**Team Software Process**

**Ciclo 1**



**Nombre Proyecto:** Sistema TSP

**Fecha:** Marzo 9 de 2011

**Realizado por:**

|  |  |
| --- | --- |
| **Persona** | **Código** |
| Carlos Ernesto González Vargas | 200819123 |
| Willian Alejandro Idrobo Luna | 201110544 |
| Erik Fernando Arcos Franco | 201110856 |
| David Pérez Chibuque | 201117818 |
| Andrés Mauricio Erazo Benavides | 201110949 |
| Sandra Milena Gómez Ríos | 201110951 |

**Control de versiones**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Autor** | **Descripción del Cambio** |
| 1.00 | Marzo 9 de 2011 | Ingenium | Creación del documento |
| 1.01 | Marzo 10 de 2011 | Líder de Desarrollo | Definición del diseño conceptual |
| 1.02 | Marzo 14 de 2011 | Lider de Desarrollo | Se define análisis y diseño |
|  |  |  |  |
|  |  |  |  |

 

**Contenido**

[**1.** **Documento de Lanzamiento** 2](#_Toc287589279)

[**1.1.** **Objetivos y Métricas** 2](#_Toc287589280)

[**1.1.1.** **Objetivos del Grupo** 2](#_Toc287589281)

[**1.1.2.** **Objetivos de los Miembros del Grupo** 2](#_Toc287589282)

[**1.1.3.** **Objetivos del Proyecto y del Proceso** 2](#_Toc287589283)

[**1.2.** **Definición de los roles del equipo** 3](#_Toc287589284)

[**1.3.** **Reglas** 3](#_Toc287589285)

[**1.4.** **Horarios de Reunión** 3](#_Toc287589286)

[**1.5.** **Información que se reporta en cada ciclo** 3](#_Toc287589287)

[**2.** **Delimitación del Proyecto** 5](#_Toc287589288)

[**2.1.** **Definición del requerimiento** 5](#_Toc287589289)

[**2.2.** **Diseño Conceptual** 6](#_Toc287589290)

[**2.3.** **Estimación de Tiempo y Esfuerzo** 7](#_Toc287589291)

[**2.3.1.** **Regresión Lineal** 9](#_Toc287589292)

[**2.3.2.** **Estimación de Tiempo** 11](#_Toc287589293)

[**2.4.** **Plan de trabajo** 12](#_Toc287589294)

[**2.5.** **Requerimientos planeados por ciclo** 12](#_Toc287589295)

[**3.** **Definición Primer ciclo** 13](#_Toc287589296)

[**3.1.** **Objetivo** 13](#_Toc287589297)

[**3.2.** **Plan de Trabajo** 13](#_Toc287589298)

[**3.3.** **Entregables** 13](#_Toc287589299)

[**4.** **PostMortem Primer Ciclo** 14](#_Toc287589300)

**Team Software Process**

**Ciclo 1**

1. **Documento de Lanzamiento**
   1. **Objetivos y Métricas**
      1. **Objetivos del Grupo**

* O1: Producir un producto de Buena Calidad
  + M1: 100% Requerimientos incluidos en el producto final
  + M2: Porcentaje de defectos encontrados en cada etapa
* O2: Realizar un proyecto bien administrado y productivo
  + M3: Porcentaje de error máximo permitido en la estimación de tamaño del producto
  + M4: Porcentaje de error máximo permitido en la estimación de cantidad de horas
    1. **Objetivos de los Miembros del Grupo**
* O3: Ser un miembro efectivo y cooperativo
* O4: Hacer el trabajo personal de manera disciplinada consistentemente
  + M5: Promedio  de evaluación dentro del grupo superior a 3
* O5: Planear y hacer seguimiento al trabajo personal
  + M6: Se registro el 10%% de las actividades  y resumen de calidad
  + M7: Porcentaje de tareas planeadas y completadas, promedio semanal
  + M8: Ser puntuales a todas las reuniones planeadas (puntualidad> 80%)
    1. **Objetivos del Proyecto y del Proceso**
* O6:Objetivos del proyecto por ciclo
  + M9:Redefinir objetivos del documento por ciclo
* O7: Tiempo definido para cada tarea
  + M10: El tiempo empleado por tarea no debe superar el 20% del tiempo planeado
* O8: Definir documento de análisis de requerimientos para cada ciclo
  + M11: Identificar el 95% de los requerimientos
* O9: Cumplir los requerimientos definidos en el ciclo
  + M12 Finalizar a tiempo
  + M13 Implementar el 80% de los requerimientos
* O10:Construir codigo mantenible y bajo estandares
  + M14: Cubrimiento de métodos y atributos con Javadoc en 70%
  + M15: Componentes nombrados de acuerdo al estándar en un 70%
  + M16: Código con formato de indentación en un 100%.
* O11:Desarrollar pruebas para los requermimientos funcionales
  + M17: Desarrollar 1 prueba automatizadas para cada requerimiento establecido del ciclo
* O12:El tiempo de los miscelaneos   debe ser bajo respecto al tiempo  del proyecto
  + M18: Los miscelaneos no debe superar el 10% por ciclo
* O13:Realizar diseño de interfaz antes de la codificación
  + M19: Diagrama de flujo por requerimiento = 1
  + M20: Prototipo de interfaz por requerimiento = 1
* O14:Realizar diagrama de clases
  + M21: Realizar un diagrama de clases para el producto
* O15:Documentos y formatos
  + M22: Definir una plantilla para los reportes de defectos = 1
  + M23: Porcentaje de defectos encontrados con “Defect Report” =  70 %
  + M24: Porcentaje de defectos aprobados para solución efectivamente solucionados (Antes de la prueba de sistema) = 90%
  + M25: Errores encontrados en pruebas de sistema no mayor al 10%
  1. **Definición de los roles del equipo**

SANDRA Descripción Breve

Acá se describiría el proceso que hicimos y colocamos los resultados en la siguiente tabla diciendo que teniendo como referencia nuestros trabajos profesionales y los trabajos que hemos desarrollado juntos a lo largo de ECOS bla bla bla:

Tabla 1. Definición de Roles

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Líder de** | | | | |
| **Integrante** | **Grupo** | **Desarrollo** | **Planeación** | **Calidad** | **Soporte** |
| Carlos Ernesto González | 50 |  |  |  |  |
| Willian Alejandro Idrobo |  | 44 |  |  |  |
| Erik Fernando Arcos |  |  | 75 |  |  |
| David Pérez |  | Van los valores |  | 43 |  |
| Andrés Mauricio Erazo |  | Reales |  |  | 54 |
| Sandra Milena Gómez | Esos me los | Invente…. | Obviamente :P | 33 |  |

Con los resultados de la anterior tabla se tiene que a criterio del grupo los mas indicados son bla bla bla:

**Líder del Grupo:** Carlos Ernesto González

**Líder de Desarrollo:** Erik Fernando Arcos

**Líder de Desarrollo:** Willian Alejandro Idrobo

**Líder de Planeación:** Sandra Milena Gómez

**Líder de Calidad:** David Pérez

**Líder de Soporte:** Andrés Mauricio Erazo

* 1. **Reglas**

SANDRA Descripción Breve, las de juego pueden ser

* 1. **Horarios de Reunión**

SANDRA Descripción Breve

* 1. **Información que se reporta en cada ciclo**

Carlos Descripción Breve

* Análisis personal
  + Objetivos definidos por rol
  + Cumplimiento de los objetivos durante el ciclo
  + Inconvenientes
  + Aspectos a mejorar
* Reflexión
  + ¿Qué nos faltó como grupo en este ciclo?
  + ¿Cómo debería ser el proceso en el próximo ciclo?
  + ¿Qué etapas fueron las más difíciles? Porqué?
  + ¿Qué no me gustó del ciclo?

1. **Delimitación del Proyecto**
   1. **Definición del requerimiento**

El producto que se desarrollará corresponde al software encargado de medir y presentar la productividad del equipo de trabajo y la de cada uno de sus integrantes a través del conteo y recuperación de anotaciones que se han ingresado en los archivos java que conforman el proyecto. El producto debe presentar una interfaz gráfica.

Se han identificado las siguientes funcionalidades del producto:

* Permitir al usuario registrar el grupo de trabajo dentro de la aplicación
* Permitir al usuario registrar las tareas planeadas por cada ciclo y asignarlas a un responsable.
* Permitir al usuario registrar el plan de calidad
* Permitir al usuario generar el reporte de productividad del grupo, así como también la productividad de cada integrante del grupo.

|  |  |
| --- | --- |
| **Identificador del Requerimiento** | RF1.0 |
| **Nombre del Requerimiento** | Administración de miembros del equipo |
| **Actores** | Usuario |
| **Propósito** | Este requisito comprende la creación, modificación y consulta de los miembros del equipo en la aplcación. Todos los miembros registrados se deben persistir en un archivo de propiedades. |
| **Principales Parámetros** | **Resultados** |
| 1. Identificador del integrante del grupo. | El integrante ha sido creado, modificado o eliminado del archivo de propiedades |
| 2. Nombre del integrante del grupo |  |
| 3. Rol del integrante del grupo |  |

|  |  |
| --- | --- |
| **Identificador del Requerimiento** | RF2.0 |
| **Nombre del Requerimiento** | Administración de planeación de tareas |
| **Actores** | Usuario |
| **Propósito** | Este requisito comprende la creación, modificación y eliminación de las tareas del plan. Este plan debe persistir en un archivo de propiedades. |
| **Principales Parámetros** | **Resultados** |
| 1. Identificador de la tarea del plan | Actividad del plan creada, modificada o eliminada |
| 2. Nombre de la tarea |  |
| 3. Duración de la tarea |  |
| 4. Responsable de la tarea |  |
| 5. % Valor ganado |  |

|  |  |
| --- | --- |
| **Identificador del Requerimiento** | RF3.0 |
| **Nombre del Requerimiento** | Administración del plan de calidad |
| **Actores** | Usuario |
| **Propósito** | Este requisito comprende la creación y modificación del plan de calidad. Este plan de calidad debe persistir en un archivo de propiedades. |
| **Principales Parámetros** | **Resultados** |
| 1. Identificador del plan de calidad | Plan de calidad creado, modificado y persistido en el archivo de propiedades |
| 2. Cantidad de errores inyectados Planeacion |  |
| 3. Cantidad de errores inyectados Diseño |  |
| 4. Cantidad de errores inyectados Análisis |  |
| 5. Cantidad de errores inyectados Pruebas |  |
| 6. Cantidad de errores inyectados Inspección |  |
| 7. Cantidad de errores inyectados Postmorten |  |

|  |  |
| --- | --- |
| **Identificador del Requerimiento** | RF4.0 |
| **Nombre del Requerimiento** | Generación Reporte Productividad |
| **Actores** | Usuario |
| **Propósito** | Este requisito comprende la visualización de las estadísticas grupales e individuales. |
| **Principales Parámetros Presentados** | **Resultados** |
| 1. LOC reales vs LOC estimadas por cada grupo | Reporte con los parámetros presentados. |
| 2. LOC reales vs LOC estimadas por cada integrante del grupo. |  |
| 3. Productividad del grupo |  |
| 4. Productividad de cada integrante del grupo |  |

|  |  |
| --- | --- |
| **Identificador del Requerimiento** | RF5.0 |
| **Nombre del Requerimiento** | Generación Reporte de Ciclos |
| **Actores** | Usuario |
| **Propósito** | Este requisito consiste en la presentación del estado de cada ciclo. Cuando se selecciona una actividad de un ciclo se presenta el estado de esta actividad |
| **Principales Parámetros Presentados** | **Resultados** |
| 1. Identificador de la tarea | Reporte con los parámetros presentados. |
| 2. Descripción de la tarea |  |
| 3. Tiempo planeado |  |
| 4. % valor ganado planeado |  |
| 5. % valor ganado real |  |
| 6. Responsable y rol |  |

* 1. **Diseño Conceptual**

Con base a la definición del requerimiento total del sistema se realiza el siguiente diagrama de clases que sirve como proxy para poder identificar el tamaño y esfuerzo necesario para construir la aplicación deseada.



Ilustración 1. Diseño Conceptual

Donde se tiene que cada una de las clases definidas cumple con las siguientes funcionalidades:

Tabla 2. Definición de Roles

|  |  |  |
| --- | --- | --- |
| **Clase** | **Tipo** | **Funcionalidad** |
| **FramePrincipal** | Presentación | Permite controlar todos los paneles de la aplicación |
| **PanelPersonas** | Presentación | Panel que permite mostrar la información de los integrantes del grupo |
| **PanelPlanCalidad** | Presentación | Panel que permite administrar el plan de calidad del proyecto |
| **PanelPlan** | Presentación | Panel que permite administrar el plan para cada ciclo |
| **PanelReporte** | Presentación | Panel que permite generar el reporte de productividad para cada ciclo |
| **Analizador** | Lógica | Analiza la información almacenada en las anotaciones realizadas para que sean mostradas al usuario |
| **Persona** | Lógica | Clase que permite gestionar la información de cada integrante del grupo |
| **PlanCalidad** | Lógica | Clase que permite gestionar el plan de calidad del sistema |
| **PlanProductoXCiclo** | Lógica | Clase que permite gestionar el plan de producto para cada plan |
| **Proyecto** | Lógica | Clase que permite gestionar la información del proyecto |
| **Interrupción** | Lógica | Clase que permite gestionar las interrupciones encontradas |
| **GestorArchivoPropiedades** | Lógica | Clase abstracta que controla las operaciones básicas del acceso a los archivos de propiedades que almacenan la información del sistema |
| **LogT** | Datos | Anotación para las tareas |
| **LogD** | Datos | Anotación para los defectos |
| **LogInt** | Datos | Anotación para las interrupciones |
| **Loc** | Datos | Anotación para contar el total de líneas de código |

* 1. **Estimación de Tiempo y Esfuerzo**

Tomando como referencia la estimación para desarrollo en C++ de Humphrey[[1]](#footnote-1), mostrada en la siguiente tabla, se realiza la estimación del total de tiempo requerido para el sistema.

Tabla 3. Estimación de LOC en C++

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Categoría | Muy Pequeño | Pequeño | Mediana | Grande | Muy Grande |
| Cálculo | 2,3 | 5,1 | 11,3 | 24,7 | 54,0 |
| Datos | 2,6 | 4,8 | 8,8 | 16,3 | 30,1 |
| I/O | 9,0 | 12,1 | 16,2 | 21,6 | 28,9 |
| Lógica | 7,6 | 11,0 | 16,0 | 23,3 | 33,8 |
| Configuración | 3,8 | 5,0 | 6,6 | 8,5 | 11,1 |
| Texto | 3,8 | 8,0 | 17,1 | 36,4 | 77,7 |

Con base a la tabla anterior y al diseño realizado se realiza la siguiente tabla que permite estimar el tamaño en LOC del sistema:

Tabla 4. Definición de LOC estimados con base al proxy

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CLASE | MÉTODO | CATEGORIA | TAMAÑO | LOC |
| FramePrincipal | MostrarPantalla | I/O | Mediano | 16,2 |
| InicializarComponentes | Lógica | Mediano | 16 |
| PanelPersonas | InicializarComponentes | Lógica | Mediano | 16 |
| GestionarPersona | Lógica | Mediano | 15,98 |
| PanelPlanCalidad | InicializarComponentes | Lógica | Mediano | 16 |
| GestionarPlanCalidad | Lógica | Mediano | 15,98 |
| PanelPlan | InicializarComponentes | Lógica | Mediano | 16 |
| GestionarPlan | Lógica | Mediano | 15,98 |
| PanelReporte | MostrarReportePersonas | I/O | Mediano | 16,2 |
| MostrarReporteGrupo | I/O | Mediano | 16,2 |
| Analizador | AnalizarLogT | Cálculo | Grande | 21,6 |
| AnalizarLogD | Cálculo | Grande | 21,6 |
| AnalizarLogInt | Cálculo | Grande | 21,6 |
| AnalizarLoc | Cálculo | Grande | 21,6 |
| BuscarArchivos | Data | Mediano | 8,8 |
| GenerarReportePersonas | Cálculo | Muy Grande | 54 |
| GenerarReporteGrupo | Cálculo | Muy Grande | 54 |
| Persona | GestionarRol | Lógica | Mediano | 16 |
| MapearDatos | Lógica | Grande | 23,3 |
| PlanCalidad | MapearDatos | Lógica | Grande | 23,3 |
| PlanProductoXCiclo | MapearDatos | Lógica | Grande | 23,3 |
| Proyecto | ListarCiclos | I/O | Mediano | 16,2 |
| MapearDatos | Lógica | Grande | 23,3 |
| Interrupción | MapearDatos | Lógica | Grande | 23,3 |
| GestorArchivoPropiedades | LeerPropiedad | Data | Grande | 16,3 |
| EliminarPropiedad | Data | Grande | 16,3 |
| InsertarPropiedad | Data | Grande | 16,3 |
| ModificarPropiedad | Data | Grande | 16,3 |
| LogT |  |  |  |  |
| LogD |  |  |  |  |
| LogInt |  |  |  |  |
| Loc |  |  |  |  |
|  |  |  | LOC | 577,64 |

* + 1. **Regresión Lineal**

Ya contamos con un valor estimado para el tamaño del proyecto, pero este valor se puede ajustar mas a la realidad si se toman como referencia valores comparativos de LOC estimados contra LOC reales.

Gracias al primer y segundo ciclo del PSP realizado de manera individual por cada uno de los integrantes del grupo se cuenta con los datos comparativos que nos permiten realizar un mejor acercamiento a la realidad, estaos datos son los siguientes:

Tabla 5. Datos iníciales del grupo de trabajo

|  |  |  |  |
| --- | --- | --- | --- |
|  | **LOC**  **Estimados** | **LOC**  **Real** | **Productividad**  **LOC/Hora** |
|  | 110 | 182 | 19,77 |
|  | 150 | 289 | 16,00 |
|  | 170 | 271 | 18,00 |
|  | 180 | 332 | 18,60 |
|  | 300 | 492 | 35,00 |
| **Total** | 910 | 1566 | 88,77 |
| **Promedio** | 182 | 313,2 | 22,19 |

Se aplica una regresión lineal teniendo en cuenta las siguientes formulas:



Tomando como referencia los datos de la tabla anterior tenemos:

 

Por tanto despejando los valores de  se obtiene:



Y al despejar  se obtiene:



Finalmente con la formula de la regresión lineal se obtiene un dato más cercano a la realidad:



Esta información se puede ver en la siguiente gráfica:

Ilustración 2. Regresión Lineal

En la gráfica anterior se puede observar que los datos históricos permiten generar la regresión lineal que indica que si el estimado es de 577,64 LOC lo más posible es que realmente se empleen 933.34 LOC, es decir aproximadamente 934 LOC

* + 1. **Estimación de Tiempo**

Con base a los resultados obtenidos en la sección anterior tenemos que para el equipo de trabajo se puede esperar lo siguiente:

**LOC Proyectadas:** 934 LOC

**Productividad Promedio:** 22,19 LOC/Hora

Por tanto tendríamos que:

**Tiempo total estimado = LOC Proyectadas / Productividad Promedio**

**Tiempo total estimado =** (934/LOC) / (22,19 LOC/Hora)

**Tiempo total estimado =** 42,09 Horas

* 1. **Plan de trabajo**

SANDRA y Carlos Descripción Breve

* 1. **Requerimientos planeados por ciclo**

SANDRA y Carlos Descripción Breve

1. **Definición Primer ciclo**
   1. **Objetivo**

Para el ciclo 1, se requiere realizar la configuración inicial de la solución, esto es, adaptar un producto existente que ya realiza la lectura de anotaciones.

Se requiere modificar las anotaciones para se pueda registrar además el ciclo del desarrollo y el responsable de la tarea.

El desarrollo de la nueva funcionalidad del producto se considera en ciclos posteriores. Algunos artefactos generados en esta fase son:

* Arquitectura de la solución
* Guía de implementación
* Prototipos
  1. **Plan de Trabajo**

Descripción Breve

* 1. **Entregables**

3.3.1 Arquitectura de la solución

Para el proceso de construcción del producto se organizará la solución por paquetes de la siguiente manera:



Paquete de presentación: Contiene las clases que se encargan de capturar y/o presentar la información al usuario.

Paquete de Lógica de Negocio: Contiene las clases que modelan el negocio.

Paquete de Pruebas Unitarias: Contiene las clases necesarias para realizar las pruebas unitarias de toda la aplicación.

3.3.2 Guía de implementación

Para el desarrollo del producto, los ingenieros de desarrollo involucrados deberán seguir un estándar común. A continuación se presentan algunos ejemplos tipo.

1. Ejemplo definición de una clase

/\*\*

\* Es la clase que representa a un perro. <br>

\*/

**public** **class** Perro

{

}

1. Ejemplo definición de un método con anotaciones

/\*\*

\* Compara dos perros según el nombre. <br>

\* **@param** p es el perro contra el que se está comparando - p !=null

\* **@return** Retorna 0 si los perros tienen el mismo nombre. <br>

\* Retorna -1 si el perro p tiene una valor "MAYOR" para el nombre. <br>

\* Retorna 1 si el perro p tiene una valor "MENOR" para el nombre. <br>

\*/

@LogsT  
 ({

@LogT(taskId = "30", date = "21/02/2011 00:45", min = 10, cicloId=1, responsableId=3)

})

@LogsInt

({

LogInt(date=”21/02/2011 00:52”, min=2, cicloId=1, responsableId=3, intId=1 )

})

@Locs

({

@Loc(size=”20”, cicloId=1, responsableId=3)

})

@LogsD

({

@LogD(date=”21/02/2011 01:45”, stage=”codificación”, cicloId=1, responsableId=3)

})

**public** **int** compararPorNombre( Perro p )

{

**//Definición del método**

}

Anotación LogT: Registra el tiempo tomado para ejecutar una tarea determinada. Se registra la fecha, tiempo en minutos, identificador del ciclo, identificador de la tarea realizada y id del responsable que ejecuta la tarea.

Anotación LogInt: Registra el tiempo de las interrupciones que se generaron durante la realización de una tarea. Requiere fecha, tiempo en minutos, identificador del ciclo y responsable de la interrupción.

Anotación LogD: Registra los defectos encontrados en el producto. Registra fecha, fase, identificador del ciclo e identificador del responsable.

Anotación Loc: Registra el número de líneas de código que una persona generó durante la ejecución de una tarea.

1. **PostMortem Primer Ciclo**

Descripción Breve

1. A Discipline for Software Engineering, page 117, by Watts S. Humphrey, Addison Wesley Publishing Co., 1995. [↑](#footnote-ref-1)