

Universidad de las Ciencias Informáticas

Facultad 4



Título: Propuesta de métrica de productividad para el proceso de producción de software en la Facultad 4.

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autor: Mailen Rivera Acosta

Tutor(es): Ing. Lizandra Arza Pérez

Ing. Raydel Muñoz Vidal

Consultante: Lic. Olga Lidia Martínez Acosta

Julio, 2007

DECLARACIÓN DE AUTORÍA

Declaro ser autora 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 _____.

Insertar nombre de Autor

Insertar nombre de tutor

Insertar nombre de tutor

“Cuando puedes medir lo que estas diciendo y expresarlo en número sabes algo sobre ello; pero cuando no puedes medirlo, cuando no puedes expresarlo con números, tu conocimiento es escaso e insatisfactorio: puede ser el comienzo del conocimiento pero en tus pensamientos, apenas estas avanzando hacia el escenario de la ciencia.”

Lord Kelvin

Agradecimientos

Agradezco a mi familia por haber confiado en mí, por haberme apoyado siempre.

Al amor de mi vida que desde lejos estuvo siempre conmigo, a su madre que me apoyo en todo momento dándome fuerzas para seguir adelante.

A mis amigos que siempre me apoyaron cuando más los necesitaba, a todos ellos muchas gracias.

A Olga Lidia y Laritza que me hicieron fuerte cuando creí que no llegaba al final, gracias por estar ahí.

A mis tutores que sin la colaboración de ellos no habría sido posible la realización de este trabajo.

Al profesor Loan Joa, quien me ha apoyado muchísimo, muchas gracias por tu ayuda.

A todos muchas gracias

Dedicatoria

A mi mamá Maritza Acosta Gamboa y mi abuela Adelina Gamboa Áreas que son mis dos pupilas por estar siempre conmigo y darme lo mejor de sí para formarme en la persona que soy hoy.

A mi tía Vilma por quererme tanto por ser tan especial para mí, por su confianza, a mis tías Odelinda y Magaly por aconsejarme por apoyarme tanto.

A mis hermanos Maykel y Kendrys que por protegerme y cuidarme siempre y adorarme con la vida.

A Paula Caballero por su apoyo, sus consejos, su fortaleza para salir adelante y su cariño.

Resumen

La investigación propone un sistema de métricas para controlar la productividad en el proceso de producción de software en la Facultad 4. Para ello se realizó un estudio del estado del arte de las métricas y su aplicación en el proceso de desarrollo de software. Se realizaron encuestas a los líderes de los proyectos más importantes de la facultad para determinar cómo se controla la productividad en los mismos y como resultante se hizo la propuesta del sistema de métricas que responde a las necesidades de los proyectos en cuestión proponiendo una guía de aplicación del mismo.

Palabras Claves: PROCESO DE DESARROLLO DE SOFTWARE, METRICA, PRODUCTIVIDAD.

Índice

Introducción	1
Capitulo1: Fundamentación Teórica	1
1.1 Introducción.....	1
1.2 Producción de software.....	1
1.3 Proceso de desarrollo de software.....	2
1.4 Ingeniería de software.....	4
1.5 Calidad del software.....	5
1.5.1 Características debe tener el software de alta calidad	6
1.5.2 Terminologías sobre Calidad	6
1.5.3 Aseguramiento de la calidad	7
1.6 Métricas de software	8
1.6.1 Definiciones	8
1.6.2 Tipos de métricas	9
1.6.2 Estándares internacionales	14
1.6.3 Métricas de productividad.....	20
1.7 Conclusiones.....	22
Capitulo 2: Solución Propuesta.....	23
2.1 Introducción.....	23
2.2 Análisis de las métricas de productividad que existen	23
2.3 Estudio del proceso de producción.	31
2.3.1 Elementos del proceso	31
2.3.2 Proceso de desarrollo de los proyectos de la facultad 4.....	31
2.3.3 Control de la productividad en el proceso productivo de la Facultad.....	32
2.3.4 Métricas en el proceso de producción de la Facultad.....	33
2.4 Definición de la métrica	33
2.4.1 Estructura del sistema de métricas propuesta.....	35

2.4.2 Guía de aplicación de la propuesta	39
2.5 Conclusiones.....	41
Capitulo 3: Validación de la Solución Propuesta	43
3.1 Introducción.....	43
3.2 Validación de métricas	43
3.3 Métricas de productividad en el proyecto	44
3.4 Métrica de productividad vs. Procedimiento del proyecto (SIGEP)	45
3.5 Proyecto SIGEP	46
3.6 Productividad en el proyecto SIGEP	46
3.7 Conclusiones.....	47
Conclusiones	49
Recomendaciones	50
Bibliografía.....	51
Anexos.....	55

Introducción

Entre las principales directivas de Cuba se encuentra sustentar sus ingresos a través de sus producciones intelectuales y no de su industria tradicional. El software es una producción puramente intelectual, es por ello que actualmente las perspectivas de la industria cubana de software están encaminadas a la capacitación del personal y la inserción de los productos que en ella se producen en el mercado nacional e internacional.

La producción de software en Cuba se ha desarrollado de forma vertiginosa abarcando diferentes esferas del país como: economía, salud y educación.

En el año 2000 se crea el Ministerio de Informática y las Comunicaciones (MIC). Entre sus principales metas se encuentran: socializar el uso de las técnicas de la Informática y las Comunicaciones (TIC) y convertir la informática en una rama que aporte beneficios a la economía del país. Para lograr estos objetivos se trazaron varias acciones como resultado de una de ellas fue creada en el año 2002 la Universidad de las Ciencias Informáticas (UCI) adscripta a dicho ministerio, cuya misión es: formar profesionales comprometidos con su Patria y altamente calificados en la rama de la informática y producir software y servicios informáticos, a partir de la vinculación estudio-trabajo como modelo de formación.

La UCI, institución universitaria donde la producción de software es una de sus actividades cotidianas y cuyos esfuerzos van dirigidos a la informatización de la sociedad cubana, consta de 10 facultades y cada una de ellas responde a un perfil determinado, en otras palabras las facultades constituyen pequeñas entidades que producen software y todas tienen como objetivo en común: la inserción de estos productos en el mercado nacional e internacional.

Sin embargo a pesar de haber producido varios productos requiere de algunos factores para alcanzar este objetivo entre los que se encuentra la calidad de los productos.

Para obtener una calidad es necesario tener procedimientos y normas que permitan a la institución evaluar si los productos han alcanzado la calidad requerida contribuyendo a la toma

de decisiones por parte de la dirección de la institución en función de alcanzar los objetivos propuestos. Las organizaciones actualmente para alcanzar niveles de calidad tienden al uso de sistemas de métricas. Las métricas se utilizan para evaluar y controlar el proceso de desarrollo del software, de forma que permitan:

- Indicar la calidad del producto.
- Evaluar la productividad de los desarrolladores.
- Evaluar los beneficios (en cuanto a calidad y productividad) derivados del uso de nuevos métodos y herramientas de ingeniería del software.

En la UCI, donde la producción y el rendimiento de sus estudiantes están en la misma escala de prioridades, se impone el control de la productividad en los procesos productivos de sus facultades, por ello se han definido métricas

que permitan llevar a cabo este control pero no se han obtenido resultados objetivos por parte de los proyectos.

Es por ello que surge esta investigación planteándose como **problema**:

No existe una métrica que permita medir la productividad en el proceso de producción, de la Facultad 4 de la Universidad de las Ciencias Informáticas.

A modo de solución al problema planteado se hace necesario un estudio profundo y detallado de las métricas existentes y la selección de la que se ajuste al proceso de producción en el marco definido, en este caso la facultad 4, por lo que se define como **objeto de estudio** el proceso de desarrollo de software de la Facultad 4 de la Universidad de las Ciencias Informáticas.

Como **campo de acción** se define: las métricas de productividad para el proceso productivo de la Facultad 4 de la Universidad de las Ciencias Informáticas.

EL **objetivo general** de esta investigación es: **proponer una métrica o conjuntos de ellas para controlar la productividad en el proceso productivo de la Facultad 4** a partir del estudio

realizado sobre el tema con el fin de proporcionar a los directivos de esta área información que permita tomar decisiones para la mejora del proceso.

Objetivos específicos:

- Realizar un estudio de uso de las métricas en el proceso de desarrollo de software.
- Definir métricas para el control de la productividad en la Facultad 4.
- Definir la estructura de validación de la métrica propuesta.

Para dar cumplimiento a los objetivos presentados anteriormente se concretan las siguientes **tareas:**

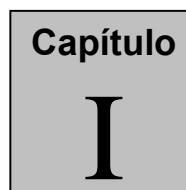
- Estudiar y analizar la relación entre las métricas y proceso de desarrollo de software, ingeniería de software.
- Analizar el proceso de producción de la Facultad.
- Estudiar y definir las métricas a utilizar.
- Definir una guía de aplicación de las métricas propuestas
- Estudiar las vías de validación de las métricas propuestas.

El presente trabajo consta de introducción, tres capítulos y conclusiones.

El Capítulo 1: **Fundamentación Teórica** aborda los temas relacionados con las métricas, conceptos, tipos y reflexiones que colaboran al esclarecimiento de la importancia de su aplicación en las entidades en el proceso productivo.

El Capítulo 2: **Propuesta de la métrica de productividad:** en este capítulo se realiza un estudio del proceso productivo de la facultad, y uno que otro elemento del proceso de la Universidad, se analiza lo que plantea la literatura sobre calcular la productividad, y a partir de aquí, se define la métrica propuesta, estableciendo una estrategia para su uso en la facultad.

El Capítulo 3: **Análisis de los resultados:** se presentan elementos que validan la propuesta, demostrando que es factible su uso en los proyectos de la Facultad 4.



Capítulo1: Fundamentación Teórica

1.1 Introducción

En este capítulo se pretende fundamentar el uso de las métricas en el proceso de desarrollo de software, y cómo resulta de gran importancia en la aplicación de un buen proceso de Ingeniería de Software, de una buena calidad del software.

1.2 Producción de software

La producción de software es una actividad compleja que requiere tiempo y esfuerzo y en la cual influyen determinadamente factores tales como: el uso de herramientas, metodologías de desarrollo entre otros.

En la actualidad la industria del software se enfrenta a retos como: mercados competitivos, clientes exigentes, la entrega de productos en menos tiempo del acordado, demanda de alto niveles de calidad, lo que hace que el control de la misma sea crucial en el proceso de desarrollo de software. Esto conlleva a que las empresas tiendan a incluir dentro de su proceso de desarrollo actividades como la gestión de la calidad y la obtención de indicadores de productividad para el mejoramiento de la producción.

Medir la productividad en vista a realizar comparaciones por parte de las empresas, ya sea entre sus proyectos productivos o con otras empresas productoras de software se ha vuelto necesario por las características del mercado y los clientes a los que se enfrentan.

Según vaya avanzando el capítulo se analizarán temas relacionados con las métricas, destacando la necesidad actual que se impone de utilizar las mismas, entre dichos temas se

destacan por ejemplo: proceso de desarrollo de software, ingeniería de software, calidad del software, métricas, entre otros.

1.3 Proceso de desarrollo de software.

Proceso de desarrollo de software: se define como el conjunto de actividades necesarias para convertir los requisitos de usuario en un conjunto persistente de artefactos que conforman el producto software. El proceso define quién está haciendo qué, cuándo hacerlo y cómo alcanzar un cierto objetivo. (Pressman 2005)

No existe ningún proceso que sea universal, el proceso debe ser configurable al proyecto que desarrolla el producto.

La formalización del proceso de desarrollo se define como un marco de referencia denominado ciclo de desarrollo del software o ciclo de vida de desarrollo del software. Se puede describir como, "el período de tiempo que comienza con la decisión de desarrollar un producto software y finaliza cuando se ha entregado éste", que comprende cuatro grandes fases: concepción, elaboración, construcción y transición.

- La concepción define el alcance del proyecto y desarrolla un caso de negocio.
- La elaboración define un plan del proyecto, especifica las características y fundamenta la arquitectura.
- La construcción crea el producto
- La transición transfiere el producto a los usuarios.

1

¹ Artefacto: Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.

El proceso del software se puede caracterizar como:

- Marco común del proceso: conjunto de actividades del marco de trabajo que son comunes y aplicables en todos los proyectos independientemente de su tamaño y complejidad.
- Conjuntos de tareas: cada una es una colección de tareas de trabajo de ingeniería de software, hitos del proyecto, productos del trabajo y puntos de garantías de calidad que permiten que las actividades del marco se adapten a las características del proyecto
- Actividades de protección: como garantía de calidad del software, gestión de configuración y medición. (Pressman 2005)

El tema de las mediciones debe estar presente durante todo el proceso, la utilización de métricas es muy importante para obtener mejoras en el proceso de desarrollo del software. La única forma de mejorar cualquier proceso, es medir atributos del mismo y utilizar estas métricas para obtener indicadores que conducirán a una estrategia de mejora. Estos atributos pudieran ser la calidad y productividad ya sea del proceso o del proyecto.

La productividad y calidad del software son generalmente elementos que están presentes durante el proceso de desarrollo. La productividad sirve para mejorar el proceso de desarrollo y la calidad para mejorar el producto final. La obtención de mediciones del esfuerzo da al gestor una idea del esfuerzo necesario para llevar a cabo el proceso de desarrollo, posibilitando al mismo determinar metas y objetivos razonables. (Pressman 2005)

Una organización podría variar su modelo de proceso para cada proyecto, según:

- ✓ La naturaleza del proyecto
- ✓ La naturaleza de la aplicación
- ✓ Los métodos y herramientas a utilizar
- ✓ Los controles y entregas requeridas.

Las fases genéricas de un proceso de desarrollo de software son: definición, desarrollo y soporte que son aplicables en todo el proceso. El problema está en determinar el modelo a utilizar para llevar a cabo la ingeniería de software para desarrollar el software. Estos sucesos se entienden mejor después de saber en que consiste la Ingeniería de Software.(Pressman 2005)

1.4 Ingeniería de software

Ingeniería de software: es una tecnología multicapa en la que se identifican los métodos, los procesos y herramientas para construir un producto software. (Pressman 2005)

El proceso de ingeniería de software se define como "un conjunto de etapas parcialmente ordenadas con la intención de lograr un objetivo, en este caso, la obtención de un producto de software con calidad. (Zavalar 2002)

Existe cierta tendencia a evitar la ingeniería del software, por considerar que burocratiza en exceso el desarrollo de proyectos, quizá por desconocimiento de algunas de las técnicas o por la sensación de pérdida de tiempo que genera la realización de tareas consideradas accesorias que no impliquen la mera construcción del producto final. Pero la realidad demuestra que cuando se acometen grandes proyectos sin rigor metodológico se desembocan normalmente fracasos, sobre todo en proyectos grandes. Los costes se disparan, las fechas no se cumplen y la calidad del producto no es la esperada. La correcta utilización de la ingeniería del software, aunque en un primer momento pueda parecer tediosa, conduce a una disminución de los costes y al cumplimiento de los plazos de entrega.

Sin embargo, para llevar los proyectos a buen término no basta con aplicar ingeniería del software. Hay que hacerlo racionalmente, sin despilfarrar recursos y utilizando en cada momento las técnicas más adecuadas.

Existen actividades protectoras en el ambiente de la ingeniería de software (Pressman 2005)

- ✓ Seguimiento y control del proyecto de software
- ✓ Revisiones técnicas formales
- ✓ Garantía de la calidad del software

- ✓ Gestión de configuración de software
- ✓ Preparación y producción de documentos
- ✓ Gestión de reutilización
- ✓ Mediciones
- ✓ Gestión de riesgos

Lo anterior demuestra que las mediciones es una de las actividades protectoras dentro de la ingeniería de software siendo estas las que se aplican a lo largo de todo el proceso de desarrollo.

La recopilación de métricas de productividad sintoniza el proceso de ingeniería de software de la organización para eliminar las causas de los defectos que tienen el mayor impacto en el desarrollo del software. (Pressman 2005)

El objetivo principal de la ingeniería de software es producir un sistema, aplicación o producto de alta calidad. Para lograr este objetivo la ingeniería brinda metodologías de desarrollo y técnicas, pero no es suficiente, puesto que no se mide si el proceso de ingeniería se esta llevando a cabo con la calidad requerida. Por lo que medir el proceso de ingeniería de software es también necesario y no solo el proceso de ingeniería de software sino medir la calidad de los productos finales nos da una estimación de si la aplicación de las técnicas de ingeniería se aplicaron correctamente.

Una buena práctica de ingeniería de software radica en aplicar mediciones a la calidad de análisis, diseño, pruebas. A partir de esto se concluye que medir la calidad es tan importante como definir los requisitos en un sistema en desarrollo los cuales describen el problema a resolver.(Pressman 2005)

1.5 Calidad del software

La calidad del software es el conjunto de cualidades que lo caracterizan y que determinan su utilidad y existencia. La calidad es sinónimo de eficiencia, flexibilidad, corrección, confiabilidad, mantenibilidad, portabilidad, usabilidad, seguridad e integridad. (Carrasco, León et al. 1995)

La calidad del software es medible y varía de un sistema a otro o de un programa a otro. Un software elaborado para el control de naves espaciales debe ser confiable al nivel de "cero fallas"; un software hecho para ejecutarse una sola vez no requiere el mismo nivel de calidad; mientras que un producto de software para ser explotado durante un largo período (10 años o más), necesita ser confiable, mantenible y flexible para disminuir los costos de mantenimiento y perfeccionamiento durante el tiempo de explotación. (Carrasco, León et al. 1995)

1.5.1 Características debe tener el software de alta calidad

○ Utilidad:

Determina si el sistema hace lo que sus usuarios esperan que haga

El sistema es útil si:

Su funcionalidad ayuda al usuario en la realización de sus actividades

○ Confiabilidad:

Baja probabilidad de falla o bajo número de errores

○ Mantenibilidad:

¿Es flexible, es modificable y es internamente comprensible?

○ Eficiencia:

Medida en relación al uso de los recursos del computador

Ejemplo: bajo tiempo de ejecución y mínimo uso del espacio de almacenamiento

○ Facilidad de prueba:

¿Se puede probar con facilidad?

En qué medida el diseño y codificación facilitan las pruebas.

1.5.2 Terminologías sobre Calidad

Gestión de calidad (SQM): Gestión de Calidad: Determinación y aplicación de las políticas de calidad de la empresa (objetivos y directrices generales). (Antonio 2004)

Control de calidad del software (SQC): Actividades para evaluar la calidad de los productos desarrollados. (Antonio 2004)

Aseguramiento de calidad (SQA): Conjunto de actividades planificadas y sistemáticas necesarias para proporcionar confianza en que el producto software satisfará los requisitos dados de calidad.(Antonio 2004)

1.5.3 Aseguramiento de la calidad

- Establecimiento de un sistema de calidad
 - ✓ Gestión de la calidad
 - ✓ Planificación de la calidad
 - ✓ Definición de políticas de calidad
- Uso de técnicas de verificación y validación del software
 - ✓ Revisiones e inspección de los productos de software
 - ✓ Pruebas de programas
- Gestión de la Configuración del Software
- Uso de normas y estándares de calidad
- Evaluación y mejoramiento de los procesos de software.

¿COMO CONTROLAR LA CALIDAD DEL SOFTWARE?

Para controlar la calidad del software es necesario, ante todo, definir los parámetros, indicadores o criterios de medición, ya que, como bien plantea Tom De Marco, "usted no puede controlar lo que no se puede medir".

Las cualidades para medir la calidad del software son definidas por innumerables autores, los cuales las denominan y agrupan de formas diferentes. Por ejemplo, John Wiley define métricas de calidad y criterios, donde cada métrica se obtiene a partir de combinaciones de los diferentes criterios.

La Metodología para la evaluación de la calidad de los medios de programas de Rusia, define indicadores de calidad estructurados en cuatro niveles jerárquicos: factor, criterio, métrica, elemento de evaluación, donde cada nivel inferior contiene los indicadores que conforman el nivel

precedente. Otros autores identifican la calidad con el nivel de complejidad del software y definen dos categorías de métricas: de complejidad de programa o código, y de complejidad de sistema o estructura.

Todos los autores coinciden en que el software posee determinados índices medibles que son las bases para la calidad, el control y el perfeccionamiento de la productividad.

Una vez seleccionados los índices de calidad, se debe establecer el proceso de control, que requiere los siguientes pasos:

- Definir el software que va a ser controlado: clasificación por tipo, esfera de aplicación, complejidad, de acuerdo con los estándares establecidos para el desarrollo del software.
- Seleccionar una medida que pueda ser aplicada al objeto de control. Para cada clase de software es necesario definir los indicadores y sus magnitudes.
- Crear o determinar los métodos de valoración de los indicadores: métodos manuales como cuestionarios o encuestas estándares para la medición de criterios periciales y herramientas automatizadas para medir los criterios de cálculo.
- Definir las regulaciones organizativas para realizar el control: quiénes participan en el control de la calidad, cuándo se realiza, qué documentos deben ser revisados y elaborados.

Como se aprecia el uso de métricas es también determinante en el control de la calidad de los productos software.

1.6 Métricas de software

1.6.1 Definiciones

- Medida. Proporciona una indicación cuantitativa de extensión, cantidad, dimensiones, capacidad y tamaño de algunos atributos de un proceso o producto. Pueden

ser directas, por ejemplo número de líneas de código, número de errores encontrados, o pueden ser indirectas, como funcionalidad, calidad y complejidad. (Navarro 2003)

- Medición. Acto de determinar una medida. (Navarro 2003)
- Métrica. Es una medida cuantitativa del grado en que un sistema o proceso posee un atributo dado. (Navarro 2003)
- Indicador. Es una métrica o combinación de métricas que proporcionan una visión del proceso, del proyecto o del software en sí, y poder hacer ajustes para que las cosas mejoren. (Navarro 2003)

Métricas: es una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado.

1.6.2 Tipos de métricas

Existen diferentes **tipos de métricas** entre las que se encuentran: (Giraldo 2004)

- Métricas de productividad: rendimiento del proceso de la IS.
- Métricas de calidad: ajuste a los requisitos implícitos y explícitos.
- Métricas técnicas: se centra en las características del software no en cómo se obtiene.
- 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 se harán del personal que desarrollará el sistema.
- Métricas orientadas al tamaño: es para saber en que 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.

Las métricas del software responden a dos objetivos: valorar y estimar.

Las magnitudes objeto de valoración son tres: la calidad, fiabilidad y productividad. La estimación parte de mediciones históricas para prever el esfuerzo y el tiempo que debe invertirse en un proyecto dado, y las características del resultado final.

Las métricas de calidad y productividad se pueden utilizar en el proyecto para el control de la calidad, evaluación de la productividad y en el control de proyectos.

Hay cuatro razones para medir los procesos:

- Caracterizar: para conocer mejor los procesos, los productos y los entornos para establecer comparaciones.
- Evaluar: para valorar la calidad y el impacto de las tecnologías y las mejoras del proceso.
- Predecir: para establecer objetivos alcanzables de costes, planificación y calidad.
- Mejorar: cuando se recoge información cuantitativa que permite identificar problemas e ineficiencias del proceso, oportunidades para mejorar la calidad del producto y el rendimiento del proceso. (Pressman 2005)

Las métricas del software abarcan muchas actividades y son múltiples las razones que justifican su uso:

- Estimación de coste y esfuerzo (o al menos reducción de estos)
- Modelos y medidas de productividad
- Recolección de datos
- Modelos y evaluación de la calidad
- Modelos de fiabilidad
- Están incluidos en la mayoría de los modelos de calidad.
- La especialización de los modelos de fiabilidad permite aumentar el entendimiento y control de los productos.
- Evaluación del rendimiento
 - ✓ Aunque es otro aspecto de la calidad, la valoración del rendimiento incluye características observables como tiempos de respuesta y características internas como eficiencia de los algoritmos.
- Métricas estructurales y de complejidad

- Para realizar predicciones sobre atributos de calidad (fiabilidad, facilidad de mantenimiento) se pueden medir atributos estructurales sobre representaciones del software que están disponibles antes que el código.
- Se pueden medir diferentes atributos del desarrollo para evaluar la capacidad de una organización de desarrollar software de calidad.
- Gestión mediante métricas
- La realización de gráficos basados en diferentes medidas a lo largo del proyecto permite conocer el estado del mismo.
- Evaluación y comparación de métodos y herramientas
- Las investigaciones cuidadosas, con análisis y mediciones controladas sobre una herramienta o método permiten hacerlos más productivos para situaciones particulares.
- Justificación del uso de nuevos métodos y /o herramientas
- Justificación de formación adicional

Las métricas se pueden clasificar en: Métricas del producto (Datos de desarrollo, Lógica), y métricas del proceso (Tiempo de desarrollo, Reusabilidad, Productividad). Las primeras son las que tradicionalmente se han empleado en la medida del software, y pueden ser obtenidas automáticamente tomando como entrada el código fuente: tamaño, estructuras de datos y lógica. Las métricas del proceso por su parte dependen esencialmente del entorno de desarrollo. Un ejemplo de este tipo de métricas es el tiempo empleado para desarrollar un elemento software que depende de distintos factores externos (dificultad del problema, capacidad del personal, metodología empleada).

Las métricas no se miden, se obtienen a través de medidas que se hacen ya sea a proyectos o al proceso de desarrollo.

Objetivos de las medidas:

- **Evaluación:** comprobación del cumplimiento de ciertas características por una entidad que ya existe (calidad del diseño, fiabilidad del software).

- **Predicción:** estimación de los atributos que tendrá una entidad que no existe aún (coste de un proyecto, esfuerzo necesario).

El proceso se mide para mejorarlo, rentabilizarlo y optimizarlo. El proyecto se mide para gestionarlo, controlarlo, adaptar el flujo del trabajo y las actividades, y para realizar estimaciones para futuros proyectos. El producto se mide para aumentar su calidad o comprobar que se ajusta a las especificaciones contractuales.

Un ingeniero de software recopila medidas desarrollando métricas para obtener indicadores.

Los indicadores son métricas o un conjunto de ellas que proporcionan una visión mas profunda del proceso de software, del proyecto y del producto en si.(Pressman 2005)

Se obtienen indicadores del proceso y del producto a través de métricas recopiladas.

Indicadores del proceso

- Una visión profunda de la eficacia del proceso ejemplo:
 - Paradigma
 - Tareas de ingeniería de software
 - Productos de trabajo e hitos
- Evaluación por parte de los gestores del proceso para determinar qué funciona bien y qué no.

Indicadores del proyecto

Permite al gestor del proyecto:

- Evaluar el estado del proyecto en desarrollo
- Seguir las pistas de riesgos potenciales
- Detectar problemas
- Ajustar el flujo y las tareas del trabajo
- Evaluar la habilidad del equipo de desarrollo del proyecto en controlar la calidad de los productos.

La medición es algo común en el mundo de la ingeniería no siendo así en el caso de la ingeniería de software, puesto que a pesar de no ser un tema novedoso no se ha tratado en los últimos tiempos correctamente.

Las organizaciones no habían comprendido la necesidad de realizar medidas para obtener métricas hasta la aparición del mercado competitivo y demás, aun así, existen organizaciones que la utilizan pero no correctamente. Esto se debe a que existen problemas a la hora de ponerse de acuerdo en aspectos como qué se debe medir y como hacerlo.

Las medidas se pueden clasificar en:

Medidas Directas: en el proceso de ingeniería se encuentran el coste, y el esfuerzo aplicado, las líneas de código producidas, velocidad de ejecución, el tamaño de memoria y los defectos observados en un determinado periodo de tiempo.

Medidas Indirectas: se encuentra la funcionalidad, calidad, complejidad, eficiencia, fiabilidad, facilidad de mantenimiento.

El primer paso de la medición es identificar los atributos o entidades a medir. Estos pueden ser de tres tipos:

Procesos: atributos de actividades relacionadas con el software.

Productos: componentes, entregas o documentos resultantes de una actividad de proceso.

Recursos: entidades requeridas por una actividad de proceso

Dentro de cada clase anterior se puede distinguir:

Atributos internos: son aquellos que pueden ser medidos examinando el proceso, producto o recurso mismo

Atributos externos: se miden con respecto a como el proceso, producto o recurso se relaciona con su entorno. **(Ver tabla 1).**

Las métricas del proceso se obtienen a partir de las métricas del proyecto. Por ejemplo:

Métricas privadas

- Índices de defectos

- Errores de desarrollo

Métricas para el equipo

- Índices de defectos
- Errores de desarrollo
- Líneas de código (LDC)
- Puntos de función (PF)

Las métricas no se pueden usar para comparar personas o equipos. Para la mejora del proceso las métricas deben aplicarse desde el nivel individual. Las métricas de un producto que son privadas para un individuo del equipo de desarrollo, se combinan para desarrollar métricas del proyecto que sean públicas para dicho equipo y estas a su vez se consolidan para crear métricas del proceso.

Partiendo de esta premisa se pueden destacar estándares que fundamentan el uso de las métricas.

1.6.2 Estándares internacionales

Existen métricas relacionadas con modelos y estándares internacionales:

PSM (Prácticas de Software y Sistemas)

El propósito de PSM es brindar a los administradores de proyectos la información cuantitativa necesaria para tomar decisiones que tengan un impacto en los costos, cronograma y objetivos técnicos de desempeño del proyecto.(Durán 2007)

Proceso PSM:

Planificación de la Medición.

Se definen las métricas necesarias para satisfacer las necesidades de información

Realización de la Medición.

Se recogen los datos de las mediciones, se realiza el análisis y se presentan los resultados

Evaluación de la Medición

Tanto el proceso de medición como las propias métricas definidas deben evaluarse y mejorarse periódicamente según sea necesario.

Establecimiento y mantenimiento del Compromiso.

Se establecen los recursos, formación y herramientas necesarias para implementar un programa de medición de forma efectiva

Principios de medición de PSM

1. Emplear acciones y objetivos para dirigir los requisitos de medición.
2. Definir y recopilar medidas basadas en los procesos técnicos y de administración.
3. Recopilar y analizar datos detallados para identificar y aislar los problemas.
4. Implementar una capacidad de análisis independiente.
5. Usar un proceso de análisis sistemático para supervisar las medidas utilizadas en la toma de decisiones.
6. Interpretar los resultados de medición en el contexto de la información de otros proyectos.
7. Integrar la medición en el proceso de gestión del proyecto durante todo su ciclo de vida.
8. Emplear el proceso de medición como una base para una comunicación objetiva.
9. Enfocarse inicialmente en el análisis al nivel del proyecto.(Durán 2007)

Normas ISO 9000

'**ISO 9000** es una organización (conjunto de normas de calidad) establecidas por la Organización Internacional para la Estandarización (ISO) que se pueden aplicar en cualquier tipo de organización (empresa de producción, empresa de servicios, administración pública...).

Su implantación en estas organizaciones, aunque supone un duro trabajo, ofrece una gran cantidad de ventajas para sus empresas. Los principales beneficios son:

- Reducción de rechazos e incidencias en la producción o prestación del servicio.

- Aumento de la productividad
- Mayor compromiso con los requisitos del cliente.
- Mejora continua.

El estándar más reconocido mundialmente es el ISO 9000, un sistema de gestión básica de calidad que se pueda utilizar en industrias de cualquier tamaño, en cualquier parte del mundo. La certificación al estándar ISO 9001 (o a cualquier otro sistema de gestión de calidad) proporciona la prueba objetiva de que un negocio ha implementado un sistema de gestión de calidad, el cual satisface todos los requisitos del estándar aplicable.

Certificación en ISO 9000

- Los estándares de calidad y procedimientos deben ser documentados en un manual de calidad organizacional.
- Una entidad externa debe certificar que el manual de calidad organizacional cumple con las normas de ISO 9000.
- Algunos clientes requieren de proveedores certificados en ISO 9000.

El modelo CMM (Capacity Maturity Model) describe un marco de referencia para el desarrollo y mantenimiento de software en las organizaciones así como la contratación de software. CMM constituye un modelo en el que el mejoramiento de los procesos de software es implementado de manera incremental.

El modelo CMM puede ser adaptado y moldeada a una particular organización pero requiere que se impongan los procedimientos a realizar.

El modelo CMM se basa en el concepto de madurez de un proceso. La madurez de un proceso depende de cuan bien definido y completo estén los procesos y si ellos pueden ser usados para manejar un proyecto de software. La madurez también depende de cuan bien los procesos permitan a los gerentes de software definir las

salidas del software y ajustar el plan, de manera que el proyecto sea controlado.

(Cárdenas 2002)

Niveles

El modelo CMM organiza los pasos para evolucionar los procesos de software en cinco niveles.

Son estos:

Inicial:

Las organizaciones se caracterizan por tener procesos caóticos, sus procesos son impredecibles y poco controlados.

Repetible:

La gestión de la ingeniería de software es utilizada para establecer procesos básicos de gestión de proyectos, para controlar costo, planificación y calidad del producto.

Definido:

Existen procesos de software estándar y documentados para la gestión y la ingeniería de software. Todos los proyectos usan un proceso estándar de la organización para desarrollar y mantener software.

Gestionado:

Los productos y los procesos son medidos y manejados cuantitativamente.

Optimizado:

Se caracteriza por la mejoría cuantificable y continua de los procesos. Lo anterior es logrado por la retroalimentación de los procesos y por el pilotaje de nuevas ideas y tecnologías.

Cada nivel (excepto el primero) es descompuesto en áreas de procesos claves (18 en total) y cada área de procesos claves contiene prácticas claves (343 en total) las que se agrupan de acuerdo a facilidades comunes. Un área de proceso clave contiene objetivos que deben ser alcanzados para mejorar el proceso de software. Las prácticas claves contribuyen a satisfacer las áreas de procesos claves, ellas describen que se debe hacer pero no cómo debe hacerse.(Cárdenas 2002)

Proceso de software personal (PSP): es un conjunto estructurado de descripciones de proceso, mediciones y métodos que permite el rendimiento personal de los ingenieros. Muestra como

definir procesos y como medir la calidad y productividad siguiendo el principio de que un método que sea efectivo para un individuo pueda que no lo sea para otro. Proporciona además guiones, formas y estándares para que los ingenieros planifiquen su trabajo.(Hernandez 2006)

PSP plantea que un individuo no tiene la misma productividad en todos los trabajos o tareas que realiza, por lo establece par conocer cuan productivo puedes ser realizando determinada actividad debes conocer el tamaño del producto y el tiempo que demoras en desarrollar la actividad en cuestión, para ello establece estimaciones de tamaño y tiempo respectivamente a partir de datos históricos almacenados.

Otro aspecto importante que destaca el PSP es de calidad la cual define en función de la introducción y eliminación de errores planteando la medida de defectos /hora con la que se obtiene una estimación del rendimiento para mejorar la efectividad

Las diferentes etapas del PSP son:

Proceso base

El programador conoce las necesidades del cliente para tener una idea clara de lo que programara.

Ambiente de mejora

En caso de tener un proyecto ya hecho esta es la etapa de depuración, se utiliza para mejorar el proyecto anterior.

Estimación de tamaño de proyecto, pruebas

En caso de no ser así se estima el tamaño que tendrá el software. Tomando en cuenta el tiempo que se le dedicara y las especificaciones del cliente.

Plantación de tareas y horario.

Ya teniendo el tamaño del software se puede hacer una estimación del tiempo que se tomara para hacer la programación así como las diferentes tareas, de debe realizar un horario para tomar en cuenta cuanto tiempo debe tomar cada tarea y terminar a tiempo.

Control de calidad personal.

Por medio de pruebas se califica la eficiencia del sistema así como los errores que puedan llegar a tener, se tiene que confirmar que la calidad del software es la deseada.

Proceso Cíclico

Teniendo en cuenta la calidad con la que el programa será liberado se decide si el software se liberara o tiene que volver a entrar en el proceso como proceso base.

El problema mas grande que se presenta con el PSP es que para los programadores es difícil tratar de visualizar todas las tareas y etapas del desarrollo.

Es por eso que en muchas ocasiones se necesita de otros programadores, pero al estar mas de uno no se puede utilizar el proceso de software personal, este problema se soluciona formando un equipo de trabajo y llevando a cabo el desarrollo por medio de TSP.

Proceso de Software en Equipo (TSP):

Este proceso se utiliza cuando en el desarrollo existe un equipo de programadores, no es necesario que sean muchos, basta con que sea mas de un programador, al igual que el PSP la meta del TSP es la de obtener un software de alta calidad en el menor tiempo posible, se utilizan todas las tareas que el PSP indica, pero además se utilizan métodos de organización de personal debido a que al haber un equipo deben realizarse roles de trabajo. Se tiene que tener un líder del equipo que este en comunicación con todos y que tenga control sobre la organización del proyecto. (Hernandez 2006)

Este proceso requiere que cada uno de los integrantes del equipo tenga bien definida su función y también este suficientemente capacitado para realizarla con éxito.

Debido a que la mayoría de los proyectos no se deja a un solo individuo sino que se deja a un equipo de trabajo el TSP es un proceso de desarrollo muy utilizado por las diferentes instituciones y compañías de todo el mundo.

Sin embargo el utilizar TSP tiene requisitos como los que son:

- Contar con personal suficientemente capacitado.
- Que el personal tenga la capacidad de trabajar en equipo.
- Debe haber convivencia en el equipo para facilitar la comunicación.

Tienen que realizarse roles de trabajo y cada integrante debe estar en el área en la que mejor se desarrolla para que sea mas eficiente.

1.6.3 Métricas de productividad

Actualmente las empresas siguen sin poder establecer indicadores que midan la productividad de los proyectos.

El problema comienza por definir qué es realmente la productividad.

Desfavorablemente, el concepto, aunque está ampliamente difundido en nuestro medio, pocos tienen claro a qué se refiere exactamente, y menos cómo se mide.

En la psicología del trabajo productividad equivale a efectividad. Según el diccionario, “productividad es la capacidad de producción, siendo éste un término análogo a creatividad”. Otras fuentes definen el concepto como “el porcentaje que mide la capacidad o grado de producción y que se determina dividiendo la cantidad total producida por la unidad de trabajo, superficie, etc.” Una definición simplista dice que “productividad es producir más con lo mismo, o lo mismo con menos”, o sea, optimizar los recursos, aunque en nuestro medio la fórmula más exacta es “hacer más con menos”. Finalmente, algo es productivo cuando es útil o genera un resultado favorable.

La palabra “productividad” se deriva de “producto”. Un producto es un resultado. Si queremos medir la productividad necesitamos enfocarnos al resultado que se desea obtener a partir de un objetivo específico. De esta manera, el resultado es la meta que se desea alcanzar, a partir de un objetivo.

Para los productos de software, productividad se define como la cantidad de esfuerzo requerido para lograr un cierto grado de funcionalidad.

Productividad en términos de empleados es sinónimo de rendimiento. En un enfoque sistemático decimos que algo o alguien es productivo con una cantidad de recursos en un periodo de tiempo dado se obtiene el máximo de productos.

La productividad es uno de los indicadores que muestra la medida más cercana de la eficiencia de la producción de la organización. Utilizar mediciones para evaluar su estado es el primer paso para su incremento. Sin embargo para que el indicador de productividad obtenido nos aporte información valiosa debemos compararlo con la productividad alcanzada en períodos anteriores, o con la planificada en ese periodo, o con la de otras empresas y a partir de aquí valorar los resultados en función de obtener mejoras en el proceso.

Algunos pasos básicos para medir la productividad sobre una base sólida y confiable son:

1. Definir objetivos generales a partir de los lineamientos estratégicos de la organización (misión, visión, valores, factores clave).
2. Alinear dichos objetivos con las áreas básicas de la organización.
3. Desplegar los objetivos en cada nivel de la estructura.
4. Establecer metas individuales.
5. Medir periódicamente los resultados.
6. Retroalimentar al personal

Las métricas de productividad son las referidas al rendimiento del proceso de desarrollo en función del esfuerzo aplicado

Puede estar definida también por:

- Volumen de información procesado por el sistema por unidad de tiempo.
- Ratio (peticiones por unidad de tiempo) al cual las peticiones pueden ser servidas por el sistema.

La productividad en una empresa se puede medir desde dos ámbitos, económico o de producción. Este último se basa en medir a productividad en función de la producción que se obtiene, para lo cual se mide la capacidad del proceso o la productividad de la mano de obra si la producción es manual o depende de un operador y tomando como base que la producción de la

facultad esta dada por varios proyectos productivos, pudiera plantearse medir la productividad de acuerdo a la capacidad de producción del proceso.

- ✓ Las métricas de productividad se pueden calcular a partir de medidas básicas como son líneas de código (LDC) y punto de función (PF).
- ✓ Las medidas de productividad inconsistentes entre proyectos pueden ser un indicador de que un proceso estándar no se está siguiendo.
- ✓ Los cambios bruscos en las medidas de productividad indican que no se está siguiendo un proceso estándar.

Los datos de LDC y PF se utilizan de dos formas diferentes durante a la estimación de un proyecto software:

- como variable de estimación para dimensionar cada elemento del software
- como métricas de línea base recopilada de proyectos anteriores y utilizados junto con otras variables de estimación para estimar costes y esfuerzos.

Línea base: datos recogidos de proyectos de software desarrollados anteriormente.

Sirve como base para la estimación

1.7 Conclusiones

Este capítulo de manera general mostró la existencia de las mediciones en todo el proceso de producción de software, y cuando se dicen mediciones, se habla de métricas, las métricas constituyen un elemento fundamental para llevar a cabo dicho proceso organizado y controlado. Fundamentalmente el control de la productividad influye en toma de decisiones para la mejora del proceso.



Capítulo 2: Solución Propuesta

2.1 Introducción

En este capítulo se hará un análisis de las métricas de productividad existentes, se seleccionará la que se ajusta al proceso productivo de la Facultad 4 de la UCI y se definirá una guía de aplicación de la misma en el proceso de desarrollo en cuestión.

2.2 Análisis de las métricas de productividad que existen

En la actualidad, cuando se habla de implantar métricas de productividad en las empresas donde se produce software se centran en conocer el tamaño de lo que se produce, en este caso se refiere al software que se desarrolla. Para calcular la productividad, se ha estandarizado el uso de dos métricas bases:

- Orientadas al tamaño
- Orientadas a puntos de función

Las métricas de productividad orientadas al tamaño y a los puntos de función constituyen las bases para calcular la productividad.

Con los resultados obtenidos de estas métricas, las organizaciones pueden estimar la cantidad de personas y tiempo, necesarios para desarrollar el producto (en este caso el software). Obteniendo la cantidad de personas y el tiempo se calcula fácilmente el esfuerzo, que constituye la medida vital para calcular la productividad. Teniendo en cuenta la explicación anterior se puede afirmar entonces que las métricas de productividad pueden ser:

- **Métrica de productividad orientada al tamaño**
- **Métrica de productividad orientada a los puntos de función.**

A partir de estas las organizaciones, en virtud de aplicar una métrica de productividad, tanto en sus proyectos o a su proceso de desarrollo, deben primeramente definir que métrica base a usar (orientada al tamaño, orientada a los puntos de función)

➤ Métricas de productividad orientadas al tamaño

El tamaño, en este ámbito es contemplado en líneas de código, calculando las líneas de código (LDC), se obtiene la métrica de productividad. Entonces cabe preguntar:

¿Qué procedimiento se sigue para la obtención de las líneas de código?

Para el cálculo de las líneas de código se tienen en cuenta diferentes consideraciones:

- Debe contabilizarse cada línea nueva o codificada.
- Las líneas para la instrumentación de código para las pruebas no deben incluirse en el tamaño total, salvo que tengan un carácter definitivo.
- Las líneas de código de programas de prueba tan solo se contabilizan si se desarrollan con el nivel de calidad requerido al entregar el proyecto.
 - Se contabilizan las líneas correspondientes a las llamadas al sistema operativo.
 - No se consideran los comentarios.
 - No se contabiliza el pseudo código.
 - Cada ocurrencia de *macro* o include se considera como una línea.
 - El código generado por *macros* o includes solo se considera una vez.

Una vez obtenidas las líneas de código, se relacionan con otras medidas de las cuales se obtienen métricas que permiten medir la calidad, productividad, costo entre otros por ejemplo:

Productividad = LDC / persona mes

Esta métrica mide la productividad en función de las líneas de código realizadas por personas en un mes.

Calidad = Errores/ LDC

El propósito de esta métrica es controlar la calidad del conteo de líneas de código.

Coste = Dólares/ LDC

Permite estimar el costo d el producto según la cantidad de líneas de código contadas.

Documentación = Paginas de documentación / LOC

Obtener el tamaño del software en líneas de código, es un método utilizado por muchas organizaciones a nivel mundial, sin embargo resulta un poco engorroso, pues contar cada línea de código en las organizaciones que produzcan software, tiende a ser mal interpretada como pérdida de tiempo, o los programadores se sentirían amenazados con el control de su trabajo, calculando que cantidad de líneas de código realizan en determinado periodo de tiempo, sumándole a esto mas desventajas sobre la líneas de código: Dependen del lenguaje de programación que se utilice, su aplicación no permite realizar estimaciones.(Giraldo 2004).

➤ Métricas de productividad orientadas a puntos de función

La técnica de medición del tamaño en punto-función consiste en asignar una cantidad de "puntos" a una aplicación informática según la complejidad de los datos que maneja y de los procesos que realiza sobre ellos. Siempre tratando de considerarlo desde el punto de vista del usuario.(Rodríguez. 1999)

Las métricas orientadas a los puntos de función se centran en la funcionalidad o utilidad del programa, por eso muchos la definen como métrica funcional. Los Puntos de Función proporcionan una medida objetiva, cuantitativa y auditable del tamaño de las aplicaciones, desde el punto de vista de los requisitos especificados por el usuario final de la aplicación. Las métricas de productividad orientadas a puntos de función, son medidas indirectas del software y del proceso por el cual se desarrolla a diferencia de las orientadas al tamaño (LOC), que son medidas directas. También son un medio de entendimiento entre lo que el usuario quiere y lo que

al final se le suministra. Su valoración se deriva a partir de los requisitos funcionales que la aplicación debe satisfacer, modelos de datos, definición de pantallas e interfaces gráficos y diagramas de análisis. Los Puntos Función constituyen una técnica de medida del software, simple de obtener pero muy potente en sus resultados. Esta potencia radica en que del valor de la medida en Puntos Función se derivan un conjunto de métricas esenciales para la gestión de la productividad, la calidad y el coste del software. Con estas medidas, registradas en distintas fases del ciclo de vida, se puede llevar a cabo un análisis exhaustivo de su evolución y, por tanto, del control de la productividad, la calidad y los costes asociados, a lo largo del tiempo. De esta forma, y almacenando en un registro histórico de datos el valor en Puntos Función de cada uno de los proyectos realizados, podremos disponer de una sólida base para futuras estimaciones del coste y duración de los proyectos, información altamente valiosa para la dirección de las organizaciones. (Rodríguez. 1999)

Cálculo del punto de función

1. Hay que completar la siguiente tabla de valores del dominio de la información

Parámetro	C uenta	Factor de ponderación			Su btotal
		S imple	M edio	Co mplejo	
Número de entradas de usuario		3	4	6	
Número de salidas de usuario		4	5	7	
Número de peticiones de usuario		3	4	6	
Número de archivos		7	10	15	
Número de interfaces externas		5	7	10	
Total					

Donde:

- Entradas de usuario. Son entradas que proporcionan diferentes datos a la aplicación. No confundirlos con las peticiones de usuario.
- Salidas de usuario. Son reportes, pantallas o mensajes de error que proporcionan información. Los elementos de un reporte, no se cuentan de forma separada.

○Peticiones de usuario. Es una entrada interactiva que produce la generación de alguna respuesta del software en forma de salida interactiva.

○Archivos. Son los archivos que pueden ser parte de una base de datos o independientes.

○Interfaces externas. Son los archivos que se usan para transmitir información a otro sistema.

- Decidir la complejidad de todo el sistema.
 - Contar cada medida por separado.
 - Asociar, de alguna manera, un valor de complejidad a cada medida.

Tipos de archivos referenciados	Tipos de datos elementales		
	1-5	6-19	20+
0-1	bajo	bajo	medio
2-3	bajo	medio	alto
4+	medio	alto	alto

○Para cada medida, multiplicar su cuenta por el factor de complejidad elegido y escribirlo en la columna de la extrema derecha.

○Sumar la columna de la extrema derecha tabla anterior y obtener un total T que indica el valor del dominio de la información.

- Evaluar cada factor en una escala de 0 a 5

0 1 2 3 4 5

Sin influencia Incidental Moderado Medio Significativo Esencial

Preguntas:

¿Requiere el sistema copias de seguridad y de recuperación fiables?

¿Requiere comunicación de datos?

¿Existen funciones de procesamiento distribuido?

¿Es crítico el rendimiento?

¿Se ejecutará el sistema en un entorno operativo existente y fuertemente utilizado?

¿Requiere entrada de datos interactiva?

¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?

¿Se actualizan los archivos maestros de forma interactiva?

¿Son complejos las entradas, las salidas, los archivos o las peticiones?

¿Es complejo el procesamiento interno?

¿Se ha diseñado el código para ser reutilizable?

¿Están incluidas en el diseño la conversión y la instalación?

¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?

¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?

Una vez obtenida la información requerida para calcular los puntos de función, se obtiene el valor de los puntos de función a través de la siguiente fórmula:

$$\text{PF} = \text{CUENTA_TOTAL} * [0.65 + 0.01 * \text{SUM}(fi)]$$

Donde CUENTA _ TOTAL es la suma de todas las entradas de PF obtenidas de la tabla anterior. Fi donde i puede ser de uno hasta 14 los valores de ajuste de complejidad basados en las respuestas a las cuestiones señaladas de la tabla anterior.

Obtenido el valor de los puntos de función, se pueden aplicar métricas que aporten información sobre aspectos como, costo, calidad, productividad, siendo este ultimo el de interés para la investigación.

$$\text{Productividad} = \text{PF} / \text{persona mes}$$

Se obtiene como resultado la productividad del equipo de desarrollo en función del esfuerzo (persona-mes) = cantidad de personas que realizan las actividades en un mes.

$$\text{Calidad} = \text{Errores} / \text{PF}$$

Con la aplicación de esta métrica se obtiene información que permite el control de que el cálculo de puntos de función se hizo correctamente.

$$\text{Coste} = \text{Dólares} / \text{PF}$$

Esta métrica permite una estimación del coste del producto que se produce. Con el valor de los puntos de función, se estima el posible coste que puede demandar el producto.

$$\text{Documentación} = \text{Páginas de documentación} / \text{PF}$$

Es una manera de controlar la documentación del software como tal, lo que permite de paso estimar el avance del proyecto.

2.3 Estudio del proceso de producción.

El análisis del proceso productivo de la facultad 4, es necesario para obtener información sobre el funcionamiento de la producción de dicha Facultad, aspectos particulares de dicho proceso que permitan a la autora de la investigación definir una métrica o sistema de métrica que contribuyan al control de la productividad.

2.3.1 Elementos del proceso

En la universidad las Facultades diferencian el marco productivo de sus proyectos según el perfil de producción que persiguen, en el caso de la Facultad 4, cuyo proceso productivo se analiza, la representa el perfil: Gestión empresarial. Esto implica que los proyectos productivos son proyectos de gestión, por lo que se centran en automatizar todos los procesos de determinada entidad, dígase Personal, Estadísticas, entre otras.

Esta información impone el análisis siguiente: en el proceso de definición de la solución propuesta, dígase la métrica que controle la productividad del proceso productivo se debe tener en cuenta que debe ser estándar, para que los proyectos en su totalidad sigan un procedimiento común, de lo contrario habría que definir una métrica que controle la productividad específica para cada proyecto.

2.3.2 Proceso de desarrollo de los proyectos de la facultad 4.

Según la entrevista realizada al vice decano de producción de la Facultad, los proyectos productivos de la facultad utilizan en su proceso de desarrollo la metodología RUP, y para el desarrollo se han estandarizados los lenguajes de programación Java y PHP. Como se aprecia no todos los proyectos utilizan un mismo lenguaje de programación, lo que obliga a realizar un análisis al definir la métrica base a usar, o sea, cual sería la masa factible para la facultad: líneas de código (LDC) o puntos de función (PF) evidentemente la que no dependa de la tecnología.

2.3.3 Control de la productividad en el proceso productivo de la Facultad.

Se realizaron encuestas a los líderes de 6 proyectos productivos de la facultad 4, entre los que se encuentra los más importantes de la Facultad como son: SIGEP, mas conocido por Prisiones, ERP MINFAR, Aduana y ERP cubano, este último no ha obtenido resultados de su desarrollo de su producción todavía, por lo que no se incluye dentro de los 6 encuestados. Estos proyectos fueron escogidos precisamente por ser los proyectos de la facultad de los cuales se han obtenido resultados de su desarrollo. Por tanto el grupo de los 6 proyectos cuyos lideres fueron encuestados esta conformado por: SIGEP (Prisiones), Aduana, Banco, ERP MINFAR. Digitalización, esta cifra de proyectos constituye el 50 % de los proyectos de la facultad los cuales se catalogan los más productivos en estos momentos, los demás proyectos están en pleno desarrollo. El objetivo principal que perseguían las encuestas (**ver anexos**), era conocer qué estrategia se utilizaba en estos proyectos para el control de la productividad. La información obtenida en las encuestas (**tabla 1**) muestra controla la productividad en 4 de los proyectos referenciados solo que el procedimiento por el que rigen el control esta basado en estimaciones. Esto da muestra de que no es un procedimiento común puesto que los proyectos están conformados por módulos, y dentro del proyecto la producción se evidencia según las actividades que realizan los módulos. Según este conocimiento se puede afirmar que los módulos son la base de la producción en los proyectos, por tanto la productividad del proceso se controla a partir de la productividad de los proyectos. Sin embargo no todos los proyectos conciben los módulos de igual forma, unos proyectos definen los módulos como mini proyectos dentro del proyecto a los cuales se les realiza todo el proceso de ingeniería de software, otros van terminando los módulos según su correspondencia con las actividades criticas (las actividades que imponen ser realizadas en esa etapa del proyecto) que realiza el proyecto. Esto demuestra que el procedimiento que utilizan los proyectos basados en estimaciones no es común para todos, puesto que cada proyecto las realiza según sus particularidades.

2.3.4 Métricas en el proceso de producción de la Facultad.

Sobre la base de que el uso de las métricas es efectivo para el control de factores decisivos de una organizacion productora de software como calidad, costo, productividad, otro de los objetivos de la encuesta realizada era conocer el nivel de experiencia de aplicación de métricas en los proyectos de la facultad, para tener una base para la futura métrica que se defina para el control de la productividad del proceso en cuestión los resultados que se arrojaron (**tabla 2**) afirman que no se han aplicado métricas en los proyectos de la facultad ni en el proceso en general. Conociendo esto se decide que la propuesta de solución que se defina debe ser simple de aplicar, teniendo en cuenta que la no aplicación de métricas en el proceso equivale a decir que no existe cultura sobre el uso de las métricas, que a su vez puede ser un trastorno en la aplicación de una métrica, este proceso requiere de un cambio cultural hacia el control de las actividades y puede ser malinterpretada como pérdida de tiempo, o tal vez mucho control de los desarrolladores lo que provocaría la irritación de los mismos, por tanto se decide que la métrica debe ser simple de aplicar en los proyectos.

2.4 Definición de la métrica

La literatura de las métricas plantea que las métricas son adaptables al proceso para el cual son aplicadas, no existen una métrica perfecta, ni restricciones sobre las mismas. Solo hay que aplicar las mayor cantidad de métricas posible para obtener el resultado lo más exacto posible. Para la aplicación de métricas en las organizaciones, dicha organización debe reconocer la necesidad del uso de métricas, seguido a esto debe definir cuáles son los objetivos que persigue con la aplicación de métricas en la organización, según los objetivos se determina el tipo de métrica a usar, puede ser de calidad, productividad y costos. En este sentido, según el estudio realizado al proceso productivo de la Facultad y teniendo en cuenta que uno de los problemas que acechan la producción de la Facultad es la no existencia de un procedimiento estándar que controle la productividad, permite definir una métrica o sistema de métricas que de solución al problema planteado.

Para la definición de la métrica es necesario saber hasta que punto debe ajustarse al proceso, para ello se exponen a continuación requisitos que debe cumplir la métrica:

- Aplicación común para todos los proyectos: sería más efectivo controlar la productividad del proceso de producción de la facultad si todos los proyectos realizan el mismo procedimiento, por tanto la productividad de los proyectos debe ser controlada de manera equivalente en todos los proyectos.
- Fácil aplicación: teniendo en cuenta que aplicar una métrica en una organización sin experiencia en el tema (**epígrafe 2.2.5**), puede traer perjuicios a los proyectos se decide que la métrica debe ser de cálculos simples o lo que es lo mismo de fácil aplicación para no influir en el tiempo de desarrollo del proyecto.
- Ajustada a la jerarquía de la Facultad: según lo planteado anteriormente (**epígrafe 2.2.4**) para el control de la productividad del proceso se requiere del control de la productividad de los proyectos y la productividad de los proyectos se consolida según las actividades que se realiza en los módulos. Por tanto la métrica propuesta debe estar concebida basándose en esta premisa.

Una vez definidos estos puntos, se procede a definir la estructura del sistema de métricas que se propone para dar solución al problema de la investigación .sin embargo para definir la estructura de la métrica se debe determinar que métrica base usar para calcular la productividad. Planteado esto se propone utilizar como medida base los puntos de función por las siguientes razones:

- Es una medida que puede utilizar cualquier proyecto para calcular la productividad.
- No depende del lenguaje de programación, o se si el proyecto utiliza el lenguaje de programación JAVA, los puntos de función se calculan de igual forma, si otros utilizan PHP, se obtienen los puntos de función aplicando el mismo procedimiento. No siendo igual si se utilizan las líneas de código, pues en algunos lenguajes tiende a reducirse el tamaño por lo que no sería estándar para todos los proyectos.
- Es completamente centrada en la funcionalidad del producto que se desarrolla, lo que anota un punto a favor de la Facultad, por ejemplo teniendo una visión de la funcionalidad

de la aplicación se conoce hasta que punto se cumplen los requisitos de los usuarios y podría utilizarse como evaluación de la calidad del producto en función de satisfacer al cliente.

- Permite realizar estimaciones: universalmente se plantea que las líneas de código suelen ser detallistas, por lo que no se pueden realizar estimaciones, al contrario de los puntos de función, con el resultado del conteo de los puntos de función se pueden realizar estimaciones para el costo, para la cantidad de personas que se requieren para realizar las actividades.

2.4.1 Estructura del sistema de métricas propuesta.

Se parte de la convicción de que se debe partir de métricas básicas a través de las cuales se obtienen las métricas derivadas se define entonces:

Métricas bases: que constituyen las métricas necesarias para obtener otras métricas, por ejemplo:

Esfuerzo del módulo (EM)

Productividad del módulo (PM)

Esfuerzo del proyecto (EP)

Puntos de Función (PF)

Otro grupo definido es el grupo de las **métricas derivadas**, que se obtienen a partir de las métricas bases entre las que se encuentran:

Productividad del proyecto (PP)

Productividad del proceso (PG).

Después de relacionar las distintas métricas definidas para conformar la propuesta de solución:

Estructura de las métricas

Se había definido anteriormente que la base del funcionamiento de la producción de software en la facultad lo constituían los proyectos productivos que la integraban, y a su vez los proyectos se complementaban por las actividades de desarrollo concebidas en los módulos de dichos proyectos. Partiendo de aquí, para obtener un valor de productividad del proceso del proceso de

desarrollo de software de la Facultad en general, se debe conocer la productividad de los proyectos, pero la funcionalidad de los proyectos se centra en las actividades que realizan los módulos, y conociendo que no todos los proyectos conciben los módulos de igual manera, o sea los módulos realizan actividades de diferentes tipos según el proyecto al cual pertenezcan, esto implica que lo único que tienen en común es que realizan actividades, lo más factible a calcular entonces de igual forma en todos los proyectos sería el esfuerzo de cada módulo que trabajan en ese periodo de desarrollo del proyecto. Por estas razones se define entonces la siguiente métrica:

1.1 Métrica: Esfuerzo del módulo

Permite conocer el esfuerzo de las personas asignadas para realizar cierta tarea, constituye una base del conocimiento para valorar el proceso de asignación de tareas en el módulo

$$E_m = P / t$$

Donde:

P: cantidad de personas asignadas a realizar la tarea.

t: tiempo planificado para el cumplimiento de la tarea.

Propósito: calcular el esfuerzo de los módulos para poder obtener el esfuerzo medio del proyecto.

E_m (esfuerzo del módulo se obtiene en personas- semanas)

Siguiendo la idea planteada anteriormente, el resultado debe ser la productividad de los proyectos, por tanto con el esfuerzo de todos los módulos y teniendo en cuenta que lo que se evalúa es la productividad en esa semana, se puede obtener entonces el esfuerzo del proyecto, se define entonces para obtener el esfuerzo del proyecto la métrica siguiente:

1.2 Métrica: Esfuerzo de proyecto

Permite evaluar el esfuerzo del proyecto en determinado periodo de tiempo.

$$E_p = \sum_{C_m} E_m (i)$$

C_m

Donde:

Em: esfuerzo del módulo (**métrica 1.1**)

M: módulo del proyecto

i: desde 1 hasta Cm

Cm: cantidad de módulos del proyecto

Propósito: calcular el esfuerzo medio del proyecto en ese periodo de tiempo.

Ep (esfuerzo del proyecto) se obtiene en persona- semana

Esta claro que para obtener la productividad de la producción se debe conocer el tamaño o complejidad del producto a producir, y se había definido que la medida básica a utilizar en la métrica de productividad que se defina párale control de la misma será puntos de función, con los puntos de función y el esfuerzo del proyecto quedan creadas las bases para obtener la productividad del proyecto se planeta entonces para obtener el valor de la productividad del proyecto y factible para su aplicación en todos los proyectos la siguiente métrica:

1.3 Métrica: Productividad del proyecto

PP = PF / EP (p/día, mes, año)

Donde:

PF: puntos de función del proyecto calculados según el procedimiento descrito en anteriores epígrafes.

EP: esfuerzo del proyecto (**métrica 1.2**)

Propósito: evaluar los puntos de función según el esfuerzo del proyecto para obtener la productividad del proyecto.

PP (productividad del proyecto) se obtiene en PF/ persona-semana (puntos de función por persona - semana)

El problema fundamental a resolver es aplicar un método que permita a los directivos de la Facultad controlar la productividad para conocer el estado del proceso de desarrollo de software y tomar la información obtenida tras la aplicación de dicho método para definir mejoras en el proceso. Partiendo de esta idea redefine de manera simple y fácil de obtener la siguiente métrica para obtener la productividad de todo el proceso de producción de la Facultad en un rango de

tiempo determinado que permita el control consecutivo de la misma. Como existen varios proyectos en desarrollo al mismo tiempo, pues se decide utilizar la siguiente métrica para calcular el valor de la productividad de la facultad.

1.4. Métrica: Productividad general (PG)

$$PG = \frac{\sum PP (i) / \text{esfuerzo (p/día, mes, año)}}{Cp}$$

Donde:

PP: productividad del proyecto (**métrica 1.3**)

P: proyecto

i: desde 1 hasta la cantidad de proyecto

Cp: cantidad de proyecto

Propósito: calcular la productividad media del proceso de producción de la facultad en general.

PG (productividad general) se obtiene en PF/ persona-semana (puntos de función por persona - semana)

El sistema de métricas se plantea como propuesta de solución al problema de la investigación. Se concibe desde la perspectiva de que siendo los proyectos de la Facultad proyectos de gestión, indudablemente deben dividir el proyecto en módulos para organizar el proceso, pues el proceso de gestión en una entidad se refiere a gestionar todos los procesos internos de esa entidad, por tanto de con la propuesta, sea cual sea la convicción de los líderes de proyectos, con respecto a la funcionalidad de los módulos integrados en dichos proyectos, respondiendo a cada módulo a cada proceso interno de la entidad o empresa, o desarrollando diferentes módulos por cada proceso el sistema de métricas puede aplicarse a cualquiera de ellos.

2.4.2 Guía de aplicación de la propuesta.

La aplicación de métricas de productividad en una organización puede realizarse persiguiendo varios objetivos:

Productividad en un periodo de tiempo

Productividad de los proyectos para demostrar que se sigue un procedimiento estándar en el proceso.

Productividad del proceso para compararlo con la productividad general.

Sin embargo para que el resultado obtenido brinde esta información, debe ser comparado con una base histórica de productividades obtenidas en periodos anteriores. El valor de productividad que se obtiene al evaluar los valores en una métrica, no aportan información convincente, solo da pie a que se realicen las comparaciones para obtener la información requerida para tomar decisiones sobre el proceso, se puede obtener información por ejemplo: si existe mucha diferencia entre las productividades de los proyectos es un indicador de que no se sigue un procedimiento estándar, si se compara la productividad del proceso general con la productividad en etapas anteriores, se puede concluir el estado de la productividad, si ha avanzado, o no. Después de analizar estos elementos cabría preguntar:

¿Cómo se lleva a cabo el proceso de aplicación de la métrica propuesta?

El primer paso para llevar a cabo el proceso de aplicación de la métrica es determinar los objetivos que se persiguen con la aplicación de la misma.

Una vez definido hacia que va estar orientado el resultado obtenido, o sea que información brindará.

O sea la propuesta consiste en un sistema de métricas que permiten obtener la productividad media del proceso de producción de la facultad.

Para obtener este resultado se debe partir de obtener la productividad de los proyectos lo que se obtiene con la métrica: $PP = PF / EP$. Sin embargo la productividad del proyecto se visualiza a

partir del esfuerzo de los módulos según las actividades que realizan, cuyo valor se obtiene aplicando la siguiente métrica $E_m = P / t$.

Evidentemente lo primero que se debe calcular es el esfuerzo de los módulos, para obtener la cantidad de personas (P) que realizan las actividades en un tiempo (t) definido en este caso por el líder del proyecto es necesario conocer las actividades que realizan y la complejidad de las mismas, para estimar la cantidad de personas que se necesitan y poder calcular el esfuerzo del modulo.

Por ejemplo se toma una muestra de módulos de un proyecto

Se ejemplifica el proyecto Prisiones (SIGEP), las actividades en los módulos se conciben como funcionalidades, o sea el modulo que se desarrolla tiene 5 funcionalidades, otro modulo en desarrollo consta de 11 funcionalidades, y puede darse el caso de otro modulo con 7 funcionalidades. Para cada modulo en dependencia de la cantidad de funcionalidades y la complejidad de las mismas, se estima una cantidad de personas para desarrollar dichas funcionalidades.

Una vez obtenidas la cantidad de personas que desarrollan dichas funcionalidades , el líder del proyecto define el tiempo limite de realizar las tareas, con estos datos puede obtener el esfuerzo por cada modulo según la métrica de esfuerzo del modulo sustituyendo los respectivos valores en la formula.

Luego de haber obtenido los esfuerzos de los respectivos módulos en desarrollo del proyecto, se puede obtener el esfuerzo medio del proyecto en general, sumando dichos esfuerzo entre la cantidad de módulos (**métrica 1.2**).

Con el esfuerzo medio se calcula entonces la productividad del proyecto para lo cual habría que calcular los puntos de función del proyecto y sustituir en la métrica al igual que el esfuerzo medio calculado.

Para calcular los puntos de función del proyecto se requiere conocer hasta donde ha avanzado el proyecto, para estimar el grado de funcionalidad que ha obtenido el mismo con el conteo de puntos de función según el procedimiento explicado anteriormente. **(Epígrafe 2.1.2)**

Con las productividades de los proyectos entonces se obtiene la productividad media del proceso general según la métrica definida **(métrica 1.4)**

El proceso definido anteriormente describe como se debe aplicar la métrica;

Sin embargo el valor obtenido, como se explica anteriormente, no aporta ninguna información, según lo que se quiere obtener se hacen las comparaciones necesarias:

Suponiendo que se quiere saber si se sigue en la facultad un procedimiento estándar

Entonces se define el rango de diferencia, por ejemplo se puede definir para las productividades obtenidas entre los proyectos no sobrepasen de una diferencia de 20, entonces se podrá apreciar si se sigue un procedimiento estándar o no.

Por otro lado si lo que se desea saber es el estado de la productividad entonces se compara con la productividad anterior y en dependencia de los valores se verifica si el proceso es productivo o no, o sea se define si es mayor que la anterior, según la complejidad de las actividades, entonces hay productividad en el proceso.

Es importante destacar que para llevar a cabo el proceso de aplicación de la métrica se debe designar en el proyecto una persona o grupo de personas que ejecuten los pasos de la aplicación de la métrica. Por supuesto estas personas deben ser capaces de realizar conclusiones según los resultados obtenidos para tomar decisiones sobre el proceso.

2.5 Conclusiones

Para definir un sistema de métrica que permita el control de la productividad en el proceso de desarrollo de software de una empresa determinada, hay que tener en cuenta elementos que caractericen el proceso para ajustar la solución que se propone, de igual forma hay que tener presente la experiencia de dicha institución en aplicaciones de métodos de este tipo para que le

proceso sea lo mas practico y eficiente posible. Sin embargo no existe una métrica perfecta que mida lo que se desea a la perfección, solo se definen las que mas cercanas estén al resultado que se desea obtener.



Capítulo 3: Validación de la Solución Propuesta

3.1 Introducción

Este capítulo aporta elementos que permitan validar la solución propuesta en la investigación. Los elementos se basan en la aplicación del sistema de métricas propuesto en uno de los proyectos de gestión de la Facultad.

3.2 Validación de métricas

En ocasiones se ha definido la medición como el proceso por el cual se asignan números o símbolos a atributos de entidades del mundo real. De esta manera, se disminuye cierto grado de subjetividad en un proceso de medición, mediante la utilización de métricas. Sin embargo, no sólo interesa contar con una diversidad de métricas, sino también, saber si dichas métricas son válidas. Es decir, si miden lo que realmente debieran medir. De esta manera, la validación se constituye en un aspecto crucial en la medición del software, ya que asegura que las medidas representen con precisión los atributos del software que se pretenden cuantificar.

En la literatura se reconocen dos tipos de validación: la interna y la externa, comúnmente referidas como teórica y empírica respectivamente

La validación interna es un ejercicio teórico que asegura que la métrica tiene una caracterización numérica apropiada de la propiedad que pretende medir. Esto es por supuesto un pre-requisito para demostrar la utilidad de dicha métrica (validación empírica).

La validación externa se lleva a cabo para demostrar con evidencia real que una métrica es útil en el sentido de que está asociada con alguna característica externa del software tal como la usabilidad, la mantenibilidad, etc.

Se propone validar la propuesta de métrica en función de esta información, demostrar que la propuesta es realmente útil y demostrar que matemáticamente se puede obtener un valor con el que se pueda obtener información.

3.3 Métricas de productividad en el proyecto

En el ambiente de las métricas, se plantea que las métricas de proyectos son la base de las métricas del proceso. En vistas a validar la propuesta de métrica de productividad definida en el capítulo anterior, se impone aplicar dicha métrica aun proyecto productivo de la facultad 4. Los resultados que se obtengan servirán de base para analizar la métrica propuesta.

Para aplicar la métrica en un proyecto se necesita conocer las características del proceso de desarrollo por el cual se rige dicho proyecto. Reconocer que en un proyecto es necesario aplicar una métrica, es un paso primordial para llevar a cabo el proceso de medición.

Las métricas para ser aplicadas en los proyectos productivos, estos deben contar con cierto nivel de madurez y organización dentro del proyecto.

La aplicación de métricas en un proyecto debe realizarse en todas las etapas del proyecto, incluso en la etapa de prueba.

La investigación arroja una propuesta que necesita ser validada, para esto es necesaria la aplicación de dicha propuesta en un proyecto de la facultad donde la aplicación va a estar basada en estimaciones.

En función de esto, se ha seleccionado un proyecto de la facultad en el que se aplica la propuesta: se hace referencia al Proyecto SIGEP (Sistema de Gestión Penitenciaria).

La autora propone que la validación de la propuesta de solución se realice a través de realizar una comparación entre el procedimiento que realizan los líderes para controlar la productividad de los proyectos y la aplicación de métricas para controlar la productividad y de esta forma obtener elementos que demuestren que el uso de las métricas es realmente útil.

Otro elemento que conforma la validación es aplicar la métrica a un proyecto donde quede demostrado que matemáticamente es factible el uso de las métricas.

3.4 Métrica de productividad vs. Procedimiento del proyecto (SIGEP).

Procedimiento.

El procedimiento consiste en definir actividades en un tiempo determinado dígame una semana, dichas actividades se conciben como funcionalidades.

Según el cumplimiento de estas funcionalidades se evalúa un por ciento de terminación demostrando el avance del proyecto y se estima igualmente la productividad. Esto quiere decir que el valor de la productividad es un estimado dado en valores de por cientos.

El valor de la productividad en por ciento no es del todo favorable para los objetivos que realmente se persiguen con el control de la productividad. Si se desea conocer si se sigue un procedimiento estándar en el proceso resulta muy engorroso realizar comparaciones basadas en por ciento. Por otro lado si lo que se desea obtener el valor de la productividad del proceso de producción de la facultad no sería factible obtener la productividad a través de la suma de por cientos. De manera general se plantea que es incomodo reunir información sobre el proceso basada en valores de por ciento.

Métrica

Aplicar métricas aporta valores consistentes. A partir de ellos se pueden obtener indicadores de diferentes aspectos que propician un análisis del proceso donde se identificaran los problemas existentes y establecerán mejoras potenciales a estos problemas. Con el resultado obtenido de la aplicación de la métrica, se pueden cumplir diferentes objetivos de interés para la organización a través de comparaciones principalmente si se sigue un proceso estándar en el proceso, evaluar la productividad del proceso en general a partir de la productividad de los proyectos.

Como se aprecia evidentemente resulta más beneficioso usar métricas que procedimientos empíricos como se usan en el proyecto en cuestión. Desde el punto de vista de que las métricas de productividad se conciben a nivel universal, no solo para saber cuan productivo esta siendo un proceso, sino para conocer si existen problemas en el proceso, identifican soluciones, que a

grandes rasgos es lo que mas aporta a una organización en vistas a producir software con calidad, lo que impone un proceso productivo con calidad, y esto es posible lograrlo con el uso de métricas.

3.5 Proyecto SIGEP.

El proyecto, ha sido concebido hace menos de un año, su estructura se basa en subsistemas, y cada sistema está compuesto por varios módulos.

Cada modulo se le asignan un conjunto de funcionalidades a desarrollar y la finalizarlas queda terminado dicho modulo.

Por tanto la productividad del proyecto tiene como base el esfuerzo de los módulos según las actividades que desarrollan.

A partir del planteamiento anterior, es factible aplicar la métrica propuesta en el proyecto, puesto que el proyecto en cuestión constituye un escenario favorable para aplicar la propuesta.

3.6 Productividad en el proyecto SIGEP

El control de la productividad en el proyecto se realiza basado en el siguiente flujo de trabajo:

Según el sistema de métricas propuesto (**epígrafe 2.5**) comienza por la aplicación en los módulos. Hay que tener presente seguir los pasos de la guía de aplicación propuesta, entonces:

Para obtener el esfuerzo de los módulos:

Se aplica la métrica: $E_m = p / t$

Luego se selecciona una muestra de 3 módulos, estimando una cantidad de personas según la cantidad de funcionalidades en un tiempo de 1 semana:

Una vez obtenido el esfuerzo de los módulos en esa semana, se aplica entonces la métrica 1.2 para obtener el esfuerzo del proyecto:

$$E_p = \sum \frac{E_m(i)}{C_m}$$

Sustituyendo

$$E_p = \sum 3 + 5 + 4$$

3

= 4 persona semana

Luego de haber obtenido el esfuerzo del proyecto se puede calcular la productividad del mismo utilizando la **métrica 1.3**

$$PP = PF / EP \text{ (p/día, mes, año)}$$

Para obtener los puntos de función, se tiene en cuenta que el proyecto esta en desarrollo por tanto se estima los puntos de función en dependencia de lo que se ha avanzado:

Se plantea que no se tiene experiencia en el cálculo de puntos de función en proyecto, suponiendo que el proyecto ha avanzado un 50 % , lo que significa un 50 % de funcionalidad, y como los puntos de función se centran en la funcionalidad del sistema, y teniendo en cuenta que el proyecto en cuestión es un proyecto con una complejidad máxima estaría en la complejidad media entonces se estima 300 puntos de función:

Por tanto

Si $PF = 200$ y $Ep = 4$ personas semana

Entonces

$PP = 200 / 4$ personas semana

PP = 50 puntos de función personas semana

Esto quiere decir que para esa semana el proyecto SIGEP concibe una productividad de 50 puntos de función por personas.

Con el valor obtenido, se puede realizar comparaciones con la productividad anterior si existe una base histórica de métrica aplicada, con la productividad de los demás proyecto para conocer si se sigue un procedimiento estándar, o sea el punto es que matemáticamente la métrica es factible utilizar la métrica obteniéndose como resultado valores consistentes con los que se pueden obtener indicadores para tomar decisiones en cuanto al proceso de producción.

3.7 Conclusiones

En este capítulo la autora se ha propuesto, de alguna manera validar la propuesta de solución planteada en el capítulo anterior, si aplicar la métricas en el proyecto que se selecciono para el

caso de estudio, que a su vez, es un proyecto de gestión, aporta un resultado válido, desde el punto de vista matemático, por lo que se infiere que es factible aplicar la propuesta en los proyectos de la facultad.

Conclusiones

La investigación tuvo gran repercusión en la preparación profesional de la autora por los conocimientos que aportó a la misma y por el cumplimiento de los objetivos trazados.

Se optó por realizar un estudio del proceso de producción de la Facultad 4 y el funcionamiento productivo de sus proyectos para determinar los elementos claves de su producción, para definir un sistema de métricas que permita controlar la productividad de los proyectos y del proceso en general fundamentalmente.

Se analizó mediante una muestra seleccionada de un proyecto de dicha facultad, la posible aplicación del sistema de métricas y se obtuvieron resultados satisfactorios para su aplicación futura en el proceso de producción de la Facultad.

La autora considera que el tema puede profundizarse un poco más para el perfeccionamiento del proceso de control de la productividad, en vistas a mejorar el proceso de producción de la Facultad 4.

Recomendaciones

A través de las encuestas realizadas de la concepción de la propuesta planteada en la investigación de arriban a una serie de conclusiones que permiten arrojar varias recomendaciones dirigidas a la mejora continua del proceso de producción de software en la Facultad 4.

- Crear las bases para el uso de métricas en los proyectos productivos de la Facultad 4 a través de cursos informacionales sobre el tema.
- Crear un rol en los proyectos productivos que se dedique a realizar los procesos de medición a través de aplicación de métricas. Permitiendo de esta manera que no influya esta actividad entre las actividades de los demás roles del proyecto.
- Automatizar este proceso para contar con un sistema automatizado de para el control eficiente y fiables de productividad del proceso productivo de la Facultad 4.

Bibliografía

Bibliografía Citada

- Antonio, A. d. (2004) GESTIÓN, CONTROL Y GARANTÍA DE LA CALIDAD DEL SOFTWARE. **Volume**, DOI:
- Cárdenas, D. S. A. (2002) CMM, las áreas de proceso y los papeles en la empresa de software. **Volume**, DOI:
- Carrasco, O. M. F., D. G. León, et al. (1995) ACIMED. Un enfoque actual sobre la calidad del software **Volume**, DOI:
- Durán, I. M. R. (2007) Mediciones Prácticas de Software y Sistemas (PSM): una propuesta para la producción de software en la UCI. **Volume**, DOI:
- Giraldo, O. P. (2004). "Métricas, Estimación y Planificación en Proyectos de Software." Retrieved 31/5, 2007, from www.WillyDev.Net.
- Hernandez, M. A. M. (2006) Personal Software Process/Team Software Process (PSP/TSP). **Volume**, DOI:
- Navarro, A. (2003). "Ingeniería de Software." Retrieved 5 mayo, 2007, from <http://www.fdi.ucm.es/profesor/anavarro/>.
- Pressman, R. S. (2005). Ingeniería del Software .Un enfoque practico. la Habana.
- Rodríguez., F. S. (1999). Medida del Tamaño Funcional de Aplicaciones Software, Universidad de Castilla-La Mancha
Escuela Superior de Informática de Ciudad-Real.
- Zavalar, J. (2002). Diseño de un Sistema de Información Geográfica sobre internet. Azcapotzalco., Universidad Autónoma Metropolitana.

BIBLIOGRAFIA

CAPITULO II - PROCESO DE SOFTWARE Y METRICAS DE PROYECTOS. 2007.
[2007]. Disponible en:
<http://netbuzos.spaces.live.com/blog/cns!994BD929B80714FB!127.entry>

ANDRADE, R. A.; A. C. GÓMEZ, *et al.* DESDE ISO 9001 HACIA CMMI, PASOS PARA LA MEJORA DE LOS PROCESOS Y MÉTRICAS. RPM-AEMES,, 2007. 4.

CARBALLO, R. *Las métricas durante el ciclo de vida de un proyecto subcontratado* 2006. [2007].

CARDENAS, D. S. A. and I. H. F. YANES. *Sitio para la implementación de las mejores prácticas de software.* Cuba, ISPJAE.

CARRASCO, O. M. F.; D. G. LEÓN, *et al.* ACIMED. *Un enfoque actual sobre la calidad del software*, 1995. 3.

CASTRO, S. R. and E. G. JARA. *Definición de un Sistema de Aseguramiento de Calidad para Actividad de Titulación en un Curriculum de Ingeniería de Software*, 2007].

Disponible en:

<http://lsm.dei.uc.pt/ribie/docfiles/txt200332513555Definici%C3%B3n%20de%20un%20Sistema%20de%20Aseguramiento%20de%20Calidad%20para%20Actividad%20de%20Titulaci%C3%B3n%20en%20un%20Curr%C3%ADculum%20de%20Ingenier%C3%ADa%20de%20Software.pdf>

CUELLAR, L. R. *Métricas Orientadas al Cliente- Definiendo una estrategia de medición.* México.

ESTRADA, A. F. *Medir el proceso de control de configuración, ¿una utopía para la Industria Nacional de Software?*

. La Habana, Cuba, Instituto Superior Politécnico “José Antonio Echeverría”

p.

FILHO, C. V. F. and J. M. D. REIS. *CONTROLA: HERRAMIENTA DE APOYO AL PROCESO DE DESARROLLO DE SOFTWARE EN LAS PEQUEÑAS COMPAÑÍAS.* *Revista Ingeniería Informática*, 2006.

G., M. I. *Modelos y métricas de calidad de software*, 2006. [2007]. Disponible en: GARCÍA, M. B. *Estimación de costes de un proyecto informático*, 2007]. Disponible en: <http://cv.uoc.es>

GIRALDO, O. P. *Métricas, Estimación y Planificación en Proyectos de Software*, 2007]. Disponible en: www.WillyDev.Net

GROMPONE, J. *Gestión de proyectos software*, 1996. [Disponible en:

HUMPHREY, W. S. *The Personal Software Process Overview, Practice, and Results*1. Pittsburgh, PA, p. *The Team Software ProcessSM (TSPSM)*. Pittsburgh, PA, 2000. p. 5213-3890

JARAMILLO, J. M. B. *“Indicadores de Gestión: herramientas para la competitividad”*. , 1999.

JOSÉ RAMÓN GUTIÉRREZ. *Construyendo un Modelo Abreviado de Métricas ITIL-COBIT*. 5ª Conferencia: Métricas e Indicadores, estándares de mercado 2006. p.

MANSO, E. Calidad del software. en. Valaldolid: 31.p.

MERCEDES, A. R. *MODELADO DINÁMICO Y APRENDIZAJE AUTOMÁTICO APLICADO A LA GESTIÓN DE PROYECTOS SOFTWARE*. *Revista de Procesos y Métricas de las Tecnologías de la Información* 2004. 1: 8.

MESTRAS, J. P. *Proceso de Software y Métricas de Proyectos*, 2004. [Disponible en: <http://www.fdi.ucm.es/>

MUÑOZ, A. P. *Indicadores de la sociedad de la información*, 2007

NARRO, M. T. *CMM Y RUP. Una perspectiva común*. , 2003.

NAVARRO, A. *Ingeniería de Software*, 2003. [2007]. Disponible en: <http://www.fdi.ucm.es/profesor/anavarro/>

OCA, C. M. D. *Team Software Process (TSP) - Un modelo de trabajo para Equipos de Desarrollo de Alto Rendimiento*. Disponible en: http://www.cimat.mx/info_general/como_llegar.html

PEREZ, J. R. P. *Configuración, Evaluación y Explotación de Sistemas Informáticos. Selección de Métricas*, 2002. 1.0.

PRESSMAN, R. S. *Ingeniería del Software .Un enfoque practico*. 5. la Habana, 2005. p.

RODRÍGUEZ, L. D. A. C.; L. R. S. NAD

AL, et al. *AUTOMATIZACIÓN DE LA GESTIÓN DE LA CALIDAD DE UNA*

ORGANIZACIÓN DE SOFTWARE PARTIENDO DE LA MEDICION DEL TAMAÑO.

. Las Villas , Cuba, UCLV“Martha Abreu” p

RODRÍGUEZ., F. S. *Medida del Tamaño Funcional de Aplicaciones Software*,
Universidad de Castilla-La Mancha

Escuela Superior de Informática de Ciudad-Real, 1999. p.

ROJAS, C. and M. VISCONTI. *Formulación y Aplicación de un Proceso de Análisis de Defectos de Software*. Disponible en:

ROLÓN, E.; F. RUIZ, *et al.* *APPLYING SOFTWARE PROCESS METRICS IN BUSSINESS PROCESS MODEL*. RPM-AEMES, 2006. 3.

RUBIO, M. G. and M. C. CUBERO. *ESTIMACIÓN DEL ESFUERZO DE UN PROYECTO*

SOFTWARE UTILIZANDO EL CRITERIO MDL-EM Y COMPONENTES NORMALES N-DIMENSIONALES.

APLICACIÓN A UN CASO PRÁCTICO.: *Revista de Procesos y Métricas de las Tecnologías de la Información*, 2005. 2.

S., B.; C. MANZANO, *et al.* *TEAM SOFTWARE PROCESS (TSP): MEJORAS EN LA ESTIMACIÓN, CALIDAD Y PRODUCTIVIDAD DE LOS EQUIPOS EN LA GESTIÓN DEL SOFTWARE*. RPM-AEMES, 2007. 4.

SAGI., D. J. B. El CMM y la mejora continua del proceso de software. en.p.

WIKIPEDIA. *Métrica de punto función*, 2007. . [2007]. Disponible en:
<http://es.wikipedia.org/wik/puntosdefuncion>

ZAVALAR, J. *Diseño de un Sistema de Información Geográfica sobre Internet*. Azcapotzalco., Universidad Autónoma Metropolitana, 2002. p.

Anexos

Tabla 1: Métricas del producto y del proceso

	Atributos internos	Atributos externos
Proceso Especificaciones, diseño, código...	Tamaño, reusabilidad, modularidad, funcionalidad, acoplamiento, complejidad...	Comprensión, mantenibilidad, calidad, fiabilidad ...
Producto Realización de la especificación, del diseño, del código...	Tiempo, esfuerzo, cambios en requisitos, fallos en la especificación	Calidad, coste, estabilidad
Recursos Personal, equipos, hardware, software...	Edad, precio, tamaño del equipo, velocidad, tamaño de memoria	Productividad, experiencia, calidad, usabilidad, fiabilidad

Tabla 2: Muestra del control de productividad en los proyectos de la Facultad.

Nombre del Proyecto	Numero de entrevistado y rol que desempeña en el proyecto	Control de la Productividad del Proyecto
SIGEP	1(Líder del proyecto)	si
MINFAR(módulo de Recursos Humanos)	1(Líder del proyecto)	si
Aduana	1(Líder del proyecto)	no
Banco	1(Líder del proyecto)	si
Digitalización de R&N	1(Líder del proyecto)	si

Tabla 3: Uso de métricas de productividad en los proyectos de la Facultad.

<i>NOMBRE DEL PROYECTO</i>	<i>USO DE METRICAS</i>
SIGEP	NO
MINFAR	NO
ADUANA	NO
BANCO	NO
DIGITALIZACION	NO

Tabla 4: Muestra de módulos seleccionados del proyecto SIGEP

Módulos	Personas por funcionalidades	Esfuerzo del módulo
Datos Personales	5 funcionalidades se asignan 3 personas	$Em = 3 / 1 = 3$ personas semana
Situación Jurídica	11 funcionalidades requiere de 6 personas aproximadamente	$Em = 5 / 1 = 5$ personas semana
Módulo de Información Operativa	7funcionalidades requiere de 4 personas	$Em = 4 / 1 = 4$ personas semana