

**ANEXOS: GUÍA DE TAILORING DEL PROCESO
UNIFICADO A PROYECTOS ESPECÍFICOS PARA LAS
PYMES DESARROLLADORAS DE SOFTWARE**



**Pablo Hernando Ruiz Melenje
Carlos Ernesto Maya Vallejo**

Director: Ing. Esp. Wilson Libardo Pantoja
Co-director: Dr. Julio Ariel Hurtado

**Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Programa de ingeniería de Sistemas
Grupo IDIS -Investigación y Desarrollo en Ingeniería de
Software
Popayán, 2009**

**ANEXOS: GUÍA DE TAILORING DEL PROCESO
UNIFICADO A PROYECTOS ESPECÍFICOS PARA LAS
PYMES DESARROLLADORAS DE SOFTWARE**



**Pablo Hernando Ruiz Melenje
Carlos Ernesto Maya Vallejo**

Trabajo de investigación para optar al título de Ingenieros de Sistemas

Director: Ing. Esp. Wilson Libardo Pantoja
Co-director: Dr. Julio Ariel Hurtado

**Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Programa de ingeniería de Sistemas
Grupo IDIS -Investigación y Desarrollo en Ingeniería de
Software
Popayán, 2009**

TABLA DE CONTENIDO

ANEXO A: RELACIONES ENTRE LAS ENTIDADES DE LAS DISCIPLINAS DEL PROCESO.....	5
A.1. TABLA RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DE LA DISCIPLINA DEL MODELADO DEL NEGOCIO.....	5
A.2. TABLA RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DE LA DISCIPLINA DE REQUERIMIENTOS.	7
A.3. TABLA RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DE LA DISCIPLINA DE ANÁLISIS Y DISEÑO.....	9
A.4. TABLA RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DE LA DISCIPLINA DE IMPLEMENTACIÓN	12
A.5. TABLA RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DE LA DISCIPLINA DE PRUEBAS.....	13
A.6. TABLA RELACIONES ENTRE ROLES Y ARTEFACTOS DE LA DISCIPLINA MODELADO DEL NEGOCIO	16
A.7. TABLA RELACIONES ENTRE ROLES Y ARTEFACTOS DISCIPLINA DE REQUERIMIENTOS.	16
A.8. TABLA RELACIONES ENTRE ROLES Y ARTEFACTOS DISCIPLINA DE ANÁLISIS Y DISEÑO.....	17
A.9. TABLA RELACIONES ENTRE ROLES Y ARTEFACTOS DISCIPLINA DE IMPLEMENTACIÓN.	17
A.10. TABLA RELACIONES ENTRE ROLES Y ARTEFACTOS DISCIPLINA DE PRUEBAS.	18
ANEXO B: PLANTILLAS	19
B.1. CARACTERIZACIÓN DE PROYECTOS	19
B.2. ARTEFACTOS QUE SERÁN ADAPTADOS	20
B.3. MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINA VS. VARIABLES DE CARACTERIZACIÓN.	21
B.4. MATRIZ FINAL DE CLASIFICACIÓN	22
B.5. ROLES Y ACTIVIDADES NECESARIOS PARA LA DISCIPLINA.....	22
ANEXO C: GUIAS	23
C.1. VARIABLES DE CARACTERIZACIÓN DE LOS PROYECTOS.	23
C.2. TABLA DE CLASIFICACIÓN DE ARTEFACTOS.	24
C.3. RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DE LA DISCIPLINA MODELADO DEL NEGOCIO.	24
C.4. RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DE LA DISCIPLINA DE REQUERIMIENTOS.....	24
C.5. RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DE LA DISCIPLINA DE ANÁLISIS Y DISEÑO.	25
C.6. RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DE LA DISCIPLINA DE IMPLEMENTACIÓN.	26
C.7. RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DE LA DISCIPLINA DE PRUEBAS.	26
C.8. PUBLICACIÓN DEL CONTENIDO DEL PLUGIN EN EPF [1]	27
C.9. DOCUMENTO ARTEFACTOS DE LA DISCIPLINA DEL MODELADO DEL NEGOCIO [2]	29
C.10. DOCUMENTO ARTEFACTOS DE LA DISCIPLINA DE REQUERIMIENTOS [2]	32
C.11. DOCUMENTO ARTEFACTOS DE LA DISCIPLINA DE ANÁLISIS Y DISEÑO [2]	33
C.12. DOCUMENTO ARTEFACTOS DE LA DISCIPLINA DE IMPLEMENTACIÓN [2].....	36
C.13. DOCUMENTO ARTEFACTOS DE LA DISCIPLINA DE PRUEBAS [2]	38
C.14. EJEMPLO DE CLASIFICACIÓN DE ARTEFACTOS.	42
C.15. CONSIDERACIONES AL REALIZAR LA ELIMINACIÓN DE ENTIDADES.....	43
C.16. CONSIDERACIONES EN EL ENSAMBLE DE ACTIVIDADES Y HOMOLOGACIÓN DE ARTEFACTOS.	43
C.17. CONSIDERACIONES AL REALIZAR LA ADICIÓN DE ENTIDADES.	49
C.18. EJEMPLOS OOPERACIÓN DE ADICIÓN VISTA COMO EL MECANISMO DE “VARIABILIDAD DE CONTRIBUCIÓN DE SPEM”.	53
C.19. EJEMPLOS OPERACIÓN DE REEMPLAZO VISTA COMO EL MECANISMO DE “VARIABILIDAD REEMPLAZA DE SPEM”.	56
REFERENCIAS ANEXO C	59
ANEXO D: APLICACIÓN DE GTUP – AMBIENTE ACADÉMICO Y EMPRESARIAL	59
D.1. ANEXOS GENERALES	59
D.1.1. DOCUMENTO TALLER SPEM Y EPF	59
D.1.2. DESCRIPCIÓN DE UP	89
D.2. ANEXOS AMBIENTE ACADÉMICO.....	104
D.2.1. PROYECTO MyPYMES	104
D.2.2. NIVELES DE LAS VARIABLES DE CARACTERIZACIÓN DE LOS PROYECTOS	108
D.2.3. ARTEFACTOS QUE SERÁN ADAPTADOS EN LA DISCIPLINA DE MODELADO DEL NEGOCIO	109
D.2.4. ARTEFACTOS QUE SERÁN ADAPTADOS EN LA DISCIPLINA DE IMPLEMENTACIÓN	110
D.2.5. ARTEFACTOS QUE SERÁN ADAPTADOS EN LA DISCIPLINA DE PRUEBAS	110

D.2.6.	MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN (MODELADO DEL NEGOCIO).....	112
D.2.7.	MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN (IMPLEMENTACIÓN).....	113
D.2.8.	MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN (PRUEBAS)	114
D.2.9.	MATRIZ FINAL DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINA (MODELADO DEL NEGOCIO)	117
D.2.10.	MATRIZ FINAL DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINA (IMPLEMENTACIÓN)	117
D.2.11.	MATRIZ FINAL DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINA (PRUEBAS)	118
D.2.12.	ROLES Y ACTIVIDADES NECESARIOS PARA LA DISCIPLINA (MODELADO DEL NEGOCIO)	119
D.2.13.	ROLES Y ACTIVIDADES NECESARIOS PARA LA DISCIPLINA (IMPLEMENTACIÓN)	122
D.2.14.	ROLES Y ACTIVIDADES NECESARIOS PARA LA DISCIPLINA (PRUEBAS)	124
D.3.	ANEXOS AMBIENTE EMPRESARIAL.....	127
D.3.1.	EMPRESA INPUT TECHNOLOGIES	127
D.3.2.	PROYECTO COMPROMISO	127
D.3.3.	MODULO DE ACTAS Y REUNIONES DE CALIDAD	128
D.3.4.	NIVELES DE LAS VARIABLES DE CARACTERIZACIÓN DE LOS PROYECTOS.....	130
D.3.5.	ARTEFACTOS QUE SERÁN ADAPTADOS EN LA DISCIPLINA DE REQUERIMIENTOS	131
D.3.6.	ARTEFACTOS QUE SERÁN ADAPTADOS EN LA DISCIPLINA DE ANÁLISIS Y DISEÑO.....	132
D.3.7.	MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN (REQUERIMIENTOS)	134
D.3.8.	MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN (ANÁLISIS Y DISEÑO)	135
D.3.9.	MATRIZ FINAL DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINA (REQUERIMIENTOS)	137
D.3.10.	MATRIZ FINAL DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINA (ANÁLISIS Y DISEÑO)	137
D.3.11.	ROLES Y ACTIVIDADES NECESARIOS PARA LA DISCIPLINA (REQUERIMIENTOS).....	138
D.3.12.	ROLES Y ACTIVIDADES NECESARIOS PARA LA DISCIPLINA (ANÁLISIS Y DISEÑO)	141
ANEXO E FOTOS APLICACIONES DE GTUP		143
E.1.	FOTOS AMBIENTE ACADÉMICO.....	143
E.2.	FOTOS AMBIENTE EMPRESARIAL.....	148
ANEXO F CERTIFICADO INPUT TECHNOLOGIES LTDA.		150

ANEXO A: RELACIONES ENTRE LAS ENTIDADES DE LAS DISCIPLINAS DEL PROCESO.

A.1. *Tabla relaciones entre artefactos y actividades de la disciplina del Modelado del Negocio*

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA MODELADO DEL NEGOCIO		
Artefactos	Actividades	
	Entrada a la Actividad	Salida de la Actividad
Actores del Negocio	Detallar los Caso de Uso del Negocio	Buscar los Actores y Casos de Uso del negocio
	Estructurar el Modelo de Casos Uso del negocio	Estructurar el Modelo de Casos Uso del negocio
Documento de la Arquitectura del Negocio	Buscar los Trabajadores y las Entidades del Negocio	Definir la Arquitectura del Negocio
	Definir las Necesidades de Automatización	
	Identificar los Objetivos del Negocio	
	Mantener las Reglas de Negocio	
Entidades del Negocio	Detallar las Entidades del negocio	Buscar los Trabajadores y las Entidades del Negocio
		Detallar las Entidades del negocio
Eventos del Negocio		Buscar los Trabajadores y las Entidades del Negocio
		Detallar las Entidades del negocio
Objetivos del Negocio	Detallar los Caso de Uso del Negocio	Identificar los Objetivos del Negocio
	Identificar los Objetivos del Negocio	
Glosario del Negocio	Buscar los Actores y Casos de Uso del negocio	Capturar el Vocabulario común de la Empresa
	Buscar los Trabajadores y las Entidades del Negocio	
	Definir la Arquitectura del Negocio	
	Definir las Necesidades de Automatización	
	Detallar los Caso de Uso del Negocio	
	Estructurar el Modelo de Casos Uso del negocio	
	Mantener las Reglas de Negocio	
Modelo de Análisis del negocio	Definir la Arquitectura del Negocio	Buscar los Trabajadores y las Entidades del Negocio
	Definir las Necesidades de Automatización	Definir la Arquitectura del Negocio
	Detallar las Entidades del negocio	
	Detallar los trabajadores del negocio.	
	Mantener las Reglas de Negocio	

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA MODELADO DEL NEGOCIO			
Reglas negocio	del	Detallar las Entidades del negocio	Mantener las Reglas de Negocio
		Detallar los trabajadores del negocio.	
Sistema negocio	del	Detallar las Entidades del negocio	Definir la Arquitectura del Negocio
		Detallar los trabajadores del negocio.	
Casos de Uso del Negocio		Detallar las Entidades del negocio	Buscar los Actores y Casos de Uso del negocio
		Detallar los Caso de Uso del Negocio	Detallar los Caso de Uso del Negocio
		Detallar los trabajadores del negocio	Estructurar el Modelo de Casos Uso del negocio
		Estructurar el Modelo de Casos Uso del negocio	
Modelo de Casos de Uso del Negocio		Buscar los Trabajadores y las Entidades del Negocio	Buscar los Actores y Casos de Uso del negocio
		Definir la Arquitectura del Negocio	Estructurar el Modelo de Casos Uso del negocio
		Definir las Necesidades de Automatización	
		Detallar los Caso de Uso del Negocio	
		Estructurar el Modelo de Casos Uso del negocio	
		Identificar los Objetivos del Negocio	
		Mantener las Reglas de Negocio	
Ejecución de los Casos de Uso del Negocio		Detallar las Entidades del negocio	Buscar los Trabajadores y las Entidades del Negocio
		Detallar los trabajadores del negocio.	
Visión Negocio	del	Buscar los Actores y Casos de Uso del negocio	Establecer y Ajustar los Objetivos
		Definir la Arquitectura del Negocio	
		Detallar los Caso de Uso del Negocio	
		Identificar los Objetivos del Negocio	
		Mantener las Reglas de Negocio	
		Capturar el Vocabulario común de la Empresa	
Trabajadores del Negocio		Detallar los trabajadores del negocio.	Buscar los Trabajadores y las Entidades del Negocio
			Detallar los trabajadores del negocio.
Especificación Suplementaria del Negocio		Buscar los Trabajadores y las Entidades del Negocio	Buscar los Actores y Casos de Uso del negocio
		Definir la Arquitectura del Negocio	Detallar los Caso de Uso del Negocio
		Definir las Necesidades de Automatización	
		Detallar las Entidades del negocio	
		Detallar los trabajadores del negocio.	
		Estructurar el Modelo de Casos Uso del negocio	
		Mantener las Reglas de Negocio	
Valoración de la Organización		Definir las Necesidades de Automatización	Evaluar la Organización Objetivo

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA MODELADO DEL NEGOCIO

Objetivo	Establecer y Ajustar los Objetivos
-----------------	------------------------------------

A.2. Tabla relaciones entre artefactos y actividades de la disciplina de Requerimientos.

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA DE REQUERIMIENTOS

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA DE REQUERIMIENTOS		
	Actividades	
Artefactos	Entrada a la Actividad	Salida de la Actividad
Atributos de los Requerimientos	Gestión de dependencias	Desarrollo de la Visión
	Priorizar los casos de Uso	Gestión de dependencias
Caso de Uso	Análisis de Casos de Uso	Buscar actores y casos de uso
	Capturar el vocabulario común	Detallar un caso de uso
	Defina los detalles de prueba	Estructurar el modelo de casos de uso
	Detallar los requerimientos del software	
	Detallar un caso de uso	
	Diseñar la Interface de Usuario	
	Diseño de Casos de Uso	
	Estructurar el modelo de casos de uso	
	Plan de integración del sistema	
	Plan de integración de subsistemas	
	Prototipo de Interface de Usuario	
	Revisión de Requerimientos	
Especificación de los Requerimientos del Software	Defina el enfoque de prueba	Detallar los requerimientos del software
	Diseñar la Interface de Usuario	
	Identifique las ideas de prueba	
	Revisión de Requerimientos	
Especificaciones Suplementarias	Análisis Arquitectural	Buscar actores y casos de uso
	Análisis de Casos de Uso	Detallar un caso de uso
	Defina los detalles de prueba	
	Describir la Arquitectura en Tiempo de Ejecución	
	Describir la Distribución	
	Detallar los requerimientos del software	
	Detallar un caso de uso	
	Diseño de Bases de Datos	
	Diseño de Casos de Uso	
	Diseño de Clases	
	Estructurar del Modelo de Implementación	
	Estructurar el modelo de casos de uso	
	Evaluar la viabilidad de la Prueba de	

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA DE REQUERIMIENTOS		
	Concepto Arquitectural.	
	Gestión de dependencias	
	Identificar Elementos del Diseño	
	Identificar Mecanismos de Diseño	
	Prototipo de Interface de Usuario	
	Revisar el Diseño	
	Revisar la Arquitectura	
	Revisión de Requerimientos	
Glosario	Análisis Arquitectural	Capturar el vocabulario común
	Análisis de Casos de Uso	
	Buscar actores y casos de uso	
	Detallar los requerimientos del software	
	Detallar un caso de uso	
	Estructurar el modelo de casos de uso	
	Evaluar la viabilidad de la Prueba de Concepto Arquitectural.	
	Revisión de Requerimientos	
Modelo de Casos de Uso	Análisis de Casos de Uso	Buscar actores y casos de uso
	Capturar el vocabulario común	Estructurar el modelo de casos de uso
	Definir el enfoque de prueba	
	Detallar los requerimientos del software	
	Detallar un caso de uso	
	Estructurar el modelo de casos de uso	
	Evaluar la viabilidad de la Prueba de Concepto Arquitectural.	
	Gestión de dependencias	
	Identifique las ideas de prueba	
	Priorizar los casos de Uso	
	Revisar el Diseño	
	Revisión de Requerimientos	
Paquete de Casos de Uso	Estructurar el modelo de casos de uso	Estructurar el modelo de casos de uso
	Revisión de Requerimientos	
Peticiones de los Stakeholders	Buscar actores y casos de uso	Recibir las peticiones de los stakeholders
	Capturar el vocabulario común	
	Desarrollo de la Visión	
	Detallar un caso de uso	
	Diseñar la Interface de Usuario	
	Gestión de dependencias	
	Identificar motivadores de pruebas	
Plan de Gestión de Requerimientos	Defina las necesidades de valoración y trazabilidad	Desarrollo del Plan de gestión de requerimientos
	Desarrollo del Plan de gestión de requerimientos	Gestión de dependencias

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA DE REQUERIMIENTOS		
	Detallar los requerimientos del software	
	Detallar un caso de uso	
	Gestión de dependencias	
Requerimientos Software	Diseñar la Interface de Usuario	Detallar los requerimientos del software
	Identificar motivadores de pruebas	Priorizar los casos de Uso
	Priorizar los casos de Uso	
	Revisión de Requerimientos	
Storyboard	Defina los detalles de prueba	Recibir las peticiones de los stakeholders
	Detallar un caso de uso	
	Diseñar la Interface de Usuario	
	Diseño de Clases	
	Identifique las ideas de prueba	
	Prototipo de Interface de Usuario	
Visión	Análisis Arquitectural	Desarrollo de la Visión
	Buscar actores y casos de uso	Gestión de dependencias
	Capturar el vocabulario común	
	Defina el enfoque de prueba	
	Desarrollo de la Visión	
	Detallar los requerimientos del software	
	Detallar un caso de uso	
	Diseñar la Interface de Usuario	
	Estar de acuerdo sobre la misión	
	Evaluar la viabilidad de la Prueba de Concepto Arquitectural.	
	Gestión de dependencias	
	Identificar motivadores de pruebas	
	Identifique las ideas de prueba	
	Priorizar los casos de Uso	
	Recibir las peticiones de los stakeholders	
	Revisión de Requerimientos	

A.3. Tabla relaciones entre artefactos y actividades de la disciplina de Análisis y Diseño

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA ANÁLISIS Y DISEÑO		
Artefactos	Actividades	
	Entrada a la Actividad	Salida de la Actividad
Análisis de clases	Análisis de Casos de Uso	Análisis Arquitectural
	Diseño de Bases de Datos	Análisis de Casos de Uso
	Diseño de Clases	
	Identificar Elementos del Diseño	

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA ANÁLISIS Y DISEÑO		
Artefactos	Actividades	
	Entrada a la Actividad	Salida de la Actividad
Arquitectura de referencia	Análisis Arquitectural	Evaluar la viabilidad de la Prueba de Concepto Arquitectural.
Capsula	Describir la Arquitectura en Tiempo de Ejecución	Diseño de Capsula
	Diseño de Capsula	Diseño de Subsistemas
	Diseño de Casos de Uso	Identificar Elementos del Diseño
Clases de pruebas	Implementar elementos de comprobabilidad	
Diseño de clases	Diseño de Bases de Datos	Diseño de Capsula
	Diseño de Casos de Uso	Diseño de Clases
	Diseño de Clases	Diseño de Subsistemas
	Diseño de Prueba de Elementos	Identificar Elementos del Diseño
		Identificar Mecanismos de Diseño
		Incorporar Elementos existentes de Diseño
Diseño de pruebas	Implementar prueba del desarrollador	Defina elementos de comprobabilidad
Diseño de subsistemas	Diseño de Casos de Uso	Diseño de Subsistemas
	Diseño de Subsistemas	Identificar Mecanismos de Diseño
		Incorporar Elementos existentes de Diseño
Documento de la arquitectura	Análisis Arquitectural	Análisis Arquitectural
	Análisis de Casos de Uso	Describir la Arquitectura en Tiempo de Ejecución
	Construir la Prueba de Concepto Arquitectural	Describir la Distribución
	Describir la Arquitectura en Tiempo de Ejecución	Identificar Mecanismos de Diseño
	Describir la Distribución	Incorporar Elementos existentes de Diseño
	Identificar Elementos del Diseño	
	Identificar Mecanismos de Diseño	
	Incorporar Elementos existentes de Diseño	
Revisar la Arquitectura		
Ejecución de los casos de uso	Análisis de Casos de Uso	Análisis de Casos de Uso
	Definir Elementos de Prueba	Diseño de Casos de Uso
	Diseño de Bases de Datos	
	Diseño de Casos de Uso	
	Diseño de Clases	
Evento	Diseño de Capsula	Identificar Elementos del Diseño
	Diseño de Clases	
Interface	Defina elementos de comprobabilidad	Diseño de Subsistemas
	Definir Elementos de Prueba	Identificar Elementos del Diseño
	Diseño de Casos de Uso	Incorporar Elementos existentes de Diseño
	Diseño de Subsistemas	

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA ANÁLISIS Y DISEÑO		
Artefactos	Actividades	
	Entrada a la Actividad	Salida de la Actividad
	Identifique las ideas de prueba	
Mapa de navegación	Prototipo de Interface de Usuario	Diseñar la Interface de Usuario
	Revisar el Diseño	
Modelo de datos	Identifique las ideas de prueba	Diseño de Bases de Datos
	Identifique las metas de la prueba	
	Implementar elementos de diseño	
	Revisar el Diseño	
Modelo de despliegue	Construir la Prueba de Concepto Arquitectural	Análisis Arquitectural
	Defina configuraciones de ambiente de prueba	Describir la Distribución
	Defina elementos de comprobabilidad	
	Defina el enfoque de prueba	
	Definir Elementos de Prueba	
	Describir la Distribución	
	Estructurar del Modelo de Implementación	
	Identifique las ideas de prueba	
	Identifique las metas de la prueba	
	Identifique mecanismos de comprobabilidad	
Modelo de diseño	Análisis Arquitectural	Análisis Arquitectural
	Análisis de Casos de Uso	Describir la Arquitectura en Tiempo de Ejecución
	Construir la Prueba de Concepto Arquitectural	Diseño de Casos de Uso
	Defina elementos de comprobabilidad	Diseño de Clases
	Definir Elementos de Prueba	Diseño de Subsistemas
	Describir la Arquitectura en Tiempo de Ejecución	Identificar Elementos del Diseño
	Describir la Distribución	Identificar Mecanismos de Diseño
	Diseño de Bases de Datos	Incorporar Elementos existentes de Diseño
	Diseño de Clases	
	Diseño de Subsistemas	
	Estructurar del Modelo de Implementación	
	Identificar Elementos del Diseño	
	Identificar Mecanismos de Diseño	
	Identifique las ideas de prueba	
	Implementar elementos de diseño	
	Implementar prueba del desarrollador	
	Incorporar Elementos existentes de Diseño	

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA ANÁLISIS Y DISEÑO		
Artefactos	Actividades	
	Entrada a la Actividad	Salida de la Actividad
	Revisar el Diseño	
Protocolo	Diseño de Capsula	Diseño de Capsula
		Identificar Elementos del Diseño
Prototipo de interface de usuario	Diseño de Clases	Prototipo de Interface de Usuario
	Identifique las ideas de prueba	
	Revisar el Diseño	
Prueba de concepto arquitectural	Análisis Arquitectural	Construir la Prueba de Concepto Arquitectural
	Evaluar la viabilidad de la Prueba de Concepto Arquitectural.	
Señal	Diseño de Capsula	Identificar Elementos del Diseño
	Diseño de Clases	

A.4. Tabla relaciones entre artefactos y actividades de la disciplina de Implementación

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA IMPLEMENTACIÓN		
Artefactos	Actividades	
	Entrada a la Actividad	Salida de la Actividad
Construcción	Ejecute el grupo de pruebas	Integrar el sistema
	Implemente grupos de pruebas	Integrar el subsistema
	Implemente la prueba	
Construir el plan de integración	Identifique las metas de la prueba	Plan de integración del sistema
	Integrar el sistema	Plan de integración de subsistemas
	Integrar el subsistema	
	Plan de integración del sistema	
	Plan de integración de subsistemas	
Documento de arquitectura de software	Defina configuraciones de ambiente de prueba	Estructurar del Modelo de Implementación
	Defina el enfoque de prueba	
	Identificar motivadores de pruebas	
	Identifique las ideas de prueba	
	Identifique las metas de la prueba	
	Identifique mecanismos de comprobabilidad	
	Implementar elementos de diseño	
Obtenga el compromiso de		

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA IMPLEMENTACIÓN		
	Comprobabilidad	
Elemento de comprobabilidad	Ejecute el grupo de pruebas	Implementar elementos de comprobabilidad
	Implementar elementos de comprobabilidad	
	Implementar elementos de diseño	
	Implementar prueba del desarrollador	
Elementos de implementación	Analizar el comportamiento de tiempo de ejecución	Implementar elementos de diseño
	Ejecutar pruebas del desarrollador	
	Implementar elementos de comprobabilidad	
	Implementar elementos de diseño	
	Implementar prueba del desarrollador	
	Integrar el subsistema	
	Plan de integración de subsistemas	
Modelo de Implementación	Estructurar del Modelo de Implementación	Estructurar del Modelo de Implementación
	Identifique las ideas de prueba	
	Identifique las metas de la prueba	
	Implemente grupos de pruebas	
	Plan de integración del sistema	
	Plan de integración de subsistemas	
Prueba stub	Implementar prueba del desarrollador	Implementar elementos de comprobabilidad
Pruebas del Desarrollador	Ejecutar pruebas del desarrollador	Implementar prueba del desarrollador
Subsistema de implementación	Implementar elementos de comprobabilidad	Estructurar del Modelo de Implementación
	Integrar el sistema	Implementar elementos de diseño
	Integrar el subsistema	Integrar el subsistema
	Plan de integración de subsistemas	

A.5. Tabla relaciones entre artefactos y actividades de la disciplina de Pruebas.

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA DE PRUEBAS		
Artefactos	Actividades	
	Entrada a la Actividad	Salida de la Actividad

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA DE PRUEBAS		
Arquitectura de automatización de prueba	Defina configuraciones de ambiente de prueba	Defina elementos de comprobabilidad
	Defina elementos de comprobabilidad	Identifique mecanismos de comprobabilidad
	Defina el enfoque de prueba	
	Estar de acuerdo sobre la misión	
	Estructure la implementación de pruebas	
	Identifique mecanismos de comprobabilidad	
	Implemente grupos de pruebas	
	Implemente la prueba	
	Obtenga el compromiso de Comprobabilidad	
Caso de prueba	Analice el fracaso de pruebas	Defina los detalles de prueba
	Defina elementos de comprobabilidad	
	Defina los detalles de prueba	
	Determine los resultados de prueba	
	Estructure la implementación de pruebas	
	Implemente la prueba	
Configuración del ambiente de prueba	Defina configuraciones de ambiente de prueba	Defina configuraciones de ambiente de prueba
	Defina el enfoque de prueba	Defina el enfoque de prueba
	Estructure la implementación de pruebas	
	Implemente grupos de pruebas	
	Implemente la prueba	
Datos de prueba	Analice el fracaso de pruebas	Defina los detalles de prueba
	Defina configuraciones de ambiente de prueba	
	Defina elementos de comprobabilidad	
	Defina los detalles de prueba	
	Ejecute el grupo de pruebas	
	Estructure la implementación de pruebas	
	Implemente grupos de pruebas	
	Implemente la prueba	
Especificación de interfaz de prueba	Defina elementos de comprobabilidad	Defina elementos de comprobabilidad
	Defina los detalles de prueba	Identifique mecanismos de comprobabilidad
	Estructure la implementación de pruebas	
	Identifique mecanismos de comprobabilidad	
	Obtenga el compromiso de Comprobabilidad	
Estrategia de prueba	Analice el fracaso de pruebas	Defina el enfoque de prueba
	Defina configuraciones de ambiente de prueba	Identifique las metas de la prueba
	Defina elementos de comprobabilidad	
	Defina los detalles de prueba	

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA DE PRUEBAS		
	Determine los resultados de prueba	
	Estructure la implementación de pruebas	
	Identifique las ideas de prueba	
	Identifique las metas de la prueba	
	Identifique mecanismos de comprobabilidad	
	Implemente grupos de pruebas	
	Implemente la prueba	
	Obtenga el compromiso de Comprobabilidad	
Grupo de pruebas	Analice el fracaso de pruebas	Estructure la implementación de pruebas
	Ejecute el grupo de pruebas	Implemente grupos de pruebas
	Estructure la implementación de pruebas	
	Implemente grupos de pruebas	
Lista de ideas de prueba	Defina los detalles de prueba	Identifique las ideas de prueba
	Determine los resultados de prueba	
	Identifique las ideas de prueba	
	Implemente la prueba	
Modelo de análisis de trabajo	Analice el fracaso de pruebas	Defina los detalles de prueba
	Defina configuraciones de ambiente de prueba	
	Defina elementos de comprobabilidad	
	Determine los resultados de prueba	
	Implemente la prueba	
Plan de pruebas	Calcule y mejore el esfuerzo de prueba	Calcule y mejore el esfuerzo de prueba
	Defina configuraciones de ambiente de prueba	Defina el enfoque de prueba
	Defina el enfoque de prueba	Defina las necesidades de valoración y trazabilidad
	Defina las necesidades de valoración y trazabilidad	Estar de acuerdo sobre la misión
	Ejecute el grupo de pruebas	Identificar motivadores de pruebas
	Estar de acuerdo sobre la misión	Obtenga el compromiso de Comprobabilidad
	Estructure la implementación de pruebas	
	Identificar motivadores de pruebas	
	Identifique las ideas de prueba	
Registro de pruebas	Analice el fracaso de pruebas	Ejecutar pruebas del desarrollador
	Analizar el comportamiento de tiempo de ejecución	Ejecute el grupo de pruebas
	Determine los resultados de prueba	
Resultado de pruebas	Calcule y apoye la calidad	Analizar el comportamiento de tiempo de ejecución
	Calcule y mejore el esfuerzo de prueba	Determine los resultados de prueba
Resumen de	Calcule y apoye la calidad	Calcule y apoye la calidad

RELACIONES ENTRE ARTEFACTOS Y ACTIVIDADES DISCIPLINA DE PRUEBAS		
evaluación de prueba	Calcule y mejore el esfuerzo de prueba	Calcule y mejore el esfuerzo de prueba
		Determine los resultados de prueba
Script de prueba	Analice el fracaso de pruebas	Defina los detalles de prueba
	Ejecute el grupo de pruebas	Estructure la implementación de pruebas
	Estructure la implementación de pruebas	Implemente la prueba
	Implemente grupos de pruebas	
	Implemente la prueba	

A.6. Tabla relaciones entre roles y artefactos de la disciplina Modelado del Negocio

RELACIONES ENTRE ROLES Y ARTEFACTOS DE LA DISCIPLINA MODELADO DEL NEGOCIO	
Roles	Responsable de los Artefactos
Analista Procesos del Negocio	Documento de la arquitectura del negocio
	Valoración del la organización objetivo
	Especificación suplementaria del negocio
	Visión del negocio
	Glosario del negocio
	Modelo de análisis del negocio
	Modelo de casos de uso del negocio
	Objetivo del negocio
	Reglas del negocio
Diseñador del negocio	Actores del negocio
	Casos de uso del negocio
	Ejecución de los casos de uso del negocio
	Entidades del negocio
	Eventos del negocio
	Sistema del negocio
	Trabajadores del negocio

A.7. Tabla relaciones entre roles y artefactos disciplina de Requerimientos.

RELACIONES ENTRE ROLES Y ARTEFACTOS DISCIPLINA DE REQUERIMIENTOS	
Roles	Responsable de los Artefactos
Analista de sistemas	Atributos de los requerimientos
	Visión
	Especificaciones suplementarias
	Glosario
	Modelo de casos de uso
	Peticiones de los Stakeholders
	Storyboard
	Plan de gestión de requerimientos

RELACIONES ENTRE ROLES Y ARTEFACTOS DISCIPLINA DE REQUERIMIENTOS	
Especificador de requerimientos	Caso de uso
	Especificación de los requerimientos del software
	Paquete de casos de uso
	Requerimientos software

A.8. Tabla relaciones entre roles y artefactos disciplina de Análisis y Diseño.

RELACIONES ENTRE ROLES Y ARTEFACTOS DE LA DISCIPLINA ANÁLISIS Y DISEÑO	
Roles	Responsable de los Artefactos
Diseñador	Análisis de clases
	Clases de pruebas
	Diseño de Clases
	Diseño de subsistemas
	Ejecución de los casos de uso
	Paquete de diseño
Diseñador de bases de datos	Modelo de Datos
Diseñador de Capsula	Capsula
Diseñador de interface	Mapa de navegación
	Prototipo de interface de usuario
Diseñador de Pruebas	Diseño de pruebas
Revisor Técnico	
Arquitecto Software	Modelo de despliegue
	Protocolo
	Arquitectura de referencia
	Documento de la arquitectura de software
	Señal
	Evento
	Interface
	Modelo de analisis
Modelo de diseños	

A.9. Tabla relaciones entre roles y artefactos disciplina de Implementación.

RELACIONES ENTRE ROLES Y ARTEFACTOS DE LA DISCIPLINA IMPLEMENTACIÓN	
Roles	Responsable de los Artefactos
Arquitecto software	Documento de arquitectura de software
	Modelo de implementación
Implementador	Elemento de comprobabilidad
	Elementos de implementación
	Prueba de stub

RELACIONES ENTRE ROLES Y ARTEFACTOS DE LA DISCIPLINA IMPLEMENTACIÓN	
Integrador	Pruebas del desarrollador
	Subsistema de implementación
	Construcción
	Construir el plan de integración

A.10. Tabla relaciones entre roles y artefactos disciplina de pruebas.

RELACIONES ENTRE ROLES Y ARTEFACTOS DE LA DISCIPLINA DE PRUEBAS	
Roles	Responsable de los Artefactos
Analista de pruebas	Caso de prueba
	Datos de prueba
	Lista de ideas de prueba
	Modelo de análisis de trabajo
	Resultados de prueba
Director de pruebas	Plan de pruebas
	Resumen de evaluación de pruebas
Diseñador de pruebas	Arquitectura de automatización de pruebas
	Configuración del ambiente de pruebas
	Especificación de interfaz de prueba
	Estrategia de prueba
	Grupo de pruebas
Probador	Registro de pruebas
	Script de pruebas

ANEXO B: PLANTILLAS

B.1. Caracterización de proyectos

CARACTERIZACIÓN DE PROYECTOS		
Documento: Plantilla de Caracterización de Proyectos.		
Propósito: Esta plantilla tiene el propósito de facilitar la ejecución de la caracterización del proyecto, en su contenido están los niveles de clasificación de las variables de caracterización, los cuales deben asignar de acuerdo a las particularidades del proyecto y de la organización.		
Nombre del Proyecto:		
Encargado(s):		
Fecha:		
NIVEL	1. Tamaño del equipo	
1	Muy pequeño: 1-10 personas	
2	Pequeño: 11-50 personas	
3	Medianas: 51-200 personas	
NIVEL	2. Distribución geográfica del equipo	
1	Misma Oficina	
2	Mismo edificio, diferentes oficinas	
3	Misma ciudad, misma empresa, diferente edificio	
4	Misma ciudad, diferente empresa	
5	Diferente ciudad	
3. Experiencia del equipo		
NIVEL	3.1 Experiencia en el proceso	
1	No hay ningún proyecto	
2	1 proyecto	
3	2 a 3 proyectos	
4	4 a 5 proyectos	
5	Más de 5 proyectos	
NIVEL	3.2 Experiencia en el dominio de la aplicación	
1	No hay ningún proyecto	
2	1 proyecto	
3	2 a 3 proyectos	
4	4 a 5 proyectos	
5	Más de 5 proyectos	
NIVEL	3.3 Experiencia técnica	
1	No hay ningún proyecto	
2	1 proyecto	
3	2 a 3 proyectos	
4	4 a 5 proyectos	
5	Más de 5 proyectos	
NIVEL	4. Criticidad del proyecto	
1	Bajo desempeño	
2	Pérdidas bajas (pérdidas fácilmente recuperables)	
3	Daños moderados	
4	Pérdidas grandes (pérdidas irre recuperables)	

CARACTERIZACIÓN DE PROYECTOS		
5		Riesgo de perder vidas
NIVEL	5. Tamaño del Proyecto	
1		Hasta \$ 5'000.000
2		Entre \$ 5'000.000 y \$ 15'000.000
3		Entre \$ 15'000.000 \$ 25'000.000
4		Entre \$ 25'000.000 y \$ 35'000.000
5		Más de \$ 35'000.000
NIVEL	6. Características que priorizan el proyecto	
1		Calidad = 90, tiempo = 10
2		Calidad = 70, tiempo = 30
3		Calidad = 50, tiempo = 50
4		Calidad = 30, tiempo = 70
5		Calidad = 10, tiempo = 90

B.2. Artefactos que serán adaptados

ATEFACTOS QUE SERÁN ADAPTADOS		
Objetivo: Esta plantilla permite realizar la selección de los artefactos, de la correspondiente disciplina, que pueden ser adaptados.		
Nombre del proyecto:		
Disciplina:		
Encargado:		
Fecha :		
Artefacto	Seleccionado (si/no)	Comentario

B.3. Matriz de clasificación de artefactos por disciplina vs. variables de caracterización.

MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN																																											
Documento: Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización																																											
Propósito: En esta matriz se recopilan los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina y se establecen los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina.																																											
Nombre del Proyecto:																																											
Encargado(s):																																											
Disciplina:	Variables de Caracterización																																										
	Tamaño del equipo			Distribución geográfica del equipo					Experiencia en el proceso					Experiencia en el dominio de la aplicación					Experiencia técnica					Críticidad del proyecto					Tamaño del Proyecto					Características que priorizan el proyecto									
	Niveles			Niveles					Niveles					Niveles					Niveles					Niveles					Niveles														
	1	2	3	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
ARTEFACTOS (que se pueden adaptar)	Muy pequeño: 1-10 personas	Pequeño: 11-50 personas	Medianas: 51-200 personas	Misma Oficina	Mismo edificio, diferentes oficinas	Misma ciudad, misma empresa, diferente edificio	Misma ciudad, diferente empresa	Diferente ciudad	No hay ningún proyecto	1 proyecto	2 a 3 proyectos	4 a 5 proyectos	Más de 5 proyectos	No hay ningún proyecto	1 proyecto	2 a 3 proyectos	4 a 5 proyectos	Más de 5 proyectos	No hay ningún proyecto	1 proyecto	2 a 3 proyectos	4 a 5 proyectos	Más de 5 proyectos	Bajo desempeño	Perdidas bajas (perdidas fácilmente recuperables)	Daños moderados	Pérdidas grandes (pérdidas irrecuperables)	Riesgo de perder vidas	Hasta \$ 5'000.000	Entre \$ 5'000.000 y 15'000.00	Entre \$ 15'000.000 y 25'000.000	Entre \$ 25'000.000 y 35'000.000	Más de \$ 35'000.000	Calidad = 90, tiempo = 10	Calidad = 70, tiempo = 30	Calidad = 50, tiempo = 50	Calidad = 30, tiempo = 70	Calidad = 10, tiempo = 90					
1																																											
2																																											
3																																											
4																																											

B.4. Matriz final de clasificación

MATRIZ FINAL DE CLASIFICACIÓN ARTEFACTOS		
Documento: Matriz Final de clasificación de artefactos por disciplinas		
Propósito: En esta matriz se recopilan la clasificación final de cada artefacto con respecto a las variables de caracterización y se adicionan los artefactos que fueron seleccionados como no adaptables		
Nombre del Proyecto:		
Encargado(s):		
Disciplina:		
ARTEFACTOS (que se pueden adaptar)	Clasificación final	Comentario
1		
2		
3		
4		
5		
6		

B.5. Roles y actividades necesarios para la disciplina

PLANTILLA ROLES Y ACTIVIDADES					
Documento: Plantilla Roles y actividades necesarios para las disciplinas					
Propósito: Este artefacto tiene el propósito de recopilar los roles y actividades indispensables para producir los artefactos del proceso adaptado					
Nombre del Proyecto: Modulo de actas y reuniones de calidad					
Encargado(s):					
Disciplina: Requerimientos					
ARTEFACTOS, ACTIVIDADES Y ROLES NECESARIOS			ACTIVIDADES Y ROLES A ELIMINAR		
ARTEFACTO	ACTIVIDADES		ROLES	ACTIVIDADES A ELIMINAR	ROLES A ELIMINAR
	Entrada a la Actividad	Salida de la Actividad			

ANEXO C: GUIAS

C.1. Variables de caracterización de los proyectos.

DESCRIPCIÓN DE LAS VARIABLES DE CARACTERIZACIÓN DE LOS PROYECTOS.															
Propósito:	Este documento tiene el propósito de brindar la definición de las variables para realizar la caracterización de los proyectos en la “ <i>GUÍA DE TAILORING DEL PROCESO UNIFICADO A PROYECTOS ESPECÍFICOS PARA LAS PYMES_DS</i> ”.														
Variables para realizar la caracterización	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">Tamaño del equipo:</td> <td>La variable de tamaño del equipo hace referencia a la cantidad de personas que serán utilizadas para efectuar el proyecto, esta variable puede indicar el numero de personas que recibirán asistencia, seguimiento y, si es necesario, capacitación.</td> </tr> <tr> <td>Distribución geográfica del equipo:</td> <td>Esta variable hace referencia a la ubicación del equipo de desarrollo implicado en la ejecución del proyecto.</td> </tr> <tr> <td>Experiencia del equipo</td> <td> La variable de experiencia del equipo tiene como fin indagar a cerca de previos conocimientos y experiencias que ha tenido el equipo de desarrollo. Por ser una característica compleja, la experiencia del equipo se divide en tres características: Experiencia en Proceso: Previas experiencias en UP. Experiencia técnica: hace referencia al conocimiento previo por parte de los integrantes del equipo de desarrollo con la tecnología, herramientas y paradigmas adoptados para el desarrollo del proyecto. Experiencia en aplicación: Experiencia en el dominio de la aplicación. </td> </tr> <tr> <td>Variable de criticidad del sistema:</td> <td>La variable de criticidad del proyecto tiene como fin indagar las consecuencias en el caso de que ocurra alguna falla en el sistema.</td> </tr> <tr> <td>Tamaño del Proyecto:</td> <td>El tamaño del proyecto será medido a través del costo del proyecto. El costo del proyecto es la inversión realizada por la organización en el proyecto.</td> </tr> <tr> <td>Características restrictivas</td> <td> Herramientas: Objetos utilizados para apoyar la ejecución del proceso. Normas aprobadas: Normas impuestas por la empresa o el mercado para el desarrollo del proyecto Presupuesto: Recurso monetario con el que se cuenta para ejecutar el proyecto. </td> </tr> <tr> <td>Características que priorizan el proyecto:</td> <td>Expectativa de la organización desarrolladora en relación con el proyecto; en otras palabras, es preciso analizar cuáles son las prioridades del proyecto, las cuales serán estudiadas, sobre la base de la calidad del producto y el tiempo necesario para que este llegue al cliente.</td> </tr> </table>	Tamaño del equipo:	La variable de tamaño del equipo hace referencia a la cantidad de personas que serán utilizadas para efectuar el proyecto, esta variable puede indicar el numero de personas que recibirán asistencia, seguimiento y, si es necesario, capacitación.	Distribución geográfica del equipo:	Esta variable hace referencia a la ubicación del equipo de desarrollo implicado en la ejecución del proyecto.	Experiencia del equipo	La variable de experiencia del equipo tiene como fin indagar a cerca de previos conocimientos y experiencias que ha tenido el equipo de desarrollo. Por ser una característica compleja, la experiencia del equipo se divide en tres características: Experiencia en Proceso: Previas experiencias en UP. Experiencia técnica: hace referencia al conocimiento previo por parte de los integrantes del equipo de desarrollo con la tecnología, herramientas y paradigmas adoptados para el desarrollo del proyecto. Experiencia en aplicación: Experiencia en el dominio de la aplicación.	Variable de criticidad del sistema:	La variable de criticidad del proyecto tiene como fin indagar las consecuencias en el caso de que ocurra alguna falla en el sistema.	Tamaño del Proyecto:	El tamaño del proyecto será medido a través del costo del proyecto. El costo del proyecto es la inversión realizada por la organización en el proyecto.	Características restrictivas	Herramientas: Objetos utilizados para apoyar la ejecución del proceso. Normas aprobadas: Normas impuestas por la empresa o el mercado para el desarrollo del proyecto Presupuesto: Recurso monetario con el que se cuenta para ejecutar el proyecto.	Características que priorizan el proyecto:	Expectativa de la organización desarrolladora en relación con el proyecto; en otras palabras, es preciso analizar cuáles son las prioridades del proyecto, las cuales serán estudiadas, sobre la base de la calidad del producto y el tiempo necesario para que este llegue al cliente.
	Tamaño del equipo:	La variable de tamaño del equipo hace referencia a la cantidad de personas que serán utilizadas para efectuar el proyecto, esta variable puede indicar el numero de personas que recibirán asistencia, seguimiento y, si es necesario, capacitación.													
	Distribución geográfica del equipo:	Esta variable hace referencia a la ubicación del equipo de desarrollo implicado en la ejecución del proyecto.													
	Experiencia del equipo	La variable de experiencia del equipo tiene como fin indagar a cerca de previos conocimientos y experiencias que ha tenido el equipo de desarrollo. Por ser una característica compleja, la experiencia del equipo se divide en tres características: Experiencia en Proceso: Previas experiencias en UP. Experiencia técnica: hace referencia al conocimiento previo por parte de los integrantes del equipo de desarrollo con la tecnología, herramientas y paradigmas adoptados para el desarrollo del proyecto. Experiencia en aplicación: Experiencia en el dominio de la aplicación.													
	Variable de criticidad del sistema:	La variable de criticidad del proyecto tiene como fin indagar las consecuencias en el caso de que ocurra alguna falla en el sistema.													
	Tamaño del Proyecto:	El tamaño del proyecto será medido a través del costo del proyecto. El costo del proyecto es la inversión realizada por la organización en el proyecto.													
	Características restrictivas	Herramientas: Objetos utilizados para apoyar la ejecución del proceso. Normas aprobadas: Normas impuestas por la empresa o el mercado para el desarrollo del proyecto Presupuesto: Recurso monetario con el que se cuenta para ejecutar el proyecto.													
Características que priorizan el proyecto:	Expectativa de la organización desarrolladora en relación con el proyecto; en otras palabras, es preciso analizar cuáles son las prioridades del proyecto, las cuales serán estudiadas, sobre la base de la calidad del producto y el tiempo necesario para que este llegue al cliente.														

C.2. Tabla de clasificación de artefactos.

Clasificación de Artefactos	
Clasificación	Explicación
SP	El artefacto se producirá, es un artefacto clave y puede causar problemas en el desarrollo si no es producido
NP	El artefacto no se producirá.
SR	El artefacto se producirá con restricciones (informal, sólo algunas partes del documento)
I	Es indiferente. La variable no afecta al artefacto

C.3. Relaciones entre actividades, roles y artefactos de la disciplina Modelado del Negocio.

RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DE LA DISCIPLINA MODELADO DEL NEGOCIO		
Roles	Responsable de los Artefactos	Responsable de las actividades
Analista Procesos del Negocio	Documento de la arquitectura del negocio	Buscar los actores y los casos de uso del negocio
	Valoración de la organización objetivo	Capturar el vocabulario común de la empresa
	Especificación suplementaria del negocio	Definir la arquitectura del negocio
	Visión del negocio	Establecer y ajustar los objetivos
	Glosario del negocio	Evaluar lo organización objetivo
	Modelo de análisis del negocio	Identificar los objetivos del negocio
	Modelo de casos de uso del negocio	Mantener las reglas del negocio
	Objetivo del negocio	
Diseñador del negocio	Reglas del negocio	
	Actores del negocio	Buscar los trabajadores y las entidades del negocio
	Casos de uso del negocio	Definir las necesidades de automatización
	Ejecución de los casos de uso del negocio	Detallar la entidades del negocio
	Entidades del negocio	Detallar los casos de uso del negocio
	Eventos del negocio	Detallar los trabajadores del negocio
	Sistema del negocio	
Trabajadores del negocio		

C.4. Relaciones entre actividades, roles y artefactos de la disciplina de Requerimientos.

RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DISCIPLINA DE REQUERIMIENTOS		
Roles	Responsable de los Artefactos	responsable de las actividades
Analista sistemas de	Atributos de los requerimientos	Buscar actores y casos de uso
	Visión	Capturar el vocabulario común
	Especificaciones suplementarias	Desarrollo de la visión
	Glosario	Desarrollo del plan de gestión de requerimientos
	Modelo de casos de uso	Estructurar el modelo de casos de uso

RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DISCIPLINA DE REQUERIMIENTOS		
	Peticiones de los stakeholders	Gestión de dependencias
	storyboard	Recibir las peticiones de los Stakeholders
	Plan de gestión de requerimientos	
Especificador de requerimientos	Caso de uso	Detallar los requerimientos del software
	Especificación de los requerimientos del software	Detallar casos de uso
	Paquete de casos de uso	
	Requerimientos software	

C.5. Relaciones entre actividades, roles y artefactos de la disciplina de Análisis y Diseño.

RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DE LA DISCIPLINA ANÁLISIS Y DISEÑO		
Roles	Responsable de los Artefactos	Responsable de las actividades
Diseñador	Análisis de clases	Análisis de casos de uso
	Clases de pruebas	Diseño de casos de uso
	Diseño de Clases	Diseño de clases
	Diseño de subsistemas	Diseño de pruebas de elementos
	Ejecución de los casos de uso	Diseño de subsistemas
	Paquete de diseño	
Diseñador de bases de datos	Modelo de Datos	Diseño de bases de datos
Diseñador de Capsula	Capsula	Diseño de Capsula
Diseñador de interface	Mapa de navegación	Diseñar la interface de usuario
	Prototipo de interface de usuario	Prototipo de interface de usuario
Diseñador de Pruebas	Diseño de pruebas	Definir elementos de prueba
Revisor Técnico		Revisar el diseño
		Revisar la arquitectura
		Revisión de código
Arquitecto Software	Modelo de despliegue	Evaluar la viabilidad de la Prueba de Concepto Arquitectural
	Protocolo	Construir la Prueba de Concepto Arquitectural
	Arquitectura de referencia	Análisis Arquitectural
	Documento de la arquitectura de software	Incorporar Elementos existentes de Diseño
	Señal	Identificar Mecanismos de Diseño
	Evento	Describir la Arquitectura en Tiempo de Ejecución
	Interface	Identificar Elementos del Diseño
	Modelo de analisis	Describir la Distribución
Modelo de diseños		

C.6. Relaciones entre actividades, roles y artefactos de la disciplina de Implementación.

RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DE LA DISCIPLINA IMPLEMENTACIÓN		
Roles	Responsable de los Artefactos	Responsable de las actividades
Arquitecto software	Documento de arquitectura de software	Estructurar el modelo de Implementación
	Modelo de implementación	
Implementador	Elemento de comprobabilidad	Analizar el comportamiento en tiempo de ejecución
	Elementos de implementación	Ejecutar pruebas de desarrollador
	Prueba de stub	Implementar elementos de comprobabilidad
	Pruebas del desarrollador	Implementar elementos de diseño
	Subsistema de implementación	Implementar pruebas del desarrollador
Integrador	Construcción	Integrar el sistema
	Construir el plan de integración	Integrar el subsistema
		Plan de integración de subsistemas
		Plan de integración del sistema
Revisor técnico		Revisión de código

C.7. Relaciones entre actividades, roles y artefactos de la disciplina de Pruebas.

RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DE LA DISCIPLINA DE PRUEBAS		
Roles	Responsable de los Artefactos	Responsable de las actividades
Analista pruebas	Caso de prueba	Definir las necesidades de valoración y trazabilidad
	Datos de prueba	Definir los detalles de prueba
	Lista de ideas de prueba	Determinar los resultados de prueba
	Modelo de análisis de trabajo	Identificar las ideas de la prueba
	Resultados de prueba	Identificar las metas de la prueba
Director pruebas	Plan de pruebas	Calcular y apoyar la calidad
	Resumen de evaluación de pruebas	Calcular y mejorar el esfuerzo de la prueba
		Estar de acuerdo sobre la misión
		Identificar motivadores de pruebas
		Obtener el compromiso de comprobabilidad
Diseñador pruebas	Arquitectura de automatización de pruebas	Definir configuraciones de ambiente de prueba
	Configuración del ambiente de pruebas	Definir el enfoque de prueba
	Especificación de interfaz de prueba	Definir elementos de comprobabilidad
	Estrategia de prueba	Estructurar la implementación de pruebas
	Grupo de pruebas	Identificar mecanismos de comprobabilidad

RELACIONES ENTRE ACTIVIDADES, ROLES Y ARTEFACTOS DE LA DISCIPLINA DE PRUEBAS

Probador	Registro de pruebas	Analizar el fracaso de las pruebas
	Script de pruebas	Ejecutar el grupo de pruebas
		Implementar grupos de pruebas
		Implementar las pruebas

C.8. Publicación del contenido del Plugin en EPF [1]

Antes de publicar el contenido del Plugin, es decir, de generar un sitio web basado en la configuración, es conveniente comprobar las preferencias generales de publicación en el menú Ventana->Preferencias->Método->Publicar/Examinar

Las opciones ofrecidas en dicha opción de menú son las siguientes:

- Vía de acceso predeterminada: Seleccionar el directorio donde se guardará la web generada.
- URL de información de retorno: Seleccionar la web o dirección e-mail de retorno de información. En todas las páginas publicadas se incluye una dirección cuyo propósito es que se puedan realizar comentarios, informar de errores, etc., sobre el contenido publicado.
- Incluir el contenido del método en las páginas del descriptor: Si se marca esta opción, las descripciones realizadas en el contenido de método también se incluirán en las páginas de los descriptors. Si no se marca, cada página de descriptor incluirá un enlace a la página del elemento de método al que referencia para poder consultar sus objetivos, descripciones, etc.
- Diagramas de roles: En este apartado se puede ajustar el espaciado horizontal y vertical entre los elementos mostrados en los diagramas de roles y el número de líneas que puede tener el texto de esos elementos. Estos parámetros se deben ajustar en función de la longitud de los nombres de las tareas y los productos con los que están relacionados los roles.
- Diagramas de actividad: En este apartado podemos configurar dos opciones relacionadas con la publicación de diagramas de actividad:
 - Publicar diagramas de actividad para extensiones de actividad que no se hayan modificado: Si no marcamos esta opción, las páginas de las actividades que extiendan otra actividad y no se hayan modificado, no incluirán el diagrama de actividad. La página cuenta con un enlace a la página de la actividad que extienden, donde se puede consultar el diagrama. Si se marca la casilla, el diagrama de actividad también aparecerá en la página de la actividad que extiende.
 - Publicar diagramas de detalle de actividad que no se hayan creado en el editor de procesos: Si marcamos esta opción también se publicarán los diagramas por defecto (que no hayamos creado ex profeso con el editor).

Para publicar una configuración, se selecciona en el menú principal Configuración->Publicar.

Las opciones que podremos elegir sobre la publicación son las siguientes:

- 1- Seleccionar la configuración que se va a publicar.
- 2- Seleccionar contenido de la configuración: publicar todos los procesos o sólo los que se seleccionen.
- 3- Seleccionar opciones de publicación:
 - Título y enlaces.
 - Título: Título del sitio web que aparecerá en la ventana del navegador web (común para todas las páginas).
 - Acerca del contenido: seleccionar una página HTML en la que se muestre información sobre versión de la publicación, autores, copyright, etc. El enlace a dicha página se muestra en la parte superior derecha de las páginas publicadas.
 - URL de información de retorno: página web o dirección e-mail de retorno de información. En todas las páginas publicadas se incluye una dirección cuyo propósito es que se puedan realizar comentarios, informar de errores, etc., sobre el contenido publicado. Dicho enlace se muestra en la parte superior derecha de la página publicada.
 - Glosario e índice.
 - Publicar glosario: Se publicará un glosario con todas las guías de tipo Definición de término que se hayan creado.
 - Publicar índice: Se publicará un índice similar al de los libros con las palabras marcadas con un tag especial en las descripciones de los elementos, que hayan sido editadas con el editor de texto enriquecido (HTML).
 - Aspecto y comportamiento.
 - Imagen del mensaje de cabecera: Seleccionar la imagen que aparecerá en la parte superior de las páginas publicadas.
 - Validación.
 - Comprobar hiperenlaces externos: Comprobar si existen los hiperenlaces externos.
 - Convertir hiperenlaces rotos a texto sin formato: Eliminar los hiperenlaces rotos convirtiéndolos en texto plano.
 - Diagramas.
 - Publicar diagramas de detalle de actividad que no se hayan creado manualmente: Si marcamos esta opción también se publicarán los diagramas por defecto (que no hayamos creado con el editor).
 - Publicar diagramas de actividad para extensiones de actividad que no se hayan modificado: Si no marcamos esta opción, las páginas de las actividades que extiendan otra actividad y no se hayan modificado, no incluirán el diagrama de actividad. La página cuenta con un enlace a la página de la actividad que extienden donde se puede consultar el diagrama. Si se marca la casilla, el diagrama de actividad también aparecerá en la página de la actividad que extiende.
 - Diseño.
 - Mostrar la relación de las subcarpetas en los árboles de navegación: De un nodo del árbol de navegación pueden colgar elementos de diverso tipo: guías, productos, etc. Al marcar esta opción, estos elementos se organizan en carpetas según su tipo.

- Mostrar los descriptores de tareas en los árboles de navegación: Al marcar esta opción, de los nodos del árbol de navegación que sean actividades colgarán los descriptores de tareas incluidos en dichos procesos.
 - Incluir el contenido del método en las páginas del descriptor: Si se marca esta opción, las descripciones realizadas en el contenido de método también se incluirán en las páginas de los descriptores. Si no se marca, cada página de descriptor incluirá un enlace a la página del elemento de método al que referencia para poder consultar sus objetivos, descripciones, etc.
 - Incluir los enlaces a los descriptores relacionados en las páginas de roles, tareas y productos de trabajo: Si se marca, en las páginas de los elementos de método aparecen enlaces a todos los descriptores de esos elementos en las actividades en los que se utilicen.
 - Separador predeterminado para páginas de actividad: Seleccionar la pestaña que se mostrará por defecto en la página de una actividad: Descripción, Estructura de desglose de trabajo, Asignación de equipos o Utilización del producto de trabajo.
- 4- Seleccionar formato del sitio web:
- Sitio web estático: páginas html organizadas en carpetas.
 - Aplicación web EE Java empaquetada en un archivo WAR: Aplicación Java para desplegar en un servidor que soporte Servlets.

C.9. Documento artefactos de la disciplina del Modelado del Negocio [2]

ARTEFACTOS DE LA DISCIPLINA DEL MODELADO DEL NEGOCIO	
Objetivo	En este documento se dan a conocer los objetivos de los artefactos de la disciplina del modelado del negocio, con el fin de ayudar en el entendimiento y en la selección de aquellos artefactos que la empresa considere necesarios para el desarrollo de los proyectos de la empresa.
Artefacto	Propósito del Artefacto
Actores del Negocio	El propósito de los Actores del Negocio es representar el rol jugado por alguien o algo dentro del entorno empresarial
Documento de la Arquitectura del Negocio	El propósito del Documento de la Arquitectura del Negocio es suministrar una visión general de la estructura y el propósito del negocio. Sirve como medio de comunicación entre los stakeholders y el equipo del proyecto, describe el qué y el por qué del negocio, por esta razón es una base para tomar decisiones respecto a cambios en el negocio.
Entidades del Negocio	El propósito de las Entidades del Negocio es representar una pieza de información relevante y persistente que es manipulada por actores y trabajadores de la empresa. Las Entidades del Negocio pueden ser usadas en la ejecución de los diferentes Casos de uso del negocio, brindando una base para compartir información del negocio entre los trabajadores que participan en la construcción de los casos de uso del negocio.
Eventos del Negocio	Los Eventos del Negocio representan acontecimientos importantes en las actividades de la empresa que requieren una acción inmediata, como tal, ayudan a gestionar la complejidad. Los Eventos del Negocio son activados y recibidos por los actores, trabajadores y entidades de la empresa mientras interactúan en la realización de los casos de uso del negocio. Los Eventos son usados en la activación de los casos de uso del negocio para indicar los cambios de estado del negocio y transmitir información entre los Casos de Uso del negocio.

ARTEFACTOS DE LA DISCIPLINA DEL MODELADO DEL NEGOCIO	
Objetivos del Negocio	El propósito de los Objetivos del Negocio es traducir la estrategia de negocio en pasos medibles con los cuales las operaciones del negocio puedan ser dirigidas correctamente y si es necesario mejorar. Estas medidas cuantificables permiten tener expectativas realistas en la medición objetiva del progreso en la implementación de cambios y mejoras en la empresa.
Glosario del Negocio	El Glosario del Negocio define términos importantes del proyecto, utilizados en el modelado del negocio. Este documento es importante para muchos desarrolladores, especialmente cuando tienen que comprender y utilizar términos que son específicos para el proyecto.
Modelo de Análisis del negocio	<p>El propósito Modelo de Análisis del negocio es describir cómo los casos de uso se llevan a cabo. El Modelo de Análisis del negocio describe qué ocurre entre los actores de la empresa y la empresa, y no hace ninguna presunción acerca de la estructura de la empresa o cómo los casos de uso se realizan. El Modelo de Análisis del negocio define los trabajadores internos del negocio y la información que utilizan (las entidades del Negocio), describe la estructura de la organización en unidades independientes (sistemas de negocio), y define la forma en que interactúan para lograr el comportamiento descrito en los casos de uso del negocio.</p> <p>El Modelo de Análisis del Negocio es utilizado por stakeholders y el Analista de los Procesos del Negocio para comprender cómo la empresa trabaja en la actualidad, y para analizar el efecto de los cambios en la empresa. El Analista de los Procesos del Negocio es responsable de la estructura y la integridad del modelo, mientras los Diseñadores del Negocio son responsables de detallar los elementos dentro del modelo. El modelo también es utilizado por Analistas del Sistema para deducir los requisitos del software con base en como el sistema de software se utilizará en los procesos de negocio. Los Arquitectos Software utilizan el modelo para definir una arquitectura de software que se adapte discretamente a la organización y también para identificar las clases en el análisis de software y modelos de diseño.</p>
Reglas del negocio	<p>Las Reglas del negocio son una declaración política o una condición que debe cumplirse. El propósito de este artefacto es definir una restricción específica o fija que debe ser satisfecha por la empresa. Las Reglas del negocio pueden aplicarse siempre (en cuyo caso se les llama fijas) o sólo bajo una condición específica. Si la condición se produce, la regla llega a ser válida, y por lo tanto debe cumplirse.</p> <p>Las Reglas del negocio son revisadas por stakeholders, Los analistas de los Procesos del Negocio, y los Diseñadores del Negocio para garantizar que las descripciones de las empresas se ajusten a la forma de hacer negocios. También son utilizados por los Analistas del Sistema y Arquitectos software cuando definen y diseñan software que soporta la empresa.</p>
Sistema del negocio	<p>El propósito del Sistema del negocio consiste en reducir y gestionar la compleja red de interdependencias e interacciones dentro de una empresa.</p> <p>El Sistema del negocio es utilizado por Analistas del Procesos del Negocio para determinar si las capacidades presentes en la organización son suficientes y permiten asegurar que el modelo de negocio esté anticipando el cambio, o por lo menos, sea resistente al cambio. Los Diseñadores del Negocio utilizan el sistema del Negocio para formar colecciones de los trabajadores relacionados con el negocio y las entidades del negocio, además lo utilizan para explícitamente definir y gestionar las dependencias dentro de la organización. Los Gestores de Proyectos lo utilizan para la programación del trabajo en paralelo</p>
Casos de Uso del Negocio	<p>Los Casos de Uso del Negocio definen un conjunto de instancias de casos de uso del negocio en los que cada instancia es una secuencia de acciones que realiza la empresa y que produce un resultado de valor observable para un actor particular del negocio.</p> <p>Los Casos de Uso del Negocio son procesos de negocio que abarcan las fronteras de la organización, incluyendo, posiblemente, socios y proveedores, a fin de proporcionar un valor para un stakeholder de la empresa.</p> <p>Los Casos de Uso del Negocio son útiles para quienes quieran saber qué valor ofrece la empresa y como interactúa con su entorno. Los stakeholders, analistas de los procesos del negocio, y diseñadores del negocio lo utilizan para describir los procesos del negocio y para entender el efecto de cualquier cambio propuesto (por</p>

ARTEFACTOS DE LA DISCIPLINA DEL MODELADO DEL NEGOCIO	
	<p>ejemplo, una fusión o una primera aplicación CRM) sobre la forma en que el negocio funciona. Los casos de uso del negocio son utilizados por los analistas de Sistemas y arquitectos software para comprender la forma que un sistema de software se adapta dentro de la organización. El Gestor de Pruebas también lo utiliza para proporcionar un contexto para el desarrollo de escenarios de prueba para los sistemas de software. El Gestor de Proyectos lo utiliza para planear el contenido de de las iteraciones y seguimiento del los progresos.</p>
Modelo de Casos de Uso del Negocio	<p>El Modelo de Casos de Uso del Negocio describe la dirección y la intención de la empresa. La dirección se presenta en forma de objetivos de negocio, que se derivan de la estrategia empresarial, mientras que la intención se expresa como el valor añadido y los medios de interacción con las partes interesadas de la empresa.</p> <p>El modelo de casos de uso del negocio es utilizado por stakeholders, analistas de los procesos del negocio y diseñadores del negocio para comprender y mejorar la forma en que la empresa interactúa con su entorno, los analistas de sistemas y arquitectos software lo utilizan para proporcionar el contexto para el desarrollo de software. El gestor de proyectos utiliza el modelo de casos de uso del negocio para planificar y dar seguimiento a los progresos del contenido de iteraciones durante el modelado de negocio.</p>
Ejecución de los Casos de Uso del Negocio	<p>Mientras que un Caso de Uso describe que pasos deben ser realizados con el propósito de entregar valor a un Stakeholder del negocio. La Ejecución de los Casos de Uso del Negocio describe cómo estos pasos son ejecutados dentro de la organización. Los Casos de Uso del Negocio son descritos desde una perspectiva externa, mientras la Ejecución de los Casos de Uso del Negocio es descrita desde una perspectiva interna.</p> <p>La Ejecución de los Casos de Uso del Negocio es utilizada por los stakeholders para verificar que el equipo del proyecto entienda cómo opera la empresa. Los stakeholders también los usan para identificar y priorizar mejoras en la organización. Los Analistas de Procesos del Negocio y los Diseñadores del Negocio los utilizan para definir los roles, responsabilidades, e información requerida dentro de la organización con el propósito de realizar casos de uso del negocio.</p>
Visión del Negocio	<p>El documento de la Visión del Negocio define a muy alto nivel los objetivos del esfuerzo en el modelado del negocio. Este comunica lo fundamental "el por qué y el qué " relacionado con el proyecto y es un indicador contra el cual todas las decisiones futuras deben ser validadas.</p> <p>El documento de la Visión del Negocio es leído por los directivos, autoridades de financiación, trabajadores de la empresa en el Modelado del Negocio, y por los desarrolladores en general.</p>
Trabajadores del Negocio	<p>El artefacto Trabajadores del Negocio se utiliza para representar el rol que una persona o sistema software jugará dentro de la organización. Esta abstracción nos permite identificar las posibles mejoras en los procesos del negocio y considerar el efecto de la automatización de los procesos del negocio o de la subcontratación de procesos empresariales (business process outsourcing). Los Stakeholders utilizan este artefacto para confirmar que las responsabilidades y las interacciones de los trabajadores de la empresa reflejan correctamente cómo se realiza el trabajo, o como se debe realizar.</p> <p>También es utilizado para examinar el impacto de los cambios en la organización (tales como la automatización de los procesos del negocio). Los Diseñadores del Negocio describen los detalles del flujo de trabajo (realizaciones) de cada caso de uso utilizando el artefacto de los Trabajadores del Negocio. El artefacto de lo Trabajadores del Negocio también es útil para los Analistas de Sistemas en el momento de identificar los actores del sistema, casos de uso y deducir los requerimientos del software.</p>
Especificación Suplementaria del Negocio	<p>Este documento captura algunas descripciones en el proceso, cuantifica o restringe que no puede ser asignado a un caso de uso del negocio. El contenido de este documento es por lo tanto apropiado para todos los casos de uso del negocio. Este documento es leído por los stakeholders, Analistas de los Proceso del Negocio y los Diseñadores del Negocio con el propósito de comprender los efectos de las cuantificaciones y restricciones sobre el diseño del negocio, por su parte los Analistas de Sistemas y Arquitectos software lo utilizan para comprender el efecto de estas</p>

ARTEFACTOS DE LA DISCIPLINA DEL MODELADO DEL NEGOCIO	
	cuantificaciones y restricciones sobre los requerimientos del software. Este documento no debe contener los objetivos del negocio, tampoco como son identificados, analizados y modelados por separado. Los objetivos del negocio son usados para planear y guiar las actividades del negocio en la dirección de la estrategia del negocio, mientras que las especificaciones suplementarias del negocio son utilizadas para definir los límites dentro de los cuales el negocio debe ser operado.
Valoración de la Organización Objetivo	La Valoración de la Organización Objetivo describe el estado actual de la organización donde el sistema será desplegado. La descripción es en términos de los procesos actuales, herramientas, competencias y actitudes de las personas, competidores, tendencias técnicas, problemas, y áreas de mejora.

C.10. Documento artefactos de la disciplina de Requerimientos [2]

ARTEFACTOS DISCIPLINA DE REQUERIMIENTOS	
Objetivo: En esta sección se dan a conocer los objetivos de los artefactos de la disciplina de requerimientos, con el fin de ayudar en el entendimiento y en la selección de aquellos artefactos que la empresa considere necesarios para el desarrollo de los proyectos de la empresa.	
Artefacto	Propósito del Artefacto
Peticiones de los Stakeholders	El propósito de este artefacto es capturar todas las solicitudes realizadas en el proyecto, así como la forma en que éstas se han abordado. Aunque el analista de sistemas es el responsable de este artefacto, muchas personas pueden contribuir en el desarrollo del mismo, por ejemplo: la gente de marketing, los usuarios finales, los clientes, en conclusión, cualquiera que se considere uno de los interesados en el resultado del proyecto. Esta información puede ser recogida en un documento o una herramienta automatizada.
Plan de gestión de Requerimientos	El propósito del plan de gestión de Requisitos es describir cómo en el proyecto se configurará y dirigirá los artefactos de requisitos, tipos de requisitos asociados, y sus respectivos atributos de requisitos. El plan también describe como la trazabilidad será dirigida. Este artefacto es desarrollado por el analista de sistemas durante la fase inicial y se actualiza en cada hito.
Visión	Este artefacto es desarrollado por el analista de sistemas y es Creado a principios de la fase de lanzamiento. Evolucionando durante la primera parte del ciclo de vida La visión ofrece un alto nivel, a veces contractuales, bases más detalladas de las necesidades técnicas. Captura la "esencia" de la solución prevista en forma de alto nivel de requisitos y limitaciones de diseño que le dan al lector una visión general del sistema a ser desarrollado a partir de una perspectiva de los requisitos. Proporciona entradas para la aprobación del proceso del proyecto y es, por consiguiente, estrechamente relacionado al Caso de Negocios. Este comunica lo fundamental "por qué y para qué" del proyecto y es una medida contra la cual todas las decisiones futuras deben ser validadas.
Glosario	Artefacto primario utilizado para capturar información sobre el dominio de negocio del proyecto. Desarrollado por el analista de sistemas en las fases de Elaboración y creación. Hay un glosario para el sistema que proporciona un conjunto coherente de definiciones para ayudar a evitar malentendidos. Los miembros del proyecto inicialmente utilizan el glosario para entender los términos que son específicos para el proyecto.
Modelo de Casos de Uso	El Modelo de casos de uso es un modelo de las funciones del sistema y su entorno, y sirve como un contrato entre el cliente y los desarrolladores. El modelo de casos de uso se utiliza como modelo de entrada esencial para las actividades de análisis, diseño y prueba.

ARTEFACTOS DISCIPLINA DE REQUERIMIENTOS	
Atributos de los requerimientos	El artefacto de los atributos de los requerimientos suministra un repositorio de texto, atributos y trazabilidad para todos los requerimientos. Debe ser accesible por todos en el desarrollo del proyecto.
Storyboard	Un Storyboard es una descripción lógica y conceptual de la funcionalidad del sistema para un escenario específico, incluye la interacción necesaria entre los usuarios y el sistema. Un storyboard “cuenta un historia específica”.
Especificaciones Suplementarias	El artefacto de especificación suplementaria captura los requerimientos del sistema que no son capturados fácilmente en los casos de uso del modelo de Casos de Uso. El Analista de Sistemas es el principal responsable de este artefacto, que es un complemento importante al modelo de casos de uso. Las Especificaciones suplementarias y el modelo de casos de uso deben capturar un conjunto completo de los requisitos del sistema. Los tipos de requisitos complementarios varían ampliamente entre los proyectos, por lo que la adaptación debería ser aplicada para definir las secciones aplicables a cada proyecto.
Caso de Uso	Un caso de uso define un conjunto de instancias de casos de uso, donde cada instancia es una secuencia de acciones del sistema, los cuales producen un resultado observable para un actor en particular. El propósito principal del caso de uso es capturar el comportamiento del sistema desde la perspectiva del usuario final en la consecución de uno o más objetivos deseados. El Especificador de Requisitos es el responsable directo del caso de uso.
Especificación de los requerimientos del Software	La Especificación de Requerimientos de software (SRS) captura los requerimientos de software para el sistema completo, o una parte de ese sistema. La especificación de los requerimientos del software (SRS- Software Requirements Specification) fija su atención en la colección y organización de todos los requerimientos circundantes del proyecto. Hace referencia al Plan de Gestión de Requisitos para determinar la localización y organización correcta de los requerimientos. Por ejemplo, se puede desear un SRS separado para describir los requerimientos del software completos para cada característica en una versión particular del producto. Este puede incluir varios casos de uso, para describir los requerimientos funcionales de esta característica, junto con un conjunto relevante de requerimientos detallados en la Especificación Suplementaria.
Paquete de casos de uso	Un paquete de casos de uso es una colección de casos de uso, actores, relaciones, diagramas y otros paquetes; es usado para estructurar el modelo de casos de uso y para dividir este en pequeños paquetes.
Requerimientos Software	La especificación para una condición o capacidad a la que un sistema debe ajustarse. Especificación de una condición o capacidad para las cuales un sistema debe amoldarse Los requerimientos software son documentados en un intento de especificar: <ul style="list-style-type: none"> • La capacidad necesaria del usuario para resolver un problema con el fin de alcanzar un objetivo • La capacidad que debe reunir o poseer un sistema o componente para satisfacer un contrato, estándar, especificación, u otra documentación formal impuesta.

C.11. Documento artefactos de la disciplina de Análisis y Diseño [2]

ARTEFACTOS DISCIPLINA ANÁLISIS Y DISEÑO
Objetivo: En esta sección se dan a conocer los objetivos de los artefactos de la disciplina de Análisis y diseño, con el fin de ayudar en el entendimiento y en la selección de aquellos artefactos que la empresa

considere necesarios para el desarrollo de los proyectos de la empresa.	
Artefacto	Propósito del Artefacto
Análisis de clases	El análisis de clases es utilizado para capturar el principal “Conjunto de responsabilidades” del sistema. Representa prototipos de clases del sistema, y son el “primer paso” a una abstracción mayor que el sistema debe direccionar. El análisis de clases da origen a una mayor abstracción del diseño del sistema, diseño de clases y subsistemas
Arquitectura de referencia	El artefacto de Arquitectura de Referencia es parte de una base de activos reusables de la organización. Su propósito es formar un punto de inicio para el desarrollo arquitectural. Pueden ser patrones, mecanismos arquitecturales y Frameworks anteriormente realizados. El uso de arquitecturas de referencia es una forma efectiva para direccionar requerimientos que no son funcionales, particularmente requerimientos de calidad. Las arquitecturas de referencia pueden existir o ser usadas en diferentes niveles de abstracción y desde diferentes puntos de vista.
Capsula	Una Capsula representa un patrón específico de estructuras y composición de clases las cuales han comprobado su éxito en el modelamiento y diseño de sistemas que tiene un alto grado de cooperación.
Clases de pruebas	El propósito de las Clases de Pruebas es capturar el diseño de las pruebas específicas de funcionalidad las cuales son requeridas para facilitar las pruebas. Las pruebas de funcionalidad deben incorporarse en Modelo de diseño de software de manera que puedan ser divididas en el diseño de software total
Diseño de clases	Las siguientes personas usan las clases: Los implementadores, para una especificación cuando se implementen las clases. Los diseñadores de partes del sistema, para entender como su funcionalidad debe ser utilizada. Los Diseñadores de Casos de uso, para indicarles la ejecución del los caos de uso. A aquellos que diseñan la próxima versión del sistema, para entender la funcionalidad en el modelo de diseño. A aquellos quienes prueban las clases para planear las actividades de prueba.
Diseño de pruebas	El diseño de pruebas incorpora todas las descripciones relativas a los elementos estructurales de prueba y su cooperación, hace posible las pruebas a los objetivos bien identificados. Este artefacto dirige las actividades de implementación de pruebas y su enfoque incrementa durante el diseño y la implementación.
Diseño de subsistemas	El diseño de subsistemas encapsula comportamiento, suministra interfaces explícitas y formales, y no expone su contenido interno (por convención). Provee la habilidad para encapsular completamente las interacciones de un número de clases y/o subsistemas
Documento de la arquitectura	El documento de la arquitectura del software suministra una introducción integral de la arquitectura del sistema. Sirve como medio de comunicación entre el arquitecto software y los otros miembros del proyecto respecto a decisiones significantes arquitecturalmente las cuales han sido elaboradas en el proyecto

Ejecución de los casos de uso	El propósito de la ejecución de los casos de uso es separar el interés del especificador del sistema a partir del interés de los diseñadores del sistema. La ejecución de los casos de uso proporciona una representación en el modelo de diseño el cual organiza artefactos relacionados para el caso de uso pero los cuales le pertenecen al modelo de diseño. Los artefactos relacionados consisten típicamente en la comunicación y en los diagramas de secuencia los cuales expresan el comportamiento de los casos de uso en términos de objetos de colaboración
Evento	Un Evento es utilizado para identificar y capturar información acerca de ocurrencias externas que el sistema debe conocer y a las cuales debe responder. Los Eventos pueden también ser usados para capturar información de eventos internos, tales como excepciones.
Interface	Una interface declara un conjunto de operaciones, incluyendo parámetros que son usados para servicios específicos ofrecidos por un elemento clasificador del modelo.
Mapa de navegación	Hay un mapa de navegación por sistema, el propósito de este es expresar los principales caminos de interfaz de usuario del sistema. Estos caminos se muestran a través de pantallas de lo que sería el sistema y no necesariamente se muestran todos los posibles caminos. El mapa de navegación puede ser pensado como un mapa de caminos de las interfaces de usuario.
Modelo de datos	El modelo de datos es utilizado para describir la estructura lógica y física de la información persistente gestionada por el sistema. El Modelo de Datos puede ser creado inicialmente a través de ingeniería inversa de bases de datos existentes o puede ser creada inicialmente desde un conjunto de Clases de Diseño persistentes en el Modelo de Diseño. El Modelo de Datos necesita en cualquier momento mecanismos de almacenamiento persistente basados en alguna tecnología no orientada a objetos. El modelo de datos es principalmente necesario cuando la estructura de datos persistente no puede ser derivada automáticamente y mecánicamente desde la estructura de las clases persistentes en el Modelo de Diseño.
Modelo de despliegue	El propósito del modelo de despliegue es capturar, en el sistema, la configuración de los elementos de procedimiento, y sus conexiones entre los mismos. El modelo de despliegue consiste de uno o mas nodos (elementos de procedimiento con al menos un procesador, memoria, y posiblemente otros dispositivos), dispositivos (estereotipos de nodos sin capacidad de procedimiento en un nivel de abstracción), y conectores, entre nodos, y nodos y dispositivos. El modelo de despliegue también procesa mapas sobre los elementos de procedimiento, permitiendo la distribución de comportamiento a través de los nodos.
Modelo de diseño	El modelo de diseño es una abstracción de la implementación del sistema. Es usado para concebir el documento del diseño del sistema software. Es de gran amplitud, compuesto de artefactos abarcando todas las clases de diseño, subsistemas, paquetes, cooperación, y de relación entre ellos
Paquete de diseño	Los paquetes de diseño son utilizados para agrupar elementos relacionados del Modelo de Diseño, y también son utilizados con propósitos administrativos. Diferente al artefacto de Diseño de Subsistemas un Paquete de Diseño no ofrece interfaces formales, sin embargo este puede exponer algunos de sus elementos (marcados como públicos) los cuales ofrecen algún tipo comportamiento. Los Paquetes de Diseño deben ser usados ante todo como una herramienta organizacional, para agrupar cosas relacionadas.

Protocolo	<p>Los protocolos permiten la especificación de un conjunto de puertos de Artefactos: Capsula a ser definidos y reutilizados. El protocolo define un conjunto de tipos de mensajes de entrada y salida (ejemplo operaciones y señales), y opcionalmente una cooperación (usualmente consiste de un conjunto de diagramas de secuencia) la que define el orden requerido de los mensajes y un maquina de estado que especifica el comportamiento abstracto de los participantes del protocolo.</p> <p>Un protocolo es una especificación del comportamiento deseado que puede tomar lugar sobre un conector – una especificación explícita de los acuerdos contractuales entre los participantes en el protocolo. Es solo comportamiento y no es específico en algunos elementos estructurales. Un protocolo abarca un conjunto de participantes, cada uno de los cuales juega un rol específico en el protocolo. Cada rol del protocolo es especificado por un nombre único y un conjunto de de señales que recibe, así como también un conjunto de señales que envía. Como una opción, un protocolo puede tener también una especificación de secuencias de comunicación validas.</p>
Prototipo de interface de usuario	<p>Los siguientes roles utilizan el Prototipo de Interface de Usuario: Diseñadores de interfaz de usuario, para explorar y/o validar el diseño de la interfaz. Especificadores de Requerimientos, para comprender la interfaz de cada usuario. Analistas de Sistemas, para entender como la interfaz de usuario impacta el análisis del sistema. Diseñadores, para comprender como la interfaz de usuario impacta en el diseño. Gestores, para planear las actividades desarrollo y pruebas.</p>
Prueba de concepto arquitectural	<p>El propósito de la Prueba de Concepto es determinar si existe, o es probable que exista, una solución que satisfaga los requerimientos significantes arquitecturalmente.</p>
Señal	<p>El propósito de una señal es proveer una forma de comunicación asíncrona desde un objeto o instancia de otro.</p> <p>Las señales son una especialización del Artefacto Evento que puede tener operaciones, atributos, y relaciones.</p>

C.12. Documento artefactos de la disciplina de Implementación [2]

ARTEFACTOS DISCIPLINA DE IMPLEMENTACIÓN	
Objetivo: En esta sección se dan a conocer los objetivos de los artefactos de la disciplina de implementación, con el fin de ayudar en el entendimiento y en la selección de aquellos artefactos que la empresa considere necesarios para el desarrollo de los proyectos de la empresa.	
Artefacto	Propósito del Artefacto

ARTEFACTOS DISCIPLINA DE IMPLEMENTACIÓN	
Construcción	El propósito, es entregar un subconjunto verificable de las funciones en tiempo de ejecución y las capacidades del sistema. El Proceso Unificado (UP) indica que una secuencia de versiones serán construidas durante una iteración, adicionando capacidades en cada iteración como elementos de implementación de subsistemas que son añadidos o mejorados. Las versiones pueden ser construidas en todos niveles de un sistema, abarcando los subsistemas solos o múltiples, pero en UP, estamos preocupados en particular con las versiones que son definidas en el artefacto: construcción del Plan de Integración, porque éstos son los peldaños para la terminación de la iteración. Si el tamaño de sistema o la complejidad lo justifican, la construcción del plan de integración puede ser refinada en los planes múltiples, cubriendo los subsistemas individuales. Note que las versiones informales pueden ser formuladas por un implementador por varias razones de unidades de prueba, por ejemplo - usando elementos del espacio de trabajo de desarrollo confidencial del implementador y el subsistema y los espacios de trabajo de integración del sistema. Sin embargo, cuando el término es usado aquí, las versiones son construidas por un integrador, de versiones identificadas de elementos repartidos por los implementadores dentro del subsistema o espacios de trabajo de integración del sistema, definidas en el artefacto: Construcción del plan de integración. Las construcciones son obviamente obligatorias, sin embargo, los tipos de construcciones que un proyecto produce podrán cambiar durante el ciclo de vida. En la fase de inicio, la preocupación puede ser la de producir prototipos como una manera de comprender mejor el problema o comunicarse con el cliente, en elaboración, para producir una arquitectura estable, y en la construcción, para añadir funcionalidad. En la transición, el enfoque cambia a asegurar que el software cumpla con la calidad como producto entregable.
Construir el plan de integración	El propósito de la construcción del Plan de Integración es definir el orden en el cual los componentes serán implementados, que se basa en la integración para crear el sistema, y cómo van a ser evaluados. El integrador es el responsable de éste artefacto. Las siguientes personas usan el plan de integración: Implementadores, para planear el orden en el cual implementar los elementos de diseño, y cómo y cuándo entregar a la integración del sistema Integrador, como una herramienta de planeación Diseñador de pruebas, para definir las pruebas para una iteración
Documento de arquitectura de software	El documento de la arquitectura del software suministra una introducción integral de la arquitectura del sistema. Sirve como medio de comunicación entre el arquitecto software y los otros miembros del proyecto respecto a decisiones significantes arquitecturalmente las cuales han sido elaboradas en el proyecto. El arquitecto de software establece la estructura general de cada vista arquitectónica: la descomposición de la vista, la agrupación de elementos, y las interfaces entre estos grandes grupos. Por lo tanto, en contraste con otros roles, la visualización del arquitecto de software tiene bastante amplitud y poca profundidad. El arquitecto de software es también responsable de mantener la integridad arquitectónica del sistema a través del proceso de desarrollo.
Elemento de comprobabilidad	El objetivo del elemento de Comprobabilidad es la aplicar una prueba específica de la funcionalidad que facilita la verificación, ya sea manual o automatizado. Hay varios tipos de comportamientos de pruebas específicas, dos de las cuáles son: "Stubs" para elementos de implementación que usted tiene que simular o haber decidido no incluir totalmente en un objetivo de prueba. Elementos que proveen interfaces especializadas o salidas El implementador es responsable de este artefacto. Las personas con rol de Probador utilizan los elementos de comprobabilidad para implementar y ejecutar las pruebas.
Elementos de implementación	Los elementos implementación son partes de una implementación, específicamente las unidades de menor intensidad de la composición física, la sustitución, el control de versión y la gestión de configuración. Incluyen archivos de código de software (fuente, binario o ejecutable), archivos de datos, y archivos de documentación, como archivos de ayuda en línea.

ARTEFACTOS DISCIPLINA DE IMPLEMENTACIÓN	
Modelo de Implementación	<p>El modelo implementación identifica las partes físicas de la implementación con el propósito de que pueden ser mejor comprendidas y dirigidas. El modelo implementación define las mejores unidades de la integración alrededor de la que los equipos son organizados, como las unidades que pueden ser versionadas, desplegadas, y reemplazadas por separado.</p> <p>Un modelo de implementación más detallado también podría incluir el código fuente de baja nivel y archivos entregables, y su relación con el modelo de diseño. Tal detalle es recomendado sólo si usted tiene sincronización automatizada entre el modelo y los archivos.</p> <p>Un arquitecto de software es responsable de la integridad del modelo de implementación, y asegura que: El modelo de implementación es correcto, consistente, y de lectura fácil y amena como un todo. El modelo de implementación es correcto cuando cubre todos los requisitos, y es compatible con el modelo de diseño. La arquitectura en el modelo de implementación, descrito en la vista de implementación, cumple su propósito. La vista de implementación es descrita en un artefacto distinto. Note que el arquitecto de software no es responsable de los subsistemas de implementación y los elementos de implementación. En lugar de esto, éstos están bajo las responsabilidades de los implementadores correspondientes</p>
Prueba stub	<p>El propósito de la prueba stub es simular un componente real que no está disponible para la prueba. Este artefacto es opcional y es usado en casos en los cuales los componentes reales no están disponibles o son demasiado caros para hacer pruebas. El responsable de éste artefacto es el implementador.</p>
Pruebas del Desarrollador	<p>El Propósito de las pruebas del desarrollador es proveer la implementación de un subconjunto de pruebas requeridas de una manera eficiente y efectiva. El Implementador es el principal responsable de éste artefacto. El objetivo es implementar un framework de prueba del desarrollador simple y eficiente. Para pruebas de "Un tiempo solamente", la mayor parte de la documentación deben ser evitada. La atención especial debe ser dada a las pruebas que serán usadas como las pruebas de regresión para los subsistemas o los componentes más "Imprevisibles", en términos de documentación, mantenibilidad, eficiencia, eficacia y robustez.</p>
Subsistema de implementación	<p>Las siguientes personas usarán el subsistema de implementación: Los arquitectos de software lo usan para estructurar el modelo de implementación en partes que pueden ser integradas y evaluadas por separado. Aquellos que diseñan la próxima versión del sistema lo usan para comprender la estructura del modelo de implementación. Los Implementadores de otras partes del sistema lo usan para comprender cómo su funcionalidad puede ser usada. Aquellos que evalúan el subsistema lo usan para planear actividades de prueba. El director de proyecto lo usa como base para asignar el trabajo de implementación. El subsistema de implementación es el análogo físico del paquete de diseño. El modelo de implementación y los subsistemas de implementación son definidos inicialmente en la vista de implementación, y son de importantes a la hora del desarrollo.</p>

C.13. Documento artefactos de la disciplina de Pruebas [2]

ARTEFACTOS DISCIPLINA DE PRUEBAS	
Objetivo: En esta sección se dan a conocer los objetivos de los artefactos de la disciplina de pruebas, con el fin de ayudar en el entendimiento y en la selección de aquellos artefactos que la empresa considere necesarios para el desarrollo de los proyectos de la empresa.	
Artefacto	Propósito del Artefacto

ARTEFACTOS DISCIPLINA DE PRUEBAS	
Arquitectura de automatización de prueba	<p>La arquitectura de automatización de prueba provee una visión general arquitectónica del sistema de automatización de pruebas, usando varias visualizaciones arquitectónicas diferentes para bosquejar los diferentes aspectos del sistema. Sirve como medio para razonar, dirigir y comunicar las características fundamentales del sistema software de automatización de pruebas. Suministra un enfoque para el software de automatización de pruebas, que permite que al sistema requerido sea realizado teniendo en cuenta los aspectos claves como: mantenibilidad, extensibilidad, confiabilidad, concordancia, distribución, seguridad y recuperación.</p>
Caso de Prueba	<p>El propósito del caso de prueba es especificar y comunicar las condiciones específicas que tienen que ser validadas para permitir una valoración de algunos aspectos especiales de los ítems objetivo de prueba. Un caso de prueba es diferente de una idea de prueba, en que el caso de prueba es una especificación más completamente de la prueba. Los casos de pruebas pueden ser motivados por muchas cosas pero usualmente incluirán un subconjunto de requisitos (casos de uso, características de rendimiento, etcétera.) Y los riesgos por los que se definen como importantes dentro el proyecto. Como una regla general, las especificaciones de casos de prueba son más útiles dónde la implementación de pruebas misma será demasiado complicado comprender por si sola sin el soporte de una explicación más abstracta que provee la prueba.</p> <p>El caso de prueba es usado principalmente:</p> <ul style="list-style-type: none"> Para enumerar un número adecuado de pruebas específicas para asegurar el estado completo de evaluación. Para identificar y razonar acerca de scripts y controladores de prueba requeridos, tanto manuales como automatizados. Para suministrar una idea general para la implementación de scripts y controladores de prueba y así proveer una descripción de puntos clave de observación y control, y cualquier pre y pos condición. <p>El analista de pruebas es el responsable de este artefacto.</p>
Configuración del ambiente de prueba	<p>Cada configuración de ambiente de prueba específica un ajuste apropiado y controlado en el cual se llevara a cabo la prueba requerida y las actividades de evaluación. Proveer un ambiente controlado desarrollado de una configuración conocida en la cual estas actividades, ayudas serán dirigidas para garantizar que los resultados de estos esfuerzos sean exactos, eficaces, y tengan una probabilidad más alta de ser reproducidas sistemáticamente. Un ambiente de prueba bien controlado es un aspecto importante para un análisis eficiente de fracasos y solución de fallos.</p>
Datos de prueba	<p>Los datos de prueba suministran tanto una capa de falta de objetivos como un punto central de modificación para las características únicas de una prueba. Cuando se dirige por separado los aspectos de procedimiento de la prueba, esto permite que a las características únicas de la prueba sean modificadas por separado.</p>

ARTEFACTOS DISCIPLINA DE PRUEBAS	
Especificación de interfaz de prueba	<p>La especificación de interfaz de prueba provee un método para documentar los requerimientos especiales del esfuerzo de prueba que pondrán restricciones o requisitos adicionales en el diseño del software. Donde los aspectos del sistema que normalmente no tienen visibilidad deben ser observados, o donde el control del software es requerido de una manera normalmente no disponible a través de la interfaz estándar, esto puede necesitar que interfaces de prueba especializadas sean desarrolladas.</p>
Estrategia de prueba	<p>El propósito principal de la estrategia de prueba es para:</p> <p>Llevar la estrategia a los stakeholders externos para obtener su aceptación en el enfoque.</p> <p>Llevar la estrategia a los miembros internos del equipo de prueba para permitir un esfuerzo de equipo coordinado.</p> <p>Una estrategia de prueba necesita ser gestionada con convencimiento y otros grupos de presión que el enfoque está sano y alcanzable, y también tiene que ser apropiado tanto en relación con el producto de software de ser evaluado y las destreza del equipo de prueba.</p>
Grupo de pruebas	<p>El grupo de pruebas provee unos medios de gestiona de la complejidad de la implementación de pruebas. Muchos sistemas de esfuerzos de prueba fallan porque el equipo pierde detalles de las pruebas, y posteriormente pierde el control del esfuerzo de prueba. Similar a los paquetes de UML, los grupos de pruebas proveen una jerarquía de encapsulamiento de contenedores para ayudar a llevar a cabo la implementación de pruebas. Proveen unos medios para gestionar los aspectos estratégicos del esfuerzo de prueba coleccionando pruebas en grupos relacionados, que pueden ser razonados, planeados, gestionados, y evaluados de manera significativa.</p>
Lista de ideas de prueba	<p>La lista de ideas de prueba provee una capa de abstracción entre el plan de prueba conceptual y el caso de prueba más detallado o el script de prueba concreto. Es usado para captar las ideas iniciales para formar a menudo pruebas potenciales o ideas parciales dentro de las cuales las pruebas pueden ser razonadas. Este artefacto es particularmente útil en el inicio del ciclo de desarrollo o cuando los artefactos de proyecto secundarios no están disponibles o son incompletos.</p> <p>El analista de pruebas es el responsable de este artefacto.</p>
Modelo análisis de trabajo	<p>El modelo de análisis de trabajo intenta definir las condiciones de carga bajo las que los ítems objetivo de prueba deben funcionar dentro de su ambiente de configuración objetivo. El objetivo principal es definir un trabajo representativo realista que permita que los riesgos de rendimiento sean calculados con exactitud. Determinar típicamente por análisis anticipado o características de actores existentes, las estadísticas de los negocios del usuario final (casos de uso), etcétera.</p>

ARTEFACTOS DISCIPLINA DE PRUEBAS	
Plan de pruebas	<p>El propósito del plan de prueba es dar una idea general y comunicar el propósito de las pruebas de esfuerzo para un determinado cronograma. En primer lugar, la planificación con otros artefactos, el objetivo principal es obtener la aceptación y aprobación de las partes interesadas en la prueba de esfuerzo. Como tal, el plan de pruebas debe evitar detalles que no se entienden, o se consideran irrelevantes por las partes interesadas en la prueba de esfuerzo.</p> <p>En segundo lugar, el Plan de Pruebas constituye el framework dentro del cual el equipo de realización de las pruebas trabaja para el cronograma determinado. Dirige, orienta y limita la prueba de esfuerzo, centra el trabajo en los entregables útiles y necesarios.</p> <p>En las culturas o ámbitos en los que este artefacto no es reconocido como un artefacto formal, éste sigue siendo importante para abordar los diferentes aspectos representada por el plan de pruebas, como parte del esfuerzo de planificación, y toma de decisiones apropiadas acerca de lo que las pruebas llevarán a cabo y cómo la prueba se llevará a cabo.</p>
Registro de Pruebas	<p>El registro de prueba provee un registro detallado, típicamente registros que sirven como verificación de que un conjunto de pruebas fue ejecutado, y provee la información relacionada con el éxito de esas pruebas. El enfoque está típicamente sobre la previsión de un rastro de auditoría exacta, permitiendo que diagnósticos post-ejecución de fracasos sean abordados. Estos datos sin procesar serán analizados posteriormente para ayudar a determinar los resultados de algunos aspectos del esfuerzo de prueba.</p>
Resultados de prueba	<p>Los resultados de prueba son usados para registrar las conclusiones detalladas del esfuerzo de prueba y posteriormente calcular las diferentes medidas clave de la prueba.</p>
Resumen de evaluación de prueba	<p>El resumen de evaluación de prueba colecciona, organiza, y presenta los resultados de prueba y las medidas claves de la prueba que permitan una evaluación y valoración de calidad objetiva. El resumen de evaluación de prueba también presenta una evaluación interina del equipo de prueba, indicando sus valoraciones del software contra la misión de evaluación y las recomendaciones correspondientes que los próximos esfuerzos de prueba requerían.</p>
Script de prueba	<p>El propósito del script de prueba es proveer la implementación de un subconjunto de pruebas requeridas de una manera eficiente y eficaz.</p>

C.14. Ejemplo de clasificación de artefactos.

Para establecer la relación entre los artefactos del proceso y las variables de caracterización, en la guía de tailoring del Proceso Unificado se utiliza la clasificación brindada por la Tabla 1. **Clasificación de Artefactos.1**; además, se utiliza la Tabla 2 para representar la relación entre artefactos y variables de caracterización, en la cual se muestra la relación, por medio de la clasificación de los artefactos, entre los artefactos de una determinada disciplina y las variables de caracterización.

CLASIFICACIÓN DE ARTEFACTOS	
Clasificación	Explicación
SP	El artefacto se producirá, es un artefacto clave y puede causar problemas en el desarrollo si no es producido
NP	El artefacto no se producirá.
SR	El artefacto se producirá con restricciones (informal, sólo algunas partes del documento)
I	Es indiferente. La variable no afecta al artefacto

Tabla 1. Clasificación de Artefactos.

Tabla 2. Ejemplo de **Matriz de clasificación de artefactos por disciplinas vs. Variables de caracterización.2**, se plasma un ejemplo de la clasificación de artefactos vs. variables de caracterización, en ésta se puede verificar que la relación de las variables de caracterización (caracterización 1 y caracterización 2), se realiza a través de los niveles de caracterización (nivel de caracterización 1 y nivel de caracterización 2), y de la clasificación (SP, NP, SR, I) de cada uno de los artefactos (artefacto 1, artefacto 2 y artefacto 3) de una determinada disciplina (disciplina ejemplo). Debido a que el mismo artefacto de una disciplina puede tener diferentes clasificaciones en cada una de las variables de caracterización, ver la clasificación del artefacto 1 en la Tabla

Tabla 2. Ejemplo de Matriz de clasificación de artefactos por disciplinas vs. Variables de caracterización.2, surge la necesidad de unificar las clasificaciones de cada artefacto por medio de la comparación de las clasificaciones de las variables.

MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN		
Disciplina: Disciplina ejemplo	Variable de Caracterización 1	Variable de Caracterización 2
Artefactos	Nivel de Caracterización1	Nivel de Caracterización 2
Artefacto 1	SP	NP
Artefacto 2	SR	SR
Artefacto 3	I	SP

Tabla 2. Ejemplo de Matriz de clasificación de artefactos por disciplinas vs. Variables de caracterización.

C.15. Consideraciones al realizar la eliminación de entidades.

Debido al enfoque de la guía de tailoring centrado en la elección de los artefactos que harán parte de las disciplinas, la operación de eliminación tiene el mismo enfoque, puesto que primero se considerará la eliminación de artefactos, para posteriormente verificar, si es posible, la eliminación de otras entidades del proceso como tareas y roles.

Eliminación de artefactos.

Más que eliminar el artefacto, es necesario también identificar las relaciones que tiene con otras entidades del proceso, para saber cómo la eliminación del artefacto afectará a otras entidades del proceso; entonces, en la eliminación de un artefacto es preciso tener en cuenta las siguientes relaciones:

- 1) Relaciones con tareas
- 2) Relaciones con roles

Las relaciones entre artefactos y tareas se pueden observar en los anexos A donde es fácil identificar las relaciones para poder efectuar la eliminación, conociendo las tareas que serán modificadas, entonces, es necesario eliminar el artefacto de las tareas afectadas ya sea de entrada o salida, y de esta forma eliminar las relaciones entre estas dos entidades del proceso.

La eliminación de las relaciones entre artefactos y roles es similar al caso anterior, en los anexos A se plasman las relaciones que tienen los artefactos con los roles en las disciplinas. Entonces, conociendo las relaciones entre estas dos entidades se puede realizar la eliminación del artefacto sin que se generen ambigüedades en el proceso, ya que los roles no tendrán ningún tipo de relación con los artefactos eliminados, lo que puede significar una disminución en las responsabilidades del rol debido a que tendrá menos artefactos bajo su responsabilidad.

Es preciso enfatizar, que todas las relaciones del artefacto a eliminar deben desaparecer, puesto que, si existen entidades del proceso que tienen relaciones con los artefactos que no serán producidos, podría generar entorpecimiento en la realización del proceso y causar confusión entre los ejecutores del mismo.

Eliminación de tareas y roles

Es necesario aclarar que la eliminación de tareas y roles dependen de la eliminación de los artefactos, pues la supresión de estas dos entidades es una consecuencia de efectuar la eliminación de artefactos debido a que, para eliminar tareas y roles es necesario que por su parte las tareas no produzcan ningún tipo de artefacto requerido por otras tareas y que los roles no tengan responsabilidad por artefactos o tareas, es decir, que tanto tareas como roles no posean relaciones con otras entidades del proceso.

C.16. Consideraciones en el ensamble de actividades y homologación de artefactos.

Es importante mencionar que el enfoque del ensamble de actividades descrito a continuación, se concentra en la actividad previa y posterior de la actividad a reemplazar.

Antes de realizar el ensamble de las actividades, es necesario verificar la compatibilidad entre dichas actividades, para luego centrarse solo en el ensamble de éstas y en la homologación de los artefactos. Para el ensamble de actividades del proceso se utilizan como base algunas

incidencias de los mecanismos de ensamble de componentes de procesos¹ definidos en [SPEM] debido a la similitud entre estos dos elementos, ver Tabla 1.

ACTIVIDAD VS. COMPONENTE		
	Actividad	Componente
Descripción	Una actividad representa una unidad de trabajo general en un proceso	Un componente de proceso contiene un proceso representado por una actividad
Entradas y Salidas	Cada actividad tiene artefactos de entrada y salida, y un objetivo claro expresado comúnmente en términos de crear o actualizar artefactos.	Un componente define un conjunto de puertos de productos de trabajo requeridos o suministrados, los cuales definen los tipos de productos de trabajo usados, producidos, o modificados por el proceso.
Estructura	Una actividad puede tener una estructura interna formada por agregación de elementos, que pueden ser de varios tipos, no solo de trabajo: actividades más simples (anidamiento) y elementos de método.	Un componente puede asumir una estructura interna compuesta por componentes de procesos más simples.

Tabla 1. Actividad vs. Componente.

Para realizar el ensamble de actividades primero se debe encapsular conceptualmente el método o el proceso interno de las actividades, es decir, no importarán los pasos o técnicas de cómo se realizan las actividades involucradas en el reemplazo. De esta forma se utiliza el mecanismo de ensamble basado en el principio de caja negra, el cual nos permite centrarnos en el ensamble de las actividades por medio de la homologación de entradas y salidas.

La Figura 1 muestra una actividad del proceso llamada desarrollo (Development) que define dos productos de entrada 'Modelo de Análisis' y 'Modelo de Diseño' así como una salida 'Aplicación'.



Figura 1. Representación gráfica de una actividad

El ensamble de las actividades del proceso inicia simplemente por conectar las actividades, a través de sus entradas y salidas. Esto parece trivial, pero hay varios problemas que se deben resolver.

- 1) El número y el tipo de productos de trabajo de entrada y salida podría no corresponder entre las actividades.

¹ COMPONENTES DE PROCESOS son paquetes especiales de procesos que por lo general son utilizados cuando la empresa realiza sub-contratación de cierta parte del proceso de desarrollo.

- 2) Los nombres y la constitución de los artefactos podría variar entre actividades.
- 3) El nombre y la definición de los roles podría variar entre las actividades.

No existe la garantía que las actividades correspondan, en particular si las actividades son complejas y originadas desde fuentes que tienen diferentes métodos de desarrollo, la concordancia puede ser imposible; sin embargo, para ensamblar y mapear diferentes actividades del proceso, se utilizará un mecanismo de ensamble similar a la notación de puertos UML 2.0² con el fin de ajustar las variaciones que puedan existir entre las actividades del proceso. En los componentes de proceso se definen puertos de entrada y salida, los cuales corresponden a los artefactos de entrada y salida de las actividades en SPEM.

Con el fin de expresar las relaciones que se necesitan establecer entre las actividades del proceso, se utilizarán líneas como conectores entre actividades; es importante recordar que una actividad puede tener entradas ó productos de trabajo opcionales, que no necesitan ser ensamblados, pero si es necesario ensamblar las entradas ó producto de trabajo obligatorios.

El esfuerzo necesario para realizar el ensamble de las actividades de la Figura 2 puede ser insignificante, puesto que la actividad nativa que será ensamblada tiene correspondencia en nombres, entradas y salidas con la actividad previa y posterior, de esta forma efectuar el ensamble de las actividades es trivial, pues solo bastaría con efectuar el ensamble de entradas y salidas entre las actividades como se muestra en la Figura 3, pero pueden existir problemas que pueden interferir en el montaje de las actividades, como la incompatibilidad de nombres y correspondencias en las entradas o productos de trabajo de dichas actividades. En esta situación, un mayor esfuerzo es necesario para ensamblar los elementos existentes en la configuración de las actividades.

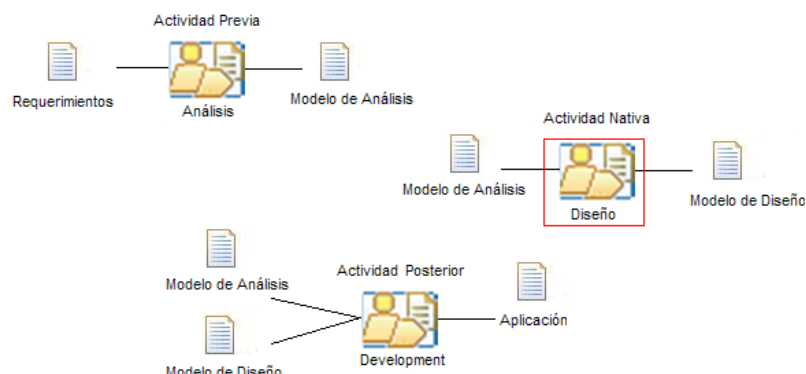


Figura 2. Actividades a ser ensambladas

² NOTACIÓN DE PUERTOS UML 2.0 se refiere a la representación de entradas y salidas utilizada en el ensamble de componentes de proceso.

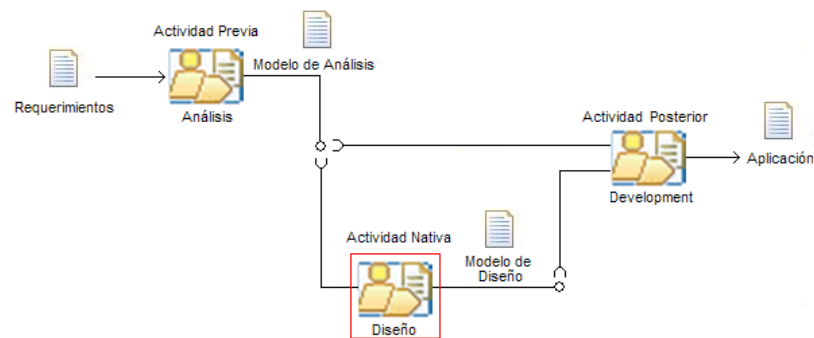


Figura 3. Ensamble de las Actividades Previa, Posterior y Nativa.

Para realizar el ensamblaje de las actividades previa, posterior y nativa, es reiterativo mencionar que se debe encapsular el método ó el proceso interno de las actividades, para de esta manera utilizar el mecanismo de ensamblaje basado en el principio de caja negra, el cual nos permite concentrarnos en las entradas ó productos de trabajo de las actividades.

Cuando se realiza el ensamblaje de actividades podemos enfrentar varias situaciones y problemas (que por motivos de alcance de este proyecto no serán abordados), pero si se explicaran y ofrecerán soluciones a algunos casos básicos que pueden servir de ayuda en momento de reemplazar actividades en el proceso. Algunos casos típicos que pueden pasar en el ensamblaje de actividades son descriptos a continuación:

- La actividad reemplazante³ o nativa requiere de una entrada no suministrada por la actividad previa del proceso.
- La actividad reemplazante o nativa no requiere de una entrada que es suministrada por la actividad previa.
- La actividad reemplazante o nativa no suministra una salida para la actividad posterior del proceso.
- La actividad reemplazante o nativa suministra una salida que no es requerida por la actividad posterior.

En la siguiente sección se ilustran los casos mencionados anteriormente y se dan posibles soluciones a cada uno de ellos. Se muestra una gráfica por cada caso, donde 'M' denota la actividad del proceso (actividad a ser reemplazada), y 'MX' denota la actividad nativa o reemplazante (actividad que sustituirá a otra en el proceso). Es importante destacar que tanto la actividad 'M' y 'MX' tienen sus artefactos de entrada, provenientes de la actividad previa, y de salida, suministrados por estas actividades a las actividades posteriores.

- **Caso 1: La actividad reemplazante o nativa, ver Figura 4, requiere de una entrada no suministrada por la actividad previa del proceso.**

³ ACTIVIDAD REEMPLAZANTE. Se define como una actividad propia del proceso de la empresa la cual sustituirá a una actividad del patrón de proceso.

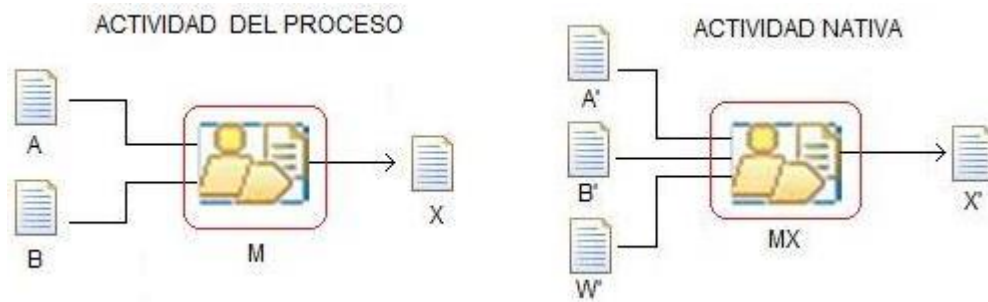


Figura 4. Actividad reemplazante o nativa, requiere de una entrada no suministrada por la actividad previa del proceso.

Una posible solución para este caso puede ser ensamblar una nueva actividad que suministre el elemento perdido ó que la actividad previa se adecue para desarrollar la entrada para la actividad nativa.

- **Caso 2: La actividad reemplazante o nativa no requiere de una entrada que es suministrada por la actividad previa. Ver Figura 5**

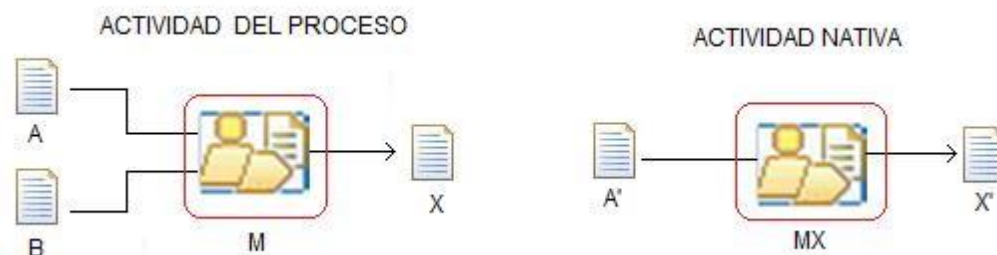


Figura 5 .La actividad reemplazante o nativa no requiere de una entrada que es suministrada por la actividad previa.

En este caso pueden surgir dos acontecimientos, que la actividad previa tenga una sola salida y por lo tanto quede huérfana, o que la actividad tenga otras relaciones con la actividad nativa y solo se pierda una relación.

En el primer acontecimiento es posible borrar la actividad previa, pues esta va a carecer de relaciones con la actividad nativa (y hacia atrás todas las demás actividades huérfanas). Esto queda bajo responsabilidad de la persona encargada de hacer la adaptación, y en el segundo acontecimiento se puede evaluar la realización de dicho artefacto por el cual están relacionadas las actividades, ya que carecería de sentido realizar un artefacto que no se utilice en el proceso.

- **Caso 3: La actividad reemplazante o nativa no suministra una salida para la actividad posterior del proceso. Ver Figura 6**

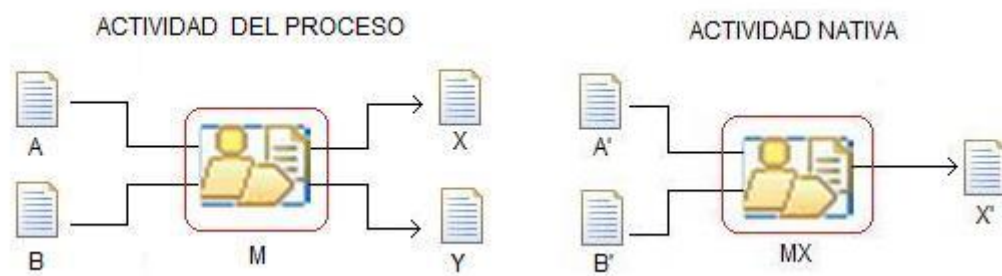


Figura 6 . La actividad reemplazante o nativa no suministra una salida para la actividad posterior del proceso.

También para este caso, una posible solución puede ser ensamblar una nueva actividad que suministre el elemento perdido ó que la actividad previa, en este caso la nativa, se adecue para desarrollar la entrada para la actividad posterior.

- **Caso 4: La actividad reemplazante o nativa suministra una salida que no es requerida por la actividad posterior, ver Figura 7**

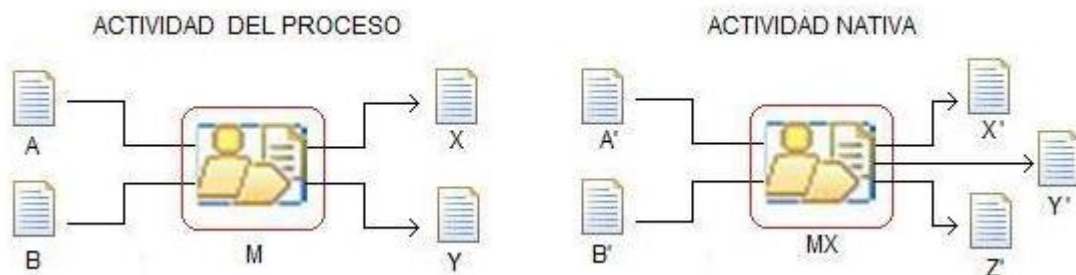


Figura 7 .La actividad reemplazante o nativa suministra una salida que no es requerida por la actividad posterior

En este caso se puede borrar el producto de salida huérfano, previa evaluación de lo que implicaría la no ejecución de este artefacto, puesto que carecería de sentido realizar un artefacto que no se utilice en el proceso.

Las soluciones planteadas para los casos básicos pueden utilizarse o pueden tenerse en cuenta en casos de reemplazo de actividades más complejos, por ejemplo cuando hay una actividad nativa que tiene un producto de entrada y suministra una salida, mientras que la actividad del proceso tiene dos productos de trabajo de entrada y salida. En particular para dar solución a este ejemplo se pueden utilizar paso a paso los casos básicos 2 y 3, de esta manera se pretende mostrar que los casos base pueden servir de solución a casos de reemplazo más complejos.

Homologación de Artefactos

En la homologación de artefactos entre la actividad del proceso y la actividad nativa pueden ocurrir los siguientes casos:

- El nombre de los artefactos puede variar entre la actividad del proceso y la nativa.

- Los artefactos pueden integrarse o separarse para permitir el ensamble entre actividades.
- La estructura de los artefactos es distinta.

En la siguiente sección se dan posibles soluciones a los casos mencionados anteriormente.

- **Caso 1: *El nombre de los artefactos puede variar entre la actividad del proceso y la nativa.***

Inicialmente hay que evaluar si corresponde al mismo artefacto, puesto que puede variar el nombre pero su propósito sea el mismo, lo cual sería el caso más trivial donde la homologación sería directa siempre y cuando se establezca una relación entre los nombres, que puede ser concatenarlos (para evitar ambigüedades en el proceso). Pero también se puede dar el caso en el que los nombres correspondan pero su propósito sea diferente, entonces para este caso es necesario llegar a un consenso entre el artefacto nativo y el del proceso, es decir, tratar de unificar los artefactos para que el cambio no sea tan traumático dentro del proceso de la organización.

- **Caso 2: *Los artefactos pueden integrarse o separarse para permitir el ensamble entre actividades.***

Puesto que varios artefactos nativos pueden conformar un solo artefacto del proceso o viceversa, se puede tratar de integrar o dividir los artefactos en función de cumplir el propósito del artefacto a ser reemplazado.

- **Caso 3: *La estructura de los artefactos es distinta.***

Cuando la estructura del artefacto es distinta, es importante enfocarse en el objetivo del artefacto puesto que, así tengan estructuras diferentes, pueden cumplir con el mismo propósito, lo que facilitaría la homologación. Dado el caso en el que las estructuras sean distintas, se puede recurrir a la misma solución del caso 1 de unificar los artefactos.

C.17. Consideraciones al realizar la adición de entidades.

La adición de entidades del proceso es una actividad que brinda la opción de añadir entidades del estándar y adición de entidades definidas por el usuario. Cuando se realiza la adición de entidades en el proceso de desarrollo, se deben tener en cuenta que las entidades del proceso pueden ser roles, tareas y artefactos, en cada caso es importante considerar algunos aspectos que pueden ser cruciales en el momento de realizar dicha adición.

Antes de mencionar los aspectos que se pueden considerar en la adición de las entidades del proceso es imprescindible recordar la relación intrínseca que tienen estas entidades. Un artefacto no puede ser adicionado por sí solo, siempre debe haber una tarea que lo produzca o lo utilice y una tarea no puede ser adicionada sin que produzca o modifique algún tipo de artefacto. Por su parte un rol no puede ser adicionado sin que sea responsable de alguna tarea o artefacto.

Algunos aspectos que pueden considerarse en la adición de entidades del proceso son descritos a continuación.

Adición de roles.

Un rol define el comportamiento y responsabilidades de un individuo, o de un grupo de individuos trabajando juntos como un equipo. Las responsabilidades de un rol son tanto el llevar a cabo un conjunto de actividades como el ser el responsable de un conjunto de artefactos. [3]

Es imprescindible que en la adición de los roles se cumpla con los aspectos mencionados en la definición de este elemento del proceso, entonces se debe definir como mínimo lo siguiente:

- 1) Especificar las habilidades necesarias.
- 2) Especificar las tareas que realiza.
- 3) Especificar los artefactos que son responsabilidad del rol.

En el primer caso, es necesario especificar las destrezas que debe tener la persona o el grupo de personas que van a desempeñar este rol, puesto que para la ejecución exitosa del proceso se debe contar con el personal idóneo que pueda cumplir con las expectativas plasmadas en las habilidades del rol.

En la especificación de las tareas que realiza y artefactos que son responsabilidad del rol, es donde se asignan los compromisos y relaciones del rol, se deben especificar las tareas y los artefactos que están bajo su responsabilidad. Es importante señalar que la descripción de las responsabilidades del rol es necesaria, puesto que carecería de valor adicionar un nuevo rol si éste no tiene ningún tipo de relación (responsable o ejecutor) con algunas de las entidades del proceso, es decir el rol debe adquirir responsabilidades con el proceso de desarrollo, ver Figura 8. Elementos necesarios en la definición del Rol, en la parte de relaciones.

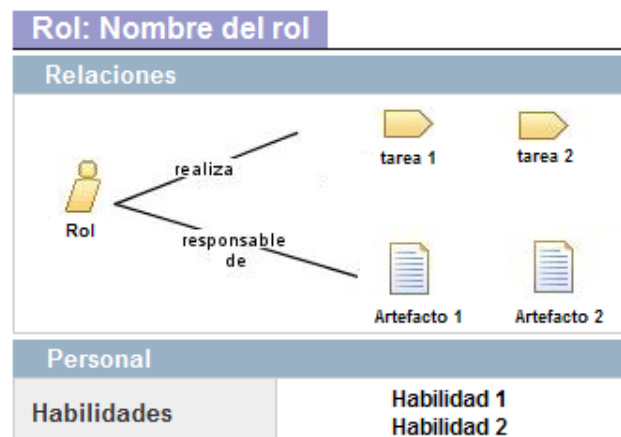


Figura 8. Elementos necesarios en la definición del Rol

Adición de tareas.

Una tarea en concreto es una unidad de trabajo que una persona que desempeñe un rol puede realizar. Las tareas tienen un objetivo concreto, normalmente expresado en términos de crear o actualizar algún producto. Cada tarea es asignada a un rol específico, y afecta a uno o un número pequeño de artefactos.

En la adición de una tarea es necesario que establezca como mínimo lo siguiente:

- Especificar el objetivo de la tarea.
- Especificar los artefactos de entrada y salida.
- Especificar el rol responsable.

La especificación del objetivo de la tarea es necesaria, ya que permite identificar los productos de trabajo que serán creados o actualizados en la ejecución, y además, orienta la ejecución del proceso, puesto que permite la valoración de la ejecución de la tarea en términos del cumplimiento de sus objetivos.

Establecer las relaciones de la tarea es uno de los aspectos importantes que hay que tener en cuenta cuando se hace la adición, es decir, especificar las entradas, salidas y el rol responsable de dicha tarea, ver Figura 9. Elementos necesarios para la definición de una tarea.

Tarea: Nombre tarea	
Objetivo	
El objetivo de esta tarea es...	
Relaciones	
Roles	Realizador principal: <ul style="list-style-type: none"> • Rol
Entradas	Obligatoria: <ul style="list-style-type: none"> • Artefacto 1 • Artefacto 2
Salidas	<ul style="list-style-type: none"> • Artefacto 1 • Artefacto 2 • Artefacto 3

Figura 9. Elementos necesarios para la definición de una tarea.

Cuando se desea adicionar una entidad dentro del proceso de desarrollo, se tiene que considerar dónde la entidad es adicionada y qué tipo de relación hay entre la nueva entidad y la entidad adyacente [4]. Entonces es determinante conocer como la adición va alterar o modificar el proceso, puesto que en la adición de la tarea previamente se debe definir en qué parte del proceso será adicionada; para realizar la adición en este trabajo sólo se tendrán en cuenta las tareas posteriores de la tarea a adicionar, donde se pueden presentar los siguientes casos:

- 1) La tarea adicionada produce artefactos que son entrada a tareas posteriores.
- 2) La tarea adicionada no afecta tareas posteriores (caso en el que ésta sea adicionada al final de un flujo de tareas).

En la siguiente sección se ilustran los casos mencionados anteriormente y se dan posibles soluciones a cada uno de ellos

- **Caso 1: La tarea adicionada produce artefactos que son entrada a tareas posteriores**
 En este caso entonces, es necesario que en la definición de la tarea se precisen los artefactos que serán producidos o modificados y que afectaran las tareas posteriores, es decir, los artefactos que son el resultado de la nueva tarea y que serán consumidos por las tareas posteriores, esto implica modificar las tareas posteriores en sus entradas para poder establecer las relaciones de las tareas por medio de los artefactos producidos por la tarea adicionada y consumidos por las tareas posteriores.
- **Caso 2: La tarea adicionada no afecta tareas posteriores.**
 La relación entre actividades es necesaria porque una actividad suelta, sin relaciones, no aportaría a la ejecución exitosa del proceso. Es importante mencionar que pueden

haber tareas que no tengan relaciones, no afectan tareas posteriores, este tipo de tareas se pueden encontrar al final de un flujo de trabajo, cuyo resultado será cumplir a cabalidad o parcialmente con el objetivo del flujo de tareas. En este caso es importante evaluar si la tarea adicionada está generando valor al flujo de trabajo en términos de conseguir los resultados esperados.

Los artefactos de entrada de la tarea adicionada pueden ser completamente nuevos o pueden estar definidos por el proceso, si es el caso de que sean nuevos hay que definirlos según la adición de artefactos que se muestra en el siguiente apartado.

Adición de Artefactos.

Un producto o artefacto es un trozo de información que es producido, modificado o usado durante el proceso de desarrollo de software. Los productos son los resultados tangibles del proyecto, las cosas que va creando y usando hasta obtener el producto final [8]. Los roles utilizan los artefactos como entrada para desarrollar una actividad, y también son el resultado o salida de dichas actividades. En la adición de artefactos se debe especificar lo siguiente:

- Especificar el objetivo del artefacto.
- Especificar las tareas donde el artefacto es entrada y salida.
- Especificar el rol responsable.

La especificación del objetivo del artefacto es inevitable, puesto que permite identificar como el artefacto ayuda parcialmente a alcanzar los objetivos del proceso.

Indicar las relaciones del artefacto es un aspecto trascendental en el momento de efectuar la adición, puesto que un artefacto no aportaría valor si no tiene relaciones, es decir, si no es entrada a alguna tarea que lo utilice o lo modifique. Por lo tanto, es necesario definir las tareas a las cuales el artefacto será entrada, salida y también definir el rol a cargo de dicho artefacto, para de esta manera establecer las relaciones entre las entidades del proceso, ver Figura 10

Artefacto: Nombre del Artefacto		
Objetivo		
El objetivo de este artefacto es...		
Relaciones		
Roles	Responsable: <ul style="list-style-type: none">• Rol 1	
Tareas	Entrada a: <ul style="list-style-type: none">• Actividad 1• Actividad 2	Salida de: <ul style="list-style-type: none">• Actividad 3• Actividad 4

Figura 10. Elementos necesarios para adicionar un artefacto.

En la adición de un artefacto se puede tener incidencias parecidas a las de la adición de una tarea, donde se pueden presentar los siguientes casos:

- El artefacto está relacionado con tareas posteriores
- El artefacto no es entrada a tareas posteriores.

Como la adición de tareas y artefactos tienen semejanza entonces las posibles soluciones serán parecidas (en la siguiente sección se describen soluciones a cada uno de los casos mencionados anteriormente).

- **Caso1: El artefacto está relacionado con tareas posteriores**
En este caso, es necesario precisar cuáles tareas posteriores consumen el artefacto para modificar sus entradas, y de esta manera establecer las relaciones entre tareas y el artefacto adicionado. Similar a lo que sucede con las tareas, un artefacto suelto sin relaciones no aporta nada a la ejecución del proceso.
- **Caso 2: El artefacto no es entrada a tareas posteriores.**
De la misma forma que en las tareas, este tipo de artefactos se pueden encontrar al final de un flujo de trabajo, cuyo resultado será cumplir a cabalidad o parcialmente con objetivo del flujo de tareas, entonces para este caso es importante evaluar si el artefacto adicionado está generando valor al flujo de trabajo en términos de conseguir los resultados esperados.

C.18. Ejemplos operación de adición vista como el mecanismo de “variabilidad de contribución de SPEM”.

Las siguientes figuras muestran entidades del contenido de un proceso, las entidades que cuyo nombre inician con “b_” serán tomadas como entidades base del proceso; y las entidades que inician con “p_” pretenden ser las entidades que son adicionadas a las entidades base del proceso.

Una entidad E que se adiciona a una entidad denotada EB de la base del proceso, añade sus valores de atributos e instancias de asociación a EB, sin modificar directamente las propiedades que ya tiene EB (es una adición). Como resultado de realizar la adición la entidad EB se muestra combinada con los atributos y relaciones de E. Las reglas que rigen dicha combinación son: (Ver ejemplos en la Figura 11 y resultado en la Figura 12):

- El valor de los atributos de E se concatena al final del contenido de los correspondientes atributos de EB.
- Las instancias de asociaciones de entrada o de salida con cardinalidad “muchos” de E se añaden a EB.
- Las instancias de asociaciones de entrada y de salida con cardinalidad “1” (como la relación “realizador principal” entre una tarea y un rol) definidas en E se ignoran si ya existen en EB. En caso contrario, se añaden.
- Una entidad de la base del proceso puede recibir varias adicciones.
- La adición es transitiva.

Ejemplo1:

Este ejemplo muestra la adición, utilizando la variabilidad de contribución de SPEM, de tres entidades del proceso (p_rol1 , p_tarea1 y $p_artefacto2$) que contribuyen a las entidades base (b_rol1 , b_tarea1 y $b_artefacto1$). La Figura 11 muestra el panorama inicial de las contribuciones que se quieren efectuar.

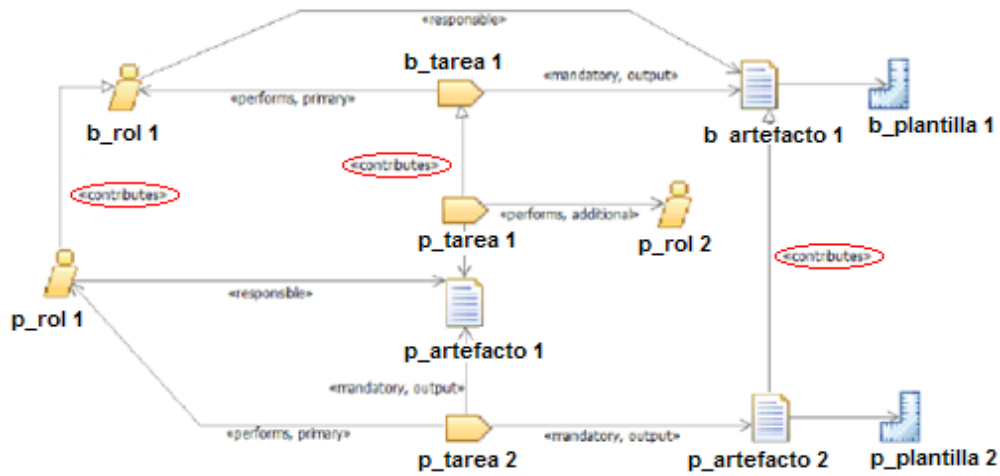


Figura 11. Adición de entidades.

Luego de interpretar las reglas que rigen la variabilidad de contribución el resultado será la Figura 12, la cual muestra que a las entidades base se les añaden las instancias de asociación de las entidades que contribuyen, por ejemplo *b_rol1* no tenía instancias de asociación con *p_tarea1*, pero que luego de realizar la contribución *b_rol1* tiene una nueva asociación con *p_rol1* y también añadirá los valores de los atributos de *p_rol1*. Es importante indicar que cuando se adicionan nuevas entidades a las entidades base del proceso, se añaden los valores de atributos e instancias de asociación de las entidades nuevas a las entidades base del proceso, es decir, no se modifica directamente las propiedades que ya tienen las entidades base del proceso (es una adición).

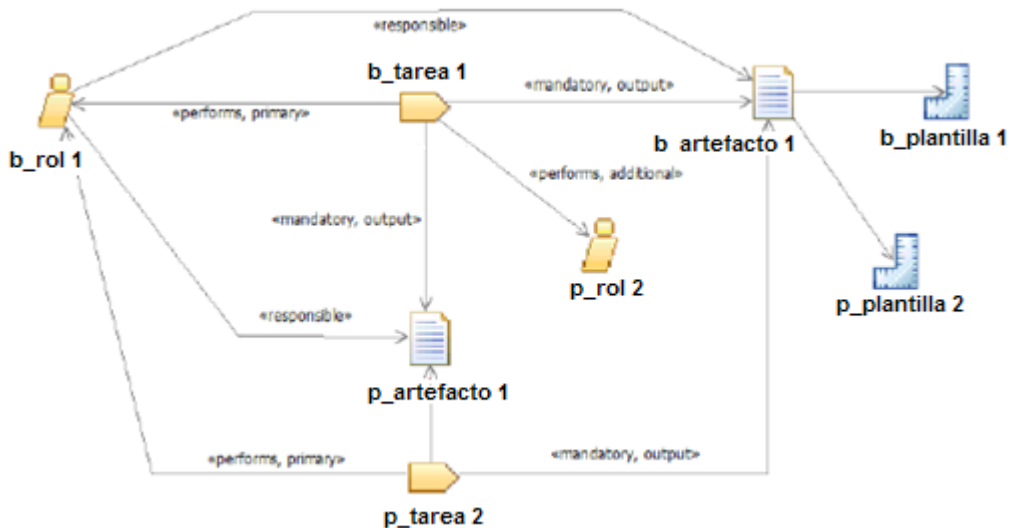


Figura 12. Resultado de la adición Figura 11.

Ejemplo 2:

Este ejemplo ilustra la adición como parte de una composición ver Figura 13, donde se puede observar que la composición está formada por los artefactos *p_artefacto1*, *p_artefacto3* y *p_artefacto2*.

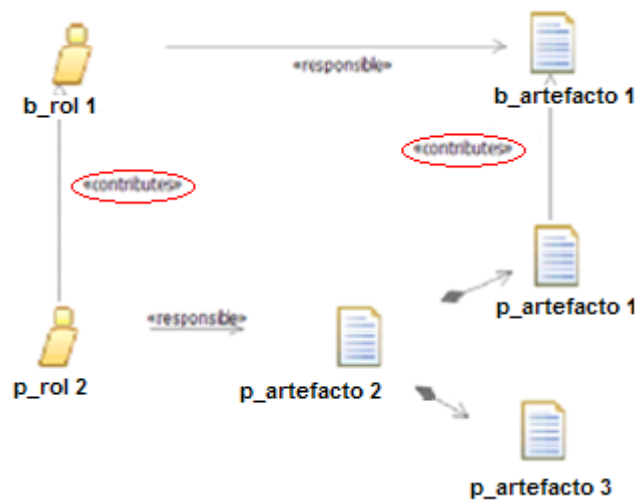


Figura 13. Adición en una composición

Cuando se han interpretado las reglas que rigen la variabilidad de contribución las entidades base del proceso también serán parte de la composición. La consecuencia de interpretar las reglas de variabilidad para este caso da como resultado la Figura 14, donde se muestra que la entidad base *b_rol1* se le añaden las instancias de asociación de *p_rol2* y por lo tanto también los valores de los atributos, en este ejemplo a diferencia del ejemplo1 muestra como la contribución es interpretada cuando hay una composición de entidades, entonces para este caso la contribución inicial entre *p_artefacto1*, *p_artefacto3* y *p_artefacto2* quedara formada por *b_artefacto1*, *p_artefacto2* y *p_artefacto3*, puesto que la entidad base *b_artefacto1* adquiere los atributos y asociaciones de *p_artefacto1*.



Figura 14. Resultado de la Adición.

Ejemplo 3:

En este ejemplo se ilustra la adición en el caso de que existan asociaciones entrantes y salientes en la entidad a ser adicionada ver Figura 15; la asociación entrante a la entidad *p_rol1* esta definida con *p_tarea1* y la asociación saliente con *p_artefacto1*.

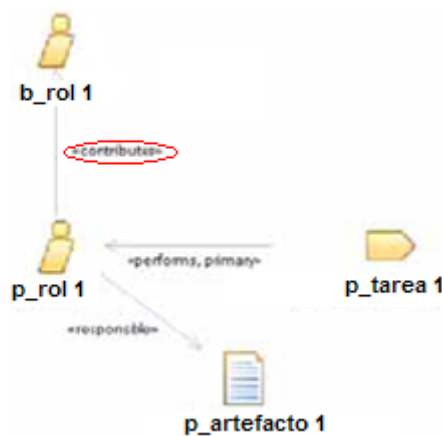


Figura 15. Asociaciones Entrantes

El resultado de aplicar las reglas que rigen la variabilidad de contribución se ilustra en la Figura 16, donde los valores de los atributos de la entidad que contribuye *p_rol1* serán adicionados a *b_rol1* y las asociaciones entrantes y salientes de *p_rol1* pasan a ser asociaciones de la entidad base *b_rol1*.

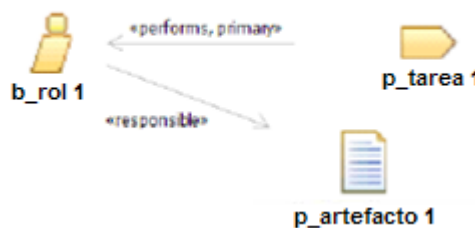


Figura 16. Resultado de la adición en asociaciones entrantes y salientes

C.19. Ejemplos operación de reemplazo vista como el mecanismo de “variabilidad reemplaza de SPEM”.

En las siguientes figuras se muestran entidades del contenido de un proceso, las entidades que su nombre inician con “b_” serán tomadas como entidades base del proceso; y las entidades que inician con “p_” pretende ser las entidades que reemplazarán las entidades base del proceso.

Una entidad E puede reemplazar (sustituir) los atributos y las asociaciones de salida de una entidad denotada EB de la base del proceso, sin modificar directamente ninguna propiedad de EB. Las reglas que regulan el reemplazo (un ejemplo se muestra en la Figura 17 y el resultado en la Figura 18) son:

- Los valores de los atributos de EB se reemplazan por los respectivos valores de los atributos no vacíos de E.
- Las instancias de asociación de salida de EB se reemplazan por las de E.

- Las instancias de asociación de entrada con cardinalidad “muchos” de EB se amplían con las instancias de dichas asociaciones existentes para E.
- Las instancias de asociación de entrada con cardinalidad “uno” de EB se dejan intactas. Las instancias incluidas en E que no están en EB se añaden a EB.
- Una entidad base sólo puede ser reemplazado por otra única entidad.
- El reemplazo es transitivo.

Ejemplo 1:

En este ejemplo se muestra cómo el artefacto base *b_artefacto1* es reemplazado y sus asociaciones reinterpretadas, las asociaciones entrantes con *b_rol* y *b_tarea1* son preservadas en el reemplazo, es decir *p_artefacto1* el cual reemplazará a *b_artefacto1* conservará las asociaciones entrantes de *b_artefacto1* y también sus propias asociaciones, ver Figura 41. La asociación de salida del artefacto base *b_artefacto1* con *b_plantilla1* es reemplazada por la de *p_artefacto1* con *p_plantilla2*.

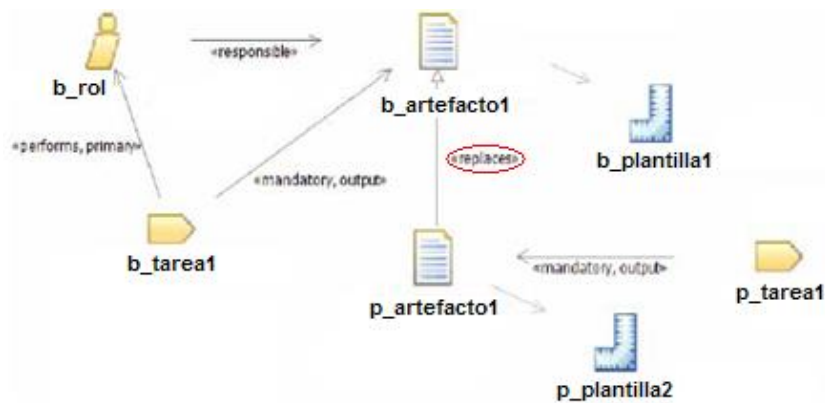


Figura 17. Reemplazo.

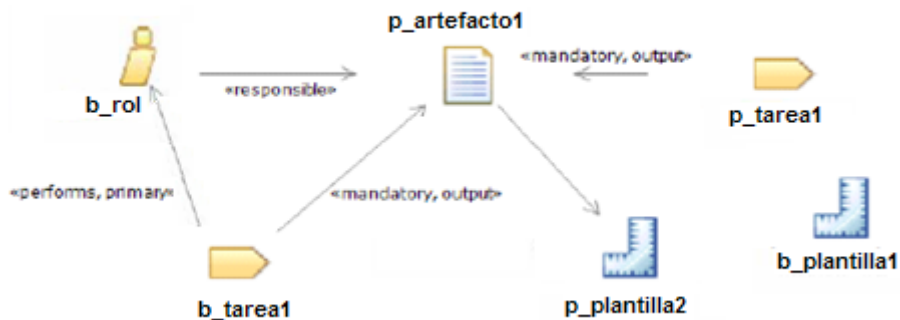


Figura 18. Resultado del reemplazo de Figura 17.

Ejemplo 2

Este ejemplo muestra la definición transitiva del reemplazo más la combinación de la adición con el reemplazo, Figura 19. La jerarquía de relaciones de los productos de trabajo son evaluados de forma top-down, primero la adición y segundo el reemplazo.

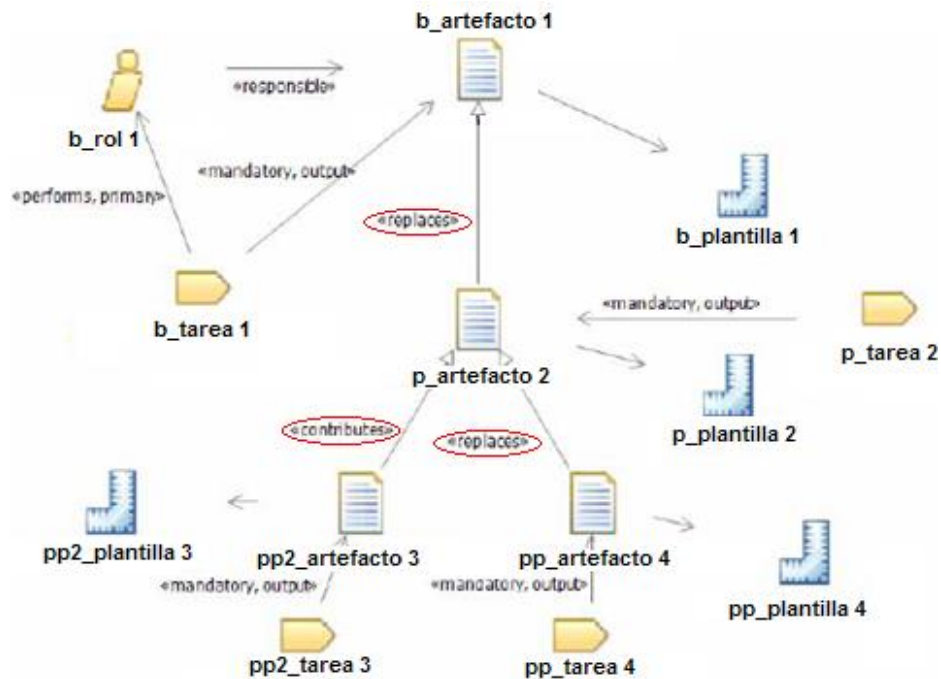


Figura 19. Combinación de la adición con el reemplazo.

Al realizar la evaluación top-down de las relaciones de los productos de trabajo en este ejemplo, en primer lugar se encuentra solo un reemplazo, entonces primero se aplican las reglas que regulan el reemplazo donde *p_artefacto2* reemplaza a *b_artefacto1*, luego en la jerarquía de relaciones quedaría pendiente realizar un reemplazo y una adición, en esta situación primero se efectúa la adición siguiendo las reglas definidas para ésta, donde el artefacto *p_artefacto2* añade las instancias de asociación de *pp2_artefacto3* y por lo tanto también los valores de los atributos, y por último también siguiendo las reglas que rigen el reemplazo *pp_artefacto4* reemplaza a *p_artefacto2*. El resultado muestra que las asociaciones entrantes desde las tareas son preservadas además solo la asociación saliente a *pp_plantilla4* es preservada, a causa de la sustitución final de *pp_artefacto4*, Figura 20.



Figura 20. Resultado de la combinación de la adición con el reemplazo.

Referencias Anexo C

- [1] *Ruiz Francisco, Verdugo Javier.* Guía de Uso de SPEM 2 con EPF Composer. Universidad de Castilla-La Mancha Escuela Superior de Informática Departamento de Tecnologías y Sistemas de Información.
- [2] Ivar Jacobson, Grady Booch, and Jim Rumbaugh, Unified Software Development Process, Addison- Wesley, 1999.
- [3] Walker Royce, Software Project Management—A Unified Framework, Addison-Wesley, 1998.
- [4] Tailoring Software Evolution Process. Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing

ANEXO D: APLICACIÓN DE GTUP – AMBIENTE ACADEMICO Y EMPRESARIAL

D.1. Anexos generales

D.1.1. Documento taller SPEM y EPF

1. Introducción

El proceso unificado de Rational (RUP), Métrica 3 (M3) y demás metodologías están basadas en un conjunto de ideas y conceptos subyacentes, que no están definidos de manera explícita, es decir, no es evidente el metamodelo subyacente utilizado. Además, el formato en que se maneja la información de dichas metodologías son documentos de texto en lenguaje natural. Esto tiene la gran desventaja de que toda la manipulación (creación, revisión, reutilización, adaptación, etc.) y generación de documentación (publicación) para dichas metodologías es un proceso puramente manual. Para evitar dicha situación, OMG (la organización industrial promotora de UML) ha desarrollado y aprobado recientemente el estándar SPEM (Software & Systems Process Engineering Metamodel) versión 2.0⁴, que pretende ser el estándar industrial para la representación de modelos de procesos de ingeniería del software e ingeniería de sistemas.

Por otro lado, dentro de la plataforma abierta ECLIPSE, se ha puesto en marcha el proyecto EPF (Eclipse Process Framework) ⁵, que ha desarrollado un editor de SPEM

⁴ <http://www.omg.org/cgi-bin/doc?ptc/07-08-07>

² <http://www.eclipse.org/epf/>

2, llamado “EPF Composer”, en adelante EPFC. EPFC se basa en el estándar SPEM 2 y permite definir, gestionar y reutilizar un repositorio de fragmentos de métodos y procesos. Con EPFC se pueden crear implementaciones en formato SPEM 2 de cualquier método, proceso o metodología de ingeniería del software.

En este documento se presentan las características y conceptos básicos de SPEM 2, así como la manera en que sus conceptos y elementos se manejan con la herramienta EPFC.

2. Contexto de Ingeniería de Procesos

SPEM 2 es un estándar de metamodelado que sirve para representar procesos de ingeniería de software, entiendo un Proceso Software (PS) como “Un conjunto coherente de políticas, estructuras organizacionales, tecnologías, procedimientos y artefactos que son necesarios para concebir, desarrollar, instalar y mantener un producto software” (Fugetta, 2000). SPEM 2 se encuadra dentro de la Ingeniería de Procesos de Software, en inglés Software Processes Engineering (SEP), que es un área nueva de la Ingeniería de Software dedicada a “la definición, implementación, medición y mejora de los procesos de Ingeniería de Software”.

Gracias al uso de SPEM 2, se puede disponer de modelos de PS en formato procesable por computador, lo que proporciona capacidades para:

- Facilitar la comprensión y comunicación humana.
- Facilitar la reutilización.
- Dar soporte a la mejora de procesos.
- Dar soporte a la gestión de procesos.
- Guiar la automatización de procesos.
- Dar soporte para la ejecución automática.

SPEM es a los procesos software lo mismo que UML es a los sistemas software. UML es un metamodelo que sirve para representar modelos de sistemas software y SPEM es un metamodelo que sirve para representar modelos de procesos software.

3. Características Básicas de SPEM 2.0

SPEM 2 (Software & Systems Process Engineering Metamodel Specification, v2.0) es un metamodelo para modelos de procesos de ingeniería del software y de ingeniería de sistemas. SPEM 2 sirve para definir procesos de desarrollo de software y sistemas y sus componentes. Su alcance se limita a los elementos mínimos necesarios para definir dichos procesos sin añadir características específicas de un dominio o disciplina particular; pero sirve para métodos y procesos de diferentes estilos, culturas, niveles de formalismo, o modelos de ciclos de vida. No es un lenguaje de modelado de procesos en general, ya que está orientado a los procesos software.

La idea central de SPEM 2 para representar procesos está basada en tres elementos básicos: rol, producto de trabajo y tarea. Las tareas representan el esfuerzo a hacer, los roles representan quien lo hace y los productos de trabajo representan las entradas que se utilizan en las tareas y las salidas que se producen. La idea central subyacente

es que un modelo de proceso consiste, básicamente, en decir quien (rol) realiza qué (tarea) para, a partir de unas entradas (productos de trabajo) obtener unas salidas (productos de trabajo). La figura 1 muestra la idea básica del Proceso.

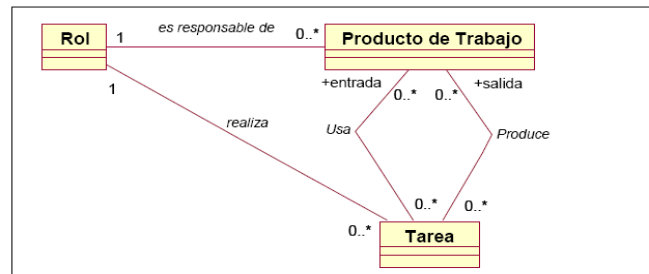


Figura 1 muestra la idea básica del Proceso.

4. Lenguaje SPEM

En SPEM 2 se distinguen dos grupos de conceptos a la hora de implementar una metodología (ver Figura 2 y Figura 3):

- a) Primero, se puebla el Method Content (contenido del método) con Content Elements (elementos de contenido), es decir, los elementos primarios o constructores básicos.
- b) Después, se combinan y reutilizan dichos elementos para obtener Processes (procesos).

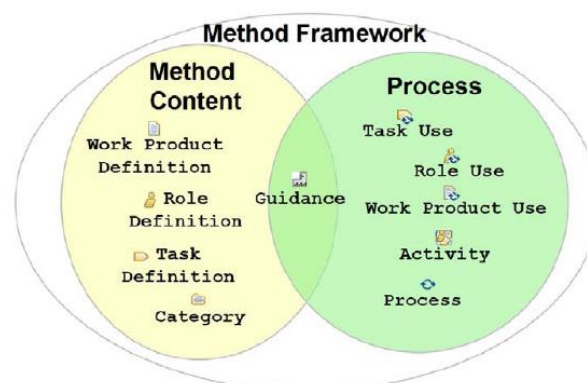


Figura 2 Aspectos principales para modelar con SPEM

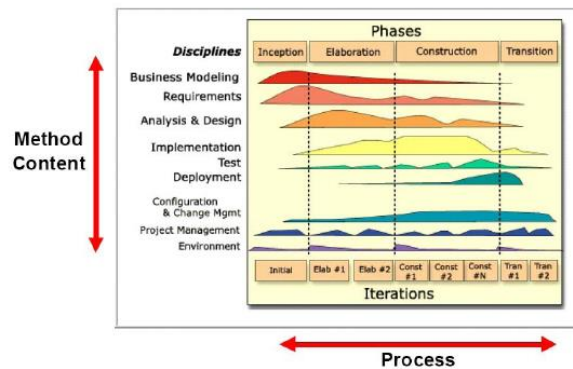


Figura 3 Ortogonalidad del Method Content y de los procesos (igual que ocurre en RUP)

En SPEM 2 la jerarquía de desglose del trabajo, es decir, los conceptos existentes para representar el esfuerzo a realizar a distintos niveles de detalle, son los siguientes del más general al más particular (ver Figura 4):

- Delivery Process (proceso de despliegue): Representa un proceso tal complejo como se necesite, que será el que sirva de base para realizar cierto tipo de proyectos.
- Capability Pattern (patrón de capacidad): representa un patrón de proceso, es decir, un fragmento de proceso que puede ser reutilizado más de una vez en un delivery process.
- Activity (actividad): en el elemento central para definir procesos ya que permite organizar los elementos básicos de proceso (roles, productos de trabajo y tareas).
- Task (tarea): es la porción más pequeña de trabajo en un modelo de proceso en SPEM 2.

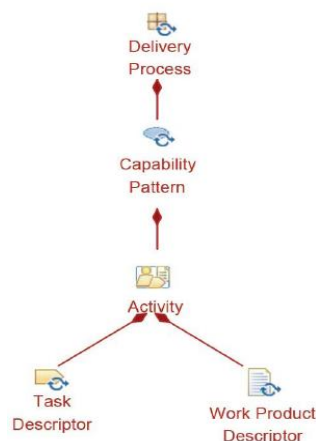


Figura 4 Conceptos para representar la jerarquía de desglose de trabajo.

2.2.2 Organización de un Repositorio SPEM

Un repositorio o biblioteca de métodos y procesos (Method Library) en SPEM 2 es una colección de uno o más plugins y una o varias configuraciones (Configuration). Cada

plugin se almacena en un directorio de disco diferente e incluye contenido de método (Method Content) y procesos (Processes), cada uno de los cuales se refiere a las dos partes, izquierda y derecha, mostradas en la Figura 2. A su vez, el contenido de método está formado por paquetes de contenido (Content Package), categorías estándar (Standard Category) y categorías personalizadas (Custom Category); y el apartado de Procesos contiene patrones de proceso (Capability Pattern) y procesos para despliegue (Delivery Process). Un ejemplo de toda esta estructura se muestra en la Figura 5.

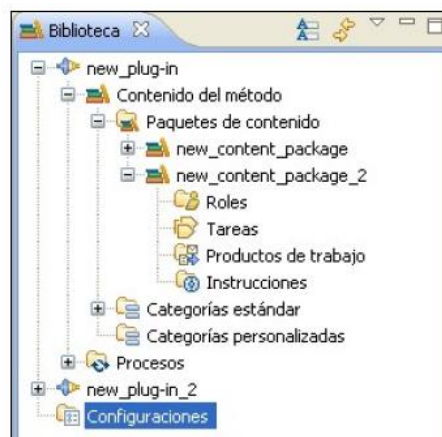


Figura 5 ejemplo Method Library)

2.2 Contenido de Método

A continuación se presentan los conceptos del Contenido de Método (Method Content) de SPEM 2. No es una revisión exhaustiva, ya que no se incluyen aquellos conceptos que quedan “ocultos” al usuario cuando modela procesos con algún editor de SPEM, por ejemplo, con EPF Composer.

El contenido de método puede ser organizado a voluntad del usuario mediante una jerarquía de paquetes de contenido (Content Package), cada uno de los cuales puede incluir roles, tareas, productos de trabajo y guías (ver Figura 5).

2.2.1 Elementos de Contenido

Los elementos de contenido (Content Element) son los constructores básicos y se derivan del patrón primitivo de trabajo: alguien (rol) hace algo (tarea) para obtener algo (producto de trabajo) basándose o ayudándose en algo (guía). En consecuencia, los cuatro tipos de elementos de contenido son: Tarea, Rol, Producto de Trabajo y Guía. Adicionalmente, existe un quinto tipo de elemento de contenido incluido con fines de clasificación y agrupación, llamado Categorías. En los siguientes subapartados se presenta cada uno de ellos.

2.2.1.1 Tarea

Una Tarea (Task Definition) describe una unidad de trabajo asignable y gestionable, es decir, es la unidad atómica de trabajo para definir procesos. Su granularidad es de unas pocas horas a unos pocos días, afectando a unos pocos productos de trabajo y vinculando a unos pocos roles. Es un Elemento de Método que define el trabajo

realizado por roles, pero también es una Definición de Trabajo (en procesos). Una Tarea está asociada con:

- 1..* Roles, distinguiendo entre:
 - 1 realizador principal obligatorio [responsable]
 - 0..* realizadores adicionales opcionales
- 1..* Productos de Trabajo como:
 - Entradas obligatorias
 - Entradas opcionales
 - Salidas
- 0..* Herramientas, que se recomienda usar.
- 0..* Pasos, que describen de forma secuencial el trabajo a realizar.
- 0..* Habilidades, que se requieren habitualmente para llevar a cabo la tarea.

2.2.1.2 Roles

Un Rol (Role Definition) define un conjunto de habilidades, competencias y responsabilidades relacionadas, de un individuo o de un grupo. No se deben confundir roles con personas, ya que la vinculación entre personas y roles se realiza durante la planificación del proyecto y puede ocurrir que un individuo desempeñe varios roles o que un rol sea desempeñado por varios individuos. Un rol es un Elemento de Método usado en las Definiciones de Tareas para señalar quienes las realizan. Un Rol está asociado con:

- 0..* Productos de Trabajo, de los que es responsable.
- 0..* Habilidades, que el rol típicamente provee.

2.2.1.3 Productos de Trabajo

Un Producto de Trabajo (Work Product Definition) es consumido, producido o modificado por Tareas. Un Producto de Trabajo puede estar asociado con otros 0..* Productos de Trabajo mediante asociaciones de los siguientes tipos:

- Composición (Composition), cuando las instancias de un producto de trabajo sirven para componer instancias de otro producto de trabajo. Ejemplo: “Actores” se emplean para componer “Casos de uso”.
- Agregación (Aggregation), si un producto de trabajo está formado por agregación de otros. Ejemplo: el “Manual de usuario” incluye el “Manual de instalación”.
- Es impactado por (Impacted by), cuando un producto de trabajo impacta en otro, es decir, si los cambios del primero obligan a cambiar el segundo. Ejemplo: si cambia el “Modelo de casos de uso”, es necesario adaptar a dicho cambio la “Realización de casos de uso”.

Existen tres tipos predefinidos (especializaciones) de Productos de Trabajo:

- Artefacto (Artifact): de naturaleza tangible (modelo, documento, código, archivos, ..).Un artefacto puede estar formados por otros artefactos más simples.
- Entregable 5 (Deliverable): provee una descripción y definición para empaquetar otros productos de trabajo con fines de entrega a un cliente

interno o externo. Representa una salida de un proceso que tiene valor para un usuario, cliente u otro participante (*stakeholder*). Esta asociado con 0.* componentes de entregable (Deliverable Component), que son los productos de trabajo, habitualmente artefactos, que lo forman.

- Resultado (Outcome): un producto de trabajo de naturaleza intangible (resultado o estado), o que no está formalmente definido.

2.2.1.4 Guías

Una guía o instrucción (Guidance) es un elemento de método (o de proceso) que provee información adicional relacionada con otros elementos. Por ejemplo: ayuda o información sobre cómo trabaja un rol, cómo crear un producto de trabajo, cómo usar una herramienta o cómo realizar una tarea.

2.2.1.5 Categorías

Una Categoría (Category) es un elemento de contenido, o de proceso, usado para categorizar, es decir, clasificar o agrupar dichos elementos en base a los criterios que desee el ingeniero de procesos. Una categoría puede tener 0..* subcategorías. Esto permite establecer cualquier tipo de jerarquía de agrupamiento de elementos. SPEM 2 distingue dos clases de categorías:

- Estándar (Standard Category): Vienen predefinidas en SPEM.
- Personalizada (Custom Category): Sirven para que el ingeniero de procesos pueda definir otras categorías nuevas.

2.2.2 Propiedades Principales

Todos los content elements de todos los tipos anteriores tienen las propiedades de los elementos describibles, incluyendo: name (nombre); presentation name (nombre de presentación); brief description (descripción breve); main description (descripción principal). La última permite utilizar un “rich text editor”, pudiendo incluir texto con formatos variados, imágenes, etc., al estilo de HTML.

Otras propiedades específicas de cada clase de content element son:

- Task: purpose (objetivo); key considerations (factores clave); alternatives (alternativas); steps (lista de pasos), que detalla el trabajo a realizar de forma ordenada.
- Role: key considerations (factores clave); skills (habilidades); assignment approaches (propuestas de asignación); synonyms (sinónimos).
- Work Product: unique ID (ID exclusivo); purpose (objetivo); key considerations (factores clave); impact of not having (impacto de no tener); reason for not needing (motivos para no necesitar). Además, los Artifacts tienen brief outline (esquemización breve), y representation options (opciones de representación). Los Deliverables también tienen external description (descripción externa), packaging guidance (guía de empaquetado), y deliverable parts (componentes de entregable), que es una lista de otros work products que forman parte del deliverable.

- **Guidance:** algunos tipos de guidances tienen otros atributos. Así, Checklist tiene una lista de check ítems (elementos de comprobación), y Practice tiene additional information (información adicional), goals (objetivos), application (aplicación), problem (problema), background (fondo), level of adoption (nivel de adopción), referentes (referencias), que es una lista de content elements referenciados. Por otro lado, un Template tiene una lista de template files (archivos de plantilla).

Cada clase de content element tiene un icono predefinido por el perfil UML de SPEM, que lo caracteriza en los diagramas y documentación utilizados o generados también por las diversas herramientas (como EPF Composer). Con fines de personalización, a cada tipo de work product y guidance se le puede asociar otro icono específico.

2.2.3 Asociaciones

Las asociaciones permitidas en SPEM entre los diversos tipos de content elements de manera resumida son:

- **Task-Steps:** lista ordenadas de pasos que se llevan a cabo en una tarea.
- **Task-Roles:** primary performer (realizador principal, obligatorio); additional performers (realizadores adicionales).
- **Task-Work Products:** mandatory inputs (entradas obligatorias); optional inputs (entradas opcionales); outputs (salidas).
- **Task-Guidances:** guidances relacionadas (sólo tipos checklist, concept, estimating guideline, example, guideline, reusable asset, supporting material, y tool mentor).
- **Task-Categories:** disciplines y custom categories a las que pertenece la task.
- **Role-Work Products:** responsible for (responsable de); work products that are output of tasks that this role performs (productos de trabajo que son salida de tareas que realiza este rol).
- **Role-Guidances:** guidances relacionadas (sólo tipos checklist, concept, example, guideline, reusable asset, y supporting material).
- **Role-Categories:** role sets y custom categories a las que pertenece el role.
- **Work Product-Guidances:** guidances relacionadas (sólo tipos checklist, concept, estimating guideline, example, guideline, report, reusable asset, supporting material, template, y tool mentor).
- **Work Product-Categories:** domains, work product kinds y custom categories a las que pertenece el work product.
- **Guidance-Guidances:** lista de guidances (de tipo checklist, concept, example, guideline, reusable asset y supporting material) que pertenecen al guidance principal. Todos los tipos de guidance pueden incluir otros guidance, salvo el tipo practice que en lugar de incluir otros guidances está relacionado con diversos content elements.

2.3 Procesos

En SPEM 2 se distinguen dos etapas a la hora de implementar un proceso o metodología: primero se puebla el Contenido de Método, cuyos conceptos han sido presentados anteriormente, y en segundo lugar, se combinan y reutilizan dichos

elementos para ensamblar actividades y procesos. A continuación se presentan los conceptos de SPEM que se manejan para ensamblar patrones de proceso y procesos completos. No es una revisión exhaustiva ya que, por motivos de sencillez, se ha optado por no incluir los conceptos de SPEM 2 que quedan “ocultos” al usuario cuando modela procesos con algún editor de SPEM. En los sub-apartados siguientes se presentan conceptos que corresponden, principalmente, con los paquetes “Core”, “Process Structure”, “Process with Methods” (en parte) y con el “Base Plug-in” (que incluye las especializaciones predefinidas de actividad).

2.3.1 Aspectos Generales

Una Definición de Trabajo (Work Definition) es un concepto abstracto que generaliza todos los tipos de definiciones de trabajo en SPEM. Una Definición de Trabajo puede estar asociada con 0..* precondiciones (restricciones que deben cumplirse para que el trabajo pueda comenzar) y con 0..* postcondiciones (restricciones que deben cumplirse para que el trabajo pueda considerarse concluido). Las definiciones de trabajo pueden ser modeladas de forma gráfica mediante diagramas de actividad de UML 2.

Un Elemento de Desglose (Breakdown Element) es una generalización abstracta para cualquier tipo de elemento que aparece en un proceso y es parte de una estructura de desglose. Aporta tres propiedades importantes:

- Admite Varias Apariciones (Has Multiple Occurrences): al realizar el proceso puede haber mas de una instancia del elemento.
- Es Opcional (Is Optional): no es obligatoria su inclusión cuando se lleva a cabo el proyecto.
- Planeado (Is Planned): El elemento es incluido al generar los planes de proyecto que se exportan a las herramientas de gestión de proyectos.

Los tipos de Elementos de Desglose son:

- Parámetros de Procesos, para asociar productos de entrada/salida.
- Realizadores de Procesos, para asociar roles.
- Secuencias de Trabajo, para las relaciones de precedencia.
- Elementos de Desglose de Trabajo.
- Roles en Uso.
- Productos de Trabajo en Uso.
- Relaciones entre Productos de Trabajo en Uso.
- Asignaciones de Responsabilidad en Procesos.

Cada elemento de desglose tiene asociada una estructura de desglose de trabajo (WBS, Work- Breakdown Structure) que representa su estructura interna y, opcionalmente, un flujo de trabajo (Work Flow). La figura 6 muestra un ejemplo (instancia) de ambos.

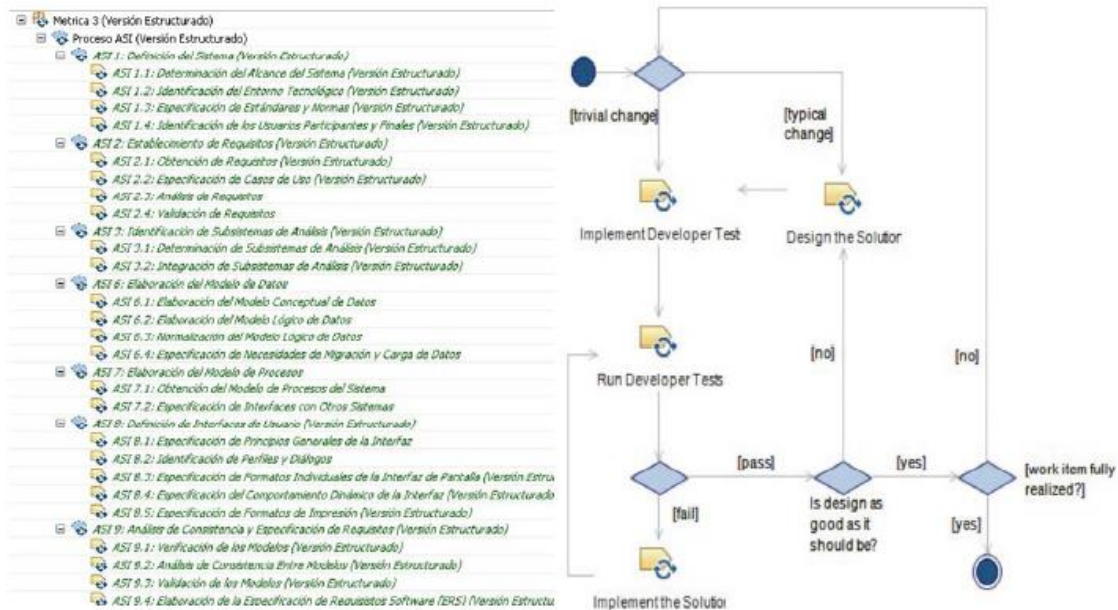


Figura 6. Ejemplos de Estructura de Desglose de Trabajo (izquierda) y de Flujo de Trabajo (derecha).

2.3.2 Elementos de Desglose de Trabajo

Los Elementos de Desglose de Trabajo (Work Breakdown Element) son el principal tipo de elemento de desglose, ya que representan la descomposición del trabajo. Existen dos tipos: Actividad e Hito. El flujo entre los Elementos de Desglose de Trabajo se representa por medio de Secuencias de Trabajo (Work Sequence). Cada Secuencia de Trabajo conecta dos Elementos de Desglose de Trabajo, nominados como predecesor y sucesor.

En SPEM 2 la descomposición del trabajo en distintos niveles de detalle se realiza mediante el concepto de Actividad (Activity), de forma que cualquier elemento de desglose de trabajo con estructura interna (es decir, que incluye elementos de desglose) recibe el nombre de Actividad, independiente del nivel de desglose, es decir, del tamaño del fragmento de trabajo representado. En consecuencia, una estructura de desglose de trabajo (WBS) se representa mediante una recursividad de agregaciones: una actividad está formada por agregación de diversos elementos de desglose, entre los que se encuentran actividades más pequeñas que a su vez pueden estar formadas de igual manera.

2.3.2.1 Actividades

Una Actividad (Activity) representa una unidad de trabajo general en un proceso. Una actividad puede tener estructura interna formada por agregación de elementos de desglose, que pueden ser de varios tipos, no solo de trabajo: actividades mas simples (anidamiento), elementos de método en uso (roles en uso, productos de trabajo en uso), e hitos. La complejidad de la estructura de desglose de una actividad puede

variar entre 0 tareas (pueden estar formadas sólo por productos de trabajo en uso) o todo un proceso completo.

SPEM 2 tiene predefinidos varios tipos especiales de Actividades: Iteración, Fase y Proceso. Puesto que Actividad es un concepto genérico, que incluye a proceso como especialización, en SPEM 2 no se respeta la nomenclatura tradicional de los estándares ISO de ingeniería del software. En especial, debe tenerse en cuenta que SPEM no sigue los tres niveles estáticos de descomposición utilizados en el modelo de procesos de ISO/IEC 12207: procesos formado por actividades formadas por tareas.

2.3.2.2 Contenido de Método en Uso

Los elementos del contenido de método (tareas, roles y productos de trabajo) reutilizados en los procesos reciben el nombre de elementos en uso. Un elemento en uso es una instancia de un elemento de método particularizada para un contexto de proceso determinado. Por ello, los elementos en uso no son reutilizables (para reuso ya están las descripciones de los elementos en el contenido de método).

Así, una Tarea en Uso (Task Use) representa la ocurrencia de una Definición de Tarea (Contenido de Método) en el contexto de una Actividad. En ella se pueden precisar y modificar, respecto de la tarea original, su documentación, pasos, roles y productos de entrada y de salida. Además, se pueden definir dos asociaciones que no aparecen en las tareas como elementos de contenido de método: los roles que asisten en la tarea y las entradas externas. Existen dos maneras de utilizar tareas en actividades:

- a) Reutilizar una descripción de tarea del contenido de método, que pasa a ser una tarea en uso, o
- b) Crear directamente en la actividad una instancia de tarea en uso. En este caso no se tendrá descripción asociada, sino sólo el nombre.

Un Producto de Trabajo en Uso (Work Product Use) representa la ocurrencia de un Producto de Trabajo real en el contexto de una Actividad, pudiendo modificar su documentación. Además, se pueden definir dos atributos que no aparecen en la definición de producto de trabajo en el contenido de método: los estados de entrada y de salida del producto en la actividad. Por ejemplo, un producto “Requisitos” puede entrar en una actividad en el estado “identificados” y salir en el estado “refinados”. También existen las dos maneras de utilizar productos de trabajo en actividades equivalentes a las dos indicadas para las tareas:

- a) Reutilizar una descripción de producto de trabajo del contenido de método, que pasa a ser un producto de trabajo en uso.
- b) Crear directamente en la actividad una instancia de producto de trabajo en uso, que sólo tendrá nombre pero no descripción asociada.

De forma similar a las dos anteriores, un Rol en Uso (Role Use) representa la ocurrencia de un Rol real en el contexto de una Actividad, pudiendo particularizar la documentación, productos de los que es responsable o que modifica, y equipos (roles

compuestos) a los que pertenece. Igualmente, existen dos maneras de utilizar roles en actividades:

- a) Reutilizar una descripción de rol del contenido de método, que pasa a ser un rol en uso. SPEM le llama Realizador (Performer).
- b) Crear directamente en la actividad una instancia de rol en uso. En este caso no se tendrá descripción asociada, sino sólo el nombre. SPEM le llama Participante (Participant).

Un Rol Compuesto (Composite Role) es un rol en uso especial, que se corresponde con más de una Definición de Rol del Contenido de Método

En la Figura 7 se muestra un ejemplo de Contenido de Método en Uso (parte derecha) referenciando a Contenido de Método (parte izquierda). La Figura 8 muestra otro ejemplo de una Tarea en Uso dentro de una Estructura de Desglose de Trabajo.

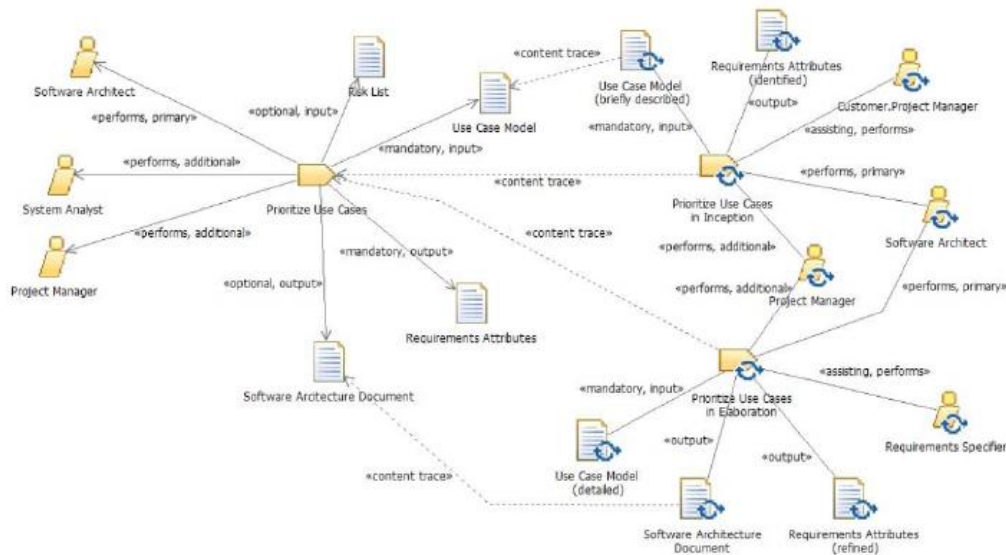


Figura 7. Ejemplo de contenido de método en uso.

Nombre de presentación	Ín...	Prede...	In...	Tipo
Fase: CENTRAL	0			Patrón de posibilidad
ASI 06: DISEÑO DE CLASES	1			Patrón de posibilidad
ASI 06.1: Selección de casos de uso	2			Descriptor de tareas
ASI 06.2: Identificación de clases asociadas a los casos de uso	3	2		Descriptor de tareas
ASI 06.3: Descripción de la interacción entre objetos	4	3		Descriptor de tareas
ASI 07: DISEÑO DE LA ARQUITECTURA DE MÓDULOS DEL SISTEMA	5	1		Patrón de posibilidad
ASI 07.1: Identificación de responsabilidades y atributos	6			Descriptor de tareas
ASI 07.2: Identificación de asociaciones y agregaciones	7	6		Descriptor de tareas
ASI 07.3: Identificación de generalizaciones	8	7		Descriptor de tareas
ASI 08: ESPECIFICACIÓN DE LA INTERFAZ DE USUARIO	9	5		Patrón de posibilidad
ASI 08.1: Identificación de perfiles y diálogos	10			Descriptor de tareas
ASI 08.2: Especificación de formatos individuales de la interfaz de pantalla	11	10		Descriptor de tareas
ASI 08.3: Especificación del comportamiento dinámico de la interfaz	12	11		Descriptor de tareas
ASI 08.4: Especificación de formatos de impresión	13	12		Descriptor de tareas
ASI 09: ESPECIFICACIÓN DE ELEMENTOS DE SEGURIDAD	14	9		Patrón de posibilidad
ASI 09.1: Identificación de elementos de seguridad	15			Descriptor de tareas
ASI 10: APROBACIÓN E INTEGRACIÓN DE CLASES DE ANÁLISIS	16	14		Patrón de posibilidad
ASI 10.1: Aprobación del modelo de clases de análisis	17			Descriptor de tareas
ASI 10.2: Integración de clases de análisis con las de iteraciones anteriores	18	17		Descriptor de tareas
DSI 03: DISEÑO DE CASOS DE USO REALES	19	16		Patrón de posibilidad
DSI 03.1: Identificación de clases de diseño asociadas a los casos de uso	20			Descriptor de tareas
DSI 03.2: Diseño de la realización de casos de uso	21	20		Descriptor de tareas

Figura 8. Una tarea en uso dentro de una estructura de desglose de trabajo

2.3.2.3 Fases, Iteraciones e Hitos

Las fases e iteraciones son dos tipos especiales de actividades (adicionales a los procesos), mientras que los hitos son un tipo de elemento de desglose de trabajo, distinto de los otros dos, actividades y tareas.

Una Fase (Phase) representa un periodo de tiempo que es significativo para un proyecto, y que acaba con un punto de control de gestión importante, un hito o un conjunto de entregables concluidos. En la práctica, en SPEM 2 una fase es una actividad que tiene el valor falso en la propiedad “Es repetible”.

Por el contrario, una Iteración (Iteration) representa un conjunto de actividades anidadas que se repiten más de una vez. Sirve para organizar ciclos repetitivos de trabajo. En la práctica, en SPEM 2 una iteración es una actividad que tiene el valor verdadero en la propiedad “Es repetible”.

Un Hito (Milestone) representa un evento significativo para el desarrollo de un proyecto:

- Una decisión importante,
- La conclusión de un entregable,
- La conclusión de una fase, etc.

Los hitos son elementos de desglose del trabajo y, por tanto, aparecen en la estructura de desglose de trabajo y pueden tener relaciones de precedencia.

2.3.3 Tipos de Procesos

En SPEM 2, un Proceso (Process) es un tipo de Actividad que describe una estructura para tipos particulares de proyectos o partes de ellos. Mediante un proceso de SPEM 2 se pueden representar distintos tipos de métodos de ingeniería del software: un proceso, un modelo de procesos completo, un ciclo de vida con sus diversos procesos, o una metodología completa.

En SPEM 2 existen dos clases principales de procesos:

- Patrón de Proceso (Capability Pattern): Es un “fragmento de proceso” que describe un grupo de actividades reutilizable como solución a algún tipo de problema o situación habitual. Se definen para poder ser empleados más de una vez en uno o varios procesos o con fines de organización. Se pueden almacenar en una jerarquía de Paquetes de Proceso (Process Package). Algunos escenarios de uso de patrones de proceso son:
 - Servir como bloques para construir Procesos para Despliegue o Patrones de Proceso más complejos.
 - Ayudar a la ejecución de proyectos que no siguen un proceso bien definido, sino que trabajan en base a fragmentos de proceso (buenas prácticas) de una manera flexible (métodos ágiles).
 - En formación, para describir el conocimiento de una cierta área clave, buena práctica, disciplina, etc.
- Proceso para Despliegue (Delivery Process): Describe una aproximación completa e integrada para realizar un tipo específico de proyecto, abarcando un ciclo de vida completo de desarrollo o mantenimiento. Sirven como plantillas para planificar y ejecutar los proyectos. En un Proceso de Despliegue se ensamblan Patrones de Proceso y Elementos en Uso (tareas, roles y productos de trabajo en uso). Ejemplos: “proceso unificado de Rational (RUP)”, “programación extrema (XP)”, “mantema, metrica 3”, etc.

Todos los procesos, independiente del tipo que sean, tienen en común las siguientes propiedades: name (nombre), presentation name (nombre de presentación), brief description (descripción breve), external ID (ID externo), purpose (objetivo), main description (descripción principal), scope (ámbito), usage notes (notas de utilización), alternatives (alternativas), how to staff (cómo proveer de personal), key considerations (factores clave). Adicionalmente, un proceso para despliegue tiene éstas otras propiedades: scale (escala), Project characteristics (características del proyecto), risk level (nivel de riesgo), estimating techniques (técnicas de estimación 13), project member expertise (especialidad de miembros del proyecto), type of contract (tipo de contrato).

Los patrones de proceso pueden ser incorporados como elementos de desglose a una WBS. Esto implica la reutilización automática de todos sus elementos y estructura (actividades, tareas, roles, productos de trabajo, WBS, etc.).

2.4 Reutilización y Variabilidad

La organización en plugins (Plug-in) permite que se puedan reutilizar los elementos de contenido y los procesos definidos en una Biblioteca (Library). Dicha reutilización se puede realizar de dos maneras:

- a) Al crear un plugin nuevo se puede referenciar a otros plugins.
- b) Usar de forma directa el contenido de un plugin desde otro plugin diferente.

A veces nos interesa reutilizar el contenido de un plugin con ciertas modificaciones. Para ello existe el mecanismo de variabilidad (Variability), que permite modificar elementos de método o de proceso sin modificar directamente el original. La variabilidad de un elemento de método o de proceso permite definir diferencias (adiciones, cambios, omisiones) con el elemento original. Dichas diferencias afectan a las propiedades, es decir, a los atributos y a las asociaciones con otros elementos.

2.4.1 Reutilización y Variabilidad de Elementos de Proceso

Una manera de definir procesos es aplicar una estrategia bottom-up desde lo pequeño hacia lo grande. Al seguirla, primero se definen patrones de proceso sencillos, que no incluyen jerarquía de actividades, sino simplemente descriptores de tarea, de rol y de producto). Después se crean patrones más complejos en los que se reutilizan los patrones más sencillos definidos anteriormente. De esta forma se van componiendo los procesos reutilizando bloques de proceso cada vez más complejos, hasta llegar a los procesos para despliegue. En general, los elementos de proceso de tipo actividad (fase, iteración, actividad, patrón de proceso) se pueden reutilizar en otros elementos de proceso más grandes. En este caso reciben el nombre genérico de Actividad en Uso (Activity Use). Una Actividad en Uso en SPEM define la habilidad para reutilizar las estructuras definidas en los elementos de desglose de una actividad en una segunda actividad, sin necesidad de copiar físicamente dichas estructuras. Por tanto, se trata de una forma para heredar dinámicamente dichas estructuras desde la actividad referenciada.

2.4.2 Configuraciones de Método

Una Configuración de Método (Method Configuration) es una selección de contenidos de los plugins de una Biblioteca de Métodos (Method Library) de forma que se limita el espacio visual de la biblioteca al subconjunto seleccionado. En consecuencia, es el mecanismo ofrecido en SPEM para poder tener diferentes **vistas** de una misma biblioteca de plugins o de un único plugin. La definición de una configuración puede estar basada en las definiciones de otras configuraciones. Por ejemplo, una configuración A podría definirse como el superconjunto de las configuraciones B, C y D más ciertos plugins y paquetes adicionales. En este caso, se dice que B, C y D son las configuraciones base de A. Si alguna de estas configuraciones base cambian, los cambios serán automáticamente válidos también para A, reduciendo de esta manera el esfuerzo de mantenimiento.

Una configuración de método define un subconjunto lógico de una biblioteca de métodos mediante la selección (filtro) de los paquetes deseados, tanto de contenido de método como de procesos.

Una actividad representando un proceso puede ser válida para diferentes configuraciones de método, pero en cada configuración se incluye o excluye contenido particular para situaciones específicas. Por ejemplo, un proceso definido para desarrollar esquemas de bases de datos puede tener dos configuraciones, una con la vista particular para SGBD (Sistemas de Gestión de Bases de Datos) relacionales y otra para el caso de sistemas orientados a objetos. Por tanto, este mecanismo ofrece una alternativa en la creación de procesos, ya que puede definirse un proceso general que incluya contenidos para varios tipos de proyectos más específicos. Mediante el uso de configuraciones se seleccionan los contenidos de cada uno de esos tipos de proyectos específicos. Crear vistas para los diversos tipos de proyecto es sencillo, ya que basta con seleccionar la configuración adecuada.

5. El Editor EPF Composer

Eclipse Process Framework Composer, en adelante EPFC, es una herramienta gratuita, desarrollada dentro del entorno ECLIPSE, que sirve para editar fragmentos de método, procesos o metodologías, y generar automáticamente la documentación adecuada en formato para la web. Dichos fragmentos se almacenan en formato XMI y, al estar basados en el estándar SPEM 2, pueden ser reutilizados por cada vez más herramientas CASE. En realidad, EPFC utiliza la "Unified Method Architecture" (UMA), que a su vez está basada en el metamodelo SPEM 2.

En suma, EPFC es un editor de procesos SPEM 2, que incluye opciones adicionales para publicar de forma automáticamente sitios web. Además de la herramienta Composer, el Eclipse Process Framework también incluye las implementaciones públicas, abiertas y gratuitas de varias metodologías: OpenUP (proceso unificado abierto), SCRUM y XP.

5.1 Poblando el Contenido de Método

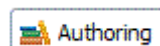
Los elementos de contenido se pueden organizar y jerarquizar mediante la creación de paquetes de contenido. En cada paquete de contenido se pueden definir los elementos de contenido de método básicos: roles, tareas, productos de trabajo y guías (instrucciones).

5.1.1 Creando un Paquete de Contenido de Método (Content Package)

El objetivo de este ejercicio es crear un nuevo Paquete de Contenido (Content Package) dentro de un Método Plugin (Method Plug-in.)

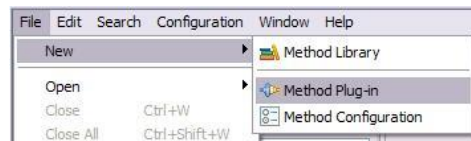
5.1.1.1 Crear un Nuevo Método Plug-in

Ubicarse en la perspectiva **Authoring**.

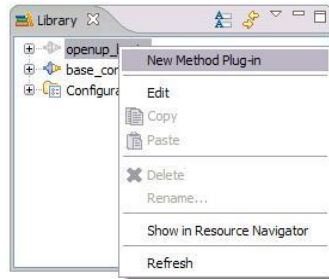


1. Hay dos formas de crear un método plug-in

- **File-> New-> Method Plug-in.**

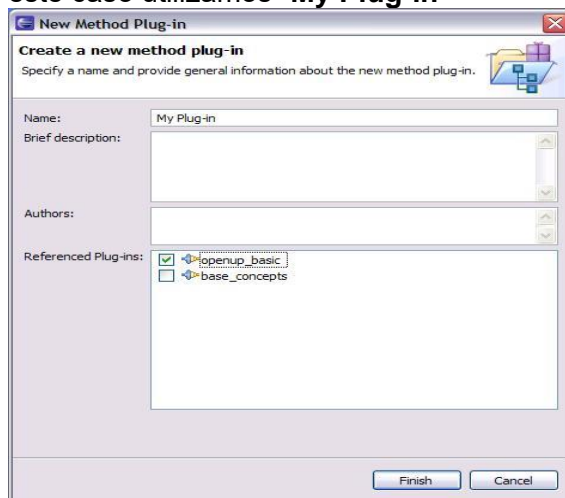


- Click derecho en un plug-in existente del árbol de navegación y seleccionar **New Method Plug-in**.



2. Aparece el wizard del **New Method Plug-in**

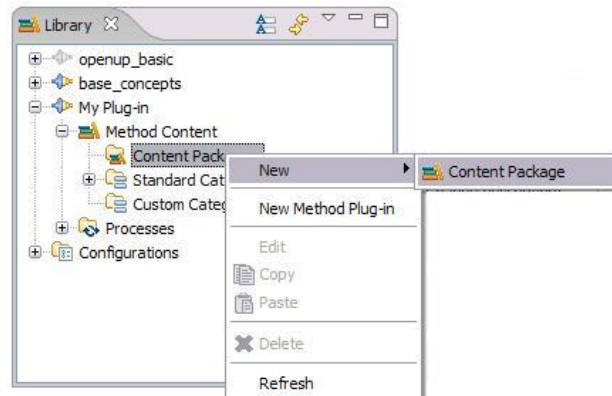
3. En wizard del **New Method Plug-in**, suministrar el nombre para el plug-in, para este caso utilizamos **“My Plug-in”**



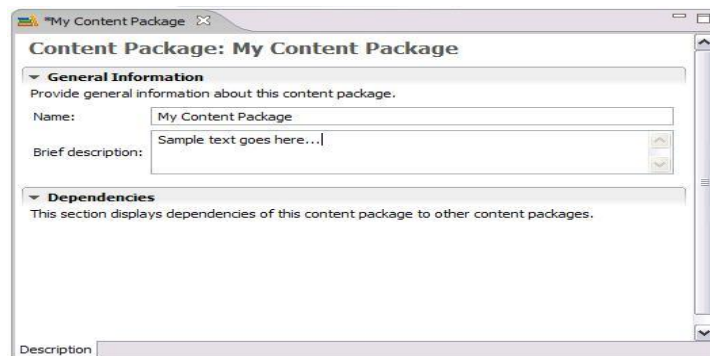
4. Seleccionar cualquier plug-in, si él plug-in nuevo va a extender de un plugin existente, esto significa que se puede seleccionar contenido desde el plugin existente para crear tu propio plug-in.

5.1.1.2 Crear un Paquete de Método contenido (method content package) en el nuevo Plugin.

1. En el Library view panel, en el árbol de navegación expandir el nodo del nuevo plugin en el símbolo +, luego Click derecho en **Content Packages->New->Content Package**
El editor del **Content Package** es desplegado.

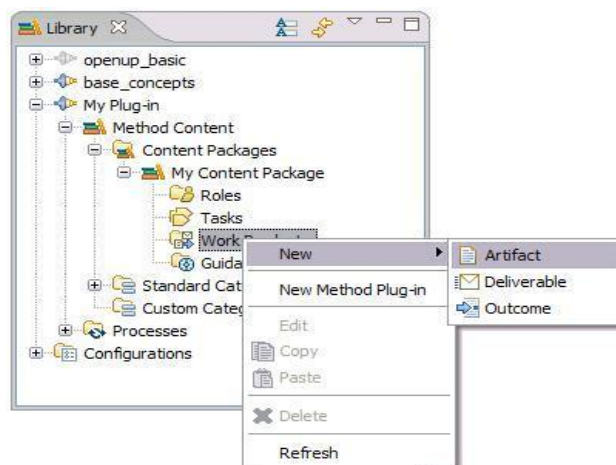


2. Nombre del paquete "My Content Package" y escriba algo en el campo **Brief description** .



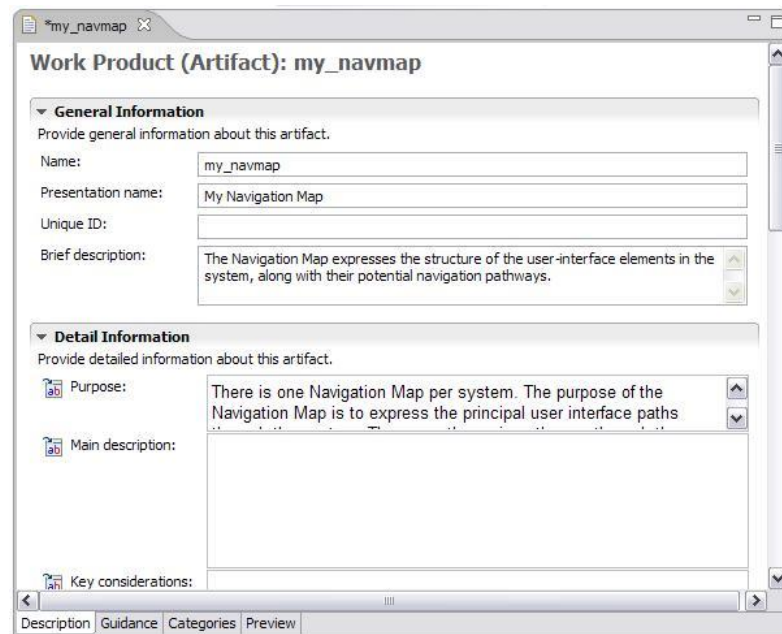
5.1.1.3 Crear un producto de trabajo (Create a Work Product)

1. Localizar el **Work Products**. Click derecho en el icono **Work Products->New->Artifact**.



2. En el campo Name ingresar "my_navmap".
3. En el campo Name **Presentation Name** ingresar "My Navigation Map"

4. En el campo **Description**, Ingresar los siguiente:
“The Navigation Map expresses the structure of the user-interface elements in the system, along with their potential navigation pathways”.
5. En el campo **Purpose**, Ingresar los siguiente:
“There is one Navigation Map per system. The purpose of the Navigation Map is to express the principal user interface paths through the system. These are the main pathways through the screens of the system and not necessarily all of the possible paths. It can be thought of as a road map of the system's user interface. The Navigation Map makes it easy to see how many "clicks" it will take a user to get to a specific screen”.



6. Click en la etiqueta **Preview** para ver el nuevo artefacto.
7. Guardar el Artefacto

5.1.1.3.1 Crear Roles y Tareas

Para crear Roles y Tareas se siguen los mismos pasos, pero en cada caso la ubicación cambia para un nuevo rol sería: en **My Content Package** ir a la carpeta **Roles** click derecho **New->Role**. Para una nueva tarea sería: en **My Content Package** ir a la carpeta **Task** -> click derecho **New->Task**.

5.2 Categorización de Elementos de Método

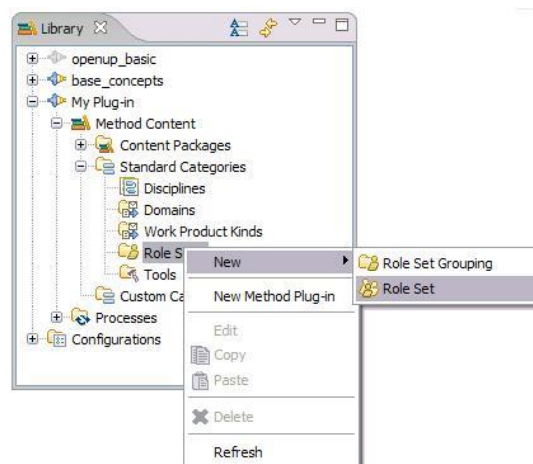
Los elementos del contenido de método se pueden categorizar mediante su asociación con categorías estándar (disciplinas, dominios, clases de productos de trabajo,

conjuntos de roles y herramientas), para lo cual existe una pestaña en EPFC. Además, los elementos del contenido de método (y también los elementos en uso en procesos) se pueden organizar alternativamente mediante categorías personalizadas. Las categorías personalizadas también se crean en EPFC a través de la pestaña correspondiente en la vista *Biblioteca*. Estas categorías personalizadas pueden contener cualquier tipo de elemento, proceso, categoría, etc. Son útiles para organizar de forma lógica elementos que no se puedan organizar mediante las categorías estándar, como por ejemplo, agrupar todas las plantillas de productos, agrupar todas las guías que sean directrices, etc. Las categorías personalizadas permiten organizar el contenido de acuerdo al esquema que queramos, sirviendo como medio eficaz para organizar el contenido para su publicación. El contenido de una categoría personalizada se especifica en la pestaña *Asignar* del editor de la categoría.

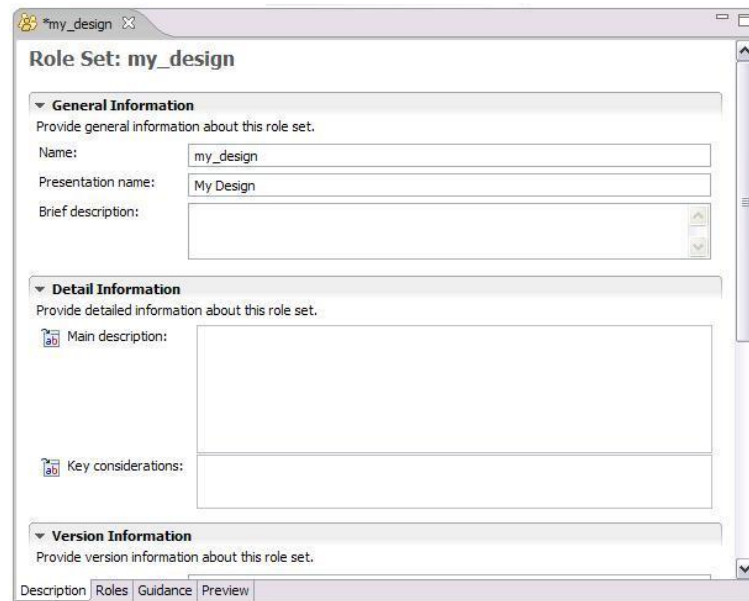
5.2.1 Crear una categoría Estándar (Standard Category)

En este ejemplo anexamos a un Role Set con roles, Disciplinas con tareas, y Dominios con productos de trabajo.

1. Localizar la carpeta **Role Set** bajo **Standard Cateories** en **My Plug-in**.
2. Click derecho en **Role Sets-> New -> Role Set**.

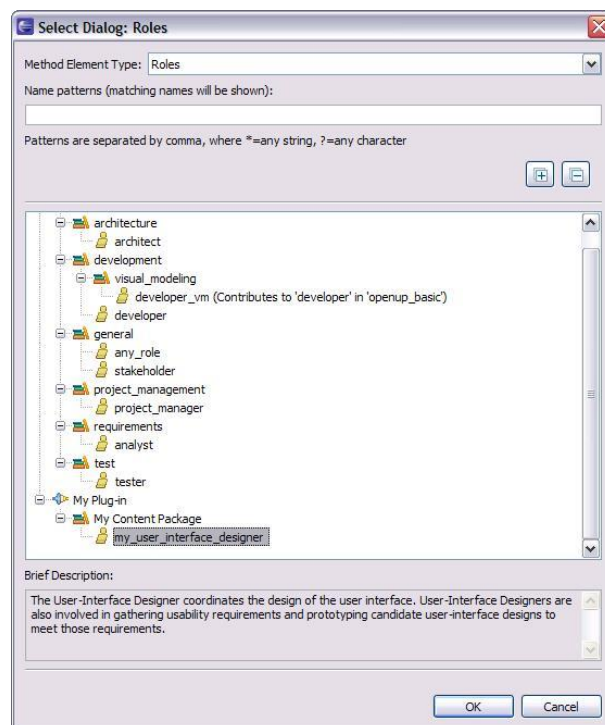


3. Crear un nuevo **Role Set** llamado "my_design" con el nombre de presentación de "My Design".



4. Adicionar un rol, en este caso **my_user_interface_designer** al Rol Set **my_design**. Hay dos formas de hacer esto:

- En el editor Standard Category doble click en **my_design**, abrir la pestaña **Roles**, y adicionar el rol al Role set



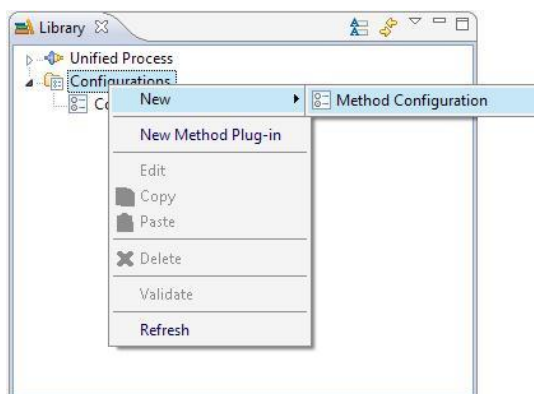
- En el editor del Rol doble click en **my_user_interface_designer**, abrir la pestaña **Categories->Add** seleccionar la categoría **my_design**.

5.3.1 Crear una Configuración

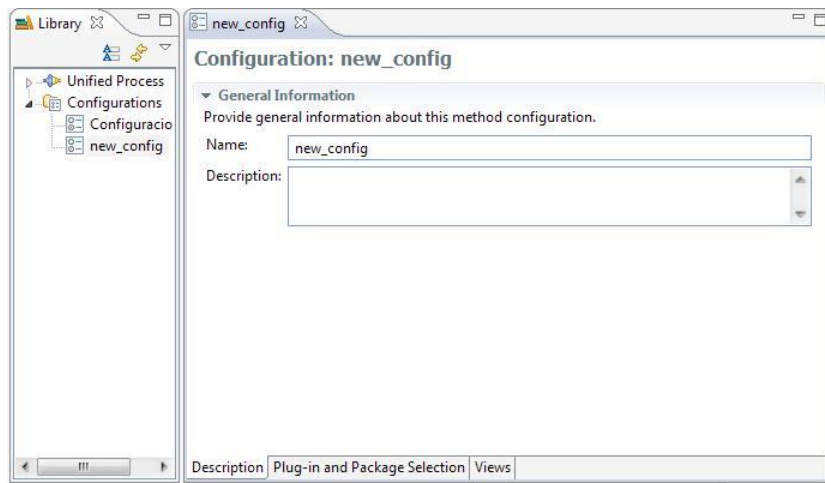
Una configuración de método es una selección de contenidos de los plug-ins de una librería, de forma que se limita la vista de la librería al subconjunto seleccionado. En una configuración se seleccionan los paquetes y procesos que queremos incluir. La selección de elementos incluidos se puede refinar añadiendo o sustrayendo de la configuración todos los elementos categorizados en una misma categoría.

Las configuraciones de método permiten definir qué elementos aparecerán en la web publicada y cuáles no. También permiten seleccionar los elementos que son visibles para los procesos creados. Ofrecen una alternativa en la creación de procesos, ya que puede definirse un proceso general que incluya contenidos para varios tipos de procesos más específicos. Mediante el uso de configuraciones se seleccionan los contenidos de cada uno de esos tipos de procesos más específicos. Crear los diversos tipos de proyecto es sencillo seleccionando la configuración adecuada para el proceso.

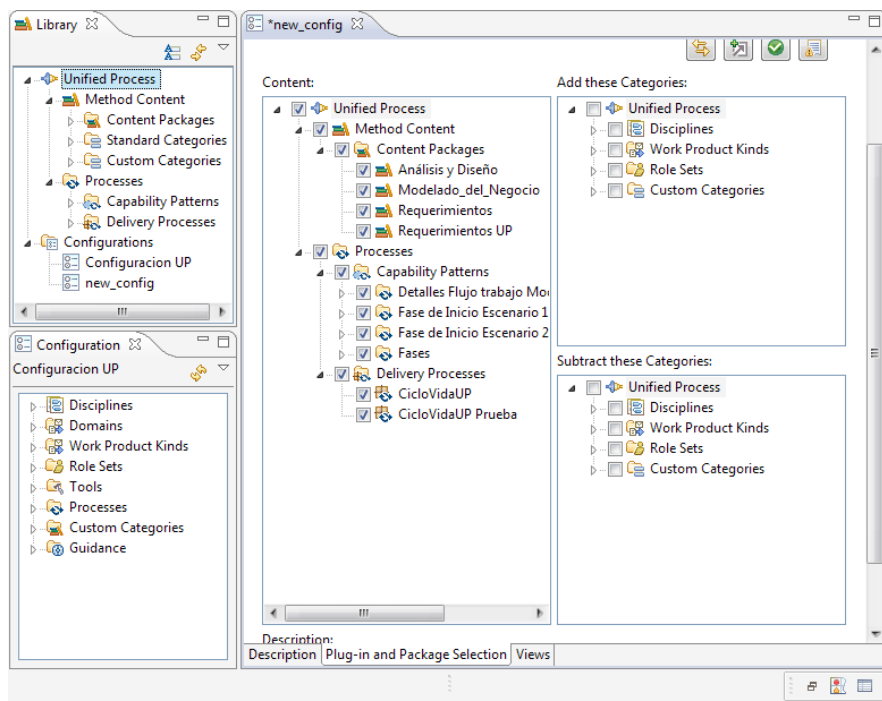
1. Click derecho en el paquete de configuraciones y seleccionar New -> Method Configuration. La configuración nueva es adicionada a la lista de configuraciones y el editor se abre



2. Cambiar el nombre de la nueva configuración.



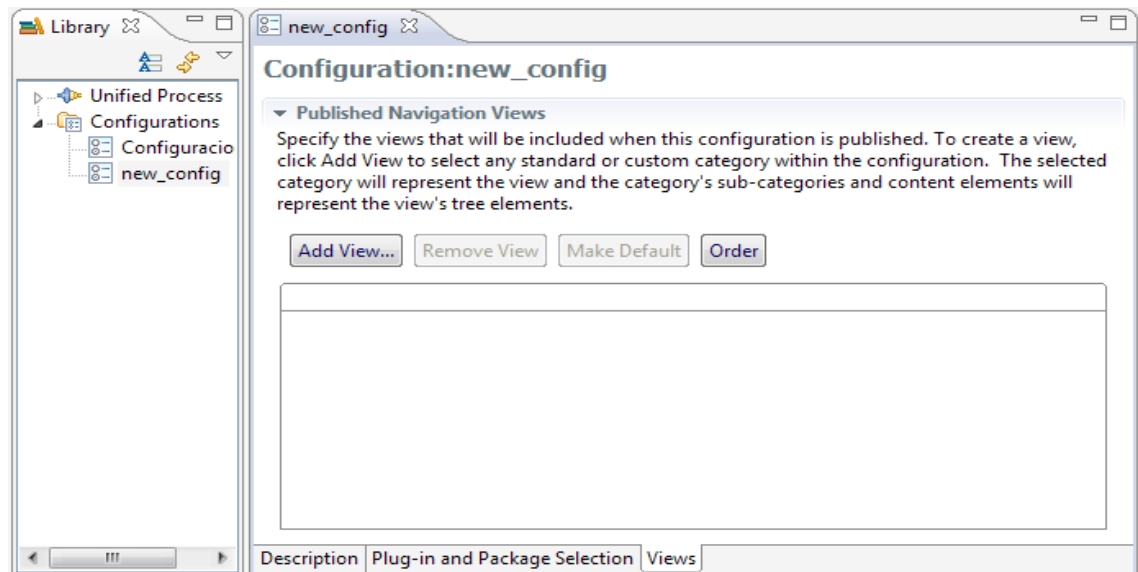
3. Especificar la configuración, click en la etiqueta "Plug-in and Package Selection" del editor de la configuración. En esta form se despliega una lista de todos los plug-ins y de cada plugin con todos los paquetes de contenido así como también el proceso. Usar las cajas de verificación para adicionar o remover plug-ins, paquetes, y procesos para la configuración.



4. Definir las vistas de navegación. Una vista de navegación es un árbol de navegación para una configuración publicada en HTML. Cada configuración puede tener varias vistas que son desplegadas en etiquetas de un árbol de navegación. Una custom Category es una colección de elementos clasificados y definidos por el usuario, los cuales pueden tener sus propias subcategorías. Esta estructura es la que define la organización del árbol de navegación.

Además, para definir una vista de navegación se selecciona un custom category y todas las subcategorías y subelementos para definir la estructura del árbol de navegación que será desplegada por la vista.

5. Para adicionar vistas de navegación para a la configuración click en la etiqueta "Views" en el editor de la configuración. Usar los botones "Add View..." y "Remove View" para seleccionar las custom categories que se necesiten adicionar o remover de esta vista.



5.4 Creando Procesos

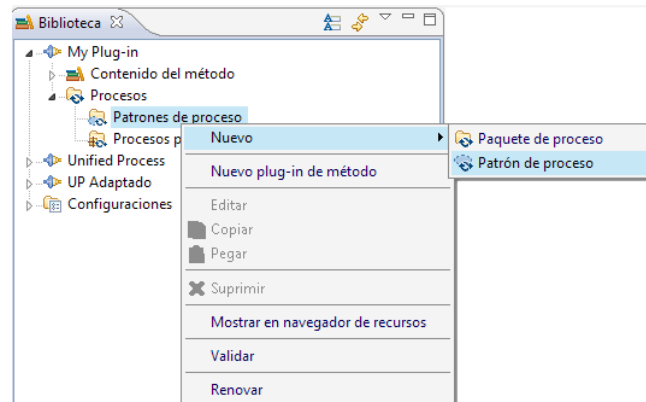
Una vez se han definido los elementos de contenido de método, éstos se pueden reutilizar para crear procesos. Cabe señalar que no es necesario haber creado elementos de contenido para reutilizarlos en la creación de procesos, pues se pueden definir directamente en un proceso, pero esta opción dificulta mucho la reutilización y actualización posterior de dichos elementos y de sus procesos.

Con EPFC se pueden crear los dos tipos de procesos de SPEM 2: - Patrón de proceso: Son los patrones de proceso de SPEM, que describen una agrupación reutilizable de tareas o actividades. Proceso para despliegue: Describen un enfoque completo e integrado para una metodología completa, un tipo de ciclo de vida, un tipo específico de proyecto, etc. Al igual que los elementos de contenido, los patrones de proceso y los procesos para despliegue se pueden organizar y jerarquizar mediante paquetes de proceso.

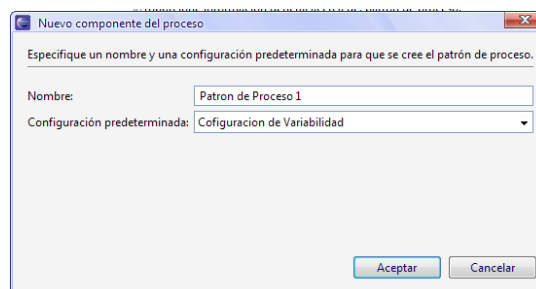
La creación de procesos y de paquetes de proceso se hace en la vista *Biblioteca*. En la pestaña *Descripción* del editor se especifican los atributos de los procesos (descripción breve, descripción principal, objetivos, etc.)

5.4.1 Crear un Patrón de Procesos

1. Ir a la vista biblioteca, en Patrones de proceso **click derecho ->Nuevo-> Patrón de Procesos**

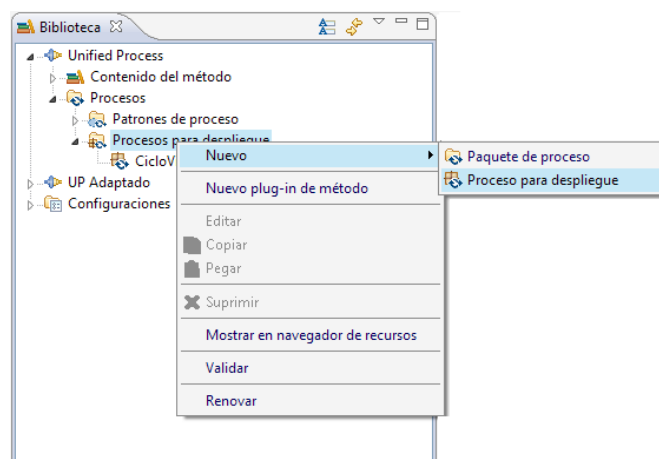


2. Especificar el Nombre del Patrón de Proceso.



5.4.2 Crear un Proceso para despliegue.

1. Ir a la vista biblioteca, En Procesos para Despliegue -> click derecho-> Proceso para despliegue, luego repetir el paso 2 de la sección anterior.

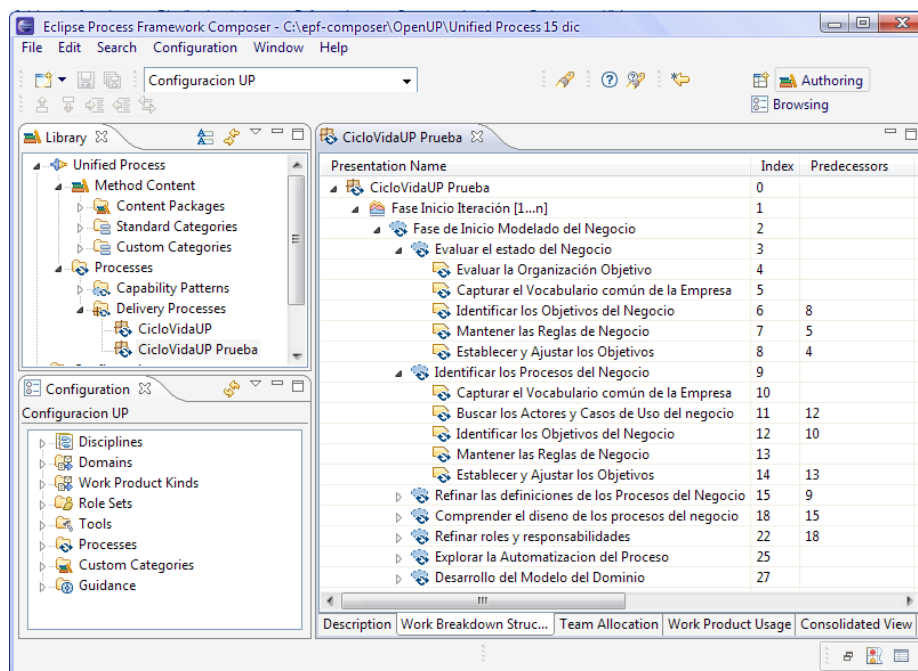


5.4.3 Estructuras de Desglose

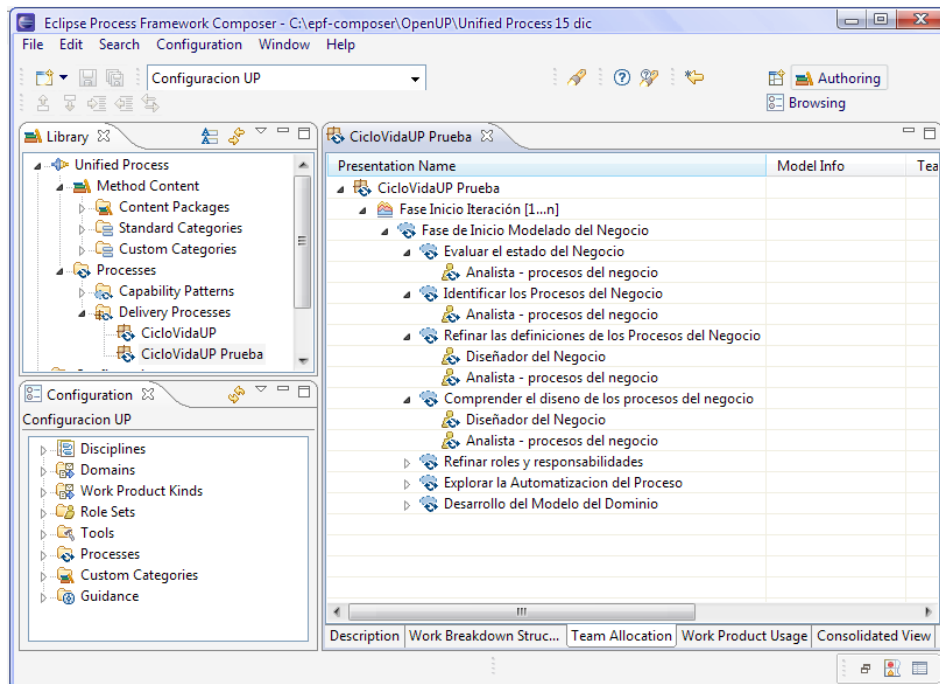
Para representar los dos tipos de procesos antes comentados (es decir, las llamadas actividades de forma general en SPEM 2), se utilizan estructuras de descomposición, mediante las cuales se representa la jerarquía de actividades (incluyendo fases e iteraciones) e hitos. Además, permiten ordenar la secuencia del trabajo (flujo de trabajo), indicando relaciones de precedencia entre los elementos de la estructura de descomposición.

Para ver y editar desde diferentes puntos de vista la estructura de desglose de un proceso/metodología completo o de un fragmento, EPFC incluye las siguientes pestañas:

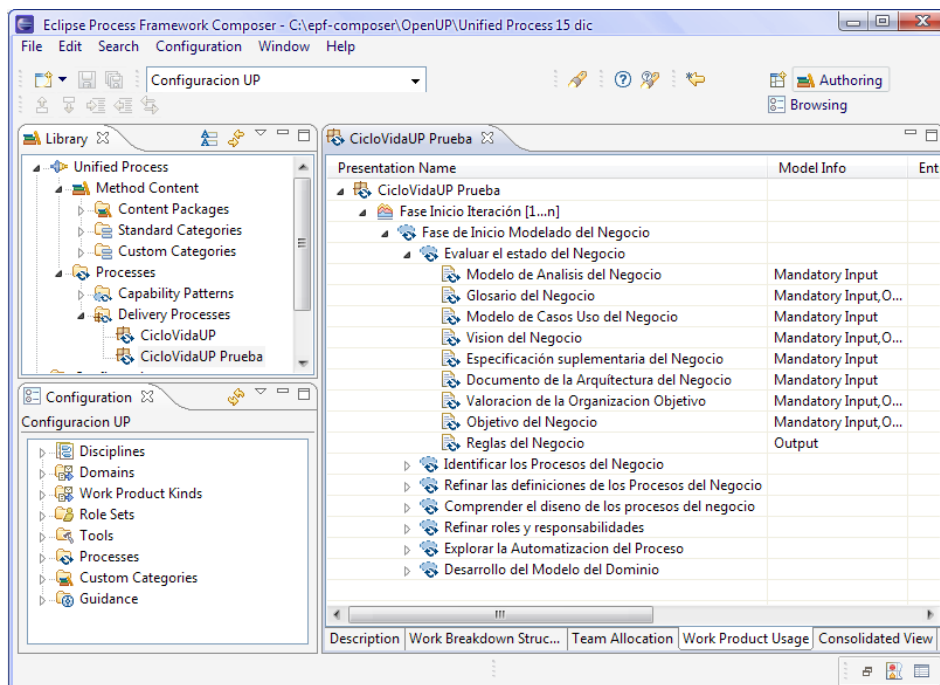
- a) Estructura de Desglose de Trabajo (Work Breakdown Structure): Corresponde al conocido WBS de gestión de proyectos. En EPFC es una tabla que contiene la descomposición jerárquica de los elementos que definen el trabajo a realizar: actividades (patrones de proceso, fases, iteraciones), hitos, tareas, etc.



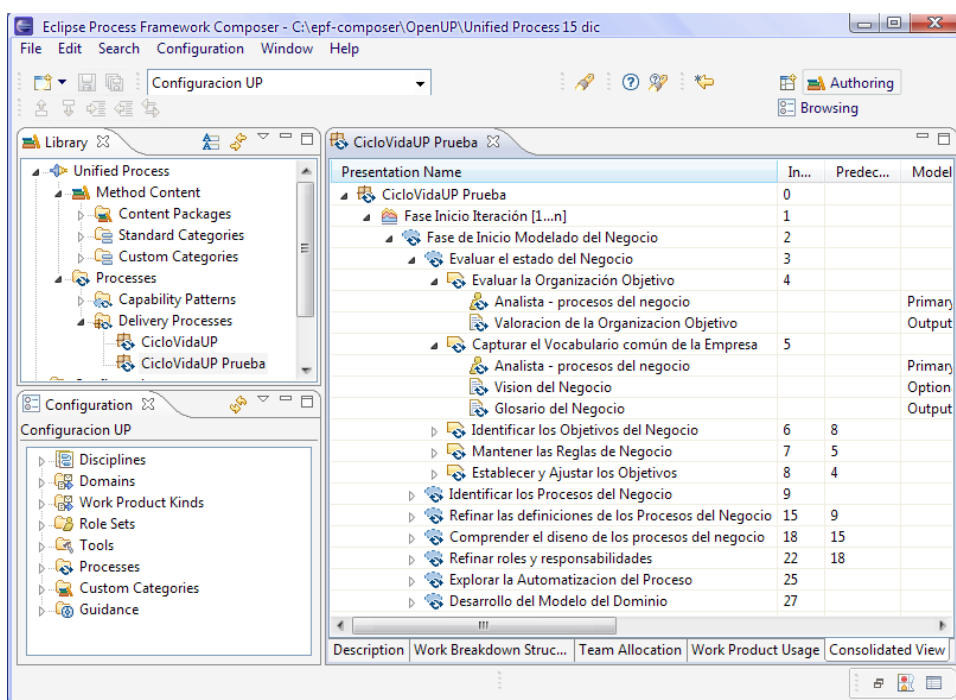
- b) Asignación de Equipos (Team Allocation): Muestra una jerarquía similar a la anterior, pero mostrando para cada actividad (cada nivel) la lista de roles que participan.



c) Utilización de Productos de Trabajo (Work Product Usage): Esta perspectiva se centra en los productos de trabajo de forma similar a como la anterior se centra en los roles.



d) Vista Consolidada (Consolidated View): Combina las informaciones que se muestran en las tres vistas anteriores. Solo existe con fines informativos, es decir, no permite edición.



Para desarrollar la estructura de desglose de un proceso o metodología se deben crear los elementos de desglose que lo forman: actividades (y sus variantes, fases e iteraciones), hitos (también llamados objetivos) y, en el nivel más bajo de la jerarquía, tareas, roles o productos de trabajo. Éstos últimos dependerán de la perspectiva de desglose (pestaña) en la que estemos trabajando.

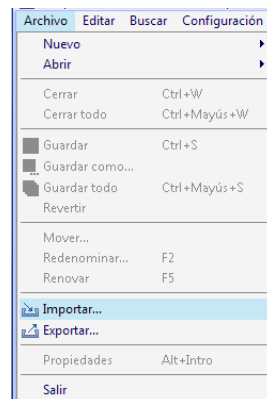
Los elementos de desglose se pueden crear directamente en la estructura de desglose o se pueden reutilizar del contenido de método (tareas, roles y productos) o de otros patrones de proceso (actividades).

5.5 Importar Plugins

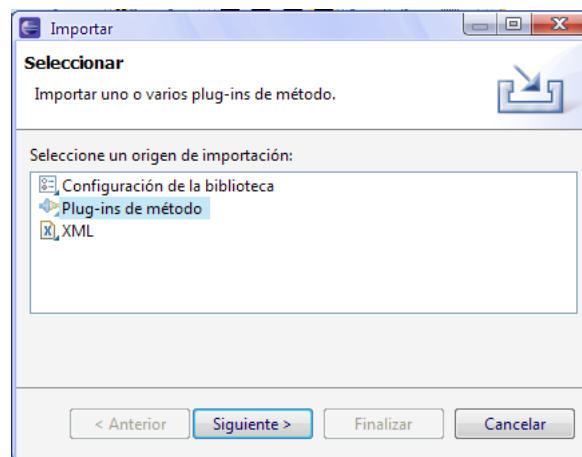
Este ejercicio tiene como objetivo mostrar como importar un Plugin, en este caso el Plugin importado se llama “Unified Process”.

Para importar un Plugin se realizan los siguientes pasos

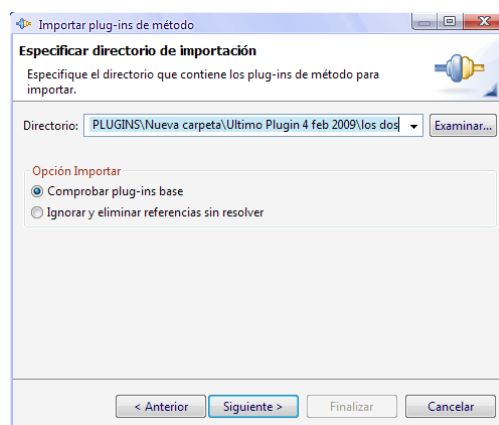
1. Ir al **Archivo->Importar, ver figura.**



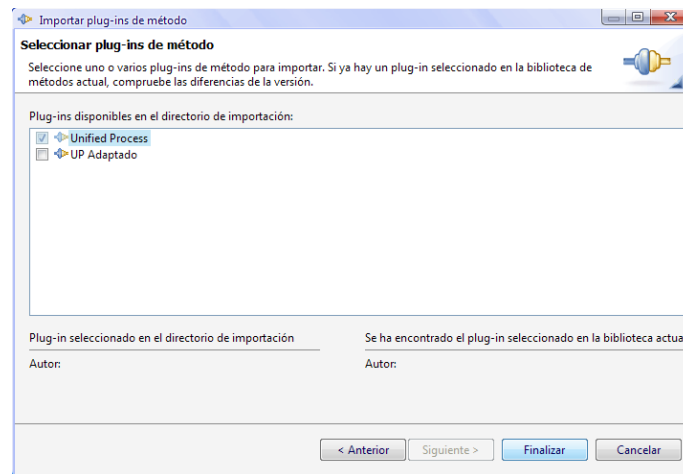
2. Seleccionar el tipo de importaciones, para este caso escogemos Plug –ins de Método. Ver figura.



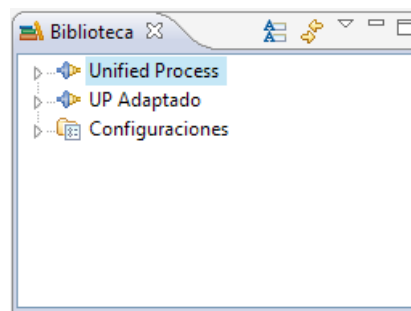
3. Especificar el directorio de Importación, Buscar el directorio donde se encuentra el Plugin que será importado.



4. Una vez elegido el directorio, se mostrarán los Plugins que están disponibles para importar, para este caso elegir el Plugin "Unified Process", ver figura.



5. Verificar que el Plugin se Importo. Ir a la vista de Biblioteca y confirmar que el Plugin “Unified Process” se cargo en el árbol de la biblioteca, ver figura.



D.1.2. Descripción de UP

1. ENTENDIMIENTO DEL PROCESO UNIFICADO

1.1 Que es el Proceso Unificado o UP?

El proceso unificado en adelante UP (Unified Process), es un proceso de ingeniería del software, provee un método disciplinado para asignar tareas y responsabilidades dentro de una organización de desarrollo de software, su objetivo es asegurar la producción de software de alta calidad que reúna las necesidades de los usuarios finales, dentro de una planificación y presupuesto predecible **¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia..**

Un proceso de software se define como un: "Conjunto de actividades, métodos, prácticas y transformaciones que las personas usan para desarrollar y mantener software y sus productos asociados (p.ej., planes, especificaciones, diseños y pruebas) **¡Error! No se encuentra el origen de la referencia..**

El UP es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, tipos de organizaciones, niveles de aptitud y tamaños de proyectos, el cual debe ser adaptado para específicas organizaciones o proyectos.

1.2 Mejores Prácticas.

UP identifica 6 *best practices* con las que define una forma efectiva de trabajar para los equipos de desarrollo de software

1.2.1 Desarrollo de software iterativo

Debido a la complejidad de los sistemas software, no es posible definir totalmente el problema, diseñar una solución y luego construir el software para probar el producto final. Un método iterativo es necesario, que permita un entendimiento y solución progresiva y efectiva del problema, a través de múltiples iteraciones con sucesivos refinamientos. UP soporta un método iterativo de desarrollo que direcciona ítems de alto riesgo en cada fase del ciclo de vida, esta característica permite reducir significativamente el riesgo. Un método iterativo hace más fácil adecuarse a los cambios estratégicos en los requerimientos, características o en la planificación **¡Error! No se encuentra el origen de la referencia..**

1.2.2 Gestión de Requisitos.

UP brinda una guía para encontrar, organizar, documentar, y seguir los cambios de los requisitos funcionales y restricciones. Utiliza una notación de Casos de Uso y escenarios que han probado ser una excelente forma de capturar requerimientos funcionales que conducen el diseño, implementación y pruebas del software, haciendo probable que el sistema final cumpla a cabalidad con las necesidades del usuario final. Los Casos de Uso y escenarios proveen un hilo coherente de trazabilidad a través del desarrollo y la entrega del sistema **¡Error! No se encuentra el origen de la referencia..**

1.2.3 Desarrollo basado en componentes

La creación de sistemas intensivos en software requiere dividir el sistema en componentes con interfaces bien definidas, que posteriormente serán ensamblados para generar el sistema. Esta característica en un proceso de desarrollo permite que el sistema se vaya creando a medida que se obtienen o se desarrollan sus componentes.

1.2.4 Modelado visual

UML (Unified Modeling Language) es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema software. Utilizar herramientas de modelado visual facilita la gestión de dichos modelos, permitiendo ocultar o exponer detalles cuando sea necesario. El modelado visual también ayuda a mantener la consistencia entre los artefactos del sistema: requisitos, diseños e implementaciones. En resumen, el modelado visual ayuda a mejorar la capacidad del equipo para gestionar la complejidad del software, donde UML es la base del éxito **¡Error! No se encuentra el origen de la referencia..**

1.2.5 Verificación de la calidad del software.

La calidad debe ser revisada con respecto a los requerimientos en base a la fiabilidad, funcionalidad y desempeño del sistema. UP brinda mecanismos para planear, diseñar, implementar, ejecutar y evaluar este tipo de pruebas

Es importante que la calidad de todos los artefactos se evalúe en varios puntos durante el proceso de desarrollo, especialmente al final de cada iteración. En esta verificación las pruebas juegan un papel fundamental y se integran a lo largo de todo el proceso. Para todos los artefactos no ejecutables las revisiones e inspecciones también deben ser continuas.

1.2.6 Gestión de los cambios

Asegurarse de que cada cambio es admisible, y ser capaz de seguir los cambios, es esencial en un entorno donde el cambio es inevitable **¡Error! No se encuentra el origen de la referencia..** UP describe como controlar, rastrear y monitorear los cambios para permitir un desarrollo iterativo exitoso. El cambio es un factor de riesgo crítico en los proyectos de software. Los artefactos software cambian no sólo debido a acciones de mantenimiento posteriores a la entrega del producto, sino que durante el proceso de desarrollo, especialmente importantes por su posible impacto son los cambios en los requisitos.

1.3 Dos dimensiones.

El proceso puede ser descrito en dos dimensiones o ejes **¡Error! No se encuentra el origen de la referencia.:**

Eje horizontal: Representa el tiempo y es considerado el eje de los aspectos dinámicos del proceso. Indica las características del ciclo de vida del proceso expresado en términos de fases, iteraciones e hitos. Se puede observar en la Figura 1 que UP consta de cuatro fases: Inicio, Elaboración, Construcción y Transición. Cada fase se subdivide a la vez en iteraciones. Eje vertical: Representa los aspectos estáticos del proceso. Describe el proceso en términos de componentes de proceso, disciplinas, flujos de trabajo, actividades, artefactos y roles

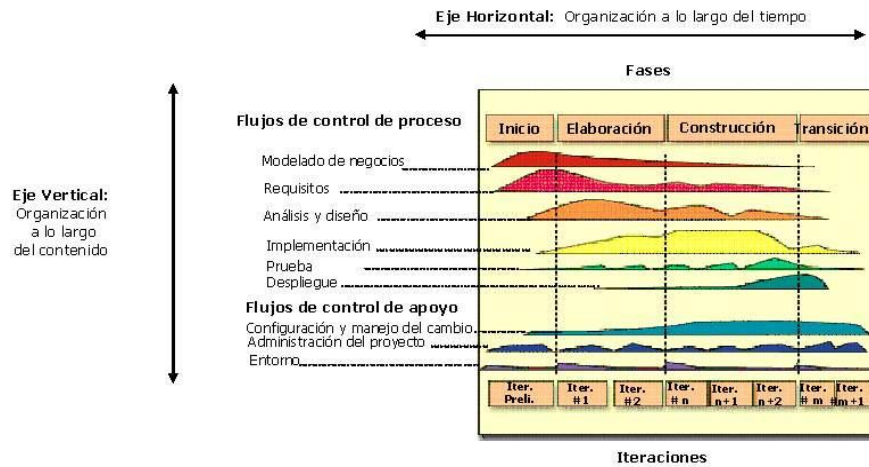


Figura 1: Estructura de UP

1.3.1 Eje horizontal Fases e Iteraciones – La Dimensión dinámica del proceso.

Eje horizontal es la organización dinámica del proceso a lo largo del tiempo. UP se repite a lo largo de una serie de ciclos que constituyen la vida de un producto. Cada ciclo concluye con una generación del producto para los clientes. Cada ciclo consta de cuatro fases consecutivas **¡Error! No se encuentra el origen de la referencia.:**

- Fase Inicio.
- Fase Elaboración.
- Fase Construcción.
- Fase Transición.

Cada fase se concluye con un hito bien definido. Los hitos para cada una de las fases son: Inicio - *Lifecycle Objectives*, Elaboración - *Lifecycle Architecture*, Construcción - *Initial Operational Capability*, Transición - Product Release. Las fases y sus respectivos hitos se ilustran en la Figura 2.

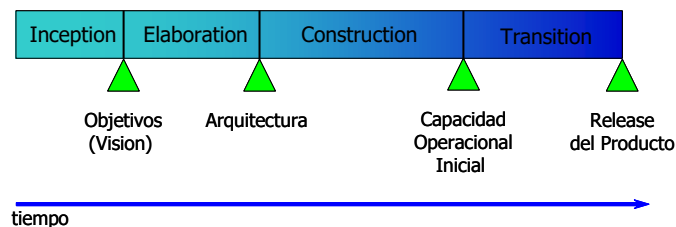


Figura 2: Fases e hitos en UP

1.3.1.1 Fase de Inicio.

Durante la fase de inicio se define el modelo del negocio y el alcance del proyecto. Para cumplir con esto se deben identificar todas las entidades externas con las cuales el sistema interactúa (actores) y se define la naturaleza de la interacción a un alto

nivel. Esto involucra identificar todos los casos de uso y describir los más importantes. Se desarrolla, un plan de negocio para determinar que recursos deben ser asignados al proyecto y un plan que muestre las fechas importantes de los principales hitos.
¡Error! No se encuentra el origen de la referencia..

Los resultados de la fase de inicio son **¡Error! No se encuentra el origen de la referencia.:**

- Un documento de visión: Una visión general de los requerimientos del proyecto, características clave y restricciones principales.
- Modelo inicial de Casos de Uso (10-20% terminado).
- Un glosario inicial: Terminología clave del dominio.
- El caso de negocio.
- Lista de riesgos y plan de contingencia.
- Plan del proyecto, mostrando fases e iteraciones.
- Modelo de negocio, si es necesario
- Prototipos exploratorios para probar conceptos o la arquitectura candidata.

Al final de la fase de inicio se debe alcanzar el Lifecycle Objectives Milestone

1.3.1.2 Fase de Elaboración.

El propósito de la fase de elaboración es analizar el dominio del problema, establecer los cimientos de la arquitectura, desarrollar el plan del proyecto y eliminar los riesgos mayores del proyecto. Para cumplir con estos objetivos, se debe tener una buena visualización de lo que será el sistema. Para la toma de decisiones arquitecturales se debe tener una comprensión completa del sistema: su alcance, principales requerimientos funcionales y no funcionales.

En esta fase se construye un prototipo de la arquitectura, que debe evolucionar en iteraciones sucesivas hasta convertirse en el sistema final. Este prototipo debe contener los Casos de Uso críticos identificados en la fase de inicio. También debe demostrarse que se han evitado los riesgos más graves.

Al final de la fase de elaboración la arquitectura de referencia debe demostrar que soportará las funcionalidades claves del sistema y exhibir un buen comportamiento en términos de desempeño, escalabilidad y costo.

Al terminar deben obtenerse los siguientes resultados **¡Error! No se encuentra el origen de la referencia.:**

- Un modelo de Casos de Uso (completo al menos hasta el 80%): todos los casos y actores identificados, la mayoría de los casos de uso han de ser desarrollados.
- Requisitos adicionales que capturan los requisitos no funcionales y cualquier requisito no asociado con un Caso de Uso específico.
- Descripción de la arquitectura software.
- Un prototipo ejecutable de la arquitectura.
- Lista de riesgos y caso de negocio revisados.
- Plan de desarrollo para el proyecto.
- Un caso de desarrollo actualizado especificando el proceso a seguir.
- Un manual de usuario preliminar (opcional).

Al final de la fase de elaboración se debe alcanzar el Lifecycle Architecture Milestone.

1.3.1.3 Fase de Construcción.

La finalidad principal de esta fase es alcanzar la capacidad operacional del producto de forma incremental a través de las sucesivas iteraciones. Durante esta fase todos los componentes, características y requisitos deben ser implementados, integrados y probados en su totalidad, obteniendo una versión aceptable del producto.

Los resultados de la fase de construcción deben ser **¡Error! No se encuentra el origen de la referencia.:**

- Modelos Completos (Casos de Uso, Análisis, Diseño, Despliegue e Implementación)
- Arquitectura íntegra (mantenida y mínimamente actualizada)
- Riesgos Presentados Mitigados
- Plan del Proyecto para la fase de Transición.
- Manual Inicial de Usuario (con suficiente detalle)
- Prototipo Operacional – beta
- Caso del Negocio Actualizado

Al final de la fase de construcción se debe alcanzar el Initial Operational Capability Milestone.

1.3.1.4 Fase de Transición.

La finalidad de la fase de transición es poner el producto en manos de los usuarios finales, para lo que se requiere desarrollar nuevas versiones actualizadas del producto, completar la documentación, entrenar al usuario en el manejo del producto, y en general tareas relacionadas con el ajuste, configuración, instalación y facilidad de uso del producto.

Los resultados de la fase de transición son **¡Error! No se encuentra el origen de la referencia.:**

- Prototipo Operacional
- Documentos Legales
- Caso del Negocio Completo
- Línea de Base del Producto completa y corregida que incluye todos los modelos del sistema
- Descripción de la Arquitectura completa y corregida
- Las iteraciones de esta fase irán dirigidas normalmente a conseguir una nueva versión.

Al final de la fase de transición se debe alcanzar el Product Release Milestone.

1.3.1.5 Iteraciones.

Es práctico dividir el trabajo en partes más pequeñas o en mini proyectos. Cada mini proyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en el flujo de trabajo, y los incrementos al crecimiento del producto. Como resultado de una iteración se tiene un versión (interna o externa) del producto o

partes del producto final en desarrollo, los cuales crecen incrementalmente de iteración a iteración hasta llegar a ser el sistema final **¡Error! No se encuentra el origen de la referencia..**

Las ventajas de un método iterativo comparado un proceso tradicional en cascada son:

- Mitigación de riesgos temprana.
- El cambio es controlable.
- Alto nivel de rehusó.
- El equipo de proyecto puede aprender a lo largo del método.
- Mejor calidad.

1.3.2 Eje vertical aspectos estáticos del proceso – Dimensión estática del proceso.

Un proceso describe **Quién** esta haciendo **Qué**, **Cómo** y **Cuándo**. UP es representado por cuatro elementos primarios de modelamiento.

- Roles, 'Quién'.
- Actividades, 'Cómo'.
- Artefactos, 'Qué'.
- Flujos de trabajo, 'Cuándo'.

1.3.2.1 Actividades, Artefactos, y Roles.

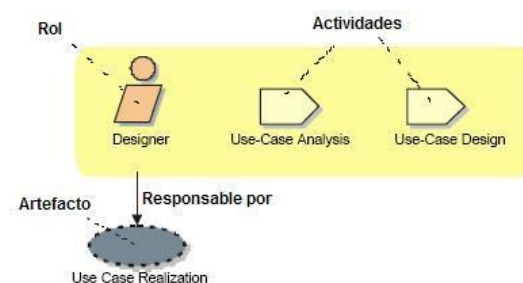


Figura 3: Roles, actividades y artefactos.

1.3.2.1.1 Roles.

Un rol define el comportamiento y responsabilidades de un individuo, o de un grupo de individuos trabajando juntos como un equipo. Una persona puede desempeñar diversos roles, así como un mismo rol puede ser representado por varias personas.

Las responsabilidades de un rol son tanto el llevar a cabo un conjunto de actividades como el ser el dueño de un conjunto de artefactos **¡Error! No se encuentra el origen de la referencia..**

1.3.2.1.2 Actividades.

Una actividad en concreto es una unidad de trabajo que una persona que desempeñe un rol puede ser solicitado a que realice. Las actividades tienen un objetivo concreto, normalmente expresado en términos de crear o actualizar algún producto. Cada

actividad es asignada a un rol específico, y afecta a uno o un número pequeño de artefactos. Una actividad debe ser usada como un elemento de planeación y progreso.

1.3.2.1.3 *Artefactos*

Un producto o artefacto es un trozo de información que es producido, modificado o usado durante el proceso de desarrollo de software. Los productos son los resultados tangibles del proyecto, las cosas que va creando y usando hasta obtener el producto final [8]. Los roles utilizan los artefactos como entrada para desarrollar una actividad, y también son el resultado o salida de dichas actividades.

Un artefacto puede tomar varias formas de las siguientes formas **¡Error! No se encuentra el origen de la referencia.:**

- Un documento, como el documento de la arquitectura del software.
- Un modelo, como el modelo de Casos de Uso o el modelo de diseño.
- Un elemento del modelo, un elemento que pertenece a un modelo como una clase, un Caso de Uso o un subsistema.
- Código fuente.
- Ejecutables.

1.3.2.2 *Flujos de Trabajo.*

Con la enumeración de roles, actividades y artefactos no se define un proceso, necesitamos contar con una secuencia de actividades realizadas por los diferentes roles, así como la relación entre los mismos. Un flujo de trabajo es una secuencia de actividades que producen un resultado de valor observable. Los siguientes son los flujos de trabajo centrales del UP.

1.3.2.2.1 *Modelado del Negocio*

Uno de los mayores problemas en las comunidades de ingeniería del software y de ingeniería del negocio es la inapropiada comunicación; UP direcciona este problema dando un proceso y lenguaje común para las dos comunidades, así como también muestra como crear y mantener una trazabilidad directa entre el negocio y los modelos software.

Los objetivos del modelado de negocio son **¡Error! No se encuentra el origen de la referencia.:**

- Entender la estructura y la dinámica de la organización para la cual el sistema va ser desarrollado (organización objetivo).
- Entender el problema actual en la organización objetivo e identificar potenciales mejoras.
- Asegurar que clientes, usuarios finales y desarrolladores tengan un entendimiento común de la organización objetivo.
- Derivar los requisitos del sistema necesarios para apoyar a la organización objetivo.

Para lograr estos objetivos, el modelo de negocio describe como desarrollar una visión de la nueva organización, basado en esta visión se definen procesos, roles y responsabilidades de la organización por medio de un modelo de Casos de Uso del

negocio y un Modelo de Objetos del Negocio. Complementario a estos modelos, se desarrollan otras especificaciones tales como un Glosario.

1.3.2.2.2 *Requisitos.*

Este es uno de los flujos de trabajo más importantes, porque en él se establece qué tiene que hacer exactamente el sistema que se construya. En esta línea los requisitos son el contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar los requisitos que se especifiquen.

Los objetivos del flujo de Requisitos es **¡Error! No se encuentra el origen de la referencia.:**

- Establecer y mantener un acuerdo entre clientes y otros stakeholders sobre lo que el sistema podría hacer.
- Proveer a los desarrolladores un mejor entendimiento de los requisitos del sistema.
- Definir el ámbito del sistema.
- Proveer una base para la planeación de los contenidos técnicos de las iteraciones.
- Proveer una base para estimar costos y tiempo de desarrollo del sistema.
- Definir una interfaz de usuarios para el sistema, enfocada a las necesidades y metas del usuario.

Los requisitos se dividen en dos grupos. Los requisitos funcionales representan la funcionalidad del sistema. Se modelan mediante diagramas de Casos de Uso. Los requisitos no funcionales representan aquellos atributos que debe exhibir el sistema, pero que no son una funcionalidad específica. Por ejemplo requisitos de facilidad de uso, fiabilidad, eficiencia, portabilidad, etc.

En este flujo de trabajo, y como parte de los requisitos de facilidad de uso, se diseña la interfaz gráfica de usuario. Para ello habitualmente se construyen prototipos de la interfaz gráfica de usuario que se contrastan con el usuario final

1.3.2.2.3 *Análisis y Diseño*

El objetivo del flujo de trabajo de análisis y diseño es traducir los requisitos a una especificación que describe cómo implementar el sistema en la fase de implementación.

Los objetivos del análisis y diseño son **¡Error! No se encuentra el origen de la referencia.:**

- Transformar los requisitos al diseño del futuro sistema.
- Desarrollar una arquitectura para el sistema.
- Adaptar el diseño para que sea consistente con el entorno de implementación, diseñando para el rendimiento.

El análisis consiste en obtener una visión del sistema que se preocupa de ver qué hace, de modo que sólo se interesa por los requisitos funcionales. Por otro lado el diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, en definitiva cómo cumple el sistema sus objetivos.

Los resultados del análisis y diseño son un modelo de diseño y opcionalmente un modelo de análisis. El modelo de diseño sirve como un abstracción del código fuente; es decir, actúa como un 'anteproyecto' de cómo el código fuente debe ser estructurado

y escrito, este modelo consiste en un diseño de clases estructuradas, que pueden ser agregadas en paquetes y subsistemas con interfaces bien definidas.

Otro producto importante de este flujo es la documentación de la arquitectura de software. Las actividades de diseño son centradas en torno a la noción de la arquitectura, la producción y la validación de esta es el foco principal de iteraciones tempranas en el diseño. La arquitectura esta representada por un número de vistas arquitecturales **¡Error! No se encuentra el origen de la referencia..** Estas vistas capturan las decisiones trascendentales del diseño. En escénica, las vistas arquitecturales son abstracciones o simplificaciones del diseño en su totalidad, en las cuales las características importantes son más visibles y los detalles son dejados a un lado.

1.3.2.2.4 *Implementación.*

En este flujo de trabajo se implementan las clases y objetos en archivos fuente, binarios, ejecutables y demás. Además se deben hacer las pruebas de unidad: cada implementador es responsable de probar las unidades que produzca. El resultado final de este flujo de trabajo es un sistema ejecutable.

Los objetivos de implementación son **¡Error! No se encuentra el origen de la referencia.:**

- Definir la organización del código, en términos de implementación de subsistemas organizados en capas.
- Implementar clases y objetos en términos de componentes (archivos fuente, binarios, ejecutables y otros)
- Probar los componentes desarrollados como unidades.
- Integrar los resultados producidos por implementaciones individuales (o equipos), en un sistema ejecutable.

La estructura de todos los elementos implementados forma el modelo de implementación. La integración debe ser incremental, es decir, en cada momento sólo se añade un elemento. De este modo es más fácil localizar fallos y los componentes se prueban más a fondo. En fases tempranas del proceso se pueden implementar prototipos para reducir el riesgo. Su utilidad puede ir desde ver si el sistema es viable desde el principio, probar tecnologías o diseñar la interfaz de usuario. Los prototipos pueden ser exploratorios (desechables) o evolutivos. Estos últimos llegan a transformarse en el sistema final.

1.3.2.2.5 *Pruebas.*

Este flujo de trabajo es el encargado de evaluar la calidad del producto que estamos desarrollando, pero no para aceptar o rechazar el producto al final del proceso de desarrollo, sino que debe ir integrado en todo el ciclo de vida.

Esta disciplina brinda soporte a las otras disciplinas. Sus objetivos son **¡Error! No se encuentra el origen de la referencia.:**

- Verificar la interacción entre objetos
- Encontrar y documentar defectos en la calidad del software
- Verificar la apropiada integración de todos los componentes del software.
- Verificar que todos los requerimientos han sido implementados correctamente
- Identificar y garantizar que los defectos son dirigidos previamente de desplegar el software

Las actividades de este flujo comienzan pronto en el proyecto con el plan de prueba (el cual contiene información sobre los objetivos generales y específicos de las pruebas en el proyecto, así como las estrategias y recursos con que se dotará a esta tarea), o incluso antes con alguna evaluación durante la fase de inicio, y continuará durante todo el proyecto.

El desarrollo del flujo de trabajo consistirá en planificar que es lo que hay que probar, diseñar cómo se va a hacer, implementar lo necesario para llevarlos a cabo, ejecutarlos en los niveles necesarios y obtener los resultados, de forma que la información obtenida sirva para ir refinando el producto a desarrollar

1.3.2.2.6 Despliegue

El objetivo de este flujo de trabajo es producir con éxito distribuciones del producto y distribuirlo a los usuarios. Las actividades implicadas incluyen:

- Probar el producto en su entorno de ejecución final.
- Empaquetar el software para su distribución.
- Distribuir el software.
- Instalar el software.
- Proveer asistencia y ayuda a los usuarios.
- Formar a los usuarios y al cuerpo de ventas.
- Migrar el software existente o convertir bases de datos.

Este flujo de trabajo se desarrolla con mayor intensidad en la fase de transición, ya que el propósito del flujo es asegurar una aceptación y adaptación sin complicaciones del software por parte de los usuarios. Su ejecución inicia en fases anteriores, para preparar el camino, sobre todo con actividades de planificación, en la elaboración del manual de usuario y tutoriales.

2 EL PROCESO UNIFICADO DESCRITO A TRAVÉS DE SPEM 2.0

Para efectos de modelar y realizar la adaptación del Proceso Unificado (UP), en este proyecto, el UP se ha implementado a través del lenguaje SPEM 2.0 (Software & Systems Process Engineering Metamodel), que sirve para la representación de modelos de procesos de ingeniería del software e ingeniería de sistemas. Como editor de procesos se ha utilizado la herramienta EPFC (Eclipse Process Framework Composer) que se basa en SPEM y permite definir, gestionar y reutilizar un repositorio de fragmentos de métodos y procesos.

En las siguientes secciones se presenta la implementación del UP en EPFC.

2.1 Aspectos empleados para implementar UP con SPEM 2.0

En SPEM 2 se distinguen dos aspectos en el momento de implementar una metodología, primero, se define y estructura el contenido del método con elementos de contenido, es decir, los elementos primarios o constructores básicos y luego, se definen los procesos mediante la reutilización de los elementos del método.

Los elementos del contenido del método fueron categorizados por las disciplinas del UP, debido a la ortogonalidad del Contenido del Método y de los Procesos en SPEM y UP, ver figura 3.

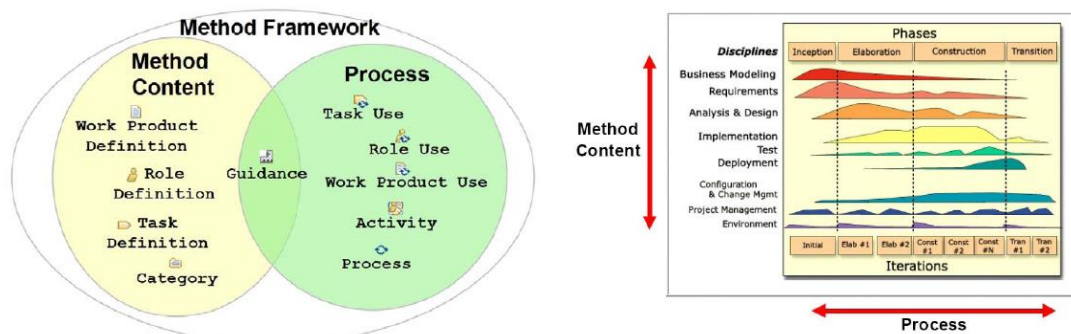


Figura 3. Ortogonalidad en SPEM y UP

2.1 Método Plugin del Proceso Unificado.

Un método Plugin es un paquete que representa un contenedor físico para Paquetes de contenido y procesos. El contenido del método define un nivel de granularidad para la modularización y organización de contenido del Método y Procesos **¡Error! No se encuentra el origen de la referencia..** Para la definición del UP se implemento un Plugin llamado Unfied Process ver figura 4.

2.1.1 Contenido del Método.

Los elementos de contenido del método del Plugin fueron organizados mediante una jerarquía de paquetes de contenido. Los cinco paquetes de contenido de acuerdo con las disciplinas del UP son: Modelado del Negocio, Requerimientos, Análisis y Diseño, Implementación y Pruebas. Ver figura 4.

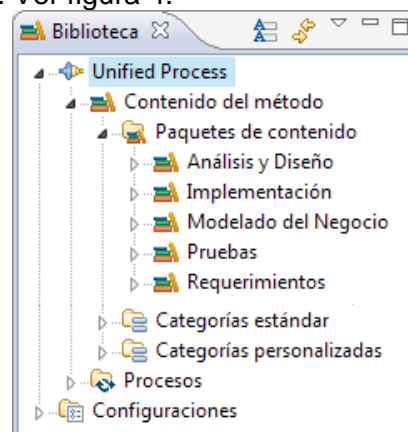


Figura 4. Paquetes de Contenido definidos para el UP.

2.1.1.1 Paquetes de Contenido

En cada paquete de contenido se definieron los elementos de contenido del método básicos: roles, tareas y productos de trabajo involucrados en cada una de las disciplinas del UP. La figura 5 muestra la división del paquete de contenido Modelado del Negocio y sus elementos básicos; por cuestiones de espacio solo se muestran algunos roles, tareas y productos de trabajo. Es importante destacar que cada uno de los cinco paquetes de contenido están estructurados de la misma forma.

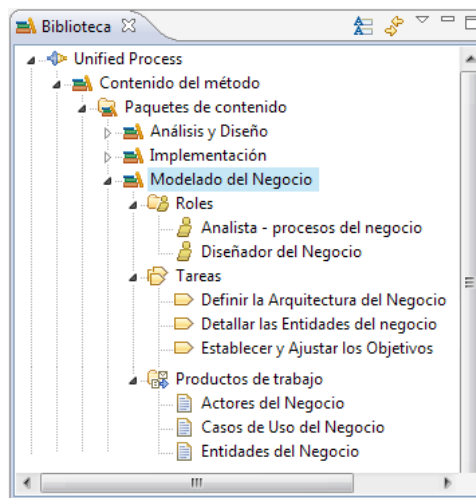


Figura 5. División del Paquete de Contenido Modelado del Negocio.

2.1.1.2 Categorías

Una Categoría es un elemento de contenido, o de proceso, usado para categorizar **¡Error! No se encuentra el origen de la referencia.**, es decir, clasificar o agrupar dichos elementos en base a los criterios del usuario. Una categoría puede tener 0..* subcategorías. Esto permite establecer cualquier tipo de jerarquía de agrupamiento de elementos. SPEM 2 distingue dos clases de categorías:

2.1.1.3 Categoría Estándar

Los elementos del contenido de método se pueden categorizar mediante su asociación con categorías estándar (disciplinas, dominios, clases de productos de trabajo, conjuntos de roles y herramientas). Las categorías Estándar vienen predefinidas en SPEM. Para el Plugin del UP se utilizaron tres tipos de categorías estándar, disciplinas, clases de productos de trabajo y conjunto de Roles. Ver figura 6.

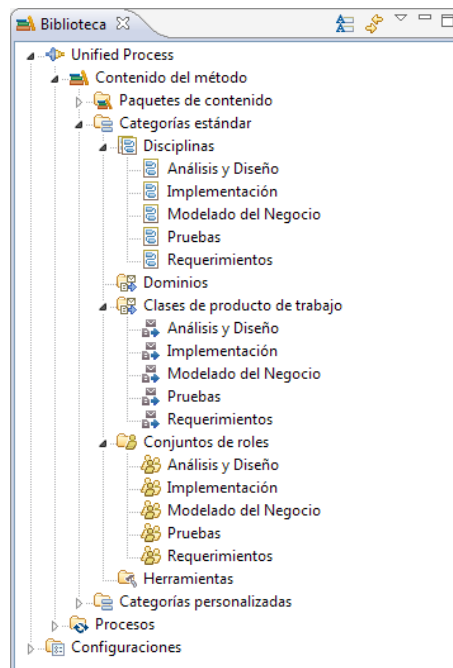


Figura 6. Categorías Estándar utilizadas en UP

2.1.1.4 Categoría Personalizada

Los elementos del contenido de método (y también los elementos en uso en procesos) se pueden organizar alternativamente mediante categorías personalizadas. Estas categorías personalizadas pueden contener cualquier tipo de elemento, proceso, categoría, etc. Son útiles para organizar de forma lógica elementos que no se puedan organizar mediante las categorías estándar. Las categorías personalizadas permiten organizar el contenido de acuerdo al esquema que queramos, sirviendo como medio eficaz para organizar el contenido para su publicación. La figura 7 muestra la estructura de las Categorías Personalizadas definidas en el Plugin y en la figura 8 se muestra su publicación en HTML.

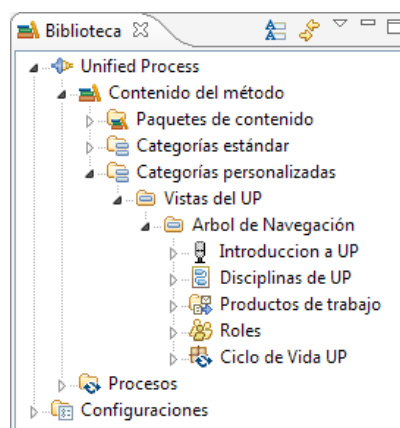


Figura 7. Estructura de las Categorías Personalizadas

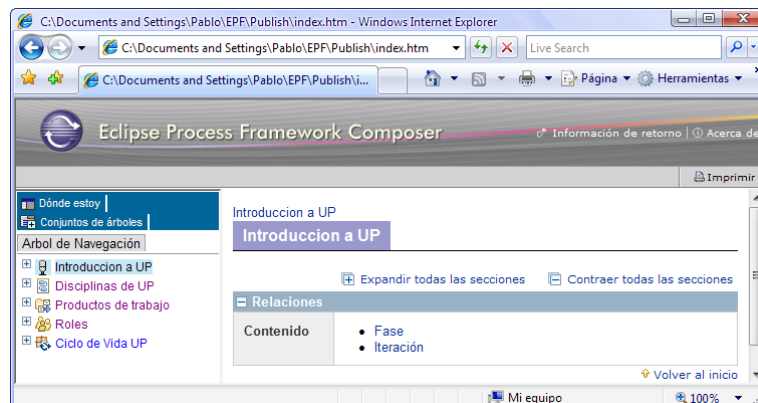


Figura 8. Publicación de las Categorías Personalizadas

2.1.2 Procesos

Un Proceso es un tipo de Actividad que describe una estructura para tipos particulares de proyectos o partes de ellos **¡Error! No se encuentra el origen de la referencia..** Los procesos software definen cómo serán ejecutados los proyectos software.

En SPEM 2 existen dos clases principales de procesos: Patrón de Proceso y Proceso para Despliegue.

2.1.2.1 Patrón de Proceso

Un patrón de proceso es un proceso especial “fragmento de proceso” que describe un grupo de actividades reusables como solución a algún tipo de problema o situación habitual **¡Error! No se encuentra el origen de la referencia..** Se definen para poder ser empleados más de una vez en uno o varios procesos o con fines de organización.

En el Plugin se definió una jerarquía de cinco Paquetes de Proceso (Process Package) por cada una de las disciplinas del UP, ver figura 9, donde cada Paquete de proceso tiene varios patrones de proceso con el fin de servir como bloques para construir el Proceso de Despliegue, ver figura 10.

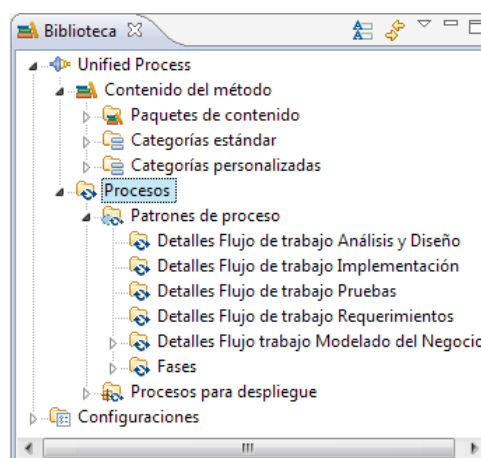


Figura 9. Paquetes de Proceso definidos para el Plugin.

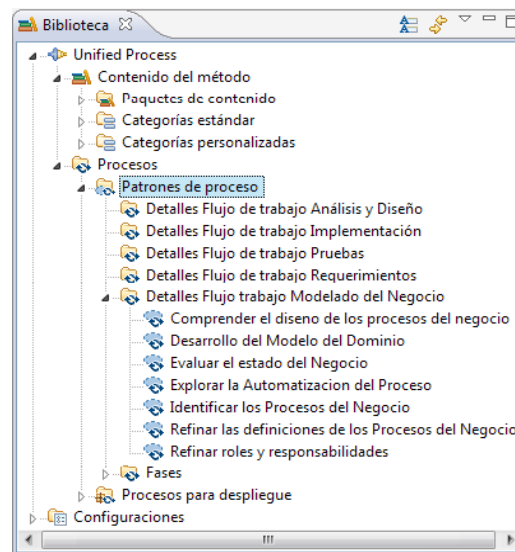


Figura 10. Patrones de proceso

2.1.2.2 Proceso para Despliegue

Describe una aproximación completa e integrada para realizar un tipo específico de proyecto, abarcando un ciclo de vida completo de desarrollo o mantenimiento **¡Error! No se encuentra el origen de la referencia..** Sirven como plantillas para planificar y ejecutar los proyectos. En un Proceso de Despliegue se ensamblan Patrones de Proceso y Elementos en Uso (tareas, roles y productos de trabajo en uso). En el proceso de despliegue para el Plugin se llama “CicloVidaUP”, esta dividido en las fases del UP, inicio, elaboración, construcción, y transición. En cada fase se reutilizaron los patrones de proceso definidos en los paquetes de proceso mencionados en la sección anterior, ver figura 11.

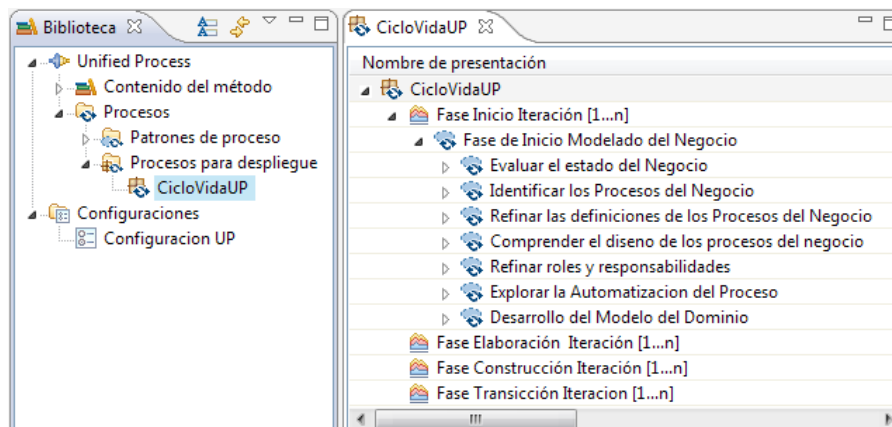


Figura 11. Proceso para despliegue.

D.2. Anexos ambiente académico

D.2.1. Proyecto MyPymes

INGENIERIA DE SOFTWARE III
DEPARTAMENTO DE SISTEMAS FIET
UNIVERSIDAD DEL CAUCA

Proyecto Final: Aplicación web para compra y venta de Productos

1. Descripción del Problema

Popayán es una ciudad colombiana, capital del departamento del Cauca y se ha caracterizado por ser una ciudad turística de arquitectura colonial. Sin embargo, su economía no es desarrollada comparada con ciudades capitales cercanas como Pasto y Cali. Las principales actividades comerciales giran entorno a la compraventa de ropa, papelería, artículos de cocina, electrodomésticos, etc. La mayoría de personas en la zona urbana viven de empleos directos e indirectos que se generan alrededor de las instituciones educativas principalmente universitarias. Las demás regiones del Departamento del Cauca principalmente se mantienen de actividades agropecuarias.

Entidades del gobierno como la cámara de comercio, Fondo Emprender y CREPIC (Centro Regional de Productividad y Competitividad) se han constituido para fomentar la creación de nuevas empresas que generen empleos y progreso para la región. Se basan en principios de apoyo a la innovación.

Sin embargo, no todo es negativo, el Cauca es un Departamento rico en recursos hídricos, turismo, fauna, flora, artesanías, educación, conocimiento, etc. Lo que se requiere es proyectos que incentiven y fomenten la economía de una región potencialmente rica.

En base a esta problemática, se requiere construir una aplicación web que permita hacer venta de productos por la web como apoyo a las nuevas empresas que se están creando y que necesitan ampliar su mercado a nivel nacional por medio de las tecnologías de la información y la comunicación (TICs).

Básicamente la aplicación permitirá mostrar información corporativa de la empresa, sus productos y permitir la compra y venta de éstos a través de Internet (Ecommerce o comercio electrónico). Un ejemplo de una aplicación web de ecommerce es <http://www.g33kchoice.com/> . Es un sitio web encargado de comercializar productos alusivos al software libre.

A continuación se muestran los requisitos funcionales y no funcionales de la aplicación especificados como items.

2. Requisitos Funcionales

R1. Gestión de usuarios.

Representa los usuarios que podrán autenticarse en el sistema y entrar al panel de control (El panel de control permitirá gestionar los principales módulos del sistema: usuarios, productos, etc).

R1.1. Debe permitir listar todos los usuarios del sistema. Debe manejar *Paginación*.

R1.2. Debe permitir agregar nuevos usuarios al sistema. Cada usuario debe tener los siguientes datos: nombre de usuario (login), contraseña, nombres, apellidos, correo electrónico, privilegio (Administrador, Publicador de Productos, Gerente y Vendedor). En general, un administrador puede entrar al panel de control y tener acceso total al sistema. El publicador de productos solo puede gestionar los productos a vender. El gerente puede solamente examinar estadísticas de ventas. El vendedor puede revisar las solicitudes de compras hechas y diligenciar el envío.

R1.3. Modificar usuarios. Debe permitir modificar cualquier dato de los usuarios registrados.

R1.4. Debe permitir consultar fácilmente un usuario (por Id, login, nombres, apellidos). El resultado de la consulta debe generar un listado con paginación.

R1.5. Debe permitir eliminar un usuario del sistema. Antes de eliminarlo debe pedir confirmación de esta acción.

R2. Gestión de Productos

Este es un módulo importante del sistema pues permite que de manera dinámica se administren los artículos que los usuarios pueden comprar.

R2.1. Listar productos. Debe manejar paginación.

R2.2. Agregar productos. Cada producto tiene un identificador (el sistema lo debe generar automáticamente), un nombre, una descripción, imagen, fecha de creación y nombre de quien la publicó.

R2.3. Modificar productos.

R2.4. Consultar productos. Debe permitir entrar palabras claves del nombre o el identificador. El resultado de la consulta debe generar un listado con paginación.

R2.5. Debe permitir eliminar un producto. Antes de eliminarlo debe pedir confirmación de esta acción.

Si el producto generó ventas no se podrá eliminar.

R3. Gestión de Comentarios y/o Preguntas

Los usuarios anónimos pueden acceder a un formulario y dejar sus comentarios y preguntas. Los administradores deben revisar estos comentarios y reponder. Se

registra: Nombre (opcional), email, fecha y comentario. El comentario debe poseer un *editor html*.

R3.1. Listar comentarios y/o preguntas. Se debe manejar paginación.

R3.2. Modificar el estado: pendiente, respondido.

R3.3. Eliminar comentario y/o pregunta. Solamente los administradores pueden eliminar preguntas cuando alguien haya colocado información irrespetuosa.

R4. Estadísticas

Las estadísticas podrán ser consultadas por los usuarios con perfil de gerentes. Todas las estadísticas deben tener *gráficas estadísticas* de barras o pastel.

R4.1. Ventas y recaudo de dinero por rango de fechas.

R4.2. Productos más vendidos.

R4.3. Otras, se pueden proponer.

R5. Carrito de Compras

Los usuarios anónimos que visiten el sitio pueden navegar y conocer los productos que se ofrecen y llenar en el carrito de compras los productos que desean comprar.

R5.1. Mostrar productos del carrito de compras

R5.2. Agregar producto al carrito de compras

R5.3. Eliminar producto del carrito de compras

R5.4. Efectuar compra. Se debe mostrar un formulario donde se solicitan los datos personales del comprador: nombres, apellidos, documento de identificación, ciudad, dirección, email.

Una vez comprado, el sistema enviará un correo electrónico al comprador enviándole los detalles de la compra y el número de cuenta de la cuenta donde debe consignar.

R6. Gestión de Compras

R6.1. Listar compras. Se debe manejar paginación.

R6.2. Actualizar Estado de la Compra. Se manejan los siguientes estados de la compra: iniciada (cuando el cliente la registró), consignada (el cliente hizo la consignación bancaria de la compra), despachada (se hizo el envío de la mercancía).

R6.3. Eliminar compras, se pueden eliminar compras con fechas viejas a las cuales no se les hizo consignación.

R7. Gestión de Temas y Plantillas

R7.1. Debe desplegar un listado con los temas (colores, fuentes, estilos) que ofrece el sistema, los cuales deben ser *acorde al producto* que se está vendiendo. Estos temas deben ser creados con CSS (Hojas de estilo en Cascada). Por lo menos deben haber dos temas disponibles.

R7.2. El administrador puede escoger uno de ellos y el sitio quedará configurado con este tema.

R8. Gestión de contenidos y menú principal

Las opciones de la barra del menú principal del sitio, así como las páginas que son enlazadas desde estas opciones deben ser generadas dinámicamente.

R8.1. Listar menús.

R8.2. Agregar menús. El agregar incluye enlazar una página estática. Cada menú tiene un ID y el título.

R8.3. Modificar menús.

R8.2. Listar páginas estáticas.

R8.3. Modificar páginas estáticas.

R8.4. Agregar páginas estáticas. Para agregar y modificar es imprescindible disponer de un editor html amigable al usuario para editar el contenido (texto, tablas e imágenes).

R.8.5. Eliminar páginas estáticas.

3. Requisitos No Funcionales y Restricciones

- La aplicación debe ser *usable*, esto significa tener un buen diseño gráfico, buenos contenidos, fácil de usar y aprender por el usuario. Si no se tiene un buen lookandfeel (apariciencia) del sitio el comprador posiblemente no visite el sitio.
- El usuario no debe tardar más de 2 minutos en hacer una compra.
- El desempeño de la aplicación debe ser bueno. El tiempo de respuesta de una página no debe tardar más de 5 segundos.
- La aplicación debe ser segura. Debe estar preparada contra ataques típicos como: inyección SQL, Crosssitescripting e inclusión remota de archivos.
- Se puede programar en cualquier lenguaje sobre la web y/o tecnología: asp.net, php, java, python, perl, etc.
- Se puede o no utilizar frameWorks MVC (Modelo Vista Controlador) de desarrollo.

- Se puede utilizar cualquier mecanismo de persistencia de datos: base de datos relacionales, archivos planos, archivos XML. Si se opta por una base de datos relacional se puede utilizar cualquier motor: MySQL, PostgreSQL, Oracle, MS Access, SQL Server, SQL Lite, Fire Bird, etc.
- Se puede escoger cualquier tecnología de Servidor Web: Apache, ISS, cherokee, lighttpd, etc.
- Se puede escoger cualquier metodología o paradigma de desarrollo: RUP, XP, otras.

D.2.2. Niveles de las variables de caracterización de los proyectos

CARACTERIZACIÓN DE PROYECTOS		
Documento: Plantilla de Caracterización de Proyectos.		
Propósito: Esta plantilla tiene el propósito de facilitar la ejecución de la caracterización del proyecto, en su contenido están los niveles de clasificación de las variables de caracterización, los cuales deben asignar de acuerdo a las particularidades del proyecto y de la organización.		
Nombre del Proyecto: Aplicación web para compra y venta de Productos.		
Encargado(s):		
Fecha:		
NIVEL	1. Tamaño del equipo	
1	X	Muy pequeño: 1-10 personas
2		Pequeño: 11-50 personas
3		Medianas: 51-200 personas
NIVEL	2. Distribución geográfica del equipo	
1	X	Misma Oficina
2		Mismo edificio, diferentes oficinas
3		Misma ciudad, misma empresa, diferente edificio
4		Misma ciudad, diferente empresa
5		Diferente ciudad
3. Experiencia del equipo		
NIVEL	3.1 Experiencia en el proceso	
1		No hay ningún proyecto
2		1 proyecto
3	X	2 a 3 proyectos
4		4 a 5 proyectos
5		Más de 5 proyectos
NIVEL	3.2 Experiencia en el dominio de la aplicación	
1		No hay ningún proyecto
2		1 proyecto
3	X	2 a 3 proyectos
4		4 a 5 proyectos
5		Más de 5 proyectos
NIVEL	3.3 Experiencia técnica	
1		No hay ningún proyecto
2		1 proyecto
3	X	2 a 3 proyectos
4		4 a 5 proyectos

CARACTERIZACIÓN DE PROYECTOS		
5		Más de 5 proyectos
NIVEL		4. Criticidad del proyecto
1		Bajo desempeño
2		Pérdidas bajas (pérdidas fácilmente recuperables)
3		Daños moderados
4	X	Pérdidas grandes (pérdidas irrecuperables)
5		Riesgo de perder vidas
NIVEL		5. Tamaño del Proyecto
1	X	Hasta \$ 5'000.000
2		Entre \$ 5'000.000 y \$ 15'000.000
3		Entre \$ 15'000.000 \$ 25'000.000
4		Entre \$ 25'000.000 y \$ 35'000.000
5		Más de \$ 35'000.000
NIVEL		6. Características que priorizan el proyecto
1		Calidad = 90, tiempo = 10
2		Calidad = 70, tiempo = 30
3		Calidad = 50, tiempo = 50
4	X	Calidad = 30, tiempo = 70
5		Calidad = 10, tiempo = 90

D.2.3. Artefactos que serán adaptados en la disciplina de Modelado del Negocio

ATEFACTOS QUE SERÁN ADAPTADOS		
Objetivo: Esta plantilla permite realizar la selección de los artefactos, de la correspondiente disciplina, que pueden ser adaptados.		
Nombre del proyecto: Aplicación web para compra y venta de Productos.		
Disciplina: Modelado del Negocio		
Encargado:		
Fecha :		
Artefacto	Seleccionado (si/no)	Comentario
Actores del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Documento de la Arquitectura del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Entidades del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Eventos del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Objetivos del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Glosario del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Modelo de Análisis del negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Reglas del negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Sistema del negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Casos de Uso del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos

ATEFACTOS QUE SERÁN ADAPTADOS		
Modelo de Casos de Uso del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Ejecución de los Casos de Uso del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Visión del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Trabajadores del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Especificación Suplementaria del Negocio	SI	Puede ser adaptado. No hay normas o modelos de procesos
Valoración de la Organización Objetivo	SI	Puede ser adaptado. No hay normas o modelos de procesos

D.2.4. Artefactos que serán adaptados en la disciplina de Implementación

ATEFACTOS QUE SERÁN ADAPTADOS		
Objetivo: Esta plantilla permite realizar la selección de los artefactos, de la correspondiente disciplina, que pueden ser adaptados.		
Nombre del proyecto: Aplicación web para compra y venta de de Productos.		
Disciplina: Implementación		
Encargado:		
Fecha :		
Artefacto	Seleccionado	Comentario
Construcción	SI	Puede ser adaptado. No hay normas o modelos de procesos
Construir el plan de integración	SI	Puede ser adaptado. No hay normas o modelos de procesos
Documento de arquitectura de software	SI	Puede ser adaptado. No hay normas o modelos de procesos
Elemento de comprobabilidad	SI	Puede ser adaptado. No hay normas o modelos de procesos
Elementos de implementación	SI	Puede ser adaptado. No hay normas o modelos de procesos
Modelo de Implementación	SI	Puede ser adaptado. No hay normas o modelos de procesos
Prueba stub	SI	Puede ser adaptado. No hay normas o modelos de procesos
Pruebas del Desarrollador	SI	Puede ser adaptado. No hay normas o modelos de procesos
Subsistema de implementación	SI	Puede ser adaptado. No hay normas o modelos de procesos

D.2.5. artefactos que serán adaptados en la disciplina de Pruebas

ATEFACTOS QUE SERÁN ADAPTADOS
Objetivo: Esta plantilla permite realizar la selección de los artefactos, de la correspondiente disciplina, que pueden ser adaptados.
Nombre del proyecto: Aplicación web para compra y venta de productos
Disciplina: Pruebas

ATEFACTOS QUE SERÁN ADAPTADOS		
Encargado:		
Fecha :		
Artefacto	Seleccionado (si/no)	Comentario
Arquitectura de automatización de prueba	SI	Puede ser adaptado. No hay normas o modelos de procesos
Caso de Prueba	SI	Puede ser adaptado. No hay normas o modelos de procesos
Configuración del ambiente de prueba	SI	Puede ser adaptado. No hay normas o modelos de procesos
Datos de prueba	SI	Puede ser adaptado. No hay normas o modelos de procesos
Especificación de interfaz de prueba	SI	Puede ser adaptado. No hay normas o modelos de procesos
Estrategia de prueba	SI	Puede ser adaptado. No hay normas o modelos de procesos
Grupo de pruebas	SI	Puede ser adaptado. No hay normas o modelos de procesos
Lista de ideas de prueba	SI	Puede ser adaptado. No hay normas o modelos de procesos
Modelo de análisis de trabajo	SI	Puede ser adaptado. No hay normas o modelos de procesos
Plan de pruebas	SI	Puede ser adaptado. No hay normas o modelos de procesos
Registro de Pruebas	SI	Puede ser adaptado. No hay normas o modelos de procesos
Resultados de prueba	SI	Puede ser adaptado. No hay normas o modelos de procesos
Resumen de evaluación de prueba	SI	Puede ser adaptado. No hay normas o modelos de procesos
Script de prueba	SI	Puede ser adaptado. No hay normas o modelos de procesos

D.2.6. Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización (Modelado del Negocio)

MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN								
Documento: Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización								
Propósito: En esta matriz se recopilan los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina y se establecen los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina.								
Nombre del Proyecto: Aplicación web para compra y venta de Productos.								
Encargado(s):								
	Variables de Caracterización							
	Tamaño del equipo	Distribución geográfica del equipo	Experiencia en el proceso	Experiencia en la aplicación	Experiencia técnica	Criticidad del proyecto	Tamaño del Proyecto	Características que priorizan el proyecto
Disciplina: Modelado del Negocio								
ARTEFACTOS	Niveles	Niveles	Niveles	Niveles	Niveles	Niveles	Niveles	Niveles
(que se pueden adaptar)	1	1	3	3	3	4	1	4
	Muy pequeño: 1-10 personas	Misma Oficina	2 a 3 proyectos	2 a 3 proyectos	2 a 3 proyectos	Pérdidas grandes (pérdidas irrecuperables)	Hasta \$ 5'000.000	Calidad = 30, tiempo = 70
Actores del Negocio	I	SR	I	SR	I	I	I	I
Documento de la Arquitectura del Negocio	I	I	SR	I	SR	I	I	I
Entidades del Negocio	SR	SR	I	I	I	SR	SR	SR
Eventos del Negocio	SR	SR	SR	SP	I	I	I	I
Objetivos del Negocio	I	SP	I	SR	SP	I	I	I
Glosario del Negocio	SR	SR	SR	I	I	I	SR	SR
Modelo de Análisis del negocio	I	SP	I	I	I	SP	SP	SP

MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN								
Reglas del negocio	I	SR	SP	I	SR	SR	SR	SR
Sistema del negocio	I	SP	SP	I	I	I	I	I
Casos de Uso del Negocio	I	I	I	SR	SR	SR	I	I
Modelo de Casos de Uso del Negocio	I	I	SR	I	SR	SR	SR	SR
Ejecución de los Casos de Uso del Negocio	SR	I	SR	SR	SR	SR	I	I
Visión del Negocio	I	SR	I	I	I	I	SR	I
Trabajadores del Negocio	SR	SR	I	I	I	I	SR	SR
Especificación Suplementaria del Negocio	I	I	SR	SR	SR	SR	SR	SR
Valoración de la Organización Objetivo	I	I	I	I	SR	SR	I	I

D.2.7. Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización (Implementación)

MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN								
Documento: Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización								
Propósito: En esta matriz se recopilan los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina y se establecen los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina.								
Nombre del Proyecto: Aplicación web para compra y venta de Productos.								
Encargado(s):								
Disciplina: Implementación	Variables de Caracterización							
	Tamaño del equipo	Distribución geográfica del equipo	Experiencia en el proceso	Experiencia en el dominio de la aplicación	Experiencia técnica	Criticidad del proyecto	Tamaño del Proyecto	Características que priorizan el Proyecto
ARTEFACTOS	Nivel	Nivel	Nivel	Nivel	Nivel	Nivel	Nivel	Nivel

MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN								
(que se pueden adaptar)	1	1	3	3	3	4	1	4
	Muy pequeño: 1-10 personas	Misma Oficina	2 a 3 proyectos	2 a 3 proyectos	2 a 3 proyectos	Pérdidas grandes (pérdidas irrecuperables)	Hasta \$ 5'000.000	Calidad = 30, tiempo = 70
Construcción	I	I	SP	SP	SP	SP	SP	SP
Construir el plan de integración	I	I	SR	SR	SR	SR	SR	I
Documento de arquitectura de software	I	I	SR	SR	SR	SR	SR	I
Elemento de comprobabilidad	I	I	NP	NP	SP	SR	NP	SR
Elementos de implementación	I	I	SR	SR	SR	SR	SR	SR
Modelo de Implementación	SR	SR	SP	SP	SP	SR	SR	SR
Prueba stub	NP	NP	NP	NP	NP	NP	NP	NP
Pruebas del Desarrollador	I	I	SR	SR	SR	SP	SR	SR
Subsistema de implementación	I	I	SR	SP	SP	SR	SR	I

D.2.8. Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización (Pruebas)

MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN	
Documento: Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización	
Propósito: En esta matriz se recopilan los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina y se establecen los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina.	
Nombre del Proyecto: Aplicación web para compra y venta de productos	
Encargado(s):	
Disciplina: Implementación	Variables de Caracterización

MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN								
	Tamaño del equipo	Distribución geográfica del equipo	Experiencia en el proceso	Experiencia en el dominio de la aplicación	Experiencia técnica	Criticidad del proyecto	Tamaño del Proyecto	Características que priorizan el Proyecto
ARTEFACTOS (que se pueden adaptar)	Nivel	Nivel	Nivel	Nivel	Nivel	Nivel	Nivel	Nivel
	1	1	3	3	3	4	1	4
	Muy pequeño: 1-10 personas	Misma Oficina	2 a 3 proyectos	2 a 3 proyectos	2 a 3 proyectos	Pérdidas grandes (pérdidas irre recuperables)	Hasta \$ 5'000.000	Calidad = 30, tiempo = 70
Arquitectura de automatización de prueba	SR	SR	SR	SR	SR	I	I	SP
Caso de Prueba	SP	SP	SR	SR	SR	SR	I	I
Configuración del ambiente de prueba	I	I	I	I	SR	SR	SR	I
Datos de prueba	SR	SR	SR	SR	SR	SR	SR	SR
Especificación de interfaz de prueba	I	I	I	I	I	SR	I	I
Estrategia de prueba	SR	SR	NP	NP	NP	NP	NP	NP
Grupo de pruebas	I	I	I	NP	NP	NP	NP	NP
Lista de ideas de prueba	SR	SR	SR	I	I	I	I	I
Modelo de análisis de trabajo	I	I	NP	NP	I	I	NP	SR
Plan de pruebas	SR	SR	SR	SR	SR	SR	SR	SR
Registro de Pruebas	SP	SP	SP	SP	SP	SR	SR	SR
Resultados de prueba	SP	SP	SP	SP	SR	SR	SP	SR
Resumen de evaluación de prueba	I	I	I	I	I	I	I	I
Script de prueba	I	I	I	I	I	I	I	SR

D.2.9. Matriz final de clasificación de artefactos por disciplina (Modelado del Negocio)

MATRIZ FINAL DE CLASIFICACIÓN ARTEFACTOS		
Documento: Matriz Final de clasificación de artefactos por disciplinas		
Propósito: En esta matriz se recopilan la clasificación final de cada artefacto con respecto a las variables de caracterización y se adicionan los artefactos que fueron seleccionados como no adaptables		
Nombre del Proyecto: Aplicación web para compra y venta de Productos.		
Encargado(s):		
Disciplina: Modelado del Negocio		
ARTEFACTOS (que se pueden adaptar)	Clasificación final	Comentario
1 Actores del Negocio	I	
2 Documento de la Arquitectura del Negocio	I	
3 Entidades del Negocio	SR	
4 Eventos del Negocio	I	
5 Objetivos del Negocio	I	
6 Glosario del Negocio	SR	
7 Modelo de Análisis del negocio	SP	
8 Reglas del negocio	SR	
9 Sistema del negocio	I	
10 Casos de Uso del Negocio	I	
11 Modelo de Casos de Uso del Negocio	SR	
12 Ejecución de los Casos de Uso del Negocio	SR	
13 Visión del Negocio	I	
14 Trabajadores del Negocio	I	
15 Especificación Suplementaria del Negocio	SR	
16 Valoración de la Organización Objetivo	I	

D.2.10. Matriz final de clasificación de artefactos por disciplina (Implementación)

MATRIZ FINAL DE CLASIFICACIÓN ARTEFACTOS		
Documento: Matriz Final de clasificación de artefactos por disciplinas		
Propósito: En esta matriz se recopilan la clasificación final de cada artefacto con respecto a las variables de caracterización y se adicionan los artefactos que fueron seleccionados como no adaptables		
Nombre del Proyecto: Aplicación web para compra y venta de Productos.		
Encargado(s):		
Disciplina: Implementación		
ARTEFACTOS	Clasificación final	Comentario

MATRIZ FINAL DE CLASIFICACIÓN ARTEFACTOS			
(que se pueden adaptar)			
1	Construcción	SP	
2	Construir el plan de integración	SR	
3	Documento de arquitectura de software	SR	
4	Elemento de comprobabilidad	NP	
5	Elementos de implementación	SP	
6	Modelo de Implementación	SR	
7	Prueba stub	NP	
8	Pruebas del Desarrollador	SR	
9	Subsistema de implementación	SP	

D.2.11. Matriz final de clasificación de artefactos por disciplina (Pruebas)

MATRIZ FINAL DE CLASIFICACIÓN ARTEFACTOS			
Documento: Matriz Final de clasificación de artefactos por disciplinas			
Propósito: En esta matriz se recopilan la clasificación final de cada artefacto con respecto a las variables de caracterización y se adicionan los artefactos que fueron seleccionados como no adaptables			
Nombre del Proyecto: Aplicación web para compra y venta de Productos.			
Encargado(s):			
Disciplina: Implementación			
ARTEFACTOS		Clasificación final	Comentario
(que se pueden adaptar)			
1	Arquitectura de automatización de prueba	SR	
2	Caso de Prueba	SR	
3	Configuración del ambiente de prueba	I	
4	Datos de prueba	SR	
5	Especificación de interfaz de prueba	NP	
6	Estrategia de prueba	NP	
7	Grupo de pruebas	NP	
8	Lista de ideas de prueba	I	
9	Modelo de análisis de trabajo	NP	
10	Plan de pruebas	SR	
11	Registro de Pruebas	SP	
12	Resultados de prueba	SP	
13	Resumen de evaluación de prueba	I	
14	Script de prueba	I	

D.2.12. Roles y actividades necesarios para la disciplina (Modelado del Negocio)

PLANTILLA ROLES Y ACTIVIDADES					
Documento: Plantilla Roles y actividades necesarios para las disciplinas					
Propósito: Esté artefacto tiene el propósito de recopilar los roles y actividades indispensables para producir los artefactos del proceso adaptado					
Nombre del Proyecto: Aplicación web para compra y venta de Productos					
Encargado(s):					
Disciplina: Modelado del Negocio					
ARTEFACTOS, ACTIVIDADES Y ROLES NECESARIOS				ACTIVIDADES Y ROLES A ELIMINAR	
ARTEFACTO	ACTIVIDADES		ROLES	ACTIVIDADES A ELIMINAR	ROLES A ELIMINAR
	Entrada a la Actividad	Salida de la Actividad			
Entidades del Negocio	Detallar las Entidades del negocio	Buscar los Trabajadores y las Entidades del Negocio	Analista del Procesos del Negocio	Establecer y Ajustar los Objetivos	
		Detallar las Entidades del negocio	Diseñador del negocio	Evaluar la Organización Objetivo	
Glosario del Negocio	Buscar los Actores y Casos de Uso del negocio	Capturar el Vocabulario común de la Empresa			
	Buscar los Trabajadores y las Entidades del Negocio				
	Definir la Arquitectura del Negocio				
	Definir las Necesidades de Automatización				
	Detallar los Caso de Uso del Negocio				
	Estructurar el Modelo de Casos Uso del negocio				

PLANTILLA ROLES Y ACTIVIDADES					
	Mantener las Reglas de Negocio				
Modelo de Análisis del negocio	Definir la Arquitectura del Negocio	Buscar los Trabajadores y las Entidades del Negocio			
	Definir las Necesidades de Automatización	Definir la Arquitectura del Negocio			
	Detallar las Entidades del negocio				
	Detallar los trabajadores del negocio.				
	Mantener las Reglas de Negocio				
Reglas del negocio	Detallar las Entidades del negocio	Mantener las Reglas de Negocio			
	Detallar los trabajadores del negocio.				
Modelo de Casos de Uso del Negocio	Buscar los Trabajadores y las Entidades del Negocio	Buscar los Trabajadores y las Entidades del Negocio			
	Definir la Arquitectura del Negocio	Definir la Arquitectura del Negocio			
	Definir las Necesidades de Automatización				
	Detallar los Caso de Uso del Negocio				
	Estructurar el Modelo de Casos Uso del negocio				
	Identificar los Objetivos del Negocio				

PLANTILLA ROLES Y ACTIVIDADES					
	Mantener las Reglas de Negocio				
Ejecución de los Casos de Uso del Negocio	Detallar las Entidades del negocio	Buscar los Trabajadores y las Entidades del Negocio			
	Detallar los trabajadores del negocio.				
Especificación Suplementaria del Negocio	Buscar los Trabajadores y las Entidades del Negocio	Buscar los Actores y Casos de Uso del negocio			
	Definir la Arquitectura del Negocio	Detallar los Caso de Uso del Negocio			
	Definir las Necesidades de Automatización				
	Detallar las Entidades del negocio				
	Detallar los trabajadores del negocio.				
	Estructurar el Modelo de Casos Uso del negocio				
	Mantener las Reglas de Negocio				

D.2.13. Roles y actividades necesarios para la disciplina (Implementación)

PLANTILLA ROLES Y ACTIVIDADES					
Documento: Plantilla Roles y actividades necesarios para las disciplinas					
Propósito: Esté artefacto tiene el propósito de recopilar los roles y actividades indispensables para producir los artefactos del proceso adaptado					
Nombre del Proyecto: Aplicación web para compra y venta de Productos					
Encargado(s):					
Disciplina: Implementación					
ARTEFACTOS, ACTIVIDADES Y ROLES NECESARIOS				ACTIVIDADES Y ROLES A ELIMINAR	
ARTEFACTO	ACTIVIDADES		ROLES	ACTIVIDADES A ELIMINAR	ROLES A ELIMINAR
	Entrada a la Actividad	Salida de la Actividad			
Construcción	Ejecute el grupo de pruebas	Integrar el sistema	Arquitecto Software		
	Implemente grupos de pruebas	Integrar el subsistema	Implementador		
	Implemente la prueba		Integrador		
Construir el plan de integración	Identifique las metas de la prueba	Plan de integración del sistema	Revisor técnico		
	Integrar el sistema	Plan de integración de subsistemas			
	Integrar el subsistema				
	Plan de integración del sistema				
	Plan de integración de subsistemas				
Documento de arquitectura de software	Defina configuraciones de ambiente de prueba	Estructurar del Modelo de Implementación			
	Defina el enfoque de prueba				
	Identificar motivadores de pruebas				

PLANTILLA ROLES Y ACTIVIDADES					
	Identifique las ideas de prueba				
	Identifique las metas de la prueba				
	Identifique mecanismos de comprobabilidad				
	Implementar elementos de diseño				
	Obtenga el compromiso de Comprobabilidad				
Elementos de implementación	Analizar el comportamiento de tiempo de ejecución	Implementar elementos de diseño			
	Ejecutar pruebas del desarrollador				
	Implementar elementos de comprobabilidad				
	Implementar elementos de diseño				
	Implementar prueba del desarrollador				
	Integrar el subsistema				
	Plan de integración de subsistemas				
	Revisión de código				
Modelo de Implementación	Estructurar del Modelo de Implementación	Estructurar del Modelo de Implementación			
	Identifique las ideas de prueba				

PLANTILLA ROLES Y ACTIVIDADES					
	Identifique las metas de la prueba				
	Implemente grupos de pruebas				
	Plan de integración del sistema				
	Plan de integración de subsistemas				
Subsistema de implementación	Implementar elementos de comprobabilidad	Estructurar del Modelo de Implementación			
	Integrar sistema el	Implementar elementos de diseño			
	Integrar subsistema el	Integrar el subsistema			
	Plan de integración de subsistemas				

D.2.14. Roles y actividades necesarios para la disciplina (Pruebas)

PLANTILLA ROLES Y ACTIVIDADES					
Documento: Plantilla Roles y actividades necesarios para las disciplinas					
Propósito: Esté artefacto tiene el propósito de recopilar los roles y actividades indispensables para producir los artefactos del proceso adaptado					
Nombre del Proyecto: Aplicación web para compra y venta de Productos					
Encargado(s):					
Disciplina: Pruebas					
ARTEFACTOS, ACTIVIDADES Y ROLES NECESARIOS				ACTIVIDADES Y ROLES A ELIMINAR	
ARTEFACTO	ACTIVIDADES		ROLES	ACTIVIDADES A ELIMINAR	ROLES A ELIMINAR
	Entrada a la Actividad	Salida de la Actividad			
Arquitectura de automatización de prueba	Defina configuraciones de ambiente de prueba	Defina elementos de comprobabilidad	Diseñador de pruebas	Identifique las ideas de prueba	Director de pruebas
	Defina elementos de comprobabilidad	Identifique mecanismos de comprobabilidad	Analista de pruebas	Identifique las metas de la prueba	
	Defina el enfoque de prueba		Probador		

PLANTILLA ROLES Y ACTIVIDADES					
	Estar de acuerdo sobre la misión				
	Estructure la implementación de pruebas				
	Identifique mecanismos de comprobabilidad				
	Implemente grupos de pruebas				
	Implemente la prueba				
	Obtenga el compromiso de Comprobabilidad				
Caso de Prueba	Analice el fracaso de pruebas	Defina los detalles de prueba			
	Defina elementos de comprobabilidad				
	Defina los detalles de prueba				
	Determine los resultados de prueba				
	Estructure la implementación de pruebas				
	Implemente la prueba				
Datos de prueba	Analice el fracaso de pruebas	Defina los detalles de prueba			
	Defina configuraciones de ambiente de prueba				
	Defina elementos de comprobabilidad				
	Defina los detalles de prueba				
	Ejecute el grupo de pruebas				
	Estructure la implementación de pruebas				
	Implemente grupos de pruebas				
	Implemente la prueba				
Plan de pruebas	Calcule y mejore el esfuerzo de prueba	Calcule y mejore el esfuerzo de prueba			

PLANTILLA ROLES Y ACTIVIDADES					
	Defina configuraciones de ambiente de prueba	Defina el enfoque de prueba			
	Defina el enfoque de prueba	Defina las necesidades de valoración y trazabilidad			
	Defina las necesidades de valoración y trazabilidad	Estar de acuerdo sobre la misión			
	Ejecute el grupo de pruebas	Identificar motivadores de pruebas			
	Estar de acuerdo sobre la misión	Obtenga el compromiso de Comprobabilidad			
	Estructure la implementación de pruebas				
	Identificar motivadores de pruebas				
	Identifique las ideas de prueba				
Registro de Pruebas	Analice el fracaso de pruebas	Ejecutar pruebas del desarrollador			
	Analizar el comportamiento de tiempo de ejecución	Ejecute el grupo de pruebas			
	Determine los resultados de prueba				
Resultados de prueba	Calcule y apoye la calidad	Analizar el comportamiento de tiempo de ejecución			
	Calcule y mejore el esfuerzo de prueba	Determine los resultados de prueba			

D.3. Anexos ambiente empresarial

D.3.1. Empresa INPUT Technologies

QUIÉNES SOMOS

Somos una empresa creada con el fin de brindar soluciones tecnológicas integrales que contribuyan al mejoramiento continuo de los procesos y faciliten la obtención de la calidad de las empresas en general. Para lo cual contamos con un talento humano idóneo y comprometido con amplia experiencia en procesos de calidad.

MISIÓN

Proporcionar soluciones tecnológicas, económicas, flexibles y de excelente calidad que contribuyan al mejoramiento continuo de los procesos y al desarrollo, obtención y mantenimiento de la calidad de las empresas en general, permitiéndoles ser más productivas y eficientes en el cumplimiento de sus objetivos empresariales mediante la optimización de sus procesos internos y externos.

VISIÓN

Ser una empresa exitosa y la primera y mejor opción de soluciones tecnológicas que apoyen los procesos de calidad de las empresas en general.

D.3.2 Proyecto Compromiso

COMPROMISO es una herramienta software basada en las normas ISO 9000 con el objetivo de apoyar el sistema de gestión de la calidad, brindando la posibilidad de hacer un seguimiento completo de los procesos y requisitos necesarios para cumplir con dicho estándar. Este producto va dirigido a las empresas certificadas, empresas en proceso de certificación o empresas que busquen organizar sus procesos internos como la comunicación y divulgación del Sistema de Gestión de Calidad, Administración de la Documentación, Seguimiento de No Conformidades y Gestión de Equipos (equipos de computo, maquinaria, instrumentos de medición y demás).

El sistema se encuentra en etapa de desarrollo, y una vez terminado contara con 9 módulos: Comunicación interna, Documentos, Actas de calidad, Recurso humano, No conformidades, Objetivos de calidad, Auditoria, Gestión de equipos, Configuración; de los cuales se encuentran implementados hasta el momento el modulo de configuración, documentos, gestión de equipos, no conformidades y comunicación interna (sujetos a posibles cambios y ajustes).

Características

1. Novedoso sistema basado en los estándares internacionales de calidad ISO 9000.
2. Permite llevar un total control de cada uno de los requisitos exigidos por la norma ISO 9001:2000 para implementación del sistema de gestión de la calidad.

3. Permite una fácil comunicación entre empleados y jefes de la organización.
4. Permite a las empresas adquirir la funcionalidad adecuada a sus necesidades debido al desarrollo por módulos de la aplicación.
5. Permite realizar una completa configuración de acuerdo a los lineamientos, políticas de cada empresa gracias al modulo de configuración que se incluye como adicional a cada uno de los otros módulos de la aplicación.

D.3.3. Modulo de actas y reuniones de calidad

1. Descripción del Proyecto (Módulo Actas).

OBJ-1	Desarrollar Modulo de Actas y Reuniones de calidad.
Descripción	Este modulo de Actas y Reuniones permite registrar y mantener actualizadas y disponibles las actas del comité de calidad y reuniones en general, además lleva control de la reunión durante el desarrollo de la misma siguiendo cada uno de los temas a tratar en el orden del día y las opiniones, aportes, actividades y tareas que surjan en cada uno de ellos. También permite almacenar las tareas generadas en cada una de las reuniones y realizar seguimiento de aquellas que están pendientes.
Subobjetivos	OBJ-1.1 Gestionar Reuniones. OBJ-1.2 Gestionar Asistencias. OBJ-1.3 Gestionar Actividades. OBJ-1.4 Gestionar Contactos Externos. OBJ-1.5 Gestionar Tipos de Reunión. OBJ-1.6 Gestionar Correctivos. OBJ-1.7 Gestionar Tipos de Correctivos.

2. Objetivos Específicos.

OBJ-1.1	Gestionar Reuniones
Descripción	(PRE-reunión) El sistema permite realizar gestión de invitaciones (generación, edición, cancelación, almacenamiento) a todos los usuarios del sistema discriminando aspectos como: Tema, consecutivo, fecha, lugar, tipo de reunión, temas o aspectos a tratar en la reunión, además las invitaciones pueden ser enviadas vía internet. Además, el sistema permite que quien citó o elaboró la reunión pueda aplazarla o cancelarla en caso de inconvenientes como poca asistencia por parte de los invitados, etc.
	(IN-reunión) El sistema permite desarrollar de manera ordenada la reunión, permitiendo lectura de acta anterior, toma de asistencia, ultimas modificaciones al orden del día, desarrollo de opiniones y asignación de actividades y correctivos.

OBJ-1.1	Gestionar Reuniones
	(POST-reunión) El Sistema permite realizar seguimiento a las actividades asignadas, asignación de nuevas actividades, visualización y generación de Actas.

OBJ-1.2	Gestionar Asistencias
Descripción	(PRE-reunión) El sistema permite que quienes reciben una invitación, realicen confirmación de asistencia brindándoles a estos la posibilidad de hacer aportes al la reunión, subir archivos y delegar un suplente. En caso de que el usuario no pueda asistir a la reunión, el sistema solicita una justificación por la no asistencia a la esta y una fecha tentativa en la que el usuario podría asistir.

OBJ-1.3	Gestionar Actividades
Descripción	(IN-reunión) El sistema permite durante el desarrollo de la reunión la asignación de actividades o tareas a los diferentes asistentes de la reunión. (POST-reunión) El sistema permite consultar el estado de las actividades vez finalizada la reunión. Permite también la asignación de nuevas actividades mientras el acta o reunión no haya sido aún aprobada y permite a los encargados de las actividades establecer la fecha en la que la actividad es ejecutada.

OBJ-1.4	Gestionar Contactos Externos
Descripción	El sistema permite registrar, editar y eliminar contactos o personas externas al sistema, para permitir hacerles invitaciones a las diferentes reuniones que se realizan.

OBJ-1.5	Gestionar Tipos de Reunión.
Descripción	El sistema permite adicionar, editar y eliminar tipos de reunión por parte del usuario administrador del sistema.

OBJ-1.6	Gestionar Correctivos
Descripción	(IN-Reunión) El sistema permite hacer asignación de correctivos o sanciones a usuarios del sistema durante la reunión.

OBJ-1.7	Gestionar Tipos de Correctivos
Descripción	El sistema permite adicionar, editar y eliminar sanciones o correctivos que luego son asignados a usuarios que cometen faltas en al proceso de desarrollo de reuniones.

3. Tamaño Equipo.

3 participantes en el proyecto.

A continuación se presentan los participantes del proyecto:

- Diego Fernando Saavedra.
- Edison Sandoval.
- Alejandro Cardozo.
- Ricardo Ledezma.

Los integrantes del proyecto no cuentan con ningún tipo de experiencia en el desarrollo de proyectos aplicando formalmente UP; pero hay que resaltar que cuentan con experiencia (cuatro proyectos) en el desarrollo de este tipo de proyectos. Los integrantes cuentan con experiencia técnica, es decir, el manejo de herramientas y/o lenguajes entre estos PHP4 – Mysql –Apache- Javascript.

4. Distribución geográfica.

Oficina Popayán.

5. Criticidad del proyecto.

Daños Moderados.

6. Presupuesto

El presupuesto del proyecto se fijo en seis millones de pesos (\$ 6.000.000)

Nota: la prioridad del proyecto es la calidad, no existe una exigencia de tiempo de parte de un cliente, lo cual no quiere decir que su desarrollo pueda retrasarse sin justificación.

D.3.4. Niveles de las variables de caracterización de los proyectos

CARACTERIZACIÓN DE PROYECTOS		
Documento: Plantilla de Caracterización de Proyectos.		
Propósito: Esta plantilla tiene el propósito de facilitar la ejecución de la caracterización del proyecto, en su contenido están los niveles de clasificación de las variables de caracterización, los cuales deben asignar de acuerdo a las particularidades del proyecto y de la organización.		
Nombre del Proyecto: Modulo de actas y reuniones de calidad		
Encargado(s):		
Fecha:		
NIVEL		1. Tamaño del equipo
1	X	Muy pequeño: 1-10 personas
2		Pequeño: 11-50 personas
3		Medianas: 51-200 personas
NIVEL		2. Distribución geográfica del equipo
1	X	Misma Oficina
2		Mismo edificio, diferentes oficinas
3		Misma ciudad, misma empresa, diferente edificio

CARACTERIZACIÓN DE PROYECTOS		
4		Misma ciudad, diferente empresa
5		Diferente ciudad
3. Experiencia del equipo		
NIVEL		3.1 Experiencia en el proceso
1	X	No hay ningún proyecto
2		1 proyecto
3		2 a 3 proyectos
4		4 a 5 proyectos
5		Más de 5 proyectos
NIVEL		3.2 Experiencia en el dominio de la aplicación
1		No hay ningún proyecto
2		1 proyecto
3		2 a 3 proyectos
4	X	4 a 5 proyectos
5		Más de 5 proyectos
NIVEL		3.3 Experiencia técnica
1		No hay ningún proyecto
2		1 proyecto
3		2 a 3 proyectos
4		4 a 5 proyectos
5	X	Más de 5 proyectos
NIVEL		4. Criticidad del proyecto
1		Bajo desempeño
2		Pérdidas bajas (pérdidas fácilmente recuperables)
3	X	Daños moderados
4		Pérdidas grandes (pérdidas irrecuperables)
5		Riesgo de perder vidas
NIVEL		5. Tamaño del Proyecto
1		Hasta \$ 5'000.000
2	X	Entre \$ 5'000.000 y \$ 15'000.000
3		Entre \$ 15'000.000 \$ 25'000.000
4		Entre \$ 25'000.000 y \$ 35'000.000
5		Más de \$ 35'000.000
NIVEL		6. Características que priorizan el proyecto
1	X	Calidad = 90, tiempo = 10
2		Calidad = 70, tiempo = 30
3		Calidad = 50, tiempo = 50
4		Calidad = 30, tiempo = 70
5		Calidad = 10, tiempo = 90

D.3.5. Artefactos que serán adaptados en la disciplina de Requerimientos

ATEFACTOS QUE SERÁN ADAPTADOS
Objetivo: Esta plantilla permite realizar la selección de los artefactos, de la correspondiente disciplina, que pueden ser adaptados.
Nombre del proyecto: Modulo de actas y reuniones de calidad

ATEFACTOS QUE SERÁN ADAPTADOS		
Disciplina: Requerimientos		
Encargado:		
Fecha :		
Artefacto	Seleccionado (si/no)	Comentario
Peticiones de los Stakeholders	Si	Puede ser adaptado. No hay normas o modelos de procesos
Plan de gestión de Requerimientos	SI	Puede ser adaptado. No hay normas o modelos de procesos
Visión	No	No puede ser adaptado.
Glosario	SI	Puede ser adaptado. No hay normas o modelos de procesos
Modelo de Casos de Uso	No	No Puede ser adaptado.
Atributos de los Requerimientos	SI	Puede ser adaptado. No hay normas o modelos de procesos
Storyboard	SI	Puede ser adaptado. No hay normas o modelos de procesos
Especificación Suplementarias	Si	Puede ser adaptado. No hay normas o modelos de procesos
Caso de Uso	SI	Puede ser adaptado. No hay normas o modelos de procesos
Especificación de los requerimientos del software	SI	Puede ser adaptado. No hay normas o modelos de procesos
Paquetes de casos de uso	SI	Puede ser adaptado. No hay normas o modelos de procesos
Requerimientos software	No	No puede ser adaptado.

D.3.6. Artefactos que serán adaptados en la disciplina de Análisis y Diseño.

ATEFACTOS QUE SERÁN ADAPTADOS		
Objetivo: Esta plantilla permite realizar la selección de los artefactos, de la correspondiente disciplina, que pueden ser adaptados.		
Nombre del proyecto: Modulo de actas y reuniones de calidad		
Disciplina: Análisis y Diseño		
Encargado:		
Fecha :		
Artefacto	Seleccionado (si/no)	Comentario
Análisis de clases	SI	
Arquitectura de referencia	SI	
Capsula	SI	
Clases de pruebas	SI	
Diseño de clases	SI	
Diseño de pruebas	SI	

ATEFACTOS QUE SERÁN ADAPTADOS		
Diseño de subsistemas	SI	
Documento de la arquitectura	SI	
Ejecución de los casos de uso	SI	
Evento	SI	
Interface	SI	
Mapa de navegación	SI	
Modelo de Análisis	SI	
Modelo de datos	SI	
Modelo de despliegue	SI	
Modelo de diseño	SI	
Paquete de diseño	SI	
Protocolo	SI	
Prototipo de interface de usuario	SI	
Prueba de concepto arquitectural	SI	
Señal	SI	

D.3.7. Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización (Requerimientos)

TRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN								
Documento: Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización								
Propósito: En esta matriz se recopilan los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina y se establecen los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina.								
Nombre del Proyecto: Modulo de actas y reuniones de calidad								
Encargado(s):								
	Variables de Caracterización							
	Tamaño del equipo	Distribución geográfica del equipo	Experiencia en el proceso	Experiencia en el dominio de la aplicación	Experiencia técnica	Criticidad del proyecto	Tamaño del proyecto	Características que priorizan
Disciplina: Requerimientos								
ARTEFACTOS (que se pueden adaptar)	Niveles	Niveles	Niveles	Niveles	Niveles	Niveles	Niveles	Niveles
	1	1	1	4	5	3	2	1
	Muy pequeño: 1-10 personas	Misma Oficina	No hay ningún proyecto	4 a 5 proyectos	Más de 5 proyectos	Daños moderados	Entre \$ 5'000.000 y 15'000.00	Calidad = 90, tiempo = 10
Peticiones de los Stakeholders	I	I	SR	I	I	SR	I	SP
Plan de gestión de Requerimientos	I	I	SR	I	I	SR	I	SP
Glosario	NP	NP	NP	NP	NP	NP	NP	SP
Atributos de los Requerimientos	NP	NP	NP	NP	NP	NP	NP	SP
Storyboard	NP	NP	NP	NP	NP	NP	NP	SP

TRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN								
Especificación Suplementarias	NP	NP	NP	NP	NP	NP	NP	SP
Caso de Uso	SP	SP	SP	SP	SP	SP	SP	SP
Especificación de los requerimientos del software	SR	SR	SR	SR	SR	SR	SR	SP
Paquetes de casos de uso	I	I	SR	SR	I	SR	I	SP

D.3.8. Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización (Análisis y Diseño)

CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN								
Documento: Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización								
Propósito: En esta matriz se recopilan los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina y se establecen los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina.								
Nombre del Proyecto: Modulo de actas y reuniones de calidad								
Encargado(s):								
Disciplina: Análisis y Diseño	Variables de Caracterización							
	Tamaño del equipo	Distribución geográfica del equipo	Experiencia en el proceso	Experiencia en el dominio de la aplicación	Experiencia técnica	Criticidad del proyecto	Tamaño del Proyecto	Características que priorizan el Proyecto
ARTEFACTOS (que se pueden adaptar)	Nivel	Nivel	Nivel	Nivel	Nivel	Nivel	Nivel	Nivel
	1	1	1	4	5	3	2	1
	Muy pequeño: 1-10 personas	Misma Oficina	No hay ningún proyecto	4 a 5 proyectos	Mas de 5 proyectos	Daños Moderados	Entre \$ 5'000.000 y 15'000.00	Calidad = 90, tiempo = 10
Análisis de clases	I	I	NP	NP	NP	SR	I	NP
Arquitectura de referencia	NP	NP	SR	SR	SR	SR	NP	I
Capsula	I	I	I	NP	NP	NP	NP	NP
Clases de pruebas	I	I	SR	SP	SP	SR	SR	SR

CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN								
Diseño de clases	SR	SR	SP	SP	SR	SR	SP	SP
Diseño de pruebas	I	I	SR	SR	SR	SR	SR	SR
Diseño de subsistemas	I	I	SR	NP	NP	NP	SR	SR
Documento de la arquitectura	SR	SR	I	SR	SR	I	I	SR
Ejecución de los casos de uso	I	I	SR	SR	SR	SR	SR	I
Evento	I	I	I	NP	NP	NP	NP	NP
Interface	I	I	I	SR	SR	SR	NP	NP
Mapa de navegación	I	I	I	SR	SR	SR	SR	SR
Modelo de Análisis	I	I	SR	SR	SR	NP	NP	NP
Modelo de datos	SR	SR	SR	SR	SR	SR	SR	SR
Modelo de despliegue	NP	NP	NP	NP	NP	NP	NP	NP
Modelo de diseño	SR	I	I	SR	SR	SR	SR	SR
Paquete de diseño	SR	SR	I	SR	SR	SR	SR	SR
Protocolo	NP	NP	NP	NP	NP	NP	NP	NP
Prototipo de interface de usuario	I	I	SR	SR	SR	SR	SR	SR
Prueba de concepto arquitectural	I	I	SR	SR	SR	SR	SR	SR
Señal	NP	NP	NP	NP	NP	NP	NP	NP

D.3.9. Matriz final de clasificación de artefactos por disciplina (Requerimientos)

MATRIZ FINAL DE CLASIFICACIÓN ARTEFACTOS			
Documento: Matriz Final de clasificación de artefactos por disciplinas			
Propósito: En esta matriz se recopilan la clasificación final de cada artefacto con respecto a las variables de caracterización y se adicionan los artefactos que fueron seleccionados como no adaptables			
Nombre del Proyecto: Modulo de actas y reuniones de calidad.			
Encargado(s):			
Disciplina: Requerimientos			
ARTEFACTOS (que se pueden adaptar)		Clasificación final	Comentario
1	Peticiones de los Stakeholders	I	
2	Plan de gestión de Requerimientos	I	
3	Glosario	I	
4	Atributos de los Requerimientos	I	
5	Storyboard	I	
6	Especificación Suplementarias	I	
7	Caso de Uso	SP	Este artefacto se producirá
8	Especificación de los requerimientos del software	SR	
9	Paquetes de casos de uso	SR	Se definen paquetes de casos de uso solo cuando determinar la funcionalidad del sistema sea compleja y desde el punto de vista de visualización sea más comprensible para los involucrados en el proyecto
10	Visión	Obligatorio	Artefacto seleccionado como no adaptable. Se produce de manera obligatoria.
11	Modelo de Casos de Uso	Obligatorio	Artefacto seleccionado como no adaptable. Se produce de manera obligatoria.
12	Requerimientos software	Obligatorio	Artefacto seleccionado como no adaptable. Se produce de manera obligatoria.

D.3.10. Matriz final de clasificación de artefactos por disciplina (Análisis y Diseño)

MATRIZ FINAL DE CLASIFICACIÓN ARTEFACTOS			
Documento: Matriz Final de clasificación de artefactos por disciplinas			
Propósito: En esta matriz se recopilan la clasificación final de cada artefacto con respecto a las variables de caracterización y se adicionan los artefactos que fueron seleccionados como no adaptables			
Nombre del Proyecto: Modulo de actas y reuniones de calidad			

MATRIZ FINAL DE CLASIFICACIÓN ARTEFACTOS			
Encargado(s):			
Disciplina: Análisis y diseño			
ARTEFACTOS (que se pueden adaptar)		Clasificación final	Comentario
1	Análisis de clases	NP	
2	Arquitectura de referencia	SR	
3	Capsula	NP	
4	Clases de pruebas	SR	
5	Diseño de clases	SR	
6	Diseño de pruebas	SR	
7	Diseño de subsistemas	NP	
8	Documento de la arquitectura	SR	
9	Ejecución de los casos de uso	SR	
10	Evento	NP	
11	Interface	NP	
12	Mapa de navegación	SP	
13	Modelo de Análisis	NP	
14	Modelo de datos	SR	
15	Modelo de despliegue	NP	
16	Modelo de diseño	SR	
17	Paquete de diseño	SR	
18	Protocolo	NP	
19	Prototipo de interface de usuario	SR	
20	Prueba de concepto arquitectural	SR	
21	Señal	NP	

D.3.11. Roles y actividades necesarios para la disciplina (Requerimientos)

Plantilla Roles y actividades				
Documento: Plantilla Roles y actividades necesarios para las disciplinas				
Propósito: Esté artefacto tiene el propósito de recopilar los roles y actividades indispensables para producir los artefactos del proceso adaptado				
Nombre del Proyecto: Modulo de actas y reuniones de calidad				
Encargado(s):				
Disciplina: Requerimientos				
ARTEFACTOS, ACTIVIDADES y ROLES NECESARIOS			ACTIVIDADES Y ROLES A ELIMINAR	
ARTEFACTO	ACTIVIDADES	ROLES	ACTIVIDADES A	ROLES A

Plantilla Roles y actividades					
				ELIMINAR	ELIMINAR
	Entrada a la Actividad	Salida de la Actividad			
Caso de uso	Análisis de Casos de Uso	Buscar actores y casos de uso	Analista de sistemas	Describir la Arquitectura en Tiempo de Ejecución	
	Capturar el vocabulario común	Detallar un caso de uso	Especificador de requerimientos	Describir la Distribución	
	Defina los detalles de prueba	Estructurar el modelo de casos de uso		Estructurar del Modelo de Implementación	
	Detallar los requerimientos del software			Identificar Elementos del Diseño	
	Detallar un caso de uso			Identificar Mecanismos de Diseño	
	Diseñar la Interface de Usuario			Revisar la Arquitectura	
	Diseño de Casos de Uso			Defina las necesidades de valoración y trazabilidad	
	Estructurar el modelo de casos de uso			Desarrollo del Plan de gestión de requerimientos	
	Plan de integración del sistema				
	Plan de integración de subsistemas				
	Prototipo de Interface de Usuario				
	Revisión de Requerimientos				
Especificación de los requerimientos del software	Defina el enfoque de prueba	Detallar los requerimientos del software			
	Diseñar la Interface de Usuario				
	Identifique las ideas de prueba				
	Revisión de Requerimientos				
Paquetes de casos de uso	Estructurar el modelo de casos de uso	Estructurar el modelo de casos de uso			
	Revisión de Requerimientos				
Visión	Análisis	Desarrollo de la			

Plantilla Roles y actividades					
	Arquitectural	Visión			
	Buscar actores y casos de uso	Gestión de dependencias			
	Capturar el vocabulario común				
	Defina el enfoque de prueba				
	Desarrollo de la Visión				
	Detallar los requerimientos del software				
	Detallar un caso de uso				
	Diseñar la Interface de Usuario				
	Estar de acuerdo sobre la misión				
	Evaluar la viabilidad de la Prueba de Concepto Arquitectural.				
	Gestión de dependencias				
	Identificar motivadores de pruebas				
	Identifique las ideas de prueba				
	Priorizar los casos de Uso				
	Recibir las peticiones de los stakeholders				
	Revisión de Requerimientos				
Modelo de Casos de Uso	Análisis de Casos de Uso	Buscar actores y casos de uso			
	Capturar el vocabulario común	Estructurar el modelo de casos de uso			
	Definir el enfoque de prueba				
	Detallar los requerimientos del software				
	Detallar un caso de uso				
	Estructurar el modelo de casos de uso				
	Evaluar la viabilidad de la Prueba de				

Plantilla Roles y actividades					
	Concepto Arquitectural.				
	Gestión de dependencias				
	Identifique las ideas de prueba				
	Priorizar los casos de Uso				
	Revisar el Diseño				
	Revisión de Requerimientos				
Requerimientos software	Diseñar la Interface de Usuario	Detallar los requerimientos del software			
	Identificar motivadores de pruebas	Priorizar los casos de Uso			
	Priorizar los casos de Uso				
	Revisión de Requerimientos				

D.3.12. Roles y actividades necesarios para la disciplina (Análisis y Diseño)

PLANTILLA ROLES Y ACTIVIDADES					
Documento: Plantilla Roles y actividades necesarios para las disciplinas					
Propósito: Esté artefacto tiene el propósito de recopilar los roles y actividades indispensables para producir los artefactos del proceso adaptado					
Nombre del Proyecto: Modulo de actas y reuniones de calidad					
Encargado(s):					
Disciplina: Análisis y Diseño					
ARTEFACTOS, ACTIVIDADES Y ROLES NECESARIOS				ACTIVIDADES Y ROLES A ELIMINAR	
ARTEFACTO	ACTIVIDADES		ROLES	ACTIVIDADES A ELIMINAR	ROLES A ELIMINAR
	Entrada a la Actividad	Salida de la Actividad			
Arquitectura de referencia	Análisis Arquitectural	Evaluar la viabilidad de la Prueba de Concepto Arquitectural.	Diseñador	Diseño de Subsistemas	
Clases de pruebas	Implementar elementos de comprobabilidad		Diseñador de bases de Datos		
Diseño de clases	Diseño de Bases de Datos	Diseño de Capsula	Diseñador de Capsula		
	Diseño de Casos de Uso	Diseño de Clases	Diseñador de interface		
	Diseño de Clases	Diseño de Subsistemas	Diseñador de Pruebas		
	Diseño de Prueba de Elementos	Identificar Elementos del Diseño	Revisor Tecnico		

PLANTILLA ROLES Y ACTIVIDADES					
		Identificar Mecanismos de Diseño	Arquitecto software		
Diseño de pruebas	Implementar prueba del desarrollador	Defina elementos de comprobabilidad			
Documento de la arquitectura	Análisis Arquitectural	Análisis Arquitectural			
	Análisis de Casos de Uso	Describir la Arquitectura en Tiempo de Ejecución			
	Construir la Prueba de Concepto Arquitectural	Describir la Distribución			
	Describir la Arquitectura en Tiempo de Ejecución	Identificar Mecanismos de Diseño			
	Describir la Distribución	Incorporar Elementos existentes de Diseño			
	Identificar Elementos del Diseño				
	Identificar Mecanismos de Diseño				
	Incorporar Elementos existentes de Diseño				
	Revisar la Arquitectura				
Ejecución de los casos de uso	Análisis de Casos de Uso	Análisis de Casos de Uso			
	Definir Elementos de Prueba	Diseño de Casos de Uso			
	Diseño de Bases de Datos				
	Diseño de Casos de Uso				
	Diseño de Clases				
Mapa de navegación	Prototipo de Interface de Usuario	Diseñar la Interface de Usuario			
	Revisar el Diseño				
Modelo de datos	Identifique las ideas de prueba	Diseño de Bases de Datos			
	Identifique las metas de la prueba				

PLANTILLA ROLES Y ACTIVIDADES					
	Implementar elementos de diseño				
	Revisar el Diseño				
Prototipo de interface de usuario	Diseño de Clases	Prototipo de Interface de Usuario			
	Identifique las ideas de prueba				
	Revisar el Diseño				
Prueba de concepto arquitectural	Análisis Arquitectural	Construir la Prueba de Concepto Arquitectural			
	Evaluar la viabilidad de la Prueba de Concepto Arquitectural.				

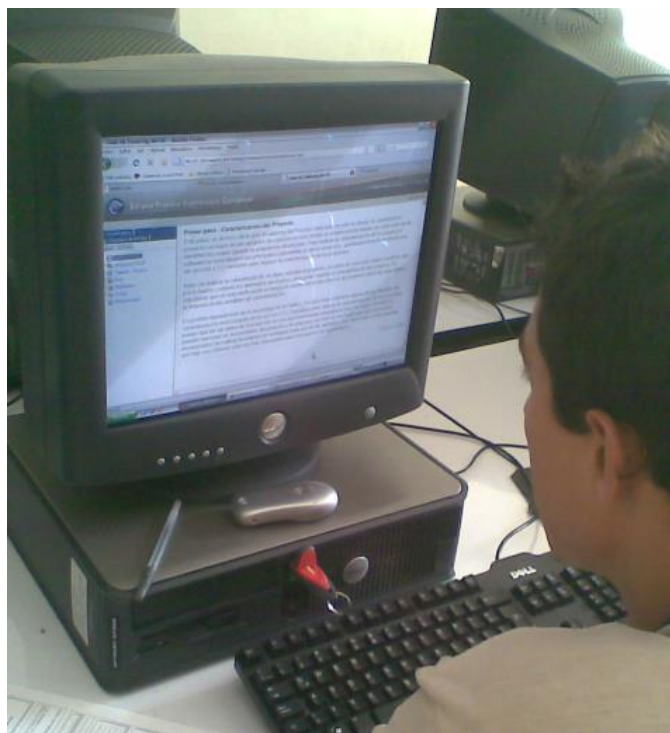
ANEXO E FOTOS APLICACIONES DE GTUP

E.1. Fotos ambiente académico













E.2. Fotos ambiente empresarial





ANEXO F CERTIFICADO INPUT TECHNOLOGIES LTDA.



INPUT TECHNOLOGIES LTDA.

NIT. 900076583-3

CERTIFICA

A SOLICITUD DEL INTERESADO QUE

Pablo Hernando Ruiz Melenje y Carlos Ernesto Maya Vallejo realizaron la experimentación de la Guía de Tailoring del Proceso Unificado a Proyectos Específicos para las Pymes Desarrolladoras de Software en un proyecto de la empresa, en el mes de Julio del año 2009.

Para constancia se firma el día 22 de septiembre de 2009

Ricardo Andrés Ledezma.

Ing. de Proyectos.

INPUT Technologies Ltda.



Carrera 15 # 5-41 Oficina 205 Popayán
Teléfonos: 8317533
E-M@IL: info@input.com.co