

MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA  
CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES



German Geovanny Garzón Bravo

Pablo Felipe Miranda Caicedo

Universidad del Cauca

Facultad De Ingeniería Electrónica Y Telecomunicaciones

Departamento De Ingeniera De Sistemas

Grupo IDIS

Popayán 2013

MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA  
CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

German Geovanny Garzón Bravo

Pablo Felipe Miranda Caicedo

Monografía para optar al título de  
Ingeniero de Sistemas

Director: Phd. Francisco J. Pino

Codirector: Ing. Carlos A. Ardila

Universidad del Cauca

Facultad De Ingeniería Electrónica Y Telecomunicaciones

Departamento De Ingeniería De Sistemas

Grupo IDIS

Popayán 2013

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

## TABLA DE CONTENIDO

<b>TABLA DE CONTENIDO</b>	<b>III</b>
<b>LISTA DE FIGURAS</b>	<b>VII</b>
<b>LISTA DE GRÁFICOS</b>	<b>VII</b>
<b>LISTA DE TABLAS</b>	<b>VIII</b>
<b>CAPÍTULO I: INTRODUCCIÓN</b>	<b>10</b>
1.1 OBJETIVOS	12
1.1.1 OBJETIVO GENERAL	12
1.1.2 OBJETIVO ESPECÍFICOS	12
1.2 ESTRATEGIA DE INVESTIGACIÓN UTILIZADA	13
1.3 ESTRUCTURA DEL DOCUMENTO	14
<b>CAPÍTULO II: MARCO TEÓRICO Y ESTADO DEL ARTE</b>	<b>16</b>
2.1 MARCO TEÓRICO.	16
2.1.1 NORMA ISO/IEC 12207 Y EL PROCESO DE CONSTRUCCIÓN DE SOFTWARE.	16
2.1.1.1 ISO/IEC 12207.	16
2.1.1.2 PROCESO DE CONSTRUCCIÓN DE SOFTWARE.	16
2.1.1.2.1 PROPÓSITO.	17
2.1.1.2.2 RESULTADOS.	17
2.1.1.2.3 ACTIVIDADES Y TAREAS.	17
2.1.1.2.3.1 ACTIVIDAD DE CONSTRUCCIÓN DE SOFTWARE.	17
2.1.1.2.3.2 TAREAS.	17
2.1.2 SPC (STATISTICAL PROCESS CONTROL).	18
2.1.2.1 DEFINICIÓN.	19
2.1.2.2 TÉCNICA DE GRÁFICO DE CONTROL.	19
2.1.2.2.1 TIPOS DE GRÁFICOS DE CONTROL.	21
2.1.2.2.1.1 GRÁFICOS DE CONTROL POR VARIABLES.	21
2.1.2.2.1.2 GRÁFICOS DE CONTROL POR ATRIBUTOS	21
2.1.2.2.1.3 CONSTRUCCIÓN DE UN GRÁFICO DE CONTROL	22
2.1.3 MEDICIÓN SOFTWARE.	25
2.1.3.1 MEDICIÓN.	25
2.1.3.2 MEDIDA.	25
2.1.3.3 MÉTRICA.	25
2.1.3.3.1 MÉTRICAS SOFTWARE.	26
2.1.4 VSEs (VERY SMALL ENTERPRISES).	26
2.1.4.1 DEFINICIÓN.	26
2.1.4.2 IMPORTANCIA.	28

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

2.2	ESTADO DEL ARTE.	31
2.2.1	MÉTRICAS Y MEDICIÓN DEL SOFTWARE	31
2.2.1.1	ONTOLOGÍA DE LA MEDICIÓN [26].	31
2.2.1.2	SPDW +: UN ENFOQUE PARA LA CAPTURA DE MÉTRICAS DE CALIDAD EN ENTORNOS DE DESARROLLO DE SOFTWARE [33].	31
2.2.1.3	UNA REVISIÓN SISTEMÁTICA DE MEDICIÓN EN LA INGENIERÍA SOFTWARE: ESTADO DEL ARTE EN LAS MEDIDAS [34].	32
2.2.2	SPC EN LA INDUSTRIA DEL SOFTWARE.	33
2.2.2.1	GESTIONANDO LA MEJORA DE PROCESOS SOFTWARE (SPI) A TRAVÉS DEL CONTROL ESTADÍSTICO DE PROCESOS (SPC) [10].	33
2.2.2.2	MEDIDAS DE CALIDAD EN PROCESO, PRODUCTO Y MANTENIMIENTO, APLICADAS AL CONTROL ESTADÍSTICO DE PROCESOS [23].	34
2.2.2.3	UTILIZACIÓN DE CONTROL ESTADÍSTICO DE PROCESOS (SPC) EN LAS ORGANIZACIONES DE SOFTWARE EMERGENTES: ERRORES Y SUGERENCIAS [8].	35
2.2.2.4	EXPERIENCIAS DE LA APLICACIÓN DE TÉCNICAS SPC A LOS PROCESOS DE DESARROLLO DE SOFTWARE [19].	36
2.2.2.5	EVALUANDO LA IDONEIDAD DE UN REPOSITORIO DE MEDICIÓN PARA EL CONTROL ESTADÍSTICO DE PROCESOS [11].	37
2.2.3	DISCUSIÓN SOBRE TRABAJOS RELACIONADOS.	38
2.2.4	APORTES.	39

## **CAPÍTULO III: CONJUNTO DE MÉTRICAS DE CONSTRUCCIÓN PARA SPC EN VSE'S** **40**

3.1	FASES DEL CICLO METODOLÓGICO.	40
3.2	IDENTIFICACIÓN DEL PROBLEMA.	42
3.2.1	IDENTIFICAR Y DETALLAR EL PROCESO DE CONSTRUCCIÓN DE SOFTWARE.	42
3.2.2	IDENTIFICAR LAS MÉTRICAS UTILIZADAS EN PROCESO DE CONSTRUCCIÓN.	46
3.2.3	IDENTIFICAR LAS PRÁCTICAS COMUNES DE LAS VSEs.	48
3.2.3.1	CARACTERÍSTICAS DE LAS VSEs.	48
3.2.3.2	PRÁCTICAS DE IMPLEMENTACIÓN DE LAS VSEs.	48
3.2.4	IDENTIFICAR LAS MÉTRICAS UTILIZADAS EN CASOS DE ESTUDIO APLICANDO LA TÉCNICA SPC EN EL DESARROLLO DE SOFTWARE.	49
3.3	PLAN DE ACCIÓN.	52
3.3.1	RECOPIRAR LAS MÉTRICAS EMPLEADAS EN LAS TAREAS ESPECÍFICAS DEL PROCESO DE CONSTRUCCIÓN.	52
3.3.2	CLASIFICAR LAS PRÁCTICAS DE LAS VSEs EMPLEADAS EN LAS TAREAS DEL PROCESO DE CONSTRUCCIÓN.	55
3.3.3	FILTRAR LAS MÉTRICAS QUE CUMPLEN CON LAS PRÁCTICAS DE LAS VSEs EJECUTADAS EN EL PROCESO DE CONSTRUCCIÓN.	61
3.3.3.1	FILTRADO DE MÉTRICAS.	61
3.3.4	RECOLECCIÓN Y CLASIFICACIÓN DE LAS MÉTRICAS QUE PUEDAN SER APLICADAS A LA TÉCNICA SPC EN VSEs UTILIZANDO LOS CRITERIOS ESTABLECIDOS POR IESRM.	64
3.3.4.1	IESMR (INSTRUMENT FOR EVALUATING THE SUITABILITY OF MEASUREMENT REPOSITORIES).	64
3.3.4.2	INSTRUMENTO PARA LA VALIDACIÓN DE LA IDONEIDAD DE MÉTRICAS PARA SPC (IVIM).	67
3.3.4.3	VERIFICAR LA UTILIDAD DEL IVIM.	69

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

3.4	RECOPIACIÓN DE DATOS.	70
3.4.1	LISTAR EL CONJUNTO DE MÉTRICAS ADECUADAS PARA LA APLICACIÓN DE LA TÉCNICA SPC EN VSEs, BASADOS EN LAS PRÁCTICAS DE LAS VSEs EJECUTADAS EN LAS TAREAS DEL PROCESO DE CONSTRUCCIÓN.	70
3.4.1.1	EVALUACIÓN DE LAS MÉTRICAS CON EL IVIM.	70
3.4.1.2	CONJUNTO DE MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN APROPIADAS PARA LA APLICACIÓN DE SPC EN PEQUEÑAS ORGANIZACIONES.	72
3.4.1.2.1	CONJUNTO PRELIMINAR DE MÉTRICAS.	72
3.4.1.2.2	GRÁFICOS DE CONTROL ADECUADOS PARA EL CONJUNTO DE MÉTRICAS PROPUESTO.	72
3.4.2	IDENTIFICAR LAS NECESIDADES DE INFORMACIÓN QUE SATISFACEN LAS MÉTRICAS.	74

## **CAPÍTULO IV: PROTOTIPO SOFTWARE** **79**

4.1	PERSPECTIVA DEL PRODUCTO.	79
4.1.1	METODOLOGÍA DEL PRODUCTO.	80
4.1.1.1	FASES DE LA METODOLOGÍA.	80
4.1.1.1.1	FASE DE EXPLORACIÓN.	80
4.1.1.1.2	FASE DE PLANEACIÓN.	80
4.1.1.1.3	FASE DE ITERACIONES.	80
4.1.1.1.4	FASE DE PRUEBAS.	81
4.1.2	ACTIVIDADES DE TRABAJO.	81
4.1.3	ASIGNACIÓN DE ROLES.	82
4.1.4	CRONOGRAMA.	82
4.2	DESARROLLO DE LAS FASES DE LA METODOLOGÍA.	83
4.2.1	FASE DE EXPLORACIÓN.	83
4.2.1.1	SELECCIÓN DE UN FRAMEWORK DE DESARROLLO.	83
4.2.1.2	CARACTERÍSTICAS DE LOS USUARIOS.	83
4.2.1.3	REQUISITOS DEL BANCO DE DATOS LÓGICOS.	84
4.2.1.4	RESTRICCIONES DE INTEGRIDAD.	84
4.2.1.5	RESTRICCIONES GENERALES.	84
4.2.1.6	RESTRICCIONES DEL PRODUCTO.	85
4.2.2	FASE DE PLANEACIÓN.	86
4.2.3	ITERACIÓN I (ARQUITECTURA).	87
4.2.3.1	ARQUITECTURA SOFTWARE.	88
4.2.4	ITERACIÓN II (MODULO 1).	89
4.2.5	ITERACIÓN III (MODULO 2).	90

## **CAPÍTULO V: EVALUACIÓN DEL PROTOTIPO SOFTWARE** **91**

5.1	PROCESO EMPLEADO PARA LA APLICACIÓN DE FOCUS GROUP EN LA EVALUACIÓN DEL PROTOTIPO SOFTWARE.	91
5.1.1	EVALUACIÓN DEL PROTOTIPO SOFTWARE CON FOCUS GROUP.	92
5.1.1.1	FASE DE PLANEAMIENTO DE LA INVESTIGACIÓN.	92
5.1.1.1.1	DEFINICIÓN DEL PROBLEMA DE INVESTIGACIÓN.	92
5.1.1.1.2	PREPARACIÓN DE MATERIALES Y MÉTODOS, A CUMPLIR POR PARTE DEL GRUPO INVESTIGADOR.	92

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

5.1.1.1.2.1	DEFINICIÓN DE ESTRUCTURA.	93
5.1.1.1.2.2	DEFINICIÓN DE INSTRUMENTOS, MATERIALES Y MÉTODOS A SER EMPLEADOS.	94
5.1.1.1.2.3	DEFINICIÓN DE MÉTODOS DE CAPTURA Y REGISTRO DE INFORMACIÓN DERIVADA DEL DEBATE.	94
5.1.1.1.2.4	DEFINICIÓN DE MÉTODOS DE ANÁLISIS DE INFORMACIÓN PARA GENERAR EL PROCESAMIENTO DE LO GENERADO EN EL DEBATE.	94
5.1.1.2	FASE DE DEFINICIÓN DE GRUPOS DE DISCUSIÓN	96
5.1.1.2.1	SELECCIÓN DE PARTICIPANTES	96
5.1.1.3	FASE DE CONDUCCIÓN DE LA SESIÓN DE DEBATE.	96
5.1.1.3.1	SECUENCIA BÁSICA.	96
5.1.1.3.2	CAPTURA DE INFORMACIÓN.	98
5.1.1.4	FASE DE ANÁLISIS DE INFORMACIÓN Y REPORTE DE RESULTADOS.	98
5.1.1.4.1	ANÁLISIS DE INFORMACIÓN.	98
5.1.1.4.1.1	ANÁLISIS DE ENCUESTAS Y LOS ARCHIVOS DE AUDIO.	98
5.1.1.4.1.2	ESTABLECIMIENTO DE ASPECTOS POSITIVOS, NEGATIVOS Y/O OBSERVACIONES DEL PROTOTIPO.	104
5.1.1.4.1.3	OPORTUNIDADES DE MEJORA CONSIDERADAS PARA EL PROTOTIPO.	104
5.1.1.5	PRODUCTOS DE TRABAJO.	106
5.2	ANÁLISIS DE LA EVALUACIÓN.	107

## **CAPÍTULO VI: CONCLUSIONES Y TRABAJO FUTURO** **108**

6.1	SUMARIO.	108
6.2	CONCLUSIONES.	109
6.3	REFLEXIÓN.	110
6.4	TRABAJO FUTURO.	112

## **BIBLIOGRAFÍA** **114**

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

## LISTA DE FIGURAS

<b>Figura 1.</b> Estrategia de Investigación.	13
<b>Figura 2.</b> Ejemplo de un gráfico de control.	20
<b>Figura 3.</b> Abstracción del ciclo de investigación metodológico.	41
<b>Figura 4.</b> Un proceso de construcción de software.	45
<b>Figura 5.</b> MOCKUP de la Historia de usuario "Iniciar Terminar".	86
<b>Figura 6.</b> Historia de Usuario "Iniciar Terminal".	86
<b>Figura 7.</b> Arquitectura del sistema CMSPC.	89
<b>Figura 8.</b> Modulo web de gestión de datos.	90
<b>Figura 9.</b> Terminal para captura de datos.	90

## LISTA DE GRÁFICOS

<b>Gráfico 1.</b> Resultado de las preguntas realizadas en el Ejemplo de uso del prototipo CMSPC.	13
<b>Gráfico 2.</b> Resultado de la pregunta 9 del ejemplo de uso del prototipo.	20
<b>Gráfico 3.</b> Resultado de la pregunta 13 del ejemplo de uso del prototipo.	41
<b>Gráfico 4.</b> Completitud de la implementación funcional.	45

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

## LISTA DE TABLAS

<b>Tabla 1.</b> Constantes para los gráficos X-barra y R.	22
<b>Tabla 2.</b> Tipos de gráfico de variable.	23
<b>Tabla 3.</b> Tipos de gráfico de atributo.	24
<b>Tabla 4.</b> Cuadro comparativo de la capacidad en número de empleados de las VSEs, Mipymes y la gran empresa.	27
<b>Tabla 5.</b> Actividad principal de las empresas.	29
<b>Tabla 6.</b> Líneas de negocios seleccionadas por número de empresas.	30
<b>Tabla 7.</b> Exportaciones.	30
<b>Tabla 8.</b> Importaciones.	30
<b>Tabla 9.</b> Número de empleados por empresa.	30
<b>Tabla 10.</b> Fases del procedimiento sistémico lógico empleado en el desarrollo del presente trabajo de investigación.	41
<b>Tabla 11.</b> Tareas del proceso de construcción de software definidas en la normativa ISO/IEC 12207.	42
<b>Tabla 12.</b> Proceso de construcción software en detalle (extraído de norma ISO/IEC 15504-5).	44
<b>Tabla 13.</b> Fuentes de información primarias.	47
<b>Tabla 14.</b> Listado de fuentes más relevantes para el presente trabajo de investigación.	47
<b>Tabla 15.</b> Prácticas más ejecutadas por las VSEs.	49
<b>Tabla 16.</b> Métricas software utilizadas en la aplicación de SPC con gráficos de control.	51
<b>Tabla 17.</b> Listado de métricas de la revisión sistemática.	55
<b>Tabla 18.</b> Prácticas de las VSEs y su relación con las tareas del proceso de construcción.	60
<b>Tabla 19.</b> Correlación entre las tareas del proceso de construcción de software y las prácticas de las VSEs.	60
<b>Tabla 20.</b> Otra perspectiva de la correlación entre las tareas del proceso de construcción de software y las prácticas de las VSEs.	61
<b>Tabla 21.</b> Métricas del proceso de construcción y su relación con las Prácticas de las VSEs.	62
<b>Tabla 22.</b> Relación práctica, proceso de construcción y métricas.	63
<b>Tabla 23.</b> Otra perspectiva de la relación práctica, proceso de construcción y métricas.	63
<b>Tabla 24.</b> Características IESRM para evaluar la idoneidad SPC de una métrica.	65
<b>Tabla 25.</b> Estudios SPC analizados.	66
<b>Tabla 26.</b> Instrumento para la Validación de la Idoneidad de Métricas para SPC (IVIM).	68
<b>Tabla 27.</b> Resultados de la evaluación con el instrumento IVIM a las métricas ya aplicadas en SPC.	69
<b>Tabla 28.</b> Resultado final de la evaluación con el IVIM al conjunto de métricas software.	72
<b>Tabla 29.</b> Conjunto preliminar de Métricas apropiadas para SPC.	72
<b>Tabla 30.</b> Métricas relacionadas con el proceso de construcción de software que pueden ser utilizadas en SPC para las VSEs.	73

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>Tabla 31.</b> Gráficos de control apropiados al conjunto de métricas propuesto.	74
<b>Tabla 32.</b> Datos relacionados con la completitud de la métrica Valor Ganado.	75
<b>Tabla 33.</b> Datos relacionados con la completitud de la métrica Densidad de Defectos.	76
<b>Tabla 34.</b> Datos relacionados con la completitud de la métrica productividad.	77
<b>Tabla 35.</b> Datos relacionados con la completitud de la métrica Porcentaje de Retrabajo.	78
<b>Tabla 36.</b> Actividades del desarrollo del prototipo.	81
<b>Tabla 37.</b> Roles de la metodología XP.	82
<b>Tabla 38.</b> Asignación de Roles para el equipo de desarrollo.	82
<b>Tabla 39.</b> Cronograma.	83
<b>Tabla 40.</b> Criterios de aceptación de desarrollo.	87
<b>Tabla 41.</b> Actividades para las iteraciones XP.	88
<b>Tabla 42.</b> Tiempos de ejecución de las actividades en la sesión de focus group.	97
<b>Tabla 43.</b> Relación de las preguntas del Ejemplo de uso del prototipo con las historias de usuario definidas.	99
<b>Tabla 44.</b> Resultado de las evaluaciones de las HU en el ejemplo de uso del prototipo.	103
<b>Tabla 45.</b> Aspectos positivos, aspectos negativos y/o observaciones consideradas de las preguntas generales.	105

# CAPÍTULO I

## INTRODUCCIÓN

---

La industria del software nacional e internacional está conformada en su gran mayoría por VSEs<sup>1</sup> (Very Small Entities) [1] [2]. Estas organizaciones desarrollan productos significativos y para construirlos necesitan prácticas eficientes de ingeniería que sean adecuadas a su tamaño y tipo de negocio [2]. En este sentido, es importante apoyar a estas organizaciones en la gestión (control y mejora) del proceso y producto software, ya que este aspecto es un factor determinante de la calidad del producto final y la productividad del proceso de desarrollo [3]. Sin embargo, gestionar los procesos software, entre ellos el de construcción de software, es una labor difícil de llevar a cabo en las VSEs [4]. En estas empresas éste es el proceso de producción fundamental, el más ejecutado y que como mínimo se lleva a cabo, en el cual: (i) se concentra una gran cantidad de esfuerzo, y (ii) se refleja el quehacer de la industria del software [2]. El propósito del proceso de construcción es crear unidades de software ejecutables que reflejan apropiadamente el diseño del software y en este proceso se realizan tareas específicas relacionadas con codificación [5].

Es importante resaltar que para gestionar el proceso de construcción una responsabilidad clave es la medición del mismo [6], sin embargo la gran mayoría de las VSEs no llevan a cabo actividades de medición de los procesos [1] (incluyendo el proceso de construcción). Según [7] en cualquier rama de la ingeniería y específicamente en la Ingeniería de software se debe medir. En este sentido, ésta es una actividad necesaria para las VSEs que desarrollan software [1] especialmente si consideran la aplicación de técnicas cuantitativas como el SPC (Statistical Process Control) para la gestión de sus procesos [8] (incluyendo el de construcción de software). La técnica SPC es una manera eficaz de identificar la estabilidad, o la falta de ella, en el proceso mediante el uso de gráficos de control y sus interpretaciones [9, 10]. La variación en los gráficos de control debe ser investigada y tratada a través de acciones de mejora cuyo objetivo es estabilizar el proceso. Esta técnica tienen su origen en la industria, pero en los últimos años ha sido aplicada al contexto del software demostrando gran capacidad de gestión en los procesos software, mejorando la estabilidad y rendimiento de los mismos.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

SPC requiere de métricas con características especiales, y los datos de estas métricas esconden algunas características inherentes del proceso que son únicas para cada organización de software [8]. Sin embargo, no se han identificado métricas adecuadas que puedan ser utilizadas para SPC en el proceso de construcción de software de las VSEs [2], debido en gran parte a la incompleta definición de las métricas. En [11] y [12] se proponen estrategias que permiten evaluar si una métrica es adecuada o no para SPC mediante la verificación de criterios definidos para tal fin. Sin embargo, estas propuestas carecen de características importantes relacionadas con las pequeñas organizaciones, por lo cual se hace necesario estructurar un instrumento de validación de métricas para SPC en este contexto, con el fin de apoyar la obtención de un conjunto de métricas relacionadas con el proceso de construcción de software que puedan ser utilizadas para el control estadístico de procesos en pequeñas organizaciones.

Un conjunto de métricas apropiadas para SPC del proceso de construcción de software y una herramienta software que apoye la captura de los datos (medidas) de dichas métricas podrían apoyar a las VSEs en la implementación del control estadístico de dicho proceso. Esto permitiría disminuir la variabilidad de los proyectos al contar con un proceso estable que tiene un comportamiento iterativo y predecible logrado a partir de las actividades de monitoreo y evaluación del proceso que permiten visualizar su evolución y resultados en el tiempo. Dicha visualización resalta las variaciones de rendimiento del proceso y ayuda a la toma de decisiones oportuna y rápida. Lo anterior podría influir positivamente en la competitividad de este tipo de organizaciones apoyando la elaboración de planes realistas de mejora del proceso de construcción e incrementando la calidad del producto [10].

Bajo este contexto es importante determinar el conjunto de métricas del proceso de construcción de software que sean adecuadas para SPC y que puedan ser aplicadas en VSEs de la industria del software. El alcance de este trabajo es identificar y especificar un conjunto de métricas que pueden ser útiles para llevar a cabo control estadístico del proceso de construcción de las VSEs. De acuerdo al atributo de medición del proceso (relacionado con la capacidad de procesos) del estándar internacional ISO 15504 [13] la completitud de las métricas está dada en términos de: (i) la definición de las métricas, (ii) la captura de sus datos (medidas) y (iii) la satisfacción de una necesidad de información de la organización.

Es importante resaltar que la evaluación del conjunto de métricas identificadas mediante la aplicación de un caso de estudio de SPC en una VSE está fuera del alcance del presente proyecto. Actualmente una metodología para SPC en VSEs

está siendo desarrollada en un proyecto de Maestría en Computación y el trabajo de esta tesis de pregrado está enmarcado al interior de dicho proyecto.

En este sentido, el presente trabajo de investigación trata de responder a la pregunta de investigación:

¿Qué métricas son adecuadas para gestionar el proceso de construcción desde un enfoque SPC en VSEs?

## **1.1 Objetivos**

Los objetivos del presente trabajo de investigación se presentan a continuación:

### **1.1.1 Objetivo General**

Proponer un conjunto de métricas relacionadas con el proceso de construcción de software que puedan ser utilizadas para el control estadístico de procesos en pequeñas organizaciones.

### **1.1.2 Objetivo Específicos**

- Determinar un grupo de métricas del proceso de construcción de software, apropiadas para el uso en control estadístico de proceso y adecuadas para pequeñas organizaciones.
- Desarrollar un prototipo software que apoye la aplicación de las técnicas de levantamiento de métricas establecidas en el proceso de construcción de software.
- Evaluar el prototipo software que apoya la aplicación de las técnicas para el levantamiento de las métricas mediante juicio de expertos desde el punto de vista de su funcionalidad<sup>1</sup>.

Con el cumplimiento de estos objetivos se pretende aportar en:

Estructurar un nuevo instrumento que permita hacer la evaluación de la idoneidad de una métrica para su uso en SPC dentro del contexto de las pequeñas organizaciones.

Proponer un conjunto de métricas del proceso de construcción de software que puedan ser utilizadas en la aplicación de control estadístico de procesos en las VSEs.

Desarrollar un prototipo que soporte la recolección de las medidas relacionadas con las métricas identificadas en el proceso de construcción software de las VSEs,

---

<sup>1</sup> Funcionalidad: el grado con el cual el producto software provee las funciones que cumplen las necesidades establecidas cuando el software es usado bajo condiciones específicas. (ISO/IE 25010).

contribuyendo así a la mejora del proceso y obteniendo un producto de buena calidad en empresas que no disponen de recursos para tal fin.

## 1.2 Estrategia de Investigación Utilizada

Para la definición y desarrollo del presente proyecto de investigación, se estructura una estrategia de investigación (ver Figura 1) basada en la metodología de investigación AR (investigación - acción) bifurcado [14] [15]. La estrategia parte de un ciclo de investigación inicial donde se identifican tres problemas de tipo conceptual, metodológico y técnico. Estos problemas permiten dividir el trabajo en tres ciclos de investigación independientes: ciclo conceptual, ciclo metodológico y ciclo técnico. El ciclo técnico se relaciona con un ciclo de evaluación del prototipo para generar la herramienta software final que da soporte al objetivo del trabajo de investigación.

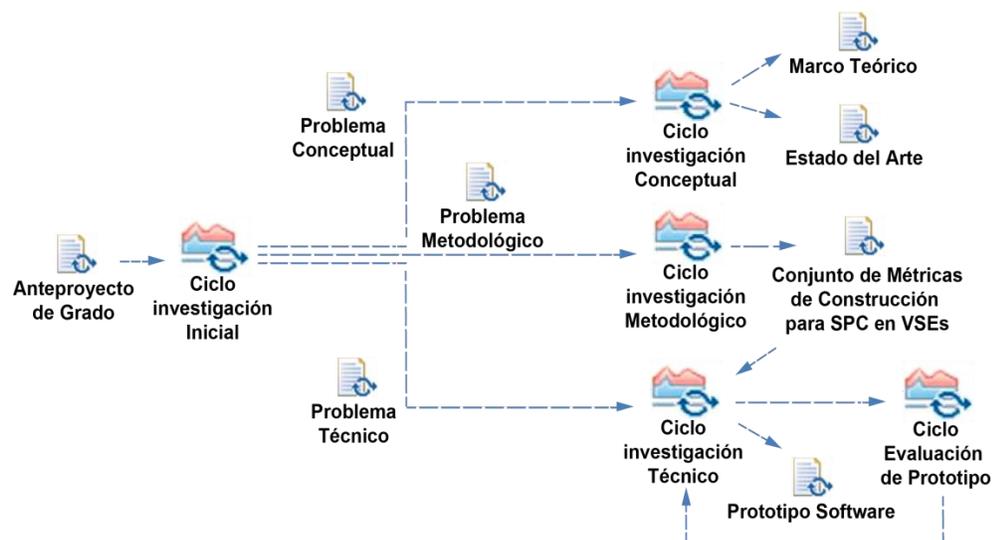


Figura 1. Estrategia de Investigación.

El problema conceptual evidencia la necesidad de información vital para contextualizar el presente proyecto. Por lo anterior se ejecuta un ciclo de investigación conceptual que está conformado por dos fases: identificación del problema y recopilación de los datos, de las cuales se obtiene la información pertinente que permite organizar el marco conceptual y el estado del arte. En la fase de identificación del problema se revisa el estado actual de la técnica SPC en la industria del software, así como el estado actual de las métricas y la medición software. En la fase de recopilación de datos se realiza una discusión de los trabajos relacionados y se organiza la definición teórica.

Por otra parte, la estrategia utilizada para obtener el conjunto de métricas apropiadas para SPC en las VSEs, esta descrita por el ciclo de investigación metodológico. Este ciclo consta de tres fases: identificación del problema, plan de acción y recopilación de los datos. En la primera fase se detalla el proceso de construcción de software y las métricas relacionadas con este proceso, se identifican las practicas software más comunes en las pequeñas organizaciones y se analizan las métricas utilizadas en la aplicación de SPC en casos de estudio dentro de la industria software. En la segunda fase se recopilan las métricas software del proceso de construcción y se clasifican las prácticas de las pequeñas organizaciones empleadas en las tareas del proceso de construcción para crear un filtro que permite obtener métricas ligadas a las tareas del proceso de construcción y que se pueden emplear en las practicas software de las pequeñas organizaciones. De [11] se extrae un instrumento (IESRM) que permite evaluar la idoneidad de una métrica para su uso en la técnica SPC, de este se organiza y estructura un nuevo Instrumento para la Validación de la Idoneidad de Métricas para SPC (IVIM) que se usa para evaluar las métricas filtradas. En la tercera fase se lista el conjunto de métricas adecuadas para la aplicación de la técnica SPC en pequeñas organizaciones y se identifican las necesidades de información que satisfacen dichas métricas.

Por último, el ciclo de investigación técnico incorpora la metodología de desarrollo empleada para la construcción del prototipo software. Este desarrollo es guiado por la metodología ágil XP (Extreme programming) [16], en donde la construcción del prototipo software se hace de forma iterativa e incremental. En cada iteración se desarrollan y se verifican las funcionalidades necesarias para la captura las métricas propuestas en el presente trabajo de investigación. La propuesta solo incluyó las fases de: Exploración, Planeación, Diseño, Codificación y Pruebas, ya que no se hace mantenimiento del prototipo software. Además, se realiza un ciclo de validación del prototipo software que permite evaluar el prototipo empleando el método de focus-group [17]. Este método es un enfoque rentable, rápido y flexible que puede ser usado en distintas fases y tipos de investigación. Permite obtener información y experiencias de profesionales en ingeniería de software y usuarios de la aplicación. El método focus-group se compone de discusiones planeadas y diseñadas cuidadosamente para obtener información relevante de las percepciones de los miembros del grupo. Los miembros son seleccionados en base a sus características individuales con relación al tema de la sesión [17].

### **1.3 Estructura del Documento**

A continuación se describe la manera en la que se organiza el trabajo desarrollado en el presente documento.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

En el *Capítulo II – Marco Teórico y Estado del Arte*, se ejecuta el ciclo de investigación conceptual con el que se obtienen los conceptos considerados significativos para la realización del marco conceptual, tales conceptos están relacionados con: la definición del proceso de construcción y su normativa ISO/IEC 12207, SPC y la técnica de gráficos de control, medición y métricas software, además de la definición e importancia de las VSEs en la industria del software. También se estructura el estado del arte, en el que se presentan los trabajos considerados relevantes por su estrecha relación con los temas: métricas y medición del software, y SPC en la industria del software, tratados por el presente trabajo de investigación.

En el *Capítulo III - Conjunto de Métricas de Construcción para SPC en VSEs*, se presenta la ejecución de un procedimiento sistémico lógico, enmarcado dentro del ciclo de investigación metodológico, para la identificación de las métricas relacionadas con el proceso de construcción de software y que puedan aplicarse al control estadístico de procesos en las VSEs. Este procedimiento busca responder a la pregunta de investigación del presente trabajo de investigación ¿Qué métricas son adecuadas para gestionar el proceso de construcción desde un enfoque SPC en VSEs?

El *Capítulo IV – Prototipo Software*, se presenta la ejecución del ciclo de investigación técnico, en el que se lleva a cabo la metodología de desarrollo Ágil XP. Aquí se muestra la aplicación de la metodología XP empleada para la construcción del prototipo software que soporta la captura de los datos del conjunto de métricas propuesto.

En el *Capítulo V – Evaluación del Prototipo Software*, se presenta un ciclo de evaluación del prototipo software que permite evaluar el prototipo empleando el método de focus-group. Aquí se muestra la planificación, diseño y resultados del trabajo realizado con el fin de evaluar el prototipo software enfocado en la característica de calidad de la funcionalidad.

Al final, en el *Capítulo VI - Conclusiones y Trabajo futuro*, se expresan las conclusiones del trabajo realizado y se describen algunas propuestas para el trabajo futuro.

# CAPÍTULO II

## MARCO TEÓRICO Y ESTADO DEL ARTE

---

En busca de una clara comprensión y entendimiento del proyecto es necesario detallar algunos de los trabajos, conceptos y normativas usadas en el desarrollo del mismo. Este capítulo enfoca todos los esfuerzos en estructurar: el *marco teórico* en donde se describen cada uno de los conceptos que se consideran de suma importancia para la comprensión del presente trabajo de investigación, y del *estado del arte* que resume los trabajos que inciden de manera determinante en la comprensión y desarrollo del mismo.

Inicialmente se presenta el marco teórico en donde se describen conceptos relacionados con la definición del proceso de construcción y su normativa ISO/IEC 12207, SPC y la técnica de gráficos de control, medición y métricas software, y la definición e importancia de las VSEs en la industria del software. Después se estructura el estado del arte, en el que se presentan los trabajos considerados relevantes para los temas sobre métricas, medición y SPC en la industria del software.

### **2.1 Marco Teórico.**

#### **2.1.1 Norma ISO/IEC 12207 y el proceso de construcción de software.**

##### **2.1.1.1 ISO/IEC 12207.**

La norma ISO/IEC 12207 define un marco común para los procesos de ciclo de vida del software [5], por esto es usada como referente para especificar el proceso de construcción del software dentro del que hacer de esta industria. La normativa está conformada por procesos, actividades y tareas que se deben emplear durante la adquisición, suministros, desarrollo, operación, mantenimiento y eliminación de productos o servicios software.

##### **2.1.1.2 Proceso de Construcción de Software.**

Este es el proceso de producción fundamental dentro de la industria del software, de ahí su importancia. En todo tipo de empresa dedicada al desarrollo de software este proceso es el más ejecutado y el que como mínimo se realiza. Según [2] las pequeñas empresas dedicadas al desarrollo de software concentran la mayor

parte de sus esfuerzos en la codificación (así como en el análisis y diseño), lo cual refleja el quehacer de este tipo de organizaciones.

#### **2.1.1.2.1 Propósito.**

El propósito del proceso de construcción de software es el de producir unidades software ejecutables, las cuales deben reflejar adecuadamente el diseño de software [5].

#### **2.1.1.2.2 Resultados.**

Como resultado de la implementación exitosa del proceso de construcción de software se obtiene [5]:

- Se definen criterios de verificación para todas las unidades software en contra de sus necesidades.
- Se producen unidades software definidas por los diseños.
- Se establece la consistencia y trazabilidad entre las unidades software y los requisitos y el diseño.
- Se realiza la verificación de las unidades software en contra de los requisitos y el diseño.

#### **2.1.1.2.3 Actividades y tareas.**

En el proceso de construcción se deberá implementar una actividad y sus respectivas tareas de acuerdo con las políticas de la organización y los procedimientos que son aplicables en relación con el proceso [5].

##### **2.1.1.2.3.1 Actividad de construcción de software.**

Para cada elemento de software esta actividad consta de las siguientes tareas [5]:

##### **2.1.1.2.3.2 Tareas.**

- Quien implementa debe desarrollar y documentar lo siguiente:
  - Cada unidad de software y bases de datos.
  - Los procedimientos para las pruebas y los datos de las pruebas de cada unidad de software y base de datos.
- Probar cada unidad software y base de datos desarrollada, asegurando que satisfacen sus requerimientos.
- Actualizar la documentación de usuario según sea necesario.
- Actualizar los requisitos de la prueba y el plan para la integración del código software desarrollado.
- Evaluar el código software y resultados de la prueba teniendo en cuenta los siguientes criterios:

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

- Trazabilidad a los requerimientos y diseños de los elementos software.
- Consistencia externa con los requerimientos y diseños de los elementos software.
- Consistencia interna entre los requerimientos de las unidades.
- Cobertura de las pruebas de las unidades.
- Adecuación de los métodos de codificación y estándares usados.
- Viabilidad de la integración y pruebas software.
- Viabilidad de operación y mantenimiento.

Los resultados de las evaluaciones y pruebas deben ser documentados.

Para que un producto software cumpla con los requerimientos del cliente, este deberá ser desarrollado con un proceso estable o con poca variabilidad. El control estadístico de procesos (SPC, por sus siglas en inglés) se comprende de un conjunto de herramientas estadísticas útiles que tienen el propósito de observar si el proceso está dentro de su variabilidad aleatoria o ha salido de control produciendo fallas que sean asignables a algún problema determinado [18].

### **2.1.2 SPC (Statistical Process Control).**

El SPC es un método de gestión de proceso que se lleva a cabo a través de un análisis estadístico y que incluye la definición, medición y control de los procesos [19] [20]. Las actividades típicas del SPC son: i) detección de situaciones fuera de control, basados en los límites de control u otros patrones de inestabilidad asociados con los gráficos de control, ii) análisis de las causas que generan la situación de fuera de control y iii) la estabilización del proceso mediante la eliminación de estas causas [19]. SPC puede aplicarse a cualquier proceso y sus siete herramientas principales son las siguientes [19] [21] [22]:

- El histograma o el diagrama de tallo y hoja.
- La hoja de verificación.
- La grafica de Pareto.
- El diagrama de causa y efecto.
- El diagrama de concentración de defectos.
- El diagrama de dispersión y
- El grafico de control (en la literatura esta es la técnica más usada en el control de los procesos software).

Este conjunto de técnicas estadísticas son utilizadas ampliamente en la industria para controlar los procesos de producción, sin embargo su aplicación al software es reciente, y aún se encuentran dificultades para su adaptación en las organizaciones software puesto que en sus procesos predomina la presencia de actividades cognitivas del factor humano, lo que implica diferencias con la ejecución de los procesos automatizados del sector industrial [18] [19] [23] [10] [24]. Por lo anterior se requiere de la adaptación de SPC al contexto del software, definiendo convenientemente un sistema de medición óptimo, mejorando la definición de las métricas a usar y captando grandes cantidades de datos para dichas métricas [8] [22] [10] [11].

#### **2.1.2.1 Definición.**

El SPC es una manera eficaz de identificar la estabilidad, o la falta de ella, en un proceso mediante el uso de gráficos de control y sus interpretaciones [23] [9].

Para el uso de SPC se debe entender que en cualquier proceso, independientemente de su diseño, siempre existirá una variabilidad inherente al proceso o variabilidad natural, generada por la acumulación de muchas causas pequeñas e inevitables. En el contexto del SPC, la variabilidad natural de un proceso es generada por causas fortuitas (o comunes), y se concluye que un proceso que opera únicamente con causas fortuitas de variación está bajo control estadístico. Sin embargo, hay casos de procesos con presencia de otro tipo de causas de variación, esta variación es grande con respecto a la variabilidad natural y puede llevar a un desempeño inadecuado del proceso. A este tipo de variación se le conoce como causas asignables (especiales) y si en un proceso existe variación debido a causas asignables, se dice que este proceso se encuentra fuera de control [22] [23] [10].

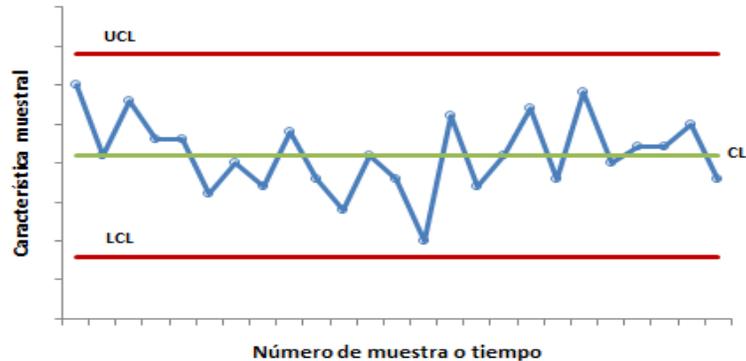
Uno de los objetivos primordiales del control estadístico de procesos es el de detectar rápidamente la aparición de causas asignables en un proceso, para luego hacer investigaciones al respecto y emprender acciones que ayuden a la corrección del problema que causa el estado de fuera de control [19] [22].

#### **2.1.2.2 Técnica de gráfico de control.**

El gráfico de control (o carta de control) es una técnica SPC usada para el monitoreo de procesos [23] [10]. En la Figura 2 se muestra un gráfico de control típico tomado de [22], en esta carta se grafican los promedios de las mediciones de una característica (métrica o atributo de calidad tomados en muestras del proceso) contra el tiempo (o el número de muestras). Como se puede observar, un gráfico de control lleva una línea central (CL, por sus siglas en inglés) más una

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

línea arriba y otra debajo de esta línea central, que representan los límites de control superior e inferior (UCL y LCL, por sus siglas en ingles), para discriminar entre variaciones asignables y comunes [10].



**Figura 2. Ejemplo de un gráfico de control.**

La línea central representa el sitio donde deberían localizarse las características del proceso que se está evaluando, en caso de no haber situaciones de variabilidad [22] [10]. Los límites de control superior e inferior (UCL y LCL) se determinan haciendo uso de ciertas consideraciones estadísticas que se discutirán más adelante, pero se puede decir que su elección debe ser congruente con la idea de que si el proceso está bajo control todos los puntos muestrales se deben localizar entre estos límites de control. Así, la variabilidad del proceso se ve representada cuando los promedios muestrales de una característica del proceso se localizan fuera de los límites de control, es decir que si uno solo de los puntos graficados se encuentra fuera de los límites de control es una señal clara de que el proceso está fuera de control, lo que demanda investigar el proceso y tomar acciones correctivas que eliminen las fuentes inusuales de variabilidad [22] [23] [10].

Aun cuando todos los puntos se localicen dentro de los límites de control, si estos tienen un comportamiento no aleatorio, esto puede ser un indicio de que el proceso se encuentra fuera de control, por ejemplo cuando en un gráfico de control se muestran una secuencia de puntos del mismo tipo (corrida) ya sea de forma ascendente o descendente es un claro patrón que revela una situación de fuera de control en el proceso evaluado. En la práctica se han sugerido varias reglas para detectar patrones no aleatorios en los gráficos de control, estas reglas buscan incrementar la sensibilidad de los gráficos, aumentando su efectividad en

la detención de la variabilidad de los procesos, a continuación se muestra algunas de estas reglas tomadas de [22] [10] [21]:

- Uno o más puntos fuera de los límites de control.
- Un patrón secuencial o no aleatorio en los datos.
- 14 puntos seguidos alternándose arriba y abajo.
- 6 puntos seguidos que aumentan o disminuyen de manera sostenida.
- Una corrida de 8 puntos consecutivos en el mismo lado de la línea central.
- Uno o más puntos cerca de los límites de control o advertencia (los límites de advertencia tienen un rango más corto que los límites de control, es decir, se encuentran entre la línea central y los límites de control, estos se usan para aumentar la sensibilidad de los gráficos).
- 2 de 3 puntos consecutivos fuera de los límites de advertencia pero aun dentro de los límites de control.

#### **2.1.2.2.1 Tipos de gráficos de Control.**

Como se plantea en [25] los gráficos de control se dividen en dos según el tipo de característica a usar:

##### **2.1.2.2.1.1 Gráficos de control por variables.**

Aquí la característica a medir es una variable y sus valores se distribuyen según una distribución normal. En la Tabla 2 se describen los tipos de gráficos de control por variables junto con las fórmulas para el cálculo de sus límites de control, donde R= Recorrido (diferencia entre el valor máximo y el mínimo de un conjunto de datos) y las constantes A, B y D se obtienen de la Tabla 1. En los tipos de gráficos de control por variables que se analizan, sus datos siguen una distribución normal.

##### **2.1.2.2.1.2 Gráficos de control por atributos**

Este tipo de gráficos de control se usa para aquellas características que no pueden ser representadas con valores numéricos, por ejemplo hay casos en los que una característica solo puede tomar los valores de conforme o no conforme, a estas características se les denomina atributos y pueden describir dos situaciones:

- El producto desarrollado en conforme (no defectuoso) o no conforme (defectuoso). Cumple con las especificaciones demandadas o no. Los datos siguen una distribución binomial.
- El producto puede tener varias no conformidades (defectos) y el número de estas es determinado. Los datos siguen una distribución Poisson.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

En la Tabla 3 se describen los tipos de gráficos de control de atributos junto con las fórmulas para el cálculo de sus límites de control, aquí encontraremos las variables:

- **N**= Tamaño de la muestra.
- $\bar{n}$  = Media del tamaño de las muestras  $n = (n_1+n_2+n_3+\dots+n_n)/N$

Este tipo de gráficos de control requieren tamaños de muestras grandes (mayor de 50), con el objetivo de poder detectar cambios.

n	2	3	4	5	6	7	8	9	10
A <sub>2</sub>	1,880	1,023	0,729	0,577	0,419	0,419	0,373	0,337	0,308
D <sub>3</sub>	0	0	0	0	0,076	0,076	0,136	0,184	0,223
D <sub>4</sub>	3,267	2,575	2,282	2,004	1,924	1,924	1,864	1,816	1,777

Tabla 1. Constantes para los gráficos X-barra y R.

### 2.1.2.2.1.3 Construcción de un gráfico de control

A continuación se describen una serie de pasos necesarios para la realización de un gráfico de control:

1. Elección de la característica a controlar.
2. Toma de datos de la característica elegida
3. Elección del tipo de grafico de control a usar.
4. Calculo de la media de los valores.
5. Calculo de los límites de control.
6. Determinar la escala del gráfico, dibujar la línea centra y los límites de control y representar los valores de la característica medida.
7. Análisis del gráfico de control.

MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

Tipo de gráfico de variable	Descripción	Grafico $\bar{X}$	Grafico R	Grafico S
Valores individuales	Se toman muestras individuales para el cálculo de la media.	$LC = \mu_x$ $LCS = \mu_x + A_0 \sigma_x$ $LCL = \mu_x - A_0 \sigma_x$  (A=3)		
X-barra y R	<p>Se hacen subgrupos racionales de las muestras. Estos gráficos son usados para monitorear la media y la variancia y los subgrupos racionales tienen un tamaño fijo. Se representan dos gráficos: uno para el control de las medias y otro de rangos para el control de la variabilidad del proceso. Este grafico se puede realizar con muestras pequeñas (menos de 8).</p> <p>R representa el recorrido que es igual a la diferencia entre el valor máximo y el valor mínimo de un conjunto de datos.</p>	$LC = \bar{\bar{X}}$ $LCS = \bar{\bar{X}} + A_2 \bar{R}$ $LCL = \bar{\bar{X}} - A_2 \bar{R}$	$LC = \bar{R}$ $LCS_R = D_4 \bar{R}$ $LCL_R = D_3 \bar{R}$	
X-barra y S	<p>Se hacen subgrupos racionales de las muestras.</p> <p>Estos gráficos son usados para monitorear la media y la variancia. Aquí, los subgrupos racionales tienen un tamaño variable.</p>	$LC = \bar{\bar{X}}$ $LCS = \bar{\bar{X}} + A_3 \bar{R}$ $LCL = \bar{\bar{X}} - A_3 \bar{R}$		$LC = \bar{S}$ $LCS_S = B_4 \bar{S}$ $LCL_S = B_3 \bar{S}$

Tabla 2. Tipos de gráfico de variable.

MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

Tipo de gráfico de atributo	Descripción	Tamaño de la Muestra	Línea Central	Limites	Distribución de Datos
P	Porcentaje de productos no conformes o defectuosos encontrados en la muestra seleccionada. El tamaño de la muestra es variable.	Variable	$\bar{P} = \sum P/N$  $LC = \bar{P}$	$LCS = \bar{P} + 3\sqrt{(\bar{P}(1-\bar{P})/n)}$  $LCL = \bar{P} - 3\sqrt{(\bar{P}(1-\bar{P})/n)}$	Binomial
Np	Número de productos no conformes o defectuosos encontrados en la muestra seleccionada. El tamaño de la muestra es constante.	Constante	$LC = n\bar{P}$	$LCS = \bar{P} + 3\sqrt{n\bar{P}(1-\bar{P})}$  $LCL = \bar{P} - 3\sqrt{n\bar{P}(1-\bar{P})}$	Binomial
U	Número de no conformidades o defectos por unidad de producto inspeccionada. El tamaño de la muestra es variable.	Variable	$\bar{u} = \sum u/N$  $LC = \bar{u}$	$LCS = \bar{u} + 3\sqrt{\bar{u}/n}$  $LCL = \bar{u} - 3\sqrt{\bar{u}/n}$	Poisson
C	Número de no conformidades o defectos encontrados en una unidad de producto inspeccionada. El tamaño de la muestra es constante.	Constante	$\bar{c} = \sum c/N$  $LC = \bar{c}$	$LCS = \bar{c} + 3\sqrt{\bar{c}}$  $LCL = \bar{c} - 3\sqrt{\bar{c}}$	Poisson

Tabla 3. Tipos de gráfico de atributo

Aquí, es importante resaltar que el desempeño de un gráfico de control se ve afectado con las decisiones acerca del tamaño de la muestra, el intervalo de muestreo y la correcta asignación de los límites de control. Cada medida tiene su propia dinámica, características inherentes y técnicas de normalización. Los datos se coleccionan en varios intervalos de tiempo, se realizan análisis en diferentes puntos del ciclo de vida y se deben usar gráficos diferentes para su representación. Por lo tanto, se requieren aproximaciones separadas para cada métrica a analizar [8].

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

A las empresas que se dedican al desarrollo del software, la medición les permite mejorar sus procesos, ayudando con el seguimiento y control de los proyectos y productos, buscando siempre una mejor calidad en los mismos. En estas empresas que consideran el uso de técnicas estadísticas como el SPC para el control y mejora de sus procesos software, la medición se convierte en una tarea trascendente. Así mismo la aplicación de SPC requiere de métricas y datos adecuados para su uso en el contexto del software, por lo que las métricas deben estar bien definidas para adaptarse al uso de SPC además de satisfacer una necesidad de información.

### **2.1.3 Medición Software.**

La medición del software permite entender, controlar y mejorar los procesos y productos software.

Para comprender que es la medición software se requiere de la explicación separada de otros conceptos, tales como: Medición, Métrica y Medida. Basados en [26], dichos conceptos se describen a continuación.

#### **2.1.3.1 Medición.**

La medición es la acción que permite obtener el valor de una medida. Cada medición produce una medida usando una métrica. Esta es una actividad obligatoria si se considera la aplicación de técnicas estadísticas para control y mejora.

#### **2.1.3.2 Medida.**

Es un dato que se genera como el resultado final de la acción de medir, basado en una métrica.

#### **2.1.3.3 Métrica.**

Una métrica se puede considerar como una estructura (plantilla) de características específicas para la obtención de una medida. Es una forma de medir (método de medición, función de cálculo o modelo de análisis) y una escala, definidas para realizar mediciones de uno o varios atributos.

Una métrica puede ser directa al realizar mediciones sin depender de ninguna otra métrica, o puede ser indirecta al depender de las mediciones de otras métricas.

### 2.1.3.3.1 Métricas Software.

En la medición del software las métricas pueden ser de tres tipos, según el objetivo de medición o el contexto de aplicación, estas son:

- Métricas de producto: Se centran en las características del software y no en cómo fue producido. Aquí se debe tener en cuenta que también son producto todos los artefactos, documentos, modelos y componentes que conforman el software.
- Métricas de proceso: Se toman de mediciones relacionadas con las actividades software y algunos de sus atributos típicos son el esfuerzo, el coste y los defectos encontrados [27]. De acuerdo con [28] las métricas del proceso software se utilizan para propósitos estratégicos y en muchas propuestas, la medición del proceso se realiza extrayendo las características de tareas específicas de la ingeniería del software.
- Métricas de Proyecto: Cuando se mide el proyecto el objetivo fundamental que se pretende es el de reducir el coste total y el tiempo de desarrollo del mismo. En relación a las métricas de proceso, las métricas de proyecto se consideran a un nivel táctico, es decir, las métricas de proyectos y los indicadores derivados de ellos son utilizados por un administrador de proyectos y por un equipo de software para adaptar el flujo de trabajo del proyecto y las actividades técnicas [25].

Los conceptos definidos en los numerales 1.1, 1.2 y 1.3 pretenden aplicarse a un entorno específico de empresa u organización, que para este caso son las VSEs dedicadas al desarrollo de software. A continuación se detalla una definición de las VSEs junto con la explicación de su importancia en la industria del software.

### 2.1.4 VSEs (Very Small Enterprises).

VSEs que para este trabajo se denomina en español entidades muy pequeñas, es una categorización de pequeñas organizaciones descrita en ISO/IEC 29110. A continuación se trata de especificar que es una VSE, con el objetivo de definir este tipo de empresas y su rol en la industria del software nacional e internacional.

#### 2.1.4.1 Definición.

Según [29] las VSEs son organizaciones, departamentos o proyectos dedicados a la implementación de software con una fuerza de trabajo de hasta 25 empleados. La mayoría de las empresas de software entran en la categoría de VSE. Para [30] [4], una pequeña empresa se compone de una serie de características,

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

competencias y necesidades que afectan su naturaleza y el alcance de sus actividades. Algunas de estas características son las siguientes:

- Estructura organizacional plana.
- Recursos económicos limitados y esperan retorno a la inversión a corto plazo.
- Pequeño número de empleados.
- Están orientadas sobre todo al desarrollo del software específico para usuarios conocidos y no para todo el mercado. Altamente dependiente de los clientes. Sus negocios son basados en el conocimiento.
- Tienen escasa planificación o la planeación es a corto plazo.
- Desconocen de la importancia que tiene el proceso de desarrollo sobre la calidad del producto, así como el uso de normativas para este fin. La construcción de software, en muchos de los casos, se realiza de forma artesanal, empírica y caótica.

A nivel nacional, existe una categorización de micro, pequeñas y medianas empresas (Mipymes), que se estructura de la siguiente manera según Artículo 2° de la ley 905 de 2004: la microempresa cuenta con planta de personal no superior a diez (10) trabajadores, la pequeña empresa tiene una cantidad de personal entre los once (11) y cincuenta (50) trabajadores, la mediana empresa cuenta con personal entre los cincuenta y uno (51) a los doscientos (200) trabajadores y con más de doscientos (200) trabajadores se categoriza a la gran empresa. En la Tabla 4 se hace un contraste de la capacidad en número de trabajadores, entre la categorización de pequeña organización (VSEs) descrita en la norma ISO/IEC 29110 y las Mipymes a nivel nacional.

# Trabajadores	Tipo de Empresa	
10	VSEs	Micro Empresa
11		Pequeña Empresa
25		
50		
200	Mediana Empresa	
Más de 200	Gran Empresa	

**Tabla 4. Cuadro comparativo de la capacidad en número de empleados de las VSEs, Mipymes y la gran empresa.**

#### **2.1.4.2 Importancia.**

El sector de las Tecnologías de la información (TI) brinda muchas oportunidades en la economía mundial. Dentro de las TI se encuentra el sector de Software que en las últimas dos décadas ha crecido en un alto porcentaje generando desarrollo en las economías de países emergentes como Brasil, México, Argentina, Chile, Colombia, entre otros [31]. Además, las pequeñas organizaciones son piezas fundamentales en el marco de la economía mundial, y en la industria del software no podría ser diferente al estar compuesta en gran parte por VSEs [4].

Para contextualizar los datos a nivel nacional, a continuación se muestran los resultados de la encuesta promovida por el Ministerio de Tecnologías de la Información y las Comunicaciones-MINTIC a través del programa para el Fortalecimiento de la Industria de TI-FITI y FEDESOFIT, aplicada a las empresas que tienen por actividad principal el desarrollo y comercialización de software, así como aquellas que se dedican a la prestación de servicios asociados de arrendamiento, mantenimiento, administración, análisis y transferencia de tecnologías de la información [32]. La encuesta tuvo como objetivo capturar información que permitiese determinar las características de la oferta y demanda de bienes y servicios de las empresas de software y servicios relacionados: sus actividades económicas principales, las líneas de negocio preferidas y el grado de especialización, los sectores económicos que demandan sus servicios, así como su distribución regional [32]. Esta encuesta fue respondida en su totalidad por 962 empresas y parcialmente por 158 empresas, generando un total de 1120 empresas encuestadas. Los resultados que más se destacan son:

- El 54.2% de las empresas encuestadas tiene como su labor primordial el Desarrollo de Software mientras que el 20.1% se dedica a la Comercialización o Licenciamiento de Software y el 25.40% a los servicios software, tal como se evidencia en la Tabla 5.
- Como se muestra en la Tabla 6, las cuatro líneas de negocios principales en la industria del software nacional son: el 63% de las empresas encuestadas ofrece Desarrollo a la Medida, el 50% Servicios de Consultoría, el 48% Desarrollo de Aplicaciones Web y el 41% ofrece Soporte y Mantenimiento de Software.
- En cuanto a las exportaciones, el porcentaje de empresas que respondió que exportaba software o servicios es muy bajo: 2% en el 2010 y 4.5% en 2011, aunque se percibe un crecimiento en el número de empresas y en el monto exportado (ver Tabla 7).

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

- Al igual que en el caso de las exportaciones los resultados de la Tabla 8 muestran que el porcentaje de empresas que importaron es muy bajo: 5% en el 2010 y 7% en 2011, aunque existe un leve incremento, los montos importados son poco significativos.
- Finalmente, en cuanto al personal empleado, el 76.8%, tiene menos de 50 empleados, con lo cual se puede inferir que este es el segmento más grande y corresponde a las micro y pequeñas empresas (en donde se encuentra enmarcada la categorización de VSEs) (ver Tabla 9).

Actividad Principal de las Empresas	No. De Empresas	%
Desarrollo de Software	607	54,20%
Comercialización o Licenciamiento de Software	225	20,10%
Servicios de Software	285	25,40%
No Informa	3	0,30%
<b>TOTAL</b>	<b>1120</b>	<b>100,00%</b>

**Tabla 5. Actividad principal de las empresas.**

Línea de Negocio	No. De Empresas
Desarrollo a la medida	706
Servicios de consultoría	562
Desarrollo de aplicaciones Web	544
Soporte y mantenimiento de software	460
Seguridad informática	384
Computación en la nube	371
Integración de sistemas	329
Capacitación y entrenamiento	328
Infraestructura tecnológica	315
Tecnologías transversales: BPM, CRM, ERP, ECM, BI y BSC	307
Integración de redes	305
Mejora de procesos	301
Aplicaciones móviles	289
Comercio electrónico	285
Portales, redes sociales, Web 2.0	272
Gestión documental	255
Servicios de migración y calidad de datos	250
Sistemas personales	242
Outsourcing de sistemas de información	234

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

e-Marketing, e-Learning	242
Testing de software	228
Planeación de sistemas, SOA e integración	214
Outsourcing de infraestructura	210
Animación digital/Video juegos	207
Outsourcing de procesos (BPO)	196
Mayorista (ventas de soluciones de software al por mayor)	190
Institución académica	190

**Tabla 6. Líneas de negocios seleccionadas por número de empresas.**

	2010		2011	
	Empresas Exportadoras	Valor de Exportación USD	Empresas Exportadoras	Valor de Exportación USD
<b>Software</b>	20	22.049.944	24	31.309.161
<b>Servicios</b>	12	3.561.668	21	4.507.928

**Tabla 7. Exportaciones.**

	2010		2011	
	Empresas Exportadoras	Valor de Exportación USD	Empresas Exportadoras	Valor de Exportación USD
<b>Hardware</b>	20	864.470	23	1.949.803
<b>Software</b>	33	1.506.961	42	1.975.728

**Tabla 8. Importaciones.**

Rango por Número de Empleados	No. De Empresas	%
1 a 50 Personas	860	76,80%
51 a 100 Personas	45	4,00%
101 a 200 Personas	21	1,90%
201 a 300 Personas	9	0,80%
301 a 400 Personas	1	0,10%
401 a 500 Personas	4	0,40%
Más de 500 Personas	8	0,70%
No informo	172	15,40%
<b>TOTAL</b>	<b>1120</b>	<b>100,00%</b>

**Tabla 9. Número de empleados por empresa.**

## **2.2 Estado del Arte.**

Al realizar una búsqueda en la literatura y hacer un estudio de los resultados sobre métricas del proceso de construcción de software y el uso de SPC en el contexto del software, se encontraron los siguientes hallazgos en términos de Métricas y medición del software y SPC en la industria del software. Posteriormente se presenta una discusión sobre los trabajos relacionados.

### **2.2.1 Métricas y Medición del Software**

#### **2.2.1.1 Ontología de la Medición [26].**

Varios grupos de investigación se han dado a la tarea de formar una propuesta de terminología bien definida, dando lugar a esta ontología de la medición software, que se basa en cuatro conceptos fundamentales: forma de medir, la acción de medir, el resultado de la medición y el concepto de métrica.

Nuestro proyecto de tesis utiliza esta ontología como la base conceptual para la terminología sobre medición y para efectos de clarificar los términos a usar en este documento se presentan tres conceptos fundamentales:

- Métrica: Una forma de medir (método de medición, función de cálculo o modelo de análisis) y una escala, definidas para realizar mediciones de uno o varios atributos.
- Medición (Acción de Medir): La acción que permite obtener el valor de una medida para un atributo de una entidad, usando una forma de medir.
- Medida: Resultado de una medición.

#### **2.2.1.2 SPDW +: Un Enfoque para la Captura de Métricas de Calidad en Entornos de Desarrollo de Software [33].**

Esta propuesta expresa que uno de los objetivos de una métrica de software es determinar cuantitativamente el grado en que un proceso, producto o proyecto posee un determinado atributo. En una organización, las métricas proporcionadas son utilizadas para centrarse en las áreas de procesos clave que necesitan mejorar, cuanto mayor sean los datos de las mediciones, la capacidad de la organización es más completa y eficaz para evaluar sus centros de atención.

Se establece la actividad de medición como relevante y esencial para la previsibilidad y la detección temprana de desviaciones, para la toma de decisiones y también indica el grado de madurez en el desarrollo de software para una organización. La medición se convierte en un componente clave de cualquier proceso de mejora en las organizaciones, y en esta propuesta se resalta la

definición de un programa de métricas como uno de los factores de éxito más importantes en la mejora. La propuesta considera también las prácticas de: (i) almacenamiento, para la consolidación de las métricas recopiladas en un repositorio central de datos, teniendo en cuenta el proyecto y los niveles de la organización, (ii) el análisis para la exploración de los datos del repositorio de métricas principalmente a través de estadísticas y lo más relevante para nuestro proyecto (iii) que la captura de métricas sea realizada frecuentemente con el objetivo final de llegar a una consolidación constante de los indicadores.

Se resalta también que para la actividad de medición en el desarrollo del software se deben abordar tres cuestiones fundamentales: (1) ¿Que métricas capturar?, (2) ¿Cómo manejar la heterogeneidad en el contexto del desarrollo de software?, y (3) ¿Cuál es la frecuencia y latencia para recoger las mediciones de las métricas?

Aquí se considera el uso de métodos estadísticos en las organizaciones para comprender mejor el comportamiento de sus procesos, hacer un monitoreo constante y establecer correlaciones para especular sobre las causas profundas de los problemas con el fin de tomar acciones correctivas y elaborar planes de mejora.

A diferencia del prototipo para la recolección de métricas expuesto en este artículo, nuestro proyecto se enfocará en el desarrollo de un prototipo que de soporte específicamente a la recolección de métricas del proceso de construcción de software en las pequeñas organizaciones (métricas que deben ser aplicables al SPC).

### **2.2.1.3 Una Revisión Sistemática de Medición en la Ingeniería Software: Estado del Arte en las Medidas [34].**

Este trabajo presenta una revisión sistemática sobre métricas de software, con una estrategia de búsqueda predefinida para lograr resumir y clasificar la información de todos los esfuerzos realizados en este campo. Los datos proporcionados por esta revisión sistemática se basa en la documentación acerca de métricas del proceso software en cuatro grandes fuentes: Wiley, Scinece Direct, ACM e IEEE, estas métricas están relacionadas con proceso, producto y proyecto, así como en las áreas de diseño, desarrollo, análisis, testing y mantenimiento, además se clasifica un compendio de métricas aplicadas a la construcción del software.

El objetivo de este trabajo es conocer y aclarar las respuestas a tres cuestiones diferentes: ¿qué medir?, ¿cuándo medir? y ¿cómo medir en el proceso software?, para esto se fundamenta en la ontología de la medición que le permite describir

aspectos de las métricas tales como: entidad, escala, atributos (internos o externos), tipo de medición (directa o indirecta), su representación, descripción y foco de aplicación.

Según este documento la entidad que más se mide es el producto, y las entidades cuya medición ha sido menos apoyada son las de proyecto y proceso, la razón es que la medición de producto es más fácil, en los otros dos casos se suele encontrar definiciones ambiguas de sus atributos lo que hace difícil su medición, por ejemplo los atributos técnicos y de calidad en producto están muy bien definidos porque la calidad se ha visto fuertemente enfocada en el producto. Sin embargo es preciso resaltar que las mediciones sobre entidades de producto ayudan a medir las de proceso y proyecto.

Aquí se proporciona un estado del arte sobre medición del software, llevando a cabo una revisión sistemática y construyendo un documento técnico en donde se resumen y clasifican las medidas más actuales y útiles en la literatura. Se apropia el aporte de un compendio de métricas aplicadas a la construcción del software, se analiza la completitud de su información y se estudian sus fuentes.

## **2.2.2 SPC en la Industria del Software.**

### **2.2.2.1 Gestionando la Mejora de Procesos Software (SPI) a Través del Control Estadístico de Procesos (SPC) [10].**

Esta propuesta expresa que las variaciones en un proceso se deben a causas comunes o asignables. Las primeras son parte del propio proceso mientras que las últimas dan lugar a un comportamiento inestable del proceso y por lo tanto a menos previsibilidad. Aquí se define la técnica SPC como un enfoque basado en estadísticas, orientado a determinar si un proceso es estable o no, discriminando si la presencia de variación se debe a causas comunes o asignables. Si la variación en el proceso es sólo debido a causas comunes, el proceso se dice que es estable y su comportamiento es predecible dentro de un margen de error determinado. SPC utiliza varios gráficos de control, junto con sus indicadores, estableciendo los límites de operación que marcan la variación aceptable del proceso, además el SPC conlleva la función de monitorear y evaluar el proceso, su evolución y resultados en el tiempo. Esto permite visualizar cuando los valores sobrepasan límites, asumiendo que una causa asignable está influyendo en el proceso, y por lo tanto se hace inestable.

Según este artículo, para el uso de SPC en procesos software se deben cumplir tres tareas básicas: i) la selección de un conjunto de métricas (las más adecuadas para los procesos de software), ii) el análisis de la interpretación (en la cual se

interpretan los gráficos de control desde una perspectiva de procesos software) y iii) la definición de un proceso de investigación para monitorear el desempeño del proceso y apoyar las decisiones en actividades de mejora de procesos software (SPI).

La diferencia del trabajo que se propone con lo planteado en este artículo es que para la aplicación de SPC en su caso de estudio fueron utilizadas métricas del contexto industrial. Esto contrasta con el hecho de que los procesos de software son diferentes a los procesos industriales y son influenciados por las variables de estado más que las de fabricación. Por otra parte, el artículo confirma la necesidad de seleccionar el conjunto de métricas apropiado para el proceso que se quiere controlar como una tarea vital y primordial si se considera la aplicación de técnicas estadísticas como es SPC. En el presente trabajo de investigación se busca determinar un grupo de métricas del proceso de construcción que sean adecuadas para la aplicación de SPC.

#### **2.2.2.2 Medidas de calidad en proceso, producto y mantenimiento, aplicadas al Control Estadístico de Procesos [23].**

Aquí se expone el SPC como una técnica bien establecida (originaria en 1990) y que ha demostrado ser efectiva en ambientes industriales. Sin embargo en el contexto software esta es una técnica muy reciente y debe ser adaptada, basado en las diferencias con los procesos industriales. En un contexto software, la presencia de actividades cognitivas provocadas por un predominante factor humano, implica diferencias en la ejecución de un proceso. Esto hace que las entradas y salidas sean diferentes para cada instancia y en consecuencia, el riesgo de inestabilidad en el proceso esté presente constantemente.

Este artículo explica que para las organizaciones del contexto industrial incluso en el caso partículas de las organizaciones software, la actividad de medición permite mejorar los procesos, ayudando en la planificación, en el seguimiento y control de los proyectos y productos, todo en aras de lograr una mejor calidad. En este sentido, las medidas de calidad del software se basan en las mediciones de atributos específicos del proceso, proyecto y producto. Aquellas mediciones relacionadas con proceso y producto cobran mayor importancia para asegurar la calidad final. Esto se debe a que, en general, no es posible modificar los recursos existentes pero si influir en el proceso de desarrollo como en el producto final. Aquí se propone un listado de métricas relacionadas con calidad de productos, proceso y la fase de mantenimiento. Se aplican y analizan tres de estas medidas (Densidad de defecto, Porcentaje de repetición de trabajo, Performance de Revisión) a la técnica SPC de gráficos de control en un caso de estudio.

Con relación a los resultados del caso de estudio, se demuestra que no es necesario tener un alto nivel de madurez para la aplicación del SPC en una organización de software. Sin embargo, se debe tener en cuenta que la cantidad y la calidad de los datos de la métrica son indispensables para la fiabilidad de los límites de control; así como el número de proyectos (o la cantidad de datos) a los que se aplican las mismas deben ser suficientemente altos. Cada medida tiene características particulares y complejidades relacionadas con su definición, colección e interpretación. La mayoría de estas dificultades se deben a la inestabilidad de las definiciones de las medidas para un análisis estadístico detallado. También los criterios para el valor de las medidas y la variabilidad de los datos hacen difícil la aplicación de SPC incluso si los datos son genéricos o escasos, por lo tanto, el nivel de detalle se debe determinar muy cuidadosamente basado en la cantidad de datos de la medida, el nivel de criticidad del proceso y el análisis de los datos de medición.

Este artículo propone una serie de características de las métricas que son vitales para el análisis y la aplicación de la técnica SPC de gráficos de control. De este análisis se generan criterios relacionados con las métricas y su aplicación al SPC que son de interés para el presente trabajo de investigación. Este artículo también brinda un aporte en métricas software usadas con la técnica SPC de gráficos de control, que son consideradas como insumo para el proyecto.

### **2.2.2.3 Utilización de Control Estadístico de Procesos (SPC) en las organizaciones de software emergentes: Errores y Sugerencias [8].**

Este artículo expresa que las organizaciones en el ámbito de la ingeniería de software que consideran la aplicación de las técnicas de control estadístico de procesos (SPC) deben ser lo suficientemente maduras. Existen muchos debates acerca de la aplicación de SPC en la industria del software, lo cual ha conllevado a que esta técnica sea estudiada por varios investigadores que describen la utilización de SPC en el contexto del modelo Capability Maturity Model (CMM) para la mejora del proceso de software.

Los autores de este artículo proporcionan información técnica detallada sobre SPC y una hoja de ruta de la aplicación del SPC que en su mayoría se centra en los principios y los métodos para evaluar y controlar el desempeño del proceso en función de los principios de Shewhart. Además se proporciona un guía con la descripción de las técnicas estadísticas disponibles en la actualidad para la gestión y el control de la calidad del software durante la especificación, diseño, producción y mantenimiento.

Algunos de las propuestas presentadas en este estudio se basan en modelos de mejora de procesos y centrados en SPC como una forma de lograr altos niveles de madurez, deduciendo los temas críticos para la implementación exitosa de SPC tales como la estabilidad del proceso, la capacidad de medición y la fiabilidad de los datos. En otras palabras, si el proceso se lleva a cabo sistemáticamente, las métricas correctas son seleccionadas y un mecanismo de recopilación de datos fiables se establece haciendo posible obtener un beneficio al aplicar la técnica de SPC.

Este artículo presenta una guía para la aplicación de SPC en compañías emergentes y describe las técnicas estadísticas a utilizar. Además se incluye un caso de estudio donde se aplica la guía y las técnicas propuestas. En dicho caso de estudio se presentó un problema con la recolección de las métricas de software y la selección de las mismas. A diferencia de este estudio, el presente trabajo de grado se enfoca en determinar un grupo de métricas solo del proceso de construcción de software que sean útiles para llevar a cabo SPC en pequeñas organizaciones y desarrollar un prototipo que apoye la recolección de sus datos.

#### **2.2.2.4 Experiencias de la aplicación de técnicas SPC a los procesos de desarrollo de software [19].**

Esta propuesta es una aplicación práctica del SPC en la industria del software japonesa. En esta industria el SPC es conocido como TQM, para nombrar a las siete herramientas básicas o siete magníficos que se utilizan con frecuencia para el SPC, de los cuales el grafico de control es el más importante según su criterio, debido a que se fundamenta en criterios científicos y sus resultados arrojan información real sobre los procesos.

Aquí se define la técnica SPC como un método de gestión de proceso a través de un análisis estadístico, que incluye la definición, medición y control de los procesos. Según los autores, las actividades típicas de SPC incluyen:

- Detección de situaciones fuera de control, basados en los límites de control u otros patrones inestables asociados con el control de gráficos
- El análisis de causa Especial para cada situación fuera de control
- El proceso de estabilización mediante la eliminación de estas causas especiales.

En Japón, SPC (también llamado: Total Quality Management - TQM) es conocido por ser una herramienta poderosa para mejorar la calidad del producto en la industria de fabricación, la manufactura o actividades centradas en el producto, pero su aplicación a la industria del software todavía no es muy popular debido a

que los proceso software son diferentes a los proceso industriales. Según el artículo, una aplicación directa de SPC para el desarrollo de software es difícil porque el proceso es complejo e implica un alto grado de actividad de la creatividad mental. Esto genera que en cada ejecución se realice una actividad creativa y única, que hace que sea difícil obtener un gran conjunto de datos homogéneos, y crea múltiple causas comunes.

En este articulo la empresa HITACHI SOFTWARE quien tiene experiencia en la aplicación de SPC en sus procesos de manufactura, realiza un caso de estudio en el que trata de aplicar la técnica SPC de gráficos de control, en uno de sus departamentos del software. Las experiencias obtenidas fueron principalmente que: i) no tiene sentido aplicar las técnicas SPC de software completo por que cargan los proyectos con actividades de medición como la recogida, el análisis y gestión de los procesos, ii) la clave para superar las dificultades al aplicar la técnica SPC de gráficos de control en los procesos software es poner más énfasis en los procesos en lugar de producto, ii) se deben utilizar no solo los datos de productos, sino también los datos de proceso ya que estos pueden aumentar los datos disponibles de manera significativa, iii) debe existir una relación del proceso a controlar con los objetivos del negocio.

Esta propuesta proporciona una experiencia práctica de la aplicación de SPC en el contexto del software, proponiendo ciertos criterios con respecto a las métricas de producto y proceso para su utilización en SPC. En la empresa HITACHI SOFTWARE utilizan diferentes métricas para el proceso de prueba en su caso de estudio, incluyendo la Tasa de Error que es una densidad de defectos en el proceso de prueba, estas métricas software son un insumo para el presente trabajo de investigación.

#### **2.2.2.5 Evaluando la idoneidad de un repositorio de medición para el Control Estadístico de Procesos [11].**

Conforme a la literatura, este documento enseña que la aplicación de SPC requiere de medidas y datos que sean adecuados para su contexto, lo que implica uno de los principales obstáculos para una implementación exitosa de SPC en los esfuerzos de SPI (Software Process Improvement). También se hace imperiosa la actividad de medición, puesto que SPC se produce después de un programa institucionalizado de medición como un requisito de CMMI nivel 2 y nivel F del MPS.BR (Modelo de Referencia para La Mejora de Procesos Brasileño).

Aquí se explica que para las empresas, el resultado de un programa de medición son las métricas y medidas que se almacenas en un repositorio de medición, y se

espera que estas métricas y medidas sean adecuadas para su aplicación en el SPC lo cual no sucede generalmente debido a la inadecuada definición de las métricas y a las medidas recogidas. En el contexto del software las métricas deben adaptarse a las técnicas del SPC, por ello este trabajo desarrolla un IESMR (Instrument for Evaluating the Suitability of Measurement Repositories) construido a partir de la información de repositorios de empresas y evaluado con casos de estudio, cuyo objetivo es apoyar la implementación de SPC en las organizaciones que construyen software. Especifican el IESMR como una estrategia definida para dar soporte a organizaciones de manera que puedan obtener y mantener repositorios de medición adecuados para SPC, y precisan que este instrumento fue construido a partir de la Ontología de la Medición Software.

Este IESMR fundamenta nuestra propuesta de trabajo con unas listas de requerimientos y verificación que se usan en la evaluación de la idoneidad de las métricas del proceso de construcción software para ser aplicadas al SPC. También hay lista de requerimientos para evaluar la idoneidad de un plan de medición y del repositorio de medidas.

### **2.2.3 Discusión Sobre Trabajos Relacionados.**

Como se puede observar de los estudios analizados con respecto al tema a abordar en el presente trabajo de investigación, las propuestas están relacionadas con la actividad de medición, la identificación del conjunto de métricas del proceso de construcción del software, el soporte a la captura de medidas, la adecuación de las métricas al SPC y el uso de SPC en el contexto del software. Estos estudios se convierten en un referente valioso para este trabajo ya que señalan un camino a seguir para lograr los objetivos planteados en esta tesis, además se han contrastado las similitudes y diferencias con el trabajo propuesto. Sin embargo, en la literatura no se conoce de propuestas que aborden conjuntamente los conceptos de métricas del proceso de construcción del software, SPC y pequeñas organizaciones. Por tanto se resalta la importancia de abordar este tema con el fin de proporcionar a las pequeñas organizaciones de un conjunto de métricas (y una herramienta que soporte su recolección) relacionadas con su proceso de construcción de software que puedan ser utilizadas para el control estadístico de procesos.

#### **2.2.4 Aportes.**

Los principales aportes generados por el presente trabajo de investigación son:

- Estructurar un nuevo instrumento que permita evaluar de la idoneidad de una métrica para su uso en SPC dentro del contexto de las pequeñas organizaciones.
- Proponer un conjunto de métricas del proceso de construcción de software, junto con sus técnicas de recolección, que puedan ser utilizadas en la aplicación de control estadístico de procesos en las VSEs.
- Desarrollar un prototipo que de soporte a la aplicación de las técnicas propuestas para la recolección de las medidas relacionadas con las métricas identificadas en el proceso de construcción software de las VSEs, contribuyendo así a la mejora del proceso y obteniendo un producto de buena calidad en empresas que no disponen de recursos para tal fin.

# CAPÍTULO III

## CONJUNTO DE MÉTRICAS DE CONSTRUCCIÓN PARA SPC EN VSES

---

En este capítulo se presenta y ejecuta un procedimiento sistémico lógico enmarcado dentro del ciclo de investigación metodológico para la identificación de las métricas relacionadas con el proceso de construcción de software y que puedan aplicarse al control estadístico de procesos en las VSEs. Este método ejecuta tres fases: (i) Identificación del problema, donde se recopila la información más relevante que permite profundizar en los conceptos primarios que soportan la investigación, (ii) Plan de acción, en esta fase se describe el proceso seguido para la obtención del conjunto de métricas apropiadas para SPC en las VSEs, mediante la aplicación con un instrumento propuesto para tal fin, (iii) Recopilación de datos, por último se presenta el listado final de las métricas propuestas, con un detalle mayor. En adelante, la técnica SPC hace referencia a la técnica específica de gráficos de control.

### 3.1 Fases del Ciclo Metodológico.

Para determinar el conjunto de métricas del proceso de construcción adecuadas para el control estadístico de procesos en pequeñas organizaciones se ha llevado a cabo el ciclo de investigación metodológico presentado en la estrategia de investigación descrita en el Capítulo I. Este ciclo se llevó a cabo siguiendo las fases descritas en la Tabla 10.

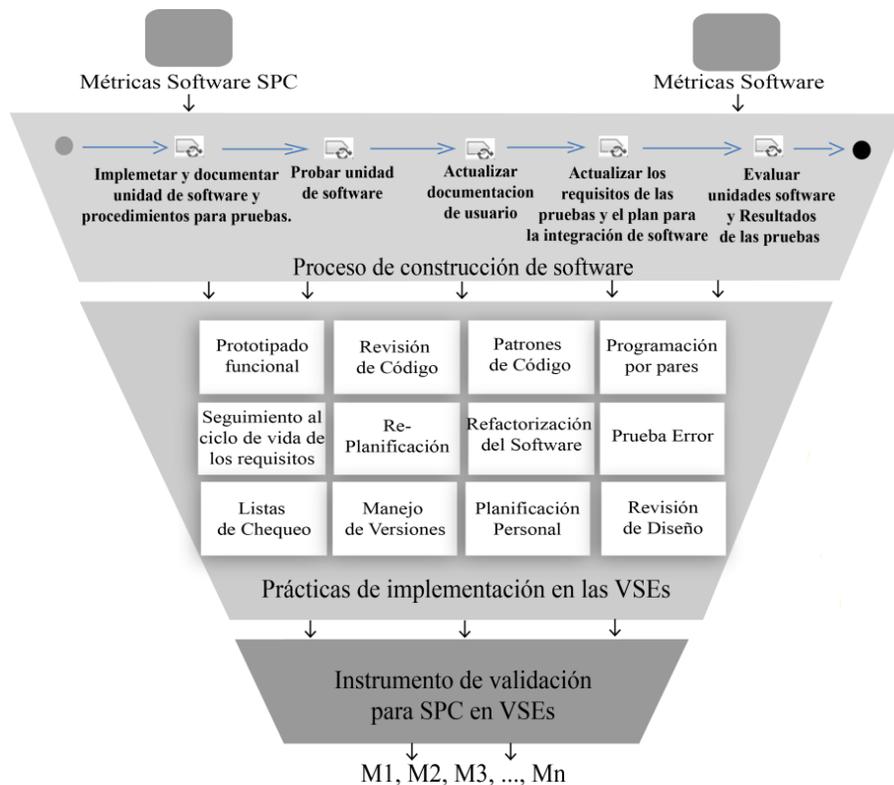
Enfocados en la metodología de investigación de [14], la Tabla 10 establece y presenta un procedimiento sistémico lógico para desarrollar y llevar a buen término la identificación de las métricas, dicho procedimiento describe un conjunto de fases que buscan, mediante su cumplimiento, responder a la pregunta de investigación ¿Qué métricas son adecuadas para gestionar el proceso de construcción desde un enfoque SPC en VSEs? En la Figura 3 se muestra una abstracción gráfica del procedimiento sistémico lógico propuesto.

A continuación, en este capítulo se presenta de manera detallada cada una de las fases descritas en la Tabla 10.

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>Fase 1.</b> <b>Identificación del Problema</b>	<ul style="list-style-type: none"> <li>• Identificar y detallar el proceso de construcción de software.</li> <li>• Identificar de las métricas utilizadas en proceso de construcción.</li> <li>• Identificar las prácticas comunes de las VSEs.</li> <li>• Identificar las métricas utilizadas en casos de estudio aplicando la técnica SPC en el desarrollo de software.</li> </ul>
<b>Fase 2.</b> <b>Plan de Acción</b>	<ul style="list-style-type: none"> <li>• Recopilar las métricas empleadas en las tareas específicas del proceso de construcción.</li> <li>• Clasificar las prácticas de las VSEs empleadas en las tareas del proceso de construcción.</li> <li>• Filtrar las métricas que cumplen con las prácticas de las VSEs ejecutadas en el proceso de construcción.</li> <li>• Recolección y clasificación de las métricas que puedan ser aplicadas a la técnica SPC en VSEs utilizando los criterios establecidos por IESRM.</li> </ul>
<b>Fase 3.</b> <b>Recopilación de datos</b>	<ul style="list-style-type: none"> <li>• Listar el conjunto de métricas adecuadas para la aplicación de la técnica SPC en VSEs, basados en las prácticas de las VSEs ejecutadas en las tareas del proceso de construcción.</li> <li>• Identificar las necesidades de información que satisfacen las métricas.</li> </ul>

**Tabla 10. Fases del procedimiento sistémico lógico empleado en el desarrollo del presente trabajo de investigación.**



**Figura 3. Abstracción del ciclo de investigación metodológico.**

### 3.2 Identificación del Problema.

#### 3.2.1 Identificar y detallar el proceso de construcción de software.

El proceso de construcción de software es un proceso definido por la normativa internacional ISO/IEC 12207 [5] y se encuentra inmerso en el grupo de procesos de Implementación de software. El propósito de este proceso es crear unidades de software ejecutables que reflejan apropiadamente el diseño del software, consta de una actividad que es la construcción de software y en la que se realizan las tareas descritas en la Tabla 11.

<u>ID</u>	<u>Tarea</u>
<b>T1</b>	Quien implementa debe desarrollar y documentar lo siguiente: i) Cada unidad de software y bases de datos y ii) Los procedimientos para las pruebas y los datos de las pruebas de cada unidad de software y base de datos.
<b>T2</b>	Probar cada unidad software y base de datos desarrollada, asegurando que satisfacen sus requerimientos. Se deberá documentar los resultados de las pruebas.
<b>T3</b>	Actualizar la documentación de usuario según sea necesario.
<b>T4</b>	Actualizar los requisitos de la prueba y el plan para la integración del código software desarrollado.
<b>T5</b>	Evaluar el código software y resultados de la prueba teniendo en cuenta lo siguiente: <ul style="list-style-type: none"> <li>○ Trazabilidad a los requerimientos y diseños de los elementos software.</li> <li>○ Consistencia externa con los requerimientos y diseños de los elementos software.</li> <li>○ Consistencia interna entre los requerimientos de las unidades.</li> <li>○ Cobertura de las pruebas de las unidades.</li> <li>○ Adecuación de los métodos de codificación y estándares usados.</li> <li>○ Viabilidad de la integración y pruebas software.</li> <li>○ Viabilidad de operación y mantenimiento.</li> </ul> Se deberán documentar los resultados de las evaluaciones.

**Tabla 11. Tareas del proceso de construcción de software definidas en la normativa ISO/IEC 12207.**

No obstante, es necesario detallar esta definición con el objetivo de realizar un análisis más preciso, que permita profundizar en el proceso para identificar los diferentes elementos usados dentro del mismo. Para esto, se extrae de la normativa ISO/IEC 15504-5 la información más detallada del proceso de construcción que se muestra en la Tabla 12, donde se describen de forma precisa las diferentes prácticas, resultados y productos de trabajo que componen al proceso.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

Proceso de Construcción de Software	
<b>Propósito</b>	El propósito del proceso de construcción de software es el de producir unidades software ejecutables, las cuales deben reflejar adecuadamente el diseño de software.
<b>Resultado</b>	<p>Como resultado de la implementación exitosa del proceso de construcción del software:</p> <p><b>R.a)</b> Criterios de verificación se definen para todas las unidades de software en contra de sus necesidades.</p> <p><b>R.b)</b> Unidades de software definidas por el diseño se producen.</p> <p><b>R.c)</b> Consistencia y la trazabilidad se establecen entre las unidades de software y los requisitos y el diseño.</p> <p><b>R.d)</b> Verificación de las unidades de software en contra de los requisitos y el diseño se lleva a cabo.</p>
<b>Practicas Base</b>	<p><b>P1) Desarrollar los procedimientos de verificación de la unidad.</b> Desarrollar y documentar procedimientos y criterios para la verificación de que cada unidad de software satisface sus requisitos de diseño. El procedimiento de verificación incluye los casos de prueba de unidad, los datos de prueba de unidad y revisión de código. <b>(Ligada con: R.a).</b></p> <p><b>P2) Desarrollar unidades de software.</b> Desarrollar y documentar las representaciones ejecutables de cada unidad de software. Actualización de los requisitos de pruebas y documentación de usuario. <b>(Ligada con: R.b).</b></p> <p><b>NOTA 1:</b> La documentación de usuario incluye las versiones preliminares de los documentos de instalación, operación y mantenimiento.</p> <p><b>P3) Garantizar la consistencia.</b> Garantizar la consistencia del diseño software con la construcción del software. La consistencia es soportada al establecer y mantener la trazabilidad entre los requisitos software, y el diseño contra las unidades software cuando sea necesario. <b>(Ligada con: R.c).</b></p> <p><b>P4) Verificar unidades software.</b> Compruebe que cada unidad de software satisface sus requisitos de diseño mediante la ejecución de los procedimientos de verificación específicos para las unidades y documentar los resultados. <b>(Ligada con: R.d).</b></p> <p><b>NOTA 2:</b> El código puede ser verificado mediante diversas técnicas tales como el análisis estático de código, revisión de código, etc.</p>
Productos de Trabajo	

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

Entradas	Salidas
Datos de Prueba. <b>(Ligado con: R.d).</b>	Datos de Prueba. <b>(Ligado con: R.a).</b>
Diseño software en bajo nivel. <b>(Ligado con: R.b, R.c).</b>	
	Manual del cliente. <b>(Ligado con: R.b).</b>
	Plan de prueba para unidad. <b>(Ligada con: R.a).</b>
	Procedimiento de prueba. <b>(Ligado con: R.a).</b>
Unidad software. <b>(Ligado con: R.c, R.d).</b>	Unidad software. <b>(Ligado con: R.b).</b>
	Registro de revisión. <b>(Ligado con: R.d).</b>
	Registro de trazabilidad. <b>(Ligado con: R.c).</b>
	Registro (Log) Pruebas. <b>(Ligado con: R.d).</b>
	Reporte de incidentes de prueba. <b>(Ligado con: R.d).</b>
Requerimientos de interfaz. <b>(Ligado con: R.a, R.c).</b>	
Requerimientos de software. <b>(Ligado con: R.a, R.c).</b>	
Especificación de diseño de pruebas. <b>(Ligado con: R.a).</b>	
Especificación de casos de pruebas. <b>(Ligado con: R.b).</b>	Especificación de casos de pruebas. <b>(Ligado con: R.b).</b>
Estándar de codificación. <b>(Ligado con: R.b, R.c, R.d).</b>	

**Tabla 12. Proceso de construcción software en detalle (extraído de norma ISO/IEC 15504-5).**

Teniendo ya la definición detallada del proceso se realizó el modelo (ver Figura 4) de un proceso de construcción de software para identificar y estructurar las diferentes tareas y productos de trabajo constitutivos del mismo. Este modelo también es propuesto por [35] en un diagrama de flujo, sin embargo en el modelo propuesto se pueden encontrar algunos aportes que surgen de un análisis más profundo en la definición del proceso. El modelo ha sido creado explícitamente para contrastar el proceso con las prácticas de las pequeñas organizaciones (lo cual se presenta en la sección 3.2.2).



### 3.2.2 Identificar las métricas utilizadas en proceso de construcción.

Para nuestro estudio es de suma importancia contar con un listado de métricas utilizadas en el proceso de construcción de software, estas métricas son el insumo primario de la investigación y se deben seleccionar de fuentes importantes en el ámbito de software cumpliendo unos criterios de selección que han sido definidos a partir del trabajo presentado en [36] y su búsqueda es basada en el método de revisiones sistemáticas propuesto en [37] [4]. Este trabajo propone un marco conceptual de atributos, métricas y heurísticas de calidad de software para la valoración del producto software orientado a objetos. Los criterios de selección definidos permitieron seleccionar los trabajos que son de interés para nuestro trabajo de investigación y son listados a continuación:

- **C1:** Disponibilidad, grado en que es posible acceder a la información existente. La escala para evaluación es:
  1. La información no está disponible al público en general.
  2. Hay disponibilidad de algunos documentos pero es limitado el acceso.
  3. Existe información suficiente disponible para ser usada.
- **C2:** Completitud: Grado en que la información presentada describe métricas de software sin dejar por fuera información importante como: definición, propósito y método (para la medición y calculo) de cada métrica. La escala para evaluación es:
  1. No se menciona toda la información necesaria con respecto a las métricas. La información de estas métricas es incompleta.
  2. La información sobre las métricas software se describe parcialmente, sin embargo deja algunos elementos importantes por fuera.
  3. La información de las métricas software describe todos los elementos y la información está completa
- **C3:** Área de Aplicación, la información sobre las métricas es aplicable a las temáticas de nuestra investigación.
  1. Incluye información métricas de software
  2. Incluye información métricas proceso de construcción
  3. Incluye información métricas proceso de construcción en VSEs

En la Tabla 13 se listan las fuentes que pueden ser accedidas y se encuentran disponibles en el servicio Web de internet de la Universidad del Cauca para la búsqueda de las métricas. La estrategia utilizada para la búsqueda en las fuentes de la Tabla 13 se forma a través de la combinación de palabras claves descritas a continuación, cabe aclarar que la búsqueda también se realizó en español:

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

- "Software metrics" AND (small AND(enterprises OR organizations OR development OR build OR firms OR settings ) OR "Statistical process control").
- "Software measures" (small AND(enterprises OR organizations OR development OR build OR firms OR settings)).

ID FUENTE	Fuente
F1	Science@Direct en el tema de Computer Science,
F2	Proquest
F3	Springer, Part of Springer Science+Business Media
F4	Engineering Village
F5	Grupo Alarcos
F6	AIRCC Publishing Corporation
F7	IEEE Xplore Digital Library

**Tabla 13. Fuentes de información primarias.**

Una vez ejecutada la búsqueda para encontrar los trabajos de interés para nuestra investigación, se obtuvieron 25 trabajos. A estos trabajos se les aplicó los criterios antes definidos con el fin de seleccionar los que son relevantes para la investigación (con una calificación mayor a 5 puntos en el total), obteniendo en definitiva 7 trabajos los cuales son listados en la Tabla 14.

Fuente	Información	Criterios			Total
		C1	C2	C3	
Grupo Alarcos	Medidas de calidad en proceso, producto y mantenimiento, aplicadas al control estadístico de procesos.	3	3	1	7
Springer	Investigating suitability of software process and metrics for statistical process control	3	2	1	6
	Calidad de productos de software: un estado del arte de la medición	3	2	2	7
	Incorporación de medidas en el modelo de procesos para la industria de software MoProSoft	3	3	3	9
AIRCC	Statistical analysis on software metrics affecting modularity in open source software	2	2	1	5
IEEE	Statistical process control: analyzing a space shuttle onboard software process	3	1	1	5
Proquest	Utilization of statistical process control (SPC) in emergent software organizations: pitfalls and suggestions	3	2	2	7

**Tabla 14. Listado de fuentes más relevantes para el presente trabajo de investigación.**

Teniendo en cuenta las calificaciones obtenidas en la Tabla 14 de los estudios relacionados con métricas de software, tomamos como referente principal al estudio con título: "Incorporación de medidas en el modelo de procesos para la industria de software MoProSoft", Este estudio recopila un conjunto de métricas de

los procesos de Gestión de Proyectos, Desarrollo de software y Mantenimiento software enfocados en la norma Mexicana MoProSoft.

### **3.2.3 Identificar las prácticas comunes de las VSEs.**

#### **3.2.3.1 Características de las VSEs.**

La industria del software nacional e internacional está conformada en su gran mayoría por VSEs. De [30] una VSEs (Very Small Entities) es una entidad (empresa, organización, departamento o proyecto) conformada por hasta 25 personas dedicada al desarrollo de software y/o al mantenimiento software que es utilizado en sistemas grandes, por lo tanto, el reconocimiento de las VSEs como proveedores de software de alta calidad es requerido de manera frecuente. Según [4] las VSEs:

- Poseen una organización flexible, no cuentan con departamentos especializados, sus recursos económicos son limitados, esperando que el retorno a la inversión sea corto, su acceso a financiación bancaria es muy poca.
- En general no pueden invertir en asesores. Sus empleados son pocos y realizan múltiples funciones, y la responsabilidad no está bien definida. Existe una gran dependencia de los individuos.
- Desconocen la importancia del proceso de desarrollo sobre la calidad del producto, recurriendo a prácticas artesanales de construcción.
- Su desarrollo está enfocado a usuarios conocidos especializándose en un área de trabajo específica (transporte, salud, contabilidad, etc.)
- Manejan pocos proyectos que no consuman mucho tiempo, con una planeación mínima o a corto plazo.

#### **3.2.3.2 Prácticas de Implementación de las VSEs.**

En [38] se cuenta con un conjunto de prácticas (ver Tabla 15) que son ejecutadas por las VSES. Estas prácticas pertenecen a la disciplina de Implementación, donde se implementa el código fuente, haciendo uso de prototipos así como de pruebas y ensayos para corregir errores. Cabe resaltar que el estudio fue tomado debido a que representaba un perfil acerca de la utilización de las prácticas más empleadas en las VSEs para construir software y fue utilizado por [2] para analizar las prácticas y los procesos claves para la mejora de procesos software en pequeñas empresas.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

Las prácticas de la Tabla 15 se pueden tomar como base para tener una visión general de las actividades comunes en las VSES, lo que permite filtrar y categorizar las métricas identificadas en la sección 3.2.1, con el fin de lograr que dichas métricas sean útiles para este tipo de organizaciones.

ID	Nombre
P1	Prototipado funcional
P2	Revisión de Código
P3	Patrones de Código
P4	Listas de Chequeo
P5	Manejo de Versiones
P6	Planificación Personal
P7	Revisión de Diseño
P8	Seguimiento al ciclo de vida de los requisitos
P9	Re-Planificación
P10	Refactorización del Software
P11	Programación por pares
P12	Prueba Error

Tabla 15. Prácticas más ejecutadas por las VSEs.

### 3.2.4 Identificar las métricas utilizadas en casos de estudio aplicando la técnica SPC en el desarrollo de software.

Siguiendo con las actividades del ciclo de investigación metodológico, fue importante conocer el estado de los estudios realizados en el área de SPC dentro de la industria del software. Mediante esta actividad se pudo identificar un grupo de métricas (ver Tabla 16) usadas en el control del proceso de desarrollo software aplicando la técnica SPC de gráficos de control, que es la técnica más empleada para controlar y mejorar los procesos software debido a su efectividad y su aporte como una interfaz visual basada en fundamentos científicos [23]. Cabe resaltar que varios de los documentos analizados, exponen un listado de métricas de producto, proyecto y proceso en el desarrollo del software que pueden ser aplicadas a las técnicas SPC, pero solo se particularizan aquellas métricas que fueron aplicadas a la técnica específica de gráficos de control en casos de estudio.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

Métrica	Formula	Descripción	Referencia
Integración de código	LOC del módulo liberado/Total LOC del Proyecto.	<p>Es el porcentaje del total de líneas de código (LOC) de todo el proyecto de software. Esta métrica crece de 0 por ciento a 100% con un patrón de aumento con el tiempo.</p> <p>La integración de código es el proceso de integrar el código software del escritorio de un desarrollador a la biblioteca del sistema y así generar una liberación temporal del producto para sus pruebas.</p>	[39]
Densidad de defectos	Número Defectos / Tamaño Producto	<p>Captura el número de defectos en un producto de trabajo, este conteo de defectos brinda evidencia no sólo de la calidad del producto, sino también de la calidad del proceso relacionado [23]. Para obtener la densidad de defectos se normaliza la información de los defectos con la dimensión tamaño. La acumulación y descubrimiento de defectos son medidas que se pueden derivar del trabajo de capturar la densidad de defectos.</p>	[39], [8], [23], [12], [19], [21], [18], [24].
Tasa de detección de errores temprana.	Número de errores detectados/Total de errores	<p>Esta medida es una relación entre el número de errores detectados por exámenes sobre todos los errores de la revisión. Se utiliza para decidir la cantidad de esfuerzo que un proyecto debe poner en los exámenes del código fuente. Una medida similar es el "Rendimiento de inspección".</p>	[19]
Porcentaje de Retrabajo	Esfuerzo de Retrabajo / Esfuerzo Total	<p>El retrabajo muestra la magnitud de esfuerzo gastado (inversión total de horas) en cambios imprevistos o errores del código software. El esfuerzo de repetición de trabajo se puede aplicar a las diferentes instancias de procesos, por ejemplo en [23] el porcentaje de retrabajo es igual al esfuerzo de revisión sobre esfuerzo total para las actividades de Testing. En otras palabras, esta métrica permite llevar a cabo un análisis para cualquier grupo de tareas que representen un dominio específico dentro del desarrollo del software.</p>	[8], [23], [21], [18].

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

Rendimiento de Inspección	Número defectos encontrado / Esfuerzo de Inspección	La inspección es el examen formal de un producto para identificar defectos. Permite a los especialistas en software, entender y corregir los errores antes de que el producto se libere. Ayuda a aumentar la calidad del producto y a reducir la cantidad de retrabajo. En [19] [18] a esta medida la llaman "Eficiencia de Revisión".	[8], [23], [19], [21], [18].
Velocidad de Revisión	Tamaño del producto de trabajo/Horas gastadas para su revisión	Se define como el tamaño del producto de trabajo revisado por horas gastadas para su revisión. Se utiliza para verificar qué tan bien se lleva a cabo la revisión, nos muestra que tan rápida está siendo la velocidad de revisión y la frecuencia con que los revisores pierden los errores.	[19], [40]
Productividad	Tamaño del producto de trabajo producido/unidad de esfuerzo.	En términos generales, la productividad es la cantidad de producto generado por insumos gastados. Como lo hacen en [10], para el contexto del software podría calcularse el número de líneas de código por programa, junto el esfuerzo (horas de trabajo) de cada programa para representar el rendimiento del desarrollo en un proyecto o la productividad del desarrollador.	[21], [10].

**Tabla 16. Métricas software utilizadas en la aplicación de SPC con gráficos de control.**

Algunas de estas métricas tienen una relación más fuerte con el proceso de construcción puesto que las podemos ligar con una (o varias) de sus tareas (ver Tabla 11), dichas métricas son: Densidad de Defectos, Porcentaje de Retrabajo, Productividad. Estas métricas son usadas para la aplicación práctica (y/o verificación) del elemento de verificación propuesto por el IESRM y adaptado a las necesidades del presente trabajo de investigación, este punto se explica en más detalle en 3.2.4. El resto de métricas de la tabla no tienen una relación muy fuerte con el proceso de construcción de software debido a que su aplicación requiere del producto integrado lo que esta fuera del objetivo de dicho proceso.

### 3.3 Plan de Acción.

#### 3.3.1 Recopilar las métricas empleadas en las tareas específicas del proceso de construcción.

En la revisión de la literatura presentada en [41] se plantea que la actividad de medición es necesaria para validar las propiedades y características de un producto. Esta actividad sirve para entender, evaluar, mejorar, controlar los productos y procesos de software, y facilita la toma de decisiones oportuna.

Para el presente trabajo de investigación la medición se centra en el proceso de construcción de software enfocándose en los atributos del producto tal como lo sugiere [41], en este proceso las unidades de software se codifican, lo que facilita recolectar información cuantitativa de las mismas como los indicadores de seguimiento y control (defectos, complejidad, tiempo y tamaño) [23], de esta manera se hace necesaria la utilización de métricas en la actividad de medición.

Para identificar las métricas del proceso de construcción, el punto de partida es la revisión sistemática [42] obtenida en la sección 3.2.2, que presenta una propuesta de incorporación de medidas en el modelo MoProSoft (esta norma mexicana fomenta la estandarización de operación en las pequeñas o medianas organizaciones dedicadas al desarrollo y/o mantenimiento de software, con o sin procesos establecidos). Esta propuesta se enfocó en determinar las métricas para las actividades de los procesos de:

- i) Gestión de Proyectos.
- ii) Desarrollo de software.
- iii) Mantenimiento software.

A continuación en la Tabla 17 se listan aquellas métricas enfocadas al proceso de construcción de software (insumo principal de métricas del presente trabajo de investigación):

ID	Medida	Descripción
MC1	FP	Puntos de función
MC2	LOC	Líneas de código
MC3	LOC+NLOC	Tamaño total del programa (incluyendo las líneas que son comentario nloc)
MC4	MHK	Métrica complejidad de expansión
MC5	Tiempo en Corregir	Tiempo en corregir defectos
MC6	EV	Valor ganado

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>MC7</b>	COQ	Costo por la calidad
<b>MC8</b>	Valoración de COQ	Valoración costo por la calidad
<b>MC9</b>	COQ TOTAL	Costo por la calidad total
<b>MC10</b>	$100 * (\text{valoración COQ}) / \text{COQ TOTAL}$	Valoración porcentual costo por la calidad total
<b>MC11</b>	Defectos/PF o Defectos/(K)LOC	Densidad de defectos
<b>MC12</b>	Tasa de Defectos	Defectos encontrados/tamaño
<b>MC13</b>	Tasa de Revisión	Tamaño del producto/tiempo revisión
<b>MC14</b>	Campo de Revisión	Porcentajes de defectos de un programa
<b>MC15</b>	LOC/unidad de tiempo o PF/Unidad de tiempo	Productividad
<b>MC16</b>	Esfuerzo	Unidad de tiempo-hombre
<b>MC17</b>	QC	Calidad del código desarrollado
<b>MC18</b>	QP	Calidad del producto entregado y probado
<b>MC19</b>	$W_{\text{tpp}} / \text{KCSI}$	Tiempo de mejoras
<b>MC20</b>	$T_t / \text{KCSI}$	Tiempo de prueba
<b>MC21</b>	$(T_p / T_d) * 100\%$	Tiempo de prueba sobre tiempo desarrollo
<b>Medidas enfocadas en el paradigma estructurado</b>		
<b>ID</b>	<b>Medida</b>	<b>Descripción</b>
<b>MC22</b>	u1	Número de operadores diferentes que aparecen en el programa
<b>MC23</b>	u2	Numero de operandos diferentes que aparecen en el programa
<b>MC24</b>	U	Vocablos del programa
<b>MC25</b>	N1	Número total de veces que aparece el operador
<b>MC26</b>	N2	Número total de veces que aparece el operando
<b>MC27</b>	N	Longitud del programa
<b>MC28</b>	$V = N \times \log_2 u$	Volumen del programa
<b>MC29</b>	N COPY	Numero de instrucciones copiadas
<b>MC30</b>	N PERF	Número de sentencias ejecutadas
<b>MC31</b>	N SEL	Numero sentencias de selección
<b>MC32</b>	N CYCLE	Número de sentencias cíclicas
<b>MC33</b>	N EXEC	Número de ejecuciones dinámicas
<b>MC34</b>	Block COV	Porcentaje de bloques ejecutados
<b>MC35</b>	ST COV	Porcentaje de sentencias ejecutadas
<b>MC36</b>	ILOC	Líneas de código separadas por punto y coma
<b>MC37</b>	Funciones	Numero de funciones
<b>MC38</b>	FP	Numero de parámetros formales de las funciones
<b>MC39</b>	FR	Número de puntos de retorno de las funciones
<b>MC40</b>	IC	Complejidad del programa
<b>MC41</b>	CO	Número de operadores de comparación

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>MC42</b>	AweBlockL	Longitud promedio de los bloques básicos
<b>MC43</b>	Y(G)	Complejidad ciclo matica basada en teoría de grafos
<b>Medidas enfocadas en el paradigma orientado a objetos</b>		
<b>ID</b>	<b>Medida</b>	<b>Descripción</b>
<b>MC44</b>	CBO	Acoplamiento entre clases, si los métodos de una clase utiliza métodos de otras clases, o viceversa.
<b>MC45</b>	RFC	Conjunto de respuesta para una clase, conjuntó de métodos que directa e indirectamente potencialmente invocan métodos de otra clase
<b>MC46</b>	MPC	Numero de invocaciones de un método en una clase
<b>MC47</b>	DAC	Numero de atributos en una clase que tienen como tipo otra clase
<b>MC48</b>	ICP	Numero de métodos invocados en una clase, ponderados por el número de parámetros de los métodos invocados
<b>MC49</b>	IH-ICP	Como icp, pero contiene invocaciones de métodos de los ancestros de esa clase
<b>MC50</b>	LCOM	Falta de cohesión en métodos: número de pares de métodos que no usan ningún atributo en común.
<b>MC51</b>	TCC	Se define como el porcentaje de métodos públicos de una clase que se encuentran conectados por atributos directos o indirectos
<b>MC52</b>	LCC	El porcentaje de métodos públicos que se encuentran conectados directamente o indirectamente
<b>MC53</b>	ICH	El número de invocaciones de otros métodos de la misma clase, ponderados por el número de parámetros de la invocación
<b>MC54</b>	DIT	Longitud del camino más largo de la clase a su raíz a si árbol de herencia
<b>MC55</b>	AID	Promedio de profundidad en clases padres
<b>MC56</b>	CLD	Máximo número de niveles en la herencia que le pertenecen a una clase
<b>MC57</b>	NOC	Número de hijos de una clase
<b>MC58</b>	NOP	número de padres inmediatos de una clase
<b>MC59</b>	NOD o NOAC	número de descendientes de una clase
<b>MC60</b>	NOA	número de ancestros de una clase
<b>MC61</b>	NMO	número de métodos sobre escritos de una clase
<b>MC62</b>	NMI	número de métodos heredados de una clase
<b>MC63</b>	NMA	número de nuevos métodos de una clase que no son heredados
<b>MC64</b>	SIX	índice de especialización
<b>MC65</b>	NM	numero de métodos totales de una clase
<b>MC66</b>	NAI	número total de atributos de una clase

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>MC67</b>	NumPub	número de métodos públicos de una clase
<b>MC68</b>	NumPara	número total de parámetros de una clase
<b>MC69</b>	APPM	promedio de parámetros por método

**Tabla 17. Listado de métricas de la revisión sistemática.**

Cabe resaltar que el proceso de desarrollo en el modelo Moprosoft está alineado con los objetivos del proceso de construcción de software de la norma ISO /IEC 12207.

### 3.3.2 Clasificar las prácticas de las VSEs empleadas en las tareas del proceso de construcción.

Como ya se describió en la sección 3.2.3.2, en [38] se plantea una panorámica del estado de las prácticas ejecutadas en varias empresas representativas de la región suroccidental colombiana, estas empresas centran su atención en la aplicación de las disciplinas relacionadas con el grupo de procesos de ingeniería (requisitos, análisis y diseño, implementación y pruebas software). Para el presente trabajo de investigación las practicas listadas en la disciplina de implementación (ver Tabla 15), son una visión general de las actividades comunes que realizan las VSEs con el propósito de generar unidades de software, de manera que se pueden ligar con las tareas del proceso construcción de software (ver Tabla 11) definido en la norma ISO/IEC 12207.

A continuación en la Tabla 18, se describen las prácticas de implementación en las VSEs, y su correlación con las tareas del proceso de construcción de software:

<b>ID</b>	<b>P1</b>
<b>NOMBRE</b>	<b>Prototipado funcional</b>
<b>DESCRIPCION</b>	El prototipo supone una primera versión del sistema con funcionalidad limitada. A medida que se comprueba si las funciones implementadas son las apropiadas, se corrigen, refinan o se añaden otras nuevas, hasta llegar al sistema final [43].
<b>Relación P1 con Tarea(s) del proceso de construcción ISO 12207.</b>	T1
<b>JUSTIFICACION</b>	El Prototipado funcional requiere implementar el sistema con funcionalidad limitada (Unidades de Software y Bases de Datos), incluyendo los procedimientos para probar dicho prototipo.
<b>ID</b>	<b>P2</b>

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>NOMBRE</b>	<b>Revisión de Código</b>
<b>DESCRIPCION</b>	Su objetivo es evaluar un producto intermedio del desarrollo para comprobar que se ajusta a sus especificaciones, el desarrollo se está realizando de acuerdo a los planes, estándares y guías aplicables al proyecto. Se clasifican en Revisiones informales de código, Revisiones formales [43].
<b>Relación P2 con Tarea(s) del proceso de construcción ISO 12207.</b>	T2, T4
<b>JUSTIFICACION</b>	Al contar con una revisión del código se asegura en gran medida que los requerimientos se satisfacen en cada unidad de software y base de datos desarrollados, actualizando los requisitos de prueba para poder evaluar el código software.
<b>ID</b>	<b>P3</b>
<b>NOMBRE</b>	<b>Patrones de Código</b>
<b>DESCRIPCION</b>	Los patrones de software son reglas o guías para resolver un problema dentro de un contexto. La solución no es determinística sino que ofrece un modelo de actuación que quien lo aplica debe adaptar a la situación concreta en la que se encuentra [44].
<b>Relación P3 con Tarea(s) del proceso de construcción ISO 12207.</b>	T5
<b>JUSTIFICACION</b>	Permite evaluar y estandarizar los métodos de codificación y los estándares usados.
<b>ID</b>	<b>P4</b>
<b>NOMBRE</b>	<b>Listas de Chequeo</b>
<b>DESCRIPCION</b>	Las listas de Chequeo son aquellas formadas por un conjunto de preguntas sencillas sobre características que deberá cumplir el producto con dos posibles respuestas ("SI" o "NO"). Las confeccionan personas especializadas en base a su experiencia y en base a los resultados de inspecciones previas sobre productos similares, el objetivo es detectar los defectos más importantes. Además, los productos intermedios deben revisarse basándose en cuatro aspectos: completación, integridad, exactitud, y calidad [43].
<b>Relación P4 con Tarea(s) del proceso de</b>	T5

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>construcción ISO 12207.</b>	
<b>JUSTIFICACION</b>	Las listas de Chequeo Permiten Evaluar el Código software para verificar la trazabilidad a los requerimientos y diseños de las unidades software basándose en cuatro aspectos: completión, integridad, exactitud, y calidad.
<b>ID</b>	<b>P5</b>
<b>NOMBRE</b>	<b>Manejo de Versiones</b>
<b>DESCRIPCION</b>	Control de Versiones: es la capacidad de proporcionar almacenamiento y acceso controlado a los productos de trabajo, así como de registrar los cambios sobre los mismos y poder recuperar versiones anteriores [43].
<b>Relación P5 con Tarea(s) del proceso de construcción ISO 12207.</b>	T5
<b>JUSTIFICACION</b>	Facilita la Viabilidad de operación y el mantenimiento al proporcionar un acceso controlado y un registro de los cambios.
<b>ID</b>	<b>P6</b>
<b>NOMBRE</b>	<b>Planificación Personal</b>
<b>DESCRIPCION</b>	Esta práctica se basa en PSP (Personal Software Process) el cual es adaptado por los individuos, por lo tanto no es una imposición organizacional, es aplicable a desarrolladores que trabajan de forma independiente o como equipos de trabajo, su producto son componentes software, las actividades de PSP son: Planificación, Diseño, Revisión de diseño, Codificación, Revisión de Código, Compilación y Prueba Unitaria Post mortem [45].
<b>Relación P6 con Tarea(s) del proceso de construcción ISO 12207.</b>	T1 , T2, T4, T5
<b>JUSTIFICACION</b>	Es un proceso inherente a cada individuo que influye en la construcción del software.
<b>ID</b>	<b>P7</b>
<b>NOMBRE</b>	<b>Revisión de Diseño</b>
<b>DESCRIPCION</b>	Según la IEEE 610.12, una revisión es un proceso o reunión durante la cual un producto de trabajo, un conjunto de productos de trabajo o la evidencia de la ejecución de un proceso se presenta al equipo del proyecto, a los administradores,

MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

	usuarios, clientes u otras partes interesadas para su comentario o aprobación.
<b>Relación P7 con Tarea(s) del proceso de construcción ISO 12207.</b>	T3
<b>JUSTIFICACION</b>	Permite ajustar los requerimientos de los usuarios actualizando la documentación de las necesidades de los usuarios.
<b>ID</b>	<b>P8</b>
<b>NOMBRE</b>	<b>Seguimiento al ciclo de vida de los requisitos</b>
<b>DESCRIPCION</b>	La Gestión de Requisitos es el conjunto de actividades que ayudan al equipo de trabajo a identificar, controlar y seguir los requisitos y sus cambios en cualquier momento. O sea, básicamente, consiste en gestionar los cambios a los requisitos acordados, las relaciones entre ellos, las dependencias entre la Especificación de Requisitos del Software (ERS) y otros documentos producidos por el proceso de desarrollo de software. Esta actividad asegura la consistencia entre los requisitos y el sistema construido (o en construcción). Consume grandes cantidades de tiempo y esfuerzo. Abarca todo el ciclo de vida del producto [46].
<b>Relación P8 con Tarea(s) del proceso de construcción ISO 12207.</b>	T5
<b>JUSTIFICACION</b>	Permite controlar la evolución de los requerimientos para garantizar la trazabilidad y consistencia de los requerimientos con el código y el diseño.
<b>ID</b>	<b>P9</b>
<b>NOMBRE</b>	<b>Re-Planificación</b>
<b>DESCRIPCION</b>	Es la actividad de refinar las estimaciones razonables de recursos, coste y planificación temporal. Las estimaciones se hacen dentro de un tiempo limitado al comienzo de un proyecto software, y con esta actividad deben actualizarse con el progreso del proyecto [28].
<b>Relación P9 con Tarea(s) del proceso de construcción ISO 12207.</b>	T5
<b>JUSTIFICACION</b>	Permite verificar la Viabilidad de la integración software y la operación.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>ID</b>	<b>P10</b>
<b>NOMBRE</b>	<b>Refactorización del Software</b>
<b>DESCRIPCION</b>	En ingeniería del software, el término refactorización se usa a menudo para describir la modificación del código fuente sin cambiar su comportamiento, lo que se conoce informalmente por limpiar el código. La refactorización se realiza a menudo como parte del proceso de desarrollo del software: los desarrolladores alternan la inserción de nuevas funcionalidades y casos de prueba con la refactorización del código para mejorar su consistencia interna y su claridad. Los tests aseguran que la refactorización no cambia el comportamiento del código [28].
<b>Relación P10 con Tarea(s) del proceso de construcción ISO 12207.</b>	T5
<b>JUSTIFICACION</b>	Facilita las operaciones de Mantenimiento del software.
<b>ID</b>	<b>P11</b>
<b>NOMBRE</b>	<b>Programación por pares</b>
<b>DESCRIPCION</b>	<p>La Programación en Pareja requiere que dos programadores participen en un esfuerzo combinado de desarrollo en un sitio de trabajo. Cada miembro realiza una acción que el otro no está haciendo actualmente: Mientras que uno codifica las pruebas de unidades el otro piensa en la clase que satisfará la prueba, por ejemplo.</p> <p>La persona que está haciendo la codificación se le da el nombre de controlador mientras que a la persona que está dirigiendo se le llama el navegador. Se sugiere a menudo para que a los dos socios cambien de papeles por lo menos cada media hora o después de que se haga una prueba de unidad [28].</p>
<b>Relación P11 con Tarea(s) del proceso de construcción ISO 12207.</b>	T1,T2,T4
<b>JUSTIFICACION</b>	Permite la codificación de unidades de software por parte del controlador, con la supervisión del navegador quien elabora las pruebas.
<b>ID</b>	<b>P12</b>

MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

NOMBRE	Prueba Error
DESCRIPCION	Es un método heurístico para la obtención de conocimiento, tanto proposicional como procedural. Consiste en probar una alternativa y verificar si funciona. Si es así, se tiene una solución. En caso contrario (resultado erróneo) se intenta una alternativa diferente [43].
Relación P12 con Tarea(s) del proceso de construcción ISO 12207.	T1,T2
JUSTIFICACION	Permite codificar y asegurar que cada unidad de software y bases de datos desarrolladas de manera muy empírica cumplan sus requerimientos.

**Tabla 18. Prácticas de las VSEs y su relación con las tareas del proceso de construcción**

En la Tabla 19, se lista el resumen de la correlación entre las tareas del proceso de construcción de software con cada una de las prácticas de las VSES.

ID	Práctica	Relación con Tarea(s) del proceso de construcción ISO 12207
P1	Prototipado funcional	T1
P2	Revisión de Código	T2,T4
P3	Patrones de Código	T5
P4	Listas de Chequeo	T5
P5	Manejo de Versiones	T5
P6	Planificación Personal	T1 , T2, T4, T5
P7	Revisión de Diseño	T3
P8	Seguimiento al ciclo de vida de los requisitos	T5
P9	Re-Planificación	T5
P10	Refactorización del Software	T5
P11	Programación por pares	T1,T2,T4
P12	Prueba Error	T2,T1

**Tabla 19. Correlación entre las tareas del proceso de construcción de software y las prácticas de las VSEs**

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

Otra perspectiva de la correlación entre las tareas de construcción y las prácticas de las VSEs se muestra a continuación en la Tabla 20:

Tarea del proceso de construcción ISO 12207	Prácticas de las VSES
T1	P1,P6,P11,P12
T2	P2,P6,P11,P12
T3	P7
T4	P2,P6,P11
T5	P3,P4,P5,P6,P8,P9,P10

Tabla 20. Otra perspectiva de la correlación entre las tareas del proceso de construcción de software y las prácticas de las VSEs.

### 3.3.3 Filtrar las métricas que cumplen con las prácticas de las VSEs ejecutadas en el proceso de construcción.

Partiendo de los resultados obtenidos en la Tabla 19 y continuando con la metodología de investigación propuesta (ver Tabla 10), las métricas obtenidas en la revisión sistemática deben ser filtradas con las prácticas más comunes de las VSEs, buscando que las métricas sean útiles para este tipo de organizaciones.

#### 3.3.3.1 Filtrado de Métricas.

Al no contar con toda la información de las métricas expuestas en la Tabla 17, se hace necesaria una búsqueda en las diferentes fuentes (libros, internet, bases datos) para identificar en que práctica de la Tabla 19 pueden ser clasificadas. Para el filtro, las métricas deben cumplir con al menos una práctica específica de las VSEs. En la Tabla 21 se lista la relación de las métricas con las prácticas de las VSEs.

En la Tabla 22 se listan los resultados organizados por las prácticas definidas en la Tabla 19 y su relación con las métricas.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

Métrica	Práctica	Justificación
MC1	P1	Esta métrica se define como una métrica funcional, dado que se enfoca a la funcionalidad que el SW proporciona al usuario [47]. Permite medir el prototipo funcional que representa los requerimientos del usuario.
MC2	P2	Al revisar el código, es posible determinar cuantas líneas de código tiene el SW.
MC3	P10	Al limpiar el código se puede conocer el tamaño total.
MC4	P7	Permite medir la complejidad del SW y cómo esta complejidad influye en el esfuerzo de integración y de pruebas [28]
MC5	P2	Al revisar el código y encontrar un defecto se mide el tiempo empleado para corregir el defecto (falta verificada de Conformidad con los requisitos.) [48].
MC6	P6	Permite medir que tanto trabajo actualmente se ha completado [49].
MC7 MC8 MC9 MC10	No Cumple	COQ mide la calidad del proceso de una forma significativa para la gerencia [50], estas métricas se deben tomar en el proceso de Pruebas Software [35] donde se cuenta con el producto software integrado y permite conocer el costo de inspeccionar el producto SW buscando defectos.
MC11	P11, P2,P12	Se refiere al número de imperfecciones por Kilo-líneas De código o Puntos de Función en la codificación de una o varias unidades de software [23].
MC12 MC13	No Cumple	Para estas métricas es necesario contar con el tamaño del SW y este solo se puede determinar después del proceso de Integración, por lo que las métricas pertenecen al proceso de Pruebas Software [35]
MC14	No Cumple	No se encontró información de la definición de la métrica
MC15	P9	La métrica se centra en conocer el rendimiento de las funciones del desarrollo [50], y permite conocer la productividad para tomar decisiones en RE Planificación.
MC16	P6, P1, P11,P12	La métrica representa el número de horas, minutos, días consumidas para realizar la codificación de las unidades de software [50].
MC17 – MC18- MC19-MC20- MC21	No Cumplen	El propósito de las métricas es ofrecer un producto de alta calidad y requieren del SW integrado [51] y sus medidas deben ser tomadas en el proceso de Pruebas Software [35].
Medidas Enfocadas en el paradigma Estructurado MC22.... MC43	P6, P1, P11	Las métricas están enfocadas en la codificación de unidades de software utilizando el paradigma estructurado.
Medidas Enfocadas en el paradigma Estructurado MC44.... MC69	P6, P1, P11	Las métricas están enfocadas en la codificación de unidades de software utilizando el paradigma Orientado a Objetos.

**Tabla 21. Métricas del proceso de construcción y su relación con las Prácticas de las VSEs**

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

ID	Relación con Tarea(s) del proceso de construcción ISO 12207	Métricas
P1	T1	MC1,MC16,[MC22,...,MC43],[MC44,...,MC69]
P2	T2,T4	MC2,MC5,MC11
P3	T5	
P4	T5	
P5	T5	
P6	T1 , T2, T4, T5	MC6,MC16,[MC22,...,MC43],[MC44,...,MC69]
P7	T3	MC4
P8	T5	
P9	T5	MC15
P10	T5	MC3
P11	T1,T2,T4	MC11,MC16,[MC22,...,MC43],[MC44,...,MC69]
P12	T2,T1	MC16,MC11

**Tabla 22. Relación práctica, proceso de construcción y métricas.**

En la Tabla 23 se expresa otra perspectiva de la relación práctica, proceso de construcción y métricas.

Tarea del proceso de construcción ISO 12207	Prácticas de las VSES	Métricas
T1	P1,P6,P11,P12	MC1, MC6, MC11, MC16, [MC22,...,MC43], [MC44,..., MC69].
T2	P2,P6,P11,P12	MC2, MC5, MC6, MC11, MC16, [MC22,..., MC43], [MC44,..., MC69].
T3	P7	MC4
T4	P2,P6,P11	MC2,MC5, MC6, MC11, MC16, [MC22,...,MC43], [MC44,...,MC69],
T5	P3,P4,P5,P6,P8,P9,P10	MC3, MC6, MC15, MC16, [MC22,...,MC43], [MC44,...,MC69]

**Tabla 23. Otra perspectiva de la relación práctica, proceso de construcción y métricas.**

### **3.3.4 Recolección y clasificación de las métricas que puedan ser aplicadas a la técnica SPC en VSEs utilizando los criterios establecidos por IESRM.**

En literatura se encuentran estudios importantes acerca de la aplicación de SPC al contexto del software, de todos ellos sobresale lo propuesto por [11] en donde se presenta un instrumento para evaluar la idoneidad de un repositorio de medición para el control estadístico de procesos (IESRM).

El presente trabajo de investigación se centra en la identificación de un conjunto de métricas apropiadas para SPC, por lo que el IESRM se convierte en un insumo importante para la validación del conjunto de métricas expuesto en la Tabla 23. Sin embargo, este instrumento no contempla criterios relacionados con las VSEs, lo que obliga a estructurar un nuevo instrumento teniendo en cuenta las necesidades del presente trabajo investigación.

#### **3.3.4.1 IESMR (Instrument for Evaluating the Suitability of Measurement Repositories).**

De [11], se cuenta con un instrumento de validación (IESMR), el cual establece una serie de criterios relacionados con la definición de la métrica, su completitud y la medida en la que dicha métrica se ajusta a los objetivos de los proyectos y/o los objetivos de una organización software. Este instrumento se estructura con listas de chequeo y una guía de aplicación que permiten evaluar la idoneidad de las métricas, el plan de medición y el repositorio de medidas, para ser aplicadas a la técnica SPC de gráficos de control. El proceso para evaluar las métricas a través del IESMR se describe a continuación: i) Se elige la métrica a evaluar, ii) Con la lista de chequeo se asigna un valor a cada atributo (Satisfied, Largely Satisfied, Reasonably Satisfied, Precariously Satisfied, Dissatisfied) y finalmente iii) se obtiene un valor único que representa el grado de idoneidad de la métrica, plan de medición y el repositorio de medidas (a través de lógica difusa).

Después de un análisis riguroso del IESRM, se identifica la necesidad de adecuar dicho instrumento debido a que:

- i) No está enfocado a las características de las VSEs (ver sección 3.2.3.1).
- ii) El grado de idoneidad para el presente trabajo de investigación depende de las características que debe cumplir una métrica para su aplicación en la técnica SPC de gráficos de control.
- iii) El proceso para calcular el grado de idoneidad de una métrica debe estar fundamentado en las características de la métrica más que en el plan de medición y el repositorio de los datos.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

A continuación en la Tabla 24, se listan las características enfocadas a evaluar la idoneidad de una métrica para su aplicación a la técnica SPC, tomadas del IESRM.

ID	Característica
C1	Definición de la Métrica
C2	Entidad de la Métrica
C3	Atributo de la Métrica
C4	Unidad de Métrica
C5	Tipo de Escala
C6	Valores de Escala
C7	Intervalo esperado para el dato
C8	Formula(s) (Si Aplica)
C9	Descripción precisa del proceso de medición
C10	Responsable de la Medición
C11	Momento de Medición
C12	Periodicidad de la Medición
C13	Descripción precisa del análisis proceso de medición (Si Aplica)
C14	Responsable del análisis (Si Aplica)
C15	Momento de análisis (Si Aplica)
C16	Periodicidad de análisis (Si Aplica)
C17	Objetivos de la Organización
C18	Objetivos de proyecto
C19	Los resultados del análisis de esta medida son relevantes para la toma de decisiones.
C20	Los resultados del análisis de medición son útiles para la mejora de procesos.
C21	La medida está relacionada con el rendimiento del proceso.
C22	La medida está relacionada con un proceso crítico
C23	La medida está asociada a una actividad o proceso que produce productos/items medibles.
C24	Las medidas relacionadas se definen
C25	Las medidas relacionadas son válidas
C26	La métrica tiene un nivel de granularidad adecuada.
C27	Es posible normalizar la medida (Si aplica)
C28	La medida se normaliza correctamente (Si aplica)
C29	Los criterios de agrupación de datos para el análisis de la medida están definidas
C30	La medida no tiene en cuenta los datos agregados

**Tabla 24. Características IESRM para evaluar la idoneidad SPC de una métrica.**

Además de las características IESRM, también se realiza un análisis de la literatura acerca de las particularidades de la aplicación de las técnicas SPC en el

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

contexto de la industria del software (ver Tabla 25), con el fin de estructurar un nuevo instrumento que permita hacer la evaluación de la idoneidad de una métrica para su uso en SPC dentro del contexto de las pequeñas organizaciones.

ID	Nombre estudio	Autor
EID1	Investigating suitability of software process and metrics for statistical process control	Ayça Tarhan, Onur Demirörs.
EID2	Medidas de calidad en proceso, producto y mantenimiento, aplicadas al control estadístico de procesos	Gema de la Incera Torres.
EID3	12 steps to useful software metrics	Linda Westfall.
EID4	Evaluating the suitability of a measurement repository For statistical process control	Monalessa Perini Barcellos, Ana Regina Rocha, Ricardo de Almeida Falbo.
EID5	Incorporación de medidas en el modelo de procesos para la industria de software MOPROSOFT	Oswaldo Gómez, Hanna Oktaba, Mario Piattini, Félix García.
EID6	Calidad de productos de software: un estado del arte de la medición	Oswaldo Gómez, Hanna Oktaba, Mario Piattini, Félix García.
EID7	Experiences of applying SPC techniques to software development processes	Mutsumi Komuro.
EID8	Practical statistical process control for software metrics	Diane Manlove and Stephen H. Kan.
EID9	Statistical analysis on software metrics Affecting modularity in open source Software	Andi Wahju Rahardjo Emanuel, Retantyo Wardoyo, Jazi Eko Istiyanto, Khabib Mustofa.
EID10	Statistical process control: Analyzing a space shuttle onboard Software process	William A. Florac, Anita D. Carleton, Julie R. Barnard.
EID11	Application of statistical process control to software Development processes via control charts	Sargut, Kamil Umut.

**Tabla 25. Estudios SPC analizados.**

Con base en los estudios de la Tabla 25, se ponderan las características de la Tabla 24 a partir del grado de incidencia que tienen en la práctica del SPC. Además se abstraen otras características importantes, que no eran contempladas por el IESRM. De esta manera, se organiza y estructura un Instrumento para la

Validación de la Idoneidad de Métricas para SPC (IVIM), propuesto en el presente trabajo de investigación.

### 3.3.4.2 Instrumento para la Validación de la Idoneidad de Métricas para SPC (IVIM).

La adecuación del nuevo Instrumento IVIM, se rige por el grado de relevancia de las características propuestas por los estudios realizados en [11, 12, 30], con el fin de contar con un instrumentó que incluya las características de las VSEs. El proceso para calcular el grado de idoneidad de cada una de las características del IVIM, se fundamenta en lo propuesto por [12], este trabajo describe una serie de características que indican la idoneidad de una métrica para SPC (el cual complementa el estudio IESRM) y presenta un procedimiento para calcular el valor de idoneidad a través de pesos porcentuales, el cual es flexible y adecuado para su aplicación en el contexto de las VSEs. Finalmente para comprobar la utilidad del instrumento IVIM, se evalúan las métricas de la Tabla 16, que han sido aplicadas a casos de estudio SPC en la industria del software. El proceso de creación del IVIM se encuentra disponible en el CD anexo, carpeta IVIM. El instrumento IVIM se detalla en la Tabla 26.

Para conocer si una métrica es Idónea se debe calcular el **IIM** (Índice de Idoneidad de la Métrica) que se obtiene de multiplicar el valor de cada característica como se hace en [12].

$$\mathbf{IIM} = C1 * C2 * C3 * C4 * C5 * C6$$

EL valor de cada característica se obtiene de la siguiente formula:

$$\mathbf{C(n)} = C(n).1 * \text{Peso}(C(n).1) + C(n).2 * \text{Peso}(C(n).2) + \dots + C(n).i-1 * \text{Peso}(C(n).i-1) + C(n).i * \text{Peso}(C(n).i)$$

Donde  $n$  = número de la característica,  $i$  = número de sub características.

De [12] también se extraen los siguientes intervalos para interpretar el valor obtenido por **IIM**:

**IF IIM BETWEEN [0.00-0.25] THEN Métrica no Idónea.**

**IF IIM BETWEEN [0.26-0.50] THEN Utilizable.**

**IF IIM BETWEEN [0.51-0.75] THEN Utilizable en Gran parte.**

**IF IIM BETWEEN [0.76-1.00] THEN Métrica Idónea.**

MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

ID	Característica	Regla	Peso
<b>1</b>	<b>Escala Métrica</b>		
<b>C1.1</b>	¿Cuál es la Escala de la Métrica? (nominal, ordinal, intervalo, ratio, absoluto)	<b>IF Ratio OR Absolute THEN 1 ELSE 0</b>	1
<b>2</b>	<b>Cantidad Datos</b>		
<b>C2.1</b>	¿La cantidad de datos son adecuados?	<b>IF numero de datos &gt; 20 THEN 1 ELSE 0</b>	1
<b>3</b>	<b>Definición Métrica</b>		
<b>C3.1</b>	¿Definición de la Métrica?	<b>IF SI THEN 1 ELSE 0</b>	1/3
<b>C3.2</b>	¿Se Identifica la Entidad de la Métrica?	<b>IF SI THEN 1 ELSE 0</b>	1/3
<b>C3.3</b>	¿Responsable de la Medición?	<b>IF SI THEN 1 ELSE 0</b>	1/3
<b>4</b>	<b>Calculo Métrica</b>		
<b>C4.1</b>	¿Formula(s)?	<b>IF SI THEN 1 ELSE 0</b>	1/9
<b>C4.2</b>	¿Atributo de la Métrica?	<b>IF SI THEN 1 ELSE 0</b>	1/9
<b>C4.3</b>	¿Valores de Escala?	<b>IF SI THEN 1 ELSE 0</b>	1/9
<b>C4.4</b>	¿Unidad de Métrica?	<b>IF SI THEN 1 ELSE 0</b>	1/9
<b>C4.5</b>	¿Momento de Medición? (Datos pueden ser tomados en el proceso Construcción )	<b>IF SI THEN 1 ELSE 0</b>	1/9
<b>C4.7</b>	¿Descripción precisa del proceso de medición?	<b>IF SI THEN 1 ELSE 0</b>	1/9
<b>C4.8</b>	¿Es posible normalizar la medida?	<b>IF SI THEN 1 ELSE 0</b>	1/9
<b>C4.9</b>	¿Intervalo esperado para el dato?	<b>IF SI THEN 1 ELSE 0</b>	1/9
<b>5</b>	<b>Métrica Ajusta a los Objetivos de las Vsés</b>		
<b>C5.1</b>	¿Los resultados del análisis de esta medida son relevantes para la toma de decisiones?	<b>IF SI THEN 1 ELSE 0</b>	1/7
<b>C5.2</b>	¿La medida está relacionada con el rendimiento del proceso?	<b>IF SI THEN 1 ELSE 0</b>	1/7
<b>C5.3</b>	¿La métrica tiene un nivel de granularidad adecuada?	<b>IF SI THEN 1 ELSE 0</b>	1/7
<b>C5.4</b>	¿Se cuenta con un propósito para los datos de la métrica?	<b>IF SI THEN 1 ELSE 0</b>	1/7
<b>C5.5</b>	¿Los datos de la métrica son analizables?	<b>IF SI THEN 1 ELSE 0</b>	1/7
<b>C5.6</b>	¿El análisis de los datos de la métrica es utilizado por los actores del proceso?	<b>IF SI THEN 1 ELSE 0</b>	1/7
<b>C5.7</b>	¿El análisis de los datos de la métrica es utilizado para tomar decisiones?	<b>IF SI THEN 1 ELSE 0</b>	1/7
<b>6</b>	<b>Esfuerzo</b>		
<b>C6.1</b>	¿La captura de los datos de la métrica requiere personal especializado?	<b>IF SI THEN 0 ELSE 1</b>	1/3
<b>C6.2</b>	¿los Recursos necesarios para la toma de los datos de la métrica son altos? (Costo, Tiempo, Personal)	<b>IF SI THEN 0 ELSE 1</b>	1/3
<b>C6.3</b>	¿La métrica es Simple y fácil de calcular?	<b>IF SI THEN 1 ELSE 0</b>	1/3

Tabla 26. Instrumento para la Validación de la Idoneidad de Métricas para SPC (IVIM).

### 3.3.4.3 Verificar la utilidad del IVIM.

Se emplean las métricas utilizadas en la aplicación de SPC en la industria del software que se describen en la sección 3.2.4 (Tabla 16) para verificar la utilidad de la propuesta IVIM, la cual está estructurada por criterios extraídos de propuestas publicadas y validadas por expertos, de manera que no es necesaria la evaluación de los criterios. Lo anterior simplifica el trabajo a verificar en conjunto la utilidad de la propuesta IVIM con las métricas SPC ya analizadas.

A continuación en la Tabla 27 se listan los resultados finales de la evaluación de las métricas SPC con el IVIM (el detalle de las evaluaciones se muestran en CD Anexo, IVIM/EVALUACIONES/EVAL\_IVIM-METRICAS\_SPC.xlsx).

Métrica	IIM(Índice de Idoneidad de la Métrica)	Interpretación de IIM	¿Aplica al proceso de construcción de software?
Densidad de defectos	0,875	<b>Métrica Idónea</b>	Si
Porcentaje de Retrabajo	0,875	<b>Métrica Idónea</b>	Si
Rendimiento de Inspección	0,5	<b>Utilizable</b>	No
Tasa de detección de errores	0,333	<b>Utilizable</b>	Si
Velocidad de Revisión	0,278	<b>Utilizable</b>	Si
Productividad	0,875	<b>Métrica Idónea</b>	Si
Integración de código	0,417	<b>Utilizable</b>	No

**Tabla 27. Resultados de la evaluación con el instrumento IVIM a las métricas ya aplicadas en SPC.**

Se observa que, como en la literatura, las métricas son utilizables para su aplicación en la técnica de gráficos de control. Sin embargo, 4 de las 7 métricas son útiles pero no idóneas para el IVIM debido a que no cumplen con alguna de las características de “Esfuerzo” que es un criterio para ser utilizada por las VSEs. Además las métricas de integración de código y rendimiento de inspección no aplican al proceso de construcción, debido a que estas se validan por el criterio C4.5 que identifica en qué etapa del proceso se realiza la medición, criterio que permite individualizar aquellas métricas específicas del proceso de construcción de software, requeridas para el presente trabajo de investigación.

Con el trabajo anterior se verifica la utilidad del IVIM para la evaluación de métricas del proceso de construcción apropiadas para la aplicación de SPC en

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

pequeñas organizaciones, de manera que es posible evaluar las métricas software de la sección 3.3.3.1 (ver Tabla 21) para obtener el conjunto de métricas final.

### 3.4 Recopilación de datos.

#### 3.4.1 Listar el conjunto de métricas adecuadas para la aplicación de la técnica SPC en VSEs, basados en las prácticas de las VSEs ejecutadas en las tareas del proceso de construcción.

##### 3.4.1.1 Evaluación de las métricas con el IVIM.

A continuación se listan los resultados finales de las evaluaciones con el IVIM para el grupo de métricas filtradas en la sección 3.3.3.1. La Tabla 28 muestra el ID de la métrica, su representación, el IIM (Índice de Idoneidad de la Métrica) y su interpretación basada en los criterios expuestos por el IVIM. Esto con el fin de obtener el listado final del conjunto de métricas del proceso de construcción de software que son adecuadas para la aplicación de SPC en VSEs. El conjunto de métricas es conformado por aquellas que tienen la etiqueta de “Métrica Idónea” en su interpretación IIM. El CD anexo (carpeta IVIM/EVALUACIONES) contiene los detalles de cada una de las evaluaciones.

ID	Medida	IMM	Interpretación IMM
MC1	FP	0,875	Utilizable en Gran parte
MC2	LOC	0,5	Utilizable
MC3	LOC+NLOC	0,5	Utilizable
MC4	MHK	0,27777778	Utilizable
MC5	Tiempo en Corregir	0,41666667	Utilizable
MC6	EV	0,875	Métrica Idónea
MC11	Densidad de Defectos	0,875	Métrica Idónea
MC15	Productividad	0,875	Métrica Idónea
MC16	Esfuerzo	0,5	Utilizable
Medidas Enfocadas en el paradigma Estructurado			
ID	Medida	IIM	Interpretación
MC22	u1	0,07936508	Métrica no Idónea
MC23	u2	0,07936508	Métrica no Idónea
MC24	u	0,11111111	Métrica no Idónea
MC25	N1	0,07936508	Métrica no Idónea
MC26	N2	0,07936508	Métrica no Idónea
MC27	N	0,11111111	Métrica no Idónea
MC28	$V=Nx\log_2u$	0,19047619	Métrica no Idónea
MC29	N COPY	0,06349206	Métrica no Idónea
MC30	N PERF	0,06349206	Métrica no Idónea

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

MC31	N SEL	0,07936508	Métrica no Idónea
MC32	N CYCLE	0,07936508	Métrica no Idónea
MC33	N EXEC	0,07936508	Métrica no Idónea
MC34	Block COV	0	Métrica no Idónea
MC35	ST COV	0,07936508	Métrica no Idónea
MC36	ILOC	0,35714286	Utilizable
MC37	Funciones	0,15873016	Métrica no Idónea
MC38	FP	0,11904762	Métrica no Idónea
MC39	FR	0,11904762	Métrica no Idónea
MC40	IC	0,16666667	Métrica no Idónea
MC41	CO	0	Métrica no Idónea
MC42	AweBlockL	0	Métrica no Idónea
MC43	Y(G)	0	Métrica no Idónea
<b>Medidas Enfocadas en el paradigma Orientado a Objetos</b>			
<b>ID</b>	<b>Medida</b>	<b>IMM</b>	<b>Interpretación IMM</b>
MC44	CBO	0,09920635	Métrica no Idónea
MC45	RFC	0,07936508	Métrica no Idónea
MC46	MPC	0,09920635	Métrica no Idónea
MC47	DAC	0,09920635	Métrica no Idónea
MC48	ICP	0	Métrica no Idónea
MC49	IH-ICP	0	Métrica no Idónea
MC50	LCOM	0,09920635	Métrica no Idónea
MC51	TCC	0,11904762	Métrica no Idónea
MC52	LCC	0,07936508	Métrica no Idónea
MC53	ICH	0,05952381	Métrica no Idónea
MC54	DIT	0	Métrica no Idónea
MC55	AID	0	Métrica no Idónea
MC56	CLD	0	Métrica no Idónea
MC57	NOC	0	Métrica no Idónea
MC58	NOP	0	Métrica no Idónea
MC59	NOD o NOAC	0	Métrica no Idónea
MC60	NOA	0	Métrica no Idónea
MC61	NMO	0,07936508	Métrica no Idónea
MC62	NMI	0	Métrica no Idónea
MC63	NMA (number of methods added)	0,07936508	Métrica no Idónea
MC64	SIX	0	Métrica no Idónea
MC65	NM	0,09920635	Métrica no Idónea
MC66	NAI	0,09920635	Métrica no Idónea
MC67	Nmpub	0,09920635	Métrica no Idónea

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

MC68	NumPara	0,09920635	Métrica no Idónea
MC69	APPM	0,13888889	Métrica no Idónea

**Tabla 28. Resultado final de la evaluación con el IVIM al conjunto de métricas software.**

### 3.4.1.2 Conjunto de métricas del proceso de construcción apropiadas para la aplicación de SPC en pequeñas organizaciones.

#### 3.4.1.2.1 Conjunto preliminar de métricas.

A continuación en la Tabla 29 se listan las “métricas idóneas” para su aplicación al SPC según el resultado de su evaluación con el IVIM mostrado en la Tabla 28:

ID	Medida	IMM	Interpretación IMM
MC6	EV	0,875	Métrica Idónea
MC11	Densidad de Defectos	0,875	Métrica Idónea
MC15	Productividad	0,875	Métrica Idónea

**Tabla 29. Conjunto preliminar de Métricas apropiadas para SPC.**

Cabe destacar que las métricas Densidad de Defectos y Productividad ya se habían analizado en la sección 3.2.4 (Tabla 16. Métricas software utilizadas en la aplicación de SPC con gráficos de control.), lo que confirma la fiabilidad del proceso llevado a cabo en el presente trabajo de investigación.

Por otra parte, de la sección 3.3.4.3 (Tabla 27) se extrae la métrica de Porcentaje de Retrabajo debido a que el resultado de su evaluación con el IVIM fue satisfactorio, logrando la etiqueta de “Métrica Idónea”, esta métrica se obtuvo de los estudios analizados en la sección 3.2.4.

Finalmente en la Tabla 30 se propone un conjunto de métricas relacionadas con el proceso de construcción de software que pueden ser utilizadas para el control estadístico de procesos en pequeñas organizaciones.

#### 3.4.1.2.2 Gráficos de control adecuados para el conjunto de métricas propuesto.

Como ya se analizó en la sección 2.1.2.2.1 (CAPITULO II: Marco Teórico y Estado del Arte), la elección de un gráfico de control, no solo depende del tipo de característica a graficar sino también de la toma de las muestras y el tamaño fijo o dinámico de los subgrupos racionales. Así, antes de la elección del tipo de gráfico de control a usar, se debe tener en cuenta la toma de los datos de la característica elegida, lo que está fuera del alcance del presente trabajo de investigación. Sin embargo, con base en el conjunto final de métricas propuesto en la tabla anterior

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

(Tabla 30) y en lo considerado en dicha sección, se propone el tipo de grafico de control (por variables o por atributos) para cada una de las métricas (ver Tabla 31).

ID	Nombre	Descripción
M-SPC1	Valor Ganado (EV)	Es la cantidad presupuestada para el trabajo realmente completado de la actividad del cronograma o el componente de la EDT durante un período de tiempo determinado[52]
M-SPC2	Densidad de Defectos	Captura el número de defectos en un producto de trabajo, este conteo de defectos brinda evidencia no sólo de la calidad del producto, sino también de la calidad del proceso relacionado. Para obtener la densidad de defectos se normaliza la información de los defectos con la dimensión tamaño. La acumulación y descubrimiento de defectos son medidas que se pueden derivar del trabajo de capturar la densidad de defectos.
M-SPC3	Productividad	En términos generales, la productividad es la cantidad de producto generado por insumos gastados. Como lo hacen en [12], para el contexto del software podría calcularse el número de líneas de código por programa, junto el esfuerzo (horas de trabajo) de cada programa para representar el rendimiento del desarrollo en un proyecto o la productividad del desarrollador.
M-SPC4	Porcentaje de Retrabajo	La repetición de trabajo es un buen indicador de la calidad del proceso, debido a que refleja la magnitud del esfuerzo dedicado a errores previos. De la misma forma, incrementa los costos del proyecto sin agregarle valor al producto. En realidad, es posible obtener un producto sin necesidad de repetición, si el proceso se ejecuta correctamente desde la primera vez. Esto se describe como "Costo de la Calidad".

**Tabla 30. Métricas relacionadas con el proceso de construcción de software que pueden ser utilizadas en SPC para las VSEs.**

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

ID	Nombre	Según el Tipo de Característica	Gráfico de Control Recomendado
M-SPC1	Valor Ganado (EV)	Gráficos de control por variables	<ul style="list-style-type: none"> <li>✓ <b>Gráficos Valores individuales</b> Si la toma de los datos son valores individuales.</li> <li>✓ <b>Gráficos X-barra y R</b> Si para toma de los datos se hacen subgrupos racionales de menos de 8 muestras.</li> <li>✓ <b>Gráficos X-barra y S</b> Si para toma de los datos se hacen subgrupos racionales con tamaño variable.</li> </ul>
M-SPC2	Densidad de Defectos	Gráficos de control por atributos	<b>Gráfico U</b>
M-SPC3	Productividad	Gráficos de control por variables	<ul style="list-style-type: none"> <li>✓ <b>Gráficos Valores individuales</b> Si la toma de los datos son valores individuales.</li> <li>✓ <b>Gráficos X-barra y R</b> Si para toma de los datos se hacen subgrupos racionales de menos de 8 muestras.</li> <li>✓ <b>Gráficos X-barra y S</b> Si para toma de los datos se hacen subgrupos racionales con tamaño variable.</li> </ul>
M-SPC4	Porcentaje de Retrabajo	Gráficos de control por atributos	<ul style="list-style-type: none"> <li>✓ <b>Gráficos Valores individuales</b> Si la toma de los datos son valores individuales.</li> <li>✓ <b>Gráficos X-barra y R</b> Si para toma de los datos se hacen subgrupos racionales de menos de 8 muestras.</li> <li>✓ <b>Gráficos X-barra y S</b> Si para toma de los datos se hacen subgrupos racionales con tamaño variable.</li> </ul>

**Tabla 31. Gráficos de control apropiados al conjunto de métricas propuesto.**

### 3.4.2 Identificar las necesidades de información que satisfacen las métricas.

De acuerdo al atributo de medición del proceso (relacionado con la capacidad de procesos) del estándar internacional ISO/IEC 15504, la completitud de las métricas está dada en términos de: (i) la definición de las métricas, (ii) la captura de sus datos (medidas) y (iii) la satisfacción de una necesidad de información de la

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

organización. A continuación se describen estos datos relacionados con la completitud de cada una de las métricas listadas en la Tabla 30, estos datos están relacionados con: la métrica, con qué objetivo de la organización está relacionada, a que pregunta responde, y que dato proporciona.

<b>ID</b>	M-SPC1
<b>Nombre</b>	Valor Ganado (EV)
<b>Formula</b>	$EV = \text{Porcentaje Ejecutado (\%)} * \text{El Presupuesto del Proyecto (BAC)}$ .
<b>Descripción</b>	Es la cantidad presupuestada para el trabajo realmente completado de la actividad del cronograma o el componente de la EDT durante un período de tiempo determinado.
<b>Entidad</b>	Programa
<b>Tipo</b>	Indicador
<b>Escala</b>	Ratio
<b>Unidad</b>	Costo
<b>Atributo</b>	Avance
<b>Método Aplicación</b>	El parámetro EV, es la expresión monetaria del avance real del proyecto (El porcentaje del Trabajo realizado) respecto al total del presupuesto planificado y aprobado del proyecto (BAC).
<b>Interpretación</b>	El valor ganado (EV) compara la cantidad de trabajo planeado contra lo que realmente se ha terminado para determinar si el costo, el cronograma y el trabajo realizado están llevándose a cabo de acuerdo con lo planeado.
<b>Preguntas</b>	✓ ¿Cuánto se ha desviado el desarrollo respecto del plan actual?
<b>Objetivos</b>	✓ Llevar a cabo las actividades de las fases de un ciclo mediante el cumplimiento del Plan de Desarrollo actual.
<b>Según el Tipo de Característica</b>	Gráficos de control por variables
<b>Gráfico de Control Recomendado</b>	<ul style="list-style-type: none"> <li>✓ <b>Gráficos Valores individuales:</b> Si la toma de los datos son valores individuales.</li> <li>✓ <b>Gráficos X-barra y R:</b> Si para toma de los datos se hacen subgrupos racionales de menos de 8 muestras.</li> <li>✓ <b>Gráficos X-barra y S:</b> Si para toma de los datos se hacen subgrupos racionales con tamaño variable.</li> </ul>

**Tabla 32. Datos relacionados con la completitud de la métrica Valor Ganado.**

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>ID</b>	M-SPC2
<b>Nombre</b>	Densidad de Defectos
<b>Formula</b>	DEN FAULT=Defectos/PF o DEN FAULT=Defectos/LOC
<b>Descripción</b>	Captura el número de defectos en un producto de trabajo, este conteo de defectos brinda evidencia no sólo de la calidad del producto, sino también de la calidad del proceso relacionado [23]. Para obtener la densidad de defectos se normaliza la información de los defectos con la dimensión tamaño.
<b>Entidad</b>	Programa
<b>Tipo</b>	Derivada
<b>Escala</b>	Ratio
<b>Unidad</b>	Defectos/tamaño
<b>Atributo</b>	Calidad
<b>Método Aplicación</b>	Tomar el número de fallos detectados en la revisión de las pruebas unitarias y compararlo con las líneas de código o los Puntos de Función.
<b>Interpretación</b>	Un valor bajo (o nulo) de DEN FAULT implica una buena calidad del producto, sin embargo DEN FAULT= 0 no necesariamente implica libre de errores.
<b>Preguntas</b>	✓ ¿Qué tan correcto es el trabajo realizado?
<b>Objetivos</b>	✓ Lograr que los productos de salida sean consistentes con los productos de entrada en cada fase de un ciclo de desarrollo mediante las actividades de verificación, validación o prueba.
<b>Según el Tipo de Característica</b>	Gráficos de control por atributos
<b>Gráfico de Control Recomendado</b>	Gráfico U

**Tabla 33. Datos relacionados con la completitud de la métrica Densidad de Defectos.**

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>ID</b>	M-SPC3
<b>Nombre</b>	Productividad
<b>Formula</b>	$P=LOC/unidad\ de\ tiempo$ o $P=PF/Unidad\ de\ tiempo$
<b>Descripción</b>	La productividad es la cantidad de producto generado por insumos gastados. Como lo hacen en [12], para el contexto del software podría calcularse el número de líneas de código por programa, junto el esfuerzo (horas de trabajo) de cada programa para representar el rendimiento del desarrollo en un proyecto o la productividad del desarrollador.
<b>Entidad</b>	Programa
<b>Tipo</b>	Derivada
<b>Escala</b>	Ratio
<b>Unidad</b>	Tamaño/Unidad de tiempo
<b>Atributo</b>	Calidad
<b>Método Aplicación</b>	Tomar el tamaño del producto en la revisión y compararlo con el tiempo que tomo desarrollarlo.
<b>Interpretación</b>	Un valor alto de P implica un buen rendimiento de desarrollo. Si $P=0$ , indica que no se generó nada nuevo.
<b>Preguntas</b>	<ul style="list-style-type: none"> <li>✓ ¿Cuánto tiempo lleva realizar el producto de software?</li> <li>✓ ¿Qué tan grande es el producto de software?</li> </ul>
<b>Objetivos</b>	<ul style="list-style-type: none"> <li>✓ Lograr que los productos de salida sean consistentes con los productos de entrada en cada fase de un ciclo de desarrollo mediante las actividades de verificación, validación o prueba.</li> <li>✓ Llevar a cabo las actividades de las fases de un ciclo mediante el cumplimiento del <i>Plan de Desarrollo</i> actual.</li> </ul>
<b>Según el Tipo de Característica</b>	Gráficos de control por variables
<b>Gráfico de Control Recomendado</b>	<ul style="list-style-type: none"> <li>✓ <b>Gráficos Valores individuales:</b> Si la toma de los datos son valores individuales.</li> <li>✓ <b>Gráficos X-barra y R:</b> Si para toma de los datos se hacen subgrupos racionales de menos de 8 muestras.</li> <li>✓ <b>Gráficos X-barra y S:</b> Si para toma de los datos se hacen subgrupos racionales con tamaño variable.</li> </ul>

**Tabla 34. Datos relacionados con la completitud de la métrica productividad.**

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>ID</b>	M-SPC4
<b>Nombre</b>	Porcentaje de Retrabajo
<b>Formula</b>	$PR = \frac{\text{Esfuerzo de Retrabajo}}{\text{Esfuerzo total}}$
<b>Descripción</b>	La repetición de trabajo es un buen indicador de la calidad del proceso, debido a que refleja la magnitud del esfuerzo dedicado a errores previos. De la misma forma, incrementa los costos del proyecto sin agregarle valor al producto. En realidad, es posible obtener un producto sin necesidad de repetición, si el proceso se ejecuta correctamente desde la primera vez. Esto se describe como “Costo de la Calidad”.
<b>Entidad</b>	Programa
<b>Tipo</b>	Derivada
<b>Escala</b>	Ratio
<b>Unidad</b>	Esfuerzo
<b>Atributo</b>	Calidad
<b>Método Aplicación</b>	El retrabajo muestra la magnitud de esfuerzo gastado (inversión total de tiempo) en cambios imprevistos o errores del código software. Por esto, se debe contar el tiempo total dedicado a cambios no planificados o a errores. El esfuerzo de repetición de trabajo se puede aplicar a las diferentes instancias de procesos, por ejemplo en [23] el porcentaje de retrabajo es igual al esfuerzo de revisión sobre esfuerzo total para las actividades de Testing. En otras palabras, esta métrica permite llevar a cabo un análisis para cualquier grupo de tareas que representen un dominio específico dentro del desarrollo del software.
<b>Interpretación</b>	Un valor bajo de PR implica un bajo costo de inversión para calidad, sin embargo $PR = 0$ no necesariamente implica libre de errores.
<b>Preguntas</b>	✓ ¿Cuánto tiempo es gastado en retrabajo?
<b>Objetivos</b>	✓ Sustentar la realización de ciclos posteriores o proyectos de mantenimiento futuros mediante la integración de la configuración de software del ciclo actual.
<b>Según el Tipo de Característica</b>	Gráficos de control por variables
<b>Gráfico de Control Recomendado</b>	<ul style="list-style-type: none"> <li>✓ <b>Gráficos Valores individuales:</b> Si la toma de los datos son valores individuales.</li> <li>✓ <b>Gráficos X-barra y R:</b> Si para toma de los datos se hacen subgrupos racionales de menos de 8 muestras.</li> <li>✓ <b>Gráficos X-barra y S:</b> Subgrupos racionales con tamaño variable.</li> </ul>

**Tabla 35. Datos relacionados con la completitud de la métrica Porcentaje de Retrabajo.**

# CAPÍTULO IV

## PROTOTIPO SOFTWARE

---

El prototipo surge del ciclo de investigación técnico para automatizar la captura de los datos y facilitar la actividad de medición en el proceso de construcción de las VSEs. Este prototipo es una herramienta software creada para apoyar la captura de las métricas propuestas. También disminuye el costo de recopilar y analizar los datos, costo que no puede ser asumido por las VSEs debido a que tienen recursos limitados, manejan pocos proyectos y el número de empleados es pequeño [7].

El ciclo de investigación técnico está estructurado a partir de la metodología de desarrollo Ágil XP. En este capítulo se muestran algunos de los elementos que surgen de la aplicación de la metodología XP empleada para la construcción del prototipo software. La gestión de los datos y la configuración se hizo a través de una aplicación web. Para la obtención de los datos, se construyó una terminal diseñada para capturarlos y almacenarlos en una base de datos. En CD Anexo (carpeta PRODUCTO SOFTWARE), se encuentran el prototipo software denominado CMSPC, su manual de usuario y un video introductorio al uso del mismo.

### 4.1 Perspectiva del producto.

El sistema CMSPC es una herramienta cuyo objetivo es “apoyar la aplicación de las técnicas de levantamiento de las métricas idóneas para SPC del proceso de construcción de software”. Y surge del problema técnico de la estrategia de investigación para automatizar la captura de los datos y facilitar la actividad de medición en el proceso de construcción de las VSEs con el conjunto de métricas encontradas en el Capítulo III del presente trabajo de investigación. CMSPC pretende disminuir el costo de recopilar y analizar los datos, los cuales no pueden ser asumidos por las VSEs debido a que tienen recursos económicos limitados, manejan pocos proyectos y el número de empleados es pequeño [4]. De esta manera el CMSPC permite que la técnica de captura asociada a cada métrica se realice de manera correcta, eliminando los malentendidos que surgen al emplear las métricas tal como lo sugiere [11], y poner más énfasis en la estabilidad de los datos en lugar de la cantidad de datos [19] para permitir la aplicación de SPC en las VSEs. A partir de los beneficios mencionados anteriormente, se identifica la importancia de aportar nuevas herramientas software en el área de captura de métricas para SPC.

#### **4.1.1 Metodología del producto.**

El desarrollo del prototipo esta guiado por la metodología ágil XP (Extreme programming) [16], donde la construcción del prototipo software se realiza de forma iterativa e incremental. En cada iteración se desarrolla y se verifica las funcionalidades necesarias (historias de usuario) para la captura de las métricas propuestas en el presente trabajo de investigación. La adaptación de la metodología solo incluyó las fases de: Exploración, Planeación, Diseño, Codificación y Pruebas, ya que no se hace mantenimiento del prototipo software, además se utilizó los valores y principios de XP. Las fases realizadas se describen a continuación.

##### **4.1.1.1 Fases de la Metodología.**

###### **4.1.1.1.1 Fase de Exploración.**

Es la fase en la que se define el alcance general del proyecto. En esta fase, el cliente define lo que necesita mediante la redacción en lenguaje natural de las “historias de usuario”. Los programadores estiman el tiempo para desarrollar el sistema en base a esta información. Las estimación en esta fase están basadas en datos de muy alto nivel y estas varían cuando se detallan en las reuniones diarias en cada iteración. Para este trabajo la Fase de Exploración se ejecutó desde el: 29 julio 2013 hasta el 2 agosto de 2013.

###### **4.1.1.1.2 Fase de Planeación.**

La planificación es una fase corta, en la que el cliente, y el grupo de desarrolladores determinan en qué orden se deben implementar las historias de usuario, y, asociadas a éstas, las entregas. Típicamente esta fase consiste en una o varias reuniones grupales de planificación. El resultado de esta fase es el Plan de Iteración. Para este trabajo la Fase de Planeación se ejecutó desde el: 5 agosto de 2013 hasta el 9 agosto de 2013.

###### **4.1.1.1.3 Fase de Iteraciones.**

Esta es la fase principal en el ciclo de desarrollo de XP. Las funcionalidades son desarrolladas en esta fase, generando al final de cada una un entregable funcional que implementa las historias de usuario asignadas a la iteración. Como las historias de usuario no tienen suficiente detalle como para permitir su análisis y desarrollo, al principio de cada iteración se realizan las tareas necesarias de análisis, recabando con el cliente todos los datos que sean necesarios. El cliente, por lo tanto, también debe participar activamente durante esta fase del ciclo.

#### 4.1.1.1.4 Fase de Pruebas.

Se diseñaron pruebas unitarias para el prototipo (ver CD Anexo/METODOLOGIA) debido a su importancia y necesidad para la aplicación de la metodología XP. En esta metodología todos los módulos deben de pasar las pruebas unitarias antes de ser liberados o publicados. Por otra parte, como se mencionó anteriormente, las pruebas deben ser definidas antes de realizar el código. En este sentido, el conjunto de pruebas debe ser guardado junto con el código, para que pueda ser utilizado por otros desarrolladores.

#### 4.1.2 Actividades de Trabajo.

Se relacionan las actividades de trabajo a cada uno de las fases de XP esta relación se describe a continuación en la Tabla 36:

FASE	ACTIVIDADES
<b>Exploración</b>	<ol style="list-style-type: none"> <li>1. Reuniones con el cliente para plantear sus necesidades.</li> <li>2. El equipo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto.</li> <li>3. Definición de las plantillas de los artefactos.</li> </ol>
<b>Planeación</b>	<ol style="list-style-type: none"> <li>1. Escribir Historias de Usuario</li> <li>2. Criterios de aceptación del producto</li> <li>3. WBS</li> <li>4. Plan de Iteración</li> </ol>
<b>Iteración I(arquitectura)</b>	<ol style="list-style-type: none"> <li>1. Documento del Modelo Global</li> <li>2. Lista Priorizada Riesgos</li> </ol>
<b>Iteración II(modulo 1)</b>	<ol style="list-style-type: none"> <li>1. Diseño Pruebas de unidad</li> <li>2. Programación en Parejas de las Historias de Usuario Modulo 1</li> <li>3. Ejecución Pruebas de unidad módulo 1</li> <li>4. Integración Continua</li> </ol>
<b>Iteración III(modulo 2)</b>	<ol style="list-style-type: none"> <li>1. Diseño Pruebas de unida</li> <li>2. Programación en Parejas de las Historias de Usuario Modulo 2</li> <li>3. Ejecución Pruebas de unidad módulo 2</li> <li>4. Integración Continua</li> </ol>
<b>Pruebas</b>	<ol style="list-style-type: none"> <li>1. Ejecución Pruebas de Aceptación</li> </ol>
<b>Iteración IV(Refinar Prototipo)</b>	<ol style="list-style-type: none"> <li>1. Hacer ajustes al sistema</li> </ol>
<b>Cierre Proyecto</b>	<ol style="list-style-type: none"> <li>1. Producto Software</li> <li>2. Manual de Aplicación (Instalación/Usuario)</li> </ol>

**Tabla 36. Actividades del desarrollo del prototipo.**

#### 4.1.3 Asignación de Roles.

A continuación en la Tabla 37 se presenta la descripción de cada uno de los roles asociados a la metodología XP:

ROL	DESCRIPCION
Programador	El programador escribe las pruebas unitarias y produce el código del sistema
Cliente	El cliente narra las historias de usuario y narra las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio.
Encargado de pruebas	Ejecuta las pruebas regularmente, y es responsable de las herramientas de soporte para pruebas.
Entrenador	Es responsable del proceso global. Es necesario que conozca a fondo el proceso XP para proveer guías a los miembros del equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente
Gestor	Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación.

**Tabla 37. Roles de la metodología XP.**

La asignación de los roles se especifica en la Tabla 38:

NOMBRE	ROLES
Pablo Felipe Miranda	Entrenador, Programador, Encargado Pruebas
German Geovanny Garzón	Programador, Encargado Pruebas
Francisco José Pino	Cliente, Gestor

**Tabla 38. Asignación de Roles para el equipo de desarrollo.**

#### 4.1.4 Cronograma.

Detalla la planificación requerida para el desarrollo del prototipo, haciendo las distribuciones de tiempo necesarias (ver Tabla 39). Tal distribución se realiza especificando las fases a realizar, indicando fechas de inicio y finalización de cada una. Ayuda a controlar las actividades y su duración, para no generar retrasos en el desarrollo de las funcionalidades.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

FASE	FECHA
Exploración	29 julio - 2 agosto de 2013
Planeación	5 agosto - 9 agosto de 2013
Iteración I(arquitectura)	12 agosto - 16 agosto de 2013
Iteración II(modulo 1)	19 agosto - 23 agosto de 2013
Iteración III(modulo 2)	26 agosto - 30 agosto de 2013
Pruebas	2 septiembre - 6 septiembre de 2013
Iteración IV(Refinar Prototipo)	9 septiembre - 13 septiembre de 2013
Cierre Proyecto	16 septiembre de 2013

**Tabla 39. Cronograma.**

Cabe aclarar que el día de trabajo se compone de 8 horas, y la semana de 5 días (no se contempla los sábados, ni domingos). Cumpliendo las 40 horas recomendadas por la metodología XP.

### **4.2 Desarrollo de las Fases de la Metodología.**

En este punto se describen los elementos más relevantes de las fases de la metodología utilizada para el desarrollo del prototipo software.

#### **4.2.1 Fase de Exploración.**

A continuación se describen algunos elementos derivados de acuerdos con el cliente.

##### **4.2.1.1 Selección de un Framework de Desarrollo.**

El desarrollo fue soportado por CodeIgniter Es un Framework MVC liviano y de código abierto, centrado desarrollar aplicaciones Web de una manera rápida y sencilla, está escrito en PHP5 y permite utilizar MySQL como herramienta para la persistencia.

##### **4.2.1.2 Características de los Usuarios.**

Los usuarios potenciales del sistema serán los Desarrolladores de las VSEs que quieran controlar su proceso de construcción utilizando la técnica SPC de gráficos de control y un encargado que puede ser un empleado asociado a la calidad de software que gestione el sistema. Para el acceso al sistema, se definen 2 perfiles de usuario: Desarrollador, Administrador. El desarrollador captura los datos de las

métricas a través de su terminal correspondiente, y el Administrador gestiona los datos por medio de un aplicativo web.

#### **4.2.1.3 Requisitos del Banco de Datos Lógicos.**

Se cuenta con una base de datos que almacenará los datos que utilizará la aplicación CMSPC. Solo el administrador, pueden accederán a ella cada vez que necesiten consultar información o modificarla, los Desarrolladores aportan con los datos de las métricas pero no se permite modificar ni eliminar dichos datos desde sus terminales.

#### **4.2.1.4 Restricciones de Integridad.**

- Dos terminales no pueden capturar datos al tiempo de un mismo módulo perteneciente a un proyecto.
- La terminal identificada con un nombre no puede tener dos instancias ejecutadas al tiempo.
- No se debe permitir borrar los registros de definición de las métricas.
- No debe permitir ingresar por parte de los usuarios datos alfanuméricos cuando se soliciten datos numéricos, ejemplo tamaño del módulo.
- No se debe permitir la edición de los datos capturados de las métricas.

#### **4.2.1.5 Restricciones Generales.**

- Las fechas de finalización de los hitos principales, ya que el atraso de una tarea crítica implica el retraso de sus sucesoras y por ende la entrega del proyecto.
- El tiempo que dispone cada integrante del equipo de proyecto, cada miembro del equipo cuenta con un total de 40 horas a la semana establecido por la metodología.
- El Desarrollo del trabajo investigativo no ha culminado lo que agota un poco al grupo de desarrollo, además que todo se debe documentar en la monografía.
- La fecha de finalización del proyecto debe ser igual o menor al 16 de septiembre del 2013.
- El cronograma definido en el Plan de Iteración. Es importante realizar una buena estimación tanto de los recursos de personal como los recursos materiales para suplir con las actividades llevadas a cabo en la realización del proyecto.

#### 4.2.1.6 Restricciones del Producto.

- Los navegadores soportados por el producto serán: Internet Explorer IE8 o superior, Google Chrome y Mozilla Firefox 11 o superior.
- Hardware: 512MB o más de RAM, resolución de pantalla 1280x800 y disco duro de 5400 RPM o superior.
- Interfaz: Debe ser una interfaz amigable, sencilla de usar, intuitiva y preferiblemente de colores que no produzcan agotamiento visual, dichos colores deben estar sujetos al tema principal que maneja la universidad del cauca.
- El producto solamente será utilizado por el jefe de desarrollo, líder de desarrollo o calidad para la gestión y por los desarrolladores para la captura de los datos.
- Se utilizará un servidor de prueba para el sistema en funcionamiento, el cual tiene las siguientes características:
  - Cuatro Procesadores Intel(R) Core 2 QUAD(R) CPU E5620 @2.4GHz
  - Memoria (Ram): 4GB
  - Sistema operativo Ubuntu Server 12.04.3 LTS
  - Disco duro de 1 TB
- La aplicación debe estar disponible tiempo completo, es decir 24 horas del día, 7 días a la semana.
- Las condiciones y formas de uso del producto deben encontrarse en los manuales respectivos.
- El producto estará sujeto a las licencias de propiedad intelectual de la Universidad del Cauca.
- Se reserva los derechos de realización del producto en cuestión a los estudiantes Pablo Miranda y German Garzón del Programa de Ingeniería de Sistemas de la Universidad del Cauca correspondiente al segundo periodo académico del 2013.
- El producto debe garantizar seguridad e integridad en toda su información, por lo tanto no debe ser vulnerable a los ataques más típicos como XSS o SQL Injection.
- El producto contará con métodos de autenticación DNIE para el acceso de usuarios con privilegios.
- El tiempo de respuesta para cualquier operación solicitada debe ser tolerable, es decir que el usuario mantenga su atención a la tarea que está realizando. Dicho tiempo no debe ser mayor a 5 segundos.

#### 4.2.2 Fase de Planeación.

Esta fase contempla la generación de las historias de usuario, criterios de aceptación del producto y el plan de iteración, que se encuentran detallados en anexos (ver CD Anexos, METODOLOGIA). A continuación, como ejemplo se muestra una historia de usuario (ver Figura 6) con su respectivo "MOCKUP" (ver Figura 5).

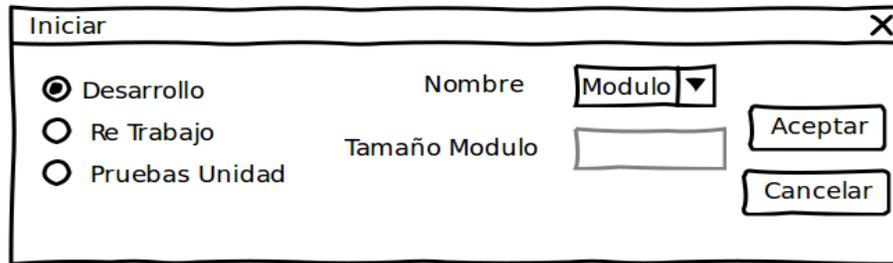
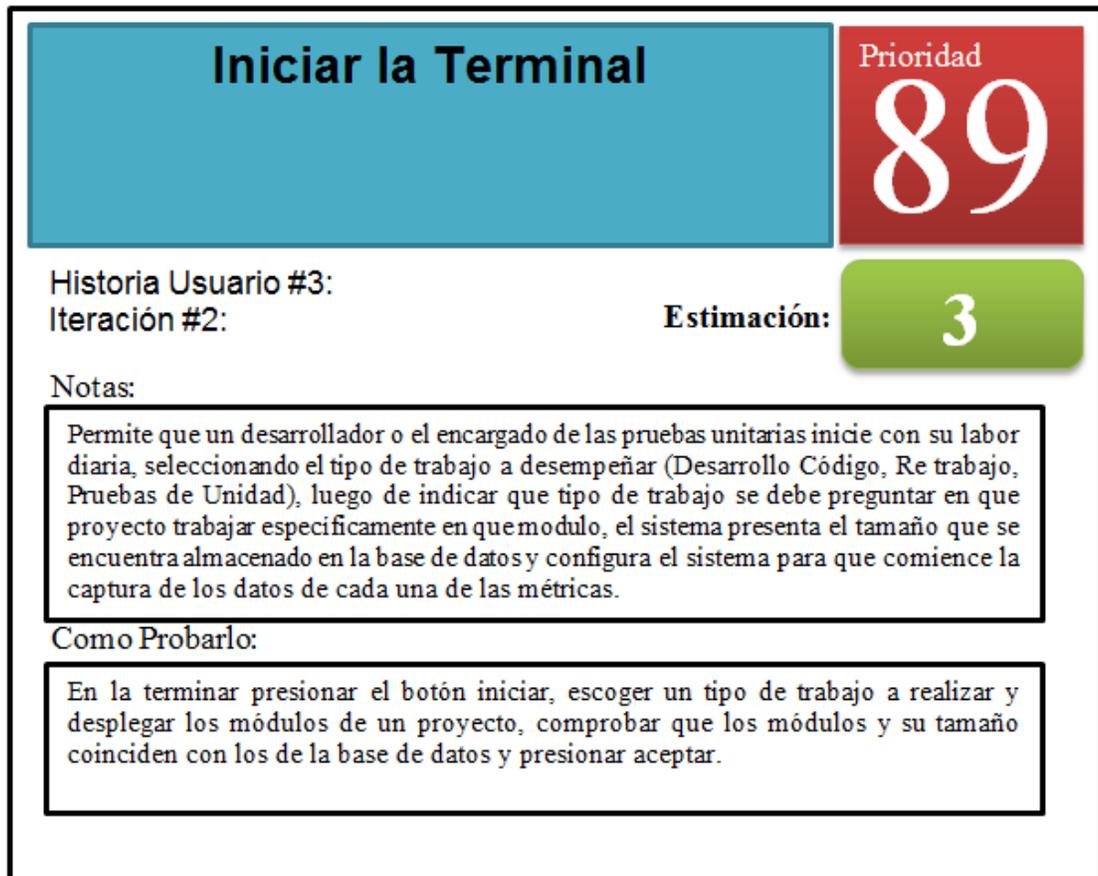


Figura 5. MOCKUP de la Historia de usuario "Iniciar Terminar".



**Iniciar la Terminal**

Prioridad **89**

Historia Usuario #3:  
Iteración #2: **Estimación: 3**

Notas:

Permite que un desarrollador o el encargado de las pruebas unitarias inicie con su labor diaria, seleccionando el tipo de trabajo a desempeñar (Desarrollo Código, Re trabajo, Pruebas de Unidad), luego de indicar que tipo de trabajo se debe preguntar en que proyecto trabajar específicamente en que modulo, el sistema presenta el tamaño que se encuentra almacenado en la base de datos y configura el sistema para que comience la captura de los datos de cada una de las métricas.

Como Probarlo:

En la terminar presionar el botón iniciar, escoger un tipo de trabajo a realizar y desplegar los módulos de un proyecto, comprobar que los módulos y su tamaño coinciden con los de la base de datos y presionar aceptar.

Figura 6. Historia de Usuario "Iniciar Terminal".

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

Como ejemplo, a continuación en la Tabla 40 se listan los criterios de aceptación en la etapa de desarrollo:

#	Criterio	Descripción
3.1	Producto software terminado	Se entrega el producto software terminado, cumpliendo el diseño y las historias de usuario, teniendo en cuenta alcance, restricciones y tiempo acordado.
3.2	Producto listo para implantación	El producto se entrega listo para su implantación en el servidor de producción para el modulo web, y las terminales para ser ejecutas en diferentes equipos.
3.3	Uso de tecnologías acordadas	El producto web se desarrolla completamente en las tecnologías acordadas inicialmente: 1. Stack de tecnologías web AMP (Apache / PHP / MySQL) no dependiente del SO. 2. Desarrollo en los lenguajes PHP (5.x), XHTML (v1.0), JavaScript. 3. Framework MVC. CodeIgniter. Para las terminales de Captura de Datos: 4. Tecnología Java (SWING) no dependiente del SO.
3.4	Soporte para varios terminales	El producto soporta la ejecución en varios equipos para la captura de los datos, sin desarrollo adicional.
3.5	Cumplimiento de las Técnicas	Los reportes que entrega el producto cumplen con la definición del conjunto de métricas del presente trabajo investigativo.

**Tabla 40. Criterios de aceptación de desarrollo.**

Como ejemplo en la Tabla 41 se listan las actividades por iteración definidas en el plan de iteración.

### 4.2.3 Iteración I (arquitectura).

Esta iteración contempla la elaboración del documento del modelo global y la lista priorizada riesgos que se encuentran detallados en anexos (ver CD Anexos, METODOLOGIA). A continuación, como ejemplo se muestra la arquitectura del sistema.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

ITERACION	ACTIVIDADES
I	<ol style="list-style-type: none"> <li>1. Se refina la definición de todas las Historias de Usuario</li> <li>2. Elaborar Arquitectura</li> <li>3. Elaborar Modelo Entidad Relación.</li> <li>4. Elaborar Modelo Clases</li> </ol>
II	<ol style="list-style-type: none"> <li>1. Diseño Pruebas de unidad Modulo gestión de Datos WEB</li> <li>2. Desarrollo Historias de Usuario: <ul style="list-style-type: none"> <li>• LOGIN GESTION DATOS</li> <li>• LISTAR PROYECTOS</li> <li>• AGREGAR UN PROYECTO</li> <li>• GESTION DE MODULOS</li> <li>• GESTION DE TERMINALES</li> <li>• CREA GRAFICO CONTROL</li> </ul> </li> </ol>
III	<ol style="list-style-type: none"> <li>1. Diseño Pruebas de unidad Modulo terminal de captura de los datos</li> <li>2. Desarrollo Historias de Usuario: <ul style="list-style-type: none"> <li>• TERMINAL DE CAPTURA DE DATOS</li> <li>• GESTION DE DATOS</li> <li>• CONFIGURACIÓN DE LA TERMINAL</li> <li>• INICIAR LA TERMINAL</li> <li>• DETENER LA TERMINAL</li> <li>• AGREGAR DEFECTOS</li> </ul> </li> </ol>
IV	<p>Refina el Desarrollo Historias de Usuario:</p> <ul style="list-style-type: none"> <li>• CREA GRAFICO CONTROL</li> <li>• LISTAR PROYECTOS</li> <li>• GESTION DE DATOS</li> <li>• CONFIGURACIÓN DE LA TERMINAL</li> </ul>

**Tabla 41. Actividades para las iteraciones XP.**

### 4.2.3.1 Arquitectura Software.

La gestión de los datos y la configuración se hace a través de un aplicativo web desarrollado en el lenguaje PHP, este puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno, y la captura de las métricas por medio de una terminal diseñada para obtener los datos y almacenarlos en una base de datos (MySQL). Para la terminal se escogió el lenguaje JAVA por su portabilidad en los diferentes Sistemas Operativo. Los dos módulos se desarrollaron empleando el patrón de arquitectura MVC (Modelo-Vista-Controlador) el principal objetivo de este patrón es separar los objetos del dominio que son usados para modelar el entendimiento del problema y la persistencia de los mismos, de los objetos que corresponden a la interfaz gráfica (GUI por sus siglas en inglés) [53]. El sistema está diseñado para soportar dos tipos de usuarios: i) Administradores: son los encargados de la configuración

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

de las terminales y los proyectos asociados a la captura de los datos, ii) Desarrolladores: son los usuarios que capturan los datos de las métricas. La arquitectura del sistema CMSPC se muestra en la Figura 7.

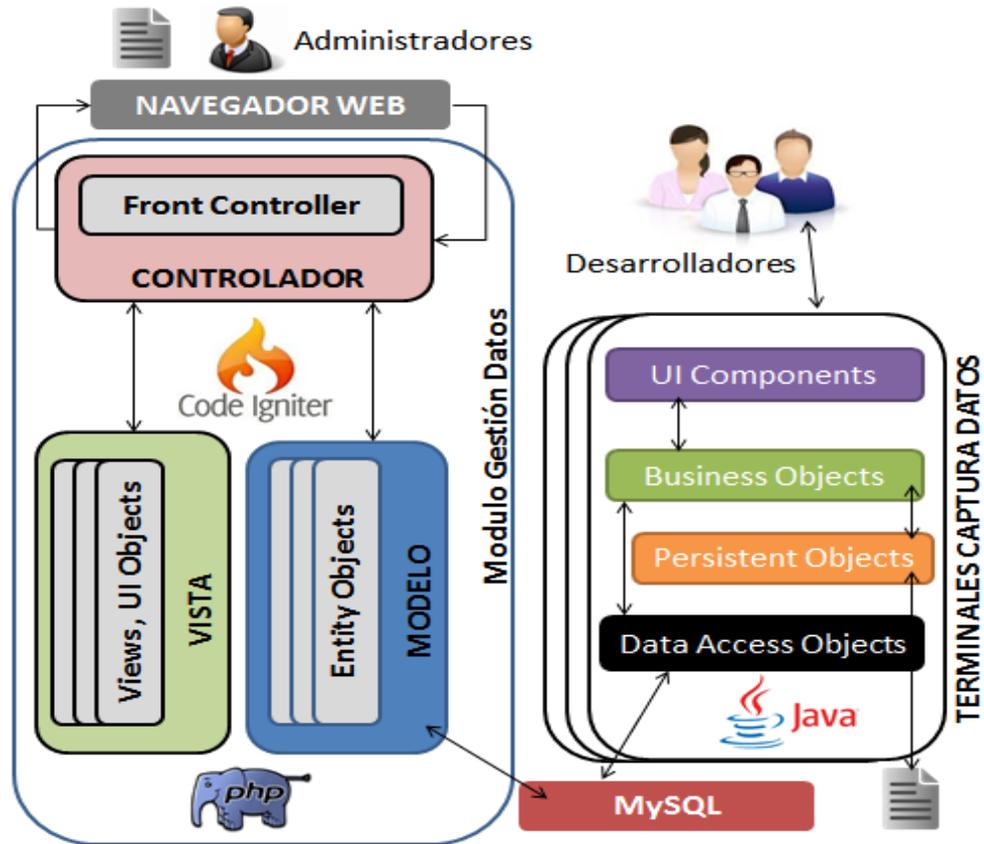


Figura 7. Arquitectura del sistema CMSPC.

## 4.2.4 Iteración II (modulo 1).

Partiendo del Documento del Modelo Global y la metodología definida anteriormente, a continuación muestra el resultado del desarrollo del módulo 1: gestión de datos web.

Esta fase contempla el diseño pruebas de unidad, programación en parejas de las historias de usuario módulo 1, la ejecución pruebas de unidad módulo 1 (ver CD Anexos, METODOLOGIA) y la Integración Continua. A continuación en la Figura 8 se presenta una imagen del módulo web.

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

ID METRICA	ID TERMINAL	ID MODULO	DATO1	DATO2	VALOR	FECHA	TIPO TRABAJO
2 (Densidad Defectos)	1 (PC_PABLO)	1 (Modulo 1 Proyecto: Prototipo)	2	50	0.04	09/09/2013	Pruebas U.
2 (Densidad Defectos)	1 (PC_PABLO)	1 (Modulo 1 Proyecto: Prototipo)	6	50	0.12	09/09/2013	Pruebas U.
2 (Densidad Defectos)	1 (PC_PABLO)	1 (Modulo 1 Proyecto: Prototipo)	7	80	0.0875	11/09/2013	Pruebas U.

Figura 8. Modulo web de gestión de datos.

## 4.2.5 Iteración III (modulo 2).

Partiendo del Documento del Modelo Global y la metodología definida anteriormente, a continuación muestra el resultado del desarrollo del módulo 2: Terminan de captura de datos.

Esta fase contempla el diseño pruebas de unidad, programación en parejas de las historias de usuario módulo 1, la ejecución pruebas de unidad módulo 1 (ver CD Anexos, METODOLOGIA) y la Integración Continua. A continuación en la Figura 9 se presenta una imagen de la Terminal desarrollada en el módulo 2.

Configuracion BD

Direccion BD: 127.0.0.1

Puerto BD: 3306

Usuario BD: root

Nombre BD: BD\_CMSPC

Contraseña BD: root

Probar

Terminal: PC\_GERMAN

Guardar Salir

Estado: CONECTADO A PC\_PABLO

Iniciar Pausar Terminar Configuracion

Salir +1 Defecto Defectos

Figura 9. Terminal para captura de datos.

# CAPÍTULO V

## EVALUACIÓN DEL PROTOTIPO SOFTWARE

---

En este capítulo se realiza el ciclo de evaluación del prototipo software propuesto en el presente trabajo de investigación. Este ciclo de evaluación emplea el método focus group para evaluar el prototipo software. Para esto, se crea un espacio de interacción con los participantes escogidos a través de una sesión de discusión sobre la funcionalidad del prototipo software desarrollado.

El capítulo inicialmente describe un proceso propuesto por [54] para aplicar Focus Group en Ingeniería de Software. Posteriormente se describe la aplicación del método focus group para la evaluación del prototipo software (desarrollado para apoyar la captura de los datos de las métricas propuestas por el presente trabajo de investigación y cuyo desarrollo se describe en el Capítulo IV) realizada con personas con experiencia en las áreas de desarrollo de software, pertenecientes tanto a la academia como a la industria. Finalmente se muestran los resultados y un análisis general sobre la evaluación del prototipo software con éste método.

### **5.1 Proceso empleado para la aplicación de focus group en la evaluación del prototipo software.**

El método focus group se compone de discusiones planeadas y diseñadas cuidadosamente para obtener información relevante de las percepciones de los miembros del grupo, los cuales son seleccionados en base a sus características individuales con relación al tema de interés para la investigación [17]. Cada sesión del método focus group produce información cualitativa y detallada acerca de los objetos de estudio, además este es un método económico y rápido para ser ejecutado [55].

La estructura procedimental que se usó para llevar a cabo el método focus group fue tomada de [54], la cual se compone de cuatro fases que se describen a continuación:

- i) Planeamiento de la investigación:** Se establecen los elementos de contenido y procedimiento que serán aplicados a la sección de discusión.
- ii) Definición de grupos de discusión:** Fundamentado en la caracterización, definición y selección de los participantes.

- iii) **Conducción de la sesión de debate:** Consiste en ejecutar los procedimientos establecidos en la primera fase (Planeamiento de la investigación) con el grupo de discusión seleccionado.
- iv) **Análisis de la información y reporte de resultados:** El propósito es obtener información de valor a partir de los productos de trabajo generados sobre la discusión llevada a cabo.

A continuación, se detallan estas fases describiendo la aplicación del método para la evaluación del prototipo software.

### **5.1.1 Evaluación del prototipo software con focus group.**

Para la evaluación del prototipo software se crea un espacio de interacción entre los participantes escogidos a través de una sesión de discusión sobre la funcionalidad del prototipo software desarrollado. Inicialmente a los participantes se les realiza una inducción sobre las funcionalidades del prototipo definidas en el video introductorio y el manual de usuario, luego cada uno de los participantes utiliza el prototipo software evaluando dichas funcionalidades a través de un “Ejemplo de uso del prototipo” (ver CD Anexo, FOCUS GROUP). Finalmente se diligencia una encuesta con preguntas generales acerca del prototipo y se realiza el debate.

La planeación, ejecución y resultados de dicha evaluación están enmarcados en las fases que se detallan a continuación. Para llevar a cabo estas fases se tuvo en cuenta la ejecución del focus group presentado en [56].

#### **5.1.1.1 Fase de planeamiento de la investigación.**

##### **5.1.1.1.1 Definición del problema de investigación.**

El objetivo de la discusión es evaluar el prototipo software que apoya el levantamiento de las métricas SPC propuestas en el presente trabajo de investigación, desde el punto de vista de su funcionalidad. Como base para el planeamiento se utiliza; requerimientos del prototipo definidos en las historias de usuario, manual de usuario del prototipo software y un video introductorio al uso del mismo.

##### **5.1.1.1.2 Preparación de materiales y métodos, a cumplir por parte del grupo investigador.**

En esta actividad se generan todos los elementos a diligenciar por los participantes como: lista de preguntas generales sobre la evaluación del prototipo software, actividades y roles del focus group, ejemplo de uso del prototipo, y ficha de asistentes a la sesión de debate. Estos elementos se encuentran en el CD

Anexos, FOCUS GROUP. Además se definen los procedimientos y técnicas para llevar a cabo la sesión de debate y la forma en que se deberá obtener la información generada en la sesión. Estos elementos, procedimientos y técnicas deberán ser empleados en fases posteriores, para ello se usan los "Recursos objeto de debate"; siguiendo las tareas mencionadas a continuación:

#### 5.1.1.1.2.1 Definición de estructura.

A continuación se definen los aspectos protocolarios para el debate.

- **Fecha:** Viernes 20 de Septiembre de 2013.
- **Hora de inicio:** 6pm.
- **Hora de finalización:** 7:35pm.
- **Lugar:** Sala 2 - Sistemas, Facultad de Ingeniería Electrónica y Telecomunicaciones - Universidad del Cauca.
- **Actividad:** Sesión de discusión (método Focus Group).
- **Tema a tratar:** Evaluación del prototipo software CMSPC que apoya el levantamiento de métricas SPC.
- **Grupo investigador:** Francisco J. Pino, Pablo F. Miranda y German G. Garzón.
- **Moderador:** German G. Garzón.
- **Relator:** Pablo F. Miranda.
- **Supervisor:** Francisco J. Pino.
- **Participantes:**
  - ✓ Ana Milena Castillo Sánchez (industria)
  - ✓ Marta Lucia Rijas Monte (industria)
  - ✓ Leydi Erazo (industria)
  - ✓ Luis F. Carvajal (industria)
  - ✓ Carlos A. Ardila (academia)
- **Objetivo general:**

Evaluar el prototipo software CMSPC que apoya el levantamiento de las métricas SPC propuestas en el presente trabajo de investigación, desde el punto de vista de su funcionalidad.
- ✓ **Objetivos específicos:**
  - Mostrar en detalle las cuatro métricas SPC propuestas.
  - Presentar la funcionalidad del prototipo software.
  - Conocer las apreciaciones acerca de la funcionalidad del prototipo.
  - Realizar una realimentación con las apreciaciones obtenidas.

La definición del protocolo para la ejecución del focus group se establece como se recomienda en [54] y el documento original del protocolo se encuentra completo en CD Anexos (carpeta FOCUS GROUP).

#### **5.1.1.1.2.2 Definición de instrumentos, materiales y métodos a ser empleados.**

A continuación se presenta el material base para la discusión:

- ✓ Protocolo para la Sesión Focus Group (ver CD Anexo, FOCUS GROUP).
- ✓ Documento de Actividades para la sesión Focus Goup (ver CD Anexo, FOCUS GROUP).
- ✓ Manual de Usuario (ver CD Anexos, PRODUCTO SOFTWARE).
- ✓ Video introductorio acerca de las funcionalidades del prototipo software CMSPC (ver CD Anexos, FOCUS GROUP).
- ✓ Ejemplo de uso del Prototipo CMSPC (ver CD Anexo, FOCUS GROUP).
- ✓ Preguntas generales (ver CD Anexo, FOCUS GROUP).
- ✓ Bitácora de discusión focus group (ver CD Anexo, FOCUS GROUP).
- ✓ Formato de participante (ver CD Anexo, FOCUS GROUP).
- ✓ Elementos para la captura de audio y video.

#### **5.1.1.1.2.3 Definición de métodos de captura y registro de información derivada del debate.**

Debido a la importancia de la discusión generada en la sesión de focus goup, se decide grabar en audio y video la totalidad de la sesión para no perder detalles de la misma. También se captura información textual acerca de las apreciaciones de los participantes en los diferentes artefactos descritos anteriormente (Ejemplo de Uso del Prototipo, Preguntas generales y Bitácora de discusión focus group por parte del relator).

#### **5.1.1.1.2.4 Definición de métodos de análisis de información para generar el procesamiento de lo generado en el debate.**

Posterior a la sesión del focus group, se realiza una revisión de la información recopilada por parte del grupo de investigación para el análisis de dicha información. De esta manera se extrae la información relevante que permite evaluar y refinar el prototipo software a partir de su funcionalidad.

Para el análisis de la información obtenida se establecen ciertas métricas tomadas de la normativa internacional ISO/IEC 9126. Estas métricas permiten medir la característica de Funcionalidad del prototipo software y están fuertemente relacionadas con el cumplimiento de los requerimientos y a la satisfacción del

usuario con respecto a las funcionalidades ejecutadas. A continuación se muestran dichas métricas:

- **Adecuación funcional**

- ¿Qué tan adecuadas son las funciones chequeadas?
- Para esto se usan las historias de usuario
- Formula:  $X = 1 - A / B$   
A= número de funciones en las que la evaluación detecta problemas.  
B= Número de funciones chequeadas. " $0 \leq X \leq 1$ ", Cuanto más cerca de 1, es más adecuado.

- **Cobertura de implementación funcional**

- ¿Cuál es la razón de las funciones implementadas correctamente según lo especificado sobre cómo se requieren?
- Para la evaluación contrastar con: i) especificación de requisitos y 2) revisión por elemento funcional.
- Formula:  $X = A/B$ .  
A= Número de implementaciones correctas (funciones que se confirmaron en la evaluación). B = Número de funciones que se describen en especificaciones de requisitos. " $0 \leq X \leq 1$ , Cuanto más cerca de 1, es más adecuado".

- **Completitud de la implementación funcional**

- ¿Qué tan completa es la puesta en práctica de acuerdo con especificaciones de requisitos?
- Al realizar las pruebas funcionales, contar el número de funciones que faltan detectadas en la evaluación y comparar con el número de función descrita en las especificaciones de requisitos
- $X = 1 - A / B$   
A= Número de funciones que faltan en la evaluación. B= Número de funciones que se describen en las especificaciones de requerimientos.  $0 \leq X \leq 1$ , cuanto más cerca 1 es el mejor.

- **Precisión**

- ¿Con qué frecuencia los usuarios finales encuentran resultados con la precisión adecuada?
- Anote el número de resultados con la precisión adecuada.
- $X = A / T$ .  
A = Número de resultados encontrados por el/los usuarios con nivel de precisión diferente del necesario. T= tiempo de operación.  $0 \leq X$ , cuanto más cerca de 0 es el mejor.

### 5.1.1.2 Fase de definición de grupos de discusión

#### 5.1.1.2.1 Selección de participantes

La selección del grupo de participantes es una actividad realizada por el grupo investigador, a través de las siguientes tareas:

- ✓ *Definición del perfil de participante:* Se establece la condición para la selección de participantes. Esta determinan que el participante debe pertenecer a la academia y/o a la industria del software, con experiencia de trabajo en el proceso de desarrollo de software.
- ✓ *Identificación de potenciales participantes:* A partir de la condición de perfil, en esta sección se identifican aquellas personas que cumplen con dicha condición y se presentan a continuación:
  - ✓ Ana Milena Castillo Sánchez, vinculada al sector industria en la empresa Seratic Ltda. con un tiempo de experiencia de 5 años como Desarrolladora de software.
  - ✓ Marta Lucia Rojas Montes, vinculada al sector industria en la empresa Nexura Internacional S.A.S, con un tiempo de experiencia de 1 año como líder de pruebas de software y servicios.
  - ✓ Leydi Erazo, vinculada al sector industria en la empresa HDS Colombia, con un tiempo de experiencia de 2 meses como Tester.
  - ✓ Luis F. Carvajal, vinculado al sector industria en la empresa HDS Colombia, con un tiempo de experiencia de 3 meses como Desarrollador Java Jr.
  - ✓ Carlos A. Ardila, vinculado al sector académico como Docente de Planta de la Universidad del Cauca, con un tiempo de experiencia de 11 años y 6 meses.

El perfil de cada participante se encuentra diligenciado en el “Formato de Asistente a la Sección de Focus Group” (ver CD Anexos, FOCUS GROUP/FORMATO DE ASISTENTES).

### 5.1.1.3 Fase de conducción de la sesión de debate.

#### 5.1.1.3.1 Secuencia básica.

La ejecución del debate fue coordinada por el moderador y supervisor, y efectuada por los participantes. Para éste se usaron los materiales y artefactos resultantes de las fases anteriores. A continuación se describe la manera en la que se realizó el debate:

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

- ✓ **La ejecución del protocolo:** Inicialmente los participantes diligenciaron el formato de asistencia. Después se realizó una corta presentación de los integrantes del grupo investigador por parte del supervisor y se contextualizó a los participantes en el tema a tratar en la sesión. El moderador realizó la presentación del protocolo a seguir junto con los objetivos de la sesión y se realizó una pequeña inducción sobre las funcionalidades del prototipo CMSPC para dar inicio al “Ejemplo de uso del prototipo” desarrollado por cada uno de los participantes, durante el cual cada participante diligenció las preguntas relacionadas con la funcionalidad del prototipo. Al terminar el ejemplo de uso se diligenció la encuesta con las preguntas generales para entrar en el debate final de la sección (generando la bitácora de discusión focus group). Por último se ejecutó el cierre de la sesión en la que se agradece a los participantes de la misma. A continuación en la Tabla 42 se muestra una tabla con los tiempos de ejecución de las actividades realizadas en la sesión de debate:

Actividad	Tiempo de Ejecución	
	Hora	Total
Diligenciar formato de asistencia	6:20pm-6:28pm	8 minutos
Contextualización y presentación del protocolo	6:28pm-6:32pm	5 minutos
Presentación de Prototipo CMSPC	6:32pm-6:44pm	12 minutos
Desarrollo del Ejemplo de uso	6:44pm-7:45pm	61 minutos
Encuesta general	7:45pm-7:55pm	10 minutos
Debate	7:55pm-8:30pm	35 minutos
Cierre	8:30pm-8:34pm	4 minutos

**Tabla 42. Tiempos de ejecución de las actividades en la sesión de focus group.**

Cabe resaltar que los tiempos de ejecución reales tuvieron desfases que resaltan con los tiempos estipulados en el documento “ACTIVIDADES PARA LA SECCIÓN FOCUS GOUP”, sobre todo en el desarrollo del “Ejemplo de uso del prototipo” que estaba estipulado para 30 minutos y tuvo una duración de casi el doble. En este caso el moderador permitió la prolongación del tiempo para desarrollar dicho ejemplo debido a que este constituía una fuente de información relevante para obtener realimentación de las oportunidades de mejora del prototipo y la discusión de la sección. Sin embargo se realizaron ajustes de las actividades para recuperar los desfases de tiempo como el diligenciamiento del formato de asistentes al inicio de la sección.

- ✓ *La aplicación de técnicas, materiales y métodos:* La sesión de debate se basó en los materiales y métodos para la captura de información establecidos anteriormente (ver sección 5.1.1.1.2.2), como son:
  - Protocolo para la Sesión Focus Group (ver CD Anexo, FOCUS GROUP).
  - Documento de Actividades para la Sesión Focus Goup (ver CD Anexo, FOCUS GROUP).
  - Manual de Usuario (ver CD Anexos, PRODUCTO SOFTWARE).
  - Ejemplo de uso del Prototipo CMSPC (ver CD Anexo, FOCUS GROUP).
  - Preguntas Generales (ver CD Anexo, FOCUS GROUP).
  - Bitácora de Discusión focus group (ver CD Anexo, FOCUS GROUP).
  - Formato de Participante (ver CD Anexo, FOCUS GROUP).
  - Elementos para la Captura de Audio y Video.

#### **5.1.1.3.2 Captura de información.**

El relator fue el encargado de coordinar la captura de información, registrando los conceptos, detalles, características y aportes más relevantes ofrecidos por los participantes sobre cada una de las actividades debatidas en esta sesión. Además fueron empleadas como técnicas de captura de información: la grabación audio y los formatos de captura de información (ver CD Anexos, FOCUS GROUP). Esta información se convirtió en el insumo primordial para la siguiente fase.

#### **5.1.1.4 Fase de Análisis de Información y Reporte de Resultados.**

##### **5.1.1.4.1 Análisis de información.**

Se realizaron y cumplieron las siguientes actividades para el análisis de información: análisis de encuestas, análisis de los archivos de audio, y establecimiento de puntos fuertes y débiles del prototipo con la información obtenida del debate. A partir de los puntos débiles se identificó la información correspondiente para establecer oportunidades de mejora a realizar en el prototipo.

##### **5.1.1.4.1.1 Análisis de encuestas y los archivos de audio.**

Aquí se realizó un análisis de la información obtenida de las respuestas a las preguntas expuestas en el “Ejemplo de uso del prototipo” y los archivos de audio del focus group. Mediante este análisis se determinaron las medidas de las métricas de funcionalidad planteadas en la sección 5.1.1.1.2.4 (Adecuación funcional, Cobertura de implementación funcional, Completitud de la implementación funcional y Precisión) como se muestra a continuación:

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

### ➤ **Adecuación funcional:**

Para el análisis de la información obtenida por las preguntas planteadas en el “Ejemplo de uso del prototipo”, diligenciadas por los participantes del debate (ver CD Anexos, FOCUS GROUP/EJEMPLO DE USO DEL PROTOTIPO CMSPC), se contrastan la respuesta a las preguntas con las funcionalidades estipuladas en las Historias de Usuario específicas (cada HU representa una funcionalidad del sistema). Las preguntas tratan de evaluar una funcionalidad especificada por una HU, como se muestra en la Tabla 43.

HU	Id Preguntas
HU1-Web	P1
HU2-Web	P2
HU3-Web	P4
HU4-Web	P3
HU5-Web	P5
HU6-Web	P6
HU2-Terminal	P7 P8
HU3-Terminal	P9 P10 P13 P14 P18 P19
HU4-Terminal	P11 P12 P16 P17 P20 P21
HU5-Terminal	P15

Tabla 43. Relación de las preguntas del Ejemplo de uso del prototipo con las historias de usuario definidas.

Un ejemplo de la relación entre una pregunta del “Ejemplo de uso del prototipo” y su HU asociada, se muestra a continuación:

**Actividad 1:** Acceder al módulo web del prototipo a través del Acceso directo.

**P1** - ¿El modulo web del prototipo se puede acceder a través de su navegador web? (**HU1-Web**)

(Si: \_\_\_\_, No: \_\_\_\_, NS / NR: \_\_\_\_, Comentarios: \_\_\_\_\_)

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

En seguida se grafican los resultados de las 21 preguntas del ejemplo de uso del prototipo que se asocian con 10 funcionalidades/HU (ver Gráfico 1). Los resultados obtenidos en dicho “ejemplo de uso del prototipo” se encuentran en CD Anexos FOCUS GROUP/EJEMPLO DE USO DEL PROTOTIPO CMSPC.

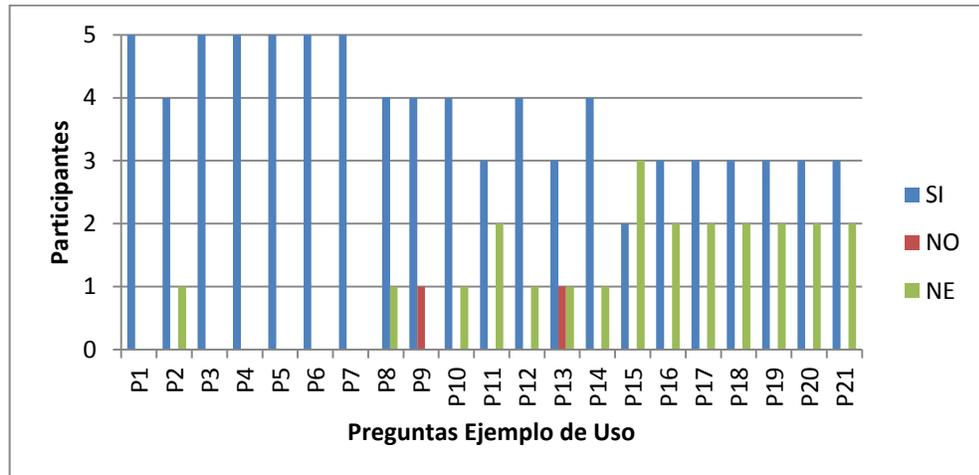


Gráfico 1. Resultado de las preguntas realizadas en el Ejemplo de uso del prototipo CMSPC.

Para el análisis del Gráfico 1 se debe tener en cuenta que los participantes para algunas preguntas no marcaron ninguna de las opciones ofrecidas, por lo que ese resultado se tabula como no encontrado (NE) y no se tiene en cuenta para el análisis.

En las preguntas P9 y P13 se detecta una inconformidad (ver Gráfico 1). Las preguntas P9 y P13 están relacionadas con la H3 de la terminal como se muestra en la Tabla 43. La inconformidad está ligada con que al usuario no se le permite elegir su actividad a desempeñar en la terminal (desarrollo, pruebas y retrabajo), esta se elige automáticamente dependiendo del estado del módulo a trabajar. A continuación se detallan las preguntas P1, P2 y se muestra su respuesta en Gráfico 2 y Gráfico 3 respectivamente:

- ✓ **Actividad:** Iniciar la actividad en modo Desarrollo, con el proyecto y el modulo agregados en los puntos anteriores.
  - P9 - ¿El prototipo permite al desarrollador seleccionar: i) su actividad a desempeñar (Desarrollo), ii) el proyecto y iii) el modulo específico a trabajar? (HU3-Terminal)**  
(Si: \_\_, No: \_\_, NS / NR: \_\_, Comentarios: \_\_)

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

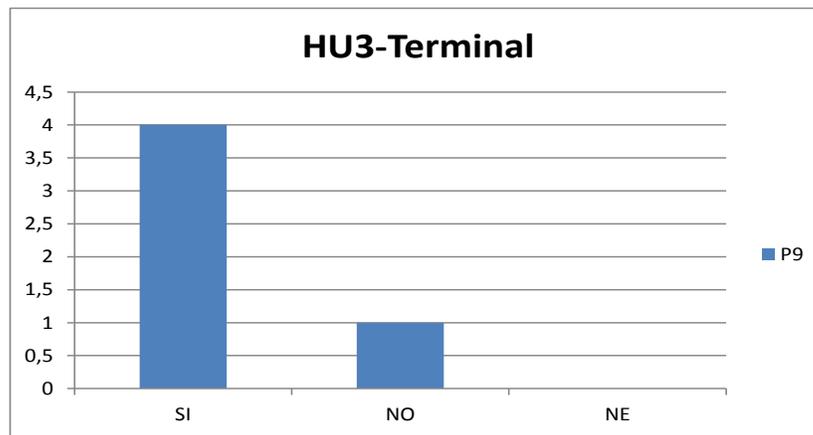


Gráfico 4. Resultado de la pregunta 9 del ejemplo de uso del prototipo.

- ✓ **Actividad:** Iniciar la Terminal (botón Iniciar) y elegir la actividad de "Pruebas Unitarias" con el Proyecto y el modulo desarrollado en los puntos anteriores. El modulo debe tener un tamaño mayor a 0 para realizar pruebas unitarias.

**P13** - ¿El prototipo permite al usuario seleccionar: i) su actividad a desempeñar (Pruebas Unitarias), ii) el proyecto y iii) el modulo específico a trabajar? (**HU3Terminal**)

(Si: \_\_, No: \_\_, NS / NR: \_\_, Comentarios: \_\_)

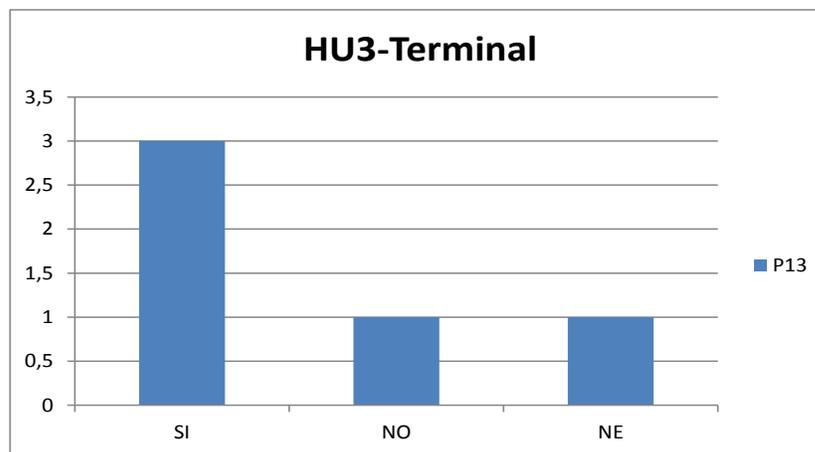


Gráfico 5. Resultado de la pregunta 13 del ejemplo de uso del prototipo.

Teniendo en cuenta esto procedemos a calcular la métrica Adecuación funcional:

$$X = 1 - (A / B)$$

A= número de funciones en las que la evaluación detecta problemas.

B= Número de funciones chequeadas.

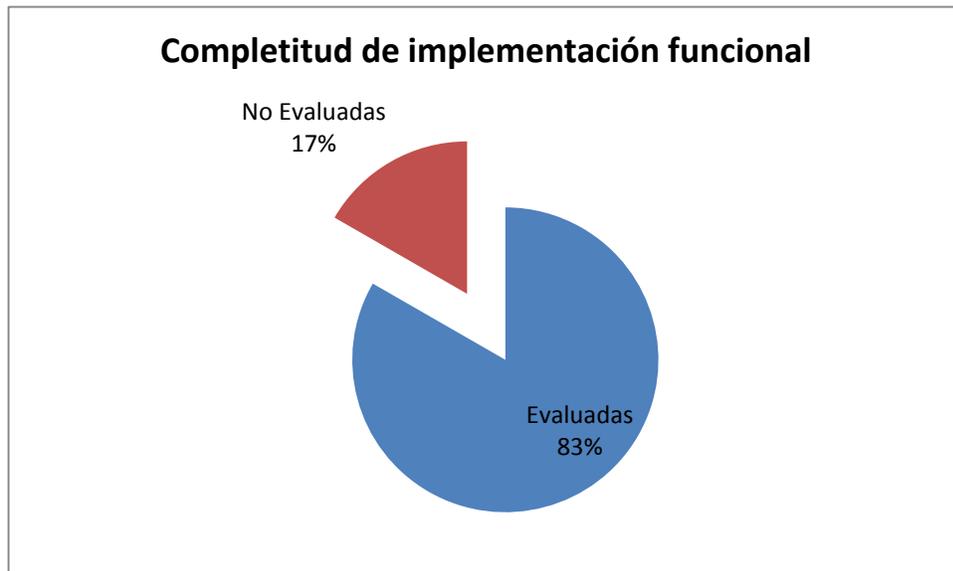
## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

X Cuanto más cerca de 1, es más adecuado.

$X = 1 - (1/10) \rightarrow X = 0.9$ . Este resultado nos dice que hay un alto grado de adecuación funcional.

### ➤ **Completitud de la implementación funcional**

Para esta métrica se debe tener en cuenta que se especifican 12 HU en el desarrollo del prototipo software CMSPC pero en el “Ejemplo de uso del prototipo” se evalúan 10 HU (ver Gráfico 6). Como se observa en la Tabla 44, las dos HU que no se evaluaron son: HU7-Web del módulo web que especifica la funcionalidad del grafico de control (que es un plus para el prototipo, ya que no hace parte del levantamiento de los datos de la métrica) y HU1-Terminal, la cual está conformada por las HU específicas (HU2, HU3, HU4, HU5) relacionadas con la terminal.



**Gráfico 6. Completitud de la implementación funcional.**

A continuación se calcula la métrica “Completitud de la implementación funcional”:

$$X = 1 - (A / B)$$

A= Número de funciones que faltan en la evaluación.

B= Número de funciones que se describen en las especificaciones de requerimientos.

$0 \leq X \leq 1$ , cuanto más cerca de 1 es mejor.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

$X = 1 - (2/12) \rightarrow X = 0.83$ , La completitud de implementación funcional es parcial por razones expuestas anteriormente.

### ➤ Cobertura de implementación funcional

Partiendo de la información planteada en las anteriores métricas (Adecuación funcional y Completitud de la implementación funcional) en las que se encontró que el número de implementaciones correctas son 9 de las 10 evaluadas y con un total de 12 HU descritas en los requerimientos (ver Tabla 44).

Historia de Usuario	Correcto	Incorrecto	No Evaluada
HU1-Web	x		
HU2-Web	X		
HU3-Web	X		
HU4-Web	X		
HU5-Web	X		
HU6-Web	x		
HU7-Web			x
HU1-Terminal			x
HU2-Terminal	x		
HU3-Terminal		x	
HU4-Terminal	x		
HU5-Terminal	x		

Tabla 44. Resultado de las evaluaciones de las HU en el ejemplo de uso del prototipo.

A continuación se calcula la métrica “Cobertura de implementación funcional”:

$$X = A/B.$$

A= Número de implementaciones correctas (funciones que se confirmaron en la evaluación).

B = Número de funciones que se describen en especificaciones de requisitos.

" $0 \leq X \leq 1$ , Cuanto más cerca de 1, es más adecuado".

$X = 9/12 \rightarrow X = 0.75$ , La cobertura de implantación funcional es parcial por razones expuestas anteriormente.

➤ **Precisión**

La ejecución del “Ejemplo de uso del prototipo” tuvo una duración promedio de 61 minutos y dos participantes encontraron un resultado con nivel de precisión diferente del necesario. Entonces, se calcula el nivel de precisión de la siguiente manera:

$$X = A / T.$$

A = Número de resultados encontrados por el/los usuarios con nivel de precisión diferente del necesario.

T= tiempo de operación.

X cuanto más cerca de 0 es el mejor.

$X = 2/61 \rightarrow X = 0.03$ , La precisión en cuanto a las funciones es muy adecuada.

**5.1.1.4.1.2 Establecimiento de aspectos positivos, negativos y/o observaciones del prototipo.**

De la encuesta con las preguntas generales (ver CD Anexos, FOCUS GROUP/ PREGUNTAS GENERALES) se extraen los aspectos positivos, aspectos negativos y/o observaciones propuestas por los participantes para ser aplicadas al prototipo software. A partir de los aspectos negativos y observaciones se identificó la información correspondiente para establecer oportunidades de mejora a realizar en el prototipo. En la Tabla 45 se describen dichos aspectos tanto para el modulo web como para la terminal.

**5.1.1.4.1.3 Oportunidades de mejora consideradas para el prototipo.**

Con base en la Tabla 45, el documento “Resultados del Ejemplo de Uso” y la bitácora de discusión focus group (ver CD Anexos, FOCUS GROUP), se extraen las oportunidades de mejora que se consideraron para el prototipo software. Para esto y con el fin de llevar una trazabilidad, se han clasificado las oportunidades de mejora obtenidas por Historia de Usuario tal y como se muestra a continuación:

**HU4-Web: Agregar un Proyecto.**

- ✓ No debe poderse establecer un proyecto como terminado en el momento de ingresarlo.
- ✓ La fecha de terminación del proyecto debe ser mayor a la fecha ingreso.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

<b>Modulo Web</b>	
<b>Aspectos Positivos</b>	<ul style="list-style-type: none"> <li>✓ Facilidad de lectura y Registro.</li> <li>✓ Aspecto Sobrio y Liviano.</li> <li>✓ Facilidad para captar la agrupación de información.</li> <li>✓ El modulo es amigable y permite ver gráfica y descriptivamente detalles.</li> <li>✓ Es una herramienta útil para la gestión de los datos de las métricas.</li> <li>✓ Se tuvieron en cuenta métricas relacionadas con el día de la VSEs o pymes como esfuerzo representado en el porcentaje de trabajo o avance de la funcionalidad, y presenta al jefe el avance total.</li> </ul>
<b>Aspectos Negativos</b>	<ul style="list-style-type: none"> <li>•No exceder el manejo de cifras decimales para los campos numéricos (máximo 5).</li> <li>•La inclusión de iconos.</li> </ul>
<b>Observaciones</b>	<ul style="list-style-type: none"> <li>○ Incluir una pequeña ayuda en línea para el uso de la aplicación.</li> <li>○ Mejorar las alertas en la aplicación, de manera que el usuario sepa que debe hacer para gestionar la aplicación en general.</li> <li>○ Al reiniciar la aplicación se debe pedir el Login nuevamente. Ya que cualquier usuario podrá iniciar la aplicación.</li> </ul>
<b>Terminal</b>	
<b>Aspectos Positivos</b>	<ul style="list-style-type: none"> <li>✓ Carga Rápido.</li> <li>✓ Los menús tienen la información necesaria.</li> <li>✓ Permite gestionar el progreso.</li> <li>✓ Es una herramienta amigable para el usuario. Y muy fácil de manejar.</li> <li>✓ Es simple, no genera distracción y realiza su función correctamente.</li> </ul>
<b>Aspectos Negativos</b>	<ul style="list-style-type: none"> <li>•El valor llamado “Porcentaje” debe redefinirse y que sea calculado automáticamente.</li> <li>•Ampliar el espacio para el menú principal.</li> <li>•Debe ser más amigable y descriptiva.</li> </ul>
<b>Observaciones</b>	<ul style="list-style-type: none"> <li>○ Especificar mejor el inicio de los proyectos, módulos y terminales para no generar confusión al usuario.</li> </ul>

**Tabla 45. Aspectos positivos, aspectos negativos y/o observaciones consideradas de las preguntas generales.**

**HU5-Web: Gestión de Módulos.**

- ✓ No se debe permitir el ingreso de un módulo sin sus datos marcados como obligatorios.
- ✓ No exceder el manejo de cifras para los campos numéricos.

**HU6-Web: Gestión de Terminales.**

- ✓ Inicializar el estado de las terminales como Inactivos.

**HU2-Terminal: Configuración de la Terminal.**

- ✓ El campo contraseña en la terminal debe estar “Camuflado” o “Enmascarado”.

**HU3-Terminal: Iniciar Terminal.**

- ✓ Remover el cronometro para que no se evidencie el control de tiempos en las actividades del usuario.
- ✓ Ampliar el espacio para el menú principal para mejorar la distribución de los botones.

**HU4-Terminal: Detener la Terminal.**

- ✓ El valor llamado "Porcentaje" debe redefinirse y que sea calculado automáticamente.
- ✓ Las etiquetas (Dato 1, Dato 2 y Valor) deben ser más descriptivas.

**5.1.1.5 Productos de trabajo.**

A lo largo del planeamiento y ejecución del focus group se generaron una serie de documentos (productos de trabajo), que relacionan información acerca de conceptos, criterios, instrumentos, métodos y materiales empleados en este proceso. Los productos de trabajo se encuentran en el CD Anexo/FOCUS GROUP y son descritos a continuación:

- ✓ Recursos objeto de debate, fueron los documentos que contextualizaron al grupo de investigadores a fin de dar inicio al proceso de planeamiento. Estos son: Prototipo software CMSPC, Manual de usuario y un video introductorio al uso del mismo.
- ✓ Reporte de contexto y reporte de estructura, estos fueron los documentos donde se especificaron las características del contexto del debate, junto con conceptos y alcances para la aplicación de focus group. Tales documentos

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

son: Protocolo para la Sección Focus Group y Documento de Actividades para la sección Focus Goup.

- ✓ El reporte de participantes esta descrito en el “Formato de Asistente”, documento que describe las características de las personas que participaron en el debate.
- ✓ El reporte de análisis de resultados se especifican en los documentos: Resultados de ejemplo de uso, Resultados de preguntas generales y en la sección 5.1.1.4.1.1 del presente capitulo.

### **5.2 Análisis de la Evaluación.**

En el Capítulo I se describe la estrategia de investigación propuesta para el desarrollo del presente trabajo de investigación, el cual ha generado como uno de los productos finales al prototipo software CMSPC para la captura de las métricas SPC (propuestas en el Capítulo III) en VSEs. El grupo investigador considero que era necesario evaluar el prototipo propuesto mediante el juicio de expertos, debido a su bajo costo y los beneficios como la obtención de información rápida y pertinente (sugerencias y observaciones). Dicha evaluación se enfocó en la funcionalidad y aplicabilidad del prototipo software CMSPC en las pequeñas organizaciones a través de las experiencias aprendidas dentro del contexto de cada participante tanto de la academia como de la industria.

A través de los aportes obtenidos de la sección de focus group, el grupo investigador pudo hacer una realimentación que permitió realizar los ajustes considerados precisos para la funcionalidad del prototipo software CMSPC y que estaban dentro del alcance del presente trabajo de investigación. Dichas mejoras están incluidas en el prototipo software final que se presenta en el CD Anexo (carpeta PRODUCTO SOFTWARE). Aspectos como el manejo de usuarios y los gráficos de control para las métricas de Densidad de defectos, Retrabajo y Valor ganado, que se generaron como observaciones en la sesión de debate, se consideran como trabajo fututo que permita evolucionar el prototipo software propuesto a un sistema completo y aplicable un ambiente real de empresa.

En la evaluación con el método focus group también se pudo considerar la utilidad de las métricas propuestas para la pequeña empresa a través de las experiencias descritas por los participantes de la industria. Las personas que participaron durante la sesión de debate y que pertenecen a la industria del software, expresaron que el prototipo software era útil y beneficioso para controlar el proceso de construcción de software y con algunas mejoras se podría ajustar a las necesidades y en general al contexto de las pequeñas organizaciones, teniendo en cuenta sus características.

# CAPÍTULO VI

## CONCLUSIONES Y TRABAJO FUTURO

---

A continuación en este capítulo se describe un sumario del trabajo realizado, las conclusiones (lecciones aprendidas) y el trabajo futuro generadas a partir del presente trabajo de investigación, en el que se proponen cuatro métricas SPC (Valor Ganado, Densidad de Defectos, Productividad y Porcentaje Retrabajo) relacionadas con el proceso de construcción de las VSEs. Estas métricas fueron identificadas a partir de la ejecución de una estrategia de investigación basada en AR (investigación - acción) bifurcado (ver Capítulo I, sección 1.2).

### 6.1 Sumario.

Una responsabilidad clave para gestionar procesos software son medición y control, sin embargo la mayoría de las pequeñas organizaciones software (VSEs) no los llevan a cabo. En este sentido, es importante apoyar a estas organizaciones en la medición y control del proceso y producto software, ya que este aspecto es un factor determinante de la calidad del producto final y la productividad del proceso de desarrollo. En este sentido, este trabajo de investigación presenta las métricas del proceso de construcción que pueden ser empleadas por VSEs para la aplicación del Control Estadístico de Procesos (SPC).

Para la definición y desarrollo de la presente investigación, se estructuró una estrategia de investigación basada en la metodología de investigación AR (investigación - acción) bifurcado [7] [8]. La estrategia parte de un ciclo de investigación inicial donde se identifican tres problemas de tipo conceptual, metodológico y técnico. Estos problemas permiten dividir el trabajo en tres ciclos de investigación independientes: ciclo conceptual, ciclo metodológico y ciclo técnico.

En el ciclo conceptual se investigó y obtuvo la información vital para contextualizar el presente trabajo. Para este, se realizó una búsqueda en la literatura y se hizo un análisis de los resultados sobre los estudios de métricas software y el uso de SPC en el contexto del software. Este ciclo tuvo una duración de aproximadamente 2 meses en concluirse generando como resultado el marco teórico y el estado del arte presentados en el capítulo II del presente trabajo.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

El ciclo metodológico presentó la estrategia utilizada para obtener el conjunto de métricas apropiadas para SPC en las VSEs. Este ciclo estaba compuesto por tres fases que permitieron responder a la pregunta de investigación ¿Qué métricas son adecuadas para gestionar el proceso de construcción desde un enfoque SPC en VSEs? En la primera fase se analizó el proceso de construcción de software y las métricas relacionadas con este proceso, también se obtuvieron las practicas software más comunes en las pequeñas organizaciones, para finalmente explorar las métricas utilizadas en la aplicación de SPC en casos de estudio dentro de la industria software. En la segunda fase se recopilaron las métricas software del proceso de construcción y se relacionaron con las prácticas empleadas en las pequeñas organizaciones. En esta misma fase también se organizó y estructuró un nuevo Instrumento para la Validación de la Idoneidad de Métricas para SPC (IVIM) que fue usado para evaluar las métricas filtradas. En la tercera y última fase se listaron el conjunto de métricas adecuadas para la aplicación de la técnica SPC en pequeñas organizaciones y se identificaron las necesidades de información para dichas métricas.

El ciclo técnico siguió la metodología de desarrollo XP empleada para la construcción del prototipo software el cual automatiza la captura de los datos y facilita la actividad de medición en el proceso de construcción de las VSEs. Este prototipo es una herramienta software creada para apoyar la captura de las métricas propuestas. La gestión de los datos y la configuración se realizan a través de una aplicación web y la obtención de los datos se soporta a través de una terminal diseñada para capturar y almacenar los datos de las medidas en una base de datos. Por otra parte, en este ciclo también se realiza la validación del prototipo software a través del método Focus Group, con resultados satisfactorios debido a la rigurosidad del método. De esta evaluación también se evidenció la utilidad del prototipo para las pequeñas organizaciones donde laboran los participantes de la sesión, ya que en dichas organizaciones no se realizan actividades de medición que permitan el control de los procesos.

### **6.2 Conclusiones.**

Según la literatura, la aplicación de SPC a los procesos software ha sido una tarea difícil. Si bien se sugiere la técnica SPC de gráficos de control para proporcionar el control de los procesos software y lograr niveles de madurez superiores para las organizaciones, son muy pocos los detalles y directrices relacionados con las métricas y sus datos que faciliten la aplicación SPC en este tipo de procesos. Sin embargo, es claro que la técnica de gráficos de control es la herramienta SPC mas empleada en el contexto del software ya que facilita la detección de problemas

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

gracias a su efectividad y su aporte como una interfaz visual basada en fundamentos científicos.

Por otra parte, en la literatura se encuentra gran variedad de métricas software, pero con un alto grado de ambigüedad en su definición. Esto genera un problema cuando se intenta identificar métricas que apoyen la medición de los procesos software, ya que existe una gran cantidad y variedad de métricas software que se dificulta obtener un conjunto apropiado de métricas que esté de acuerdo a los objetivos de medición.

Para el caso de la captura de los datos, se debe medir en periodos cortos, frecuentemente y de manera automatizada, ya que la granularidad de los datos es un factor importante para los resultados de SPC. Hay que tener en cuenta que si los periodos de medición son muy grandes, los datos de la medición solo servirían para confirmar problemas de software que ya es tarde resolver.

Un problema importante para la aplicación correcta de la técnica SPC de gráficos de control en las VSEs es que en este tipo de organizaciones la cantidad de los datos de las métricas son escasos y dependen de las características propias del proceso. Un criterio de éxito para aplicar SPC es que el proceso debe estar definido y ejecutado en la organización. Por otra parte la actividad de medición es posible a través de la herramienta software CMSPC que soporta la captura de los datos de las métricas propuestas. Así, la aplicación exitosa de SPC no requiere de un alto nivel de madurez por parte de la organización.

Un conjunto de métricas apropiadas para SPC del proceso de construcción de software de las VSEs y una herramienta software que apoye la captura de los datos (medidas) de dichas métricas podrían apoyar a las VSEs en la implementación del control estadístico de dicho proceso. Esto permitiría disminuir la variabilidad de los proyectos al contar con un proceso estable (con un comportamiento iterativo y predecible), logrado a partir de la aplicación de SPC.

### **6.3 Reflexión.**

La estrategia de investigación propuesta facilitó la culminación del presente trabajo de investigación debido a que; i) el primer ciclo conceptual permitió la contextualización del problema, ii) el ciclo metodológico a través de sus fases facilitó responder a la pregunta de investigación y iii) el ciclo técnico junto con el de evaluación permitieron obtener un prototipo software funcional para la captura de los datos de las métricas. Esta estrategia se respaldó con reuniones periódicas del grupo investigador y el director, donde se debatían y aclaraban los aspectos importantes que influían en el curso de la investigación. El trabajo mancomunado

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

del grupo investigador fue inspirado en la programación en parejas de la metodología XP, donde los dos investigadores participaron en un esfuerzo combinado en un sitio de trabajo específico, dedicando en promedio 50 horas semanales. También cabe resaltar la importancia que tiene el uso de servicios de “cloud computing” (como Dropbox) que permitió el almacenamiento y la sincronización de la información generada por el trabajo de investigación.

En el ciclo conceptual se investigó y obtuvo la información vital para contextualizar el presente trabajo. Este ciclo permitió visualizar el estado actual de la aplicación de la técnica SPC de gráficos de control en el contexto del software, donde se encontraron casos de estudio fallidos que generaban conclusiones en contra de la aplicabilidad de dicha técnica en el área del software, sin embargo se evidenció que el problema estaba relacionado en gran medida a la elección de las métricas a usar para la aplicación de SPC. Dichas falencias fortalecieron el trabajo investigativo realizado. En este ciclo también se encontraron limitaciones relacionadas con el acceso a la información, debido a que solo se tenían las fuentes para la búsqueda de información que podían ser accedidas y que estaban disponibles en el servicio Web de internet de la Universidad del Cauca.

El ciclo metodológico presento la estrategia utilizada para obtener el conjunto de métricas apropiadas para SPC en las VSEs. Para la identificación del problema fue necesario modelar el proceso de construcción de software ya que en la literatura solo se encontraba su definición en la normativa ISO/IEC 12207 que es la normativa más detallada acerca del ciclo de vida del software, para esto también se utilizó la normativa ISO/IEC 15504 ya que describía de forma precisa las diferentes prácticas, resultados y productos de trabajo que componen a dicho proceso. Por otra parte, la estrategia utilizada para la búsqueda de métricas de construcción se basó en el método de revisiones sistemáticas que facilitó la identificación formal de 69 métricas relacionadas con el proceso de construcción, sin embargo muchas de estas métricas carecían de información por lo que fue necesario realizar una búsqueda extra de información de las mismas de tal manera que se pudieran clasificar en las diferentes prácticas de las VSEs. Además se identificaron las prácticas que pertenecen a la disciplina de Implementación, donde se implementa y prueba el código fuente, de manera que se hace un análisis comparativo con las tareas del proceso de construcción. Por otra parte, se encontraron pocos casos de estudio exitosos de la literatura, en los que proponían métricas usadas en los gráficos de control para la aplicación de SPC en el contexto software, sin embargo se obtuvieron y analizaron 7 métricas que sirvieron como base para la validación del IVIM. Para la construcción del IVIM

se utilizaron los procedimientos y criterios de verificación extraídos por propuestas publicadas y validadas por expertos lo que redujo el trabajo a verificar la utilidad del mismo, sin embargo fue necesario adicionar tres criterios que reflejaran las características de las VSEs.

En el ciclo técnico, la metodología de desarrollo XP empleada para la construcción del prototipo software permitió tener un prototipo funcional en un corto tiempo de desarrollo, esto gracias a la experiencia adquirida en dicha metodología a través de los fundamentos recibidos el transcurso de la carrera, y a la experticia en los lenguajes de programación (Java y PHP) empleando frameworks que aceleraron el desarrollo del prototipo. Sin embargo en la aplicación de la metodología XP se utilizó una iteración para tareas específicas de diseño que no estaban contempladas en dicha metodología lo que facilitó tener un prototipo aceptable que al ser evaluado obtuvo un conjunto aceptable de sugerencias de mejora. Por otra parte, la validación del prototipo software a través del método Focus Group generó resultados satisfactorios debido a la rigurosidad del método que facilitaría la evaluación formal de herramientas software, de manera que se refina el desarrollo de las mismas.

Por último y con el fin de divulgar lo ejecutado en el presente trabajo de investigación se ha escrito un artículo titulado “Identificación de Métricas para la Aplicación de SPC en el Proceso de Construcción Software de las VSEs” cuyos autores son: German Geovanny Garzón Bravo, Pablo Felipe Miranda Caicedo y Francisco J. Pino. El artículo se envió a la revista “IEEE Latin America Transactions” (ISSN: 1548 – 0992) con el fin de divulgar el trabajo realizado de manera que sea conocido por otras personas y evaluado por entes externos para obtener realimentación del mismo. La realización de dicho artículo fue un ejercicio interesante porque permitió aclarar y sintetizar la información relevante del presente trabajo.

#### **6.4 Trabajo futuro.**

A partir de los resultados obtenidos en el presente trabajo de investigación y por cuestiones de tiempo y alcance, a continuación se describen siete aspectos relacionados con el presente trabajo que pueden ser abordados en un futuro por otros estudios de investigación:

- Realizar una validación del prototipo software en un entorno real de VSEs mediante un caso de estudio.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

- Evolucionar el prototipo a una herramienta software más completa en la que se generen los gráficos de control en tiempo real para las cuatro métricas propuestas y se gestionen los módulos teniendo en cuenta el manejo de usuarios.
- Evaluar el instrumento IVIM en la obtención de métricas software relacionadas con otros procesos del desarrollo del software.
- Estructurar una guía que permita obtener métricas para Control estadístico de procesos en VSEs.
- Definir un proceso de medición que permita levantar los datos de las métricas en las VSEs.
- Se planea incorporar las métricas propuestas en el presente trabajo de investigación para llevar a cabo SPC del proceso de construcción en VSEs siguiendo el proceso propuesto en [57].

# MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

## BIBLIOGRAFÍA

1. Garzás, J., et al., *An Organizational Maturity Model for the Spanish Software Industry based on ISO Standards 2006.*
2. Garcia, F., M. Piattini, and F.J. Pino, *Key processes to start software process improvement in small companies. 2009.*
3. Rothenberger, M.A., Y.-C. Kao, and L.N.V. Wassenhove, *Total quality in software development: An empirical study of quality drivers and benefits in Indian software projects. 2010.*
4. García, F., et al., *COMPETISOFT. Revisión Sistemática de Mejora de Procesos Software en Pequeñas y Medianas Empresas de Software. 2006.*
5. IEEE, *ISO/IEC 12207:2007(E), in Systems and software engineering — Software life cycle processes. 2007.*
6. Florac, W.A., R.E. Park, and A.D. Carleton, *Practical Software Measurement: Measuring for Process Management and Improvement, in Software Engineering Measurement and Analysis, S.E. Institute, C.M. University, and P. Pittsburgh, Editors. 1997.*
7. DeMarco, T., *Software Engineering: An Idea Whose Time Has Come and Gone? IEEE Computer Society 2009.*
8. Sargut, K.U. and O. Demirors, *Utilization of statistical process control (SPC) in emergent software organizations: pitfalls and suggestions. Springer Science + Business Media, 2006.*
9. *SWEBOK, in Guide to the Software Engineering Body of Knowledge, É.d.t.s. Pierre Bourque and U.d.Q.à.M. Robert Dupuis, Editors. 2004, Angela Burgess.*
10. Baldassarre, T., et al., *Managing Software Process Improvement (SPI) through Statistical Process Control (SPC). 2004.*
11. Barcellos, M.P., A.R. Rocha, and R.d.A. Falbo, *Evaluating the Suitability of a Measurement Repository for Statistical Process Control. 2010.*
12. Tarhan, A. and O. Demirörs, *Investigating Suitability of Software Process and Metrics for Statistical Process Control. I. Richardson, P. Runeson, and R. Messnarz (Eds.): EuroSPI 2006, LNCS 4257, pp. 88 – 99, 2006., 2006: p. 12.*
13. ISO/IEC, *15504-2:2003/Cor.1:2004(E) Information technology - Process assessment - Part 2: Performing an assessment, International Organization for Standardization, in 15504-2:2003/Cor.1:2004(E). 2004.*
14. Pino, F.J., M. Piattini, and G. Travassos, *Managing and Developing Distributed Research Projects by Means of Action-Research. 2010.*
15. McNiff, J., *Action Research.Principles and Practice. London (UK): Macmillan, 1988. 1988.*
16. P, L., C. J., and W. H., *Extremme Programming in a Software Development Laboratory. 2003.*
17. Kontio, J., L. Lehtola, and J. Bragge, *Using the Focus Group Method in Software Engineering: Obtaining Practitioner and User Experiences, in Proceedings of the 2004 International Symposium on Empirical Software Engineering ISESE'04) 0-7695-2165-7/04 ©2004 IEEE. 2004.*
18. Vijaya, G. and S. Arumugam, *Monitoring the Stability of the Processes in Defined Level Software Companies Using Control Charts with Three Sigma Limits. 2006.*
19. Komuro, M., *Experiences of Applying SPC Techniques to Software Development Processes. 2006.*

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

20. Florac W. A, C.A.D., *Measuring the Software Process*. 1999.
21. Sargut, K.U., *Application of Statistical Process Control to Software Development Processes Via Control Charts*, in *Department of Information Systems*. 2003, The Graduate School of Informatics of The Middle East Technical University. p. 124 pages.
22. Montgomery, D.C., *Control Estadístico de la Calidad*. 3ª Edición ed. 2004, Mexico: Limusa Wiley.
23. Torres, G.d.I.I., *Medidas de calidad en proceso, producto y mantenimiento, aplicadas al Control Estadístico de Procesos*. 2002.
24. Hong, G.Y., M. Xie, and P. Shanmugan, *A Statistical Method for Controlling Software Defect Detection Process*. PERGAMON. *Computers & Industrial Engineering* 37 (1999) 137-140, 1999.
25. Piattini, M., et al., *Medición y Estimación del Software: Técnicas y Métodos para Mejorar la Calidad y la Productividad*, ed. R.-M. Alfaomega. 2008.
26. Garcia, F., et al., *Una Ontología de la Medición*. 2004.
27. Fenton, N. and S. Pfleeger, *Software Metrics: A Rigorous & Practical Approach*. PWS Publishing Company, Second Edition. , 1997.
28. Pressman, R., *Ingeniería del Software, Un Enfoque Práctico*. 6ta Edición ed.
29. ISO/IEC, *ISO/IEC 29110*, in *Software Engineering - Lifecycle Profiles for Very Small Enterprises (VSE) - Part 1: Overview*. 2007.
30. ISO/IEC, *ISO/IEC 29110*, in *Software Engineering - Lifecycle Profiles for Very Small Enterprises (VSE) - Part 1: Overview*. 2007.
31. Zuluaga, K.C.P., *Estudio del comportamiento de la industria del software en Colombia ante escenarios de capacidades de innovacion y ventajas comparativas por medio de dinámica de sistemas*. 2011.
32. FEDESOFTE and MINTIC, *Informe Sectorial de la Industria de Software y Servicios Asociados de Colombia 2012*. 2012: Bogotá, D.C.
33. Silveira, P.S., K. Becker, and D.D. Ruiz, *SPDW+: a seamless approach for capturing quality metrics in software development*. Springer Science + Business Media, 2010.
34. Gómez, O., et al., *A Systematic Review Measurement in Software Engineering: State-of-the-Art in Measures*. 2006.
35. Andres Gallegos Velásquez, P.A.O.R., *Elaboración del Estándar de Aplicación de la Norma ISO/IE 12207, al Desarrollo de Aplicaciones de Software para la UTIC de la ESPE*. 2011, Escuela Nacional del Ejército, Dpto. de Ciencias de la Computación, Carrera de Ingeniería de Sistemas e Informatica.: Sangolquí - Ecuador.
36. Bolaños, L.P. and M.A. Navia, *Marco Conceptual de Atributos, Métricas y Heurísticas de Calidad de Software Para la Valoración del Producto Software Orientado a Objetos*, in *Facultad de Ingeniería Electrónica y Telecomunicaciones. Programa de Ingeniería de Sistemas - Univeridad del Cauca*. 2008.
37. Kitchenham, B., *Procedures for performing systematic reviews (Joint Technical Report)*. 2004.
38. Pino, F.J. and C.A. Ardila, *Reporte técnico sobre el estado de la práctica del proceso de desarrollo de software en la región suroccidental colombiana. Versión 1.0*. 2006.
39. Manlove, D. and S.H. Kan, *Practical Statistical Process Control for Software Metrics*. ASQ, 2007. SQP VOL. 9, NO. 4/© 2007.
40. Florac, W.A., A.D. Carleton, and J.R. Barnard, *Statistical Process Control: Analyzing a Space Shuttle Onboard Software Process*. IEEE Software, 2000: p. 10.
41. Gómez, O., et al., *Calidad de Productos de Software: Un estado del arte de la medición*. 2006.

## MÉTRICAS DEL PROCESO DE CONSTRUCCIÓN SOFTWARE PARA CONTROL ESTADÍSTICO DE PROCESOS EN PEQUEÑAS ORGANIZACIONES

42. Gómez, O., et al., *Incorporación de medidas en el modelo de procesos para la industria de software MoProSoft*. 2006.
43. Piattini, M., et al., *Análisis y Diseño de Aplicaciones Informáticas de Gestión. Una Perspectiva de Ingeniería del software*. Primera Edición ed. 2006, Madrid, España: RA-MA Editorial. 2003.
44. Gamma, E., et al., *Patrones de Diseño*. 2003.
45. Humphrey, W.S., *Introducción al PSP (Personal Software Process)*. 1997.
46. Sánchez, L.F., *Gestión de los requisitos de un proyecto software* 2006.
47. Rubio, S.E.D., *Puntos por Función. Una métrica estándar para establecer el tamaño del software*. Boletín de política informática Número 6, 2003., 2003.
48. Navarro, A., *Planificación de proyectos de software*. 2009.
49. Alba, J., *Método del Valor Ganado (Earned Value Management - EVM)*. 2009.
50. Cuadrado, J.L.L. *Métricas: Estimación de tamaño y recursos*. 2012; Available from: <http://ocw.uc3m.es/ingenieria-informatica/desarrollo-de-sistemas-de-informacion-corporativos-1/documentos/metricas-de-software>.
51. Kaur, A., B. Suri, and A. Sharma, *Software Testing Product Metrics - A Survey*. 2007.
52. (PMI), P.M.I., *Guía de los Fundamentos para la Dirección de Proyectos (Guía del PMBOK)*. 2008, Project Management Institute, Inc. 14 Campus Boulevard. Newtown Square, Pennsylvania 19073-3299 EE.UU. Teléfono: +610-356-4600, Fax: +610-356-4647, Correo electrónico: customercare@pmi.org.
53. Craig, L., *UML y Patrones. Una Introducción al Análisis y Diseño Orientado a Objetos y al Proceso Unificado*, P.E. S.A, Editor. 2003: Madrid (España).
54. Mendoza, M., C. González, and F. Pino, *Focus group como proceso en ingeniería de software: una experiencia desde la práctica*. *Revista DYNA*, Vol 80 (181), septiembre de 2013, pp: 5-14.
55. Widdows, R., T.A. Hensler, and M.H. Wyncott, *The Focus Group Interview: A Method for Assessing User's Evaluation of Library Service*. 1991.
56. Erazo, L., G. Guerrero, and F.J. Pino, *Método para la Adquisición de Software en Pequeñas Organizaciones*, in *Facultad de Ingeniería Electrónica y Telecomunicaciones, Departamento de Sistemas*. 2013, Universidad del Cauca: Popayán - Cauca.
57. Carlos Ardila, Pino F.J. Marco de Trabajo para la Gestión Cuantitativa de Procesos de Desarrollo de Software en Pequeñas Empresas. Universidad del Cauca. 2013.