entidad que se presenta en este trabajo, los artefactos que son generados en los procesos que tienen lugar en las distintas unidades que la conforman, se almacenan en un repositorio de componentes para su reutilización, que tiene definida una estructura de carpetas. Habrá una carpeta raíz donde estaría localizado el repositorio, esta carpeta tendrá subcarpetas para los procesos de gestión de proyecto, incluyéndose dentro de esta una carpeta para la organización del proceso, la cual contiene tres carpetas más, una para el proceso de control de proceso, otra para el proceso descripción de proceso y una última donde se guardará toda la planificación del proyecto, ver figura 10. En el repositorio se establece una serie de políticas para asegurar la confiabilidad de los artefactos generados durante todo el proceso en las que se define quién o quiénes tienen derecho de insertar y/o modificar un artefacto determinado, ver anexo 11.

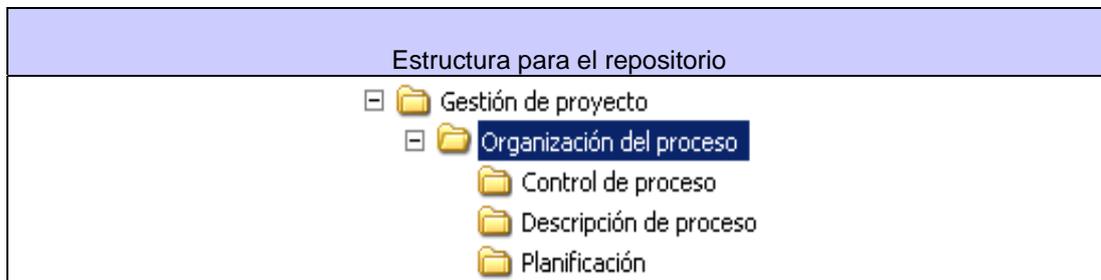


Figura 10.- Representación gráfica de la estructura del repositorio.

### Bases Tecnológicas

La relación entre las entidades bases tecnológicas y proceso, se basa en que en la primera es donde se definirán los mecanismos y herramientas necesarios para la entidad en cuestión.

Todo el proceso de desarrollo está sustentado en métodos, herramientas y mecanismos que ayudan a solucionar una determinada problemática.

*La matriz DAFO:* es una metodología de estudio de la situación competitiva de una empresa dentro de su mercado y de las características internas de la misma, a efectos de determinar sus Debilidades, Amenazas, Fortalezas y Oportunidades. Las debilidades y fortalezas son internas a la empresa; las amenazas y oportunidades se presentan en el entorno de la misma. El objetivo final del análisis DAFO es poder determinar las ventajas competitivas que tienen la empresa bajo análisis y la estrategia genérica a emplear por la misma que más le convenga en función de sus características propias y de las del mercado en que se mueve.

*Diagrama causa- efecto:* ayudan a pensar sobre todas las causas reales y potenciales de un suceso o problema, y no solamente en las más obvias o simples, son idóneos para motivar el análisis y la discusión grupal, de manera que cada equipo de trabajo pueda ampliar su comprensión del problema, visualizar las razones, motivos o factores principales y secundarios, identificar posibles soluciones, tomar decisiones y, organizar planes de acción.

*Microsoft Project:* Es una aplicación que ayuda al usuario a crear planes de proyectos, comunicarlos a otros usuarios y adaptarse a los cambios a medida que éstos se van produciendo. Se utiliza por los directivos de la factoría para introducir las tareas del proyecto y sus duraciones. La organización de tareas es en estructura jerárquica y la vinculación de tareas para diferentes tipos de relaciones está sumamente

simplificada. Con un procesador Pentium II de 200 Mhz. y 32 MB de memoria RAM, el programa se desenvuelve normalmente.

*COCOMO II:* Es un método que ayuda a lograr una estimación del costo, del tiempo y las personas necesarias para tener un proceso estandarizado, eficiente y repetible.

*Puntos de función:* Es un método que ayuda a lograr una estimación del costo, del tiempo y las personas necesarias para tener un proceso estandarizado, eficiente y repetible.

Un resumen de estas bases tecnológicas más especificadas se puede ver en el anexo 12.

## **2.5. Conclusiones.**

En el presente se describió la Universidad de las Ciencias Informáticas, en ella la situación existente en los proyectos productivos lo que indica que existen serias deficiencias, sin cuya adecuada y rápida solución no podrá resolverse el desarrollo de software y el avance en la industria de software a que se aspira. Se describieron los métodos, procedimientos y técnicas utilizadas para llevar a cabo la investigación y se presentó además la propuesta de Entidad Proceso de Desarrollo de Software, la cual plantea una serie de pasos para organizar y definir un proceso con calidad, flexible y mejorable continuamente, para así contribuir a la solución de los problemas que están presentes actualmente en la producción.

## Capítulo 3: Aplicación práctica.

### 3.1. Introducción

Según la situación problemática planteada anteriormente, tomando en cuenta el análisis hecho de los modelos presentados en capítulos anteriores y los métodos descritos a aplicar, en este capítulo se presentan los resultados de los métodos aplicados y se muestra un ejemplo práctico del proceso de desarrollo de software propuesto. Seguidamente se muestran los resultados de la evaluación técnica del modelo.

### 3.2. Análisis de los resultados de la aplicación de las encuestas y entrevistas.

Se entrevistó según lo previsto a un total de 10 personas, 20 especialistas externos a la UCI de las casas consultoras DISAIC, Biomundi, GECYT, del ISPJAE, de la UCI, directivos de la UCI, líderes de proyecto, dirigentes de las organizaciones políticas y de masas y desarrolladores.

La entrevista arrojó que se tiene conciencia de la situación de la producción y que hay gran interés por parte de los desarrolladores y directivos para la búsqueda de soluciones. Que aunque no hay un modelo único de producción para todos se sigue metodologías de desarrollo de software para definir el proceso y los roles, en la mayoría de los casos es el Proceso Unificado de Desarrollo (RUP) y en algunos Programación Extrema (XP).

Consideran que se puede trabajar mucho más en la planificación y en la definición de los procesos pues aunque dos proyectos que trabajan en una misma línea de producción usen la misma metodología para guiar el proceso lo aplican de manera totalmente diferente, que uno de los mayores problemas está en que los procesos no se documentan, que se ha avanzado muchísimo en la documentación de los productos, en la calidad de estos documentos y en la entrega de los mismos a calidad para la liberación del producto, pero que en el otro sentido no. Otro elemento que se debe fortalecer es la gestión de la calidad, el tiempo, costo y los recursos y el uso de estándares así como la reutilización de código.

La mayoría de los entrevistados aunque no conocían las características del enfoque de factoría al hacerles la presentación de las mismas plantearon que un modelo de factoría adecuado a las características de la UCI, con el apoyo de la institución y de los involucrados pudiera cambiar los entornos

productivos y elevar la producción de software y la productividad. Entre los elementos que más se mencionaron que no debía faltarle a un modelo de esta magnitud es que fuera flexible para aplicarse a diferentes líneas de producción, que permitiera la definición de procesos y roles ajustados a las necesidades, la gestión de la calidad, y de los recursos, que permita la definición de las bases tecnológicas y la reutilización de código.

Los entrevistados en las casas consultoras plantearon que en esta era del conocimiento el valor de una empresa aumentará con la capacidad que tenga de realizar una buena gestión del conocimiento y adelantarse a las demandas del mercado y en especial las que se dediquen a la exportación de productos y servicios informáticos y que el objetivo debe ser definir un modelo de producción y de Inteligencia que incentive el compartimiento del conocimiento. Plantearon además que el mayor problema que existe para acometer esto es la falta de cultura en estos temas, poca experiencia, límites en la creación real y práctica de la manipulación de la información y la necesidad de formar más personal que se dedique a estos aspectos.

Las encuestas se les realizaron a 26 líderes de proyecto, 11 planificadores y 77 desarrolladores, a un total de 114 personas de 32 proyectos de todas las líneas de producción y facultades de la universidad. En la figura 11 se muestra el gráfico de pastel que representa la composición de los encuestados.

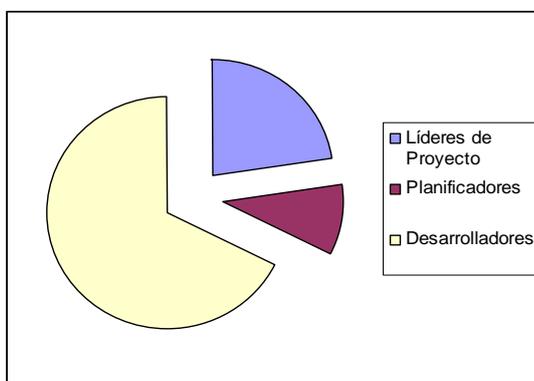


Figura 11.-Gráfico de la composición de los encuestados.

Hay definición de roles y flujo de procesos, en el 75% de los proyectos encuestados fue el resultado más homogéneo que se obtuvo, pues en el 73,33% de los proyectos hubo discrepancias entre líder, planificador y desarrolladores. La gestión de los recursos se realiza en cerca del 30 % de los proyectos.

Los resultados arrojaron que el 64.51% de los proyectos no tiene planificador, es esa la razón de la cifra tan baja de planificadores encuestados; de ellos sólo el 55% utiliza herramientas para llevar a cabo su labor.

En el caso de los desarrolladores, el 30 %, los planificadores el 64% y los desarrolladores el 44%, el resultado más bajo lo presentan los líderes de proyecto que son los que deberían controlar, exigir y dar el ejemplo en el uso del estándar.

Solamente en el 54 % de los proyectos encuestados se firma la aceptación de los entregables al cliente y solamente el 45 % de los entrevistados en los proyectos donde se firman los entregables tiene conocimiento de ello.

En los proyectos productivos de la UCI se ha identificado que existen muchos proyectos en los que el desarrollo de software tiene un alto porcentaje de artesanía y trabajo a la medida, deficiencias en la definición de los flujos de procesos, roles y responsabilidades, los cuales no siempre responden a sus necesidades y a la metodología utilizada, afectándose la eficiencia, la calidad, y el tiempo de desarrollo de un producto. Esto empeorará con el aumento de la fuerza de trabajo y de la demanda del cliente, ocasionando que el desarrollador se sienta desorientado y no sepa qué hacer en cada momento ni a quién dirigirse, llevando a la desorganización de la producción afectando la productividad. La planificación del trabajo tanto personal como a nivel de equipo no es la mejor, en la mayoría de los casos no se siguen estándares establecidos en la Ingeniería de Software, afectándose la efectividad del equipo de desarrollo. A partir de los resultados se realizó un diagrama Causa-Efecto o de Ishikawa que se encuentra en el Anexo 29.

### **3.3. Aplicación Práctica**

#### **3.3.1. Caso de estudio**

El jefe de capital humano de una universidad requiere informatizar el proceso de matrícula de su centro, tanto para estudiantes de nuevo ingreso, como para continuantes. Este cliente necesita que su sistema sea eficiente, pero requiere que se realice en el plazo pactado.

#### **Solución del caso de Estudio**

Para llevar a cabo este proyecto de informatización de matrícula, primeramente se debe definir todos los aspectos importantes del proceso antes de comenzar la realización del sistema. Se deben aclarar y

analizar las necesidades del cliente, definir que es un proyecto que requiere de un sistema eficiente de matrícula de estudiantes, y persigue una línea de producción orientada hacia la informatización de un sistema de matrícula.

El alcance de este proyecto se dirige hacia la factoría de proyectos de software puesto que este tipo de factoría abarca todo el ciclo de vida sistémico para la realización del software, correspondiente al análisis, diseño, implementación, prueba e implantación con conocimiento al detalle del negocio a automatizar, por tanto este proyecto comprenderá 4 fases para ser completado (inicio, colaboración, construcción y transición).

Para desarrollar el sistema con la calidad requerida por el usuario y la entidad de Gestión de proyecto define la cantidad de personas y recursos que necesitara el proyecto y le entregará estos datos al líder del proyecto.

Se define el equipo de desarrollo que interactuará activamente con los clientes y los encargados de realizar la matrícula en la Universidad durante todo el ciclo de desarrollo del producto.

Ya conocido todo el equipo de desarrollo, hay que definir las actividades, los roles y los artefactos que deben estar presentes que no tienen que ser todos los que define una metodología, este caso la metodología a utilizar por ser un proyecto bastante complejo es el Proceso Unificado de Software (RUP), esta define un grupo de trabajadores, artefactos y actividades para el desarrollo de software, aspectos que son cambiantes para cada proyecto. Por ejemplo para la informatización de la matrícula, no necesitamos todas las actividades que define la metodología por lo que podemos omitir algunas.

### 3.3.2. Proceso “Modelamiento del negocio”

Durante este epígrafe se desarrollará la aplicación práctica de la propuesta para la etapa de modelamiento del negocio en la fase de inicio. A continuación en la tabla 2 se muestran las actividades fundamentales que debe concebir el proyecto para esta etapa.

Tabla 2. Tabla para definir actividades

No.	Nombre de la actividad	Descripción	Artefactos de entrada	Tareas	Artefactos de salida	Rol	Bases tecnológicas
1	Identificar procesos del negocio.	Esta actividad consiste en realizar un estudio del	Cita oficial con el cliente.	Entrevistar al cliente para comprender el funcionamiento	Documento Visión. Glosario de términos.	Analista de procesos del	Técnicas de entrevistas.

		negocio que se desea automatizar, logrando entender cómo se lleva a cabo ese proceso manualmente, qué personas son las involucradas y quiénes son los beneficiados en el proceso.		del negocio. Analizar todo el proceso del negocio. Definir quienes son los involucrados en el negocio. Aclarar para beneficio de quién se lleva a cabo el negocio.	Reglas del negocio.	negocio.	
2	Identificar roles y responsabilidades	Esta actividad consiste en verificar todos los datos obtenidos por el cliente y los involucrados en el proceso y plantear los roles que interviene en el negocio y definir las actividades que posee cada uno de ellos dentro del proceso a automatizar.	Glosario de términos. Documento Visión.	Agrupar las actividades que se llevan a cabo manualmente y definir casos de uso del negocio. Identificar quiénes llevan a cabo cada grupo de actividades, que sería las responsabilidades que cada uno posee en el negocio. Definir actores y casos de uso del negocio.	Actor. Casos de uso del negocio. Descripción de casos de uso del negocio.	Analista de procesos del negocio. Diseñador del negocio	Herramienta Case. Técnicas de modelado UML
3	Describir el negocio actual.	En esta actividad se toman los actores y casos de uso ya definidos y	Glosario de términos. Documento Visión. Casos de uso del	Analizar los casos de uso y los actores. Definir la relación que se establece entre actores y	Modelo de casos de uso del negocio. Documento de arquitectura	Analista de procesos del negocio.	Racional Rose, Técnicas de modelado UML

4	Explorar posibles automatizaciones	<p>se estructura el modelos de casos de uso del negocio asignando a cada actor el caso de uso que le corresponde.</p> <p>Esta actividad consiste en analizar el negocio actual, identificar las entidades que no pueden faltar en el negocio y los trabajadores con los que se relacionen para plantear las funciones no deben faltar a la hora de automatizar el sistema.</p>	<p>negocio. Actores. Descripción de los casos de uso del negocio.</p> <p>Modelo de casos de uso del negocio. Documento de arquitectura del negocio. Descripción de los casos de uso del negocio.</p>	<p>casos de uso. Describir el negocio mediante un modelo de casos de uso del negocio.</p> <p>Estudiar la descripción del negocio actual. Identificar entidades y trabajadores del negocio y sus relaciones. Confeccionar el diagrama de actividades. Define los requerimientos de automatización.</p>	<p>del negocio.</p> <p>Diagrama de actividades. Documento de arquitectura del negocio (actualizada).</p>	<p>Diseñador del negocio.</p>	<p>Racional Rose, Técnicas de modelado UML</p>
---	------------------------------------	--	--	---	--	-------------------------------	--

Como ya se encuentran definidas las actividades que serán ejecutadas durante esta primera etapa y cada una de ellas tiene establecido el rol que debe realizarla, se pasa a la asignación de responsabilidades para cada uno de los roles presentes, resaltando las habilidades, valores y sistema de conocimientos que de poseer cada uno, (Ver Tabla 3).

Tabla 3. Tabla para definir roles

Nombre del rol	Responsabilidades	Competencias		
		Habilidades	Sistema de conocimientos	Valores
Líder de proyecto	El líder de proyecto asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos, también establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto y se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos. Planificación y control del proyecto	Un líder tiene que ser una persona con una gran capacidad para dirigir y organizar. Debe tener cualidades de líder, espíritu de iniciativa. Capacidad para la toma de decisiones y para actuar. Sensibilidad para detectar e identificar problemas.	El líder de Proyecto debe conocer de análisis, diseño, arquitectura y programación. Debe saber de calidad de software, debe tener conocimientos sobre planificación de proyectos y control de riesgos. Conocimientos organizacionales y conocimientos administrativos	Todo líder de proyecto debe ser una persona sumamente responsable, honesta, organizada, leal y sobre todo altamente comprometida con Fidel y la Revolución
Analista de procesos del negocio	Es el responsable de la arquitectura del negocio por lo que dirige y coordina el proceso de modelamiento del negocio. Decide cuales son los actores y los procesos del negocio y las relaciones entre ellos y cuales son las reglas del negocio a tener en cuenta.	Debe ser buen facilitador y tener excelentes habilidades de comunicación.	Debe conocer sobre las técnicas de las entrevistas. Debe tener conocimiento sobre las técnicas de modelado UML. Debe saber trabajar con la herramienta Rational Rose.	Debe ser una persona sumamente responsable, honesta, organizada, leal y sobre todo altamente comprometida con Fidel y la Revolución.
Diseñador del negocio	Es el responsable de describir los procesos del negocio y como parte de la realización de estos procesos	Debe ser una persona con una gran habilidad de captación. Debe ser hábil en el	Debe conocer sobre las técnicas de modelado UML, debe conocer sobre automatización de	Debe ser capaz, honesto, responsable, y revolucionario.

identifica las entidades y trabajadores del negocio y las relaciones entre estos. Además es el encargado de definir cuáles son los requerimientos en la automatización.	modelado y debe ser una persona muy expresiva.	sistemas y debe saber y trabajar con la herramienta Racional Rose.	
---	--	--	--

### 3.3.3. Organización

Hasta el momento no deben quedar dudas en cuanto al proceso, pues ha quedado bastante detallado todo el proceso donde cada persona involucrada conoce que papel desempeña, que actividades y tareas le corresponde le desarrollar y sobre todo que artefactos necesita para realizar su trabajo y cuáles debe obtener como resultado de su esfuerzo, solo falta definir la estructura organizacional del equipo de desarrollo para el modelamiento del negocio en la fase de inicio, la cual se puede observar en la Figura 12, donde el líder de proyecto debe dirigir y coordinar las tareas del analista de procesos del negocio y del diseñador del negocio garantizando buena eficiencia en el trabajo.

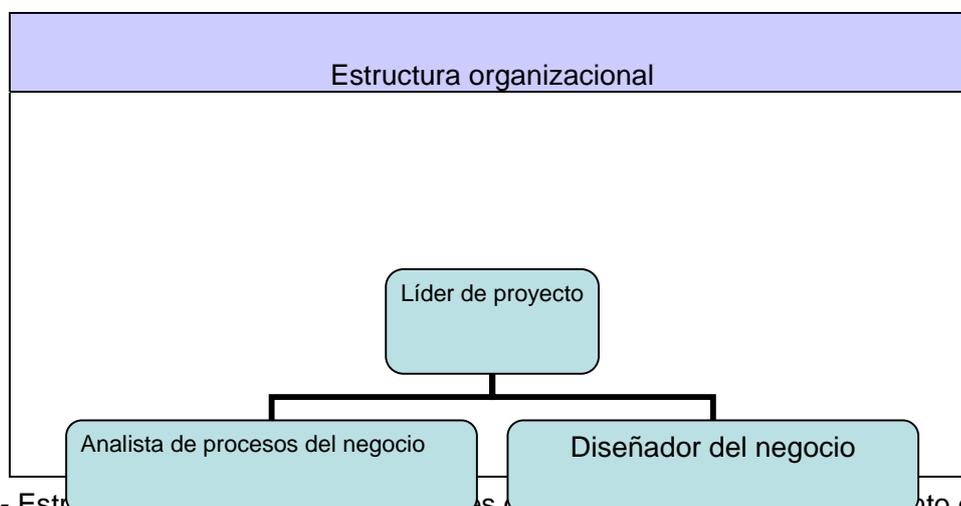


Figura 12.- Estructura organizacional del equipo de desarrollo del negocio.

### 3.3.4. Repositorio

Ya detallado todo en cuanto a proceso y personas involucradas, se procede a crear un archivo o servidor para guardar toda la información del proyecto de forma segura, la Entidad Repositorio de Componente es la encargada de asignar dónde y cómo se va a guardar la información de todo el proyecto.

En este proceso, los artefactos que son generados en las actividades, se almacenan en un repositorio de componentes para su reutilización, éste tiene definida una estructura de carpetas. Habrá una carpeta raíz donde estaría localizado el repositorio, esta carpeta tendrá subcarpetas para las actividades de identificar roles y responsabilidades, una donde se guardará toda la planificación del proyecto, otra donde se guardará toda la documentación de la actividad de identificar posibles automatizaciones y una identificar procesos del negocio, incluyéndose dentro de esta una carpeta para la descripción del negocio actual. Ver figura 13.

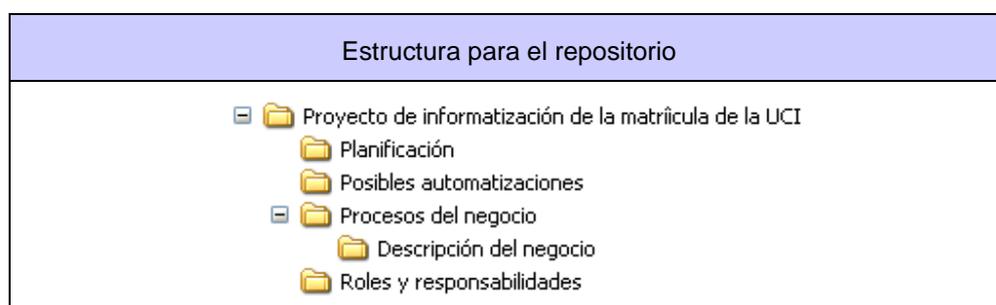


Figura 13 Representación gráfica de la estructura del repositorio.

Todos los roles conocen donde debe guardar su información, solo se hace necesario plantear las políticas de seguridad que se deben de seguir pues no es correcto que las carpetas estén accesibles para todos, pues existen artefactos que solo pueden ser modificados por la persona que los crea y otros a los que solo el líder de proyecto tiene acceso como se puede observar en la tabla 4.

Tabla 4. Tabla resumen de las políticas de seguridad de los artefactos

Artefacto/Acciones	Insertar	Modificar
Documento Visión	Analista de procesos del negocio.	Líder de proyecto
Glosario de términos	Analista de procesos del negocio, líder de proyecto	Analista de procesos del negocio, líder de proyecto
Reglas del negocio	Analista de procesos del negocio	Líder de proyecto
Descripción de los casos de uso	Diseñador del negocio	Diseñador del negocio
Modelo de casos de uso del negocio	Analista de procesos del negocio.	Analista de procesos del negocio.
Documento de arquitectura del negocio	Analista de procesos del negocio, Diseñador del negocio	Líder de proyecto
Diagrama de actividades	Diseñador del negocio	Líder de proyecto

### 3.3.5. Bases tecnológicas

Todo modelado del negocio necesita una serie de bases tecnológicas a continuación en la tabla 5 se muestran las más importantes, para qué se necesitan y las propiedades que poseen.

Tabla 5. Tabla para definir las bases tecnológicas

Nombre de la Bases Tecnológicas	Descripción
Herramienta Rational Rose	Esta herramienta es imprescindible para el modelado del negocio pues posee una serie de artefactos y diagramas que hacen más cómodo y rápido el trabajo a la hora de hacer un modelo o diagrama.
Técnicas de modelado UML	Estas técnicas ayudan a la hora de realizar diagramas y apoyan la comprensión cuando se va a utilizar la herramienta Rational Rose y establecen normas y artefactos estandarizados para diagramar.
Técnicas de entrevista	Estas técnicas son útiles para desarrollar una entrevista pues brindan los aspectos que no pueden faltar cuando se va a entrevistar a alguna persona.

### 3.3.6. Elementos del Proceso

Para saber cuando una actividad puede concluir y puede comenzar otra sin faltar ningún aspecto se crean las listas de chequeo que son las encargadas de monitorear los artefactos que deben de estar presentes en cada actividad, por ejemplo para el Documento Visión la lista de chequeo quedaría:

Tabla 6. Lista de Chequeo para el artefacto “Documento Visión”

Lista de chequeo para el Documento Visión

	¿Se cumple?		Condición	
	SI	No	Entrada	Salida
¿Fue creado a partir de la entrevista con el cliente?	X		X	
¿Están descritos todos los usuarios y los involucrados?	X			X
¿Fue creado por el Analista de procesos del negocio?	X			X
¿Esta descrito detalladamente todo el proceso?	X			
¿Contiene las necesidades de los usuarios y los involucrados?	X			X

Para finalizar con la descripción del proceso de modelamiento del negocio, se realiza un mapa del proceso (Ver Figura 14) para mejor entendimiento de los involucrados en este, y se concibe la lista de riesgos del proyecto (tabla 7), la cual debe de tenerse en cuenta para realizar las actividades en función de mitigar estos riesgos.

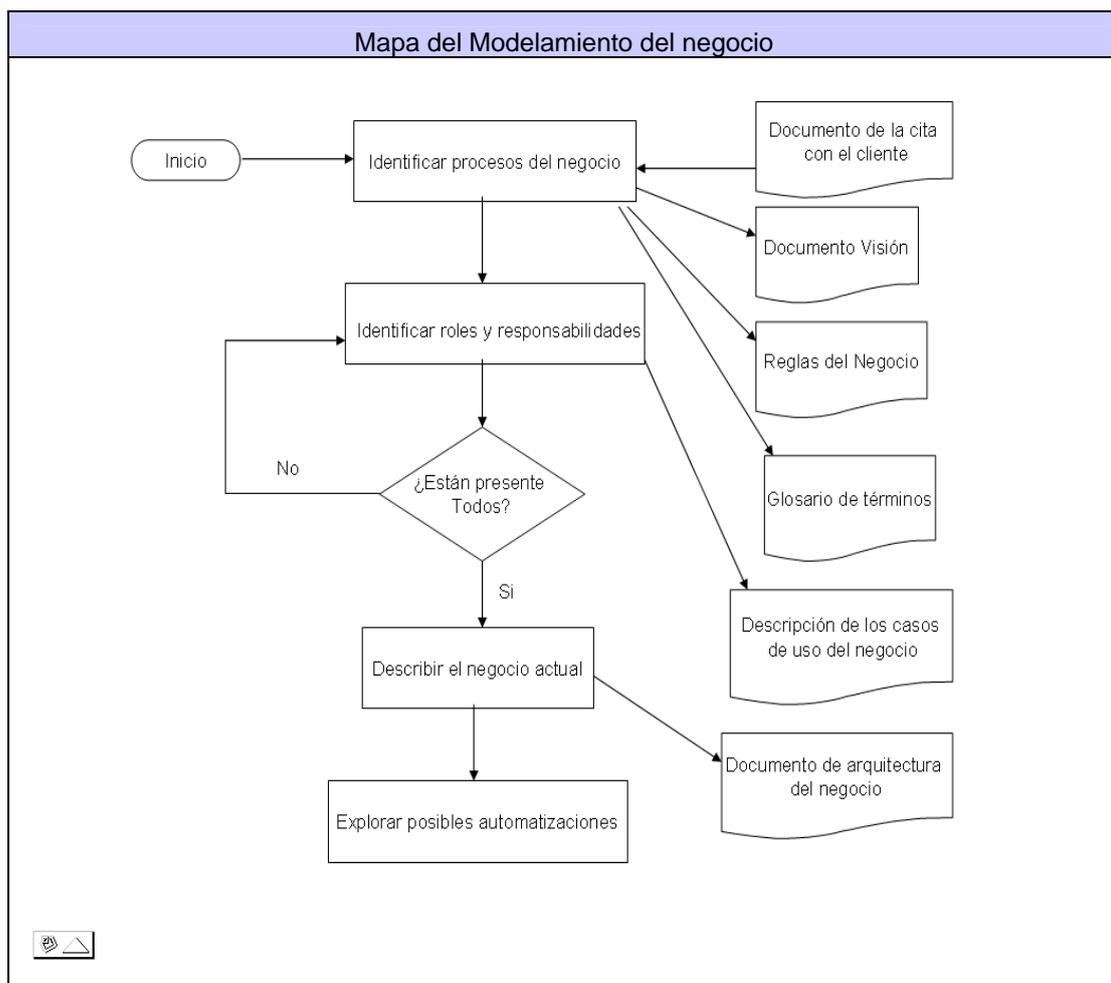


Figura 14.- Mapa del Modelamiento del negocio en la fase de inicio.

El propósito de la lista de riesgos es proporcionar la información necesaria para controlar los riesgos que se puedan presentar en el proyecto. Esta lista debe ser revisada frecuentemente por todo el personal de proyecto para detectar la presencia de un riesgo y cuales son los planes de contingencia que debemos llevar a cabo, y muy importante cómo prevenir este riesgo.

Tabla 7. Ejemplo de lista de riesgo para esta etapa

<b>ID</b>	<b>Riesgo</b>	<b>Tipo de Riesgo</b>	<b>Impacto</b>	<b>Descripción</b>	<b>Probabilidad</b>	<b>Efectos</b>
3SA	El proceso es conocido por los desarrolladores de manera teórica pero nunca se han enfrentado a un proceso real.	Personal	Problemas a la hora de enfrentar los procesos reales.	Puede traer errores pues de manera práctica no se ha tenido experiencia.	Altas	Catastrófico
4SA	La definición del proceso se realiza basado en la metodología RUP, pero no se tiene experiencia de su uso.	Organización	Problemas a la hora de implementar la metodología RUP.	Falta de interacción con la práctica pues se conoce la metodología pero no hay experiencia de su puesta en práctica.	Alta	Serias
5TB	Los desarrolladores nunca han trabajado en equipo y el proceso demanda gran trabajo en equipo.	Personal	Hay problemas a la hora de integración del equipo.	Hemos tenido experiencia en el trabajo en Factoría pero como desarrolladores individuales, no desarrollando módulos en equipo.	Baja	Tolerante
6TB	El equipo de desarrollo no ha trabajado anteriormente con el cliente.	Organización	Podría traer dificultades a la hora de saber que es lo que desea el cliente.	No se conoce ni se tiene experiencia como comunica sus ideas el cliente.	Baja	Tolerante
16TM	No se asignen las máquinas necesarias.	Tecnológico	No se pueda aprovechar el tiempo	Hay que realizar tareas del proyecto y de estudio en	Media	Tolerante

			necesario en la maquina.	las que son necesarias las computadoras y no nos alcanza el tiempo en la máquina.		
19BI	Problemas con la electricidad.	Tecnológico	Si no se tienen guardados los cambios podrían traer demora en la entrega del proyecto. Rotura de maquinas.	Puede ser por: Fluctuación del voltaje (Alto o Bajo Voltaje). Falta de electricidad y los grupos electrógenos no funcionen.	Baja	Tolerante

3SA > <El proceso es conocido por los desarrolladores de manera teórica pero nunca se han enfrentado a un proceso real.>

1.1.1. Indicadores

Cuando se selecciona el personal, ya se sabe el nivel de experiencia que presenta el mismo.

1.1.2. Estrategia de Mitigación

Buscar personal calificado que se encargue de hacer múltiples revisiones para perfeccionar el trabajo de los desarrolladores que no tienen experiencia.

1.1.3. Plan de Contingencia

Buscar personal calificado e incluirlo en nuestro grupo de desarrollo.

<4SA > <La definición del proceso se realiza basado en la metodología RUP, pero no se tiene experiencia de su uso.>

1.1.1. Indicadores

Cuando se selecciona el personal, ya se sabe el nivel de experiencia en la metodología RUP que presenta el mismo.

1.1.2. Estrategia de Mitigación

Buscar personal calificado en RUP que se encargue de hacer múltiples revisiones para perfeccionar el trabajo de los desarrolladores que no tienen experiencia.

#### 1.1.3. Plan de Contingencia

Buscar personal calificado e incluirlo en nuestro grupo de desarrollo.

<5TB > <Los desarrolladores nunca han trabajado en equipo y el proceso demanda gran trabajo en equipo.>

##### 1.1.1. Indicadores

El trabajo recae sobre algunas personas del equipo.

##### 1.1.2. Estrategia de Mitigación

Se definirá el proceso y se discutirá con los miembros del equipo de desarrollo para que se identifiquen con el y entiendan la necesidad del trabajo en equipo.

#### 1.1.3. Plan de Contingencia

Planificar el horario de trabajo para desarrollador de manera que se logre una convergencia en el equipo.

<6TB > <El equipo de desarrollo no ha trabajado anteriormente con el cliente.>

##### 1.1.1. Indicadores

No hemos tenido interacción previa con el cliente.

##### 1.1.2. Estrategia de Mitigación

Crear una conciencia en todos los miembros del equipo que aunque no se halla trabajado antes con el cliente deben lograr una buena comunicación con él para que el proyecto se realice en tiempo y con éxito.

#### 1.1.3. Plan de Contingencia

Planificar reuniones donde se fomente la comunicación cliente-desarrollador.

<16TM > <No se asignen las máquinas necesarias.>

##### 1.1.1. Indicadores

Los entregables no están en tiempo.

##### 1.1.2. Estrategia de Mitigación

Planificar con antelación las máquinas necesarias y hacer una gestión previa. Se les impartirá Personal Software Process (PSP) para la organización personal y para la gestión de defectos.

1.1.3. Plan de Contingencia

Planificar horario especial de trabajo.

<19BT > <Problemas con la electricidad.>

1.1.1. Indicadores

Problemas eléctricos.

1.1.2. Estrategia de Mitigación.

Gestionar 2 o 3 Backup (que sean también estabilizador de voltaje), y que estén de forma permanente en las maquinas designadas que son los servidores y una máquina para trabajo de urgencia (la de jefe de proyecto, la cual utilizaríamos para la comunicación con el cliente).

1.1.3 Plan de Contingencia

Gestionar computadoras en un lugar que no tenga problemas con el fluido eléctrico.

Todos los aspectos detallados anteriormente contribuyen a eliminar errores en cuanto a proceso de desarrollo de software se refiere, solo queda que el personal este claro de la importancia de tener un proceso eficiente, planificado y eficaz.

### **3.4. Conclusiones**

En el capítulo se analizaron los resultados que se obtuvo de la aplicación de los métodos, identificando los problemas en la producción de software en la UCI y su relación causa efecto en un diagrama de Ishikawa, se pudo demostrar deficiencias existentes en la producción particularmente los errores en la definición de los flujos de trabajo, los roles y las responsabilidades.

Se muestra brevemente un ejemplo de la aplicación práctica de la propuesta para el modelamiento del negocio en la fase de inicio de un proceso de desarrollo de software. Esta entidad contribuyó a la definición del flujo de proceso, los roles y las responsabilidades a partir de una metodología, así como a la coordinación de las tareas a desarrollar por cada rol definido, agregando a la organización de los artefactos, mecanismos y herramientas que se reflejan en cada etapa, con ello se eleva el sentido de pertenencia para con el proyecto.

## **CONCLUSIONES**

Después de identificar las características del enfoque de Factoría y los modelos mediante los que se aplica este enfoque, de ver la experiencia internacional de su aplicación en el mundo, de evidenciar las principales características que debe presentar un proceso de desarrollo de software e identificar los problemas en la producción se puede concluir que una Entidad Proceso que lleve a cabo un proceso con flexible y adaptable como el que se define en el Capítulo 2, puede solucionar los problemas existentes en la producción de software puesto que:

Desde el punto de vista teórico la aplicación del Modelo de Factoría aplicando Inteligencia con la descripción de esta entidad contribuye a industrializar el desarrollo de software, logrando un proceso con calidad y por tanto un producto eficiente. Puesto que esta entidad establece el uso de una metodología sin estandarizar ninguna, la definición de artefactos, actividades, roles y una serie de herramientas que apoyan la producción, sin obligar ni presionar al equipo de desarrollo a utilizar ninguno en particular.

## **RECOMENDACIONES**

Los objetivos de este trabajo no incluyen una evaluación técnica de la propuesta de Proceso de Desarrollo de Software, ni la realización de pruebas pilotos de dicha propuesta, aspecto que es esencial para lograr lo que se persigue con el Modelo de Factoría aplicando Inteligencia. Por lo que se propone:

- Realizar un estudio de factibilidad económica que demuestre la factibilidad o no de este proceso de desarrollo de software.
- La implantación de esta entidad en el Modelo de Factoría de Software aplicando Inteligencia y posteriormente realizar su aplicación y evaluación en proyectos pilotos en la UCI.
- Realizar una evaluación técnica del proceso definido y analizar los resultados obtenidos

## REFERENCIAS BIBLIOGRÁFICAS

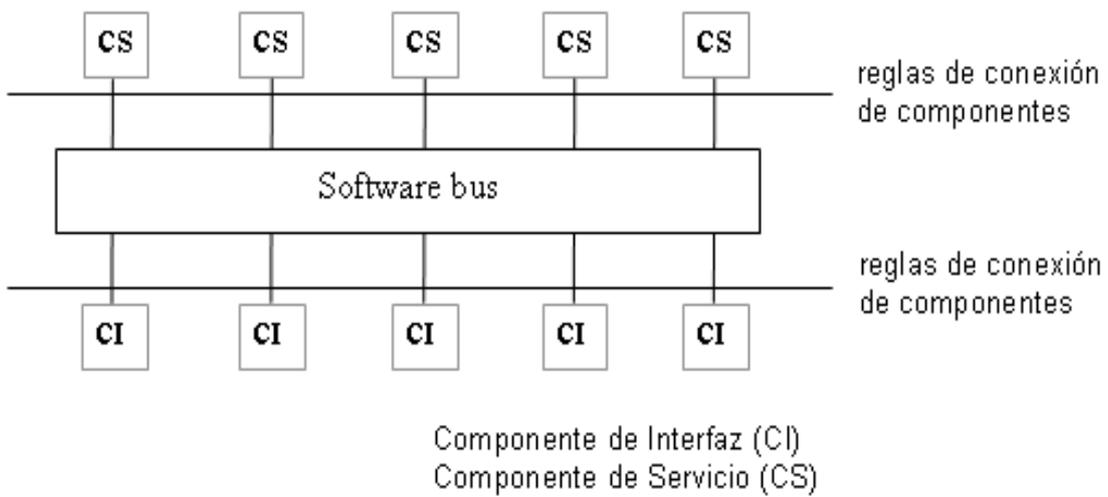
- BARZANALLANA, R. M.-A. *METODOLOGIA DE DESARROLLO DE SOFTWARE*, 2007. [Disponible en: [http://www.wikilearning.com/metodologia\\_de\\_desarrollo\\_de\\_software-wkccp-3617-1.htm](http://www.wikilearning.com/metodologia_de_desarrollo_de_software-wkccp-3617-1.htm)]
- BASILI, V. R. C., G.; CANTONE, G. *A Reference Architecture for the Component Factory*. *ACM Transaction on Software Engineering and Methodology*. 1. 1992. 53-80 p.
- BIBLIOTECANET. *GESTIÓN PROCESOS* 2006. [Disponible en: [http://web.jet.es/amosarrain/Gestion\\_procesos.htm](http://web.jet.es/amosarrain/Gestion_procesos.htm)]
- BLANCO, C. *Prosodie amplía sus actividades de outsourcing en España mediante un acuerdo de colaboración con el Grupo Gesfor.*, 2006. [2006]. Disponible en: [http://www.prosodie.es/uploads/tx\\_smartcommunique/20060831\\_04.pdf](http://www.prosodie.es/uploads/tx_smartcommunique/20060831_04.pdf)
- BLANDO, C. M. *Modelo de Capacidad y Madurez (CMM y CMMI)*, 2005.
- BLAYA, I. *Gestión por procesos* 2006. [Disponible en: [www.upm.es/innovacion/calidad/documentos/Gestion\\_Procesos.pp](http://www.upm.es/innovacion/calidad/documentos/Gestion_Procesos.pp)]
- CANTONE, G. *Factory 2000, 1992. 'Competitive Performance Through Advanced Technology'.*, *Third International Conference on (Conf. Publ. No. 359)*, 1992. [Disponible en: [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=171866](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=171866) 4249517.
- CASAÑOLA, Y. T. *Modelo de Factoría aplicando Inteligencia*. La Habana, UCI, 2007. p.
- CORDERO, Y. *Área de Gerencia Gubernamental: ANÁLISIS DE LOS PROCESOS DE TRABAJO* 2007. [Disponible en: [http://www.ogp.gobierno.pr/html/GG\\_E008.html](http://www.ogp.gobierno.pr/html/GG_E008.html)]
- CUSUMANO, M. A. *Software Factory: A Historical Interpretation*. en. 2. 1989. 6: 23-30.p.
- FERNANDES, A. A. T., DESCARTES DE SOUZA. *Fábrica de Software: Implementação e Gestão de Operações*. . 2004. p.
- FERNSTROM, C. N., K. H; OHLSSON, L. *Software Factory Principles, Architecture and Experiments*. *IEEE Software*. . 2. 1992. 36-44 p.
- FUGGETTA, A. *“Il Processo Software, Aspetti strategici e organizzativi*. Venecia, 1995. p.
- GARCIA, A. R. *Gestión de los Procesos*, 2005. [2007]. Disponible en: <http://www.localret.net/jornades/materials/ptcal/procesosalcobendas.pdf>

- GONZALEZ, J. F. Qué son las metodologías ágiles, , 2006.
- HOZ, Y. R. L. *Modelo funcional de la Factoría de Software de la UCI para la línea Carrefour.*, 2005. p.
- I JACOBSON, G. B., J RUMBAUGH *Applying UML in The Unified Process*, 1998. [Disponible en: <http://www.rational.com/uml>]
- ISABEL RAMOS ROMAN, M. R. C., MIGUEL TORO BONILLA. *Los Modelos Dinámicos y la Ingeniería del Software*  
1998. [Disponible en: [www.sc.ehu.es/jiwdocoj/remis/docs/modelos.html](http://www.sc.ehu.es/jiwdocoj/remis/docs/modelos.html)]
- IVAN AAEN, P. B., LARS MATHIASSEN. *Software Factories*, 1997. [Disponible en: [http://www.cs.auc.dk/~larsm/Dr\\_Tech/Volume\\_II/17.pdf](http://www.cs.auc.dk/~larsm/Dr_Tech/Volume_II/17.pdf)]
- LABRIN, B. C. *ENFOQUES DE DESARROLLO DE SOFTWARE DESDE UNA PERSPECTIVA DE GESTIÓN DEL CONOCIMIENTO*, 2005. [Disponible en: [http://www.inf.ucv.cl/~usr\\_fac/revista/bcl\\_ideas2004.doc](http://www.inf.ucv.cl/~usr_fac/revista/bcl_ideas2004.doc)]
- LEON, R. A. H. *EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTÍFICA*. 2002. p. 959-16-0343-6
- LETELIER, P. *Introducción a Rational Unified Process*, 2004. [Disponible en: <http://www.dsic.upv.es/~letelier/pub/p16.ppt>]
- LI, C. L., H.; LI, M. *A Software Factory Model Based on ISO 9000 e CMM for Chinese Small Organization*. . Second Asia-Pacific Conference on Quality Software (APAQS'01), Hong Kong, 2001. p.
- MARTINEZ, G. *Aberdeen Software Factory*, 2006. [Disponible en: <http://solutions.mysql.com/solutions/partner.php?partner=1031>].
- MATSUMOTO, Y. *Essence of Toshiba Software Factory.*, 2002. [2006]. Disponible en: <http://www5d.biglobe.ne.jp/~y-h-m/EssenceOfToshibaSoftwareFactory.pdf>.
- PALACIO, J. *Introducción a ISO/IEC 15504*, 2004. [Disponible en: [www. qualitatis.org](http://www.qualitatis.org)]
- PAYAN, W. A. J.-C. A. *GESTIÓN POR PROCESOS. TECNICAS BASICAS*. 2007. p.
- PEREZ, C. L. *Modelo de Madurez de la Capacidad del Software. Revista de Ingeniería Informática del CIIRM*, 2004.
- PRESSMAN, F. *Software engineering: a practitioner's approach*. . 1992. p.
- PROCESIX. *CMMI*, 2004. [Disponible en: ]
- RIOS, O. B. *Gestión Educativa*, 2006. [Disponible en: ]

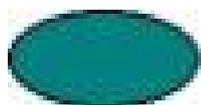
- RODRIGUEZ, J. M. P. *Consideraciones teóricas acerca de la Gestión por Procesos* 2006 [Disponible en: [http://www.wikilearning.com/consideraciones\\_teoricas\\_acerca\\_de\\_la\\_gestion\\_por\\_procesos-wkccp-12497-1.htm](http://www.wikilearning.com/consideraciones_teoricas_acerca_de_la_gestion_por_procesos-wkccp-12497-1.htm)]
- ROSALES, J. C. L. *Programa en Microsoft Visual Basic 6.0 para el análisis de riesgos eléctricos en oficinas y centros de cómputo*. Facultad de Ingeniería Industrial. EAP. de Ingeniería Industrial, 2003, Universidad Nacional Mayor de San Marcos., 2003. p.
- SCALONE, F. L. *Estudio comparativo de los modelos y estándares de calidad de software*.: Facultad Regional. Buenos Aires, Universidad Tecnológica Nacional, 2006. p.
- VALENTIN, J. M. J. *Gestión de proyectos*, 2006. [Disponible en: [http://www.gestionempresarial.info/VerItemProducto.asp?Id\\_Prod\\_Serv=28&Id\\_Sec=8](http://www.gestionempresarial.info/VerItemProducto.asp?Id_Prod_Serv=28&Id_Sec=8)]
- VILLA, M. D. L. *Modelos de Evaluación y Mejora de Procesos: Análisis Comparativo* 2005.
- WIKIPEDIA. *CMMI*, 2007. [Disponible en: [http://es.wikipedia.org/wiki/Modelo\\_de\\_Capacidad\\_y\\_Madurez](http://es.wikipedia.org/wiki/Modelo_de_Capacidad_y_Madurez)]
- ZARZUELA, J. F. *Metodologías Ágiles*. GERMINUS, 2006.

## ANEXOS

### Anexo 1 Enfoque Software bus en el Modelo Eureka



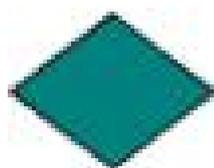
## Anexo 2 Símbolos utilizados en el Mapa de Proceso



Ovalo - indica el inicio (insumo) o el final (producto) del proceso



Cuadrado - indica operación, acción, paso, tarea o actividad



Diamante - indica decisión



Cuadrado cortado - indica información o documento impreso



Triángulo - documento en espera de una acción

### Anexo 3 Diseño de la entrevista #1

1. ¿Considera que en la UCI hay un modelo de producción de software?
2. ¿Considera que se necesita un modelo de producción de software en la universidad?
3. ¿Cuáles elementos cree vitales en ese modelo?
4. ¿Considera que la producción de software debe responder a características como:
  - Proceso definido y estandarizado para el desarrollo de software basado en una metodología y con el uso de los principios de la industrialización.
  - Control y almacenamiento en bibliotecas de componentes de software (documentos, código, métodos, etc.).
  - Producción de software fuertemente basada en métodos y técnicas estandarizadas.
  - Estimación de costos y tiempo basados en el conocimiento real de la capacidad productiva, mediante métodos de obtención basados en datos históricos.
  - Producción a gran escala con productos de diferentes magnitudes?
5. ¿Sabe que el enfoque de Factoría de software responde a esas características?
6. ¿Qué cree si se propone un modelo de factoría de software que además de responder a esas características permita:
  - Uso de la inteligencia para la orientación estratégica a corto, mediano y largo plazo.
  - Gestión de proyecto, de la Calidad y de los recursos.
  - Definición del mapa de proceso y estructura organizacional basado en una metodología y en estándares.
  - La producción basada en componentes donde exista un área de producción de software y otra de componentes.
  - El uso de estándares como CMMI, PSP y TSP, ISO.
  - La definición de reglas que permitan la coordinación de cada una de las personas que intervienen en el proceso y el ensamblaje de cada uno de los componentes.
  - La clasificación de las factorías según el alcance?

---

## **Anexo 4 Diseño de la Entrevista # 2 a especialistas en la producción de software.**

1. ¿Considera importante definir un modelo de producción de software?
2. ¿Considera que la producción de software debe responder a características como:
  - Proceso definido y estandarizado para el desarrollo de software basado en una metodología y con el uso de los principios de la industrialización.
  - Control y almacenamiento en bibliotecas de componentes de software (documentos, código, métodos, etc.)
  - Producción de software fuertemente basada en métodos y técnicas estandarizadas.
  - Estimación de costos y tiempo basados en el conocimiento real de la capacidad productiva, mediante métodos de obtención basados en datos históricos.
  - Producción a gran escala con productos de diferentes magnitudes?
3. ¿Sabe que el enfoque de Factoría de software responde a esas características?
4. ¿Qué cree si se propone un modelo de factoría de software que además de responder a esas características permita:
  - Uso de la inteligencia para la orientación estratégica a corto, mediano y largo plazo.
  - Gestión de proyecto, de la Calidad y de los recursos.
  - Definición del mapa de proceso y estructura organizacional basado en una metodología y en estándares.
  - La producción basada en componentes donde exista un área de producción de software y otra de componentes.
  - El uso de estándares como CMMI, PSP y TSP, ISO.
  - La definición de reglas que permitan la coordinación de cada una de las personas que intervienen en el proceso y el ensamblaje de cada uno de los componentes.
  - La clasificación de las factorías según el alcance?
5. ¿Qué otras cosas usted considera que deba tener?

---

## **Anexo 5 Diseño de la Entrevista # 2 a especialistas en el uso de Inteligencia.**

1. ¿Practican en el centro la Vigilancia Tecnológica, Gestión del Conocimiento y/o Inteligencia Empresarial?
2. ¿Cómo lo realizan?
3. ¿Qué especialistas utilizan?
4. Inicios de su aplicación.
5. ¿Qué resultados han obtenido con la aplicación de los mismos?
6. ¿Por qué utilizar Vigilancia Tecnológica, Gestión del Conocimiento y/o Inteligencia Empresarial?
7. Ventajas y desventajas del empleo de estas técnicas.
8. ¿Cómo comenzar a aplicar en una empresa las mismas?
9. ¿En qué esferas de la producción se ha aplicado?
10. ¿En la Industria del Software qué resultados han alcanzado o qué aportes existirían?
11. ¿Considera importante definir un modelo de producción de software?
12. ¿Considera que la producción de software debe responder a características como:
  - Proceso definido y estandarizado para el desarrollo de software basado en una metodología y con el uso de los principios de la industrialización.
  - Control y almacenamiento en bibliotecas de componentes de software (documentos, código, métodos, etc.).
  - Producción de software fuertemente basada en métodos y técnicas estandarizadas.
  - Estimación de costos y tiempo basados en el conocimiento real de la capacidad productiva, mediante métodos de obtención basados en datos históricos.
  - Producción a gran escala con productos de diferentes magnitudes?
13. ¿Sabe que el enfoque de Factoría de software responde a esas características?
14. ¿Qué cree si se propone un modelo de factoría de software que además de responder a esas características permita:
  - Uso de la inteligencia para la orientación estratégica a corto, mediano y largo plazo.
  - Gestión de proyecto, de la Calidad y de los recursos.
  - Definición del mapa de proceso y estructura organizacional basado en una metodología y en estándares.

- La producción basada en componentes donde exista un área de producción de software y otra de componentes.
- El uso de estándares como CMMI, PSP y TSP, ISO.
- La definición de reglas que permitan la coordinación de cada una de las personas que intervienen en el proceso y el ensamblaje de cada uno de los componentes.
- La clasificación de las factorías según el alcance?

15. ¿Qué otra cosas usted considera que deba tener?

## Anexo 6 Diseño de la encuesta # 1

### Cuestionario

Con este cuestionario pretendemos identificar potencialidades y deficiencias en el proceso productivo. Le pedimos sinceridad a la hora de responder las preguntas, le aseguramos confidencialidad y anonimato a su respuesta, solamente debe mencionar el rol que desempeña dentro del proyecto al que usted pertenece.

**Rol** \_\_\_\_\_

Responde las siguientes preguntas y marcar con una x en el caso que haga falta:

1. ¿Están definidos los roles y responsabilidades de los miembros del proyecto de forma que respondan a las necesidades del mismo?

Sí \_\_\_ No \_\_\_

a. En caso que sea afirmativo responde las siguientes preguntas

- ¿Cómo se define en su proyecto?

---



---



---

- ¿Los miembros del proyecto asumen las responsabilidades que les corresponden?

Sí \_\_\_ No \_\_\_ A veces \_\_\_ Nunca \_\_\_

- ¿Conocen las funciones de su rol? ¿Cuáles son?

---



---



---

2. ¿Se define el flujo de trabajo en su proyecto?

Sí \_\_\_ No \_\_\_ A veces \_\_\_ Nunca \_\_\_

a. ¿Se tiene en cuenta la necesidad de la capacitación del personal?

Sí \_\_\_ No \_\_\_ A veces \_\_\_ Nunca \_\_\_

**b.** ¿Cómo lo definen?

---



---



---



---

**3.** ¿Se identifica el proyecto con una línea de desarrollo, con el fin de poner a los especialistas de grupo en función de la misma?

Sí \_\_\_ No \_\_\_ A veces \_\_\_ Nunca \_\_\_

**a.** ¿Por qué lo hacen? ¿Cómo lo hacen?

---



---



---

**4.** ¿Los miembros del proyecto hacen una planificación de las tareas de manera individual o en equipo, de acuerdo al cronograma del proyecto?

No \_\_\_ A veces \_\_\_ Nunca \_\_\_ Individual \_\_\_ Colectivo \_\_\_

**a.** ¿Aplican técnicas como PSP y TSP?

Sí \_\_\_ No \_\_\_

**b.** ¿Cuál utilizan?

TSP\_\_\_ PSP\_\_\_ Ambas\_\_\_ Otras \_\_\_\_\_

**c.** Enlace con una sola línea, lo que se relaciona de la columna izquierda con la columna derecha.

TSP\_\_\_                    \_\_\_ Trabajo individual.

PSP\_\_\_                    \_\_\_ Trabajo en colectivo o grupal.

                              \_\_\_ Trabajo en factorías de software.

                              \_\_\_ Ninguna de las anteriores.

**5.** ¿Se gestionan los costos, plazos y recursos?

---

Sí \_\_\_ No \_\_\_ A veces \_\_\_ Nunca \_\_\_

a. ¿Qué tipo de herramientas se utilizan o deberían utilizar para dichas tareas? Diga cuál corresponde para cada uno.

---

---

b. ¿Se establece una fecha fija para la revisión y control de los plazos de entrega del trabajo del proyecto?

Sí \_\_\_ No \_\_\_ A veces \_\_\_ Nunca \_\_\_

c. ¿De qué forma se realiza? ¿Cómo creen ustedes que se deba realizar?

---

---

---

d. ¿Los jefes del proyecto tienen el control sobre los recursos utilizados por los miembros del proyecto y la utilización de los mismos en el tiempo establecido?

Sí \_\_\_ No \_\_\_ A veces \_\_\_ Nunca \_\_\_

6. Teniendo en cuenta que las cuestiones técnicas importantes para el desarrollo del sistema y las tecnologías sobre las que el sistema va a ser implantado. Responda las siguientes preguntas:

o ¿Se definen las bases tecnológicas?

Sí \_\_\_ No \_\_\_

o ¿Cuáles son las bases tecnológicas que son implementadas en su proyecto?

---

---

---

o ¿Se definen los estándares a utilizar en el proyecto? (Gráficos, de codificación, etc.)

Sí \_\_\_ No \_\_\_

o ¿Los componentes realizados se almacenan en un Repositorio?

Sí \_\_\_ No \_\_\_

o ¿Se reutilizan estos componentes para la continuidad del trabajo en otras etapas del proyecto o en proyectos futuros?

Sí\_\_\_\_\_ No\_\_\_\_\_ A veces\_\_\_\_\_

7. ¿Se modelan las funcionalidades del proyecto a desarrollar?

Sí\_\_\_\_\_ No\_\_\_\_\_

o ¿Se firman en un acta memorando la aceptación de la documentación entregada al cliente?

Sí\_\_\_\_\_ No\_\_\_\_\_ A veces\_\_\_\_\_ Nunca\_\_\_\_\_

**Anexo 7 Tabla: Resumen de las actividades del proceso 201 Organizar trabajo**

Código	Nombre de la actividad	Descripción	Código	Artefactos de entrada	Código	Tareas	Código	Artefactos de salida	Rol	Bases tecnológicas
20101	Planificar tareas	Responsable de planificar las tareas en la entidad con la cual se buscará organizarla.	201	Lista de objetivos entidad.	2010101	Analizar los objetivos	204	Plan de trabajo	Especialista en tecnología	Microsoft Project.
			202	Lista de tareas.	2010102	Analizar las tareas				
					2010103	Asignar responsables				
20102	Controlar plan de trabajo	Responsable de verificar sistemáticamente las tareas aprobadas en la planificación, permitiendo irle dando cumplimiento y tener constancia de quiénes han cumplido y quiénes no lo han podido hacer con sus respectivas razones	204	Plan de trabajo	2010201	Asegurar las actividades	204	Plan de trabajo.	Especialista en tecnología	Microsoft Project.
					2010202	Chequear estado de las actividades	203	Informe de auditoria		
					2010203	Actualizar el plan				
					2010204	Auditar sistema de trabajo				

### Anexo 8 Resumen de las actividades del proceso 202 Definir el proceso de desarrollo de software.

Código	Nombre de la actividad	Descripción	Código	Artefactos de entrada	Código	Tareas	Código	Artefactos de salida	Rol	Bases tecnológicas
20201	Identificar la línea de producción	Identifica la línea de producción a partir de la visión de la solución informática, de las principales necesidades del cliente y las características básicas del producto a obtener.	205	Documento de descripción del proyecto.	2020101	Solicitar documento de descripción del proyecto.	205	Documento de descripción del proyecto	Especialista en ingeniería de software.	
			206	Lista de proyectos ejecutados	2020102	Identificar línea de producción				
					2020103	Comparar el proyecto con otros existentes anteriormente				
20202	Identificar clasificación de factoría para el proyecto	Define a partir de la descripción del proyecto y la descripción de la clasificación según el alcance, el tipo de factoría que debe asumir el proyecto	205	Documento de descripción del proyecto.	2020201	Analizar el producto a obtener Modelado del Sistema de Información.	205	Documento de descripción del proyecto	Especialista en ingeniería de software.	
			207	Descripción de la Clasificación de la Factorías	2020202	Identificar clasificación de factoría para el proyecto				
20203	Definir el proceso	Determina el flujo de procesos a ejecutar para obtener el producto deseado	205	Documento de descripción del proyecto	2020301	Determinar la metodología.	205	Documento de descripción del proyecto	Especialista en ingeniería de software	Cocomo II.
			206	Lista de proyectos ejecutados	2020302	Determinar las actividades.	209	Documento de descripción	Especialista en procesos	Puntos de función

			208	Documentación de descripción de la metodología	2020303	Determinar las tareas.		n del proceso	Especialista en planificación	
					2020304	Determinar artefactos.			Contador	
					2020305	Determinar los roles.				
					2020306	Caracterizar el rol.				
					2020307	Estimar los recursos.				
					2020308	Organizar grupo de desarrollo				
					2020309	Identificar bases tecnológicas.				
					2020310	Publicar información del proceso				
20204	Identificar riesgos del proceso de desarrollo de software	Se debe evaluar continuamente, para prever las insuficiencias en el proyecto.	205	Documento de descripción del proceso	2020401	Priorizar riesgos que afecten al proceso	210	Plan de gestión de riesgo	Especialista en ingeniería de software	
			206	Lista de proyectos ejecutados	2020402	Establecer plan de gestión de riesgos			Especialista en procesos	

### Anexo 9 Resumen de las actividades del proceso 203 Control de proyecto

Código	Nombre de la actividad	Descripción	Código	Artefactos de entrada	Código	Tareas	Código	Artefactos de salida	Rol	Bases tecnológicas		
20301	Entregar al equipo de proyecto la definición de proceso	El equipo de desarrollo tendrá toda la definición de cómo debe realizar el proceso de desarrollo de software.	209	Documento de descripción del proceso	2030101	Entregar al equipo de proyecto la descripción del proceso	209	Documento de descripción del proceso	Especialista en ingeniería de software			
					2030102	Discutir el flujo de actividades	218	Acta de entrega				
					2030103	Redefinir el proceso						
20302	Controlar el proyecto	Controla la evolución del proyecto, el cumplimiento del cronograma de trabajo y el funcionamiento de equipo de proyecto.	209	Documento de descripción del proceso	2030201	Analizar evolución del proyecto	219	Lista de las no conformidades (mejoras)	Especialista en ingeniería de software	Matriz DAFO		
					211	Cronograma del proyecto	2030202	Evaluar el proceso	220		Matriz DAFO	Especialista en proceso
					212	Cronograma individual	2030203	Analizar DAFO				Especialista en planificación Contador
					213	Documentación del proyecto	2030204					
					LLPP000	Producto	2030205					
20303	Cerrar Proyecto	Analiza si el proyecto cumplió los objetivos previstos y obtuvo el producto esperado por el cliente.	213	Documentación del producto	2030301	Entregar el producto final	221	Documento cierre de proyecto	Especialista en ingeniería de software			
					216	Banco de problemas	2030302	Analizar la entrega al cliente	216		Banco de problemas	Especialista en proceso
					217	Plantilla del proyecto	2030303	Verificar almacenamiento del proyecto	222		Plantilla del personal que se libera	Especialista en planificación

*ANEXOS*

					2030304	Analizar el cierre del proyecto	223	Descripción de mejoras del proyecto	Contador	
					2030305	Liberar el personal				
					2030306	Proponer mejoras				

### Anexo 10 Resumen de roles de la unidad organización del proyecto.

Nombre del rol	Responsabilidades	Competencias		
		Habilidades	Sistema de conocimientos	Valores
Especialista en ingeniería software	Guía, organiza, dirige todo el proceso de definición del proceso de desarrollo de software. Coordina el trabajo del especialista en procesos, en planificación y el contador para lograr una organización del trabajo. Controla y sigue la evolución del proyecto hasta su terminación y entrega.	De liderazgo, comunicación y organización,	Especialista en temas de Ingeniería de software, conocedor de técnicas de organización de la producción y de dirección.	Modestia, sencillez, profesionalidad, responsabilidad, ética profesional.
Especialista en procesos	Evalúa, establece y describe el proceso de desarrollo de software.	Comunicación, locuaz, creativo, capacidad de análisis.	Especialista en temas sobre procesos de software, debe ser o tener conocimiento sobre la ingeniería industrial.	Modestia, sencillez, profesionalidad, responsabilidad, ética profesional.
Especialista en planificación	Guía y evalúa a los planificadores en función de que lleven a cabo el proyecto de la manera establecida. Estima las personas que se necesitan para llevar a cabo el proyecto.	Comunicación, locuaz, creativo, capacidad de análisis.	Especialista en temas sobre procesos de software, debe ser o tener conocimiento sobre la ingeniería industrial. Debe dominar técnicas de estimación de tiempo y personas.	Modestia, sencillez, profesionalidad, responsabilidad, ética profesional.
Contador	Estima y controla el costo del proceso de desarrollo.	Comunicación, locuaz, creativo, capacidad de análisis.	Especialista en temas de contabilidad y economía. Debe dominar técnicas de estimación de costo.	Modestia, sencillez, profesionalidad, responsabilidad, ética profesional.

### Anexo 11 Tabla: Resumen de las políticas de seguridad de los artefactos.

Código	Artefacto/Acciones	Insertar	Modificar
201	Lista de objetivos	Especialista en ingeniería de software	Especialista en ingeniería de software
202	Lista de tareas	Especialista en ingeniería de software	Especialista en ingeniería de software
203	Informe de auditoria	Especialista en ingeniería de software	Especialista en ingeniería de software
204	Plan de trabajo	Especialista en ingeniería de software	Especialista en ingeniería de software
205	Documento de descripción del proyecto	Especialista en ingeniería de software	Especialista en ingeniería de software
206	Lista de proyectos ejecutados	Especialista en ingeniería de software	Especialista en ingeniería de software
207	Descripción de la Clasificación de la Factorías	Especialista en ingeniería de software	Especialista en ingeniería de software
208	Documentación de descripción de la metodología	Especialista en ingeniería de software	Especialista en ingeniería de software
209	Documento de descripción del proceso	Especialista en ingeniería de software	Especialista en ingeniería de software
210	Plan de gestión de riesgo	Especialista en ingeniería de software	Especialista en ingeniería de software

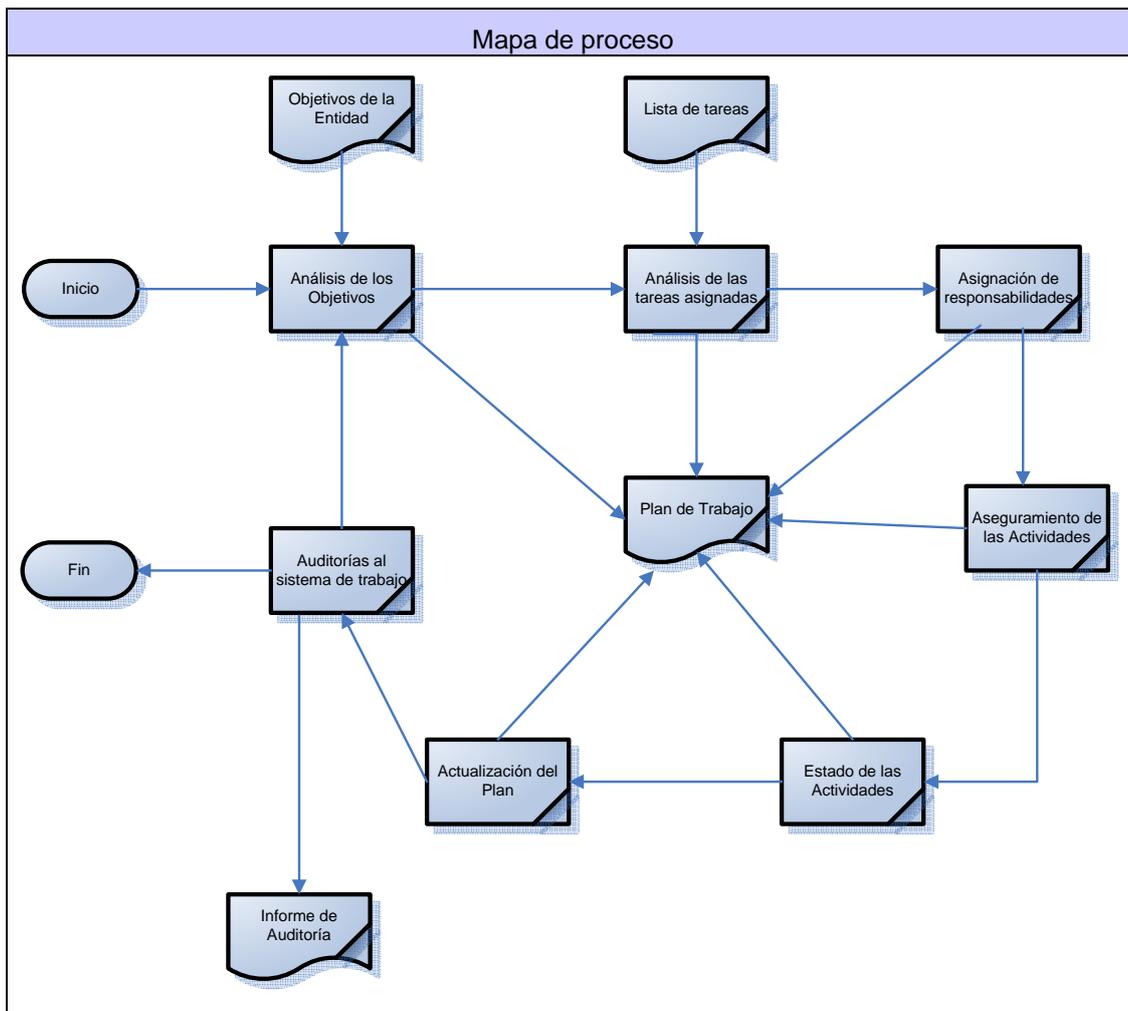
---

211	Cronograma del proyecto	Especialista en ingeniería de software	Especialista en ingeniería de software
212	Cronograma individual	Especialista en ingeniería de software	Especialista en ingeniería de software
213	Documentación del proyecto	Especialista en ingeniería de software	Especialista en ingeniería de software
214	Producto	Especialista en ingeniería de software	Especialista en ingeniería de software
215	Documentación del producto	Especialista en ingeniería de software	Especialista en ingeniería de software
216	Banco de problemas	Especialista en ingeniería de software	Especialista en ingeniería de software
217	Plantilla de roles	Especialista en ingeniería de software	Especialista en ingeniería de software
218	Acta de entrega	Especialista en ingeniería de software	Especialista en ingeniería de software
219	Listado de las no conformidades	Especialista en ingeniería de software	Especialista en ingeniería de software
220	Análisis DAFO	Especialista en ingeniería de software	Especialista en ingeniería de software
221	Documento cierre del proyecto	Especialista en ingeniería de software	Especialista en ingeniería de software
222	Plantilla del personal que se libera	Especialista en ingeniería de software	Especialista en ingeniería de software
223	Descripción de mejora del proyecto	Especialista en ingeniería de software	Especialista en ingeniería de software

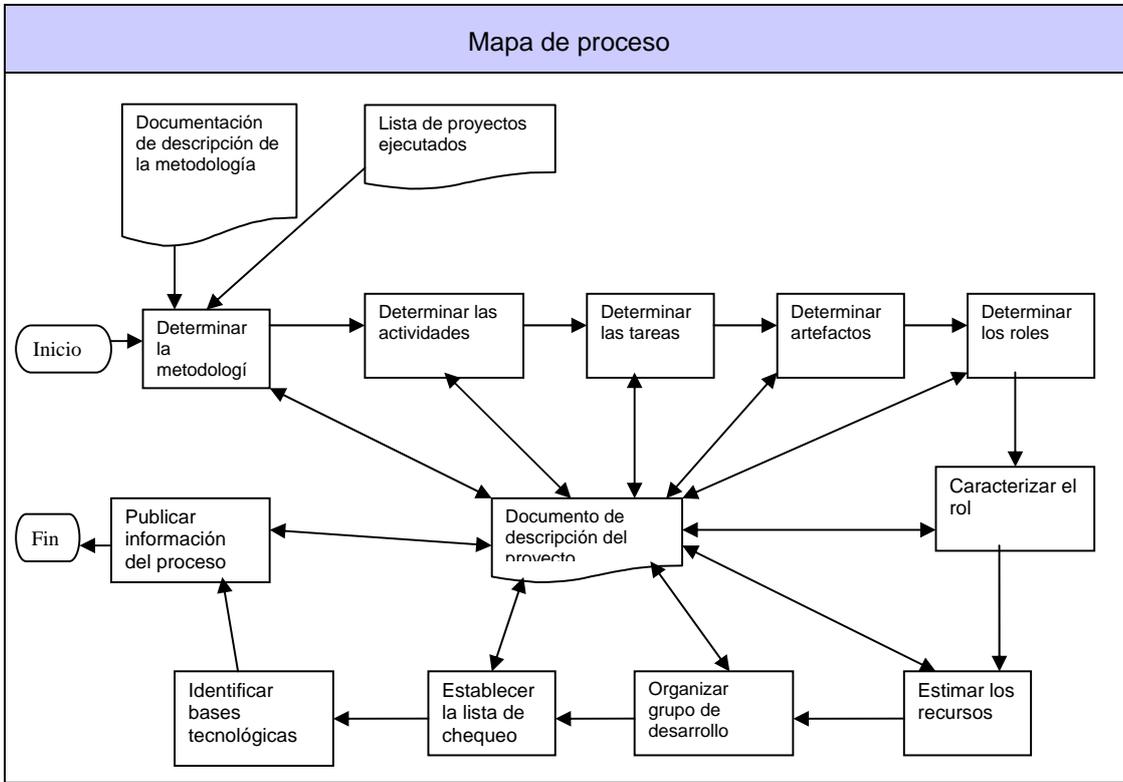
## Anexo 12 Resumen de las Bases tecnológicas

Nombre de la Bases Tecnológicas	Utilidad
El Análisis DAFO	Metodología para el estudio de la situación competitiva de una empresa dentro de su mercado y de las características internas de la misma.
Diagrama causa- efecto	Ayudan a pensar sobre todas las causas reales y potenciales de un suceso o problema.
Microsoft Project	Es una aplicación que ayuda al usuario a crear planes de proyectos, comunicarlos a otros usuarios y adaptarse a los cambios a medida que éstos se van produciendo.
COCOMO II	Es un método que ayuda a lograr una estimación del costo, del tiempo y las personas necesarias para tener un proceso estandarizado, eficiente y repetible.
Puntos de función	Es un método que ayuda a lograr una estimación del costo, del tiempo y las personas necesarias para tener un proceso estandarizado, eficiente y repetible.

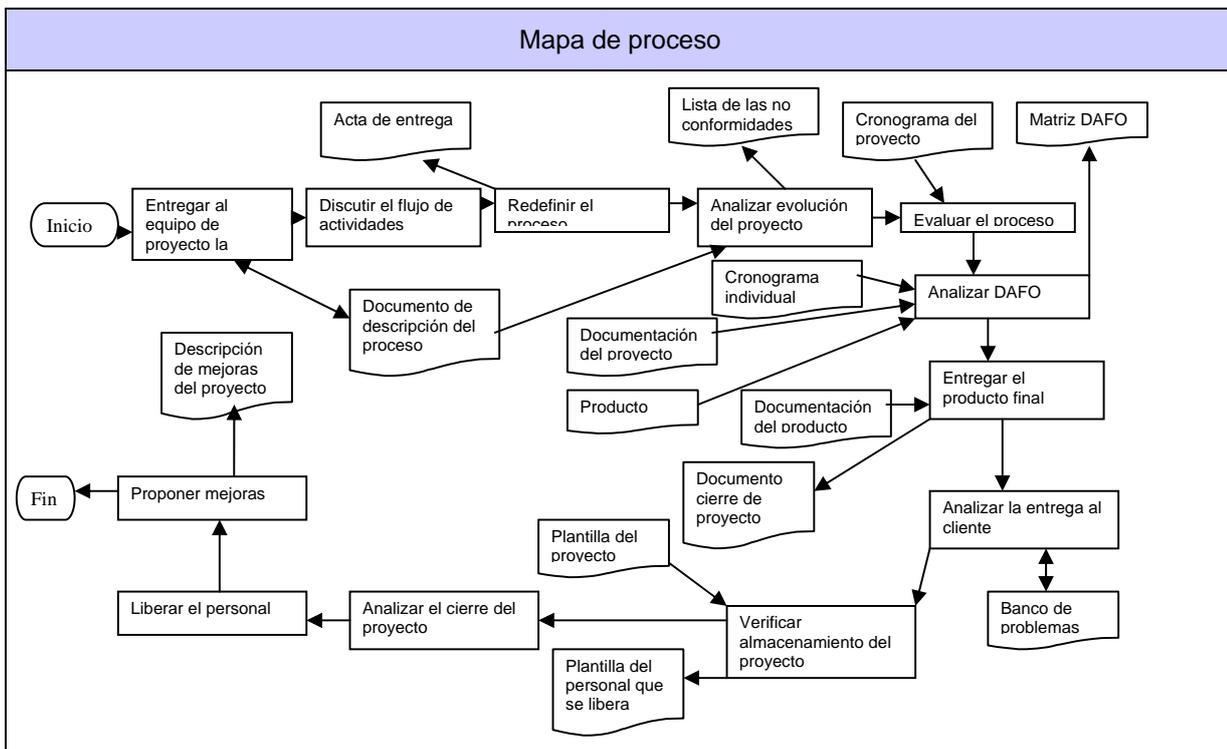
Anexo 13 Tabla: Mapa del proceso 201 Organización del trabajo.



**Anexo 14 Mapa de la actividad 202 Definir el proceso.**



Anexo 15 Mapa de procesos 203 Controlar proyecto



## Anexo 16 Plantilla 201 Lista de objetivos

<b>Lista de objetivos</b>	Día	Mes	Año
Factoría de Software:	<b>00</b>	<b>00</b>	<b>0000</b>

Código: 201

Autor: \_\_\_\_\_

Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

### 1. Introducción

<Breve introducción a la organización, organismo al que pertenece, perspectivas futuras>

### 2. Alcance

- <Describir la visión y el alcance de la factoría.>
- <El objetivo general>
  - <Lineamiento 1>
    - <Indicadores>
    - <Acciones>
  - <Lineamiento 2>
    - <Indicadores>
    - <Acciones>
  - <Lineamiento n>
    - <Indicadores>

## Anexo 17 Plantilla 202 Lista de tareas

<b>Lista de tareas</b>	<b>Día</b>	<b>Mes</b>	<b>Año</b>
Factoría de Software:	<b>00</b>	<b>00</b>	<b>0000</b>

Código: 202

Autor: \_\_\_\_\_

Descripciones			
Datos de creación		Fecha	
		Versión	
Datos sobre el acceso		Todos	
		Acceso restringido	
		Sin acceso	
Disponibilidad:		Si	
		No	
Cambio	Indicadores a medir	Área de impacto	Resultados esperado

## Anexo 18 Plantilla 203 Informe de auditoria

<b>Informe de auditoria</b>	Día	Mes	Año
Factoría de Software:	<b>00</b>	<b>00</b>	<b>0000</b>

Código: 203

Autor: \_\_\_\_\_

Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

### 1. Objetivos

- <Describir la visión y el alcance de la auditoria.>
- <El objetivo general>
  - <Lineamiento 1>
    - <Indicadores>
    - <Acciones>
  - <Lineamiento 2>
    - <Indicadores>
    - <Acciones>
  - <Lineamiento n>
    - <Indicadores>
    - <Acciones>

## 2. Plan de ejecución de la auditoria

No.	Tareas	Fecha Comienzo	Fecha Fin	Responsable

## 3. Resultados de la auditoria

- Se reflejan los resultados de la auditoria cualitativa y cuantitativamente por indicadores.

## Anexo 19 Plantilla 204 Plan de trabajo

Plan de trabajo	Día	Mes	Año
	Factoría de Software:	<b>00</b>	<b>00</b>

Código: 204

Autor: \_\_\_\_\_

Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

Fecha: 00/00/0000

Período del plan: 00/00/0000 - 00/00/0000

No.	Tareas	Fecha Comienzo	Fecha Fin	Responsable

## Anexo 20 Plantilla 205 Descripción del proyecto

Descripción del proyecto	Día	Mes	Año
Factoría de Software:	<b>00</b>	<b>00</b>	<b>0000</b>

Código: 205

Autor: \_\_\_\_\_

Código del proyecto: \_\_\_\_\_

Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

### Introducción

<Breve introducción al producto que se desee>

### Propósito

<Propósito que persigue el cliente con el producto>

### Alcance

- <Describir la visión y el alcance del producto.>
- <El objetivo general>
  - <Lineamiento 1>
    - <Indicadores>
    - <Acciones>
  - <Lineamiento 2>
    - <Indicadores>
    - <Acciones>

- <Lineamiento n>
  - <Indicadores>
  - <Acciones>

## Definiciones, Acrónimos, y Abreviaturas

<Se presenta todas las definiciones y acrónimos que pueden ayudar al mejor entendimiento del producto solicitado>

## Posición

<Posición actual del cliente sobre el tema a analizar>

## Establecimiento del problema

Problema	<Problema a analizar>
Afectados	<A quienes o a que repercute>
Impacto que provoca	<impacto del problema>
Una solución satisfactoria pudiera implicar	<Propuesta de solución a valorar>

## Resumen de los clientes

Nombre	Descripción	Responsabilidades	Involucrados
<Organismos institución o entidad>	<Breve descripción del cliente>	<Responsabilidad que establece al solicitar el servicio>	<Persona que representara al cliente ante la entidad>

## Necesidades del cliente:

Necesidades	Orden de prioridad

## Anexo 21 Plantilla 206 Lista de proyectos ejecutados

<b>Lista de proyectos ejecutados</b>	Día	Mes	Año
Factoría de Software:	<b>00</b>	<b>00</b>	<b>0000</b>

Código: 206

Autor: \_\_\_\_\_

Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

Proyecto	Código proyecto	Línea de producción	Componentes	Ubicación

## Anexo 22 Plantilla 207 Descripción de la Clasificación de la Factorías

Descripción de la Clasificación de la Factorías	Día	Mes	Año
	Factoría de Software:	<b>00</b>	<b>00</b>

Código: 207

Autor: \_\_\_\_\_

Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

Tipos de Factoría	Descripción
Factoría de implementación de software	Es la menor de las clasificaciones en ella solamente se implementan y prueban programas, componentes y/o sistemas.
Factoría de diseño e implementación	Además de los flujos de la factoría de implementación realiza el análisis y diseño
Factoría de modelado, diseño e implementación	Contiene los elementos de la factoría de análisis y diseño además de ellos modela el sistema según los requerimientos.
Factoría de ciclo completo	Es la mayor de las clasificaciones y realiza todos los flujos de trabajo.

## Anexo 23 Plantilla 209 Documento de descripción del proceso

Documento de descripción del proceso	Día	Mes	Año
Factoría de Software:	<b>00</b>	<b>00</b>	<b>0000</b>

Código: 209

Autor: \_\_\_\_\_

Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

- **Introducción**
- **Procesos**
  - Proceso
  - Definir artefactos de entrada
  - Definir artefactos de salida
  - Definir las actividades
- **Personas**
  - Organización
- **Repositorio**
- **Bases tecnológicas**
- **Anexos**

## Anexo 24 Plantilla 210 Plan de gestión de riesgo

<b>Plan de gestión de riesgo</b>	Día	Mes	Año
Factoría de Software:	<b>00</b>	<b>00</b>	<b>0000</b>

Código: 210

Autor: \_\_\_\_\_

Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

### Riesgos

Código	Riesgo	Tipo de Riesgo	Impacto	Descripción	Probabilidad	Efectos

### Plan de Riesgo

Código	Riesgo	Probabilidad	Impacto	Mitigación del riesgo	Monitoreo del riesgo	Administración del riesgo

## Anexo 25 Plantilla 216 Banco de problemas

<b>Banco de problemas</b>	Día	Mes	Año
Factoría de Software:	<b>00</b>	<b>00</b>	<b>0000</b>

Código: 216

Autor: \_\_\_\_\_

Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

Fecha	Necesidad	Prioridad	Inquietud	Problema	Solución propuesta

## Anexo 26 Plantilla 217 Plantilla del proyecto

<b>Plantilla del proyecto</b>	Día	Mes	Año
Factoría de Software:	<b>00</b>	<b>00</b>	<b>0000</b>

Código: 217

Autor: \_\_\_\_\_

Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

**Plantilla del proyecto**

Fecha: \_\_/\_\_/\_\_\_\_

Proyecto: \_\_\_\_\_

Jefe de Proyecto: \_\_\_\_\_

**Integrantes del proyecto**

#	Nombre	Grupo	Rol
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

## Anexo 27 Plantilla 218 Acta de entrega

<b>Acta de entrega</b>	Día	Mes	Año
Factoría de Software:	<b>00</b>	<b>00</b>	<b>0000</b>

Código: 218

Autor: \_\_\_\_\_

Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

**Acta de entrega**

Fecha: \_\_/\_\_/\_\_\_\_

Proyecto: \_\_\_\_\_

Líder del proyecto: \_\_\_\_\_

Especialista en ingeniería de software: \_\_\_\_\_

**Documentos que entrega:**

- 

Firma del líder del proyecto: \_\_\_\_\_

Firma del especialista en ingeniería de software: \_\_\_\_\_

## Anexo 28 Plantilla 220 Matriz DAFO.

Análisis DAFO	Día	Mes	Año
	Factoría de Software:	<b>00</b>	<b>00</b>

Código: 220

Autor: \_\_\_\_\_

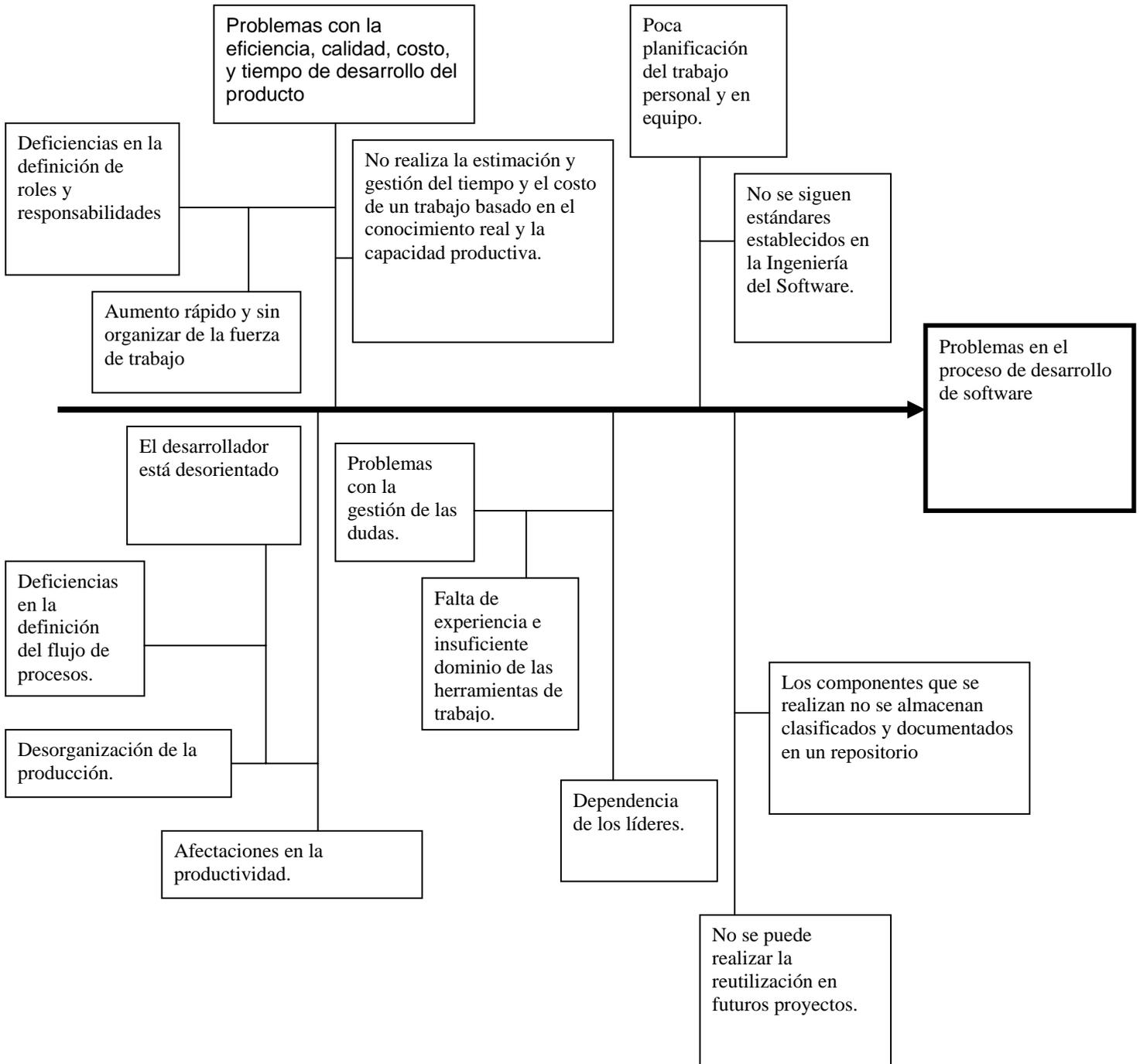
Descripciones		
Datos de creación	Fecha	
	Versión	
Datos sobre el acceso	Todos	
	Acceso restringido	
	Sin acceso	
Disponibilidad:	Si	
	No	

Tema del análisis: (defina aquí el tema a analizar)	
Fortalezas	Debilidades
<ul style="list-style-type: none"> <li>¿Ventajas de la propuesta?</li> <li>¿Capacidades?</li> <li>¿Ventajas competitivas?</li> <li>¿PUV's (propuesta única de vetas)?</li> <li>¿Recursos, activos, gente?</li> <li>¿Experiencia, conocimiento, datos?</li> <li>¿Reservas financieras, retorno probable?</li> <li>¿Marketing – alcance, distribución, avances?</li> <li>¿Aspectos innovadores?</li> <li>¿Ubicación geográfica?</li> <li>¿Precio, valor, calidad?</li> <li>¿Acreditaciones, calificaciones, certificaciones?</li> <li>¿Procesos, sistemas, TI, comunicaciones?</li> <li>¿Cultural, actitudinal, de comportamiento?</li> <li>¿Cobertura gerencial, sucesión?</li> </ul>	<ul style="list-style-type: none"> <li>¿Desventajas de la propuesta?</li> <li>¿Brechas en la capacidad?</li> <li>¿Falta de fuerza competitiva?</li> <li>¿Reputación, presencia y alcance?</li> <li>¿Aspectos Financieros?</li> <li>¿Vulnerabilidades propias conocidas?</li> <li>¿Escala de tiempo, fechas tope y presiones?</li> <li>¿Flujo de caja, drenaje de efectivo?</li> <li>¿Continuidad, robustez de la cadena de suministros?</li> <li>¿Efectos sobre las actividades principales, distracción?</li> <li>¿Confiabilidad de los datos, predictibilidad del plan?</li> <li>¿Motivación, compromiso, liderazgo?</li> <li>¿Acreditación, etc.?</li> <li>¿Procesos y sistemas, etc.?</li> <li>¿Cobertura gerencial, sucesión?</li> </ul>
Oportunidades	Amenazas

¿Desarrollos del mercado?  
¿Vulnerabilidades de los competidores?  
¿Tendencias de la industria o de estilo de vida?  
¿Desarrollos tecnológicos e innovaciones?  
¿Influencias globales?  
¿Nuevos mercados, verticales, horizontales?  
¿Mercados objetivo nicho?  
¿Geografía, exportación, importación?  
¿Nuevas propuestas únicas de venta?  
¿Tácticas - sorpresa, grandes contratos, etc.?  
¿Desarrollo de negocios o de productos?  
¿Información e investigación?  
¿Sociedades, agencias, distribución?  
¿Volúmenes, producción, economías?  
¿Influencias estacionales, del clima, o de la moda?

¿Efectos políticos?  
¿Efectos legislativos?  
¿Efectos ambientales?  
¿Desarrollos de TI?  
¿Intenciones de los competidores?  
¿Demanda del mercado?  
¿Nuevas tecnologías, servicios, ideas?  
¿Contratos y alianzas vitales?  
¿Mantener las capacidades internas?  
¿Obstáculos enfrentados?  
¿Debilidades no superables?  
¿Pérdida de personal clave?  
¿Respaldo financiero sostenible?  
¿Economía – local o extranjera?  
¿Influencias estacionales, del clima, o de la moda?

**Anexo 29 Diagrama de Causa - Efecto.**



## GLOSARIO DE TÉRMINOS

**Actividad:** Conjunto de operaciones o tareas propias de una persona o entidad que permite que el trabajo que se realiza sea descrito y entendido de manera precisa por aquellos que tienen que ejecutarlo.

**Artefacto:** Es un término general, para cualquier tipo de información creada, producida, cambiada o utilizada por los trabajadores en el desarrollo del sistema.

**Calidad:** Conjunto de propiedades y características de un producto o servicio que le confieren su aptitud para satisfacer unas necesidades explícitas o implícitas.

**Calidad del software:** Es el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario. La calidad del software ha pasado de una simple inspección y detección de errores a un cuidado total en su proceso de fabricación, desarrollo y mantenimiento; y es que el correcto funcionamiento de éste es fundamental para el óptimo comportamiento de los sistemas informáticos.

**Comunicación:** Proceso de intercambio de mensaje entre dos elementos, un emisor y un receptor. Este intercambio de mensaje se realiza a través de un canal o medio, que es un dispositivo físico de transmisión. Para el intercambio de mensajes es imprescindible que el emisor y el receptor tengan un conocimiento o repertorio en común y así puedan realizar la codificación y decodificación de mensajes en forma congruente, lo que posibilita generar en el receptor un incremento del conocimiento.

**Costo:** Es el sacrificio económico incurrido en la obtención de activos, con la finalidad de obtener beneficios futuros.

**Costo intangible:** Son costos asociados a: el conocimiento, las habilidades, el capital intelectual, es un enfoque novedoso.

**Diseño detallado:** Traduce el diseño lógico en una solución implementable y costo-efectiva o económica.

**Especificación lógica:** Traduce los escenarios de uso creados en el diseño conceptual en un conjunto de objetos de negocio y sus servicios. Se convierte en parte en la especificación funcional que se usa en el diseño físico. Es independiente de la tecnología. El diseño lógico refina, organiza y detalla la solución de negocios y define formalmente las reglas y políticas específicas de negocios.

**Evento sistémico:** Es un evento de alto nivel generado por un actor externo. Se asocia a operaciones del sistema: las que se emiten en respuesta a los eventos del sistema. Por ejemplo, cuando un cajero que usa una terminal de punto de venta oprime el botón "terminar venta", está generando un evento sistémico que

indica que "la venta ha terminado". Del mismo modo, cuando alguien que usa un editor de texto pulsa el botón "revisar ortografía", está produciendo un evento del sistema.

**Estándar:** Lo que es establecido por la autoridad, la costumbre o el consentimiento general. En este sentido se utiliza como sinónimo de norma.

**Equipo de desarrollo:** Es un grupo de trabajo constituido por una serie de profesores, investigadores, colaboradores y alumnos unidos en la ilusión de acometer un determinado proyecto o avanzar en el conocimiento y en la investigación teórica y aplicada.

**Factoría:** Cualquier tipo de fábrica o industria, es decir, a cualquier tipo de instalación en la cual se produce la transformación de materias primas o productos semiterminados en otros productos.

**Factoría de software:** Organización que aplica conceptos de ingeniería (métricas de tiempos, errores, conceptos de calidad total, reutilización de componentes software, alta productividad, etc.) a la producción de software. Fundamentalmente, lo que se busca es lograr que el proceso de crear software deje de ser artesanal, para convertirse en un proceso industrial, documentado y repetible.

**Feedback o retroalimentación:** Es un elemento de evaluación que permite al emisor saber si el mensaje enviado es recibido y si fue interpretado correctamente por el receptor.

**Gestión:** Gestión es la acción y efecto de gestionar o la acción o efecto de administrar. Comprende todas las actividades de una organización que implican el establecimiento de metas u objetivos, así como la evaluación de su desempeño y cumplimiento; además del desarrollo de una estrategia operativa que garantice la supervivencia de la misma, según al sistema social correspondiente.

**Gestión de los Recursos Humanos:** Función administrativa en la que se maneja el reclutamiento, asignación, capacitación y el desarrollo de los miembros de una organización o empresa.

**Herramientas CASE:** (del inglés Computer Aided Software Engineering, que viene a significar Ingeniería de Software asistida por ordenador). Son instrumentos o sistemas automatizados que brindan soporte a las actividades de producción de software.

**Humano:** Perteneciente al hombre o propio de él.

**Ingeniería de Software:** Se puede definir como el tratamiento sistemático de todas las fases del ciclo de vida del software.

**Inteligencia:** Se define como la capacidad de adquirir conocimiento o entendimiento y de utilizarlo en situaciones novedosas para la toma de decisiones.

**Intranet:** Es una adaptación de las mismas tecnologías que existen en Internet, para que sean utilizadas dentro de la red interna de una empresa u organización de forma tal que sus miembros puedan intercambiar información de todo tipo, utilizando el Web como interfaz común.

**Modelo de Factoría de Software:** La forma en que se han llevado a la práctica el enfoque de factoría de software por distintas empresas y entidades que lo han adaptado.

**Organización del proceso:** Es la forma en que distribuyen las tareas o actividades dentro del equipo de desarrollo, es asignar a cada persona del equipo el rol de acuerdo a las capacidades mostradas y velar por el cumplimiento de las tareas.

**Planificación:** La planificación comporta el establecimiento de objetivos, y la decisión sobre las estrategias y las tareas necesarias para alcanzarlas.

**Proceso de desarrollo de software:** Es un conjunto de actividades y resultados asociados que producen un producto de software.

**Productividad del trabajo:** Es un indicador que refleja que tan bien se están usando los recursos de una economía en la producción de bienes y servicios

**Proyecto:** Combinación de recursos humanos y no humanos reunidos en una organización temporal para conseguir un propósito, tiene un punto de de comienzo definido y con objetivos definidos mediante los que se identifican.

**Recursos:** Conjunto de elementos disponibles para resolver una necesidad o llevar a cabo una tarea.

**Repositorio de componentes:** Biblioteca de componentes software reutilizables. Los componentes almacenados en el repositorio deben tener una representación estándar y estar bien documentados, siendo el sistema gestor de la biblioteca el encargado de organizar, proteger y gestionar dichos componentes.

**Tiempo:** Variable que no podemos modificar. No podemos, alargarlo, estirarlo, comprarlo o detenerlo. Sin embargo, podemos controlarlo. Mientras mejor controlemos el tiempo más eficiente será nuestro trabajo.

## **GLOSARIO DE SIGLAS**

**UCI:** Universidad de Ciencias Informáticas

**CMM:** El Modelo de Capacidad y Madurez o CMM (Capability Maturity Model), es un método de definir y gestionar los procesos a realizar por una organización. Fue desarrollado inicialmente para los procesos relativos al software por la Universidad Carnegie-Mellon para el SEI (Software Engineering Institute).

**CMMI:** Modelo de Madurez de Capacidades Integrado. Está dedicado a las actividades de ingeniería de sistemas.

**IEEE:** Corresponde a las siglas de The Institute of Electrical and Electronics Engineers, el Instituto de Ingenieros Eléctricos y Electrónicos, una asociación técnico-profesional mundial dedicada a la estandarización, entre otras cosas.

**ISO 9001:** La norma ISO 9001, es un método de trabajo, con el fin de mejorar la calidad y satisfacción de cara al consumidor. Esta dirigido a mejorar los aspectos organizativos de una empresa.

**ISO:** Organización Internacional de Estándares.

**TIC:** Tecnologías de la Información y las Comunicaciones.

**IP:** Infraestructura Productiva.

**OACE:** Organismos de la Administración Central del Estado.

**RUP:** Proceso Unificado de Racional.

**SEI:** Instituto de Ingeniería de Software. El SEI es un centro de investigación y desarrollo patrocinado por el departamento de defensa de los Estados Unidosy gestionado por la Carnegie-Mellon.

**IEEE:** Corresponde a las siglas de The Institute of Electrical and Electronics Engineers, el Instituto de Ingenieros Eléctricos y Electrónicos, una asociación técnico-profesional mundial dedicada a la estandarización, entre otras cosas.