Sistema de Información Integrado (SII)

Plan de medición

Versión 1.0

Historial de revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 20/09/2014 | 1.0 | Plan de medición de proyecto. | Brayham Davian Burgos Bedoya  Jorge Hernán Ospina León |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Tabla de contenido

1. Introducción 4

1.1 Propósito 4

1.2 Alcance 4

2. Administración de metas y submetas 4

3. Métricas 5

3.1 Horario y progreso 5

3.1.1 Tareas completadas 5

3.1.2 Tareas tardías 6

3.1.3 Estado del caso de uso 6

3.1.4 Solicitud de cambio de prioridad 6

3.1.5 Caso de prueba de estado 7

3.2 Recursos y costos 7

3.2.1 Variación del cronograma 7

3.2.2 Variación de los gastos 8

3.2.3 Horas de trabajo 8

3.3 Tamaño del producto y estabilidad 9

3.3.1 Líneas de código creadas, modificadas y borradas 9

3.3.2 Requerimientos 9

Plan de medición

# Introducción

La medición de la calidad de software brinda unas métricas específicas para valorar y calificar la calidad de un proceso y de un producto software. Para poder asegurar que un proceso o sus productos resultantes son de calidad o poder compararlos, es necesario asignar valores, descriptores, indicadores o algún otro mecanismo mediante el cual se pueda llevar a cabo dicha comparación.

En este documento quedaran plasmadas las métricas trazadas para medir tanto la calidad de los procesos, como el producto final entregado al cliente.

## Propósito

* Ayudarnos a entender que ocurre durante el desarrollo y el mantenimiento del proyecto.
* Permitir controlar que es lo que ocurre en nuestros proyectos.
* Poder mejorar los procesos y los productos.
* Todo proceso de medición del software tiene como objetivo fundamental satisfacer necesidades de información.
* A partir de las necesidades de información se deben identificar las entidades y los atributos de dichas entidades que son candidatos a ser medidos.
* Una vez identificados los atributos objeto de la medición se deben definir las medidas necesarias. En la definición general de una medida se deben especificar aspectos como la unidad en la que se expresa, la escala a la que pertenece, el atributo o atributos para los que se define.

## Alcance

El alcance del plan de medición es demostrar la importancia de gestionar los proyectos y las actividades críticas de la organización en base a métricas, de acuerdo a un conjunto de indicadores alineados a las necesidades del proyecto.

Está dirigido a todos los responsables de la definición de indicadores y métricas dentro de un proyecto software, además de estar dirigido a todos los integrantes del equipo de desarrollo que buscan la mejora

de sus procesos y de la calidad de los productos realizados.

# Administración de metas y submetas

El objetivo de un programa de medición es proporcionar datos objetivos para que los equipos de gestión, desarrollo y pruebas puedan tomar decisiones basadas en la necesidad de información. Por lo tanto, un programa de medición exitosa consiste en identificar primero las necesidades de información.

Las siguientes necesidades de información se han identificado para el proyecto:

* En el proyecto anterior de software III, la entrega inicio a retrasarse en el momento de iniciar la implementación del proyecto, ya que no fue sencillo dominar el framework de desarrollo que elegimos, y en la implementación se presentaron todas las fallas para entregar el producto en el tiempo establecido.
* En el mismo proyecto, también se contó con el agravante de tiempo, ya que solo fue desarrollado por dos personas, y el tiempo estipulado en el cronograma de trabajo no se pudo mantener y cumplir.
* Además, las pruebas unitarias fueron un requisito para cada una de las entregas del proyecto, y en ninguna de ellas se lograron entregar, ya que no se encontró buena documentación y tutoriales que sirvieran de base para la realización de las pruebas.

# Métricas

Sobre la base de las necesidades de información, podemos refinar las medidas utilizando la siguiente tabla de Información de Categoría, Conceptos medibles y Medidas:

|  |  |  |
| --- | --- | --- |
| **Categoría de la información** | **Conceptos medibles** | **Medidas** |
| **Horario y progreso** | Trabajo/Unidad/Progreso | * Tareas completas * Tareas tardías * Estado del caso de uso * Solicitud de cambio de prioridad * Caso de prueba de estado |
| **Recursos y costos** | Desempeño financiero | * Variación del cronograma * Variación de los gastos * Horas de trabajo |
| **Tamaño del producto y estabilidad** | Tamaño físico y estabilidad | * Líneas de código creadas, modificadas y borradas |
| Tamaño funcional | * Requerimientos * Revisiones * Casos de uso * Secuencia de casos de uso |
| **Calidad del producto** | Corrección funcional | * Defectos por gravedad |

## Horario y progreso

### Tareas completadas

|  |  |
| --- | --- |
| **Nombre** | **Tareas completadas** |
| **Definición** | Determinar si las tareas previstas se realicen de manera progresiva. |
| **Metas** | El objetivo de esta métrica es garantizar que las tareas sean terminadas progresivamente y en las fechas acordadas. |
| **Procedimiento de análisis** | Esta medida se utiliza para realizar el seguimiento del número de tareas completadas en un tiempo dado. El número de tareas completadas deben aproximarse a cero cuanto más nos acercamos a la fecha estipulada de la entrega. Esto demuestra que se está logrando progreso y que se están cumpliendo con las entregas. |
| **Responsabilidades** | Estará a cargo de los responsables del proyecto y del usuario, al que en las fechas establecidas y acordadas, se le hará la entrega de las tareas acordadas, es decir de los casos de uso ya implementados. |

# 

### Tareas tardías

|  |  |
| --- | --- |
| **Nombre** | **Tareas tardías** |
| **Definición** | Información del número de tareas que no se entregaron a tiempo |
| **Metas** | El objetivo de esta medida es garantizar que las tareas se realicen de manera oportuna. |
| **Procedimiento de análisis** | El número de tareas terminadas debe coincidir con las tareas establecidas en las entregas. Una tarea está determinada a ser tardía si a la fecha Final de la entrega no se a terminado totalmente. Es decir, si la fecha Final real se establece, entonces la tarea pasa a ser tardía si la fecha Final real de entrega es posterior a la fecha de Final prevista. |
| **Responsabilidades** | Estará a cargo de los responsables del proyecto, ya que toda tarea tardía tiene que estar debidamente justificada y verificar cual fue el motivo por el cual no se terminó a tiempo. |

### Estado del caso de uso

|  |  |
| --- | --- |
| **Nombre** | **Estado del caso de uso** |
| **Definición** | Determinar la evolución de cada uno de los casos de uso del sistema |
| **Metas** | El objetivo de esta métrica es asegurar que el desarrollo de los requisitos o casos de uso del proyecto estén progresando de manera oportuna. |
| **Procedimiento de análisis** | Para el proyecto SII, específicamente los casos de uso del módulo PQR, los requisitos progresan a través de varias etapas (o estados). Estas etapas son: propuesta, aprobada, Incorporada, y validada. Cuando se acerque a una entrega, el número de requisitos debe estar en un cierto estado. Por ejemplo, al final de la implementación, se puede esperar que todos los requisitos del proyecto estén en el estado Aprobado. De lo contrario, la entrega del módulo PQR no fue completa |
| **Responsabilidades** | Estará a cargo de los responsables del proyecto, los cuales cambiaran los estados de los casos de uso a medida que se cumplan con todos los requerimientos |

### Solicitud de cambio de prioridad

|  |  |
| --- | --- |
| **Nombre** | **Solicitud de cambio de prioridad** |
| **Definición** | Evaluar la disposición que tiene el equipo de trabajo para el lanzamiento del producto mediante el seguimiento al número de solicitudes de cambio solicitadas por prioridad. |
| **Metas** | El objetivo de esta métrica es determinar la preparación del equipo de trabajo para afrontar la liberación del producto mediante el seguimiento al progreso de las solicitudes de cambio resueltos por prioridad. Se presta especial atención a la prioridad 1 y 2 de solicitudes de cambio. |
| **Procedimiento de análisis** | Esta medida se utiliza para seguir el progreso de las solicitudes de cambio de alta prioridad, los cuales son aquellos defectos y solicitudes de mejora que se deben incluir en un caso de uso. El número de solicitudes de cambio de alta prioridad estará abierta antes de cada entrega y deberá disminuir a medida que el proyecto ya esté en su fase final. Solicitudes de cambio pueden ser clasificados por los siguientes valores de prioridad:   * Prioridad 1 - Resolver inmediatamente * Prioridad 2 - Dar de alta Atención * Prioridad 3 - Cola normal * Prioridad 4 - Bajo |
| **Responsabilidades** | Las solicitudes de cambio de alta prioridad serán manejadas por el equipo de desarrollo, y deben tener un tiempo prudente de notificación para poder analizar cómo repercute en el proyecto y medir el tiempo de respuesta. |

### Caso de prueba de estado

|  |  |
| --- | --- |
| **Nombre** | **Caso de prueba de estado** |
| **Definición** | Determinar el progreso de la ejecución de casos de prueba |
| **Metas** | El objetivo de esta métrica es asegurar que la planificación y ejecución de casos de prueba este progresando de manera oportuna. |
| **Procedimiento de análisis** | Para el proyecto se proveerán los casos de prueba y luego de ser ejecutados, se registrara el número de casos de prueba aprobados y los que no pasaron, para después realizar el debido seguimiento. Al acercarse la fecha de la entrega, el número de casos de prueba exitosos debe ser igual al número de casos de prueba planificados y ejecutados. |
| **Responsabilidades** | El equipo desarrollador debe usar una herramienta especial para manejar los casos de prueba dependiendo del lenguaje de programación en el que se esté desarrollando el proyecto |

## Recursos y costos

### Variación del cronograma

|  |  |
| --- | --- |
| **Nombre** | **Variación del cronograma** |
| **Definición** | Determina si la cantidad de tiempo de trabajo realizado es más o menos de lo previsto y se verifica su efectividad. |
| **Metas** | El objetivo de esta métrica es dar el director del proyecto y a sus integrantes un indicador de si el valor del trabajo realizado es más o menos de lo que estaba previsto. |
| **Procedimiento de análisis** | El valor de variación del cronograma debe estar cerca de 0 Si el valor cae por debajo de 0 (negativo), entonces el valor del trabajo efectivamente realizado es menor de lo previsto y que están detrás de lo programado. Si el valor es superior a 0 (positivo), entonces el valor del trabajo efectivamente realizado es más de lo esperado y está por delante de lo previsto. En este caso servirá de apoyo el kanban con los casos de uso que se implementaran para cada caso de uso, y de acuerdo a las tareas de cada caso de uso se le dará un valor por cada tarea finalizada. Si al llegar la fecha de la entrega, el caso de uso tiene tareas aun inconclusas, se le dará valores negativos dependiendo del número de tareas que aun falten |
| **Responsabilidades** | El equipo desarrollador dividirá las tareas de cada caso de uso en días útiles dependiendo su complejidad, y basado en el kanban y las fechas de entrega se le dará los valores para verificar si se está cumpliendo con el cronograma de trabajo. |

### Variación de los gastos

|  |  |
| --- | --- |
| **Nombre** | **Variación de los gastos** |
| **Definición** | Determina si el proyecto en cuestión de gastos y presupuesto está más o menos a lo planificado |
| **Metas** | El objetivo de esta métrica es dar el director del proyecto un indicador que le permita verificar si el valor del trabajo realizado es más o menos de lo que estaba previsto. |
| **Procedimiento de análisis** | La variación de gastos y presupuestos para este proyecto variara de acuerdo a la cantidad de horas totales utilizadas por cada integrante en el proyecto. Si se requieren usar más horas diarias en el proyecto, los gastos fueron superiores, por el contrario, si en la implementación las horas hombre utilizadas, es menor a las previstas, los gastos fueron menores y el proyecto genero ganancia. |
| **Responsabilidades** | El equipo desarrollador se basara en el kanban del proyecto para dejar constancia de las horas empleadas en el proyecto y en cada caso de uso del módulo PQR, para al final calcular la cantidad de horas utilizadas y verificar si la variación de los gastos fue favorable o si genero perdidas |

### Horas de trabajo

|  |  |
| --- | --- |
| **Nombre** | **Horas de trabajo** |
| **Definición** | Determinar en qué punto se encuentra el proyecto en su desarrollo, y la cantidad de horas de trabajo realizado. |
| **Metas** | El objetivo de esta métrica es mostrar el progreso del desarrollo en términos de la cantidad de horas de trabajo realizadas frente a las horas de trabajo planificadas. |
| **Procedimiento de análisis** | A medida que el proyecto avanza y el trabajo se ha completado, la cantidad de trabajo se mide en horas de trabajo. Este valor puede ser comparado con las horas de trabajo planificadas para determinar qué tan avanzado está el proyecto en el desarrollo. A medida que el proyecto avanza, las horas reales de trabajo realizadas deben acercarse a las horas de trabajo planificadas. |
| **Responsabilidades** | El equipo desarrollador comparara las horas registradas en el kanban del proyecto, con las horas establecidas en el documento de medición y estimación del proyecto. |

## Tamaño del producto y estabilidad

### Líneas de código creadas, modificadas y borradas

|  |  |
| --- | --- |
| **Nombre** | **Líneas de código creadas, modificadas y borradas** |
| **Definición** | Determinar la estabilidad del software para la preparar la entrega final al cliente |
| **Metas** | El objetivo de esta métrica es mostrar la estabilidad de los elementos de software configurados. |
| **Procedimiento de análisis** | Esta medida se utiliza para rastrear la estabilidad del elemento de software configurado. El número de líneas añadidas, modificadas y eliminadas, debe aproximarse a cero al momento de acercarse la entrega al usuario. Esto demuestra que no hay más cambios que se estén aplicando a los elementos configurados en el momento y se puede concluir que el software es estable |
| **Responsabilidades** | El equipo desarrollador usara de apoyo para verificar la calidad del código dela herramienta sonarqube, y así llevar un control sobre todo el código del proyecto. |

### Requerimientos

|  |  |
| --- | --- |
| **Nombre** | **Requerimientos** |
| **Definición** | Determina el número de requisitos y sub-requisitos solicitados en el proyecto |
| **Metas** | El objetivo de esta métrica es realizar un seguimiento del tamaño y la complejidad del proyecto. |
| **Procedimiento de análisis** | Esta medida se utiliza para realizar el seguimiento del tamaño y la complejidad del proyecto. La medida también se puede utilizar para realizar un seguimiento de la estabilidad de los requisitos del proyecto. El número de nivel 0 y 1 identifica que requerimiento ha sido resuelto con éxito y cual no. En espera 1. Resuelto |
| **Responsabilidades** | El equipo desarrollador dejara plasmado el estado de los requerimientos dentro del mismo kanban del proyecto |

3.3.3 Revisiones

|  |  |
| --- | --- |
| **Nombre** | **Revisiones** |
| **Definición** | Determinar la estabilidad de los requisitos de software para la preparación de liberación. |
| **Metas** | El objetivo de esta métrica es mostrar la estabilidad del proyecto y de los procesos de desarrollo. Esta medida se puede utilizar para determinar si una entrega está completa. También se puede utilizar para determinar la calidad del producto. Las continuas revisiones mostraran la inestabilidad de las funcionalidades del proyecto que pueden conducir a horarios extras o a excesos de costos. |
| **Procedimiento de análisis** | Esta métrica se utiliza para rastrear la estabilidad de los requisitos. Cuando un proyecto está a punto de concluir, el número de requisitos a revisar ​​debería aproximarse a cero. Y debe ser uno de los criterios utilizados para determinar la terminación de las entregas. |
| Responsabilidades | El equipo desarrollador realizara revisiones semanales para determinar el estado de los casos de uso, los tiempos establecidos, organización de entregas y determinación de ampliación de horarios de trabajo. |

3.3.4 Casos de uso

|  |  |
| --- | --- |
| **Nombre** | **Casos de uso** |
| **Definición** | Determinar el número de casos de uso para el proyecto |
| **Metas** | El objetivo de esta métrica es realizar un seguimiento de todos los casos de uso del proyecto |
| **Procedimiento de análisis** | Esta medida se utiliza para realizar el seguimiento del tamaño y la complejidad del proyecto mediante el seguimiento del número de casos de uso. La medida también se puede utilizar para realizar un seguimiento de la estabilidad de la definición de casos de uso. El número de casos de uso resueltos debe estar limitada por cada entrega y al final del proyecto deben estar finalizados todos |
| **Responsabilidades** | El equipo desarrollador realizara el control de los casos de uso dependiendo cada una de las entregas y su control se llevara a cabo en el kanban. |

3.3.5 Secuencia de casos de uso

|  |  |
| --- | --- |
| **Nombre** | **Secuencia de casos de uso** |
| **Definición** | Determina el número de pasos de los casos de uso para el proyecto |
| **Metas** | El objetivo de esta métrica es realizar un seguimiento del tamaño y la complejidad del proyecto. |
| **Procedimiento de análisis** | Esta métrica se utiliza para realizar el seguimiento del tamaño y la complejidad del proyecto mediante el seguimiento de los números de pasos de casos de uso. La medida puede ser usada para rastrear la estabilidad de la definición de los casos de uso. |
| **Responsabilidades** | El equipo desarrollador tendrá definido cada uno de los pasos e iteraciones que debe cumplir cada caso de uso del proyecto, y estarán plasmados dentro del kanban del proyecto. |

**3.4 Calidad del producto**

3.4.1 Defectos por gravedad

|  |  |
| --- | --- |
| **Nombre** | **Defectos por gravedad** |
| **Definición** | Evaluar la calidad del producto antes de ponerlo a disposición del cliente, evaluando el número de defectos encontrados y la implicación que trae al proyecto |
| **Metas** | El objetivo de esta métrica es realizar un seguimiento de la calidad del software entregado. |
| **Procedimiento de análisis** | Esta medida se utiliza para realizar un seguimiento a la calidad del proyecto mediante el seguimiento de defectos por gravedad. Los defectos de severidad alta hacen que el producto resulte inservible o difícil de usar. La métrica se puede utilizar también para rastrear el progreso de la resolución de los defectos por gravedad Los defectos se pueden clasificar según su gravedad:  Gravedad 1: defecto crítico. El defecto hace que el producto resulte inutilizable.  Gravedad 2: defecto mayor. Puede existir una solución alternativa pero es indeseable.  Gravedad 3: defecto promedio. Puede seguir trabajando. Existe una solución provisional.  Gravedad 4 - defecto menor. Defecto cosmético. |
| **Responsabilidades** | El equipo desarrollador debe hacer una prueba final a todo el producto antes de ser entregado, para determinar uno a uno los defectos encontrados y clasificarlos para solucionar todos los problemas antes de la entrega final al cliente. |