



Centro de Investigación en Matemáticas, A.C.

Maestría en Ingeniería de Software



Líneas de Productos, una guía para arrancar

María Eugenia Pérez Santacruz

Dr. Cuauhtémoc Lemús Olalde

CONTENIDO

1	Introducción	3
2.	Problemática a resolver	4
3.	Justificación	6
3.1	Framework BAPO	7
3.2	Áreas de una Línea de Productos.....	9
3.3	Administración de Desarrollo de Proyectos de Software de una Línea de Productos y Metodologías ágiles.	14
4.	Metodología Propuesta.....	20
5.	Conclusiones.....	26
5.1	Tendencias Futuras.....	26
5.2	Beneficios obtenidos	26
	Referencias	27

1 Introducción

Existen grandes beneficios si se adopta el enfoque de Líneas de Productos, se reducen costos, tiempos y tiempos de entrega al cliente y al mismo tiempo se aumenta la calidad de los productos que se desarrollan.

Pero hasta el día de hoy existen barreras organizacionales que impiden la adopción de este enfoque, y aunque son muy atractivas las ventajas que ofrece, no hay una receta de cocina que nos diga que debemos hacer y como lo debemos implementar para que la adopción del enfoque sea exitoso.

El objetivo de la adopción se basa en tener componentes base, procesos y estructuras organizacionales que soporten el desarrollo de productos desde el uso de componentes base para alcanzar los objetivos de un negocio y para fortalecer las organizaciones de desarrollo de software haciéndolas más competitivas para que puedan colocar de forma más rápida sus productos y que obtengan mayores ganancias.

Si no se tiene una guía que nos lleve a través de la adopción, esta puede llegar a volverse costosa e infructuosa, por lo tanto este trabajo presenta un conjunto de procesos realizados para poder adoptar y trabajar con líneas de productos, además se presenta un empate novedoso entre SCRUM la cual es una metodología de administración de desarrollo de proyectos de software y Líneas de productos, esto con la meta de fortalecer la parte de procesos de las Líneas de Productos para el CDS del Instituto Tecnológico Superior de Fresnillo.

En la Sección 2, revisaremos el problema que motivo este trabajo. En la Sección 3 se justifica el porqué del uso de las metodologías seleccionadas para crear este set de procesos y el empate entre SCRUM y Líneas de Productos. La metodología desarrollada se describe en la Sección 4 y la Sección 5 contiene una breve conclusión del trabajo realizado y presenta trabajo futuro a realizar en este ámbito. Además en el anexo A, se presentan los procesos creados para la adopción de Líneas de Productos.

2. Problemática a resolver

Dado los exigentes requerimientos en el cumplimiento de tiempos y presupuestos que las empresas desarrolladoras de software viven constantemente, la eficiencia en el proceso de desarrollo se ha constituido en una necesidad creciente. Sin embargo, ésta es una variable que no necesariamente propicia su efectividad.

La mayoría de las empresas de desarrollo, actualmente se dedican a construir aplicaciones a la medida y siempre desde cero, lo cual hace que se consuma mucho tiempo y haya gran cantidad de errores.

Debido a esto se han buscado otras estrategias de trabajo para poder realizar una industrialización en la construcción de software para de esta forma reducir tiempos y costos, esto como consecuencia de la globalización ya que las empresas deben ser más competitivas para atraer clientes y poder subsistir.

En los años anteriores se ha tenido un gran crecimiento en la adopción de diferentes metodologías o paradigmas para aumentar la efectividad, tales como la adopción de mejoramiento de procesos, lo cual es indispensable para el objetivo que se persigue pero tiene límites. Otra metodología que se ha tratado de implementar desde hace ya tiempo sin tener el éxito esperado es la reutilización (La mayoría de las veces se reduce a copiar y pegar módulos o partes de estos).

Como podemos ver la industrialización del software es un problema complejo en el cual se deben involucrar diferentes aspectos, por lo tanto los modelos que se adopten deben considerar los siguientes aspectos críticos [2]:

- **Reutilización y adaptación:** Elementos y estructuras adaptables, administración de la reutilización y mecanismos de adaptación.
- **Mecanismos de composición:** Métodos para ensamblar partes, con el grado de granularidad adecuado.
- **Lenguajes y frameworks especializados:** Arquitecturas con funcionalidad básica y lenguajes de dominio específico.
- **Focalización:** Alto grado de especialización y un dominio de trabajo específico.
- **Formación y especialización:** Personal especializado y actualización permanente.
- **Control de calidad y procesos:** Procesos y roles bien definidos y mejora continua de procesos.

Por lo tanto surgen varias preguntas dadas de esta situación, tales como:

- ¿Basta con utilizar bien lo que ya existe?
- ¿Es posible disminuir costos sin perder calidad?
- ¿Hay manera de industrializar el proceso de construcción de software?
- ¿Existe una metodología que cumpla con los aspectos anteriores?
- ¿Existe una guía de cómo realizar la industrialización del software?

El Instituto Tecnológico Superior de Fresnillo desde agosto de 2008 cuenta con un Centro de Desarrollo de Software (CDS) cuyo objetivo principal es:

“Generar soluciones de software y hardware que ayuden a los sectores público, privado y social. Así mismo formar recurso humano competitivo que se integre rápidamente a la industria de software”.

Durante su primer año de operaciones del CDS del Instituto Tecnológico Superior de Fresnillo, se han realizado proyectos a la medida para la comunidad tecnológica, así como algunos proyectos para el sector gobierno.

Uno de los problemas con los que cuenta es que los proyectos que se han realizado no generan ninguna ganancia económica para el CDS y por lo tanto no es autofinanciable, esto provoca que no se tenga personal de planta en el CDS y que únicamente se cuente con alumnos de residencia profesional los cuales prestan sus servicios durante un periodo de 6 meses.

La principal motivación para la realización de este trabajo es adoptar una metodología que permita al CDS del Instituto Tecnológico Superior de Fresnillo ser autofinanciable mediante la venta de los productos que se desarrollan ahí, pudiendo contratar a los alumnos para que su estancia sea mayor al tiempo de su residencia profesional y de esta forma reducir la rotación de personal que por el momento es un problema presente en el CDS.

3. Justificación

Un paradigma que cumple con algunas de las características para la industrialización del software es Líneas de Productos de Software (LPS). Podemos definir una LPS como “Un conjunto de sistemas de software que comparten un común y administrado conjunto de características que satisfacen las necesidades de un segmento de mercado particular y son desarrollados de un conjunto común de activos de software en una forma prescrita” [1].

Pero Líneas de Productos de Software por si solas no ayudan a cumplir los objetivos de la industria del software tales como productividad a gran escala, mejorar el tiempo de entrega, mantener presencia en el mercado, reducción de costos de producción, aumentar la calidad de los productos que se generan. A este paradigma es necesario añadir procesos y herramientas para que ayuden al logro de estos objetivos.

Derivado de esto desde ya hace varias décadas en la Ingeniería de Software surgió el término de “Fábrica de Software” la cual podemos definir como “Una línea de productos de software que configura contenido, procesos y herramientas extensibles utilizando una plantilla de fábrica de software basada en un esquema de fábrica de software para automatizar el desarrollo y mantenimiento de variantes de un producto arquetípico adaptando, ensamblando y configurando componentes basados en frameworks”. Para comprender un poco mejor esta definición, un esquema de fábrica de software describe los artefactos que se deben desarrollar para producir un producto de software y una plantilla de fábricas de software se entiende como el conjunto de patrones, frameworks y herramientas que permiten implementar el esquema para construir un miembro de la familia de productos[2].

Por lo tanto como vemos, una fábrica de software engloba las características necesarias para la industrialización del software. Pero no es tarea sencilla que una empresa de desarrollo de software ya sea que trabaje con el enfoque tradicional o bien una que inicie operaciones adopte de manera fácil y sencilla este paradigma ya que implica muchos aspectos.

En este documento se propone una guía de trabajo para la adopción que engloba Líneas de Productos de Software, así como el uso de procesos basados en metodologías ágiles para la administración de proyectos. Este trabajo será de utilidad para las diferentes organizaciones que deseen adoptar esta forma de trabajo y necesiten una guía para poder trabajar bajo este enfoque.

3.1 Framework BAPO

Cuando una organización decide realizar el cambio de construir productos uno a la vez al desarrollo de una línea de productos, o bien que desde su inicio decida hacer esto debe considerar diferentes aspectos.

Se han identificado 4 aspectos importantes de acuerdo con [4] que se deben considerar al momento de trabajar con Líneas de Productos de Software. Estos aspectos son Bussines, Architecture, Process y Organization los cuales se han agrupado en un Framework llamado BAPO. Este framework sirve para evaluar el estado de nuestra Línea de Productos de Software así como poder encontrar mejoras.

Cada aspecto se enfoca en diferentes actividades:

- **Bussines:** Se enfoca a obtener beneficio de los productos, es decir a todo lo referente al mercado.
- **Architecture:** Todo lo concerniente a la tecnología y tareas de ingeniería para construir los productos.
- **Process:** Roles, responsabilidades y sus relaciones, necesarias construir los productos de software.
- **Organization:** El mapeo de roles y responsabilidades en una estructura organizacional de trabajo.

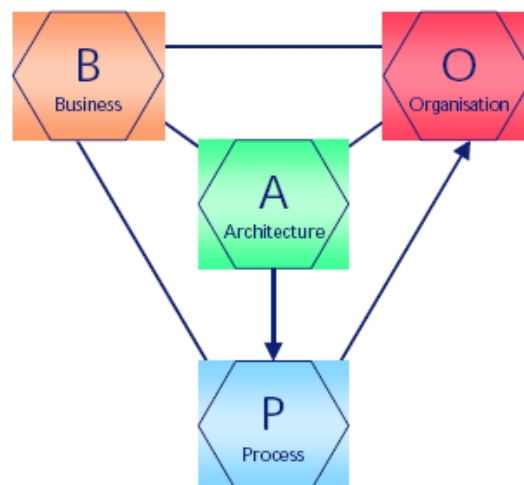


Fig. 3.1.1 Framework BAPO

La figura anterior representa los aspectos BAPO, las líneas representan las relaciones entre los aspectos, esto significa que si se realizan cambios en un aspecto este induce cambios en otros. Las flechas denotan una transición natural de aspectos.

Como se explicó anteriormente, el aspecto Business es el factor de mayor influencia, por lo tanto es el que se considera primero. La arquitectura refleja las preocupaciones del negocio en una estructura y reglas, el proceso permite construir los productos determinados por la arquitectura y finalmente la organización permite ejecutar el proceso.

La evaluación de la dimensión del negocio se enfoca a la habilidad de una organización para manejar y predecir el costo y el beneficio del desarrollo. El negocio es dependiente de la arquitectura, procesos y organización elegidos y el beneficio se relaciona con la satisfacción del cliente y la posición en el mercado. Cuando se ha elegido trabajar con líneas de productos, los objetivos de la organización deben alinearse con los objetivos de la línea de productos.

La arquitectura de una familia de productos significativamente difiere de una arquitectura para el desarrollo de un solo producto. Detectar esto es esencial, diseñar y modelar las partes variables y comunes de una familia de software. La administración de la variabilidad necesita comenzar desde la ingeniería de requerimientos. Las partes comunes se pueden manejar como una plataforma de reuso.

La dimensión de procesos enfatiza los procesos, roles, productos de trabajo y las correspondientes responsabilidades y relaciones con el desarrollo del software.

La dimensión de organización se enfoca en la forma en que la organización maneja las complejas relaciones y responsabilidades de la creación de artefactos para el desarrollo.

La mayoría de las veces cuando se quiere adoptar una metodología o paradigma, nos preocupamos por los aspectos técnicos y tecnológicos que nos ayudarán a la ejecución de la metodología. Pero pocas veces nos preocupamos por lo que soporta al desarrollo, la organización y el mercado.

3.2 Áreas de una Línea de Productos

Para alcanzar los objetivos de una línea de productos, es necesario poner en práctica ciertas actividades relevantes de forma continua, es decir no solo una vez sino que se puedan realizar de varias veces y de forma exitosa.

Un área de práctica es un conjunto de tareas que una organización debe ejecutar de forma exitosa para el trabajo con Líneas de productos de software. Las áreas prácticas ayudan a realizar actividades de forma más sencilla definiendo actividades más pequeñas que las tareas principales de una línea de productos como el desarrollo de productos. Las áreas prácticas proveen puntos de partida de los cuales una organización puede realizar y medir el progreso de adoptar el enfoque de Líneas de Productos.

Cabe mencionar que las áreas prácticas que se presentan, fueron obtenidas del Framework para Líneas de Productos elaborado por el SEI, en este framework se organizan las áreas prácticas en 3 grupos los cuales se describen enseguida [5].

1. Las áreas de ingeniería de software son aquellas necesarias para aplicar la tecnología apropiada para crear y evolucionar los componentes base y los productos.
2. Las áreas de administración técnica son aquellas necesarias para administrar la creación y evolución de los componentes base y los productos.
3. Las áreas de administración de la organización son aquellas necesarias para organizar el trabajo de la línea de productos.

Debido a que existen varias áreas de práctica, es necesaria una forma de organizarlas para una mejor referencia. En la agrupación que hace el SEI de las áreas prácticas, existen áreas prácticas de negocio en la parte tecnológica y viceversa y no quedan claras las responsabilidades de las personas que se involucran en las áreas de desarrollo con línea de productos. Es importante definir las responsabilidades de los involucrados por tal motivo se decidió agruparlas de acuerdo a los aspectos que contempla el framework BAPO, ya que de esta forma quedan más claras las responsabilidades que tendrán las personas que se involucrarán en cada una de los aspectos de una Línea de Productos.

La siguiente figura presenta las áreas de práctica en relación al framework BAPO.

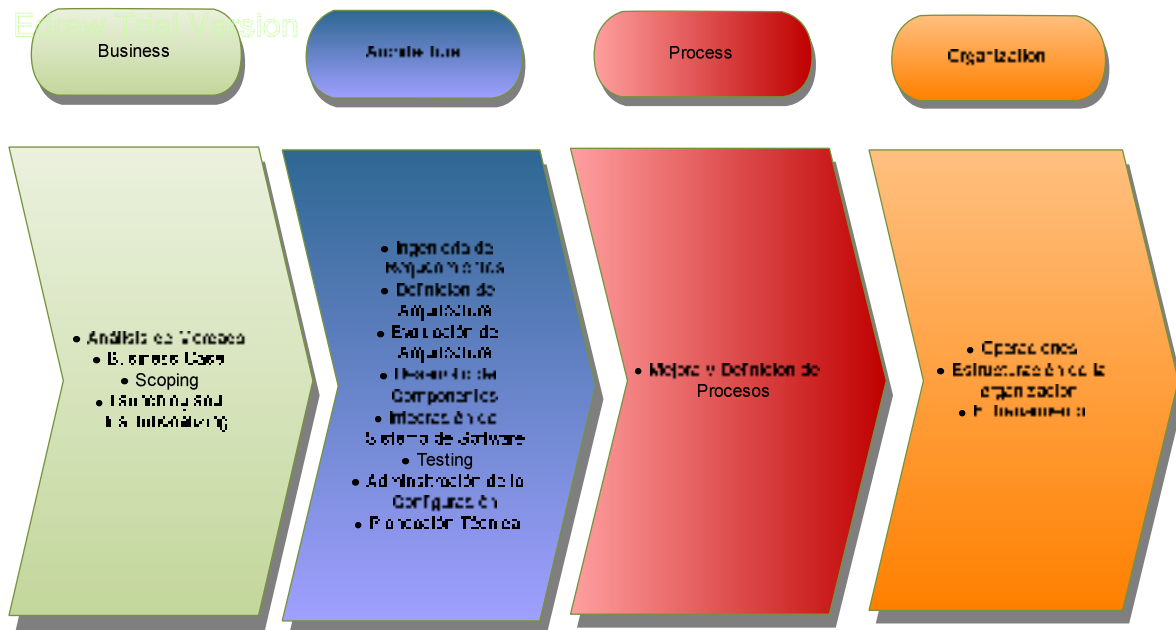


Fig. 3.2.1 Áreas de una LPS organizadas en BAPO

A continuación se definen cada una de las áreas de práctica para una línea de productos referentes al área de negocios.

- **Análisis de Mercado:** Es la investigación sistemática y el análisis de los factores externos que determinan el éxito de una línea de productos en un mercado específico. Involucra la obtención de estudios de competitividad, segmentación del mercado, estrategias para atraer clientes y la integración de la información para generar un plan y estrategia de negocio.
- **Business Case:** Un business case es una herramienta que ayuda a realizar decisiones de negocio, prediciendo como afectarán la organización. Inicialmente, la decisión será ir o no identificar nuevas oportunidades de negocio o nuevos enfoques.
- **Scoping:** Es una actividad que delimita un sistema o un conjunto de sistemas, definiendo las características o aspectos que se deben contener y lo que no se deben contener en el producto. Todo el desarrollo de un sistema involucra definición del alcance (scoping), no existe ningún sistema en el que se deseen todas las características. Kruchten define scoping como capturar el contexto y los requerimientos más importantes y restricciones que debes considerar para los criterios de aceptación del producto final.

- **Launching and Institutionalizing:** Actividad que ayuda a preparar una organización para la adopción de una nueva tecnología o en una nueva forma de hacer negocios. Estos proyectos son altamente dependientes del contexto de la organización.

A continuación se definen cada una de las áreas de práctica para una línea de productos referentes al área arquitectura.

- **Ingeniería de Requerimientos:** La parte más difícil de un sistema de software es decidir precisamente que construir. Los requerimientos son sentencias que indican que debe hacer el sistema y como debe comportarse. Podemos entender a los requerimientos como características o condiciones necesarias para un usuario con el objetivo de resolver un problema o alcanzar un objetivo.
- **Definición de Arquitectura:** Esta práctica describe las actividades que deben realizarse para definir una arquitectura de software. Podemos describir una arquitectura de software como una estructura o estructuras de un sistema, que contiene los elementos de software, sus relaciones y sus propiedades externas. La arquitectura es la clave del éxito de cualquier proyecto de software, Es el primer artefacto de diseño que comienza a involucrar lo requerimientos en una solución.
- **Evaluación de Arquitectura:** La arquitectura es uno de los primeros artefactos de diseño que mapea los objetivos de calidad del sistema, tales como seguridad, confiabilidad, usabilidad, etc. La arquitectura describe la estructura del sistema y sirve como un vehículo de comunicación entre los stakeholders del sistema: desarrolladores, administradores y mantenedores, usuarios, clientes, testers, mercadotecnicos y cualquier otra persona que tenga intereses en el desarrollo o uso del sistema, debido a esto es necesario evaluar la arquitectura para determinar su correctitud y completitud para que el producto se desarrolle de forma correcta.
- **Desarrollo de componentes:** Una de las tareas que se genera de la arquitectura, es la lista de componentes que se deberán crear o utilizar para crear el producto. Esta área práctica tiene que ver con el desarrollo in-house y en cómo construir los componentes que se necesitan, esto lo podemos encontrar en la arquitectura. El desarrollo de componentes se enfoca en proveer el software como sea necesitado por los productos a ser desarrollados. Los componentes resultantes se añaden a los componentes base y se pueden utilizar en diferentes

productos de la línea, o bien son componentes específicos para ciertos productos. Los componentes que se incluyen en los componentes base, deben soportar flexibilidad y satisfacer los puntos de variación especificados en la arquitectura de la línea de productos.

- **Integración del Sistema de Software:** Esta área se refiere a la práctica de combinar individualmente los componentes de software probados como un todo. El software es integrado cuando los componentes son combinados en subsistemas o cuando los subsistemas son combinados en productos. Los componentes pueden integrarse después de que todo ha sido probado en un modelo de cascada. Una integración continua es menos arriesgada ya que los componentes son integrados como son desarrollados y de esta forma se obtienen varias versiones del sistema.
- **Testing:** El área de pruebas tiene dos funciones: 1) Ayudar a identificar errores que deriven en fallas y que estos errores puedan ser reparados y 2) Determinar cuando el software bajo pruebas puede ser ejecutado como especifican los requerimientos. Probar en una línea de productos examina los componentes base del software y el producto específico de software, las interacciones entre ellos y los productos completos. Las responsabilidades de pruebas en una organización con el enfoque de línea de productos deben ser distribuidas de forma diferente que en la construcción de un software tradicional.
- **Administración de la configuración:** Es un área que se refiere a evaluar, coordinar, aprobar y desaprobado, implementar cambios en artefactos que son usados para construir y mantener los sistemas de software. Un artefacto puede ser una pieza de hardware o software o documentación. La administración de la configuración permite la administración de artefactos desde el concepto inicial a través de todo el ciclo de desarrollo. La administración de la configuración pretende eliminar la confusión y los errores que acarrear la existencia de diferentes versiones de artefactos. La administración de la configuración requiere políticas y estándares bien definidos.
- **Planeación Técnica:** Planeación es una de las funciones fundamentales de administración a cualquier nivel. Provee las bases para las otras actividades de administración como control y tracking del proyecto. Esta área concierne con la planeación de proyectos.

A continuación se definen cada una de las áreas de práctica para una línea de productos referentes al área de procesos.

- **Mejora y definición de procesos:** Esta área es sobre la capacidad que tiene una organización para definir, seguir y mejorar procesos. Un aspecto esencial de la ingeniería de software es la disciplina requerida por un grupo de personas que trabaja en conjunto para resolver un problema común. Humphrey define el término proceso de software, como “Una secuencia de pasos requeridos para desarrollar o mantener software”. Procesos definidos establecen los límites de cada persona sus roles y responsabilidades para que la colaboración sea exitosa y eficiente. Esta práctica se refiere a las habilidades requeridas para planear, definir y ejecutar varios procesos exitosamente.

A continuación se definen cada una de las áreas de práctica para una línea de productos referentes al área de organización.

- **Operaciones:** Esta área se refiere a como el negocio debe realizarse, es el motor que hace que la organización trabaje. Sin esto la organización se reduce a un conjunto de unidades que tienen la intención de hacer las cosas bien pero sin asegurarse de que es lo correcto. Cualquier organización en el negocio de desarrollo de productos opera bajo estrategias organizacionales y administrativas, las operaciones conjuntan todo esto en un coherente grupo de políticas y prácticas.
- **Estructura de la organización:** Se refiere a como la organización forma grupos para la realización de actividades relacionadas con el esfuerzo de desarrollo de software. Todas las organizaciones tienen una estructura, incluso si se tiene una implícita, la cual define roles y responsabilidades apropiadas para la realización de las principales actividades de la organización.
- **Entrenamiento:** Es una actividad principal de cualquier organización de desarrollo de software. El propósito es proveer las habilidades y el conocimiento necesario para desempeñar los roles técnicos y administrativos de la organización. Un programa de entrenamiento envuelve las necesidades de capacitación de la organización de las personas involucradas en esta. Es necesario desarrollar un programa de entrenamiento adecuado.

3.3 Administración de Desarrollo de Proyectos de Software de una Línea de Productos y Metodologías ágiles.

Como se comentó al inicio del documento uno de los problemas que enfrenta la industria del software es que los proyectos se retrasan, la mayoría de las veces exceden el presupuesto, tienen fallas de operación (la calidad es mala), son cancelados antes de terminar, por mencionar los más importantes.

Uno de los enfoques existentes para solucionar algunos de estos problemas es la Administración de Desarrollo de Proyectos de Software, la cual podemos definir como “Aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto para cumplir con los requerimientos del proyecto”.

La administración de desarrollo de proyectos se viene usando por algún tiempo y actualmente no todas las empresas manejan enfoque a procesos de administración de desarrollo de proyectos.

También como ya se mencionó anteriormente el enfoque de Líneas de Productos ofrece entrega de productos a tiempo ya que promueve una forma de reuso mediante la construcción de componentes base para usarlos posteriormente en la construcción de un producto final.

Pero aunque Línea de Productos ofrece todas esas ventajas como se vio en el segmento anterior es necesario agregar administración de desarrollo de proyectos de software como lo dice el área práctica de Planeación Técnica. Dentro de Metodologías de Procesos de Administración de Desarrollo de Proyectos de software existen 2 grandes grupos [6]:

- **Metodologías Ágiles:** Intentan evitar los tortuosos y burocráticos caminos de las metodologías tradicionales enfocándose en la gente y los resultados y enfatizan las comunicaciones cara a cara en vez de de la documentación.
- **Metodologías Tradicionales:** Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo, contienen una cantidad de políticas y normas.

Desgraciadamente metodologías específicas para administrar proyectos con líneas de productos no existen, por lo tanto fue necesario seleccionar cuales de las metodologías existentes podían ayudar a cumplir el propósito de administración de desarrollo de proyectos de software.

La siguiente tabla presenta las características de cada uno de los tipos de metodologías en base a los factores que determinan la creación del software [6].

FACTORES	METODOLOGÍAS ÁGILES	METODOLOGÍAS TRADICIONALES
REQUERIMIENTOS	<ul style="list-style-type: none"> • No definidos completamente • El alcance puede variar • Cambian 	<ul style="list-style-type: none"> • Bien definidos • El alcance delimitado • No cambian constantemente
RECURSOS	<ul style="list-style-type: none"> • Ingenieros no aprobados y disponibles completamente • Presupuesto del cliente • El cliente tiene una participación activa • Ingenieros aprendiendo tecnologías nuevas • Equipos pequeños (incluso una persona) • Trabajo on-site • Pocos roles • Se desarrollan pocos artefactos 	<ul style="list-style-type: none"> • Ingenieros aprobados y disponibles completamente • Presupuesto establecido bajo contrato • Equipos grandes • Trabajo distribuido • Muchos roles • Se desarrollan muchos artefactos
TIME TO MARKET	<ul style="list-style-type: none"> • Presión de mercado alta 	<ul style="list-style-type: none"> • Presión de mercado baja
TIME	<ul style="list-style-type: none"> • Claramente delimitado • Milestones delimitados 	<ul style="list-style-type: none"> • Las características pueden variar • Claramente delimitado
RIESGOS	<ul style="list-style-type: none"> • Bien entendidos 	<ul style="list-style-type: none"> • No conocidos • Nuevas tecnologías • Alto impacto

Fig. 3.3.1 Comparación entre Métodos Ágiles y Tradicionales

Si el enfoque sobre el cual se va a trabajar es Líneas de Productos, es necesario considerar varios aspectos en cuanto al desarrollo de los productos ya que no se desarrolla de forma tradicional.

Las características del desarrollo de software basado en Línea de productos son:

- Alto contacto con el cliente para definición y redefinición de requerimientos.
- Arquitectura única para la línea de productos.
- Construcción de componentes base, estos se pueden ser uno componente o varios que forman un subsistema.

- Ensamble de componentes para la creación de productos.
- Adaptación de componentes base dependiendo de las variaciones de los productos que se generan.
- La organización de los equipos de desarrollo puede variar dependiendo del tamaño de la organización y de los productos que genera esta. Por ejemplo los componentes base y los productos pueden ser desarrollados por el mismo equipo, se puede tener un equipo para creación de componentes base y otro para ensamblar (estos realizan las adecuaciones).
- Alta presión del mercado
- Es necesario definir el alcance de los productos y puede variar dependiendo de las condiciones del mercado.
- El trabajo se hace en el mismo lugar
- El trabajo se subdivide bastante, por lo tanto no existen equipos tan grandes.

Como podemos ver, las metodologías ágiles embonan dentro de la mayoría de las características del desarrollo con Líneas de Productos, ya que las metodologías tradicionales como TSP (Team Software Process) son muy rígidas y no se adaptan a los cambios de estructura o lo que exige el mercado, va enfocada a clientes específicos para sistemas específicos. Por estas razones se decidió seleccionar el uso de metodologías ágiles para la administración de proyectos en el enfoque de LPS.

Dentro de metodologías ágiles existen varias tales como XP (EXtreme Programming), Crystal, Peer Programing, Scrum, etc., pero dadas las características y los objetivos que perseguimos con el uso de Líneas de Productos no todas se adaptan a esta metodología de trabajo.

La metodología seleccionada para unirla a los proyectos de Líneas de Productos para su administración es SCRUM, debido a que es una de las más estables y mejor definidas, cuenta con roles, división de tareas, reuniones para ver el avance del proyecto, definición de objetivos, división del desarrollo en ciclos (sprint), planeación del proyecto y administración de riesgos

SCRUM es una metodología de administración para la producción de un producto o un prototipo [7].

La terminación de un producto de software es planeada dependiendo de las siguientes variables:

- **Requerimientos del Cliente:** Como debe funcionar el sistema en base a sus necesidades.
- **Tiempo delimitado:** Que tiempo se requiere para una ventaja competitiva.
- **Competitividad:** Que se necesita para ser competitivos
- **Calidad:** Cual es la calidad requerida, dadas las variables anteriores.
- **Visión:** Que cambios se necesitan para completar la visión del producto.
- **Recursos:** Que staff se requiere y cual tengo disponible.

Estas variables se determinan al inicio del proyecto, pero cambian durante éste, una exitosa metodología de administración de proyectos debe tomar en cuenta esas variables y su evolución.

La siguiente figura presenta un diagrama que representa la forma de trabajo de SCRUM [8].

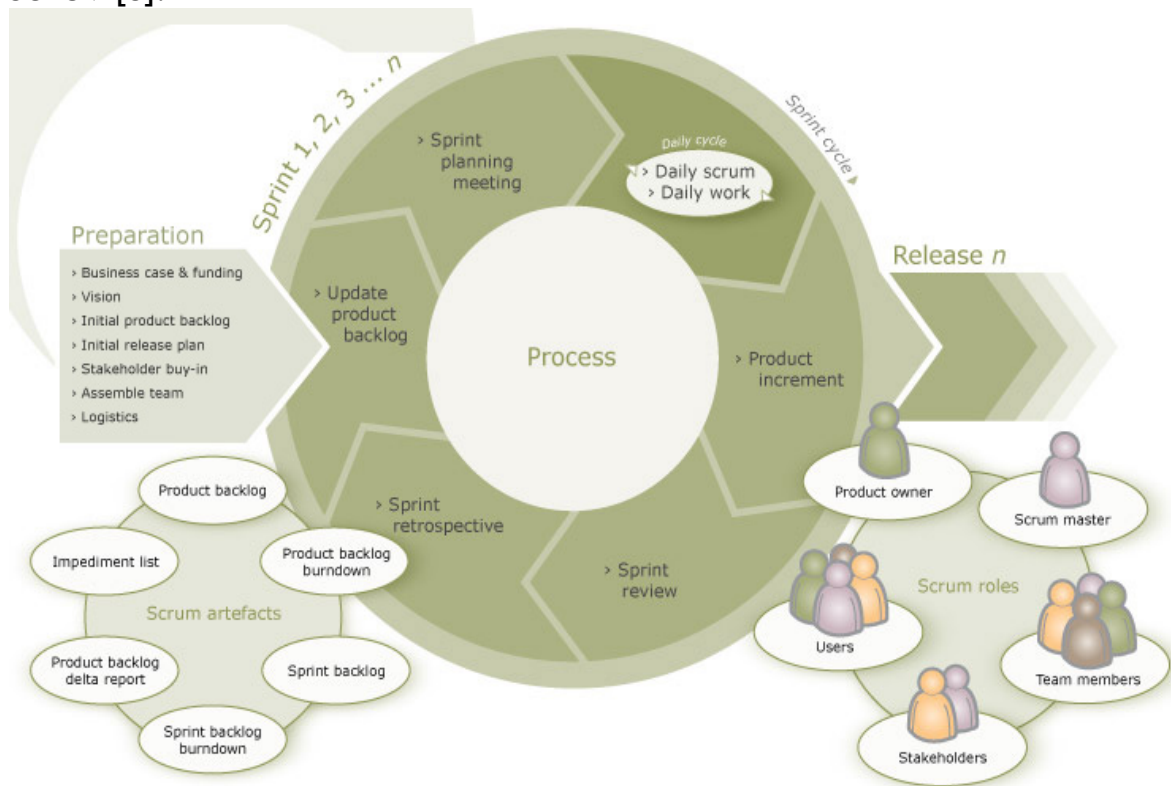


Fig. 3.3.2 Metodología SCRUM

A continuación se describe de forma breve el procedimiento de SCRUM.

- **Preparación:** Esta fase contiene actividades para iniciar con el trabajo, esta fase se le conoce como Sprint 0. Dentro de esta fase, es necesario

que se determine el tiempo que se invertirá en el proyecto, los costos que se desencadenarán de este, que se entienda que objetivos se desean alcanzar con el producto, determinar una lista de ítems que se deben realizar para generar el proyecto, documentarlos en un productbacklog, describir toda la funcionalidad del producto, estimar los ítems a construir, seleccionar la cantidad de sprints(ciclos) a realizar para generar el producto, definir la estrategia de desarrollo que se llevará a cabo y designar el equipo que construirá los ítems a desarrollar, así como asignar los roles.

- **Planeación de un sprint:** Se determinan los objetivos del sprint así como las tareas que se van a realizar (las cuales se documentan en un sprint backlog), se estiman y se priorizan por parte del equipo. Además estas tareas son asignadas a los miembros del equipo.
- **Daily work:** Se realiza el trabajo asignado a los miembros del equipo, durante la duración del sprint.
- **Daily scrum:** El equipo de trabajo tiene una junta diaria para ver el estatus del proyecto y presentar problemas o sincronizar actividades con otros miembros del equipo.
- **Sprint review:** Provee una inspección del progreso del proyecto al final de cada sprint. Durante esta fase se pueden hacer cambios a los ítems a construir del proyecto y se actualiza el product backlog. Al final del sprint, el equipo presenta el incremento del producto que se ha construido.
- **Sprint retrospective:** Esta es una reunión en la cual el equipo discute el sprint que se terminó para definir mejoras o propuestas que hagan mas productivo el siguiente sprint. El sprint review se puede entender como que hizo el equipo y el sprint retrospective se puede entender tal como se está trabajando.
- **Update product backlog:** Es necesario que al finalizar cada sprint se re-evalúe la priorización de los ítems a construir, verificar si se necesitan más ítems para alcanzar el objetivo, eliminar algunos, etc.
- **Release:** Es un conjunto de funcionalidad que es entregada en uno o más sprint y es usualmente puesto en producción en un release sprint. El primer release de un nuevo sistema tendrá al menos un conjunto de características que el usuario o clientes aceptarán.

SCRUM tiene roles necesarios para la administración de proyectos, los roles son:

- **Product Owner:** Es la persona que representa los intereses de una organización en referencia al proyecto. Es usualmente el representante del cliente, encargado de determinar los requerimientos funcionales y no funcionales para el proyecto. Es el responsable de asegurar que la

funcionalidad ha sido desarrollada e implementada cumpliendo las necesidades del negocio.

- **Team members:** Es un grupo de personas con diferentes habilidades que son necesarias para convertir los requerimientos en un producto palpable. Los equipos pueden ser de máximo 7 personas mínimo 2.
- **Scrum Master:** Es el administrador del proyecto, es la persona responsable de que el proceso se utilice adecuadamente. Algunas de sus actividades son proponer mejoras en las áreas de ingeniería y herramientas para incrementar la funcionalidad y productividad, eliminar problemas entre el equipo, detectar problemas entre el equipo y el cliente, por mencionar las más importantes.

SCRUM cuenta con ciertos artefactos que se deben producir, tales como:

- **Product backlog:** Es una lista priorizada de requerimientos funcionales y no funcionales con un alto nivel de estimación. El dueño del producto es el encargado de mantener este documento y asegurarse que la lista es actualizada y priorizada constantemente para ser empatada con las necesidades del negocio.
- **Sprint backlog:** Es una lista de tareas que define el equipo de trabajo de un sprint. Esta lista surge durante la planeación del sprint. Las tareas en el sprint backlog son la que el equipo define, se documenta quien es responsable de realizar cada tarea y se estima el tiempo en que debe realizarse la tarea.
- **Impediment List:** Cualquier aspecto que impida alcanzar la productividad o calidad para el producto en construcción, es responsabilidad del Scrum Master remover cualquier impedimento que detenga la producción de un producto de calidad. La lista de impedimentos es simplemente una lista de tareas que el Scrum Master usa para dar seguimiento a los impedimentos que son necesarios resolver.

4. Metodología Propuesta

Como se explicó anteriormente para comenzar a trabajar con Líneas de Productos no es una tarea trivial, es necesario considerar diferentes aspectos relacionados con el Negocio, Tecnología, Procesos y la Organización.

Dentro del trabajo con Líneas de Producto existen compañías grandes que ya trabajan con este enfoque y han obtenido los beneficios que busca en cuanto al desarrollo de software, pero estas compañías no han revelado su secreto, es decir no existe una fórmula predeterminada para realizar la adopción de Líneas de Productos y comenzar a obtener los beneficios que ofrece. La mayoría de la información existente referente a la adopción de Líneas de Productos, nos habla sobre el Qué, pero nunca nos dice Cómo.

El objetivo de esta metodología propuesta es que la organización que desea cambiar su enfoque de trabajo a Líneas de Productos, tenga una guía a seguir sobre Qué necesita realizar y lo más importante, Cómo hacerlo.

Esta metodología consta de un conjunto de procesos definidos para cada uno de los aspectos que se deben considerar en el trabajo con Líneas de Productos, así como un ciclo de vida para estas tareas y un organigrama propuesto para la organización, además una propuesta de cómo la Metodología SCRUM ayudará a la administración de desarrollo de proyectos de Software de este tipo.

Para comenzar es necesario que se defina un ciclo de vida, es decir las fases por las cuales debemos pasar para trabajar con Líneas de Productos. El siguiente diagrama representa el ciclo de vida propuesto en esta metodología considerando los aspectos necesarios para el trabajo con Líneas de Productos.

En la primer parte del ciclo de vida encontramos las actividades pertenecientes al negocio que son las tareas iniciales con las cuales se debe trabajar ya que es necesario determinar nuestro mercado y delimitar los productos a construir, así como las características de estos. Las actividades a realizar en esta área son:

- Realización del análisis de mercado
- Realización del business case
- Determinación del alcance de la línea de productos
- Lanzamiento e institucionalización

Para más detalle de actividades a realizar ver el proceso específico de cada una de las actividades, ver Anexo A.

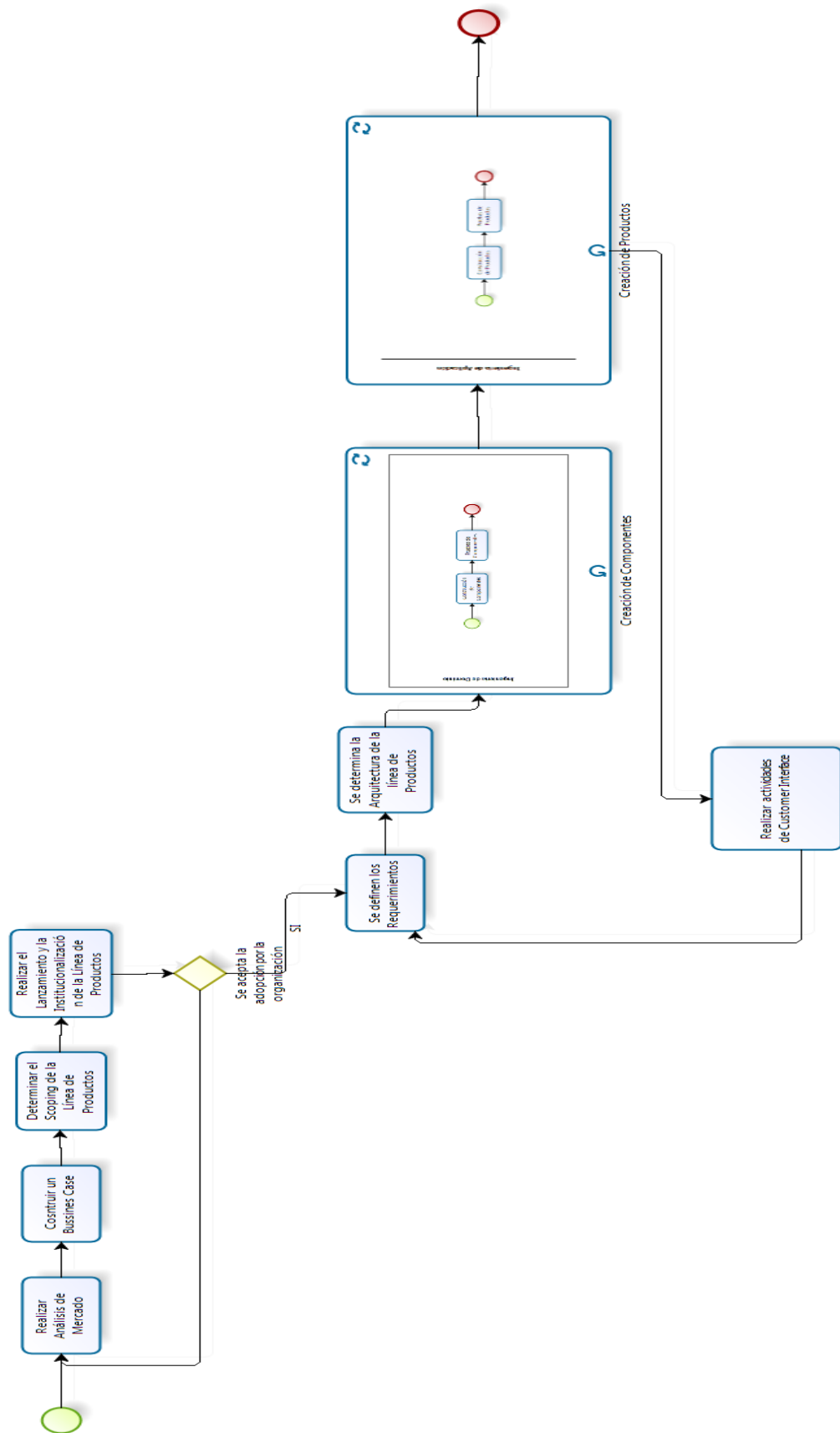


Fig. 4.1 Ciclo de Vida de una LPS

Las siguientes tareas a realizar en el ciclo de vida una vez que las tareas de negocio fueron exitosamente realizadas, son las referentes a la parte de Ingeniería de Software, es decir construcción de los productos. Las actividades referentes a esta son:

- Obtención de requerimientos
- Definición de la Arquitectura
- Construcción de los componentes base (incluyen pruebas)
- Construcción de los productos (incluyen pruebas)
- Actividades de Customer Interface

Para más detalle de actividades a realizar ver el proceso específico de cada una de las actividades, ver Anexo A.

Una de las actividades principales es la Organización, por lo tanto es necesario determinar un organigrama de la organización, debido a que las actividades a realizar en el desarrollo tradicional son diferentes, es necesario adaptar este para que la organización trabaje como se requiere y de esta forma poder asignar responsabilidades de forma más sencilla. La siguiente figura muestra un organigrama propuesto para empresa que deseen adoptar este enfoque.

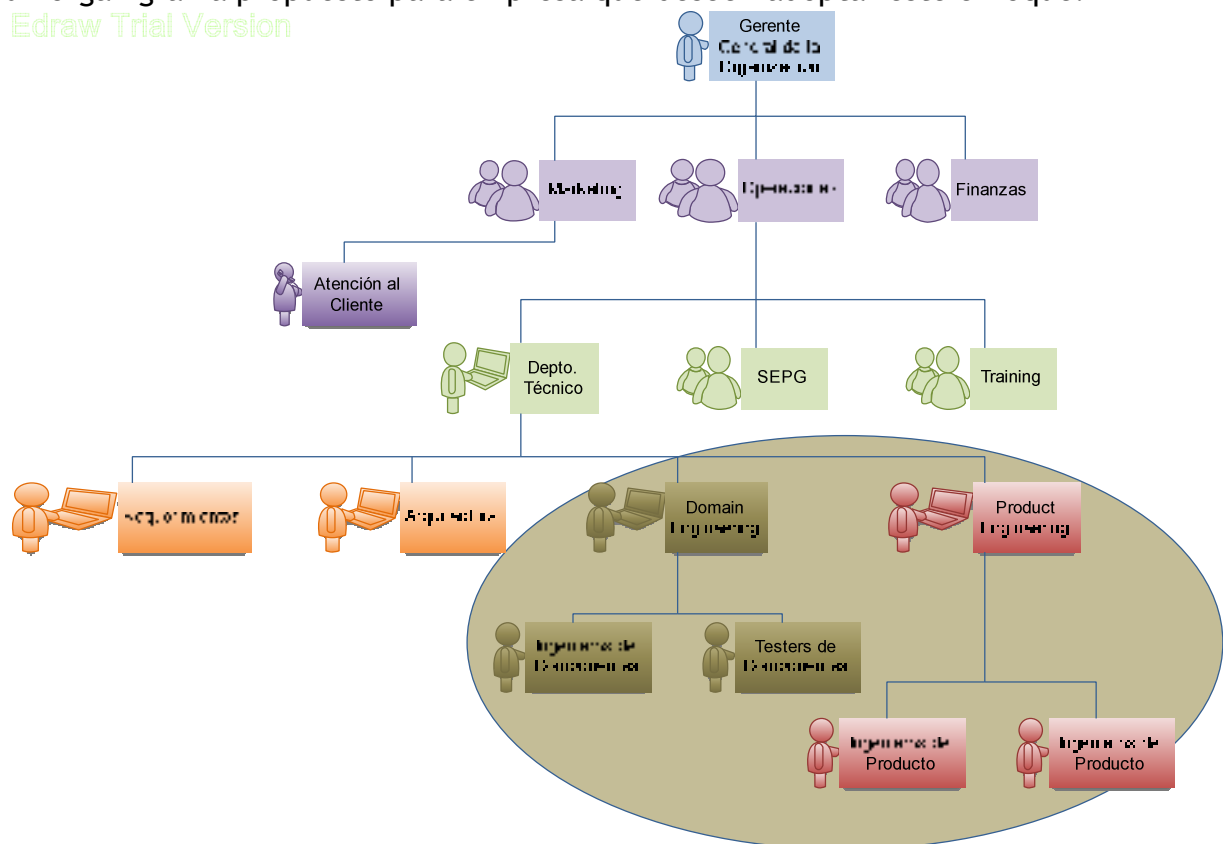


Fig. 4.2 Organigrama para una organización usando LPS

A continuación se describen las actividades de las que se encargará cada uno de los departamentos del organigrama.

- **Gerente General de la Organización:** Es el encargado de la administración de la organización, finanzas, toma de decisiones del negocio.
- **Departamento de Marketing:** Es el encargado de realizar todas las actividades que tienen que ver con los clientes y el negocio, tales como creación del estudio de mercado, definición del alcance, delimitación del mercado del producto, etc. De este departamento se desprende otra unidad llamada atención al cliente, la cual nos ayudará a determinar las necesidades del cliente ya sea para modificación en los productos o para detectar nuevas áreas de oportunidad.
- **Departamento de Operaciones:** Este departamento es el encargado de coordinar las actividades relacionadas al business core de la organización. Tendrá a su cargo el departamento técnico, de procesos y el de entrenamiento.
- **Departamento de Financiamiento:** Encargado de las finanzas de la organización.
- **Departamento Técnico:** Es el encargado de la construcción de los productos que saldrán al mercado, de este se desprenden las áreas de requerimientos, arquitectura, ingeniería de dominios y ingeniería de productos , cabe mencionar que dichas áreas pueden unificarse dependiendo del tamaño de la organización.
- **Departamento de SEPG:** Es el encargado de la creación y mejora continua de los procesos de la organización.
- **Departamento de Training:** Encargado del reclutamiento y entrenamiento de personal de nuevo ingreso. Además es el encargado de detectar áreas de oportunidad para el entrenamiento y actualización de los ingenieros que forman parte de la organización.
- **Departamento de Requerimientos:** Son los encargados de la definición de los requerimientos que deben tener tanto los productos finales como los componentes base.

- **Departamento de Arquitectura:** Encargado de la creación y mantenimiento de la arquitectura de la o las líneas de productos que maneje la organización.
- **Departamento Ingeniería de Dominio:** Encargado de la creación de los componentes base para la construcción de los productos, así como de las pruebas de los componentes.
- **Departamento Ingeniería de Producto:** Encargado del ensamble de los componentes base para crear los productos necesarios, además se encargan de realizar las variaciones a los componentes base y se encargan de realizar las pruebas a los productos que salen de la Línea de producción.

Como sabemos cualquier tipo de proyecto necesita administración, en el capítulo anterior se habló sobre diversas metodologías de administración de desarrollo de proyectos de software. Por las características propias de los proyectos de Líneas de Productos se decidió seleccionar las metodologías ágiles y de entre estas se seleccionó SCRUM para empatarla al trabajo con Líneas de Productos.

La siguiente figura presenta como podemos empatar la metodología de SCRUM con Líneas de Productos.

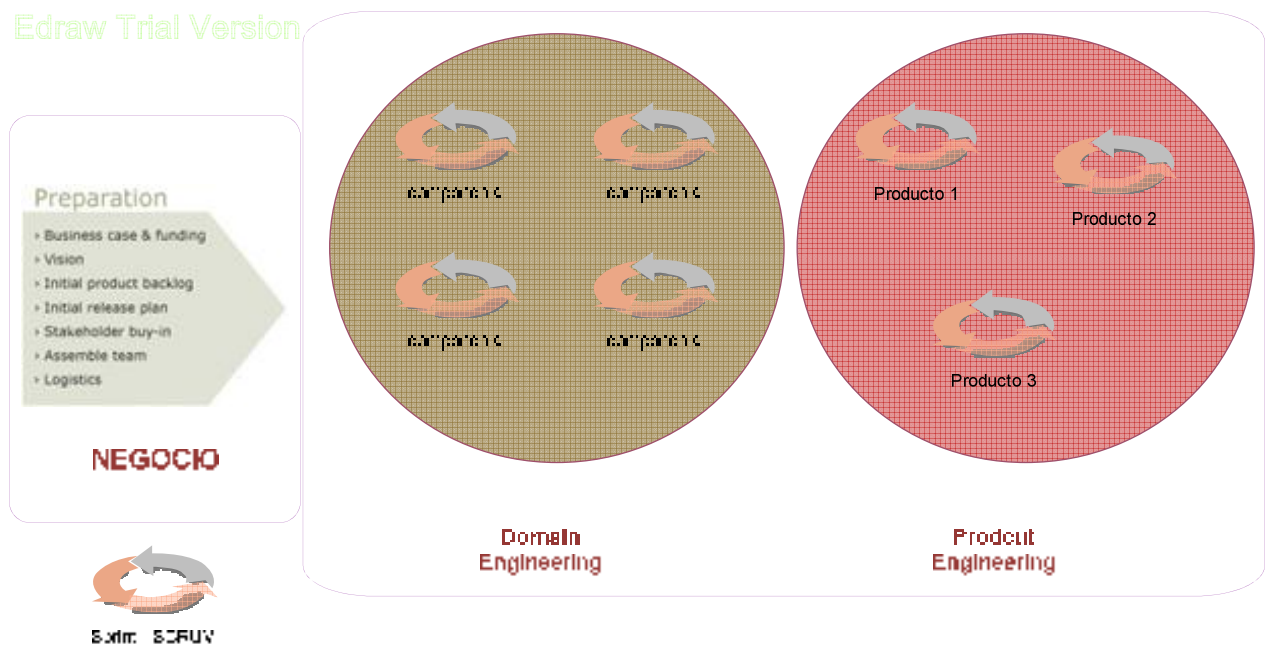


Fig. 4.3 SPL-SCRUM

Como ya mencionamos en un apartado anterior, SCRUM tiene una fase de Preparación, en la cual es necesario determinar las características del producto, los stakeholders involucrados, el tiempo de duración del proyecto, los objetivos a cumplir, las actividades a realizar, que todos los involucrados entiendan el proyecto, delimitar el alcance del proyecto, determinar los costos que se generarán para la realización del proyecto, en fin todo lo referente a la planeación del proyecto. Por otro lado, Líneas de Productos en las actividades del área de negocio, la cual es la inicial en el ciclo de vida de la adopción es necesario determinar nuestro mercado, las características de los productos a construir, delimitar el alcance de los productos.

Como podemos observar ambas fases tienen objetivos en común por lo tanto SCRUM se adapta perfectamente en la etapa inicial de la adopción de líneas de producto.

Una vez delimitado el proyecto y realizada la planeación de este, se procede a entrar a la etapa de construcción de componentes y productos. En el caso de la construcción de componentes, se pueden estar construyendo diferentes componentes base para el mismo producto o para varios productos al mismo tiempo, por lo tanto se propone que para la creación de cada uno de los componentes base, se realicen los sprint necesarios para la terminación de estos.

Lo mismo se propone para la parte de productos ya que se pueden estar ensamblando productos diferentes o bien realizando las variaciones a los componentes base, se propone que por cada mini-proyecto de variación se realicen los sprint necesarios para su terminación posterior uso, de igual forma se pueden hacer sprints para obtener n cantidad de productos ensamblados.

Es necesario que cuando se realice la preparación de la parte de administración, donde se definen cuantos componentes base necesitamos y como se ensamblará el producto, se agregue al product backlog la cantidad de sprints y su duración, así como su orden cronológico de ejecución para cada componente base y para la construcción de productos, ya que en un sprint se pueden ensamblar una cantidad finita de productos.

Se debe tener un control de la creación de productos y componentes base debido a que siempre se va a tener un fecha de entrega, y es muy posible por la forma de trabajo de la línea de productos que la creación de los componentes base se retrase resultando en que el producto no se encontrará listo para la hora de entrega.

Como podemos ver Líneas de Productos por si solas no pueden cumplir tan fácilmente su objetivo de reducción de tiempos y costos, siempre es necesario de que sean acompañadas de la parte de administración de proyectos de desarrollo de software.

5. Conclusiones

5.1 Tendencias Futuras

Existe mucho que hacer en esta área en cuanto a las Líneas de Productos de Software. A continuación listo algunos issues que identifiqué y que pueden ser tema de estudio futuro.

- No se encontró disponible en la literatura una metodología de Administración de Proyectos de Desarrollo de Software específica para Líneas de Productos.
- En la parte de Administración de Procesos es necesario identificar las métricas necesarias a medir en proyectos que utilicen este enfoque.
- No se encontró disponible información sobre los procesos de la parte de Negocio, especialmente para Líneas de Productos.
- El área de Administración de la Configuración hay mucho trabajo ya que es necesario tener el control de los componentes base y los productos de una línea de productos y no se encontró disponible en la literatura un método específico para línea de productos lo más parecido que podemos usar es Administración de la Configuración para componentes.
- Dentro de cualquier organización la parte administrativa es muy importante y falta trabajo para la definición de los procesos administrativos de una organización dedicada al desarrollo de software y sobre todo en Líneas de Productos.

En cuanto al trabajo presentado, se identifican las siguientes áreas de oportunidad:

- Seleccionar un caso de estudio para realizar pruebas en el CDS.
- Poner en práctica los procesos creados en el CDS en un proyecto piloto para poder mejorarlos.
- Agregar procesos como el de administración de la configuración y su integración con las áreas de construcción de componentes base y realización de productos.
- Detallar los procesos existentes.
- Probar que al unir Líneas de Productos y SCRUM se realiza un desarrollo rápido y de calidad.

5.2 Beneficios obtenidos

- Se obtuvo una base fuerte para comenzar a trabajar con Líneas de Productos de Software.
- El CDS podrá experimentar la adopción de un enfoque de trabajo nuevo para poder comercializar sus productos.
- Se identificaron áreas en Líneas de Productos que no han sido exploradas.

Referencias

- [1] Lawrence G. Jones, Albert L. Soule, *Software Process Improvement And Product Line Practice: CMMI And The Framework For Software Product Line*. CMU/SEI-2002-TN-012.
- [2] Kenyer Dominguez, *Una ontología hacia Fábricas de Software*
- [3] Gunter Bockle, Jesús Bermejo, Meter Knauber, Charles W. Krueger, Julio César Sampiano, Linda Nurthop, Michael Stark and David M. Weiss. *Adopting and Institutionalizing a Product Line Culture*. Software Product Lines, Second Internacional Conference SPLC2.
- [4] Frank van der Linden, Jan Bosch, Erik Kamsties, Kari Känsälä, Henk Obbink. *Software Product Family Evaluation*.
- [5] Linda Northrop, Paul Clements. *A Framework for Software Product Line Practice -Version 2.0*. [online]. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, July 1999. Available WWW: <URL: <http://www.sei.cmu.edu/plp/framework.html>>.
- [6] Rob Keefer. *Agil Software Development The button Line*. Strategic Data Systems, Inc.
- [7] Ken Schwaber. *SCRUM Development Process*. Advanced Development Methods.
- [8] Ken Schwaber. *SCRUM for Team System*. <URL: <http://scrumforteamssystem.com/processguidance/v2/Scrum/Scrum.aspx>>.
- [9] [Kit de Procesos para adoptar una Línea de Productos](#)