

**UNIVERSIDAD GABRIELA MISTRAL
FACULTAD DE INGENIERIA**

**GESTIÓN ÁGIL DE DESARROLLO DE
SOFTWARE PARA SAP ERP**

Memoria para optar al título de Ingeniero de Ejecución en Informática

Autor : Juan Luis Araneda Urrutia
Profesor Guía : Jaime Orellana R.
Profesor Integrante : Jorge Tapia C.

Santiago – Chile
Abril, 2009

INDICE

Introducción.....	1
1 Capítulo I.....	4
1.1 Prefacio de las aplicaciones ERP.....	4
1.2 Descripción del problema.....	5
1.3 Motivación.....	7
1.4 Hipótesis.....	8
1.5 Objetivo General.....	8
1.6 Objetivos Específicos.....	8
1.7 Alcances del trabajo de título.....	9
1.8 Antecedentes del proyecto.....	9
2 Capítulo II - Marco teórico.....	10
2.1 Gestión de Proyectos: ¿Formal o ágil?.....	10
2.2 SAP ERP.....	21
2.3 Metodología para implementación de proyectos ASAP.....	22
2.4 SCRUM.....	25
3 Capítulo III - Desarrollo del Trabajo.....	33
3.1 Tecnología involucrada.....	33
3.2 Organización del proyecto.....	35
3.2.1 Fase 1 - Preparación del Product Backlog.....	36
3.2.2 Fase 2 - Análisis funcional y diseño.....	43
3.2.3 Fase N - Iteraciones Semanales de Desarrollo.....	44
3.3 Recursos utilizados.....	56
4 Lecciones aprendidas.....	57
5 Hallazgos.....	59
6 Conclusión.....	62
7 Glosario.....	63
8 Bibliografía.....	71

Introducción

En el proceso de globalización que se vive hoy en la economía mundial, las empresas requieren de servicios tecnológicos eficientes para sobrevivir. Es por esta razón que la tendencia de los últimos años ha sido la de adquirir un *Enterprise Resource Planning* (ERP) para mantenerse competitivos.

¿Qué es un ERP? Un ERP es una “Aplicación de Planificación de Recursos Empresariales”. Se trata de un software que le permite a las compañías automatizar e integrar la mayor parte de sus procesos de negocio, compartir datos, producir y acceder a la información en tiempo real.

Las aplicaciones *ERP* proveen una integración empresarial, ya que administran la información de las diferentes transacciones y operaciones.

Fue diseñado para mejorar el flujo de datos entre los distintos departamentos de una organización.

Estos sistemas soportan las múltiples actividades o procesos de las empresas, debido a su diseño modular, incluyendo planeación de producción, compras, inventarios, proveedores, etc. La tendencia por adoptar este tipo de aplicaciones ha dominado desde los inicios de la década de los 90, y desde su creación han sido conceptualizados como plataformas que demandan una gran inversión económica y requieren, la mayor parte de las veces, de una reestructuración organizacional o reingeniería de procesos de la empresa.

Uno de los mayores proveedores de este tipo de soluciones es SAP AG. Ofrece aplicaciones y servicios que permiten a compañías de todos los tamaños, en más de 25 sectores de la industria, gestionar sus procesos de negocios de manera más eficiente. Con más de 75.000 clientes en más de 120 países.

SAP AG es el líder mundial en sistemas de administración para empresas, y ofrece completos software de gestión y servicios que satisfacen necesidades específicas; uno de los principales software es SAP ERP. Esta solución brinda una funcionalidad de principio a fin para el análisis de los negocios, finanzas, gestión del capital humano, operaciones y servicios corporativos.

Producto de la gran demanda por parte de las organizaciones por implementar SAP ERP, existen empresas que se dedican a proveer, asesorar y brindar soluciones para mejorar los procesos de negocios. Uno de los principales partner en Chile de SAP es Crystalis Consulting (**SAP “GOLD” Channel Partner**)¹, quien se dedica a desarrollar nuevas soluciones para empresas, además de prestar servicios de consultoría SAP y asistencia a organizaciones que ya tienen instalados sistemas de gestión.

Asimismo busca constantemente la mejora y posicionamiento como líderes en el desarrollo e implementación de soluciones de negocios para empresas, ayudándolas a aumentar su rendimiento y optimizar sus procesos.

Hasta ahora en la mayoría de las implementaciones SAP ERP realizadas por Crystalis Consulting, los desarrollos adicionales para satisfacer necesidades específicas de los clientes se han realizado bajo una metodología de desarrollo en cascada, con las etapas correspondientes a: levantamiento de requerimientos, análisis, diseño, construcción y pruebas. Tanto la gestión como el desarrollo son realizados siguiendo rigurosamente un plan de trabajo establecido por la metodología recomendada por SAP llamada *ASAP*².

La necesidad de construir aplicaciones, al mismo tiempo que cambian y aparecen nuevos requisitos, está obligando a las organizaciones, como Crystalis Consulting (que se dedican al desarrollo de aplicaciones), a replantear la forma en que

¹ La categoría más alta que otorga SAP entre sus Partners de Canal

² ASAP es un acrónimo para la metodología AcceleratedSAP, desarrollada por SAP AG para la implementación de proyectos de sus soluciones de software.

gestionan sus proyectos de software. Es por esta razón que este trabajo pretende demostrar que gestionando de manera ágil, utilizando SCRUM³ como metodología de desarrollo, se pueden lograr resultados de calidad en entornos inestables donde se requiere flexibilidad y calidad en el producto final, cumpliendo con los plazos y presupuestos comprometidos.

³ Proceso de desarrollo de software basado en un ciclo de vida iterativo e incremental utilizado comúnmente en entornos ágiles de desarrollo de software.

1 Capítulo I

1.1 Prefacio de las aplicaciones ERP

Es indudable que el ambiente competitivo que se vive actualmente en el entorno empresarial insta a promover los procesos y actividades de negocio que generan las ventajas corporativas de las compañías ante sus más fuertes competidores.

Producto de lo anterior, las empresas han dado mayor importancia a las Tecnologías de Información y su alineación con las estrategias del negocio para mejorar los procesos claves de negocio. Prueba de ello es el incremento sustancial de adquisiciones de paquetes de software empresariales tales como el ERP⁴, con el cual los directivos de las compañías esperan tener integradas todas las áreas o departamentos de ellas, para apoyar la generación de sus productos y servicios.

Dentro de las compañías de software productoras de ERP presentes, en el mercado nacional como extranjero, podemos mencionar DeFontana, Dynamics Axapta (Microsoft), JD Edwards, Peoplesoft, Oracle Financials y SAP; adicionalmente existen algunas casas (o empresas) productoras de software ERP que tienen presencia en Chile, pero que se encuentran rezagadas en cuanto a cuota de mercado; esto incluye el tímido ingreso de las aplicaciones de código abierto (Open Source) como Compiere ERP u OpenPro ERP.

⁴ Enterprise Resource Planning

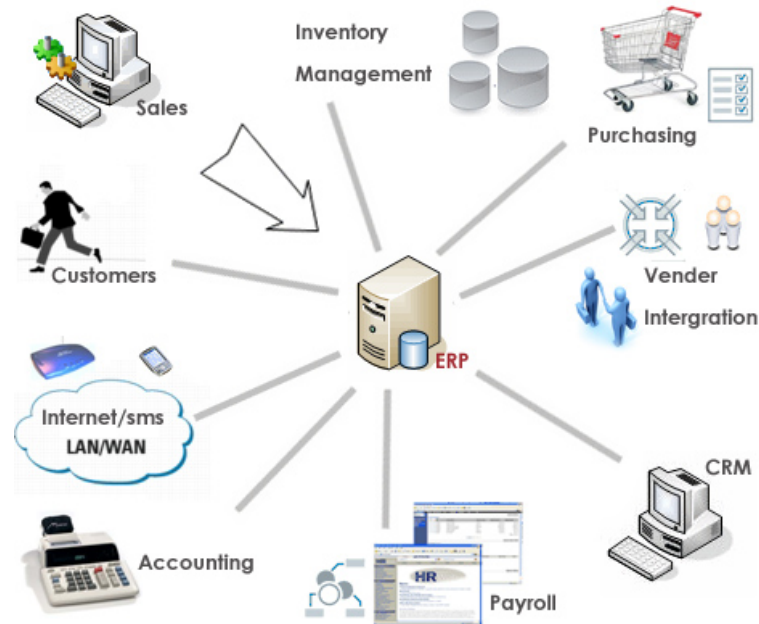


Figura (1.0): Vista general de los módulos que contiene un ERP.

1.2 Descripción del problema

Las posibilidades de uso que permite un ERP y la forma en que va a fundirse dentro de la operación diaria de las empresas, hacen que la instalación y configuración de una aplicación así pueda tomar varios meses, involucrando a todas las áreas. Sin duda un trabajo crítico para el proveedor (quien instala y configura) y el cliente (empresa que adquiere el producto).

La implementación de un sistema de ERP por lo general es complejo, ya que implica rediseñar los procesos y métodos de trabajo. Un proyecto de esta naturaleza encierra complejidad, tamaño, altos costos, un equipo considerable de desarrollo e inversión de tiempo de parte de las organizaciones que deseen implementarlo.

En general, cada ERP, cuenta con metodologías de implementación particulares, con las mejores prácticas, para llevar a buen término el proyecto. Estas incluyen actividades dependiendo de la empresa donde se pretenda implementar utilizando las

experiencias recopiladas de instalaciones realizadas. La gran mayoría de estas metodologías tienen como base los procesos de gestión de proyectos del PMBOK⁵ establecidos por el PMI⁶.

El levantamiento de los procesos actuales de la organización, es una de las actividades principales para configurar un ERP, el cual queda por escrito en un documento para revisar y verificar los procesos y requerimientos de la empresa. Una vez aprobado el documento por el cliente, se procede a la configuración del ERP, en función de los requerimientos recopilados.

Los requerimientos de parte del cliente, que se generen por la revisión de los procesos o como parte de la especialización de la organización donde se va a implementar, son tratados como una mejora en el ERP. Estas mejoras implican contar con un equipo de desarrollo que pueda construir las funcionalidades o reportes adicionales.

El problema que presenta la mayoría de las metodologías tradicionales utilizadas por los proveedores de soluciones ERP en el desarrollo de mejoras o aplicaciones de software son las siguientes:

1. El cliente al estar inmerso en los procesos de levantamiento de requisitos, instalación y configuración del ERP, no considera el tiempo necesario para entender como se desarrollará, verá y comportará un requerimiento en el sistema.
2. No se cumple con las expectativas del cliente; esta percepción se debe a que no se ve nada del desarrollo hasta que el requerimiento está completamente construido y entregado.

⁵ Project Management Body of Knowledge es un estándar en la gestión de proyectos desarrollado por el Project Management Institute.

⁶ Project Management Institute (PMI) es una asociación profesional sin fines de lucro en gestión de proyectos.

1.3 Motivación

Las experiencias en proyectos basados en implementación SAP ERP muestran un denominador común, y es que, el desarrollo para mejoras o construcción de aplicaciones de software son realizadas bajo un ciclo de vida en cascada (ver figura 2.0). Este modelo tiene una gran falencia: no permite la modificación del proyecto por nuevas necesidades del cliente, las que se suscitan a medida que avanza el proyecto.

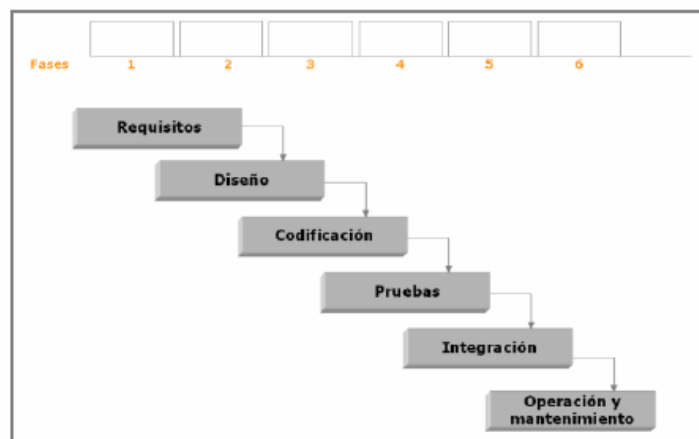


Figura (2.0): Ciclo de vida en cascada

La implementación de cualquier proyecto siempre debe ajustarse a las necesidades de agilidad y flexibilidad para trabajar en entornos inestables y rápidos.

La motivación de este trabajo consiste en demostrar que es posible revertir esa carencia en el desarrollo de software, construyendo aplicaciones al mismo tiempo que cambian y aparecen nuevos requerimientos, y como las circunstancias de los mercados y de las empresas no se pueden cambiar, son las formas en las que se gestionan los proyectos de Software las que tienen que cambiar para dar respuesta a las nuevas necesidades que surjan del cliente.

Esto resulta un requisito indispensable para que el cliente sienta que se ha cumplido con sus expectativas y exigencias, otorgándole un producto de calidad final que calce con lo buscado por su organización.

1.4 Hipótesis

Comprobar que utilizando una metodología ágil de desarrollo que se adapte a las necesidades del proyecto y del cliente es posible manejar los alcances y entregar un producto de calidad en los plazos establecidos.

1.5 Objetivo General

Demostrar al término de la tercera iteración⁷ que es posible gestionar exitosamente los objetivos de un proyecto de desarrollo de software en una implementación SAP ERP, utilizando la metodología ágil SCRUM.

1.6 Objetivos Específicos

Los objetivos específicos de este trabajo son los siguientes:

- Establecer la metodología SCRUM en una implementación SAP ERP.
- Ejecutar la metodología SCRUM en un proyecto real.
- Obtener un producto de software entregable en base a la metodología SCRUM.

⁷ En la metodología SCRUM un proyecto se ejecuta en bloques temporales (iteraciones), Cada iteración tiene que proporcionar un resultado completo, un incremento de producto que sea susceptible de ser entregado con el mínimo esfuerzo cuando el cliente lo solicite.

1.7 Alcances del trabajo de título

Desarrollar las funcionalidades que se puedan construir dentro de los plazos establecidos para las tres primeras iteraciones, siendo lo único necesario para llegar a los hallazgos y conclusiones de mi trabajo de título.

1.8 Antecedentes del proyecto

Una de las empresas interesadas en instalar un software ERP es Ópticas GMO, y es en este proyecto donde se pretende poner en práctica la utilización de SCRUM, como metodología de desarrollo.

Ópticas GMO decide, en mayo del 2008, comenzar la implementación de SAP ERP como recurso corporativo para disponer de una solución integral ERP, que resuelva las necesidades de operación y del negocio GMO/ Sun Planet, a nivel global, estableciendo un núcleo común que pueda ser replicado en los distintos países en los que tienen sucursales, e implementar la solución en Chile.

Como parte de la implementación de SAP ERP, Ópticas GMO necesita de una interfaz simplificada y amigable para los puntos de venta, que cuente con integración directa con el ERP. Esta aplicación contará con tres módulos principales: gestión de personal, gestionar el stock y gestión de encargos⁸.

Para dar soporte a los requerimientos se propuso la utilización de una arquitectura estándar SAP para desarrollo de aplicaciones en entornos WEB llamado WebDynpro, soportado por un SAP WAS⁹ con motor J2EE.

⁸ Los encargos son el resultado de una venta que puede estar compuesta por un armazón y dos cristales.

⁹ Servidor de aplicaciones Web de SAP

2 Capítulo II - Marco teórico

2.1 Gestión de Proyectos: ¿Formal o ágil?

¿Ágil, clásica, predictiva...?

Hasta los años 80 se utilizaba una gestión de proyectos basada en la planificación del trabajo, ejecución, seguimiento y control; esta forma es conocida comúnmente como predictiva o clásica. Es en este periodo (década de los 80), cuando surge una nueva forma de gestionar proyectos, y se hizo necesario añadir un “apellido” al término “gestión de proyectos”, para matizar si se refería a la nueva o a la de siempre. La nueva, al autodenominarse ágil, obligó a dar un apellido al modelo de gestión de proyectos que hasta entonces, por único, no lo había necesitado. Las denominaciones empleadas para cada tipo son:

Gestión de proyectos	Clásica Tradicional Predictiva Formal Pesada
Gestión de proyectos	Ágil Adaptable Adaptativa

En algunos ámbitos hay cierta rivalidad académica o profesional entre defensores de uno y otro modelo. Es preferible por tanto no emplear el término “pesado” que puede aportar connotaciones peyorativas. También es recomendable no emplear “adaptativa” y usar en su lugar “adaptable”, para evitar un anglicismo innecesario.

Premisas de la gestión de proyectos predictiva.

Premisas sobre las que se desarrolló la gestión de proyectos tradicional:

1. Todos los proyectos mantienen características y comportamientos regulares¹⁰.
2. El objetivo de la ejecución de un proyecto es lograr el producto previsto en el tiempo planificado sin desbordar los costos estimados¹¹.

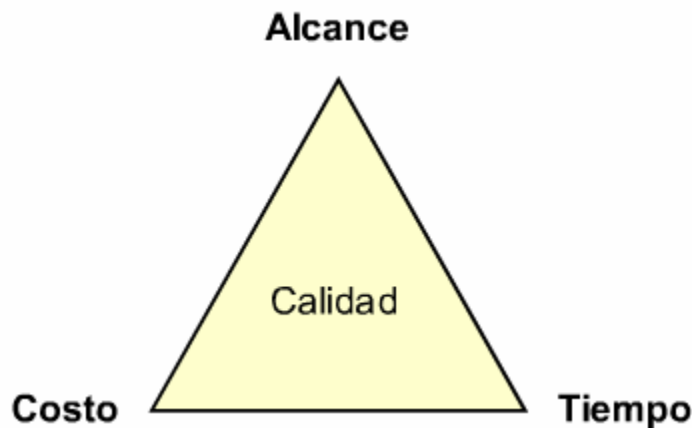


Figura (3.0): Triple restricción de la gestión de proyectos.

Características de la gestión de proyectos predictiva

Estas premisas han dado dos características a la gestión de proyectos predictiva:

1. **Universalidad.** Los proyectos, pese a su diversidad, comparten patrones comunes de ejecución. Las prácticas de gestión se basan en estos patrones comunes y resultan válidas para cualquier tipo de proyecto.

¹⁰ Meter Norden 1960, seminario de Ingeniería de Presupuesto y Control presentado ante American Management Association

¹¹ Guía del PMBOK del PMI, Tercera Edición

2. **Carácter predictivo.** La gestión clásica define con detalle cuál es el “producto previsto” y elabora un plan de desarrollo, a partir del cual calcula costos y fechas. Durante la ejecución realiza actividades de seguimiento y vigilancia para evitar desviaciones sobre lo planificado.

Hay otras premisas

Las dos premisas que cimientan el desarrollo de la gestión de proyectos:

- Todos los proyectos comparten idénticos patrones de ejecución.
- El objetivo es conseguir el producto definido en costos y fechas.

1. No hay una forma única y válida para gestionar cualquier tipo de proyecto

Es cierto que muchas características que diferencian unos proyectos de otros son superficiales y resultan indiferentes para el modelo de gestión; pero hay otras que permiten adoptar estrategias de gestión muy diferentes en cada caso.

Características diferenciales:

- Componente innovador que se espera del resultado.
- Grado de estabilidad de los requisitos durante el desarrollo.
- Costo de los prototipos.
- Maleabilidad del producto para modificar su funcionalidad una vez desarrollado.

- La gestión de proyectos predictiva, al auto-considerarse válida para cualquier proyecto, no contempla que, según las características del proyecto, puedan resultar más apropiados otros criterios de gestión.
- Conseguir el mayor valor innovador del producto no es uno de sus objetivos, y por tanto no aplica prácticas diferentes según el grado innovador que se desee.
- Considera que los requisitos deben permanecer estables durante la ejecución. Si no ocurre esto, presupone que se debe a deficiencias en el proceso de la fase de requisitos, porque no contempla posibles evoluciones rápidas o inestabilidades del entorno tecnológico o de negocio.
- El objetivo es la eficiencia, el cumplimiento del plan y no el valor del producto. Desde este punto de vista, el trabajo adicional siempre es caro. No se considera la relación entre el costo del trabajo adicional y el valor de los prototipos.

2. Hay proyectos en los que no se quiere “hacer el producto descrito en las fechas y con los costos estimados”.

Cuando en 1978 la dirección de Honda encargó el diseño de un nuevo automóvil, sólo dio dos instrucciones al equipo de ingenieros: “Primero: necesitamos un concepto de automóvil completamente diferente a lo que cualquier otra compañía haya hecho nunca; y segundo: debe ser un coche económico, pero no barato”. El resultado fue el Honda Civic.

Se pedía un proyecto, pero no se quería garantizar la ejecución de un plan, sino obtener el máximo valor en las líneas marcadas. Hay proyectos en los que importa más el valor y la innovación que el cumplimiento del plan.

La gestión predictiva pide al equipo el cumplimiento del plan. La gestión ágil pide al equipo el mayor valor posible para una visión de producto

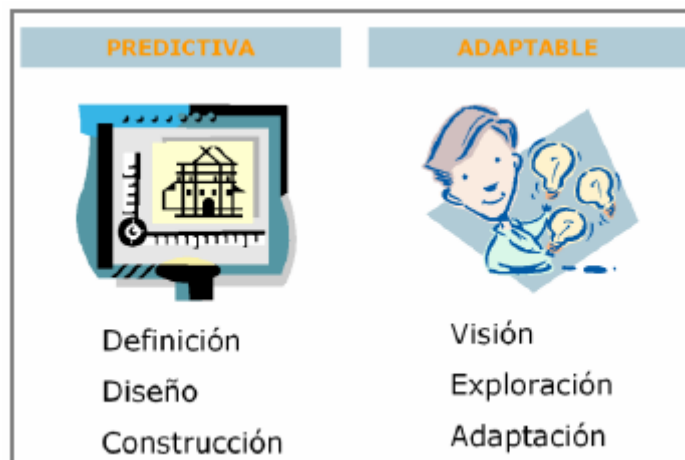


Figura (4.0): Principales diferencias entre la gestión predictiva y ágil.

Cuándo y por qué emplear uno u otro estilo de gestión

Para obtener los mayores beneficios que cada estilo de gestión puede ofrecer, éste tiene que ser compatible no sólo con las características del proyecto, sino también con las de la organización que las va a aplicar.

Características del proyecto

Las características relevantes para decidir el estilo de gestión más adecuado son:

- Principal prioridad de negocio.
- Estabilidad de los requisitos.
- Rigidez del producto.
- Costo de prototipo.
- Criticidad del sistema.
- Tamaño del sistema.

El orden expuesto se corresponde con nuestro criterio de relevancia, siendo por tanto la prioridad de negocio la principal razón de compatibilidad o incompatibilidad, y el tamaño del sistema la menos relevante. Por la relativa novedad de la gestión ágil, estos criterios no están aún consensuados. Así por ejemplo: Mientras algunos textos opinan que el tamaño o la criticidad del sistema son aspectos muy relevantes, hay opiniones autorizadas en sentido contrario:

En la "International Conference on Complex Systems 2006", Jeff Sutherland presentó el informe "Adaptive Engineering of Large Software Projects with Distributed / Outsourced Teams ", que demostraba los buenos resultados obtenidos con prácticas de gestión ágiles en un desarrollo de grandes dimensiones: un millón de líneas de código Java, y un equipo de 50 personas distribuidos en dos empresas ubicadas en países distintos.

El modelo específico de Scrum desarrollado por Ken Schwaber para software contempla el desarrollo de sistemas críticos, incluyendo los requerimientos de conformidad también como resultados entregables de las iteraciones junto con las funcionalidades a las que afectan.

	ADAPTABLE	PREDICTIVA
PRIORIDAD DE NEGOCIO	Valor	Cumplimiento
ESTABILIDAD DE REQUISITOS	Entorno estable	Entorno inestable
RIGIDEZ DEL PRODUCTO	Difícil de modificar	Modificable
COSTE DE PROTOTIPADO	Bajo	Alto
CRITICIDAD DEL SISTEMA	Alta	Baja
TAMAÑO DEL SISTEMA	Grande	Pequeño

Figura (5.0): Criterios de idoneidad para un proyecto.

Prioridad de negocio

¿Cuál es la principal prioridad para los intereses de negocio del cliente? ¿Qué tiene más importancia: el cumplimiento de agendas y fechas o el valor innovador del producto? Este es el primer aspecto que se debe considerar. La gestión predictiva es un modelo construido y especializado en garantizar el cumplimiento de los planes. La gestión adaptable es un modelo construido y especializado en dar el mayor valor posible al producto.

Por supuesto los dos objetivos son deseables, pero hay que elegir, porque simplemente son excluyentes. No se pueden planificar diagramas de Gantt o rutas críticas sobre una visión general. Cuanto mayor valor se desea en uno u otro extremo (valor o predicción), más contraproducente resulta emplear el estilo de gestión inadecuado.

Estabilidad de los requisitos

¿Se puede obtener una descripción de requisitos detallada al inicio del proyecto, y ésta se mantendrá estable durante el desarrollo? O lo que es lo mismo, ¿se puede

saber con certeza y detalle qué es lo que se quiere construir, siendo improbable que cambien los criterios o las necesidades?

Rigidez del producto

¿Cuan fácil o difícil resulta modificar el producto? Esta es una interrogante básica, porque no es lo mismo modificar un software, circuitos electrónicos, construcciones civiles, etc. Modificar la estructura de una base de datos es viable y los parámetros para determinar la complejidad de dicha modificación, dependerá de la comparación que se haga con otras modificaciones que ocurran en cualquier ámbito, así por ejemplo si se compara con la de la estructura de un edificio para rectificar el número de plantas, la transformación de una base de datos resulta ser extremadamente fácil.

Costo de los prototipos

Otra cuestión relevante para el modelo de gestión ágil es la relación: costo del prototipo / valor conseguido para el producto. Este factor suele estar relacionado con la rigidez del producto. Ver, tocar e interactuar con las partes ya desarrolladas o con simulaciones o prototipos, genera ideas y posibilidades que sobre el concepto inicial y en el papel no llegan a concebirse. Los prototipos y el feed-back que proporciona son extremadamente importantes, sobre todo en el desarrollo de nuevos productos, o de sistemas innovadores. A medida que el equipo lo va “tocando” y “probando” surgen funcionalidades y posibilidades nuevas que aportan mayor valor al concepto inicial. En este sentido, el argumento: “la forma más eficiente de desarrollar un trabajo es hacerlo bien a la primera”, que se emplea con frecuencia para defender la validez de la gestión predictiva en cualquier proyecto, resulta tendencioso.

La afirmación “per se” es obviamente cierta; pero también son ciertas dos circunstancias relacionadas: Se puede hacer “bien a la primera” cuando es posible

conocer con detalle el resultado sin necesidad de hacer pruebas antes. Las posibilidades al hacer un trabajo no son sólo “bien” o “mal”. Bien es un término amplio. Puede ser aceptable o suficientemente bien, o lo mejor posible. Estos factores, junto con la relación entre costo de los prototipos y valor que aporta, deben tenerse también en cuenta para elegir el modelo de gestión más adecuado para el proyecto.

Criticidad del sistema

La evaluación estructurada de las características del producto Seguridad, complejidad, rendimiento, entre otros para determinar la severidad del impacto de una falla en el sistema, de su degradación o de su no cumplimiento con los requisitos o los objetivos del sistema o dicho de otra forma, si el sistema falla, degradándose o no consiguiendo realizar las funciones de los requisitos, ¿qué impacto podría tener en la seguridad o en el rendimiento? La respuesta no es única, puede presentarse una amplia gama de impactos, a continuación se exponen algunos, ordenados de mayor a menor gravedad:

- Causará daño a las personas.
- Causará daño al medio ambiente.
- Producirá pérdidas económicas graves.
- Producirá pérdidas económicas.
- Fallará la finalidad principal del sistema.
- Fallarán funcionalidades auxiliares del sistema.
- Se producirán fallos de organización ó de comodidad para los usuarios.

Tamaño del sistema

Una de las principales bases del desarrollo ágil es la preferencia de la comunicación e interacción directa de los implicados en el proyecto. Los grandes proyectos implican equipos numerosos y en ocasiones físicamente distantes, circunstancias que dificultan la comunicación directa. No obstante hay desarrollos incipientes de prácticas ágiles que implantan esquemas de agrupamiento y comunicación directa en estructuras celulares de equipos de hasta 6 personas.

Condiciones de la organización

Los elementos empleados por las organizaciones para ejecutar proyectos son: personas, procesos y tecnología. Los resultados de la gestión ágil dependen más del valor de las personas que de los procesos de la organización.

Las personas tienen características propias:

- Sus resultados son “sensibles” al entorno. La falta de motivación y los ambientes laborales hostiles reducen significativamente el valor intelectual del trabajo.
- Cuando el trabajo depende del talento, la diferencia de valor entre los mediocres y los mejores es muy grande.

Adoptar modelos de desarrollo ágil no consiste sólo en realizar las prácticas formales: equipo único, reuniones periódicas, desarrollo evolutivo de los requisitos, etc. Si la organización mantiene un modelo de desarrollo basado en procesos y no en personas, y no tiene la correlatividad necesaria con los principios ágiles: cultura y estructura organizativa, no obtiene los resultados propios del desarrollo ágil.

Nivel profesional

“En el mundo del diseño informático, los mejores lo hacen entre 50 y 100 veces mejor que el promedio, y la cifra aumenta, conforme se incrementa la complejidad de la tecnología”¹²

“La diferencia entre los promedios y los mejores ya no es de 1:2, como en el pasado. Es 1:100 o incluso 1:1000”¹³

Si el proyecto, más que innovación, lo que requiere es la ejecución controlada de un plan detallado, posiblemente sean los procesos de la organización los garantes del resultado; y con un modelo de gestión predictiva, el factor relevante sea la capacidad de los procesos empleados, y no tanto el nivel profesional de las personas del equipo. Si por ser el valor del producto el objetivo del proyecto, se emplea un modelo de desarrollo ágil, son las personas y no los procesos los encargados de proporcionarlo, y en ese caso el equipo debe estar compuesto por personas con el mayor conocimiento y experiencia posibles.

Cultura organizativa

Para la ejecución sistemática y controlada de procesos, no resulta especialmente relevante el tipo de cultura de la organización. Sin embargo, para el desarrollo de trabajo basado en el talento de las personas resultan inhibidores los ambientes laborales basados en el control, excesivamente normalizados y jerarquizados.

¹² Pilar Jericó. “La gestión del talento”, Prentice-Hall, Madrid, 2008

¹³ Nathan Myhrvold (Ex-director de I+D de Microsoft)

Modelo de desarrollo

Los entornos de desarrollo basados en procesos son adecuados para modelos de gestión predictiva. Los entornos de desarrollo basados en las personas son adecuados para modelos de gestión ágil.



Figura (5.0): Principales contrastes que diferencian el desarrollo tradicional del ágil.

2.2 SAP ERP

SAP AG (*Systeme, Anwendungen und Produkte*) (Sistemas, Aplicaciones y Productos), con sede en Walldorf (Alemania), es el primer proveedor de software empresarial en el mundo. Como empresa, comercializa un conjunto de aplicaciones de software para soluciones integradas de negocios, entre ellas mySAP Business Suite, que provee soluciones escalables que permiten mejorar continuamente, con más de 1.000 procesos de negocio consideradas las mejores prácticas empresariales.

SAP es considerada como el tercer proveedor independiente de software del mundo y el mayor fabricante europeo de software; con 12 millones de usuarios,

100.700 instalaciones y más de 1.500 socios, es la compañía más grande de software Inter-empresa. A finales de 2005, SAP empleaba a 35.873 personas en más de 50 países y sus ingresos anuales fueron de 8.513 millones de euros.

SAP fue fundada en 1972 en la Ciudad de Mannheim, Alemania, por antiguos empleados de IBM (Claus Wellenreuther, Hans-Werner Héctor, Klaus Tschira, Dietmar Hopp y Hasso Plattner) bajo el nombre de "SAP Systemanalyse, Anwendungen und Programmentwicklung". El nombre fue tomado de la división en la que trabajaban en IBM.

La corporación se ha desarrollado hasta convertirse en la quinta más grande compañía mundial de software. El nombre SAP R/3 es al mismo tiempo el nombre de una empresa y el de un sistema de información. Este sistema comprende muchos módulos completamente integrados, que abarca prácticamente todos los aspectos de la administración empresarial. Ha sido desarrollado para cumplir con las necesidades crecientes de las organizaciones mundiales y su importancia está más allá de toda duda. SAP ha puesto su mirada en el negocio como un todo: así ofrece un sistema único que soporta prácticamente todas las áreas en una escala global. SAP proporciona la oportunidad de sustituir un gran número de sistemas independientes, que se han desarrollado e instalado en organizaciones ya establecidas, por un solo sistema modular. Cada módulo realiza una función diferente, pero está diseñado para trabajar con otros módulos. Está totalmente integrado, ofreciendo real compatibilidad a lo largo de las funciones de una empresa.

2.3 Metodología para implementación de proyectos ASAP

ASAP es un acrónimo para la metodología AcceleratedSAP, desarrollada por SAP AG para la implementación de proyectos de sus soluciones de software. Inicialmente fue creada para dar un orden estándar en la ejecución de las actividades tanto de consultoría como de logística y otros aspectos necesarios para que los proyectos de implementación del ERP de la compañía (SAP R/3) funcionasen con el

menor riesgo posible, luego fue evolucionando para abarcar la implementación de las diferentes soluciones de negocios, como por ejemplo mySAP CRM, PLM, SCM, SRM, Enterprise Portals, y otras.

El nombre "Accelerated SAP" pretende dar a entender la posibilidad de realizar implementaciones en tiempos cortos, a diferencia de lo que había estado sucediendo cuando las implementaciones se realizaban sin una metodología estándar, cuando se dibujaban proyectos que abarcaban años de trabajo, sin entregables definidos, dejando sin cubrir las expectativas de los clientes o empresas que adoptaban las soluciones, lo cual generaba un ambiente adverso en los proyectos en cuestión.

La metodología propone la implementación de un proyecto de la mayoría de las soluciones de negocios basados en actividades definidas en un Roadmap (hoja de ruta). El roadmap de implementación (o Implementation Roadmap) es el más común de todos, y divide las actividades y tareas en cinco fases principales:

- Fase 1: Preparación Inicial (Initial Preparation).
- Fase 2: Mapa de procesos (Business Blueprint).
- Fase 3: Realización (Realization).
- Fase 4: Preparación final (Final preparation).
- Fase 5: Inicio y soporte post-productivo (Go Live and Support).

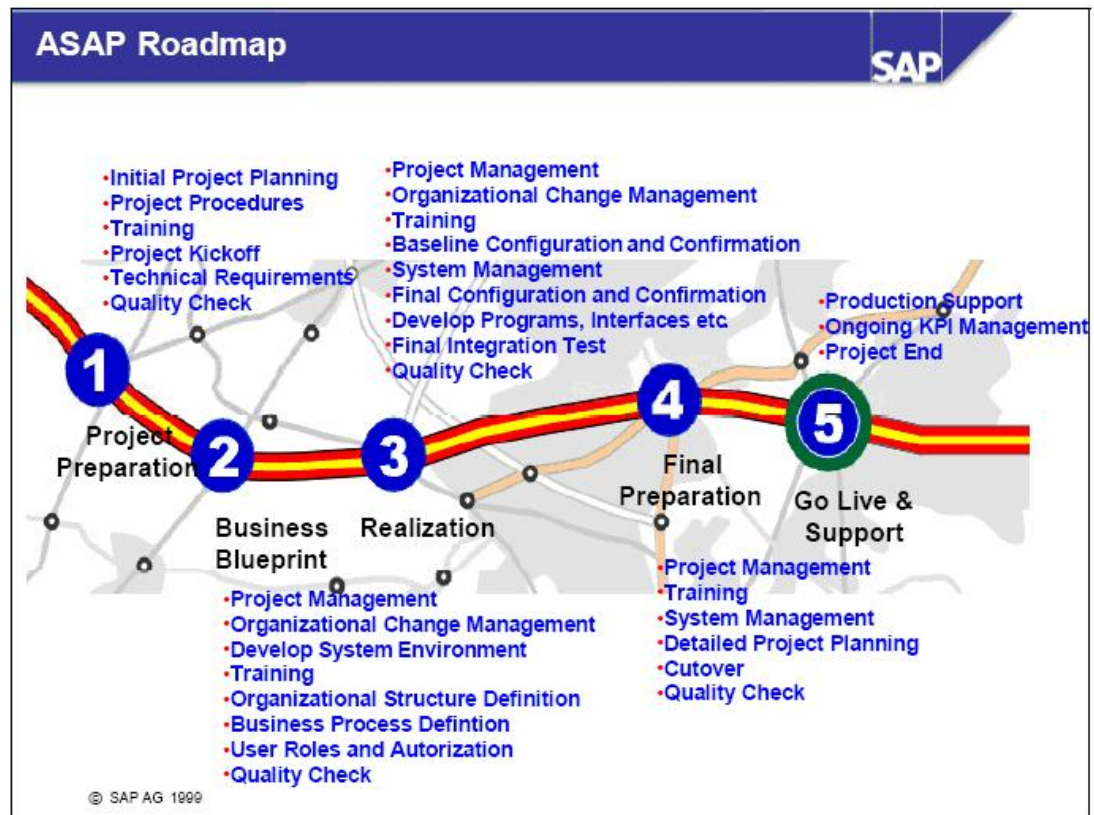


Figura (6.0): Fases del ciclo de vida metodología ASAP.

Hasta el año 2004, la metodología estaba soportada por la herramienta llamada "ASAP Implementation Tools", gratuita, que se entregaba en CDs y se instalaba en un PC o servidor, de forma individual (stand-alone) o como Cliente-Servidor, que disponía de generadores de informes estructurados pero carecía de herramienta de gestión del proyecto. Desde ese año, SAP liberó la nueva herramienta SAP Solution Manager, incluida en SAP Business Suite, que requiere un servidor y una administración similar a otros productos SAP, que permite realizar la documentación del proyecto y dispone de una herramienta ligera de seguimiento del proyecto de implementación o de cualquiera de sus soluciones, incluyendo Upgrades (actualizaciones de versión) y creación de nuevas funcionalidades. Las anteriores herramientas fueron discontinuadas.

2.4 SCRUM

El origen

Scrum es una metodología ágil de desarrollo de proyectos que toma su nombre y principios de las observaciones sobre nuevas prácticas de producción, realizadas por Hirotaka Takeuchi e Ikujiro Nonaka a mediados de los 80. Aunque las prácticas observadas por estos autores surgieron en empresas de productos tecnológicos, también se emplean en entornos que trabajan con requisitos inestables y que requieren rapidez y flexibilidad, situaciones frecuentes en el desarrollo de determinados sistemas de software. Jeff Sutherland aplicó los principios observados por Nonaka y Takeuchi al desarrollo de software en 1993 en Easel Corporation (Empresa que en los macrojuegos de compras y fusiones se integraría en VMARK, luego en Informix y finalmente en Ascential Software Corporation). En 1996 lo presentó junto con Ken Schwaber como proceso formal, también para gestión del desarrollo de software en OOPSLA 96. Más tarde, en 2001 serían dos de los promulgadores del Manifiesto ágil.

Introducción al modelo

Scrum es una metodología de desarrollo muy simple, que requiere trabajo duro, porque no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias de la evolución del proyecto. Como método ágil:

- Es un modo de desarrollo adaptable, antes que predictivo.
- Orientado a las personas, más que a los procesos.
- Emplea el modelo de construcción incremental basado en iteraciones y revisiones.

Comparte los principios estructurales del desarrollo ágil: A partir del concepto o visión de la necesidad del cliente, construye el producto de forma incremental a través

de iteraciones breves que comprenden fases de especulación – exploración y revisión. Estas iteraciones (en Scrum llamadas sprints) se repiten de forma continua hasta que el cliente da por cerrado el producto.

Se comienza con la visión general del producto, especificando y dando detalle a las funcionalidades o partes que tienen mayor prioridad de negocio, y que pueden llevarse a cabo en un periodo de tiempo breve (según los casos pueden tener duraciones desde una semana hasta no más de dos meses). Cada uno de estos periodos de desarrollo es una iteración que finaliza con la entrega de una parte (incremento) operativa del producto. Estas iteraciones son la base del desarrollo ágil, y Scrum gestiona su evolución en reuniones breves diarias donde todo el equipo revisa el trabajo realizado el día anterior y el previsto para el siguiente.

Control de la evolución del proyecto

Scrum controla de forma empírica y adaptable la evolución del proyecto, a través de las siguientes prácticas de la gestión ágil:

- **Revisión de las Iteraciones:** Al finalizar cada iteración (sprint) se lleva a cabo una revisión con todas las personas implicadas en el proyecto. Es por tanto la duración del sprint, el periodo máximo que se tarda en reconducir una desviación en el proyecto o en las circunstancias del producto.
- **Desarrollo incremental:** Las personas implicadas no trabajan con diseños o abstracciones. El desarrollo incremental implica que al final de cada iteración se dispone de una parte de producto operativa, que se puede inspeccionar y evaluar.
- **Desarrollo evolutivo:** Los modelos de gestión ágil se emplean para trabajar en entornos de incertidumbre e inestabilidad de requisitos. Intentar predecir

en las fases iniciales cómo será el resultado final, y sobre dicha predicción desarrollar el diseño y la arquitectura del producto, no es realista, porque las circunstancias obligarán a remodelarlo muchas veces. ¿Para qué predecir los estados finales de la arquitectura o del diseño si van a estar cambiando? Scrum considera a la inestabilidad como una premisa, y se adoptan técnicas de trabajo para permitir la evolución sin degradar la calidad de la arquitectura que también evoluciona durante el desarrollo. Durante éste se genera el diseño y la arquitectura final de forma evolutiva. Scrum no los considera como productos que deban realizarse en la primera “fase” del proyecto. (El desarrollo ágil no es un desarrollo en fases).

- **Auto-organización:** En la ejecución de un proyecto son muchos los factores impredecibles en todas las áreas y niveles. La gestión predictiva confía la responsabilidad de su resolución al gestor de proyectos. En Scrum los equipos son auto-organizados (no auto-dirigidos), con margen de decisión suficiente para tomar las decisiones que consideren oportunas.
- **Colaboración:** Las prácticas y el entorno de trabajo ágiles facilitan la colaboración del equipo. Ésta es necesaria, porque para que funcione la auto-organización como un control eficaz cada miembro del equipo debe colaborar de forma abierta con los demás, según sus capacidades y no según su rol o su puesto.
- **Visión general del proceso:** Scrum denomina “sprint” a cada iteración de desarrollo y según las características del proyecto y las circunstancias del sprint puede determinarse una duración desde una semana hasta dos meses, aunque no suele ser recomendable hacerlos de más de un mes. El sprint es el núcleo central que proporciona la base de desarrollo iterativo e incremental.

Los elementos que conforman el desarrollo Scrum son:

- **Planificación del sprint:** Jornada de trabajo previa al inicio de cada sprint, en la que se determina cuál va a ser el trabajo y los objetivos que se deben conseguir en la iteración.

- **Seguimiento del sprint:** Breve revisión diaria, en la que cada miembro describe tres cuestiones:
 1. El trabajo que realizó el día anterior.
 2. El que tiene previsto realizar.
 3. Cosas que puede necesitar o impedimentos que deben suprimirse para realizar el trabajo. Cada persona actualiza en la pila del sprint el tiempo pendiente de sus tareas, y con esta información se actualiza también el gráfico con el que el equipo monitoriza el avance del sprint (burn-down)

- **Revisión del sprint:** Análisis y revisión del incremento generado.

Los elementos

- **Pila del producto (product backlog):** Lista de requisitos de usuario que a partir de la visión inicial del producto crece y evoluciona durante el desarrollo.

- **Sprint:** Periodo de trabajo para desarrollar un incremento del producto.

- **Pila del Sprint (sprint backlog):** Lista de los trabajos que debe realizar el equipo durante el sprint para generar el incremento previsto.

- **Incremento:** Resultado de cada Sprint.

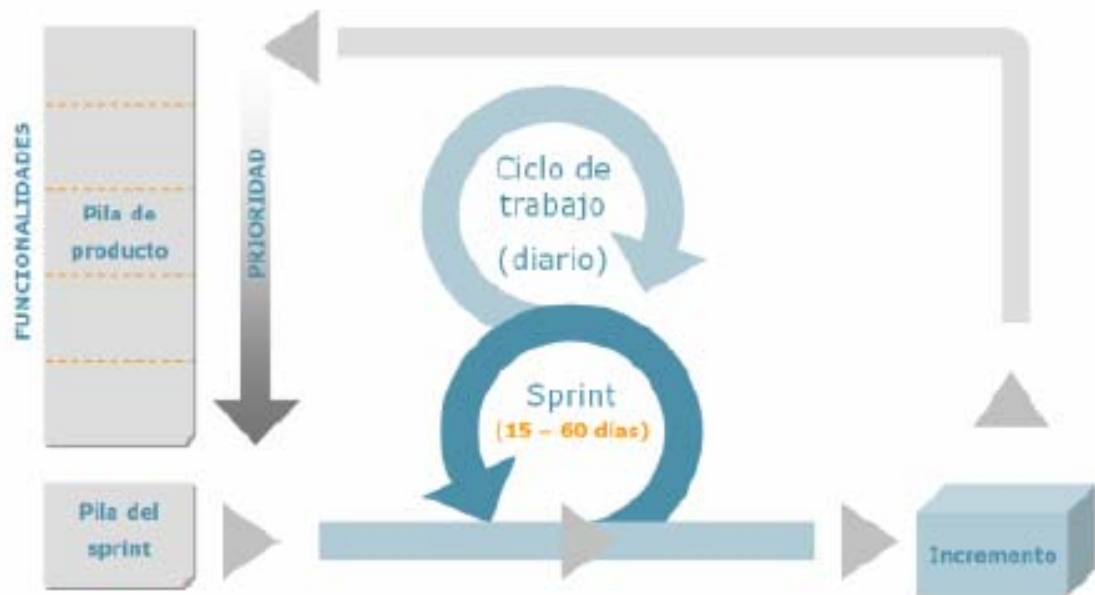


Figura (7.0): Flujo de las fases y actividades de SCRUM.

Los roles

Todas las personas que intervienen, o tienen relación directa o indirecta con el proyecto, se clasifican en dos grupos: comprometidos e implicados. En círculos de Scrum es frecuente llamar a los primeros (sin ninguna connotación peyorativa) “cerdos” y a los segundos “gallinas”. El origen de estos nombres es esta metáfora que ilustra de forma gráfica la diferencia entre “compromiso” e “implicación” con el proyecto:

Una gallina y un cerdo paseaban por la carretera. La gallina preguntó al cerdo: “¿Quieres abrir un restaurante conmigo?”. El cerdo consideró la propuesta y respondió: “Sí, me gustaría. ¿Y cómo lo llamaríamos?”. La gallina respondió: “Huevos con tocino”. El cerdo se detuvo, hizo una pausa y contestó: “Pensándolo mejor, creo que no voy a abrir un restaurante contigo. Yo estaría realmente comprometido, mientras que tu estarías sólo implicada”.

COMPROMETIDOS (cerdos)	IMPLICADOS (gallinas)
Propietario del producto, Equipo de desarrollo.	Otros interesados (Dirección general, Dirección comercial, Marketing, Usuarios, etc.)

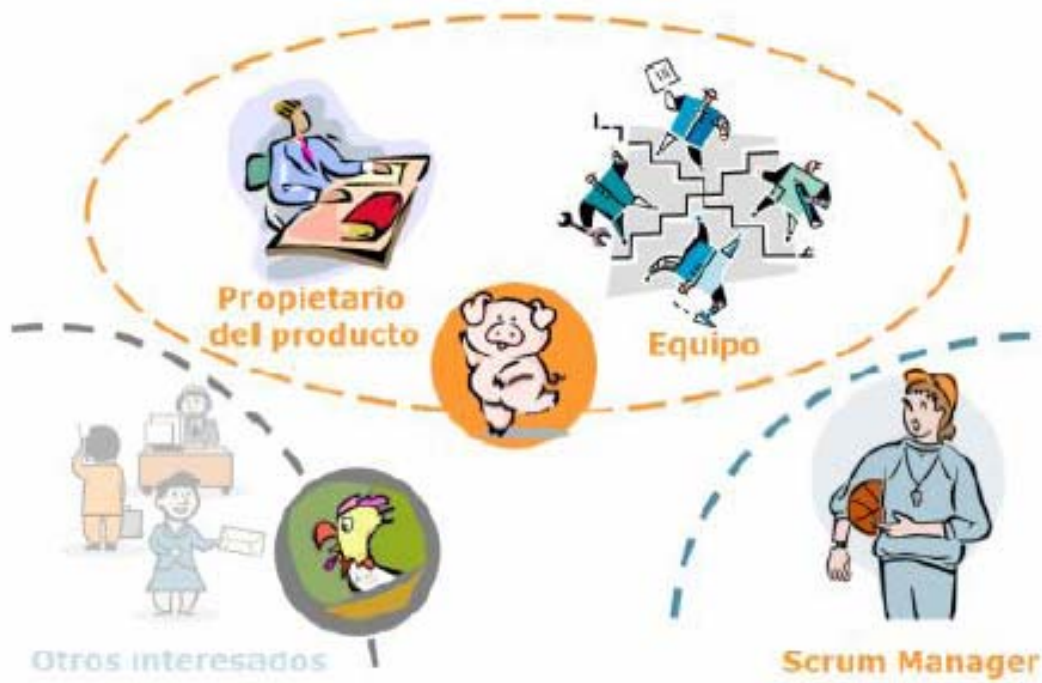


Figura (8.0): Ilustración de la diferencia entre cerdos y gallinas.

Propietario del producto (Product Owner): Es la persona responsable de lograr el mayor valor de producto para los clientes, usuarios y resto de implicados.

- Equipo de desarrollo: grupo o grupos de trabajo que desarrollan el producto.
- Scrum Manager – Team Leader – Scrum Master: Responsable del funcionamiento de la metodología Scrum.

Valores

Scrum es una “carrocería” que da forma a los principios ágiles. Es una ayuda para organizar a las personas y el flujo de trabajo. La carrocería sin motor, sin los valores que dan sentido al desarrollo ágil, no funciona:

- Delegación de atribuciones (*empowerment*) al equipo para que pueda auto-organizarse y tomar las decisiones sobre el desarrollo.
- Respeto entre las personas. Los miembros del equipo deben confiar entre ellos y respetar sus conocimientos y capacidades.
- Responsabilidad y auto-disciplina (no disciplina impuesta).
- Trabajo centrado en el desarrollo de lo comprometido
- Información, transparencia y visibilidad del desarrollo del proyecto

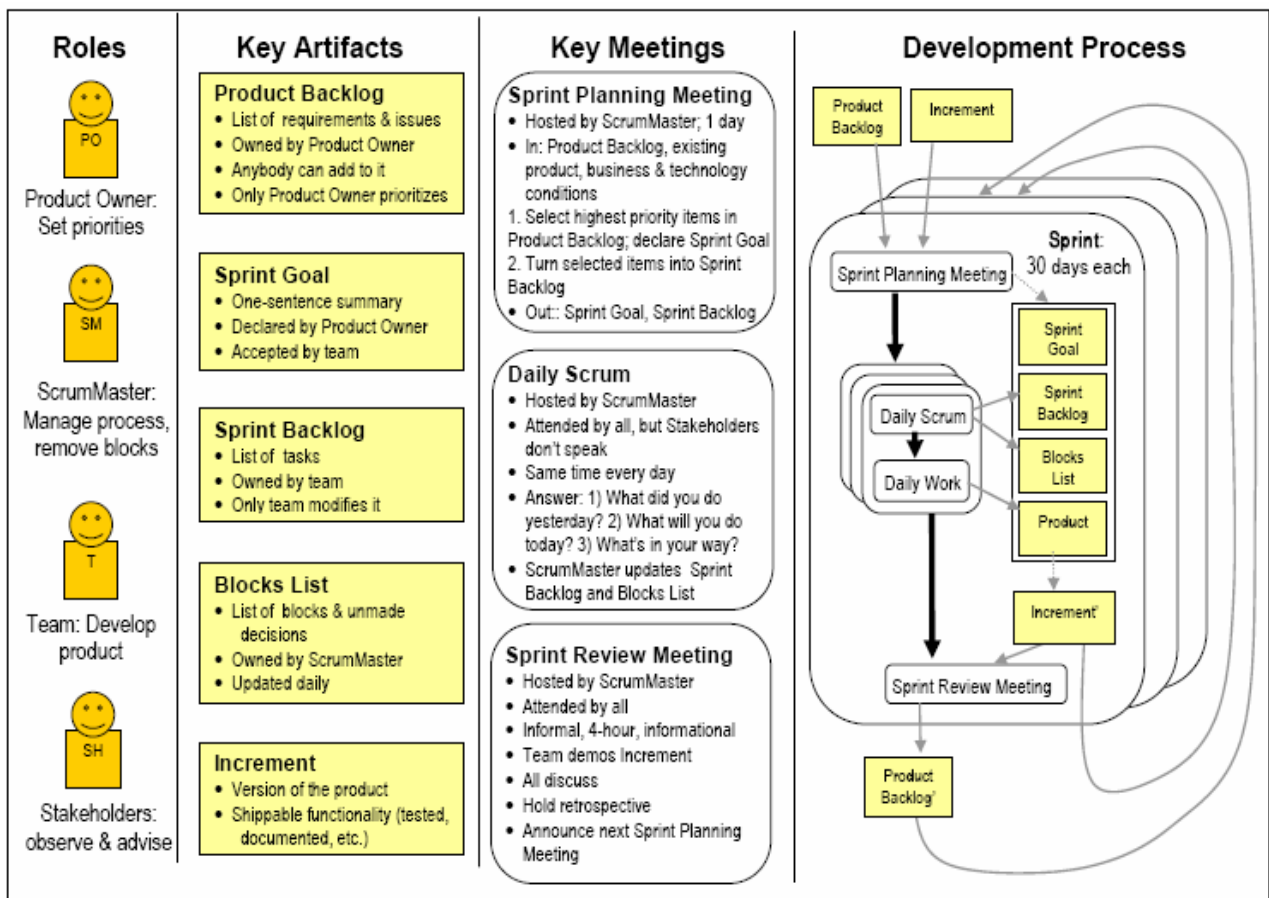


Figura (9.0): Descripción de roles, artefactos, reuniones y proceso de desarrollo de Scrum. William C. Wake.

3 Capítulo III - Desarrollo del Trabajo

3.1 Tecnología involucrada

El proyecto se llevará a cabo en el entorno suministrado por SAP para los desarrollos de aplicaciones WEB llamado WebDynpro¹⁴, con arquitectura J2EE y con SAP NetWeaver Development Studio como GUI¹⁵ para la construcción de las aplicaciones, todo soportado por un servidor de aplicaciones SAP WAS¹⁶. La comunicación entre el servidor SAP WAS y SAP ERP será realizada mediante conectores JCO¹⁷ y configurados desde cada servidor.

Web Dynpro JAVA

La tecnología "Web Dynpro" es el recurso que ofrece SAP Web Application Server para manejar la capa de presentación de las aplicaciones, es decir, la interfaz de usuario de las aplicaciones Web.

Totalmente integrado al SAP NetWeaver Application Server, Web Dynpro provee un conjunto de herramientas para modelar y diseñar las interfaces de usuario basándose en el paradigma MVC (Model View Controller).

Las herramientas de diseño de WebDynpro son independientes del entorno de ejecución, lo que permite unificar el desarrollo de interfaces para los distintos estándares soportados por el Web Application Server: ABAP, J2EE y .NET.

¹⁴ Web Dynamic Program

¹⁵ Graphical User Interface

¹⁶ SAP Web Application Server

¹⁷ Java Connector

Las aplicaciones Web Dynpro Java son desarrolladas dentro del SAP NetWeaver Developer Studio¹⁸.

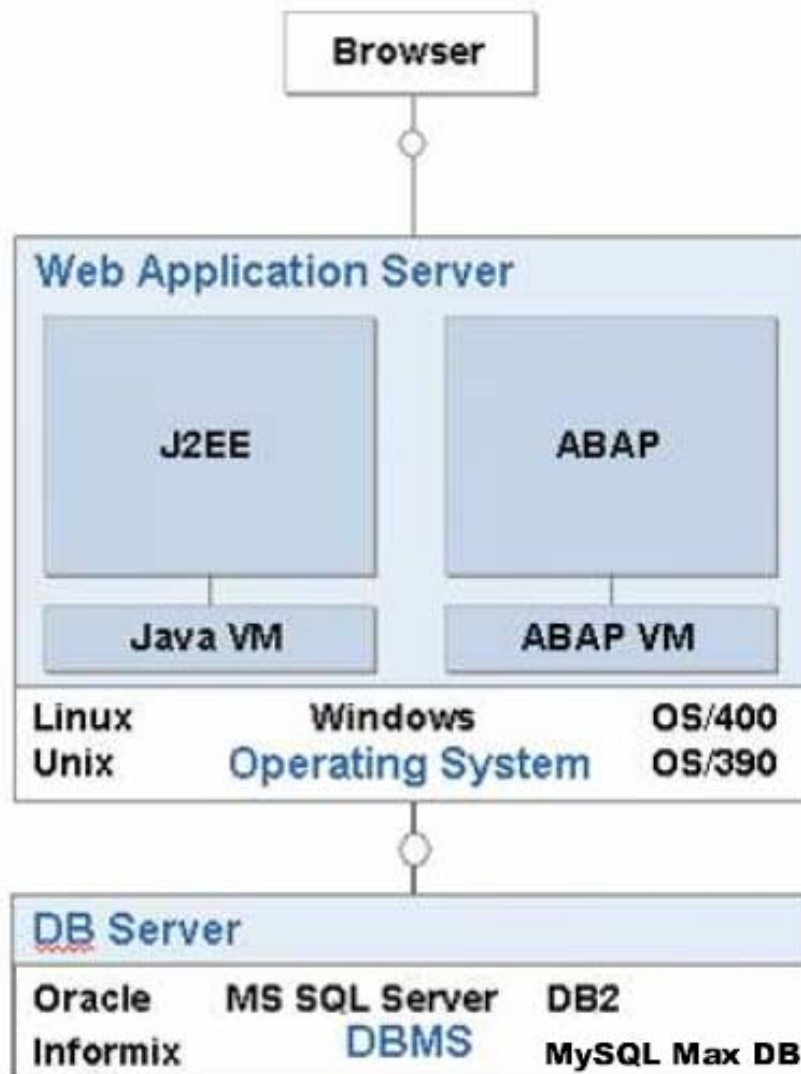


Figura (10.0): Esquema de la arquitectura utilizada.

¹⁸ Entorno de programación proveído por SAP para programar las aplicaciones.

3.2 Organización del proyecto

En Scrum un proyecto se ejecuta en bloques temporales (iteraciones) de un mes natural (en el trabajo se utilizaron iteraciones de seis días). Cada iteración tiene que proporcionar un resultado completo, un incremento de producto que sea susceptible de ser entregado con el mínimo esfuerzo cuando el cliente lo solicite.

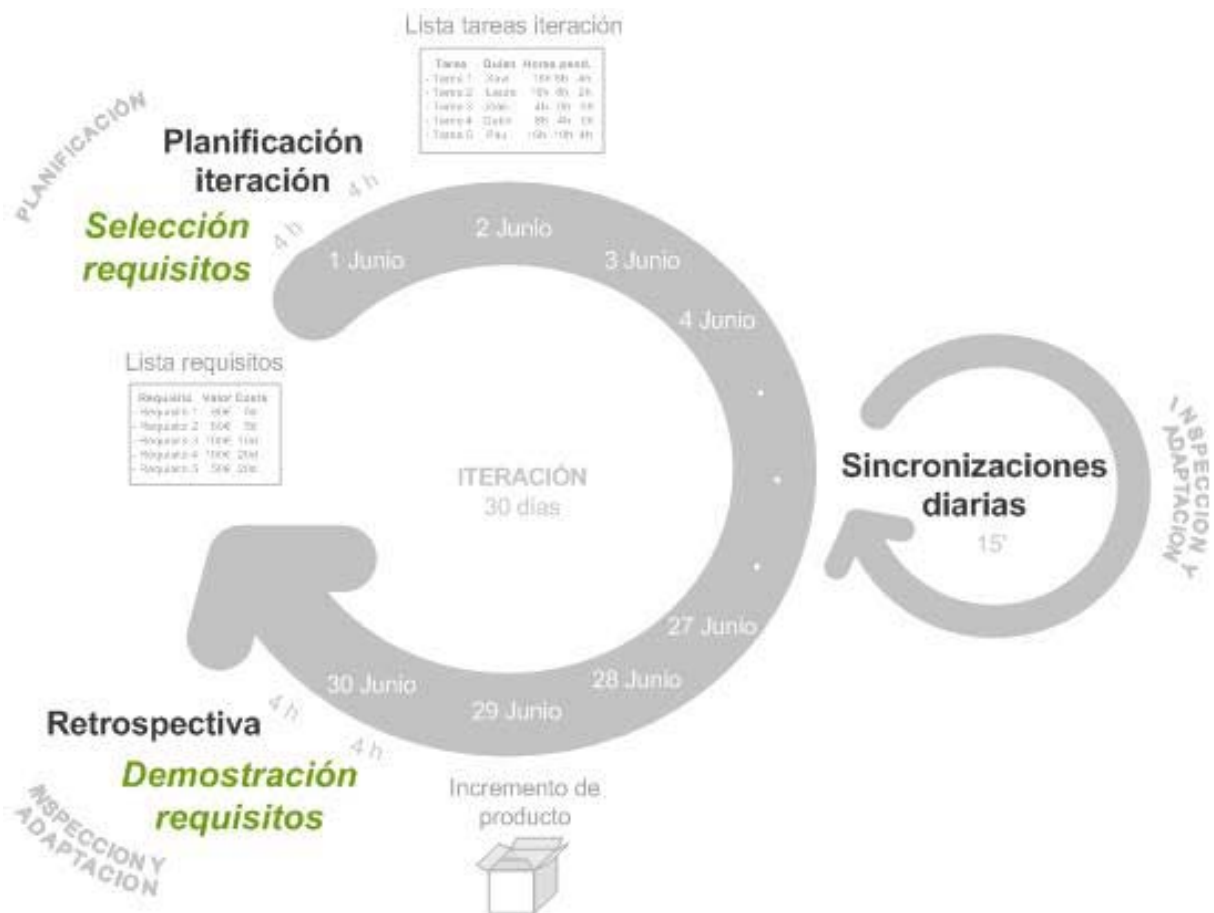


Figura (11.0): Ilustración de las fases, actividades y artefactos utilizados en SCRUM.

El proceso parte de la lista de requisitos priorizada del producto, que actúa como plan del proyecto. En esta lista el cliente ha priorizado los requisitos balanceando el valor que le aportan y han sido divididos en iteraciones y entregas.

En esta sección se explicará cómo se organizó el proyecto y qué fases se realizaron. Considerando que el desarrollo está inmerso en una metodología rígida recomendada por SAP, fue necesario seguir algunos protocolos que se describen más abajo, pero de igual forma los utilizamos como entradas para SCRUM.

3.2.1 Fase 1 - Preparación del Product Backlog

En un proyecto de implementación SAP ERP, una de las fases más importantes descritas en la metodología ASAP es la Fase 2: Mapa de procesos (Business Blueprint). En esta fase se elabora, entre otros documentos, un archivo Word llamado comúnmente BBP (por lo de Business Blueprint).



Figura (12.0): ASAP Roadmap con fases de la metodología

Dentro de las secciones más importantes de un BBP podemos mencionar:

1. Descripción de la situación actual (AS IS)

Detalle de la situación actual	: Descripción de cómo se ejecutan los procesos y actividades dentro de la organización.
Requerimientos del Negocio	: Requerimientos de parte del cliente para mejorar sus procesos.

Salidas requeridas del Proceso : Qué se espera como resultado del proceso descrito.

2. Solución Propuesta (TO BE)

Descripción general de la solución : Los consultores expertos en SAP responden a las necesidades descritas en el AS IS y cómo serán realizadas en el sistema.

3. Descripción de desarrollos

Todo lo que el consultor Senior responde en la sección TO BE es funcionalidad estándar del sistema SAP. Pero no siempre el sistema cuenta con todas las funcionalidades y reportes requeridos por el cliente. Es por esta razón que el consultor detalla una lista con requerimientos que deben ser cubiertos, los cuales pueden ser: Reportes, Mejoras o Aplicaciones a medida para resolver problemas puntuales de los clientes, como por ejemplo integración con capturadores de datos, carga de datos desde archivos de proveedores, etc.

Revisión de Requerimientos

Terminada la Fase Business Blueprint, el jefe de proyectos ya tiene toda la información necesaria para evaluar qué requerimientos deben ser desarrollados puesto que no se encuentran en el sistema de forma estándar. Para la ejecución y control de la construcción de los requerimientos, el jefe de desarrollos de software del proyecto, elabora una lista de requerimientos, donde se detalla lo siguiente:

- Nombre de requerimiento : ID o forma de identificación del requerimiento.

-
- Tipo de desarrollo : Esta es una identificación del tipo de desarrollo que se necesita clasificarlos en: Carga Inicial, Formularios, Reportes, Ampliaciones, Interfaces.
 - Tipo de solución : Smart Forms, Sap Script, ABAP, Manual, WebDynpro
 - Nombre del programa : Nombre del programa o aplicación.
 - Prioridad : Se cataloga a los desarrollos con prioridad Alta, Media o baja.
 - Estimación : Días aproximados de desarrollo calculados mediante Juicio experto
 - Fecha de inicio : Fecha de Inicio del desarrollo.
 - Fecha de término : Fecha de término del desarrollo
 - Observaciones : Observaciones o consideraciones a tomar en cuenta En el desarrollo.
 - Estado : Corresponde al estado en que se encuentra el desarrollo; 0-Sin Comenzar, 1-En especificación, 2-En Programación, 3-Terminado.

Análisis de Requerimientos

Entregada la lista de requerimientos, la cual fue estimada con juicio experto después de un análisis de manera general, los profesionales expertos en cada módulo SAP proceden a realizar un análisis profundo y detallado de cada requerimiento para construir un documento llamado **Requerimiento Técnico Funcional**, el cual contiene las especificaciones técnicas de cómo el desarrollador debe resolver la problemática dentro del entorno del sistema. La solución del problema expresada en este documento por el experto SAP sólo es a nivel de datos o flujos de información. En términos de solución integral, es el responsable de los desarrollos quien evalúa junto con el jefe de proyectos SAP cuál es la mejor solución tecnológica para llevar a cabo el desarrollo. Dentro de las soluciones tecnológicas tenemos:

- SAP Script : Entorno antiguo para construcción de formularios dentro SAP.
- Smart Forms : Nuevo entorno para construcción de formularios dentro SAP.
- ABAP : Lenguaje de programación de los sistemas SAP.
- LSMW : Entorno de carga de datos masivos proveído por SAP.
- WebDynpro. : Entorno de desarrollo Web integrado con SAP.

Dentro del análisis que se realizó con el equipo tecnológico, se decidió construir una aplicación Web para suplir 3 necesidades relevantes para el negocio: “*Gestión de Personal*”, “*Gestión de Stock y Gestión de Encargos*”, desde los puntos de ventas, y propusimos la arquitectura estándar SAP WebDynpro para este fin.

Construcción del Product Backlog

Tomada la decisión de construir una aplicación con WebDynpro, el jefe de desarrollo procedió a la elaboración de una planilla Excel con todos los requerimientos de lista de desarrollo oficial que debían ser incluidos en el Product Backlog y que correspondían a los requerimientos de Gestión de Personal, *Stock* y *Encargos* con los siguientes datos, que sería la lista oficial para aplicar la metodología SCRUM:

ID Requerimiento : Un identificador único, simplemente un número auto-incremental. Esto nos permite no perder la pista a las historias cuando cambiamos su nombre.

Nombre : Una descripción corta de la historia; por ejemplo, “Ver tu historial de transacciones”. Suficientemente claro como para que el Dueño del Producto comprenda aproximadamente de qué estamos hablando, y suficientemente clara como para distinguirla de las otras historias. Normalmente, 2 a 10 palabras.

Importancia : El ratio de importancia que el Propietario del Producto da a esta historia. Por ejemplo, 10 ó 150. Más alto es más importante. Evitamos el término “prioridad” porque típicamente “1” se considera la “máxima prioridad, lo que es muy incómodo si posteriormente decidíamos que algo es más importante. ¿Qué prioridad le daríamos a ese nuevo elemento? ¿Prioridad 0? ¿Prioridad -1?

Estimación : La valoración inicial del Equipo acerca de cuanto trabajo es necesario para implementar la historia, comparada con otras historias. La unidad son “puntos de historia” y usualmente corresponde a “días-persona ideales”. La Pregunta para el Equipo era: “Si tuvieras el número óptimo de personas para esta historia (ni muchos ni pocos, típicamente 2) y te encerraras en una habitación con cantidad de comida, y trabajaras sin distracciones, ¿en cuántos días saldrías con una implementación terminada, demostrable, testeada y liberable?”. Si la respuesta es “Con

3 personas encerrados en una habitación nos llevaría 4 días”, entonces la estimación inicial son 12 puntos.

Lo importante no es que las estimaciones absolutas sean correctas (es decir, que una historia de 2 puntos deba durar 2 días), lo importante es que las estimaciones relativas sean correctas (es decir, que una historia de 2 puntos debería durar la mitad que una historia de 4 puntos).

Como probarlo : Una descripción a alto nivel de cómo se demostrará esta historia en la Demo al final del Sprint. Se trata, esencialmente, de una simple especificación de un test: “Haz esto, entonces haz lo otro, y entonces debería ocurrir aquello”.

Categoría : Corresponde a un agrupamiento para identificar a el tipo de Desarrollo

Modulo : Esta columna permite visualizar a qué modulo de SAP Corresponde.

Estado : Muestra en el Product Backlog el estado del requerimiento.

Sprint : Muestra qué Sprint fue tomado el requerimiento.

Product Backlog

ID Requerimiento	Importancia	Nombre	Notas	Comprobación	Estimación	Categoría	Modulo	Estado	Sprint
WEBDYNPRO004	1000	Bonos	El jefe de local debe registrar los diferentes bonos de los empleados	Asignar bonos a un conjunto de empleados correspondientes a una tienda (Probar todos los Bonos)	0,5	HR	HR	Terminado	1
WEBDYNPRO005	980	Gestión de Personal - Tipo de día	El jefe de local debe poder corregir la asistencia de los empleados de su tienda (trabajado, libre, ausencia, tantos registros como días en el mes)	Corregir la asistencia de un periodo y comprobarlo en SAP	1	HR	HR	Terminado	1
WEBDYNPRO003	970	Asistencia	El usuario debe poder registrar la hora de llegada y salida	Marcar entrada y salida	1	HR	HR	Terminado	1
WEBDYNPRO001	960	Login	Modelar la solución y considerando los datos necesario.	Acceder al sistema y que entregue los datos del usuario	1	Acceso	Acceso	Terminado	1
WEBDYNPRO002	950	Gestión de Usuarios	Mantenedores necesarios para controlar el acceso al sistema	crear usuarios, modificar usuarios, eliminar usuarios.	0,5	Acceso	Acceso	Terminado	1
WEBDYNPRO028	945	Correcciones HR	Corrección de la opción de bonificación, se agregaran validaciones dependiendo del tipo de usuario y validaciones de bloqueo de usuarios	Asignar bonificaciones con un perfil de Supervisor y otro de Jefe. Probar mensajes de bloqueo de usuario	2	Encargo	POS	Terminado	1
WEBDYNPRO029	944	Correcciones HR Planilla Mensual	Modificar el título del menú "Bonificaciones" por "Planilla Mensual" 0 días, En el caso del ingreso de la "Planilla Mensual" mostrar la "meta tienda" como dato informativo. 0,5 días, Incluir la validación que las metas personales por tienda "cuadre" con la meta tienda. La meta tienda se ingresará a SAP directamente y lo hará el área de Remuneraciones. 0,5 días		1		HR	Pendiente	3
WEBDYNPRO030	943	Correcciones HR - Gestión Planilla Mensual	Incluir el filtro "concepto" (bono, meta, etc.), en la pantalla de visualización por empleado. 0,5 días, Permitir que el Supervisor pueda modificar la información de todos los empleados a su cargo (según tiendas). Incluir filtro por tipo de bono y agregar tienda, ordenar por tienda tipo y agregar total		1,5		HR	Pendiente	3
WEBDYNPRO031	942	Correcciones HR - Datos Generales de Usuario	Mostrar a qué tienda pertenece el empleado que ha ingresado el sistema. 0,5 días, Modificar el título del menú "Bonificaciones" por "Planilla Mensual" 0 días		0,5		HR	Pendiente	3
WEBDYNPRO032	941	Correcciones HR - Control de Cambio por concepto HR	Cada concepto debe tener un control de periodo y fecha límite de ingreso. Por ejemplo: las metas personales no pueden modificarse para un mes anterior. 2,5 días.		2,5		HR	Pendiente	3

Figura (13.0): Parte del Product Backlog que tenemos al 30-10-2008 en el proyecto.

3.2.2 Fase 2 - Análisis funcional y diseño

Como se debe cumplir con algunas formalidades propias de la metodología ASAP, fue necesario incorporar a nuestro trabajo una fase de análisis y diseño. En esta fase se elaboró un documento formal que contenía el detalle de los requerimientos, que permitió gestionar el alcance. De todas formas el documento que se construyó fue una descripción general de la aplicación con sus funcionalidades mínimas para que fueran aprobadas por el cliente.



	<h2>Proyecto Everest</h2>	
<p>1.2 Descripción</p> <p>Gestionar todos aquellos movimientos de stock que se realizan en una tienda son una tarea fundamental para la organización, estos movimientos corresponden a solicitudes de envío, aprobación de solicitudes, envío de traslados, recepciones de traslados, recepciones de proveedor, emisión de guías, inventarios cíclicos, consulta de stock, consumos y cambios de estados de artículos.</p> <p>1.3 Requerimientos</p> <p>La aplicación debe considerar las siguientes funcionalidades:</p> <ul style="list-style-type: none"> • Solicitudes de envío. • Aprobación de solicitudes. • Envío de traslados. • Recepciones de traslados. • Recepciones de proveedor. • Emisión de guías. • Consumos. • Inventarios cíclicos. • Consulta de stock. • Gestión de estados. • Reporte de Diferencias. <p>1.3.1 Solicitudes de Envío</p> <p>Objetivo: Solicitar envío de productos desde un centro a otro centro. Estos centros son las tiendas o los centros de distribución.</p> <p>Descripción: La aplicación permitirá generar una solicitud de Envío y el ingreso de los datos asociados a ella, estos datos se ingresarán a SAP, a través de RFC para que posteriormente puedan ser aprobados.</p> <p>Requerimientos:</p> <ul style="list-style-type: none"> • Consultar Stock: Se debe habilitar la consulta de Stock para un artículo en el centro suministrador usando los siguientes parámetros de entrada: <ul style="list-style-type: none"> ○ Artículo ○ Descripción 		

Figura (14.0): Extracto de uno de los documento de requerimientos de que se construyó para controlar el alcance.

3.2.3 Fase N - Iteraciones Semanales de Desarrollo

Las N iteraciones corresponden a todos los Sprint que vamos a realizar, dentro de los cuales la metodología Scrum realiza el seguimiento y la gestión del proyecto a través de las tres reuniones que forman parte del modelo:

- Planificación del sprint
- Seguimiento del sprint
- Revisión del sprint

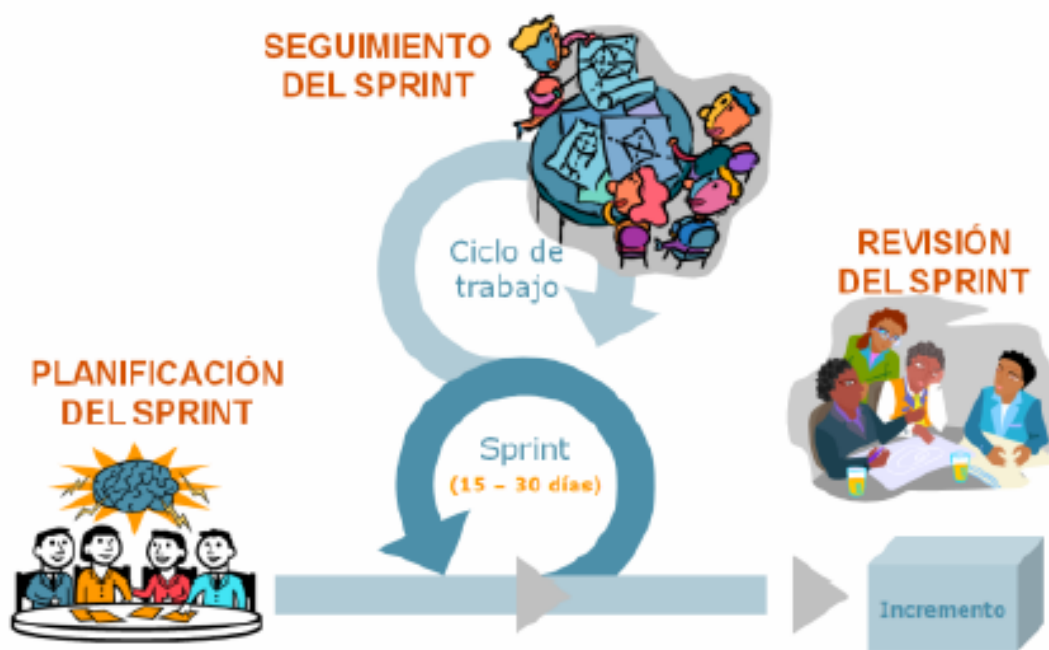


Figura (15.0): Actividades para seguimiento y gestión del proyecto

- **Planificación del Sprint:** En esta reunión se tomaron como base las prioridades y necesidades de negocio del cliente, determinando cuáles y cómo van a ser las funcionalidades que incorporará el producto tras el siguiente sprint.

La reunión consistía en dos partes: La primera decidía qué elementos de la pila del producto se iban a desarrollar, en la segunda se desglosaron éstos para determinar las tareas necesarias, estimar el esfuerzo para cada una, y asignarlas a las personas del equipo. La planificación del sprint no debe durar más de un día.



Figura (16.0): Duración de la planificación del Sprint.

Realizada la reunión de planificación se obtuvo una lista con el detalle de las tareas a realizar para poder cumplir con el Sprint. Además dicha lista fue la forma de controlar la ejecución y el avance diario del Sprint.

		SPRINT	INICIO	DURACIÓN	VELOCIDAD						
		3	28-oct-08	6		M	X	J	L	M	
						28-oct	29-oct	30-oct	31-nov	4-nov	
						Tareas pendientes	17	15	13	10	
						Horas de trabajo pendientes	47	43	39	30	
PILA DEL SPRINT											
Backlog ID	Tarea	Tipo	Estado	Responsable							
WEBDYNPRO029	Validación tiendas por personal	Codificación	En curso	rOlmos	3	3	3				
WEBDYNPRO029	Validación Meta Personal	Pruebas	Pendiente	rOlmos	3	3	3	3			
WEBDYNPRO029	Pruebas de validación	Pruebas	Pendiente	rOlmos	2	2	2	2			
WEBDYNPRO030	Filtro Por Bonos	Codificación	Terminada	jAraneda	2						
WEBDYNPRO030	Filtro Por Tiendas	Codificación	Terminada	jAraneda	2						
WEBDYNPRO030	Modificación Despliegue por tienda, bono	Codificación	Terminada	jAraneda	2	2	2				
WEBDYNPRO030	Modificación de asignación de bonos	Codificación	Terminada	jAraneda	4	4	4				
WEBDYNPRO030	Validación meta tienda	Pruebas	Pendiente	jAraneda	4	4	4	4			
WEBDYNPRO030	Pruebas Funcionalidad Completa	Pruebas	Pendiente	jAraneda	1	1	1	1	1		
WEBDYNPRO031	Modificacion RFC acceso, integración WebDynpro y cambio menu	Codificación	Terminada	jAraneda	3,5	3,5					
WEBDYNPRO031	Pruebas de Cambios	Pruebas	Terminada	jAraneda	0,5	0,5					
WEBDYNPRO032	Reunion con HHRR		Pendiente		2	2	2	2			
WEBDYNPRO032	Analisis de diseño (TABLA)	Análisis	Pendiente		2	2	2	2			
WEBDYNPRO032	Construcción de tabla y mantenedor	Codificación	Pendiente		2	2	2	2			
WEBDYNPRO032	Integración de la funcionalidad Ingreso de planilla mensual	Codificación	Pendiente		6	6	6	6			
WEBDYNPRO032	Integración de la funcionalidad Modificación de planilla mensual	Codificación	Pendiente		6	6	6	6			
WEBDYNPRO032	Pruebas de Cambios	Pruebas	Pendiente		2	2	2	2			

Figura (17.0): Lista de tareas realizadas en el tercer Sprint Backlog.

Para facilitar la comunicación y el trabajo utilizamos un tablón de tareas para el Sprint, el cual contiene las siguientes secciones:

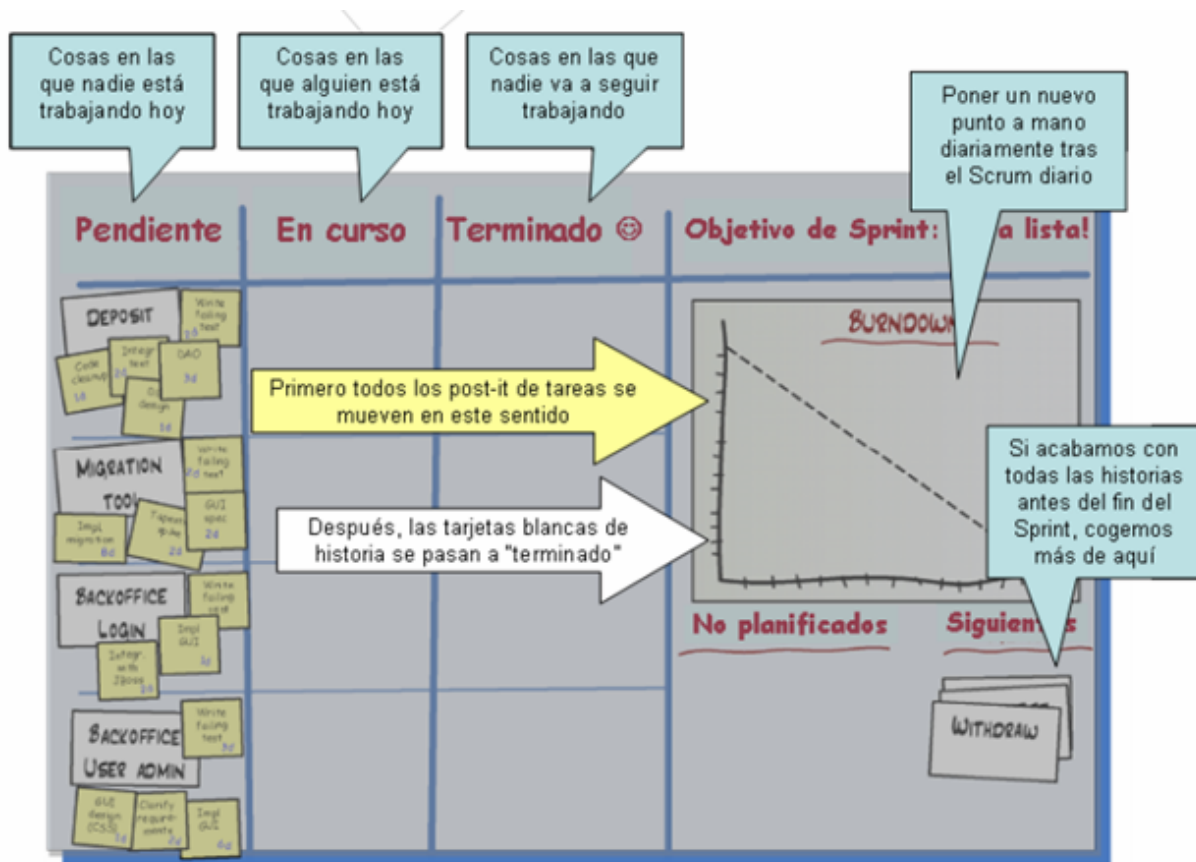


Figura (18.0): Explicación de una pizarra en SCRUM.

Para el tablón de tareas se utilizó un muro de la sala que asignaron al equipo de desarrollo.



- Información General del Sprint:

Sprint Backlog N° 3	
Equipo de desarrollo WebDynpro	
Objetivo	Preparación Aplicaciones HR para pruebas
Velocidad estimada	
Periodo del Sprint	28.10.2009 - 04.11.2009
Scrum Diario	SALA 2 9:30
Fecha Demo	Miercoles 05 de Noviembre
Fecha Revisión	Miercoles 05 de Noviembre
Equipo	Ruby Olmos 100% Juan Aranedo 70%

- Requerimiento y tareas asociadas al Sprint:

Pendiente

Backlog Item # WEBDYNPRO003

Correcciones HR Planilla Mensual

944

1.5

Backlog Item # WEBDYNPRO003

Correcciones HR - Gestión Planilla Mensual

943

1.5

En Curso

VALIDACIÓN META PERSONAL
OLMOS

PRUEBAS FUNCIONALES COMPLETAS.

MODIFICAR INFORMACIÓN DE PERSONAS

- **Seguimiento y Control del Sprint:** Para el seguimiento y control del sprint se efectuó una reunión diaria breve, de no más de 15 minutos, en la que cada miembro del equipo relató lo que realizó el día anterior, las tareas en las que se encontraba trabajando, si le surgió algún tipo de impedimento o problema para realizar su trabajo, y por último, la actualización sobre el tablán de las tareas ya terminadas, o las inconclusas y el tiempo restante para su finalización.

Las horas que quedan para terminar una tarea se actualizan en la planilla de tareas del Sprint mostrada en la planificación del Sprint y dan origen a un grafico burn-down.

Gráfico “burn-down”

Es el gráfico que actualiza el equipo de trabajo en las reuniones de seguimiento del Sprint, para comprobar el ritmo de avance y detectar desde el primer momento si es el previsto o no, ello tiene como fin salvaguardar el tiempo acordado entre partes (cliente y proveedor) a fin de que no se vea comprometida la entrega prevista al final de Sprint. La estrategia ágil para el seguimiento de los proyectos se basa en:

- Medir el esfuerzo que falta, no el realizado.
- Seguimiento muy cercano (diario a ser posible) Y en este gráfico toman forma los dos principios:
 - En el eje Y se registra el trabajo que aún falta por realizar
 - Se actualiza a diario

Día a día, cada miembro del equipo actualiza en la pila del Sprint el tiempo que le queda a las tareas que va desarrollando, hasta que se terminan y van quedando a 0 los tiempos pendientes. La figura 20.0 muestra un ejemplo de pila en el sexto día del Sprint:

las tareas terminadas ya no tienen esfuerzo pendiente, y del esfuerzo total previsto para el Sprint: 276 (A), en el momento actual quedan 110 (B).

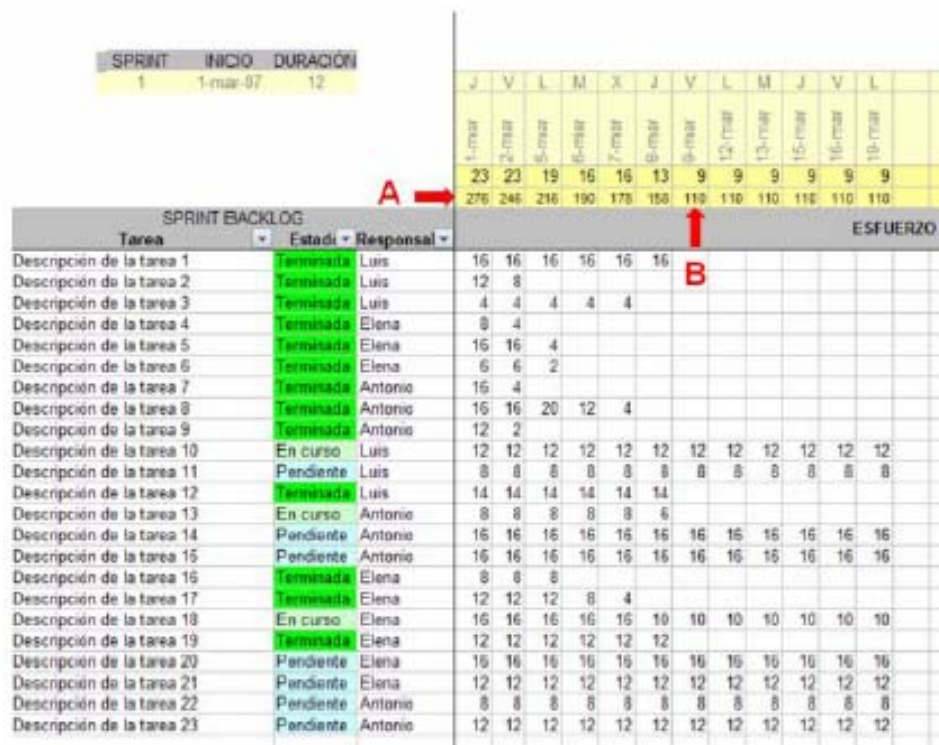


Figura (20.0): Ejemplo pila al sexto día.

Con esta información de la pila del Sprint se actualiza el gráfico, poniendo cada día el esfuerzo pendiente total de todas las tareas que aún no se han terminado.

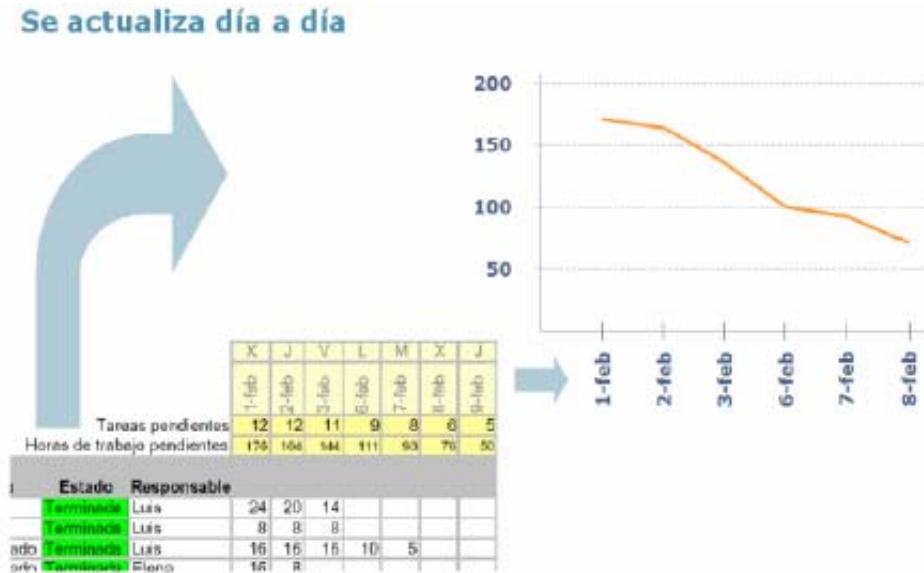


Figura (21.0): La actualización diaria se ve reflejada día a día en el grafico.

El gráfico de avance o burn-down es una herramienta ágil que monitoriza el ritmo de trabajo (normalmente de un sprint). En el eje vertical de un diagrama cartesiano representa el trabajo pendiente a lo largo del tiempo del sprint (eje horizontal). Las desviaciones sobre, o bajo la línea diagonal que representaría el avance ideal del sprint alertan de forma temprana de desviaciones sobre el ritmo de desarrollo previsto.

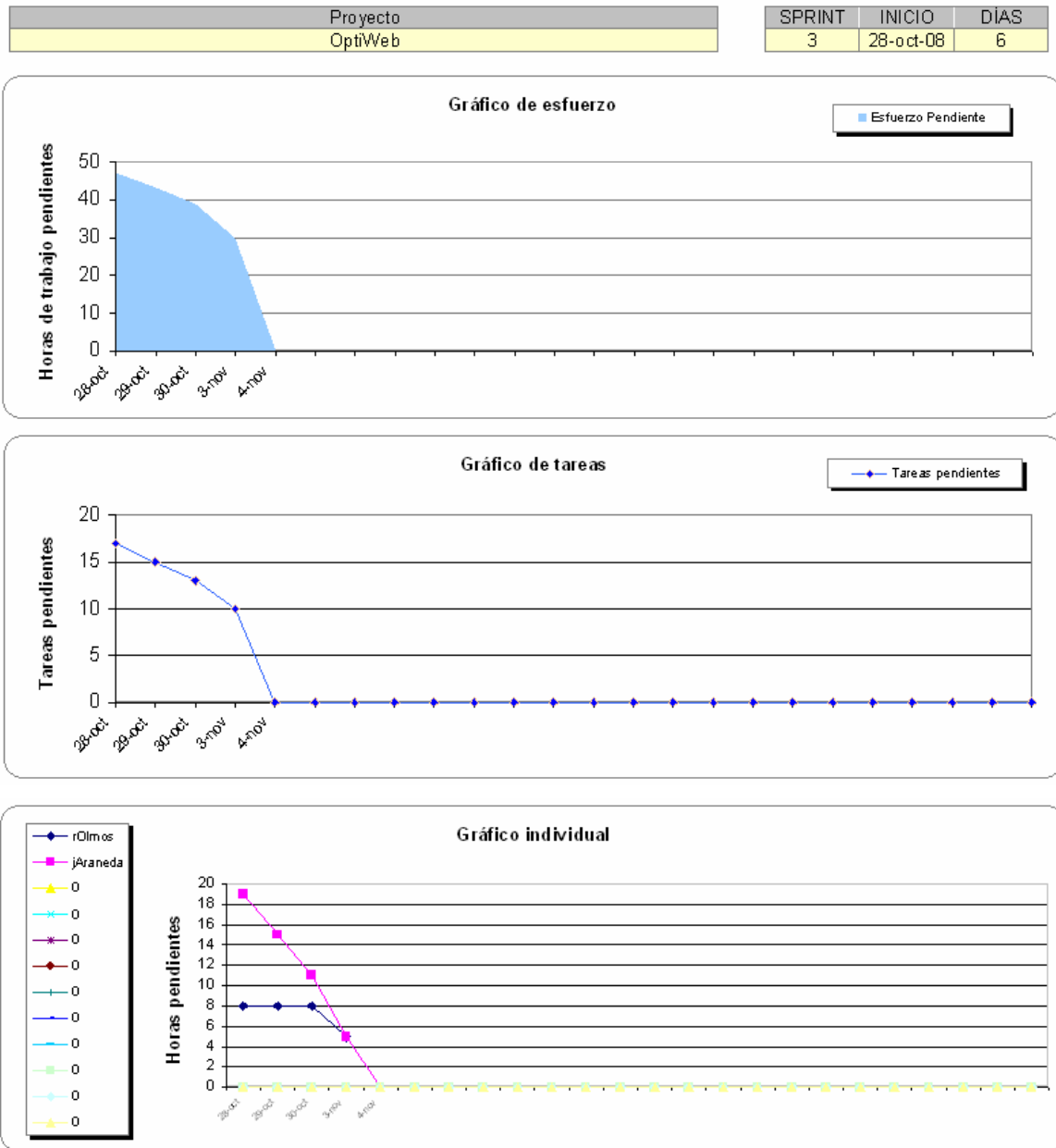


Figura (22.0): Grafico burn-down para monitoriza el avance del sprint del proyecto real.

Trabajo ya realizado

Medir el trabajo ya realizado no entraña especial dificultad. Se puede hacer con unidades relativas al producto (por ejemplo líneas de código) o a los recursos

empleados. Para medirlo basta contabilizar lo ya realizado, empleando las unidades con las que se opere: líneas de código, horas trabajadas (reales o teóricas). Medir el trabajo realizado NO es una métrica Ágil.

LA GESTIÓN ÁGIL NO DETERMINA EL GRADO DE AVANCE DEL PROYECTO POR EL TRABAJO YA REALIZADO, SINO POR EL PENDIENTE DE REALIZAR

Scrum mide el trabajo pendiente para:

- Estimar el esfuerzo y la duración prevista para cada tarea.
 - Determinar el avance del proyecto, y en especial de cada Sprint.
-
- **Revisión del Sprint:** Es la reunión que realizamos al final del Sprint, en la que con una duración máxima de 4 horas, el equipo presenta al propietario del producto, clientes, usuarios, gestores, el incremento construido en el Sprint.

Objetivos

- El propietario del producto obtiene información objetiva del progreso del sistema. Esta reunión marca a intervalos regulares el ritmo de construcción del sistema y la trayectoria que va tomando la visión del producto.
- Al ver y probar el incremento, el propietario del producto y el equipo en general obtienen feedback clave para evolucionar y dar más valor a la pila del producto.
- Otros ingenieros y programadores de la empresa también pueden asistir para conocer cómo trabaja la tecnología empleada.

Resumen

La gestión y evolución de un proyecto con Scrum se determina en tres reuniones:

- Planificación del Sprint
- Seguimiento del Sprint
- Revisión del Sprint

Planificación del Sprint

- Duración máxima 1 día
- Se determinan las funcionalidades que se desarrollarán en el Sprint
- Cada funcionalidad se desglosa en tareas
- Cada tarea se estima y se asigna a una persona del equipo
- El resultado es la pila del Sprint

Seguimiento del Sprint

- Breve reunión diaria en la que el equipo revisa la evolución del Sprint.
- Cada uno expone la tarea en la que ha estado trabajando, en cuál va a trabajar y si necesita algo para poderla realizar.
- Cada miembro actualiza la estimación de tiempo pendiente de sus tareas

Revisión del Sprint

- Duración máxima 4 horas
- Muestra el incremento desarrollado a todas las personas implicadas en el proyecto.

3.3 Recursos utilizados

Los recursos utilizados para el proyecto de desarrollo son los siguientes:

- Una sala sólo para desarrolladores: La sala estuvo compartida por dos equipos de desarrollo: una para desarrollos propios de la implementación SAP ERP, y el otro equipo fue el responsable de llevar a cabo este proyecto de desarrollo en WebDynpro.
- 2 desarrolladores a tiempo completo cada uno con un notebook acondicionados para las necesidades del proyecto.
- 1 Servidor SAP WAS de desarrollo
- 1 Servidor SAP ERP de desarrollo

4 Lecciones aprendidas

- Aunque claramente la metodología Scrum menciona que debe haber un dueño del producto, es importante que participen en las reuniones de planificación del Sprint todas las personas que se verán directa o indirectamente comprometidas por la construcción del entregable de la iteración.
 - **Jefes de Proyecto:** Deben participar los jefes de proyectos tanto del cliente como del proveedor, ya que éstos tienen una visión general del desarrollo del proyecto a implementar, además de conocer los acuerdos contractuales, bajo los cuales se detallan las contraprestaciones de cada parte.
 - **Usuarios claves:** Debe participar, a lo menos un usuario clave, para validar los requerimientos del negocio.
 - **Usuario Final:** Corresponde que participe a lo menos un usuario final para certificar que la construcción cumple con las necesidades de la operación diaria del negocio.
 - **Consultor experto:** Participará un consultor experto en el módulo SAP con el cual la aplicación se integrará. Los aportes que el experto pueda entregar suelen ser de vital importancia, pues su conocimiento hará que no se comprometan funcionalidades que en la práctica son pocos realistas.

Consideraciones Técnicas:

1. Aclarar cada campo de la pantalla dibujado, en forma previa, en la pizarra, independientemente a que durante el transcurso del desarrollo se agreguen más campos o se quiten algunos.
2. Conseguir identificar cada función a realizar, asociada a cada requerimiento.
3. Identificar parámetros y validaciones, análisis de validación ¿SAP o WebDynpro?, ¿son reutilizables? Cuestionar los sucesos que puedan provocar cada evento (a veces un pequeño cambio puede provocar mucho trabajo).
4. Discutir cada componente de software a realizar por el equipo de desarrollo, al menos las funciones a usar en común.
5. No aceptar cambios en el plan de iteración (sólo corrección de “bugs¹⁹” de la iteración anterior, para lo que se deja un tiempo).
6. Hacer reuniones diarias permite mantener la idea clara de lo se está haciendo y de cual es el objetivo a alcanzar.

Comunicación:

Los involucrados y comprometidos deben estar al tanto de todo el proceso, ya sea por el lado ABAP como por el lado WebDynpro, manifestando todo cambio que se presente a fin de evaluar sus impactos en el desarrollo (hay que evaluar si es posible llevar un control de cambio formal).

¹⁹ Falla, error o defecto en el sistema.

5 Hallazgos

Los problemas recurrentes de cómo se verá y comportará un requerimiento de desarrollo, son resueltos tempranamente al utilizar la metodología, ya que las iteraciones permiten ver, tocar e interactuar con partes del producto final, y que son las que el cliente considera más importantes.

La mayoría de las necesidades del proyecto son cubiertas porque el cliente decide al comienzo de cada iteración qué se construirá, de acuerdo a la importancia que él asigne a los requerimientos. Terminada una iteración, el cliente decide si desea continuar mejorando lo entregado o continuar con los requerimientos del product backlog.

Existe una serie de inconvenientes y restricciones para la aplicación de metodologías ágiles, problemas que pueden presentarse con cualquier metodología, pues no existe ninguna que pueda satisfacer las necesidades del usuario en un ciento por ciento sin ningún aspecto en contra, en la metodología SCRUM, ellas son:

- Que se encuentran dirigidas a equipos pequeños o medianos.
- El entorno físico debe ser un ambiente que permita la comunicación y colaboración entre todos los miembros del equipo durante todo el tiempo.
- Cualquier resistencia del cliente o del equipo de desarrollo hacia las prácticas y principios puede llevar al fracaso (el clima de trabajo, la colaboración y la relación contractual son claves).
- El uso de tecnologías que no tengan un ciclo rápido de retroalimentación o que no soporten fácilmente el cambio

Los elementos de gestión de proyectos incorporados desde SCRUM, y los resultados obtenidos con cada uno de ellos se presenta a continuación:

Elemento Scrum	Resultados obtenidos
Sprint Planning	Es realmente importante que todos y cada uno de los afectados, por el desarrollo de los requerimientos que se tratarán en el Sprint Planning, participe en las reuniones de planificación. En Scrum quien representa al cliente es el <i>Product Owner</i> ; en este caso, el cliente tuvo algunos conflictos internos entre los usuarios y los requerimientos solicitados por el <i>Product Owner</i> , y es por esta razón se concluye a los afectados directa o indirectamente en el desarrollo, y de esa forma asegurar que el desarrollo cumplirá con las expectativas.
Product Backlog	La lista de requerimientos fue realmente importante, ya que permite al cliente y al equipo visualizar cual de los requerimientos tiene prioridad, de acuerdo con el nivel de importancia para el negocio. Además, los casos de prueba especificados junto a cada requerimiento permiten establecer criterios de aceptación y guían las pruebas del equipo de proyecto.
Sprint Backlog	El Sprint Backlog fue sin duda es uno de los elementos más importantes. Cuando se está en pleno desarrollo, entrega la información de avance de la iteración y qué tareas de los requerimientos están siendo realizados por el equipo.
Gráficos Burn	El gráfico ilustra si el equipo de desarrollo está cumpliendo

Down	con las tareas necesarias para completar los requerimientos de la iteración, apoyando a medir el esfuerzo restante. Esto permite saber si el sprint está sub-estimado o sobre-estimado, facilitando las acciones correctivas oportunamente
Daily Meeting	Las reuniones ayudaron a mantener al equipo completamente informado de lo que cada uno estaba realizando. Si algún integrante tenía un problema se establecía un plan de acción para ayudar y avanzar, lo que en ocasiones implicaba agregar tareas al sprint, de manera controlada.
Retrospective	El cliente obtiene información objetiva del progreso del sistema. Al ver y probar el incremento, el cliente y el equipo en general obtienen feedback clave para evolucionar y dar más valor al product backlog. El equipo obtiene retro-información sobre buenas prácticas y problemas durante el sprint, elemento esencial para aplicar una mejora continua.

6 Conclusión

Tras haber iniciado este trabajo de Memoria de Título, bajo el postulado de la factibilidad de llevar a cabo un desarrollo de software en una implementación SAP con Scrum, podemos concluir que la utilización de dicha metodología demostró ser efectiva para trabajos cuyos proyectos cambian como consecuencia de nuevos requerimientos, los que se presentan conforme avanza el desarrollo del mismo, haciendo hincapié eso sí, en la necesidad de mantener una luz de alerta respecto al alcance del mismo, debido a que en dichos proyectos se suele vender funcionalidades acotadas a los requerimientos del cliente

Este trabajo permitió explicar y concluir además, que es posible gestionar exitosamente los objetivos de un proyecto de desarrollo de software en una implementación SAP ERP, utilizando una metodología Scrum.

Los entregables producidos por las tres primeras iteraciones, cumplían con las expectativas del cliente dentro de los plazos y con la calidad esperada.

También es preciso destacar, que como resultado del examen de los antecedentes utilizados para la consecución de este trabajo, es posible afirmar que no existe una metodología universal para hacer frente con éxito a todo proyecto de desarrollo de software, ya que cualquier metodología, por muy completa que ésta sea, deber ser adaptada al contexto del proyecto, considerando siempre los recursos técnicos, humanos, tiempos de desarrollo, tipos de sistemas- por mencionar algunos-.

A pesar de ciertas falencias que se presenta en la aplicación de metodologías ágiles- las que se detallan en los hallazgos- puede reconocerse que es una de las pocas metodologías que permite abordar la mayor cantidad de situaciones existentes en proyectos pequeños y con requerimientos muy cambiantes.

7 Glosario

.NET

La plataforma .NET de Microsoft es un componente de software que puede ser añadido al sistema operativo Windows. Provee un extenso conjunto de soluciones predefinidas para necesidades generales de la programación de aplicaciones, y administra la ejecución de los programas escritos específicamente con la plataforma.

API (Application Programming Interface)

Especificación de una librería o utilidad que documenta su interfaz y permite su uso sin conocimiento de su interior.

ABAP (Advanced Business Application Programming)

Es un lenguaje de cuarta generación, propiedad de SAP, que se utiliza para programar la mayoría de sus productos (R/3, mySAP Business suite, Etc.). Utiliza sentencias de Open SQL para conectarse con prácticamente cualquier base de datos. Cuenta con miles de funciones para el manejo de archivos, bases de datos, fechas, etc. Permite conexiones RFC (Remote Function Calls) para conectar a los sistemas SAP con cualquier otro sistema o lenguaje de programación.

Artefactos

Un artefacto es un producto tangible resultante del proceso de desarrollo de software. Un artefacto puede ser la documentación generada, Diagramas, códigos de programas, programas ejecutables, Etc.

AS IS

Sección del documento formal elaborado en la fase de Business Blueprint que detalla la situación actual de la organización.

ASAP

Es un acrónimo para la metodología AcceleratedSAP, desarrollada por SAP AG para la implementación de proyectos de sus soluciones de software.

BBP (Business Blueprint)

Corresponde a la segunda fase de la metodología ASAP, y su propósito es entender las metas del cliente y determinar los procesos de negocio necesarios para cumplir las mismas.

Bugs

Falla, error o defecto de un software.

Grafico burn-down

Herramienta de seguimiento para el equipo, que muestra el avance del Sprint día a día y revela de forma temprana posibles desviaciones. Es un grafico cartesiano que representa en el eje x los días laborables del Sprint, y en el y la cantidad de esfuerzo estimada.

Desarrollo en cascada

Se trata de un modelo lineal, posible en aquellos proyectos donde los requerimientos están clara y altamente definidos. También se conoce como ciclo de vida clásico. Las actividades siguen un flujo en secuencia.

Empowerment

Es el hecho de delegar poder y autoridad a los subordinados y de conferirles el sentimiento de que son dueños de su propio trabajo.

Entregable

Es cualquier resultado tangible que se produce. Éstos pueden ser documentos, planes, aplicaciones de software, edificios, aeronaves, etc.

ERP (Enterprise Resource Planning)

Sistemas de gestión de información que integran y automatizan muchas de las prácticas de negocio asociadas con los aspectos operativos o productivos de una empresa.

Feedback

Retroalimentación, conjunto de reacciones o respuestas que manifiesta un receptor respecto a la actuación del emisor, lo que es tenido en cuenta por este para cambiar o modificar su mensaje.

GUI (Graphical User Interface)

Es un tipo de interfaz de usuario que utiliza un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Habitualmente las acciones se realizan mediante manipulación directa para facilitar la interacción del usuario con la computadora.

J2EE (Java 2 Enterprise Edition)

Define un estándar para el desarrollo de aplicaciones empresariales multicapas diseñado por Sun Microsystems. J2EE simplifica las aplicaciones empresariales basándolas en componentes modulares y estandarizados, proveyendo un completo conjunto de servicios a estos componentes, y manejando muchos de las funciones de la aplicación de forma automática, sin necesidad de una programación compleja.

Java

Plataforma para el desarrollo de software creada por Sun Microsystems, ampliamente extendida hoy en día, que otorga independencia de plataforma al software creado en ella y lo provee de una gran cantidad de APIs estandarizados.

JCo (Java Connector)

Es un componente middleware que permite el desarrollo en Java de componentes y aplicaciones compatibles con SAP. SAP JCo soporta la comunicación con un servidor SAP en ambas direcciones: llamadas de entrada, en las que un cliente Java externo se comunica con la BAPI (Business Application Programming Interface) o con los RFM (Remote Function Modules) de SAP; y llamadas de salida, donde desde ABAP (Advanced Business Application Programming) se establece una comunicación con un servidor Java externo.

Juicio Experto

Se denomina juicio experto a la opinión de una persona con grandes conocimientos en un ámbito determinado. La opinión del experto es considerada para realizar generalmente estimaciones, consideraciones, etc.

Manifiesto Ágil

El manifiesto ágil es un documento que resume en cuatro valores y doce principios las mejores prácticas para el desarrollo de software, basados en la experiencia de 17 industriales del software, en procura de desarrollos más rápidos conservando su calidad.

Valores Manifiesto Ágil:

- Los individuos y sus interacciones por encima de procesos y las herramientas.
- Software funcionando por encima de documentación comprensiva.
- Colaboración con el cliente por encima de negociación de contratos.
- Responder al cambio por encima de seguir un plan.

Middleware

Software de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas.

MVC (Model-View-Controller)

Patrón arquitectónico desarrollado para interfaces gráficas que resalta la importancia de una separación clara entre la presentación de datos y la lógica de negocio de una aplicación.

Open Source

Calificación de software que cumple una serie de requisitos, principalmente aquel que permite una libre redistribución, distribuye el código fuente, y permite modificaciones y trabajos derivados.

PMBOOK (Project Management Body of Knowledge)

Estándar en la gestión de proyectos desarrollado por el Project Management Institute.

PMI (Project Management Institute)

Es una asociación profesional sin fines de lucro en gestión de proyectos

Proceso

Conjunto de actividades mutuamente relacionadas o que interactúan, las cuales transforman elementos de entrada en resultados (ISO 9000:2000).

Product Backlog

Lista de requisitos priorizada que representa las expectativas del cliente respecto a los objetivos y entregas del producto o proyecto.

Product Owner

Persona responsable de lograr el mayor valor de producto para los clientes, usuarios y resto de implicados. Sus funciones son:

- Define las funcionalidades del producto
- Decide sobre las fechas y contenidos de los releases
- Es responsable por la rentabilidad del producto (ROI)
- Prioriza funcionalidades de acuerdo al valor del mercado/negocio
- Ajusta funcionalidades y prioridades en cada iteración si es necesario

Ratio

Relación o proporción entre dos magnitudes.

RoadMap ASAP

Proporciona las indicaciones paso a paso a través de la implantación de SAP ERP R/3.

SAP (Systeme, Anwendungen und Produkte)

(Sistemas, Aplicaciones y Productos)

SAP "GOLD" Channel Partner

La categoría más alta que otorga SAP entre sus Partners de Canal.

SAP AG

Compañía alemana productora de Software, que opera en todo el mundo, con 28 sucursales afiliadas y 6 compañías asociadas.

Sap Business Suite

Conjunto de soluciones que permiten mantener y optimizar las operaciones de una organización, al mismo tiempo ayudan a maximizar la rentabilidad de las relaciones comerciales, integrando personas, información y procesos.

Sap NetWeaver Developer Studio

Ambiente de desarrollo SAP basado en eclipse.

Sap Script

Es un lenguaje que se utiliza en SAP para crear formularios, Listados Simples, etc.

Sap Solution Manager

Herramienta básica para gestionar la administración de sistemas SAP e incluye:

- Monitorización de sistemas y procesos de negocio claves del cliente.
- Support Desk (Gestión de incidencias del cliente con workflow integrado) , Implementación automática de notas OSS , Soporte Remoto , integración del marketplace de servicio de SAP.
- Mejores prácticas de gestión de sistemas SAP
- Etc.

SAP WAS (SAP Web Application Server)

Plataforma para el desarrollo eficiente aplicaciones Web SAP que soporta distintos estándares: ABAP, J2EE y .NET, Etc.

SCRUM

Proceso de desarrollo de software basado en un ciclo de vida iterativo e incremental utilizado comúnmente en entornos ágiles de desarrollo de software.

Scrum Master

- Es la persona responsable de lograr el mayor valor de producto para los clientes, usuarios y resto de implicados.
- Responsable de promover los valores y prácticas de Scrum.
- Se asegura de que el equipo es completamente funcional y productivo.
- Permite la estrecha cooperación en todos los roles y funciones. Escudo del equipo de interferencias externas.

Smart Forms

Herramienta para la construcción de formularios y listados dentro SAP.

Sprint

Periodo de trabajo para desarrollar un incremento del producto.

Sprint Backlog

Lista de los trabajos que debe realizar el equipo durante el Sprint para generar el incremento previsto.

TO BE

Respuesta de los consultores a las necesidades descritas en el AS IS de un documento BBP, donde se detalla cómo serán realizados los requerimientos en el sistema.

WebDynpro

Ambiente de desarrollo de SAP para crear interfaces de usuario profesionales para aplicaciones de negocio.

8 Bibliografía

Internet

- www.controlchaos.com
- www.navegapolis.net/
- www.sap.com
- www.scrummanager.net
- www.sei.cmu.edu/CMMI/cmml.html
- www.xp123.com

Libros

- Cockburn, A. Agile Software Development. Addison-Wesley. 2001.
- Palacio Juan, Scrum Manager, Gestión de proyectos, 2008
- Poppendieck M., Poppendieck T. Lean Software Development: An Agile Toolkit for Software Development Managers. Addison Wesley. 2003.
- Schwaber K., Beedle M., Martin R.C. Agile Software Development with SCRUM. Prentice Hall. 2001.
- Schwaber K., Beedle M., Martin R.C. Agile Software Development with SCRUM. Prentice Hall. 2001.
- Sutherland Jeff, Cohn Mike, Scrum y XP desde las trincheras, C4Media Inc. 200 .