

ESTANDARIZACIÓN DEL PROCESO SOFTWARE PARA UNA EMPRESA
DE CONTACT CENTER

JOSÉ DE JESÚS LÓPEZ JIMÉNEZ

UNIVERSIDAD EAFIT
DEPARTAMENTO DE INFORMÁTICA Y SISTEMAS
MEDELLIN
2009

ESTANDARIZACIÓN DEL PROCESO SOFTWARE PARA UNA EMPRESA
DE CONTACT CENTER

JOSÉ DE JESÚS LÓPEZ JIMÉNEZ

Trabajo de grado para optar al título de
Ingeniero de Sistemas

Asesor

JUAN DAVID CARRASQUILLA ARANGO
Especialista en gerencia de proyectos informáticos

UNIVERSIDAD EAFIT
DEPARTAMENTO DE INFORMÁTICA Y SISTEMAS
MEDELLIN
2009

Nota de Aceptación

Presidente del Jurado

Jurado

Jurado

DEDICATORIA

A mis padres por haber luchado tanto y a pesar de los dolores haberse mantenido en este proyecto llamado José.

A mi esposa, Gracias por amarme como lo has hecho, gracias por amarme con ese amor que transforma.

A mi hija, Espero que algún día esto te motive a luchar por tus sueños.

A mi madre como paga por la dedicatoria que hizo en su trabajo de grado para ser secretaria general del CES. Espero que esta dedicatoria te haga tan feliz como a mí me hizo la tuya.

AGRADECIMIENTOS

A Dios como el todo de mi vida.

A mi familia: mis cuatro padres, mi esposa, mi hija, mis abuelos, mis tíos, don Jaime, doña Beatriz y a mis amigos Mauricio y Claudia.

Agradezco a EMTELCO en cabeza de Juan David Carrasquilla por permitirme ser parte del proceso de crecimiento de esta empresa.

CONTENIDO

INTRODUCCIÓN

1. Modelos del proceso Software	1
1.1 Modelo lineal secuencial o en cascada	2
1.2 Proceso de software por prototipos	4
1.3 Proceso software basado por componentes.....	6
1.4 Proceso del software por entregas incrementales	8
1.5 Proceso software en espiral	10
2. Método, Metodología y Métrica	13
2.1 Método.....	13
2.2 Metodología	13
2.2.1 Metodologías estructuradas	14
2.3 Métricas del software	15
2.3.1 Métrica de Punto Función	17
3. Análisis situación actual área de desarrollo de aplicaciones empresa de contact-center emtelco s.a.....	19
3.1 Visión General	19
3.2 Dinámica de las aplicaciones EMTELCO S.A.....	22
3.3 Análisis Oficina de desarrollo de aplicaciones	23
3.3.1 Generalidades.....	23
3.3.2 Dimensionamiento y tamaño de los proyectos	25

3.3.3	Enfoque de empoderamiento de los proyectos y perfil del empleado del área de desarrollo de aplicaciones Emtelco S.A.....	27
3.3.4	A quienes están dirigidos los proyectos informáticos Emtelco S.A	28
3.4	Análisis de área desde el proceso de cada analista	29
3.4.1	Acerca del uso de un Modelo de Proceso de desarrollo	30
4.	Propuesta guía para el proceso de desarrollo de software EMTELCO S.A	41
4.1	Preliminares.....	41
4.2	Guía de desarrollo	43
4.2.1	Condiciones iniciales.....	43
4.3	Propuesta guía por etapas de desarrollo	44
4.3.1	Pre análisis: Elicitación de requisitos:	44
4.3.2	Análisis.....	45
4.3.3	Definición del proceso software.....	47
4.3.4	Diseño	49
4.3.5	Desarrollo.....	50
4.3.6	Pruebas e Implementación.....	51
4.4	Recomendaciones Adicionales	53
	Conclusiones.....	55
	Bibliografía.....	57
	Anexos	

LISTA DE FIGURAS

Figura 1. Ciclo de vida del Software.....	1
Figura 2. Modelo Lineal Secuencial.....	2
Figura 3. El paradigma de construcción por prototipos.....	4
Figura 4. El proceso software basado por componentes.....	6
Figura 5. Proceso del software por entregas incrementales.....	9
Figura 6. Proceso del software en espiral.....	11
Figura 7. Relación empleados EMTELCO S.A vs personal área de desarrollo de aplicaciones.....	20
Figura 8. Distribución de proyectos por tipo de aplicaciones.....	24
Figura 9. Distribución de proyectos por tamaño.....	26
Figura 10. Distribución de proyectos teniendo en cuenta el método de trabajo con el que se desarrolla.....	27
Figura 11. Distribución de proyectos teniendo en cuenta el propósito de uso.....	30
Figura 12. Uso de un proceso de desarrollo de software por parte de los analistas del área de desarrollo.....	31
Figura 13. Familiaridad de la oficina de desarrollo con los requisitos EMTELCO S.A.....	34
Figura 14. Reusabilidad de los proyectos EMTELCO S.A.....	36
Figura 15. Aplicabilidad del proceso de desarrollo por entregas parciales en EMTELCO S.A.....	37
Figura 16. Aplicabilidad del proceso de desarrollo por prototipos en EMTELCO S.A.....	39
Figura 17. Modelos de distribución de las aplicaciones EMTELCO S.A.....	40

INTRODUCCIÓN

La tendencia de las compañías en la sociedad actual para lograr que día a día los procesos que se realizan sistemáticamente sean cada vez más eficientes y susceptibles de mejorar conllevan a que al interior de estas cada uno de los componentes de su cadena de valor sea puesto en la mira, en aras de alcanzar el mejoramiento que les permita conseguir ser cada día más eficientes y funcionales. El fenómeno antes propuesto, sumado a la fructífera carrera de muchas personas en el mundo por realizar las cosas de una manera “mejor que...” hacen que el escenario de mejoramiento de las compañías se base en mejores prácticas y en la acomodación de éstas en un entorno de trabajo propio, con una única premisa: El mejoramiento continuo.

De esta manera y fijando nuestra mirada específicamente en el proceso de desarrollo de software, ámbito en el cual cada vez existen más propuestas de mejoramiento continuo, se puede observar al interior de una compañía como EMTELCO S.A, que el proceso de desarrollo de software pueda ser llevado a un nivel superior de eficiencia y calidad como principal premisa. Este proyecto promueve el logro de este objetivo a través de un análisis y el desarrollo de una guía a la medida para obedecer a la dinámica propia y genuina del área de desarrollo de la compañía y su proceso específico de desarrollo de aplicaciones.

En EMTELCO S.A el análisis de la situación actual nos puede guiar a que la primera falencia por atacar es el hecho de no tener establecido un proceso. Esta es la situación de una gran mayoría de compañías: Todos los procesos se realizan de acuerdo a una directriz o forma en la que se hizo o se hace, el problema es que tan efectiva es dicha forma de hacer las cosas y más aún que tan claro es esto para los implicados en dicho proceso. Precisamente

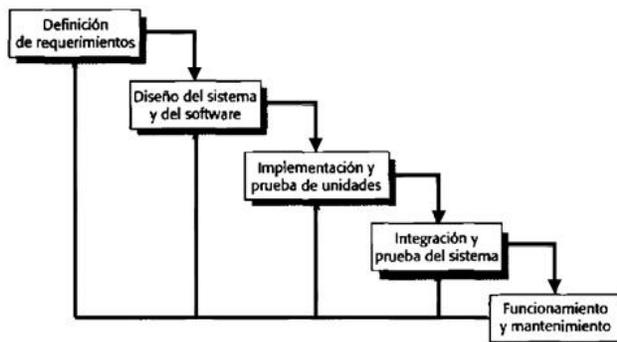
esto es uno de los mayores aportes que proveen los procesos de certificación que vienen adoptando muchas empresas; El personal tiene claro como se debe realizar cada proceso, cuando, donde, quien lo debe realizar, donde se encuentra la información, etc.

De esta manera, una guía metodológica es un primer paso para el buen desarrollo de la labor tecnológica de EMTELCO S.A pero no deberá ser un proceso estancado. El análisis y el uso de metodologías o mejores prácticas son un buen indicio para hacer una tarea, ya que garantizan que existen casos de éxito de su aplicación, empero las dinámicas sólo pueden ser experimentadas a fondo en la marcha, en ese sentido esta propuesta deberá ser alimentada en el día a día.

1. MODELOS DEL PROCESO SOFTWARE

El modelo de proceso de desarrollo de software es quizás la pieza más importante de este engranaje conocido como ingeniería de software. Existen varios modelos para el proceso de desarrollo software. Los modelos están conformados por etapas que son generales a todos los enfoques. Las diferencias están básicamente en los tiempos en los cuales se realizan dichas etapas, la simultaneidad, la prioridad que se le da a cada etapa, entre otros elementos. Otras características se pueden observar en detalle en la definición de cada uno de los modelos.

Figura 1. Ciclo de vida del Software



Fuente: Ian Sommerville. Ingeniería del Software, Séptima Edición

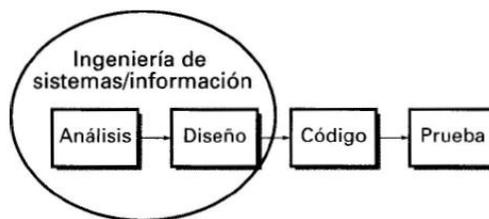
Algo importante en el análisis del proceso software es que los procesos no son excluyentes y un proceso bien puede ser la unión de varios de ellos.

1.1 **MODELO LINEAL SECUENCIAL O EN CASCADA**¹

El modelo en lineal secuencial es también conocido como el ciclo de vida del software, está conformado por las etapas de Análisis de requerimientos, Diseño, Codificación y pruebas. Cada etapa tiene dependencias de finalización y orden la una de la otra. En ese orden de ideas no se podrá iniciar la etapa de codificación sino no se ha pasado ya por la etapa de análisis y diseño previamente y respectivamente.

Debido a la forma en la que este proceso aborda la solución de proyectos software tiene algunas ventajas y falencias.

Figura 2. Modelo Lineal Secuencial



Fuente: Roger S. Pressman. Ingeniería del Software, Un enfoque práctico. Quinta edición. McGraw Hill

Entre otras las ventajas y desventajas de este modelo son:

- Los proyectos reales raras veces siguen el modelo secuencial que propone el modelo. Aunque el modelo lineal puede acoplarse

¹ Roger S. Pressman. Ingeniería del Software, Un enfoque práctico. Quinta edición. McGraw Hill

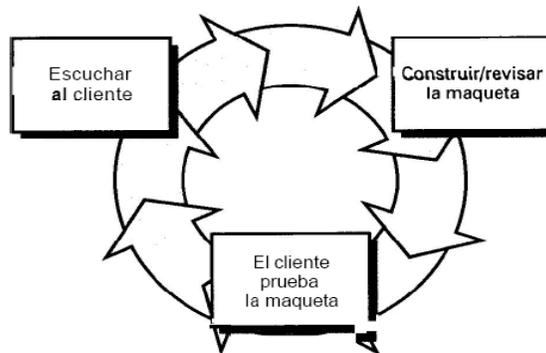
interacción, lo hace indirectamente. Como resultado, los cambios pueden causar confusión cuando el equipo del proyecto comienza.

- Este modelo es ampliamente usado en la industria del software y pese a que tiene algunas dificultades como las mencionadas arriba es mejor que no aplicar ningún modelo de proceso para el desarrollo de software.
- A menudo es difícil que el cliente exponga explícitamente todos los requisitos. El modelo lineal secuencial lo requiere y tiene dificultades a la hora de acomodar la incertidumbre natural al comienzo de muchos proyectos.
- El cliente debe tener paciencia. Una versión de trabajo del (los) programa(s) no estará disponible hasta que el proyecto esté muy avanzado. Un grave error puede ser desastroso si no se detecta hasta que se revisa el programa.
- Es difícil afrontar cambios propuestos por el cliente en fases diferentes de la fase de análisis y definición de requisitos y entre más adelantado esté el proyecto más costosos son los cambios o modificaciones.

1.2 PROCESO DE SOFTWARE POR PROTOTIPOS²

El paradigma de desarrollo por prototipos permite refinar sistemas complejos con base en un sistema mínimo definido al principio del proceso de la especificación del sistema y de los cuales el cliente no tiene la definición completa de requisitos.

Figura 3. El paradigma de construcción por prototipos



Fuente: Roger S. Pressman. Ingeniería del Software, Un enfoque práctico. Quinta edición. McGraw Hill

El proceso comienza con la definición de requisitos por parte del cliente y así se llega a los objetivos globales del sistema. Con base en los requisitos conocidos se diseña un primer acercamiento a la solución el cual se implementa como una versión funcional del software. Dicho prototipo es evaluado por el cliente y así definir nuevos alcances y nuevos

² Roger S. Pressman. Ingeniería del Software, Un enfoque práctico. Quinta edición. McGraw Hill

requisitos. Por medio de dichas iteraciones, de un prototipo a otro se logra el sistema completo.

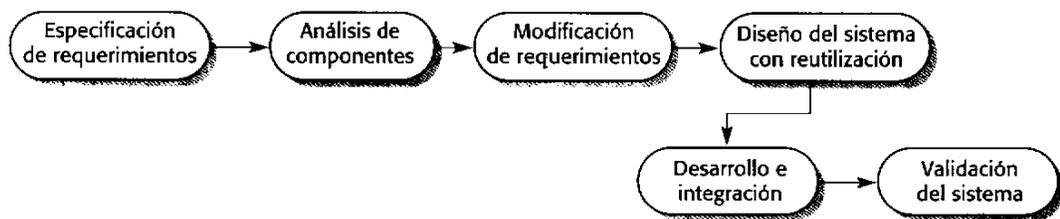
Este enfoque tiene ventajas y desventajas entre otras están:

- El cliente ve lo que parece una versión del software y no tiene en cuenta que para realizarse un prototipo rápido, se rebaja la calidad del producto software de esa manera el cliente puede no entender que se debe realizar de nuevo el sistema y querrá tener como versión final de su sistema el prototipo ya desarrollado.
- De forma demasiado frecuente la gestión de desarrollo del software es muy lenta.
- Documentar el desarrollo suele ser tedioso y puede perderse control por parte de los jefes de proyectos debido a la cantidad de versiones desarrolladas.
- Por el otro lado el paradigma por prototipos también cuenta con una aceptación amplia debido a que el cliente le gusta ver versiones inmediatas de su sistema. También suele ser muy efectivo para el manejo y la identificación de errores en etapas tempranas del proceso de desarrollo.

1.3 PROCESO SOFTWARE BASADO POR COMPONENTES³

Es una práctica habitual por parte de los desarrolladores la inclusión de códigos desarrollados previamente por el equipo de desarrollo. Esta práctica informal no depende del proceso software que se esté siguiendo en el desarrollo de un sistema de esta manera dicha práctica haya una definición formal en el procesos software basada por componentes. Aunque en la definición de requisitos y la etapa de validación es comparable con los demás modelos en las etapas intermedias los cambios son significativos.

Figura 4. El proceso software basado por componentes



Fuente: Ian Sommerville. Ingeniería del Software, Séptima Edición

Este consta de 4 etapas de desarrollo:

- **Análisis de componentes:** Con base en los requerimientos del cliente se analizan los componentes factibles de ser implementados para dicha necesidad.

³ Ian Sommerville, (2005). Ingeniería del Software. Addison Wesley. Séptima Edición

- **Modificación de componentes:** Con base en los requerimientos del cliente y los componentes encontrados en la etapa de análisis de componentes se hacen modificaciones a los componentes para hallar versiones ajustadas a las necesidades del cliente.
- **Diseño del sistema con reutilización:** Se realiza un diseño con base en los componentes que se reutilizarán. Si ciertos componentes no están disponibles de acuerdo a lo requerido por el diseño se debe diseñar nuevo software, nuevos componentes.
- **Desarrollo e Integración:** Los componentes que no existen y fueron diseñados en la etapa anterior se deben desarrollar, y luego comenzar con el proceso de integración de los componentes tanto los existentes como los nuevos componentes.

Ventajas:

- Entregas más rápidas debido a la reducción en los tiempos de desarrollo.
- Las aplicaciones son modulares permitiendo el desarrollo independiente de componentes los cuales alimentan el almacén de componentes para ser incluidos en otros proyectos.

Desventajas:

- Si la organización no tiene control sobre cada componente del almacén de componentes pierde sentido la orientación hacia este modelo.

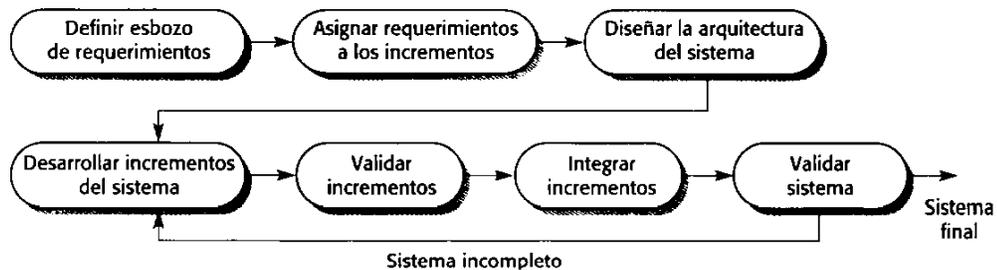
- El cliente exige tiempos basado en el hecho que ya muchas cosas están desarrolladas y no siempre los componentes existentes cumplen con las necesidades de los clientes.

1.4 PROCESO DEL SOFTWARE POR ENTREGAS INCREMENTALES⁴

Este modelo del proceso software es un proceso intermedio entre las mejores características del modelo en cascada y el modelo evolutivo. El analista comienza por realizar una definición de los requerimientos del sistema completo y luego diseña una estructura por incrementos o entregables de la aplicación. Cada incremento se desarrolla siguiendo los pasos que propone el modelo en cascada o el ciclo de vida del software. Cada incremento define o entrega un subconjunto de funcionalidad del software completo y son definidos, desarrollados y entregados con base en la prioridad del cliente. Una vez que los incrementos del sistema se han identificado, los requerimientos para los servicios que se van a entregar en el primer incremento se definen en detalle, y éste se desarrolla. Durante el desarrollo, se puede llevar a cabo un análisis adicional de requerimientos para los requerimientos posteriores, pero no se aceptan cambios en los requerimientos para el incremento actual.

⁴ Ian Sommerville, (2005). Ingeniería del Software. Addison Wesley. Séptima Edición

Figura 5. Proceso del software por entregas incrementales



Fuente: Ian Sommerville. Ingeniería del Software, Séptima Edición

Ventajas:

- Los clientes no tienen que esperar hasta que el sistema completo se entregue para sacar provecho de él. El primer incremento satisface los requerimientos más críticos de tal forma que pueden utilizar el software inmediatamente.
- Los clientes pueden utilizar los incrementos iniciales como prototipos y obtener experiencia sobre los requerimientos de los incrementos posteriores del sistema.
- Existe un bajo riesgo de un fallo total del proyecto. Aunque se pueden encontrar problemas en algunos incrementos, lo normal es que el sistema se entregue de forma satisfactoria al cliente.
- Puesto que los servicios de más alta prioridad se entregan primero, y los incrementos posteriores se integran en ellos, es inevitable que los servicios más importantes del sistema sean a los que se les hagan más pruebas. Esto significa que es menos probable que los clientes

encuentren fallos de funcionamiento del software en las partes más importantes del sistema.

Desventajas:

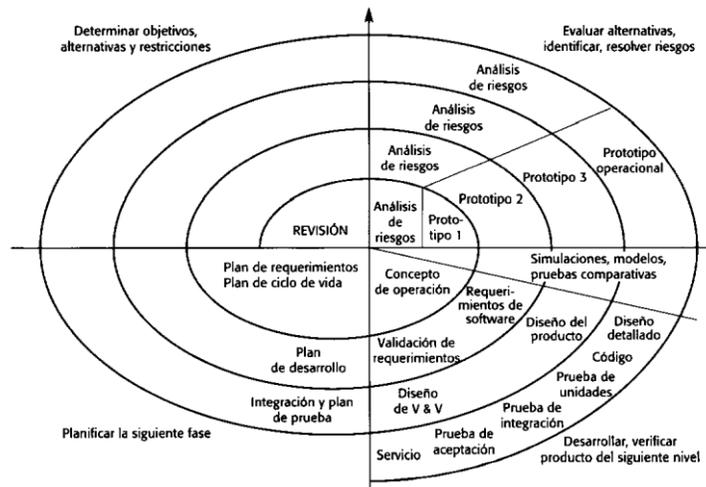
- Puede ser difícil adaptar las necesidades del cliente a incrementos de tamaño apropiado.
- Pueden existir componentes comunes a todos los incrementos y debido a que los requerimientos no se definen en detalle hasta que se implemente el incremento puede ser difícil identificar los recursos comunes a todos los incrementos.

1.5 PROCESO SOFTWARE EN ESPIRAL⁵

El modelo en espiral es una mirada diferente al proceso software. Más que representar el proceso del software como una secuencia de actividades con retrospectiva de una actividad a otra, se representa como una espiral. Cada ciclo en la espiral representa una fase del proceso del software. Así el ciclo más interno podría referirse a la viabilidad del sistema. el siguiente ciclo a la definición de requerimientos el siguiente ciclo al diseño del sistema, y así sucesivamente. Cada ciclo del espiral se divide en 4 sectores:

⁵ Ian Sommerville, (2005). Ingeniería del Software. Addison Wesley. Séptima Edición

Figura 6. Proceso del software en espiral



Fuente: Ian Sommerville. Ingeniería del Software, Séptima Edición

Definición de objetivos: Para esta fase del proyecto se definen los objetivos específicos. Se identifican las restricciones del proceso y el producto, y se traza un plan detallado de gestión. Se identifican los riesgos del proyecto. Dependiendo de estos riesgos, se planean estrategias alternativas.

Evaluación y reducción de riesgos: Se lleva a cabo un análisis detallado para cada uno de los riesgos del proyecto identificados. Se definen los pasos para reducir dichos riesgos. Por ejemplo, si existe el riesgo de tener requerimientos inapropiados se puede desarrollar un prototipo del sistema.

Desarrollo y Validación: Después de la evaluación de riesgos se elige un modelo para el desarrollo del sistema. Por ejemplo si los riesgos en la interfaz de usuario son dominantes un modelo de desarrollo apropiado podría ser la construcción de prototipos evolutivos. Si los riesgos de

seguridad son la principal consideración, un desarrollo basado en transformaciones formales podría ser el más apropiado, y así sucesivamente. El modelo en cascada puede ser el más apropiado para el desarrollo si el mayor riesgo identificado es la integración de los subsistemas.

Planificación: El proyecto se revisa y se toma la decisión de si se debe continuar con un ciclo posterior de la espiral. Si se decide continuar. Se desarrollan los planes para la siguiente fase del proyecto.

La diferencia principal entre el modelo en espiral y los otros modelos del proceso del software es la consideración explícita del riesgo en el modelo en espiral. Informalmente, el riesgo significa sencillamente algo que puede ir mal.

Desventajas:

- Se requiere una considerable habilidad para evaluar riesgos, si un riesgo importante no es considerado podría traer consecuencias graves para el buen desarrollo del proyecto.

Ventajas:

- Permite un alto control de las actividades que serán desarrolladas a la vuelta de cada ciclo del espiral.
- Administración y control de riesgos en medio del proceso de desarrollo. Es explícitamente definido para cada etapa.

2. MÉTODO, METODOLOGÍA Y MÉTRICA

2.1 MÉTODO

Un método de ingeniería del software es un enfoque estructurado para el desarrollo de software cuyo propósito es facilitar la producción de software de alta calidad de una forma costeable. Métodos como Análisis Estructurado (DeMarco, 1978) y JSD (Jackson, 1983) fueron los primeros desarrollados en los años 70. Estos métodos intentaron identificar los componentes funcionales básicos de un sistema, de tal forma que los métodos orientados a funciones aún se utilizan ampliamente. En los años 80 y 90 estos métodos orientados a funciones fueron complementados por métodos orientados a objetos como los propuestos por Booch (1994) y Rumbaugh (Rumbaugh et al., 1991). Estos diferentes enfoques se han integrado en un solo enfoque unificado basado en el Lenguaje de Modelado Unificado (UML) (Booch 1999; Rumbaugh 1999a; Rumbaugh 1999b).

2.2 METODOLOGÍA

En un proyecto de desarrollo de software la metodología define **Quién** debe hacer **Qué**, **Cuando** y **Como** hacerlo.⁶

No existe un método o metodología ideal, y métodos diferentes tienen distintas áreas donde son aplicables. Por ejemplo, los métodos orientados a objetos a menudo son apropiados para sistemas interactivos, pero no

⁶ <http://users.dsic.upv.es/~letelier/pub/>

para sistemas con requerimientos rigurosos de tiempo real.⁷

2.2.1 METODOLOGÍAS ESTRUCTURADAS⁸

Los métodos estructurados que se basan en la producción de modelos gráficos del sistema y en muchos casos, código automáticamente generado desde estos modelos.

Un método estructurado incluye un modelo del proceso de diseño, notaciones para representar el diseño, formatos de informes, reglas y pautas de diseño. Los métodos estructurados pueden ayudar a alguno o a la totalidad de los siguientes modelos de un sistema:

- Un modelo que muestra los objetos que muestra las clases de objetos utilizadas en el sistema y sus dependencias.
- Un modelo de secuencias que muestra cómo interactúan los objetos en el sistema cuando éste se ejecuta.
- Un modelo del estado de transición que muestra los estados del sistema y los disparadores de las transiciones desde un estado a otro.
- Un modelo estructural en el cual se documentan los componentes del sistema y sus agregaciones.

⁷ Ian Sommerville, (2005). Ingeniería del Software. Addison Wesley. Séptima Edición

⁸ Roger S. Pressman. Ingeniería del Software, Un enfoque práctico. Quinta edición. McGraw Hill

- Un modelo de flujo de datos en el que el sistema se modela utilizando la transformación de datos que tiene lugar cuando se procesan. Éste no se utiliza normalmente en los métodos orientados a objetos pero todavía se utiliza frecuentemente en el diseño de sistemas de tiempo real y de negocio.

2.3 MÉTRICAS DEL SOFTWARE⁹

De acuerdo al glosario del IEEE “*Standard Glossary of Software Engineering Terms*” define **métrica** como “una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado”.

La medición es fundamental para cualquier disciplina de ingeniería, y la ingeniería del software no es una excepción. La medición nos permite tener una visión más profunda proporcionando un mecanismo para la evaluación objetiva. Por medio de las mediciones se podrían implementar procesos de mejoramiento y la elección de opciones en el desarrollo de proyectos software se convierte en una labor más racional.

Las métricas del software están relacionadas con el desarrollo del software como funcionalidad, complejidad y eficiencia.

- **Métricas técnicas:** Se centran en las características de software por ejemplo: la complejidad lógica, el grado de modularidad. Mide la estructura del sistema, el cómo está hecho.
- **Métricas de Calidad:** Proporcionan una indicación de cómo se ajusta el software a los requisitos implícitos y explícitos del cliente. Es decir cómo voy a medir para que mi sistema se adapte a los requisitos que me pide el cliente.

⁹ Roger S. Pressman. Ingeniería del Software, Un enfoque práctico. Quinta edición. McGraw Hill

- **Métricas de Productividad:** Se centran en el rendimiento del proceso de la ingeniería del software. Es decir que tan productivo va a ser el software que voy a diseñar.
- **Métricas orientadas a la persona:** Proporcionan medidas e información sobre la forma que la gente desarrolla el software de computadoras y sobre todo el punto de vista humano de la efectividad de las herramientas y métodos. Son las medidas que voy a hacer de mi personal que va hará el sistema.
- **Métricas orientadas al tamaño:** Es para saber en qué tiempo voy a terminar el software y cuantas personas voy a necesitar. Son medidas directas al software y el proceso por el cual se desarrolla, si una organización de software mantiene registros sencillos, se puede crear una tabla de datos orientados al tamaño.
- **Métricas orientadas a la función:** Son medidas indirectas del software y del proceso por el cual se desarrolla. En lugar de calcularlas las LDC, las métricas orientadas a la función se centran en la funcionalidad o utilidad del programa. Las métricas orientadas a la función fueron el principio propuestas por Albercht quien sugirió un acercamiento a la medida de la productividad denominado método del punto de función. Los puntos de función que obtienen utilizando una función empírica basando en medidas cuantitativas del dominio de información del software y valoraciones subjetivos de la complejidad del software.

2.3.1 MÉTRICA DE PUNTO FUNCIÓN¹⁰

La métrica del punto función es un método utilizado en ingeniería del software para medir el tamaño del software. Fue definida por Allan Albrecht, de IBM, en 1979 y pretende medir la funcionalidad entregada al usuario independientemente de la tecnología utilizada para la construcción y explotación del software, y también ser útil en cualquiera de las fases de vida del software, desde el diseño inicial hasta la explotación y mantenimiento.

Los puntos de función se derivan con una relación empírica según las medidas contables (directas) del dominio de información del software y las evaluaciones de la complejidad del software.

Se determinan cinco características de dominios de información y los valores de los dominios de información se definen de la forma siguiente:

- **Número de entradas de usuario:** Se cuenta cada entrada de usuario que proporciona diferentes datos orientados a la aplicación. Las entradas se deberían diferenciar de las peticiones, las cuales se cuentan de forma separada. Los puntos de función se derivan de medidas directas del dominio de la información.
- **Número de salidas de usuario:** Se cuenta cada salida que proporciona al usuario información orientada a la aplicación. En este contexto la salida se refiere a informes, pantallas, mensajes de error, etc. Los elementos de datos particulares dentro de un informe no se cuentan de forma separada.

¹⁰ Roger S. Pressman. Ingeniería del Software, Un enfoque práctico. Quinta edición. McGraw Hill

- **Número de peticiones de usuario:** Una petición se define como una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva. Se cuenta cada petición por separado.
- **Número de archivos:** Se cuenta cada archivo maestro lógico (esto es, un grupo lógico de datos que puede ser una parte de una gran base de datos o un archivo independiente).
- **Número de interfaces externas:** Se cuentan todas las interfaces legibles por la máquina (por ejemplo: archivos de datos de cinta o disco) que se utilizan para transmitir información a otro sistema.

Una vez que se han recopilado los datos anteriores, a la cuenta se asocia un valor de complejidad. Las organizaciones que utilizan métodos de puntos de función desarrollan criterios para determinar si una entrada en particular es simple, media o compleja. No obstante la determinación de la complejidad es algo subjetiva.

3. ANÁLISIS SITUACIÓN ACTUAL ÁREA DE DESARROLLO DE APLICACIONES EMPRESA DE CONTACT-CENTER EMTELCO S.A.

3.1 VISIÓN GENERAL

Emtelco S.A es una sociedad mixta del orden municipal que hace parte del grupo EPM Telecomunicaciones S.A que tiene por objeto social principal el desarrollo de actividades como: Prestar servicios de Contact-Center, prestar servicios de asesorías, capacitación, consultorías e interventorías profesionales asociadas a las actividades propias de la sociedad, prestar servicios de soporte de Outsourcing en tecnología, informática, eléctricos, electrónicos, help desk o mesas de ayuda, gestión, administración, procesamiento de información, digitación e ingeniería de procesos y afines a EPM Telecomunicaciones S.A. E.S.P., en tanto los mismos sean requeridos por esta última dentro de las soluciones ofrecidas a sus clientes¹¹.

Dentro de los procesos que conforman su sistema de gestión integral se encuentra nuestro macro proceso de enfoque: Gestionar Información y Tecnología, el cual a su vez está conformado por los procedimientos de: Gestionar la infraestructura tecnológica, Gestionar incidentes o requerimientos tecnológicos, gestionar aplicaciones e información y gestionar la seguridad de la información e infraestructura. Nuestro target es el procedimiento Gestionar aplicaciones e información.

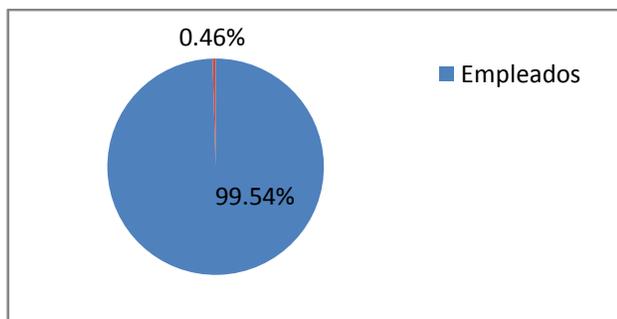
El proceso de Gestión de información y tecnología es un proceso de apoyo. Esto sugiere que Emtelco S.A no es una empresa dedicada al

¹¹ http://www.emtelco.com.co/index.php?option=com_content&view=article&id=2&Itemid=7

desarrollo de software, y todo a lo que en materia de proyectos informáticos se refiere, está sometido a las necesidades que la operación y la ejecución de las actividades a las cuales está dedicada, así lo demanden.

Con base en las cifras actuales, podemos observar que la compañía está conformada por aproximadamente 2600 personas de las cuales 12 hacen parte de la oficina de desarrollo de aplicaciones. Si hiciéramos una relación de personas del área de desarrollo de aplicaciones Emtelco S.A versus el número de personas que trabajan en la compañía en las diferentes áreas de apoyo y desarrollo de la actividades centrales del negocio encontramos un cociente de 0.46%, lo que quiere decir que por cada 217 personas desarrollando otras actividades relacionadas con el negocio hay 1 en el área de desarrollo de aplicaciones.

Figura 7. Relación empleados EMTELCO S.A vs personal área de desarrollo de aplicaciones



Fuente: Elaboración Propia

Emtelco S.A es una compañía que desarrolla y adquiere software. En cuanto al software que adquiere se encuentran:

- Aplicaciones ofimáticas
- Software de telecomunicaciones

- Software para desarrollo (compiladores, soluciones tecnológicas etc.)
- Software de base de datos
- Software para soporte de informes
- Sistemas operativos
- Software para soluciones de telefonía

En cuanto a desarrollo de software principalmente:

- Desarrolla aplicaciones de propósito general: Informes, aplicaciones para soporte de las actividades cotidianas del negocio y sus clientes.
- Nuevas soluciones para las plataformas de soporte de telefonía que posee.

Toda su infraestructura tecnológica esta soportada en herramientas comerciales licenciadas, principalmente Microsoft en lo que a aplicaciones ofimáticas, sistemas operativos e infraestructura de telecomunicaciones se refiere y en cuanto a herramientas de soporte a la operación de telefonía se encuentra apoyada en soluciones para Contact-Center desarrolladas por Interactive Intelligence y AVAYA.

Su infraestructura tecnológica permite el desarrollo de aplicaciones distribuibles a través de web (Intranets, servicios web, aplicaciones web, páginas de contenido HTML, etc.), a través de estaciones cliente (aplicaciones de propósito general y específico Stand Alone) y adicionalmente pueden ser desarrolladas aplicaciones para soluciones de telefonía que se desarrollan en las plataformas destinadas para tal propósito.

3.2 DINÁMICA DE LAS APLICACIONES EMTELCO S.A

Las aplicaciones desarrolladas por EMTELCO S.A tienen dos puntos de enfoque principalmente:

- **Necesidades propias del negocio de Contact-Center:** sistemas que apoyan las tareas administrativas y operativas del negocio. Sistemas de administración del recurso humano, sistemas de administración de requerimientos entre otros.
- **Necesidades específicas del cliente:** Sistemas desarrollados para los clientes de EMTELCO S.A, aplicaciones a la medida. En estos sistemas se incluyen las soluciones de telefonía que hacen parte de la infraestructura ofrecida por EMTELCO S.A a sus clientes.

La vida útil de las aplicaciones EMTELCO S.A está ligada específicamente a la dinámica de los puntos de enfoque. Así los sistemas ligados a los procesos propios del negocio de Contact-Center tienen la tendencia a ser sistemas estables y de larga durabilidad con índices menores de mantenibilidad.

Para el caso de los sistemas desarrollados para atender las necesidades específicas de los clientes EMTELCO S.A, las aplicaciones cambian más en el tiempo. En un alto porcentaje no son sistemas definitivos sino que están hechos para asistir alguna campaña o proceso temporal de la compañía cliente y por otro lado no necesariamente están hechos para una larga vida de útil debido a que la permanencia de los clientes puede ser un factor volátil.

3.3 ANÁLISIS OFICINA DE DESARROLLO DE APLICACIONES

3.3.1 GENERALIDADES

La oficina de desarrollo de aplicaciones Emtelco S.A está conformada por doce personas, de la cuales tres están encargadas del desarrollo de aplicaciones de propósito general, otras tres personas están dedicadas a dar soporte a la plataforma de telefonía Interactive Intelligence quienes a su vez apoyan a dos personas más en el desarrollo de aplicaciones telefónicas basadas en la plataforma telefónica AVAYA. Por otro lado una persona es encargada del montaje y soporte de base de datos y finalmente tres personas más son encargadas de desarrollar herramientas de consolidación de datos e información entre esos informes que están relacionados con todas las plataformas telefónicas.

La oficina de desarrollo de aplicaciones Emtelco S.A no sigue ningún modelo de proceso de desarrollo de software, cada uno de los analistas e ingenieros que la componen, siguen un proceso de desarrollo de acuerdo a sus propios métodos. Sin embargo lo que dicha oficina sí estableció es la documentación mínima con la que debe contar cada desarrollo hecho al interior de la compañía:

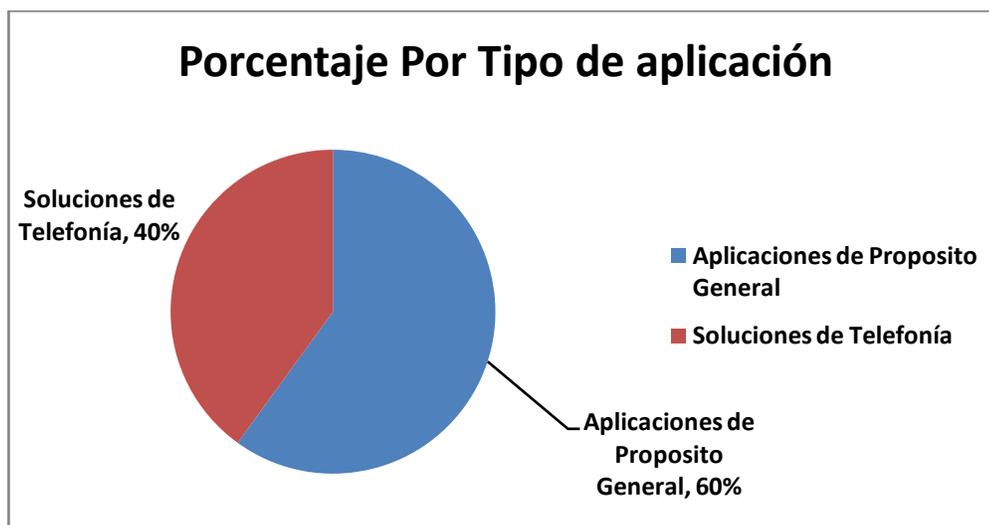
Manuales de usuario, documentos de análisis y diseño de aplicaciones tales como:

- Diagrama de clases.
- Documento de análisis de casos de uso.
- Documento de descripción detallada de requisitos.
- Diagrama de flujo.
- Manual de usuario.

También está establecido que se debe generar documentación de control interno como: Controles de cambios, documento de análisis de riesgo de montaje, entre otros. Dichos documentos deben ser versionados y almacenados en un banco central de información de cada una de las aplicaciones.

El 60% de los proyectos desarrollados en el área son de aplicaciones de propósito general; Dichas aplicaciones están enfocadas a personal de toda la compañía bien sea para desarrollar un proceso específico inherente a la dinámica de un cliente o también puede estar enfocado a procesos internos. El 40% restante de los proyectos a desarrollar son soluciones de telefonía: Desarrollo de IVRs (Sigla en inglés de Interactive Voice Response, Respuestas de Voz Interactiva en español), captura y procesamiento de datos arrojados por las plataformas entre otras.

Figura 8. Distribución de proyectos por tipo de aplicaciones



Fuente: Elaboración Propia

A pesar del no establecimiento de un modelo de proceso de desarrollo de software la desviación en tiempos de los proyectos desarrollados en el área están en el orden del 0-20% lo cual es atribuido principalmente a:

- Falta de información o conocimiento por parte de los usuarios que definen la necesidad.
- Especificaciones y requisitos incompletos.
- Especificaciones y requisitos cambiantes.
- Mal dimensionamiento de tiempos y recursos.
- No seguimiento de un modelo de proceso de desarrollo estándar.

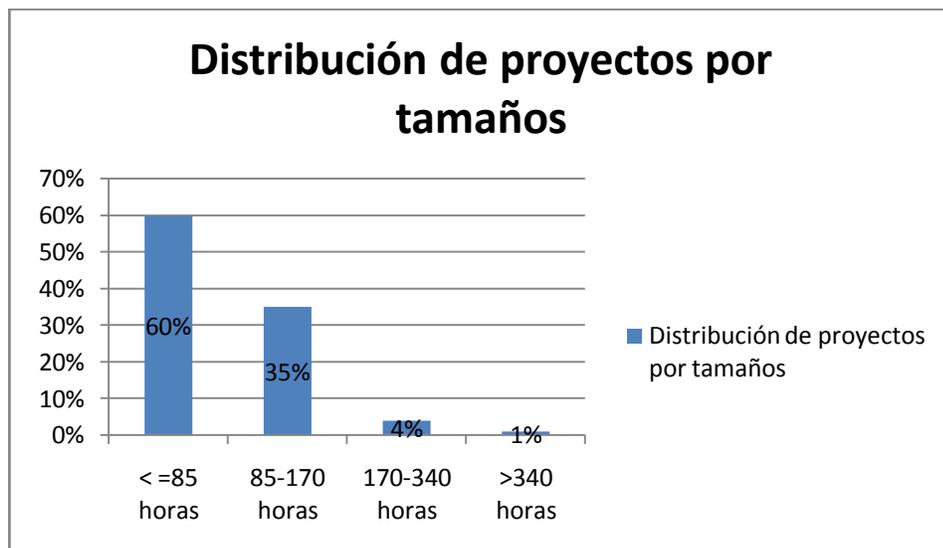
3.3.2 DIMENSIONAMIENTO Y TAMAÑO DE LOS PROYECTOS

El tamaño de los proyectos es determinado por cada uno de los integrantes del área sin seguir un procedimiento estándar o algún método conocido. Dicho dimensionamiento es el derrotero del tiempo de desarrollo de la aplicación y cada persona es responsable de ejecutar y dimensionar su proyecto de acuerdo a su discreción, sin embargo la jefatura cumple un papel activo en este último ya que cada proyecto es aprobado o revaluado cuando este no está en los tiempos observados en la experiencia de proyectos antes hechos. Teniendo en cuenta el razonamiento anterior se desarrollan proyectos en las siguientes escalas de tiempo:

- 60% de los proyectos que se desarrollan en la compañía exigen un máximo de 85 horas desarrollo, son aplicaciones de menos de 10 días laborales.

- 35% de los proyectos se encuentran en un rango de 85 a 170 horas desarrollo entre 10 y 20 días laborales.
- 4% son proyectos de 170 a 340 horas desarrollo entre 20 y 40 días laborales.
- 1% de los proyectos son más de 340 horas desarrollo, más de 40 días laborales.

Figura 9. Distribución de proyectos por tamaño

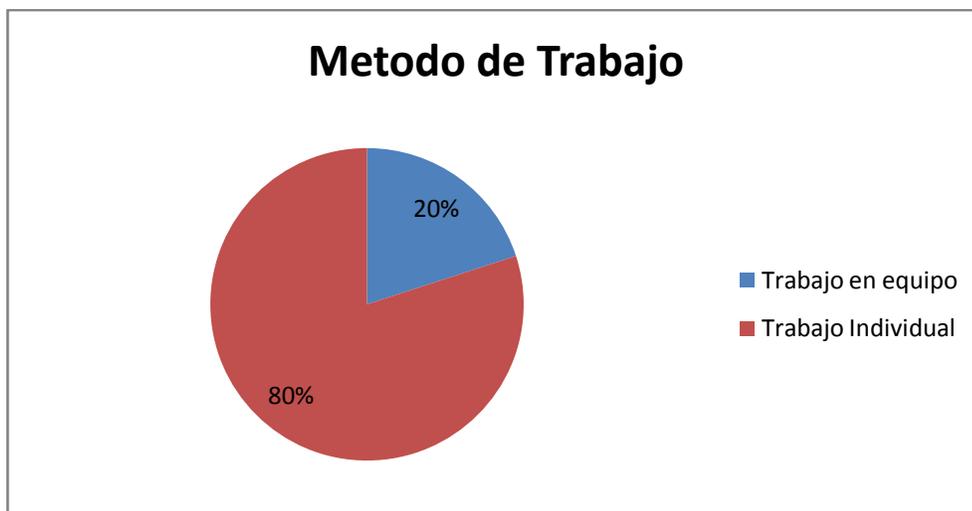


Fuente: Elaboración Propia

3.3.3 ENFOQUE DE EMPODERAMIENTO DE LOS PROYECTOS Y PERFIL DEL EMPLEADO DEL ÁREA DE DESARROLLO DE APLICACIONES EMTELCO S.A

Cada integrante del área esta empoderado de un tema especifico, cada tema está respaldado por dos o una persona. Sin embargo existen proyectos que requieren de la ayuda de varios integrantes del área de desarrollo. El 20% de los proyectos es asignado y desarrollado por equipos de trabajo y el otro 80% por una sola persona.

Figura 10. Distribución de proyectos teniendo en cuenta el método de trabajo con el que se desarrolla



Fuente: Elaboración Propia

Este método de trabajo obedece básicamente al tipo de proyectos que le son demandados al área; una variable que puede ser tenida en cuenta que ya fue analizada y es crítica para este método de trabajo es el tamaño de las aplicaciones sabemos que el 60% de estos no excede las 85 horas.

Adicionalmente otro criterio a tener en cuenta en el análisis de la situación actual del área de desarrollo de aplicaciones Emtelco es el hecho que no existe una especialización de tareas del proceso de desarrollo de software; no existe un diseñador, un analista de requisitos, un desarrollador, un arquitecto, como podría plantearse en muchos enfoques de desarrollo. Los analistas juegan todos los papeles en el proceso de desarrollo, el mismo analista elícita requisitos, se encarga de diseñar las soluciones y a su vez de desarrollarlas e implementarlas.

3.3.4 A QUIENES ESTÁN DIRIGIDOS LOS PROYECTOS INFORMÁTICOS EMTELCO S.A

Los proyectos que son desarrollados en el área como se estableció ya en la parte de atrás, están dirigidos a solucionar las problemáticas generadas por el desarrollo de las actividades principales por parte de Emtelco S.A. Así las aplicaciones de propósito general podrían ser usadas por personas que pertenecen a toda la compañía o a áreas específicas y apoyan su labor en dichas herramientas. Otro caso también se puede dar cuando dichas aplicaciones son usadas por otras como es el caso de algunos servicios web que cumplen funciones complementarias a algunas aplicaciones o simplemente cumplen con propósitos de modularización de tareas.

Por el otro lado están las soluciones de telefonía, generalmente el desarrollo de IVR's está dirigido a personas del común que no tienen nada que ver con la empresa; En muchos casos en las líneas de servicio al cliente de algunas compañías se presta un servicio de asesoría en servicios nuevos esto hace que el cliente no sepa con que exactamente se va a encontrar al realizar su llamada, sin embargo los IVR's son

desarrollados con preguntas comunes que el cliente responderá sin necesidad de saber mucho acerca de un tema.

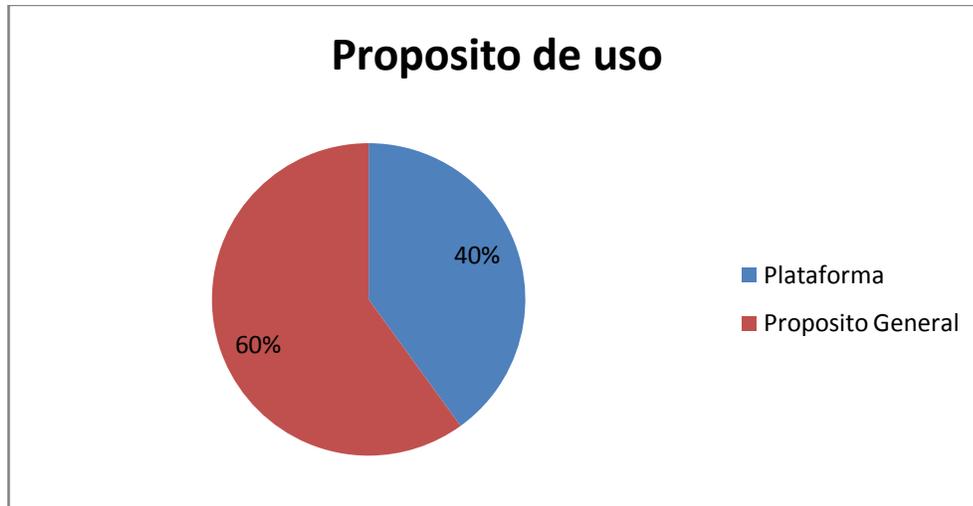
El área de desarrollo está encargada de implementar dichos IVR's las preguntas y el concepto de navegación a través de estos, les es entregado por completo por parte de las personas encargadas del diseño de IVR's. Algunas soluciones de telefonías están más relacionadas con la captura de datos generados en el contacto telefónico, datos como: fecha y hora de la llamada identificación de la misma, son procesados a través de herramientas de soluciones de telefonía para ser utilizadas por aplicaciones de propósito general desarrolladas por Emtelco S.A o como en muchos casos software licenciado del cual no es propietario.

3.4 ANÁLISIS DE ÁREA DESDE EL PROCESO DE CADA ANALISTA

Cada analista de acuerdo a su ocupación y su perfil al interior del área de desarrollo fue encuestado en su labor con la finalidad de identificar puntos comunes y tener un acercamiento más detallado del tipo de aplicaciones con las que se enfrentan en el día a día y de acuerdo a su criterio cuales podrían ser los elementos que den luces en la definición de un proceso estándar para el área.

De acuerdo a los porcentajes mostrados en nuestro análisis del área desde la óptica de la jefatura (todo el análisis atrás desarrollado es hecho con base en las respuestas a la encuesta realizada al jefe de la oficina de desarrollo Emtelco S.A.) el 40% de los proyectos desarrollados por el área son soluciones de telefonía y el 60% restante son aplicaciones de propósito general.

Figura 11. Distribución de proyectos teniendo en cuenta el propósito de uso



Fuente: Elaboración Propia

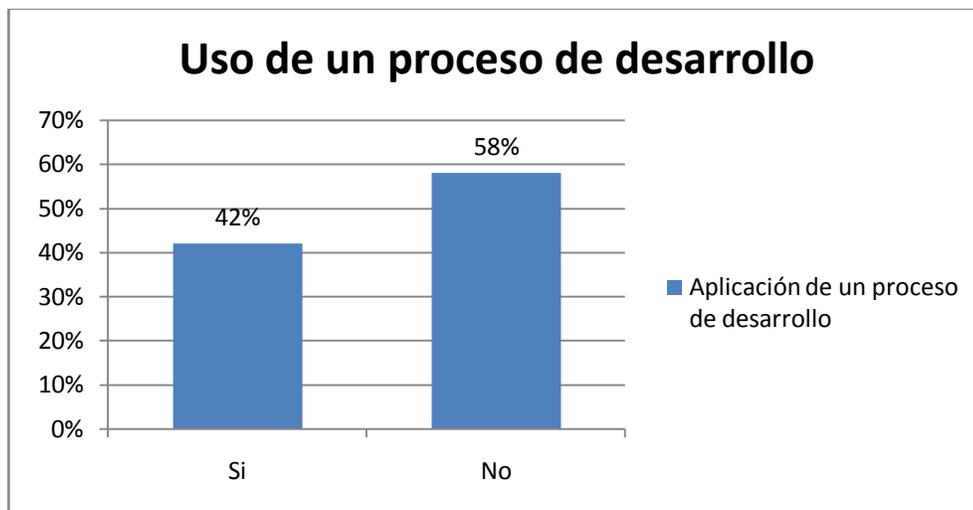
Mirando en detalle las labores de los analistas en su día a día nos damos cuenta el 100% de ellos desarrolla en sus proyectos algo relacionado con desarrollo de base de datos; todas las aplicaciones desarrolladas exigen que por lo menos se desarrolle una base de datos o se implementen o usen módulos para implementaciones en estas. El 60% del personal está dedicado a desarrollar soluciones telefónicas y el 100% de ellos desarrolla aplicaciones de propósito general unos en más o menos medida que otros.

3.4.1 ACERCA DEL USO DE UN MODELO DE PROCESO DE DESARROLLO

El 58% de los analistas no utiliza ningún modelo de proceso de desarrollo de software. Las actividades que desarrollan para los diferentes proyectos no están establecidas o por lo menos no de una manera consciente (de acuerdo a la experiencia es posible que siempre usen un modelo en

cascada).La realización de sus tareas no obedece a un proceso sistemático establecido.

Figura 12. Uso de un proceso de desarrollo de software por parte de los analistas del área de desarrollo



Fuente: Elaboración Propia

Del 42% que usan un modelo de desarrollo de software un 25% usa el modelo de proceso de desarrollo en cascada, el 16% usa el modelo por componentes y otro 16% usa el modelo por prototipos.

Sólo el 16% del personal desarrollan basándose en más de un modelo de proceso de desarrollo los demás sólo usan uno.

Tomando en cuenta algunos factores fundamentales en la implementación de un modelo de proceso de desarrollo de software se

diseñaron algunos ítems para evaluar la idoneidad de un modelo X dada la experiencia de cada uno de los integrantes del área de desarrollo Emtelco S.A.

A la pregunta: *Asumiendo que su compañía adopta un modelo de proceso de desarrollo de software, enumere cada una de las siguientes opciones de 1 a 6, siendo 1 la opción de mayor importancia y 6 la menor. (Tenga en cuenta su experiencia y el tipo de aplicaciones que le son asignadas, no considere el orden ideal).*

a. *El modelo de procesos de desarrollo de software me debe permitir realizar cambios en la marcha de los requerimientos iniciales de la aplicación.*

Los requisitos cambiantes como elemento relevante para el buen desarrollo de la labor software. Este ítem fue el de mayor prioridad y tomo una importancia del 24% en la eventual implementación de un modelo de proceso de desarrollo de software para Emtelco S.A

b. *El modelo de proceso de desarrollo de software debe estar orientado a la reutilización de código y clases o componentes que ya han sido desarrollados en otros proyectos.*

El uso eficiente de los recursos para incidir en el factor tiempo. Este ítem fue el segundo en prioridad con un porcentaje de importancia del 22:%

c. *El modelo de proceso de desarrollo de software debe ser flexible de tal manera que puedan realizarse u omitirse algunas de las actividades según sea el caso.*

La capacidad y flexibilidad del modelo de proceso de software de convertirse en una solución y no un tropiezo más del proceso de desarrollo e implementación de proyectos informáticos. Este ítem fue el tercero en prioridad con un porcentaje de importancia del 17%.

- d. *El modelo de proceso de desarrollo de software debe ser rápido y exigir poca documentación.*

Eficiencia en el tiempo y formalidad o informalidad del proceso de desarrollo de software. La prioridad de la calidad versus la inmediatez de las necesidades de una organización. Este ítem fue el cuarto en prioridad con un porcentaje de importancia del 13%.

- e. *El modelo de proceso de desarrollo de software debe permitirme hacer entregas parciales (hitos) de la aplicación.*

El modelo de proceso como herramienta para la especificación de requisitos de sistemas complejos y la completitud.

Este ítem fue el quinto en prioridad con un porcentaje de importancia del 12.5%.

- f. *El modelo de proceso de desarrollo de software debe ser la mezcla de diferentes modelos de procesos que se adaptan según el tipo de software.*

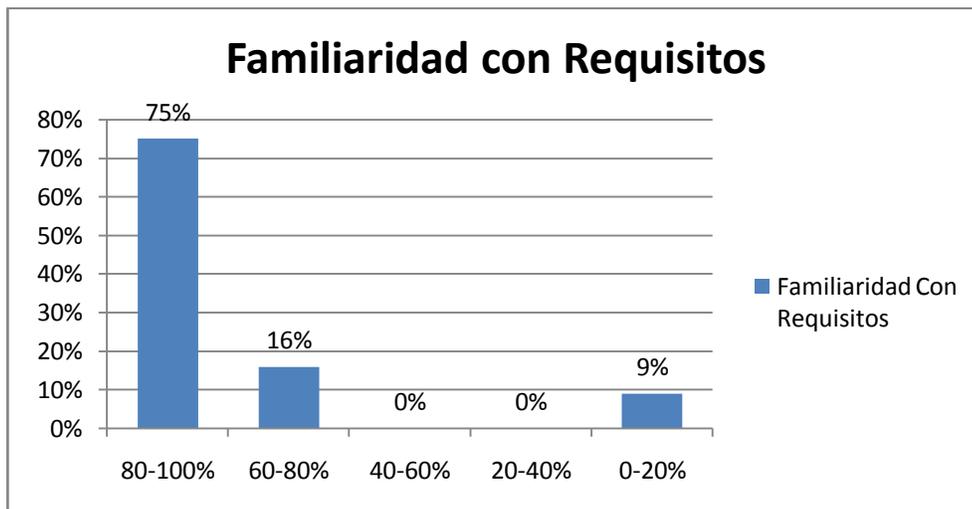
La naturaleza del proyecto como argumento de selección del modelo de proceso adecuado. El modelo de proceso de software no es necesariamente único, este puede obtener lo mejor de cada modelo y ser una mezcla de varias propuestas.

Este ítem fue el sexto en prioridad con un porcentaje de importancia del 11.5%.

3.4.2 Acerca de los requisitos de los proyectos

El 75% del personal del área dice conocer o estar familiarizado en un 80-100% con los requisitos de las asignaciones a desarrollar, un 16% reconoce estar familiarizados con los requisitos de sus asignaciones en 60-80% y el 9% restante en un 0-20%. Esto refleja un alto conocimiento de los requisitos del cliente y de las labores que se deben desarrollar.

Figura 13. Familiaridad de la oficina de desarrollo con los requisitos EMTELCO S.A



Fuente: Elaboración Propia

Por otro lado es importante analizar cuál es el nivel de estabilidad de los requisitos. Es así como el 16% del personal dice que los requisitos de sus asignaciones cambian en etapas posteriores a la etapa de análisis de requisitos en un 0-20%, el 50% del personal dicen que varían en un 20-40% y el 34% restante dicen que varían en un 40-60%. En ningún caso se presentan variaciones que estén por encima del 60%.

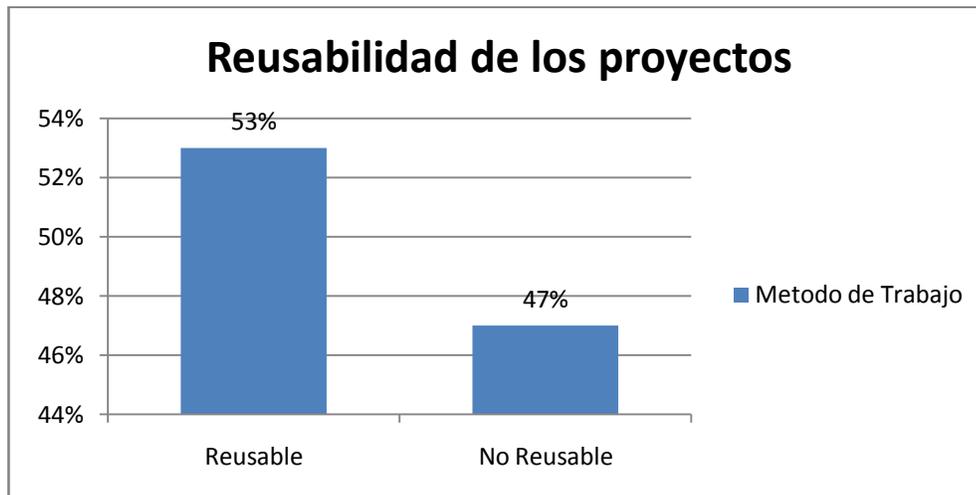
3.4.3 Acerca de la reutilización de código

La reutilización de código se podría convertir en una herramienta poderosísima para el desempeño eficiente de un área de desarrollo. A la pregunta:

Cuando desarrolla aplicaciones del mismo tipo, si pensara en la posibilidad de implementar una aplicación utilizando componentes de otras aplicaciones ya desarrolladas, ¿qué porcentaje de la nueva aplicación podría desarrollar utilizando los componentes ya hechos en proyectos anteriores?

El promedio total de desarrollo que podría alcanzarse por medio de este método sería de un 53%. Siendo 20% el valor más bajo en la información tomada y 80% el más alto.

Figura 14. Reusabilidad de los proyectos EMTELCO S.A



Fuente: Elaboración Propia

3.4.4 Acerca de la utilización de un modelo de proceso de desarrollo que permita entregas parciales.

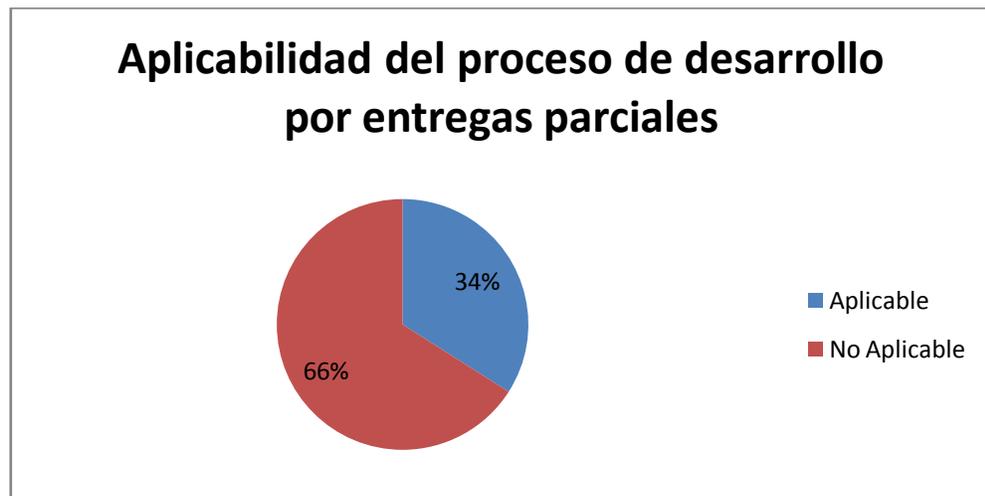
De acuerdo a la pregunta:
De acuerdo a la naturaleza de los proyectos que le son asignados ¿considera usted que deberían ser desarrollados realizando entregas parciales?

El 34% del personal considera que si debería implementarse un modelo de proceso de desarrollo que contemple las entregas parciales aduciendo las siguientes razones:

- Porque las entregas parciales sirven para ir evaluando la satisfacción del cliente y también para medir el cubrimiento o no de los requerimientos y/o especificaciones.

- Debido al tamaño de los proyectos. Se hace realmente difícil pensar en que un proyecto de más de un mes de desarrollo (hay casos en que de más de 15 días) no tenga entregas que permitan saber que también se están haciendo las cosas.

Figura 15. Aplicabilidad del proceso de desarrollo por entregas parciales en EMTELCO S.A



Fuente: Elaboración Propia

El 66% restante consideran que no debería implementarse un modelo con esta característica entre otras razones por:

- La dinámica del negocio y de los requerimientos que solicitan, pocas veces permite que un desarrollo pueda ser entregados parcialmente.
- Se podría perder el objetivo del requerimiento inicial.
- El desarrollo de aplicaciones de telefonía en su mayoría no permiten tener un parcial entregable; siempre se concibe la solución total.

- Los proyectos asignados son pequeños, con unas pocas funcionalidades, por lo tanto por la naturaleza de los mismos requieren una respuesta muy rápida.
- Nuestros proyectos están relacionados de manera directa con la fecha de inicio de la producción de un cliente y la mayor parte del tiempo es validada su disponibilidad y funcionalidad antes de la fecha de inicio. Un alto nivel de urgencia en los requerimientos.

3.4.5 Acerca de la utilización de un modelo de proceso por prototipos.

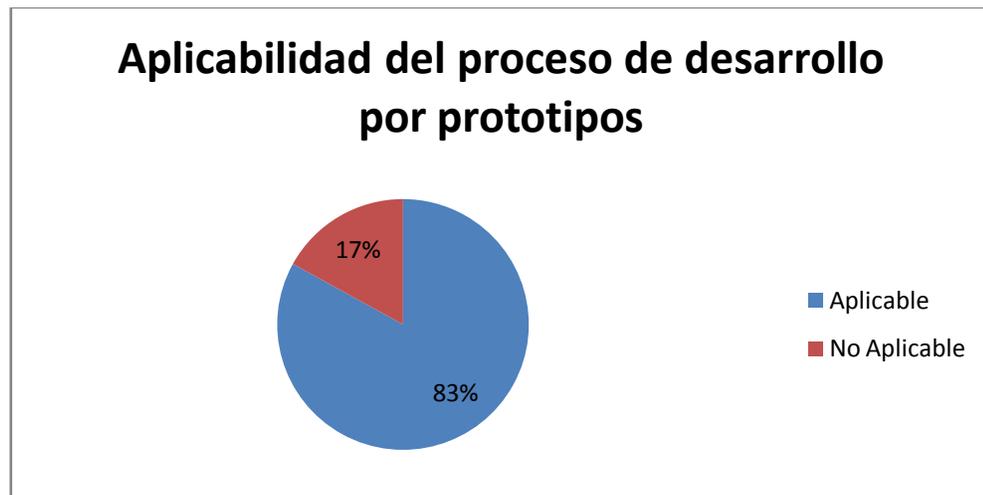
De acuerdo a la pregunta:
Teniendo en cuenta su experiencia y el tipo de aplicaciones que le son asignadas ¿considera usted que sería útil la implementación de metodologías de desarrollo de prototipos funcionales que son mejorados con base en el nivel de conocimiento de los requerimientos del cliente?

El 83% del personal encuestado respondieron que si, aduciendo las siguientes razones:

- Porque esto permite la reutilización de metodologías y elementos funcionales, lo cual ahorra tiempos de desarrollo y fija un margen estandarizado de trabajo.
- Cuando los requisitos de los usuarios son muy complejos me parece que podrían ser muy útiles.
- Es una buena opción para el desarrollo de aplicaciones complejas y de esta forma asegurar su calidad y el cumplimiento total de los requerimientos. Pero dada la naturaleza del negocio esta metodología no aplicaría para la totalidad de los proyectos, ya que en su mayoría son pequeños y deben realizarse en corto tiempo.

- Por agilidad en el proceso de desarrollo.
- Proporciona la forma de que los requerimientos sean bien desarrollados y cumplan con las expectativas y funciones que pidió el cliente.
- Sería importante implementar una metodología que logre tener las características necesarias para el desarrollo de aplicaciones de telefonía. Esto permitiría tener un trabajo más organizado y medible.

Figura 16. Aplicabilidad del proceso de desarrollo por prototipos en EMTELCO S.A



Fuente: Elaboración Propia

El 17% restante respondió que no se debería implementar un modelo de proceso de desarrollo por prototipos por las siguientes razones:

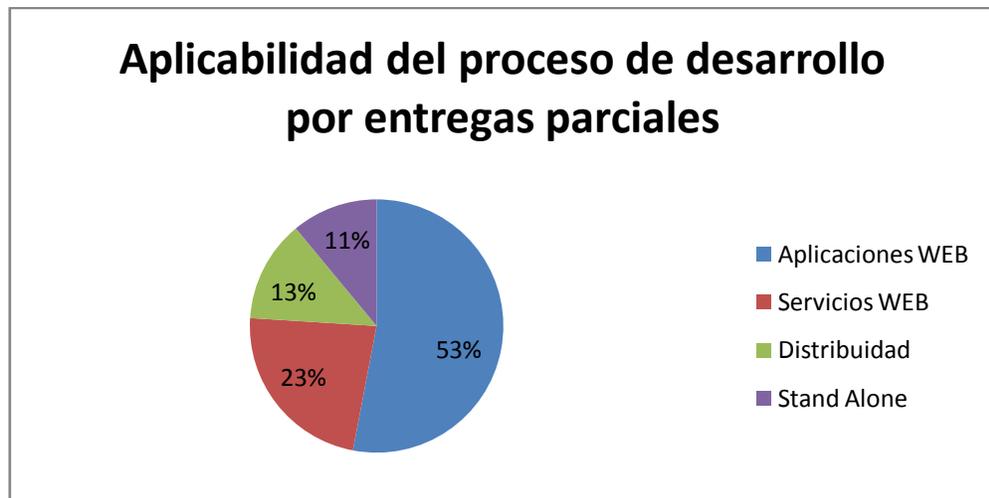
- Todas las metodologías de desarrollo son validas, aunque dependiendo de la dimensión de los desarrollos algunas pueden ser más adecuadas que otras, desde ese punto de vista para desarrollos

cortos no me parece que el desarrollo de prototipos sea el más adecuado para este caso.

3.4.6 Acerca del modelo de distribución de aplicaciones

La capacidad tecnológica de Emtelco S.A permite la distribución de sus proyectos a través de web, de clientes o Stand Alone, servicios web, o aplicaciones distribuidas. En la práctica los proyectos desarrollados son distribuidos de la siguiente manera: Un 76% de las aplicaciones son distribuidas a través de web; 53% son aplicaciones web y el 23% Servicios Web, un 11% son aplicaciones de escritorio o desarrolladas Stand Alone y el 13% restante son aplicaciones distribuidas.

Figura 17. Modelos de distribución de las aplicaciones EMTELCO S.A



Fuente: Elaboración Propia

4. PROPUESTA GUÍA PARA EL PROCESO DE DESARROLLO DE SOFTWARE EMTELCO S.A

4.1 PRELIMINARES

EMTELCO S.A obedece a diferentes variables que le hacen una necesidad en términos de desarrollo de software particular y especial.

Si bien criterios como proceso de software actual, tipo de software a desarrollar, tipo de proyectos tecnológicos son de un peso importante para la elección de una metodología o un proceso de desarrollo estándar también es verdad que hay otros factores que terminan siendo altamente relevantes en la elección de una alternativa de estandarización, factores impositivos de orden interno y externo tales como: Herramientas de desarrollo y plataformas tecnológicas predefinidas (La herramienta de desarrollo no se elige después del análisis del tipo de proyecto sino que está predefinida desde el principio del proyecto así como la plataforma tecnológica sobre la cual se implementará), limitantes de tipo presupuestal y las condiciones del mercado entre otras son decisivos en el proceso de desarrollo de software.

Como consecuencia de lo anterior el proceso de desarrollo de software en principio debe cumplir con las siguientes restricciones mínimas para la optimización y la racionalidad de los elementos presentes en el estándar:

- **Optimización de la documentación:** Una de las premisas del proceso de desarrollo EMTELCO S.A es que no se deberán seguir todos los documentos propuestos por el mercado y sobre todo por lenguajes de modelado como UML que tienen un número de diagramas que puede ser insostenible por la compañía debido a los

altos costos en tiempo. Esto básicamente haría que se filtren los principales diagramas para que hagan parte de los desarrollos EMTELCO S.A.

- **Orientación por objetos:** EMTELCO S.A. desarrolla su software a través de software licenciado Microsoft y tiene como base principal .NET en todas sus versiones. Esta herramienta es orientada completamente a objetos. Entre otras cosas las condiciones externas nos hacen ir en esa dirección: La gran mayoría de las compañías de desarrollo de software trabajan bajo este paradigma.
- **Tipo de negocio:** EMTELCO S.A es una compañía cuya actividad principal no es la producción de software esto la hace una compañía que no está dispuesta a invertir (entre otras cosas no tiene sentido) para que su proceso software alcance un CMM nivel 3 o se haga el proceso de desarrollo con base en una metodología estructurada y rigurosa como RUP por esta razón el estándar debe proveer una guía que le apunte a la simpleza pero también a la efectividad en sus resultados.
- **Flexibilidad:** Debido al tipo de negocio que es EMTELCO S.A la flexibilidad es importante en el proceso software. En ocasiones es claro que las compañías requieren que algo se haga en un tiempo determinado inclusive si esto incluye que se sacrifique la calidad. Esto es un riesgo pero como la mayoría de los riesgos si se conocen se pueden controlar, el estándar asume este riesgo pero pone de antemano que este ocurre si los pasos no son seguidos y la omisión de algunos pasos puede tornarse en más riesgoso que la omisión de

otros.

- **Limitación presupuestarias económicas:** La dinámica de la oficina de desarrollo EMTELCO S.A obedece a las limitaciones que EMTELCO como compañía impone a través de sus posiciones estratégicas, es así como variables como el número de personas del área, especialización de tareas, procesos de calidad y otros podrían convertirse en inamovibles.

4.2 GUÍA DE DESARROLLO

4.2.1 *CONDICIONES INICIALES*

- **Paradigma de enfoque:** El enfoque de solución para cada una de las aplicaciones que sean demandadas es la visión orientada a objetos; Los procesos de análisis, diseño, desarrollo e implementación se realizaran teniendo como norte de análisis el paradigma de orientación por objetos. El paradigma de orientación por objetos será la columna vertebral, cualquier mejora o retroalimentación realizada en el proceso deberá mantener dicho enfoque como hilo conductor.
- **Reutilización:** Cualquiera de los elementos presentes en el proceso de desarrollo que sea susceptible de ser reutilizado deberá ser implementado como pilar fundamental del estándar de desarrollo de software.

- **Factibilidad Técnica:** El proceso de desarrollo comienza luego de un proceso de evaluación de factibilidad tecnológica por parte de la jefatura del área de desarrollo de aplicaciones.

4.3 PROPUESTA GUÍA POR ETAPAS DE DESARROLLO

4.3.1 PRE ANÁLISIS: ELICITACIÓN DE REQUISITOS:

Análisis: En la actualidad no existe un proceso estándar para el proceso de elicitación de requisitos de las aplicaciones EMTELCO S.A. En el mercado no existen muchas propuestas metodológicas que sean completas en este sentido; son más esfuerzo o pautas sueltas para apoyar el proceso basadas en mejores prácticas sin embargo existe una metodología de elicitación de requisitos propuesta por Amador Durán Toro y Beatriz Bernárdez Jiménez de la Universidad de Sevilla en España que es una idea global para el proceso de elicitar requisitos de software. El resultado es un documento de especificación de requisitos global con elementos ligados que permiten un alto nivel de rastreabilidad.

Independiente del proceso de desarrollo de software que se lleve a cabo esta será la metodología de elicitación de requisitos. Adicionalmente dicha metodología está apoyada en la herramienta CASE REM.

Guía:

El proceso o metodología de elicitación de requisitos será obtenido siguiendo la metodología: Metodología para la Elicitación de Requisitos de Sistemas Software versión 2.3. Amador Durán Toro, Beatriz Bernárdez Jiménez Universidad de Sevilla, España.

Productos:

Documento de especificación de requisitos.

4.3.2 ANÁLISIS

Análisis: En la actualidad EMTELCO S.A realiza el dimensionamiento de sus proyectos de software realizando un proceso personal de dimensionamiento. Dicho proceso demuestra que es efectivo debido a que los niveles de desviación de los proyectos desarrollados por EMTELCO S.A se encuentran en el orden del 0 al 20% lo que no nos garantiza dicha cifra es que el proceso de dimensionamiento de proyectos hecho por cada integrante es objetivo o lo que es mejor eficiente, así es necesario que cada persona desarrollo el dimensionamiento que actualmente hace y apoyará dicho proceso en la aplicación de la métrica de tamaño de software por puntos de función. La aplicación de dicha métrica busca lograr los siguientes objetivos en el proceso de desarrollo de EMTELCO S.A:

- Crear un indicador de eficiencia de cada integrante del área en aras de que más adelante cuando el proceso demuestre cierto grado de madurez se pueda aplicar como indicador de calidad del área.
- Hacer del dimensionamiento de proyectos de desarrollo de EMTELCO S.A un proceso argumentado y serio que de credibilidad y confianza a los clientes.

- Crear un indicador de capacidad de desarrollo que le permita a la jefatura del área de desarrollo EMTELCO S.A tomar decisiones a corto y largo plazo.
- Proveer una herramienta que le permita a la jefatura del área fijar tiempos estándar de desarrollo y de esta manera tener criterios de evaluación de tiempos propuestas por los analistas.

Nota: La tendencia del estándar es que el único proceso de dimensionamiento sea realizado por medio de la métrica de puntos de función, pero esto se logrará a medida que el área madure en el proceso de medición por medio de dicha técnica.

Guía

Aplicación de los conceptos de dimensionamiento personal de aplicaciones acompañado por la aplicación de la métrica de puntos de función.

Producto

- Documento de cálculo personal de tamaño de aplicaciones.
- Formulario de cálculo de puntos de función.

4.3.3 DEFINICIÓN DEL PROCESO SOFTWARE

Análisis: En la actualidad EMTELCO S.A no adopta un proceso de desarrollo estándar, cada uno de sus integrantes escoge de acuerdo a su criterio de que manera desarrollar sus aplicaciones. El proceso de desarrollo estándar para los proyectos EMTELCO S.A será un proceso con un componente fijo en la definición y un componente variable:

Modelo de proceso fijo: Cualquiera sea el resultado del ejercicio de elección del componente variable de la selección del proceso de desarrollo de software todos los proyectos deberán ser desarrollados siguiendo como eje fundamental el proceso de desarrollo por componentes, conformando una biblioteca de componentes, evaluando que porcentaje de un proyecto puede ser desarrollado por componentes ya desarrollados y haciendo que cada proyecto que se desarrolle sea un componente más con el cual se puede contar para desarrollar proyectos posteriores.

Modelo de proceso variable: Esta parte en la elección del proceso de desarrollo de software en EMTELCO S.A permite que de acuerdo a la variable del pre análisis tamaño del proyecto en horas el cual se calcula con base en el dimensionamiento personal de cada uno de los integrantes se desarrolle un proceso de desarrollo diferente de la siguiente manera:

Proyectos de tamaño ≤ 340 Horas: Deben ser desarrollados por medio del modelo de proceso de desarrollo en Cascada

Proyectos de tamaño > 340 Horas: deben ser desarrollados por medio del modelo de proceso de desarrollo incremental y cada incremento a su vez siguiendo el proceso de desarrollo en cascada para cada incremento.

El modelo de proceso de desarrollo en cascada es ideal para la dinámica EMTELCO S.A debido a que en la mayoría de los casos el conocimiento de los requisitos del sistema es muy alto y estos no son muy cambiantes. Por otro lado es inmanejable un proyecto que supere cierto tamaño para ser manejado en cascada, por esta razón la variabilidad del proceso básicamente está en si se realiza por medio de incrementos o no. Sistemas grandes desde la óptica de EMTELCO S.A deberán ser entregados en incrementos priorizados y funcionales. Esta dinámica de trabajo permitirá evaluar la aplicación en tiempos tempranos y así permitir hacer correcciones sobre la marcha.

Nota: Si bien el estándar se desarrolla en una medida de gran importancia y manejo por parte de EMTELCO S.A como es el tamaño en horas de desarrollo, lo ideal es que la variable de análisis sean el número de puntos de función.

Guía:

De acuerdo a la variable de tamaño del proyecto en horas se define así:

Proyectos de tamaño ≤ 340 Horas: Modelo de proceso de desarrollo por Componentes, los componentes no desarrollados deben ser desarrollados en Cascada.

Proyectos de tamaño > 340 Horas: Modelo de proceso de desarrollo por Componentes en entregas incrementales y desarrollando cada incremento en Cascada.

Producto:

Referencia del modelo de proceso a desarrollar a lo largo del proyecto.

4.3.4 DISEÑO

Análisis

Por medio de los casos de uso dimensionados en la metodología de elicitación de requisitos hacer un proceso de refinamiento de casos de uso que muestren un croquis de la aplicación. Dichos casos de uso refinados serán sometidos a evaluación por parte del cliente dueño de la aplicación. Dicho documento se convertirá en contrato entre partes de lo que el cliente solicito y lo que el área de desarrollo se comprometió a desarrollar.

Adicionalmente es propuesto un formato de definición de entradas y salidas en búsqueda de estandarización de estos y reutilización de información.

Guía

Refinamiento de casos de uso a través del insumo provisto por la metodología de elicitación de requisitos. Cada caso de uso es un croquis textual de la aplicación real. Cada campo debe ser descrito detalladamente en el formato de entradas y salidas estándar.

Producto

- Casos de uso refinados como diseño.
- Casos de uso como contrato de desarrollo entre cliente y oficina de desarrollo de aplicaciones.
- Documento de entradas y salidas estándar.

4.3.5 DESARROLLO

Análisis

EMTELCO S.A actualmente no tiene una biblioteca de clases o componentes, tampoco enfoca sus esfuerzos para la reutilización de código. Siguiendo el modelo de proceso hallado en el paso de modelo de proceso de desarrollo cada componente será desarrollado siguiendo las siguientes instrucciones:

- Todas las funcionalidades de datos de una aplicación deberán ser implementados a través de servicio web. Estas a su vez deben ser gestionadas usando una clase principal de gestión de datos.
- Toda la lógica de desarrollo de una aplicación debe ser desarrollada a través de librerías de clases o componentes.
- Se debe desarrollar un Framework de componentes desarrollados de formularios e imagen. Este se alimentará de todos las aplicaciones que se convierten en insumo de desarrollo.

Guía

- Desarrollo de librerías de clases, componentes y servicios web.
- Reutilización de clase, componentes y servicios web para manejo de datos.

Productos

- Librería de componentes.
- Librería de Clases.
- Librería de Servicios web para manejo de datos y funcionalidades genéricas que consumen datos (manejo de fechas hábiles por ejemplo).

4.3.6 PRUEBAS E IMPLEMENTACIÓN

Análisis

Actualmente EMTELCO S.A no realiza un plan de pruebas preestablecido, cada uno de los integrantes del área realizan pruebas funcionales cada que terminan un proyecto y luego es entregado al cliente para que realice pruebas desde su óptica. El resultado de las pruebas genera un proceso de modificación o aceptación de la aplicación. Todo lo anterior ha resultado un modelo exitoso para EMTELCO S.A los niveles de devolución de aplicaciones para modificaciones es relativamente bajo, probablemente porque no existen los mecanismos para medir dicho indicador de la manera adecuada. Sin embargo dicho modelo es bastante efectivo.

Por el momento sólo se encuentra un mecanismo de Check-List para cualquier tipo de implementación tecnológica, mediante el cual se prueba

en un ambiente de desarrollo y posteriormente se corrobora en ambiente de producción. Cada Check-List debe ser desarrollado o hecho con base en otros pero no está establecido aún un estándar de Check-List. La propuesta metodológica está obligada a usar dicho modelo pues la documentación generada en dicho proceso hace parte de la documentación de control instaurada como política al interior de EMTELCO S.A.

Actualmente existen en el mercado algunos modelos de aplicación de pruebas al software de aplicación sin embargo los costos en tiempo de la mayoría de dichos modelos podría hacer que EMTELCO S.A pierda oportunidad en la atención de requerimientos de desarrollo de aplicaciones si le sumamos a esta razón el hecho que el modelo actual aplicado en la compañía es efectivo se puede concluir que el modelo actual puede permanecer pero quitando la posibilidad de ser opcional y fuera de eso realizando un formato estándar de pruebas.

Guía

Aplicación de Check-List establecido para cada tipo de desarrollo y hecho de manera estándar. Librería de Check-List o pruebas.

Desarrollo de pruebas por parte del analista desarrollador, el segundo analista y pruebas por parte del cliente en ambiente de pruebas. De acuerdo a los resultados de las pruebas implementar en ambiente de producción o llevar a correcciones de desarrollo o diseño dependiendo de la naturaleza del error.

Producto

- Formato de Check-List con resultados de pruebas.

- Formato de Pruebas con resultado de las pruebas aplicadas por el Analista desarrollador.
- Formato de Pruebas con resultado de las pruebas aplicadas por el cliente.

4.4 RECOMENDACIONES ADICIONALES

Separación por Roll

La especialización por roles del proceso de desarrollo de software es una práctica muy común por parte de las compañías productoras de software de mundo. Podría ser importante realizar un piloto al interior de EMTELCO S.A aplicando dicho modelo y verificar pros y contras de dicha implementación.

De acuerdo a un criterio personal podría ser un factor para hacerle sentir al cliente que tiene más oportunidad en la atención de sus requerimientos.

Reutilización de conocimiento e investigación

Debido a la alta velocidad con que las tecnologías cada vez son más avanzadas y sofisticadas así mismo las necesidades lo son. Es muy común ver áreas de desarrollo consultando o mejor investigando acerca de la manera de enfrentar una problemática de desarrollo. Es así como en la mayoría de las compañías herramientas como google se han vuelto cada vez más poderosas para generación de conocimiento: foros, páginas web dedicadas a temas de desarrollo, libros en línea etc.

En EMTELCO S.A cada analista tiene su banco de documentación o conocimiento pero no existe un banco central de consulta y conocimiento. Sería de un alto valor implementar una herramienta de centralización del conocimiento colectivo; Consultas comunes, foros internos, motivación a la aclaración de dudas. Adicionalmente un banco central de documentos, libros y cualquier otro tipo de material que apoye la labor del área en su proceso de desarrollo. De la misma manera la generación de metadata para dicho material sería crucial en mejoras de tiempo de consulta.

Un primer paso podría ser coleccionar todo el material documental disponible por analista para conformar uno colectivo.

Modelo de mejoramiento continuó

Con base en las cifras provistas por la propuesta metodológica de desarrollo asunto de esta tesis sería muy oportuno hallar los indicadores que puedan dar luces a un proceso de mejoramiento del área. Entre otros el tamaño de las aplicaciones y las mediciones de aplicaciones podrían ser un factor importante para proponer mejoras a la propuesta metodológica.

CONCLUSIONES

Los procesos de de estandarización de tareas cada vez están tomando mayor importancia en la empresa actual. La velocidad con la que la tecnología está avanzando devela un abismo entre lo ideal y la forma en la que realmente se están haciendo las cosas.

Este proyecto propone comenzar a cerrar dicha brecha en el proceso de desarrollo de software de la compañía EMTELCO S.A. Por medio del estudio de los diferentes modelos de procesos de desarrollo y sumado al conocimiento de la dinámica de la compañía hacen que se puedan dar conceptos para el desarrollo de ideas que guíen el desarrollo de software. Sin embargo es vital comprender que las propuestas que en una compañía se adopten no necesariamente son la solución a los problemas por los cuales se está pasando. En muchos casos las metodologías o procesos estandarizados generarán resistencia y por decirlo en otros términos algunos problemas de adopción. Pero el primer paso en un proceso de mejoramiento es identificar que problemas se tienen.

Unos de los principales problemas de las áreas de desarrollo de software de algunas compañías es que su labor se debe a esfuerzos individuales y conocimiento de individuo. La grave consecuencia está en la fuga de conocimientos. Las empresas no quieren asignar recursos a prácticas que produzcan frutos de largo plazo, por lo menos esta parece ser la señal enviada desde algunas gerencias a áreas de tecnología las cuales son enfocadas en políticas más reactivas que preventivas. Como consecuencia de esto cada vez existe más software no documentado lo que genera por supuesto altos costos de soporte (momento en el que los esfuerzos del pasado dan frutos).

El primer paso para combatir este mal es crear ideas que unifiquen los procesos al interior de las compañías, es poco probable que en un ambiente en el que todos los miembros de un equipo hablan el mismo idioma pueda prestarse para confusiones. Este es uno de los papeles preponderantes del estándar. Estos deben ser divulgados al interior de las compañías y socializados para que sean adoptadas y por supuesto mejorados.

Este proyecto da un paso adelante en este sentido; analiza la situación actual del área de desarrollo de aplicaciones de EMTELCO S.A y propone una guía para ser usada al interior como principio básico de consulta y como principio básico de mejoramiento de procesos.

Dentro del análisis una conclusión de suma importancia es el hecho que al interior del área no se está realizando un esfuerzo por desarrollar un enfoque de reutilización. Si se observan hacia donde están dirigidas las mejores prácticas es precisamente al re uso de componentes a todo nivel: re uso a nivel de análisis, re uso a nivel de diseño, re uso a nivel de desarrollo etc. Este principio básico debe ser la primera búsqueda o el primer escalón a subir en la búsqueda de mejoramiento del área de desarrollo de aplicaciones de la empresa EMTELCO S.A. Generación de una biblioteca común de conocimiento, componentes, aplicaciones, clases, diseños etc.

BIBLIOGRAFIA

Ian, Sommerville. Ingeniería de Software. Addison Wesley. Séptima edición.

Instituto colombiano de normas técnicas y certificación. Norma NTC-1486. Documentación. Presentación de Tesis, Trabajos de Grado, Otros trabajos de investigación. Sexta actualización. Julio 23 de 2008.

Pressman, Roger S. Ingeniería del Software, Un enfoque práctico. Quinta edición. McGraw Hill

Análisis de capacidad de la productividad individual y en equipo en empresas que desarrollan o adquieren productos de Software. Proyecto de grado. Claudia Gallego, Maritza Montes, Maribel Quintero. 2004.

Larman, C.: Uml y Patrones : Introducción al análisis y diseño orientado a objetos , 2e. PrenticeHall. 2005. 627 p.

Gamma E., Helm, R., Johnson, R., Vlissides J.: Design Patterns: Elements of Reusable Object Oriented Software, Addison Wesley, 1995.

Lenguaje Unificado de Modelado.
http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado

Puntos por Función. Una métrica estándar para establecer el tamaño del software. Sergio Eduardo Duran Rubio.
<http://www.inegi.org.mx/inegi/contenidos/espanol/prensa/Contenidos/Articulos/tecnologia/puntosxfuncion.pdf>

Metodología para la Elicitación de Requisitos de Sistemas Software Versión 2.1. Amador Durán Toro, Beatriz Bernardez Toro. Universidad de Sevilla. Octubre de 2000. <http://www.lsi.us.es/~informes/lsi-2000-10.pdf>

The Software Engineering Institute. Carnegie Mellon University

[.http://www.sei.cmu.edu](http://www.sei.cmu.edu).

The team process software: an overview and preliminary results of using disciplined practices. <http://www.sei.cmu.edu/tsp/index.cfm>

Departamento de lenguajes y sistemas informáticos, Universidad de Sevilla
<http://www.lsi.us.es/>

Ejemplo de desarrollo de software utilizando la metodología RUP

<http://users.dsic.upv.es/asignaturas/facultad/lsi/ejemplorup/>

Fundamentos de ingeniería del software, Joaquín Nicolás Ros, Universidad de Murcia
http://dis.um.es/~jnicolas/09BK_FIS.html

Philip Greenspun, Eve Andersson and Andrew Grumet, Software Engineering for Internet Applications, The MIT Press

Ariadne training, UML Applied, Object Oriented Analysis And Design Using the UML.
<http://www.ariadnetraining.co.uk/software-courses/index.php>

Choose the Right Software Method for the Job

<http://www.agiledata.org/essays/differentStrategies.html#Table1ComparingMethodologies>

Portal desarrollo de software, Universidad Politécnica de Valencia (UPV)
<https://pid.dsic.upv.es/default.aspx>

Scott W. Amber, Agile Database Techniques Effective Strategies for the Agile Software Developer, John Wiley & Sons

Joseph Schmuller, Aprendiendo UML en 24 horas, Pearson Educación Latinoamérica

Eric Freeman, Elisabeth Freeman, Kathy Sierra and Bert Bates, Head First Design Patterns, O'Reilly

Anexo 1

Evaluación de la situación actual del área de desarrollo EMTELCO S.A

FICHA TECNICA
Número de Empleados: 13
Muestra: 13
Metodología: Uso del correo electrónico Entrevistas Personales
Fuente: Elaboración Propia

Objetivo: Evaluación para determinar el estado actual del área de desarrollo de EMTELCO S.A. También se incluyeron preguntas que permitieran encontrar posibles de modelos de procesos a aplicar en el área.

Fuente: Elaboración propia y algunas preguntas fueron tomadas de la encuesta

Preguntas jefe de área 1 a 14.

1. La Empresa en cuanto a Software:

Desarrolla _____ Adquiere _____ Ambas _____

2. ¿Cuál es el número total de empleados en su compañía?

3. ¿Cuál es el número total de empleados dedicados al desarrollo de software en su compañía?

4. ¿Elija los diferentes tipos de aplicaciones que son desarrolladas en su compañía?

- a. Aplicaciones de Sistema de control y automatización industrial
- b. Aplicaciones ofimáticas
- c. Software educativo
- d. Software empresarial
- e. Bases de datos
- f. Telecomunicaciones

- g. Videojuegos
- h. Software médico
- i. Software de Cálculo Numérico
- j. Software de Diseño Asistido (CAD)
- k. Software de Control Numérico (CAM)
- l. Otros _____

5. ¿Cuál o cuáles modelos de distribución de aplicaciones usa para los proyectos de la compañía?

- a. Stand Alone
- b. Web
- c. Distribuidas
- d. Web Services
- e. Otros _____

6. ¿Tienen un modelo de proceso de Software establecido en su compañía?

Sí ___ No ___

Cual

- a. En cascada
- b. Incremental Evolutivo
- c. Por componentes
- d. Por prototipos
- e. Espiral
- f. Otro _____

7. ¿Cuál es la desviación de los proyectos con respecto a la estimación inicial de tiempo?

- a. 0% – 20% ___
- b. 20% – 40% ___
- c. 40% - 60% ___
- d. 60% - 80% ___
- e. 80% - 100% ___

8. ¿Cuáles son los principales factores a los que se debe el desfase con respecto a la estimación inicial de tiempo?

- a. Falta de información por parte de los usuarios
- b. Especificaciones y requisitos incompletos
- c. Especificaciones y requisitos cambiantes
- d. Tiempos muy ajustados exigidos por el negocio
- e. Mal dimensionamiento de tiempos y recursos
- f. Falta de elementos tecnológicos necesarios
- g. No seguimiento de un modelo de proceso de desarrollo estándar
- h. Otro _____

9. Al mes ¿qué porcentajes de requerimientos se atienden de cada clasificación de proyectos (De los que haya seleccionado en la pregunta 4)?

- 1. _____
- 2. _____
- 3. _____
- 4. _____

10. Teniendo en cuenta que en su empresa el tamaño de un proyecto de desarrollo de software se determina en el número de horas que este puede tomarse. Defina los porcentajes de proyectos que se encuentran en las siguientes escalas de tiempo:

Menos de 85 horas _____ %

Entre 85 y 170 horas _____ %

Entre 170 y 340 horas _____ %

Más de 340 horas _____ %

11. ¿Hacia qué tipo de usuarios están dirigidas las aplicaciones que se desarrollan en el área de desarrollo de EMTELCO S.A?

- a. Usuarios finales en general (Software de propósito general)
- b. Personal de la compañía
- c. Personal externo a la compañía
- d. Usuarios finales de otras compañías
- e. Otros _____

12. Cuando se desarrolla un proyecto de desarrollo de software enuncie la documentación que este debe tener asociada.

- a. Manuales de usuario
- b. Guía de implementación y soporte
- c. Documentos de análisis de requisitos y diseño de la aplicación
- d. Contratos
- e. Otros documentos de control
- f. Otros _____

13. Cuáles son los factores críticos en la dinámica propia de su compañía, que interfieren en el buen desarrollo de software en su empresa (Implementación de buenas prácticas).

14. En cuanto al uso de recursos, los proyectos son asignados a:

a. Un equipo de trabajo.

b. Una persona.

c. Ambos.

Si la respuesta es c. defina el porcentaje de proyectos que obedecen a cada caso:

Equipo_____%

Individual_____%

Si desea agregar algo lo puede escribir en esta parte

Preguntas analistas, 15 a 21

15. De los siguientes tipos de software, en que porcentaje son asignados a usted

- a. Aplicaciones ofimáticas
- b. Software educativo
- c. Software empresarial
- d. Bases de datos
- e. Telecomunicaciones
- f. Videojuegos
- g. Software médico
- h. Software de Cálculo Numérico
- i. Software de Diseño Asistido (CAD)
- j. Software de Control Numérico (CAM)
- k. Otros _____

16. Qué modelo o modelos de proceso de software usa para sus asignaciones:

- a. En cascada
- b. Incremental Evolutivo
- c. Por componentes
- d. Por prototipos
- e. Espiral
- f. Ninguno
- g. Otro _____

17. Al momento de desarrollar sus aplicaciones ¿qué tan familiarizado se encuentra usted con los requisitos de sus asignaciones?:

- a. 0-20% _____
- b. 20-40% _____
- c. 40-60% _____

d. 60-80% _____

e. 80-100% _____

18. Cuando le es asignado el desarrollo de una aplicación ¿en qué porcentaje le piden cambios en los requisitos, en etapas posteriores a la etapa de definición de requerimientos?

a. 0-20% _____

b. 20-40% _____

c. 40-60% _____

d. 60-80% _____

e. 80-100% _____

19. Cuando desarrolla aplicaciones del mismo tipo, si pensara en la posibilidad de desarrollar una aplicación utilizando componentes de otras aplicaciones ya desarrolladas, ¿qué porcentaje de la nueva aplicación podría desarrollar utilizando los componentes ya desarrollados en proyectos anteriores?

_____ %

20. De acuerdo a la naturaleza de los proyectos que le son asignados ¿considera usted que deberían ser desarrollados realizando entregas parciales?

a. No

b. Si

¿Porque?

21. Asumiendo que su compañía adopta un modelo de proceso de desarrollo de software, enumere cada una de las siguientes opciones de 1 a 6, siendo 1 la opción de mayor importancia y 6 la menor. (Tenga en cuenta su experiencia y el tipo de aplicaciones que le son asignadas, no considere el orden ideal).

El modelo de procesos de desarrollo de software me debe permitir realizar cambios en la marcha de los requerimientos iniciales de la aplicación. _____

El modelo de proceso de desarrollo de software debe ser rápido y exigir poca documentación. _____

El modelo de proceso de desarrollo de software debe permitirme hacer entregas parciales (hitos) de la aplicación _____

El modelo de proceso de desarrollo de software debe estar orientado a la reutilización de código y clases o componentes que ya han sido desarrolladas en otros proyectos. _____

El modelo de proceso de desarrollo de software debe ser la mezcla de diferentes modelos de procesos que se adaptan según el tipo de software. _____

El modelo de proceso de desarrollo de software debe ser flexible de tal manera que puedan realizarse u omitirse algunas de las actividades según sea el caso. _____

22. Teniendo en cuenta su experiencia y el tipo de aplicaciones que le son asignadas ¿considera usted que sería útil la implementación de metodologías de desarrollo de prototipos funcionales que son mejorados con base en el nivel de conocimiento de los requerimientos del cliente?

a. Si

b. No

¿Por qué?

23. ¿Cuál o cuáles modelos de distribución de aplicaciones usa para los proyectos de la compañía y en qué porcentajes. Las opciones que escoja deben sumar un 100%?

- a. Stand Alone _____ %
- b. Web _____ %
- c. Distribuidas _____ %
- d. Web Services _____ %
- e. Otros Cual _____ %

Si desea agregar algo lo puede escribir en esta parte

Anexo 2

Plantilla de casos de uso (Use Case Template)

Fuente: Use Case Template www.processimpact.com. Versión traducida por Jose López

Identificador	
Nombre:	
Creador por:	Última actualización por:
Fecha de creación:	Fecha de la última actualización:

Actores:	
Descripción:	
Iniciador:	
Precondiciones:	1.
Pos condiciones:	1.
Flujo Normal:	1.0.1
Flujos Alternativos:	
Excepciones:	

Inclusiones:	
Prioridad:	
Frecuencia de uso:	
Reglas de negocio:	•
Requisitos especiales:	
Supuestos:	•
Notas y pendientes:	

Historial de revisiones

Nombre	Fecha	Razón del cambio	Versión