Historial de Cambios

|  |  |  |
| --- | --- | --- |
| **Versión** | **Realizo** | **Descripción** |
| 0.1 | Juan Pablo Rodríguez Montoya | Elaboración plan |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Bibliografía

|  |  |
| --- | --- |
| [1] | IEEE Computer society, «IEEE Citation Reference,» 09 2009. [En línea]. Available: http://www.ieee.org/documents/ieeecitationref.pdf. [Último acceso: 10 09 2012]. |
| [2] | J. M. Cueva Lovelle, «Calidad del Software,» 1999 Octubre 21. [En línea]. Available: http://www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r35043.PDF. [Último acceso: 20 08 2012]. |
| [3] | E. Rolón, F. Ruiz, F. García y M. Piattini , «Aplicación De Métricas Software En La Evaluación De Modelos De Procesos De Negocio,» [En línea]. Available: http://users.dcc.uchile.cl/~mmarin/revista-sccc/sccc-web/Vol6/Art09.pdf. [Último acceso: 1 09 2012]. |

# Plan de Recolección de métricas

El plan de recolección de métricas será implementado para poder conocer la calidad, la productividad y las funcionalidades implementadas, esto ayudara a que el gerente tome decisiones acerca de mejorar la calidad, amonestar o premiar a un integrante y conocer el progreso las funcionalidades implementas.

## 1.1. Objetivos

* Identificar con un indicador la calidad de los productos de trabajo que realiza el grupo FifthFloorCorp.
* Identificar un indicador la productividad de los integrantes del grupo.
* Identificar un porcentaje para conocer progreso de las funciones implementadas.

## 1.2. Responsables

El responsable del manejo de este plan es el Director de Calidad el cual evaluara los indicadores que le den resultado por este plan.

## 1.3. Recursos

* Plan de Verificaciones y Validaciones (ver plan Verificaciones y Validaciones).
* Plantilla de horas de trabajo
* Plantilla Funcionalidad Implementada.
* Director de Calidad

## 1.4. Desarrollo

### 1.4.1. Métricas de calidad

Para las métricas de calidad se establece una plantilla que calificara el mismo (ver V&V ‘s), la cual se realiza después de que un integrante halla terminado un producto y le informe vía correo electrónico al gerente, el gerente asignara equitativamente las revisiones así un integrante revisara el producto y lo calificara en la plantilla del mismo.

Otra métrica de calidad es la que se realiza con el auditor (ver plan de revisiones y auditorias) que son una ayuda externa para verificar que el producto sea de calidad, estas revisiones se realizan en citas programadas con el auditor donde este evaluara lo realizado y propondrá o comentara acerca del documento evaluado.

### 1.4.1.1. Calidad de ortografía

Para los documentos se va a calificar la ortografía del integrante desarrollador del mismo, para ello se tienen las siguientes calificaciones

* 5 - muy buena
* 4 - buena
* 3 - regular
* 2 - mala
* 1 - muy mala

### 1.4.1.2. Calidad en uso de referencias

Para los documentos y la documentación del código se tendrá una calificación de referencias de tipo IEEE [1] la cual es la siguiente:

* 5 - muy buena
* 4 - buena
* 3 - regular
* 2 - mala
* 1 - muy mala

### 1.4.1.3. Calidad cohesión, coherencia y redacción

Para los documentos, la documentación del código y el orden del código se realizaran revisiones cruzadas (ver revisiones cruzadas) la cual permitirá que cada integrante verifique y valide cada uno de los productos de trabajo.

### 1.4.1.4. Revisiones cruzadas

Es un tipo de revisión que se realiza cada viernes durante el desarrollo del proyecto por los integrantes del FifthFloorCorp, este se estableció de la siguiente manera:

### 1.4.2. Métricas de Productividad

Para esta métrica cada integrante tiene a su disposición un programa llamado ManicTime [(ver Herramienta ManicTime)](#_1.4.2.1_Herramienta_ManicTime) que permite al integrante ver la cantidad de tiempo exacto que trabajo en un producto. Este con ayuda de la herramienta diligenciara una plantilla donde digitara las horas trabajadas en que producto y una breve explicación de lo que realizó (ver plantilla de Métricas de Tiempo).

### 1.4.2.1 Herramienta ManicTime

Es una herramienta que mide el tiempo utilizado en los diferentes software del sistema por ejemplo Microsoft Office, Bizagi Process Modeler ,etc. Además de medir permite sacar una estadística general de todo el trabajo realizado por cada integrante, la disponibilidad de este software se encuentra en <http://www.manictime.com/>

#### 1.4.2.1.1 Proceso de Medición ManicTime

La medición del tiempo la realiza una vez se abra el programa. El integrante debe abrir este programa cada vez que valla a trabajar sobre algún producto de trabajo del proyecto.

#### 1.4.2.1.2 Proceso Grafica de tiempo empleado

Este proceso se realiza para sacar la grafica que muestra las horas productivas del integrante de FifthFloorCorp la cual se muestra en a siguiente grafica:

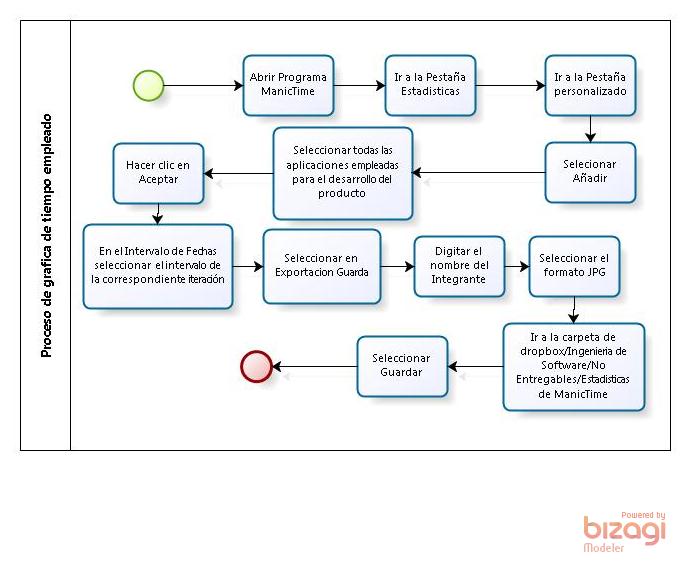


Ilustración 1: Proceso de grafica de tiempo empleado

### 2.4.3. Métricas de Funcionalidad Implementada

Para esta métrica se tendrán en cuenta los casos de uso (ver casos de uso) ya que sobre estos se realizaran los puntos de función. Para la medición se tendrá en cuenta el tiempo empleado para desarrollar el punto de función, cantidad de líneas de código, defectos del punto de función y ventajas del mismo, para ello se tendrá una plantilla que tenga en cuanta los anteriores puntos (ver plantilla de funcionalidad implementada). Esta métrica comienza a funcionar desde la 3 fase en adelante ya que se medirá desde el primer prototipo.

### 2.4.3.1 Medición [2]

Cada factor de calidad se puede obtener como combinación de una o varias métricas:

– Ci es el factor de ponderación de la métrica i, que dependerá de cada aplicación específica

– mi métrica i

– Estos factores de ponderación y las métricas puntúan de 0 a 10 en las métricas y en los factores de calidad

### 2.4.3.2 Métricas para determinar los factores de calidad [2]

– Facilidad de auditoria

– Exactitud

– Normalización de las comunicaciones

– Completitud

– Concisión

– Consistencia

– Estandarización de los datos

– Tolerancia de errores

– Eficiencia de la ejecución

– Facilidad de expansión

– Generalidad

– Independencia del hardware

– Instrumentación

– Modularidad

– Facilidad de operación

– Seguridad

– Auto-documentación

– Simplicidad

– Independencia del sistema software

– Facilidad de traza

– Formación