

**UNIVERSIDAD GABRIELA MISTRAL
FACULTAD DE INGENIERIA**

CALIDAD ÁGIL

Memoria para optar al título de Ingeniero de Ejecución en Informática

Autor : José Ignacio Elzo Merino
Profesor Guía : Roberto Carú Cisternas
Profesor Integrante : Jorge Tapia Castillo

Santiago – Chile
Noviembre, 2017

1 ÍNDICE

1	ÍNDICE.....	2
2	INTRODUCCIÓN.....	4
3	MARCO TEÓRICO.....	7
	3.1 El problema.....	7
	3.2 Objetivos generales.....	8
	3.3 Motivación.....	9
4	DESARROLLO.....	10
	4.1 Calidad.....	11
	4.1.1 Parámetros de la calidad informática:.....	12
	4.1.2 Sistema y calidad.....	12
	4.1.3 Aseguramiento de calidad.....	14
	4.2 Metodología ágil.....	15
	4.2.1 Principios de las metodologías ágiles:.....	19
	4.2.2 La filosofía de las metodologías ágiles.....	19
	4.2.3 Algunas de las metodologías ágiles.....	20
	4.2.4 Herramientas más utilizadas en la gestión ágil.....	38
5	Requisitos para implementar QA.....	44
	5.1 Requisitos Generales.....	44
	5.2 Responsabilidades, autoridad y comunicación.....	47
	5.3 Gestión de los recursos.....	48
	5.4 Procesos en la ejecución de los proyectos.....	50
	5.5 Seguimientos, medición y propósitos.....	50
6	CONCLUSIONES.....	52
7	GLOSARIO.....	57
	Algoritmo.....	57
	Auditoría.....	58
	Bugs.....	59
	Check list.....	60
	Ciclos de proyecto.....	61
	CMMI.....	62
	Control de calidad.....	64

<i>Diagrama de flujo</i>	65
<i>Encriptación</i>	66
<i>Estimar</i>	67
<i>Freemium</i>	68
<i>Gestión de proyectos</i>	69
<i>Help Desk</i>	71
<i>ISO 9000</i>	72
<i>IT 73</i>	
<i>Línea base</i>	74
<i>Mejora continua</i>	75
<i>Metodología cascada</i>	76
<i>Ofuscación</i>	77
<i>Open Source</i>	78
<i>Procesos</i>	80
<i>Product Owner</i>	81
<i>Requerimientos:</i>	82
<i>Balanced Scorecard</i>	84
<i>Scrum Master</i>	85
<i>Stakeholders</i>	87
<i>Sprint de entregables:</i>	88
<i>BIBLIOGRAFIA</i>	89

2 INTRODUCCIÓN

Debido a los grandes descubrimientos que se han hecho a través del mundo, el progreso de la tecnología es cada vez más rápido, las necesidades aumentan en todos lados, especialmente en la ciencia de la informática.

Las tecnologías se mueven a gran velocidad y los desarrolladores por su parte, elaboran software usando nuevas tecnologías y metodologías de desarrollo, tratando de adaptarse a los plazos en que se estiman los requerimientos.

Dado que el mercado es muy competitivo, el software es desarrollado en el menor tiempo posible para cumplir con los plazos solicitados por los clientes, pero se deja la calidad como última prioridad.

Conociendo la importancia del software, el mercado y cuan exigente es el cliente, la necesidad de adaptarse a los constantes cambios de clientes, que cada vez son más exigentes, que requieren de sistemas más rápidos y de fácil navegación, en un entorno más globalizado, donde las exigencias del mercado son cada día son mayores, hacen de la calidad un factor determinante para lograr la excelencia en la gestión de las organizaciones. De la misma forma, éstas se han visto en la obligación de entregar un mejor servicio al cliente, por lo que el diseño, la implementación y el mejoramiento continuo de un sistema de calidad permite que las empresas sean más competitivas en el mercado futuro, teniendo en cuenta que en éste sólo permanecerán quienes enfrenten el cambio con calidad, capacitación y desarrollo.

Uno de los propósitos de las empresas debe estar dirigido a satisfacer las necesidades de todos sus clientes, los cuales están en todo el derecho de exigir y elegir la empresa que mejor satisfaga sus necesidades. Para esto y entre otros factores elegirán el producto que consideren les brinde una mayor confianza en el momento de su utilización, incluso sentirán más confianza con aquellas empresas que posean procesos y estándares de calidad.

En esta investigación, validaremos la hipótesis en que las metodologías ágiles gestionadas correctamente, mejoran considerablemente:

- La calidad en los sistemas
- Los plazos de entrega o deadlines.
- El manejo de prioridades
- La asignación de recursos
- La comunicación entre las áreas interesadas

También analizaremos si es pertinente institucionalizar dentro de la organización el principio de gestión por valor de negocio por sobre consideraciones de carácter técnico y es aquí donde juega un papel importante el área de aseguramiento de calidad, ya que vela por una correcta interacción entre lo técnico y el negocio.

También será necesario enfatizar en que se deben mejorar las herramientas que dan soporte al modelo de desarrollo y calidad, con la finalidad que el sistema tenga un menor costo operacional como por ejemplo un administrador de proyectos, en el cual puedan interactuar las distintas áreas interesadas.

Profundizaremos en que se debe buscará mejorar la interacción y comunicación con las áreas comerciales debido a que perciben la dificultad en los desarrollos y procesos de una forma distinta a los que realmente son.

Enfatizar en que se deben efectuar medidas de análisis y sacar lecciones aprendidas de los distintos procedimientos, de manera que, en una fase posterior, estos se encuentren lo suficientemente maduros como para gestionar de una mejor manera los proyectos.

El aumento de valor en el tiempo es un objetivo que toda compañía persigue y para lo cual es necesario madurar los procesos y procedimientos, comunicarlos a toda la compañía de manera de estar alineados con las distintas áreas. No obstante, se requieren dos condiciones para lograr el aumento de valor esperado:

- I. Que el producto generado por un proyecto efectivamente tenga valor para el cliente, los usuarios internos y externos.
- II. Que los proyectos se terminen en el tiempo en que el negocio requiere el producto o servicio.

Lo anterior es especialmente relevante en organizaciones cuyo negocio se sustenta en tecnologías, pues se ven sometidas a una dinámica muy grande, tendiente a estar en constante modernización y cambios de los servicios que prestan.

Las metodologías ágiles, son una propuesta innovadora, que pretenden entregar un nuevo enfoque para este tipo de problemas.

Estas metodologías, tienen una aceptación cada vez mayor debido a que logran hacer un uso más eficiente de los recursos además de direccionar el desarrollo por el valor que tiene para el cliente.

Un principio importante en las propuestas ágiles, es que los requerimientos se descubren progresivamente durante el desarrollo del proyecto y por lo tanto es mejor ocuparse primero de aquellos de mayor valor en acuerdo con la visión del cliente.

La técnica empleada por los métodos ágiles consiste en considerar que el proyecto evolucionará en avances progresivos y cada avance o sprint de entregables deberá ocurrir en un plazo y con recursos acotados, de manera de obligar a priorizar los requerimientos de acuerdo al valor que aportan al negocio.

Este cambio de paradigma provocará que las organizaciones que usan metodologías ágiles tiendan a optimizar el proceso de priorización de requerimientos, anticiparán la puesta en funcionamiento de soluciones que generen valor (ingresos u otros beneficios) y otorgarán flexibilidad a la compañía para reasignar recursos a proyectos de mayor impacto de acuerdo a los objetivos estratégicos establecidos.

3 MARCO TEÓRICO

3.1 *El problema*

La falta de compromiso de los empleados para hacer sus actividades con calidad, la dificultad para detectar las verdaderas necesidades de los clientes, la falta de infraestructura y la escasez de recursos financieros para lograr la conformidad con los requisitos del producto, son problemas muy frecuentes en cualquier compañía del área de tecnología.

Esto, se debe al poco interés de las empresas en materia de calidad en todos sus procesos, y la falta de indicadores que obliguen a los empleados alcanzar rendimientos y metas que los obliguen a mejorar cada vez los procedimientos que se utilizan en el trabajo diario.

De no ser controladas las situaciones que se pueden presentar, existirán serios problemas para la toma de acciones correctivas y preventivas que eviten la presencia de inconformidades, dificultando así la consecución de la satisfacción de los clientes.

De acuerdo a lo anterior, se pretende elaborar un diagnóstico y plan de implementación de un área de Quality Assurance (QA) que utilice metodologías ágiles y flexibles que se adopten mejor a los procesos de una compañía.

3.2 *Objetivos generales*

- Esta investigación busca conocer las metodologías ágiles, sus orígenes, conceptos, principios y elementos.
- Demostrar la importancia que tiene la calidad en los ciclos de desarrollos de un software desde la gestión del proyecto hasta la creación de un sistema.
- Exponer los fundamentos de diferentes metodologías ágiles y propuestas para la implementación del área de calidad (QA) en un proyecto de software.
- Analizar cuando es pertinente aplicar técnicas y herramientas de metodologías ágiles para planificar, estimar, priorizar y programar los trabajos que implican los proyectos.
- Conocer cómo implantar la gestión ágil en toda una organización desde las herramientas disponibles, hasta la estructura de los equipos y asignación de roles y recursos
- Demostrar que las metodologías ágiles son una excelente opción a la hora de gestionar proyectos ya que son flexibles y tienen la ventaja de adaptarse a distintos tipos de desarrollos y gestión de proyectos, desde los muy pequeños y sencillos hasta los más estructurados y complejos.
- Los impactos positivos en los procesos internos de una compañía luego de implantar metodologías ágiles

3.3 Motivación

En este trabajo de investigación expondremos los conceptos novedosos, respecto a la forma de encarar proyectos en el área de QA, si bien en la actualidad es frecuente escuchar hablar de metodologías ágiles, no es común encontrar organizaciones que empleen bien el concepto, generalmente las grandes organizaciones tienen plasmado el concepto de metodologías ágiles pero inconscientemente utilizan las metodologías tradicionales, donde cada proceso de desarrollo está muy marcado por documentar cada paso, pero este tipo de metodologías para proyectos pequeños y mantenciones es más demoroso y costoso de lo que debiese ser, lo que genera proyectos caros, de mala calidad y al largo plazo.

Existen diversos estudios en torno a la necesidad de incorporar buenas prácticas en los procesos de desarrollo de software. Sin embargo, la adopción de buenas prácticas no es una tarea sencilla debido a falta de tiempo, falta de conocimiento, falta de motivación, falta de recursos. Debido a esto, existen diversos estándares orientados a mejorar y certificar procesos de desarrollo de proyectos de software. Estos estándares consideran el uso de buenas prácticas enfocadas a establecer reglas y convenios, aportar calidad al código, estandarizar el desarrollo, facilitar la lectura, facilitar la escalabilidad del código, facilitar la reutilización y la integración de manera homogénea, para entregar finalmente un software de calidad.

4 DESARROLLO

En la actualidad muchas compañías se encuentran implementando modelos de procesos tendientes a dar una mayor gobernabilidad a la gestión de proyectos con la finalidad de lograr los objetivos estratégicos que se han planteado.

En el caso de las organizaciones en Chile, el modelo de proceso promedio alcanza a CMMI N3 certificado, lo cual les ha permitido contar con un proceso de desarrollo de soluciones tecnológicas que se evalúan internamente en forma muy positiva, no obstante, la satisfacción de los clientes es del orden del 40% y donde el 80% de los proyectos finaliza con atrasos superiores a un mes.

La gestión de proyectos está mucho mejor estructurada que hace un par de años, es más profesional, cuenta con métricas más precisas, y la institucionalización de los procesos es muy alta. No obstante, los retrasos aún son significativos, y en algunos casos, llegan a superar el año de espera por recursos.

El principal problema es que inicialmente se define una planificación, se declara que se debe cumplir y en la práctica los líderes de proyecto no tienen cómo garantizar la asignación de los recursos en los distintos departamentos. Esto se debe al tipo de organización adoptada y que corresponde a una estructura débil, lo que ocasiona que el poder de los líderes de proyectos sea relativamente bajo respecto a los miembros del equipo.

Las decisiones de prioridad de los proyectos son tomadas de acuerdo al nivel de influencia de las jerarquías gerenciales, quienes en base a su percepción pueden definir o alterar el orden de ejecución de los proyectos que se encuentran en marcha.

Si bien, las organizaciones cuentan con un proceso de planificación estratégica que se realiza una vez en el año, se requiere extender este ejercicio de evaluación y priorización en forma permanente de manera de asegurar que las decisiones tomadas respondan a criterios establecidos.

Sólo hasta hace muy poco tiempo se ha iniciado la práctica de gestionar a nivel operacional en los proyectos, esto se lleva a cabo en un comité Integrado por todos los actores relevantes y algunos gerentes, lo que permite disponer de un espacio de coordinación para mejorar la integración de las planificaciones y donde las decisiones son entendidas por todos los presentes.

4.1 Calidad

Calidad es un concepto subjetivo, y el término proviene del latín, está relacionada con las percepciones de cada individuo para comparar una cosa con cualquier otra de su misma especie y diversos factores, como la cultura, el producto, el servicio, las necesidades y las expectativas influyen directamente en esta definición. También se refiere a la capacidad que posee un objeto para satisfacer necesidades implícitas o explícitas en el cumplimiento de requisitos.

Una visión actual del concepto de calidad indica que calidad es entregar al cliente no lo que quiere, sino lo que nunca se había imaginado que quería y que una vez que lo obtenga, se dé cuenta que era lo que siempre había querido.

Otro ejemplo de calidad es la calidad de los datos, la falta de calidad de los datos es uno de los principales problemas a los que se enfrentan los responsables de sistemas de información y las empresas en general, pues constituye uno de los problemas "ocultos" más graves y persistentes en cualquier organización en el mundo.

En las áreas de tecnología, la calidad se logra revisando los procesos formales de reingeniería. Existen tres elementos que se deben de tomar en cuenta para obtener calidad informática:

- I. El plan estratégico y procedimientos que se deben cumplir.
- II. La arquitectura de procesos.
- III. Las tecnologías y herramientas que se utilizaran.

4.1.1 Parámetros de la calidad informática:

- **Calidad de diseño:** Es el grado en el que los productos o servicios tecnológicos (software, base de datos, hardware, comunicaciones), se ven reflejado en el diseño.
- **Calidad de conformidad del cliente:** Es el grado de fidelidad con el que es reproducido el producto o servicio tecnológico, respecto al diseño.
- **Calidad de uso:** Debe de ser fácil de usar, seguro, fiable, etc.
- **El cliente es el nuevo objetivo:** Las nuevas teorías sitúan al cliente como parte activa en la calificación de la calidad de un producto o servicio y esta no se va a determinar solamente por parámetros y objetivos establecidos por la compañía. El factor más importante en la calidad lo deben incluir los mismos clientes con opiniones que aporten a la mejora continua del producto.
- **Criterios para evaluar la calidad del software:** En la Actualidad el concepto de calidad de software, requiere de una congruencia total entre los requerimientos y características del producto para lograr plena satisfacción del usuario, surgen ahora componentes y criterios de calidad tales como: confiabilidad, soporte logístico, agilidad de respuesta, flexibilidad, facilidad de uso, integridad, consistencia y sencillez en el diseño.

4.1.2 Sistema y calidad.

Sistema es una totalidad conformada por partes o elementos interrelacionados que persiguen algún fin, es decir, un conjunto organizado de elementos que interactúan entre sí y que son interdependientes, formando un todo complejo, diferente y distinto. Sobre los elementos o partes de un sistema es importante que no sólo son componentes físicos, sino también sus funciones son el resultado de las interacciones que se dan entre los mismos.

La calidad en los sistemas, es el conjunto de propiedades y características que otorgan la capacidad de satisfacer las necesidades de los clientes, cualquier empresa que desee lograr la certificación de su sistema de gestión de calidad, debe contar con organismos existentes de certificación, con el fin de medir métricas y cumplir con los parámetros establecidos en auditorías externas.

Las auditorías son exámenes o análisis, que permiten determinar si la calidad de un sistema, servicio, producto o proceso, concuerda con la norma previamente adoptada y satisface los requisitos establecidos, así mismo es la pieza clave de un sistema de mejora continua, que permite reducir fallos y prevenir inconformidades.

Para llevar a cabo el enfoque basado en procesos en una organización, lo primero que se debe hacer es definir un mapa conceptual donde los flujos se encuentren bien definidos, de esta manera se podrán comprender los procesos para posteriores mejoras.

Un sistema de gestión de calidad tiene como eje central la mejora continua, esta debe ser integrada como una actitud permanente y que todas las personas se encuentren comprometidas con la organización para superar continuamente los resultados mediante esfuerzos progresivos.

Cuando se emprende el cambio hacia la calidad, el énfasis en el cliente es importante puesto que la satisfacción es uno de sus derechos. Los clientes insatisfechos pueden conllevar a la pérdida de ventas, es en este momento cuando se hace necesario tomar acciones correctivas y/o preventivas.

Una acción correctiva es un conjunto de acciones para eliminar la causa de inconformidades (incumplimiento de un requisito o necesidades de los clientes, problemas con los productos y/o servicios, deficiencia en el sistema o cualquier otra situación donde lo sucedido no era lo que se requería o esperaba) con objeto de prevenir que vuelva a ocurrir.

4.1.3 Aseguramiento de calidad

El aseguramiento de la calidad es el conjunto de acciones planificadas y sistemáticas necesarias para proporcionar la confianza adecuada de que un producto o servicio satisfaga los requerimientos dados sobre calidad.

El aseguramiento de la calidad nace como una evolución natural del control de calidad, que resultaba limitado y poco eficaz para prevenir la aparición de defectos. Para ello, se hizo necesario crear sistemas de calidad que incorporasen la prevención como forma de vida y que, en todos los casos, sirvieran para anticipar los errores antes de que estos se produjeran.



Imagen 1.
Conjunto de acciones necesarias para dar calidad.

Un sistema de calidad se centra en garantizar que lo que ofrece una organización cumple con las especificaciones establecidas previamente por la empresa y el cliente, asegurando una calidad continua a lo largo del tiempo.

El aseguramiento de la calidad es un sistema que pone énfasis en los productos, desde su diseño hasta el momento de su implementación al cliente, y concentra sus esfuerzos en la definición de procesos y actividades que permiten la obtención de productos conforme a determinadas especificaciones. Sus objetivos son:

- Que no puedan llegar al cliente productos o servicios defectuosos
- Evitar que los errores se produzcan de forma repetitiva.

El aseguramiento de la calidad consiste en el seguimiento de líneas de acciones planificadas y sistemáticas, implantadas dentro del sistema de gestión de calidad de la empresa. Estas acciones deben ser demostrables con el objeto de proporcionar la confianza adecuada, tanto a la propia empresa como a los clientes y proveedores.

El aseguramiento de calidad ha sido englobado en la gestión de la calidad, proporcionando un modelo que establece una serie de requisitos al propio sistema de calidad, y no a los procesos y actividades que se realizan en la empresa y organización.

4.2 Metodología ágil

Cada vez son más las empresas a nivel mundial que apuestan por las metodologías ágiles. Las metodologías tradicionales, no se adaptan a las nuevas expectativas de los usuarios y a las exigencias del mercado.

Por definición, las metodologías ágiles son aquellas que permiten adaptar la forma de trabajo a las condiciones del proyecto, consiguiendo flexibilidad e inmediatez en la respuesta para amoldar el proyecto y su desarrollo a las circunstancias específicas del entorno.

En esencia, las empresas que apuestan por estas metodologías consiguen gestionar sus proyectos de forma eficaz, reduciendo los costos e incrementando su productividad,

mejoran la satisfacción de sus clientes dado que se involucrarán y comprometerán a lo largo del proyecto.

En cada etapa del desarrollo se informará al cliente sobre los progresos del mismo, de ese modo, el cliente puede sumar su experiencia para optimizar las características del producto final, así se pueden evitar numerosos malentendidos dado que el cliente poseerá en todo momento una completa visión del estado del producto.

Asimismo, mejora la motivación e integración de todos los equipos involucrados, pero esta mejora no es casual, las metodologías ágiles permiten a todos los miembros del equipo conocer el estado del proyecto en cualquier momento. Los compromisos son negociados y aceptados por todos y las ideas de cualquiera de sus integrantes son tomadas en cuenta.

En las metodologías ágiles, se trabaja con mayor velocidad y eficiencia, realizando entregas parciales pero funcionales del producto, de ese modo, es posible entregar en el menor intervalo de tiempo posible una versión funcional del producto. Gracias a las entregas parciales (centradas en primer lugar, aquellas funcionalidades que realmente aportan valor funcional al negocio) y al aporte del cliente será posible eliminar aquellas características con menos prioridad.

Las metodologías ágiles permiten mejorar la calidad del producto, la continua interacción con los clientes, tienen como objetivo asegurar que el producto final sea exactamente lo que el cliente quiere y necesita, este enfoque permite abrazar la excelencia tecnológica y permite obtener un producto tecnológicamente superior. Por otro lado, permite alertar rápidamente errores y problemas técnicos en la etapa de planificación, desarrollo y certificación de calidad,

El equipo tiene la facultad de realizar reuniones diarias, informando el status en que se encuentra el proyecto, así como también la oportuna alerta de problemas que se puedan presentar, de esta forma se pueden agilizar los procesos entre los equipos.

Con las metodologías tradicionales, los errores no identificados en las primeras fases del proyecto suelen tener costos muy altos, los plazos de resolución de errores y fallas técnicas, generalmente son muy burocráticas y los procesos de documentación son una limitante para dar respuesta y soluciones rápidas.

Durante los últimos años cada vez es más habitual el uso de metodologías ágiles para la gestión de proyectos.

La certificación en alguna de estas metodologías de gestión resulta un valor añadido tanto a nivel individual como a nivel de empresa, y se ha convertido en un criterio de decisión a la hora de trabajar con determinados clientes.

- ¿Por qué son necesarias las metodologías de gestión de proyectos?
- ¿Qué beneficios puede aportar trabajar en base a sus recomendaciones?



Imagen 2.

Esta imagen responde a las preguntas anteriores.

El objetivo general en la gestión de proyectos es ser capaces de estandarizar, estructurar y organizar la manera de trabajar. Así se ayuda a enfocar todos los proyectos de la misma forma y nos permite repetir los éxitos y aprender de los errores. Por lo que una metodología ágil se ajusta perfectamente a estas prioridades y es una

gran herramienta para generar eficiencia en la gestión de proyectos. El uso de una metodología en la gestión de un proyecto persigue los siguientes beneficios específicos:

- Organizar los tiempos del proyecto.
- Proporcionar herramientas para estimar de forma correcta tiempos y costos.
- Ayudar a gestionar y minimizar los riesgos del proyecto.
- Mejorar la relación entre costo y beneficio de los recursos.
- Desarrollar las habilidades del equipo.

Desde el punto de vista de los recursos, contar con una metodología ayuda a acortar la curva de aprendizaje del equipo y a medida que se utiliza en proyectos, se adapta y se transforma al estilo personal de la compañía.

Un factor decisivo al aplicar metodologías de gestión es que no todas sirven para cualquier proyecto, por lo que es fundamental conocer los puntos fuertes de cada una de ellas para saber aplicarlos en el momento que sea necesario.

Metodologías tradicionales	Metodologías ágiles
Proyectos de duración media o elevada	Proyectos de corta duración
Proceso mucho más controlado, con numerosas políticas y normas	Proceso menos controlado, con más flexibilidad
Respuesta lenta a los cambios	Respuesta rápida a los cambios
Gestión de equipos grandes de personas, generalmente distribuidas	Gestión de equipos pequeños (menos de diez personas)
El cliente interactúa con el equipo de proyecto mediante reuniones	El cliente suele ser parte del equipo de proyecto
Curva de aprendizaje media o larga	Curva de aprendizaje corta

Imagen 3.

Tabla de comparación entre metodologías ágiles y tradicionales.

4.2.1 Principios de las metodologías ágiles:

- La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de trabajo que aporten valor a la compañía (Sprint).
- Todas las áreas de trabajo interesadas al proyecto deben trabajar juntas a lo largo del proyecto.
- Construir el proyecto en torno a personas motivadas y confiar en sus aptitudes.
- Considerar las opiniones de las personas en pro del proyecto.
- El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo.
- El software con calidad certificada, es la medida principal del progreso.
- La atención continua a la calidad y al buen diseño, mejora la agilidad de los proyectos.
- La simplicidad es esencial.
- Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados.

4.2.2 La filosofía de las metodologías ágiles

La filosofía de las metodologías ágiles valoran al individuo y las interacciones de los equipos del proyecto, sobre el proceso y las herramientas, la gente es el principal factor de éxito de un proyecto de software, es más importante construir un buen equipo que construir el entorno.

Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente, siempre va a ser mejor crear el equipo y que éste genere su propio entorno de desarrollo en base a sus necesidades.

La colaboración con el cliente es más importante que la negociación de un contrato, se debe proponer que exista una interacción constante entre el cliente y los equipos del proyecto. Esta colaboración entre ambos será la que marque la marcha del proyecto y la que asegure su éxito, ya que se podrán detectar las prioridades y características que se requieren.

Responder a los cambios más que seguir estrictamente un plan, (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo, por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

4.2.3 Algunas de las metodologías ágiles

4.2.3.1 Scrum

Scrum fue desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle y define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Fue especialmente creada para proyectos con un rápido cambio de requisitos y sus principales características se pueden resumir en dos:

- El desarrollo de un proyecto se realiza mediante iteraciones, denominadas sprint, con una duración de 30 días, donde el resultado de cada sprint es un incremento ejecutable que se muestra al cliente.
- La segunda característica importante son las reuniones a lo largo proyecto, entre ellas destaca la reunión diaria de 15 minutos para coordinar e integrar los avances realizados por el equipo del proyecto.



Imagen 4

Se grafica lo que se debe realizar en una reunión de Scrum.

Scrum es un proceso de desarrollo de software iterativo y creciente utilizado comúnmente en entornos basados en el desarrollo ágil de software.

Aunque Scrum estaba enfocado a la gestión de procesos de desarrollo de software, puede ser utilizado en equipos de mantenimiento de software, o en una aproximación de gestión de programas.

Scrum también se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita como por ejemplo cuando:

- Las entregas se alargan demasiado
- Los costos se disparan
- La calidad no es aceptable
- Se necesita capacidad de reacción ante la competencia
- La moral de los equipos es baja y la rotación alta
- Cuando es necesario identificar y solucionar ineficiencias
- Se quiere trabajar utilizando un proceso especializado en el desarrollo de un producto.

En Scrum un proyecto se ejecuta en bloques temporales cortos y fijos (iteraciones que normalmente son de 2 semanas, aunque en algunos equipos son de 3 y hasta 4 semanas, límite máximo de feedback y reflexión). Cada iteración tiene que proporcionar un resultado completo, un incremento del producto final que sea susceptible de ser entregado con el mínimo esfuerzo al cliente cuando lo solicite.

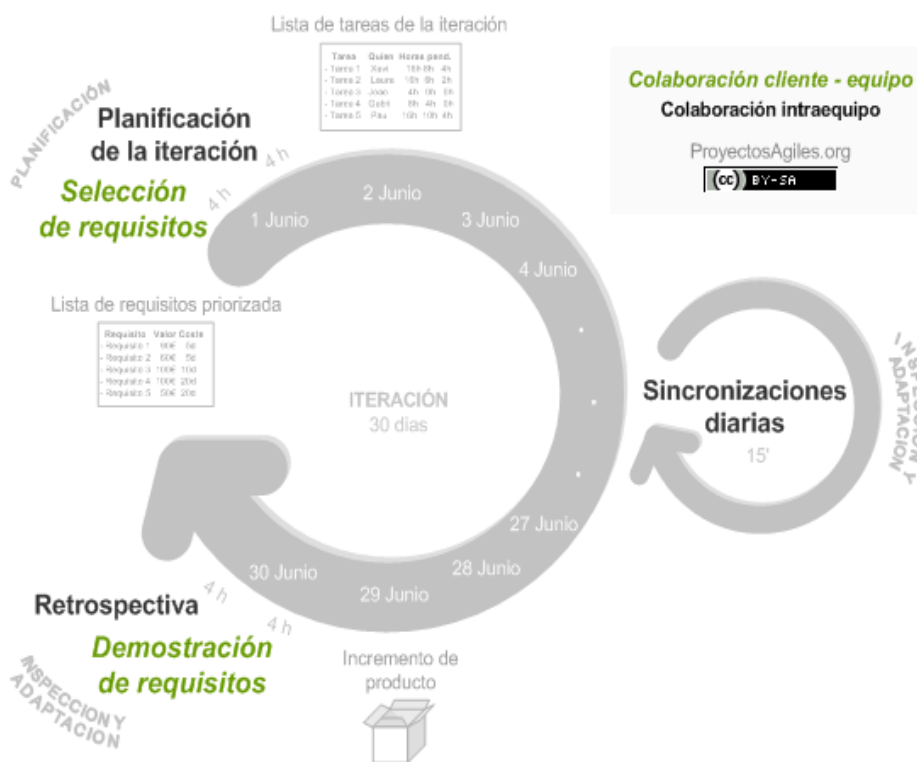


Imagen 5.
Procesos que se deben realizar en las iteraciones.

El proceso parte de la lista de objetivos/requerimientos priorizados, esta lista es considerada como plan de proyecto, donde el cliente prioriza y define los objetivos, balanceando el valor que le aportan respecto a su costo/beneficio y donde los objetivos

son considerados como requisitos para posteriormente dividirlos en iteraciones y entregas.

El primer día de la iteración se realiza la reunión de planificación de la iteración, la que se compone de dos partes:

- I. El cliente presenta al equipo la lista de requerimientos y objetivos priorizados, el equipo pregunta al cliente las dudas que surgen dentro de la reunión para luego seleccionar los requerimientos más prioritarios, los cuales se deben completar en la iteración, de manera que puedan ser entregados si el cliente lo solicita.
- II. El equipo elabora las iteraciones necesarias para desarrollar los requerimientos a los que se han comprometido. La estimación de esfuerzo se hace de manera conjunta y los jefes de proyecto asignan las tareas.

Cada día el equipo realiza una reunión de sincronización (15 minutos máximo), cada miembro del equipo inspecciona el trabajo que el resto está realizando (dependencias entre tareas, progresos hacia el objetivo de la iteración y los obstáculos que puedan impedir este objetivo) para así realizar las adaptaciones necesarias que permitan cumplir con el compromiso adquirido. En la reunión cada miembro del equipo responde a tres preguntas:

- I. ¿Qué he hecho desde la última reunión de sincronización?
- II. ¿Qué voy a hacer a partir de este momento?
- III. ¿Qué impedimentos tengo o voy a tener?

Durante la fase de iteraciones, el facilitador (Scrum Master) se encarga de que el equipo pueda cumplir con sus compromisos y de que no se pierda la productividad.

- Elimina los obstáculos que el equipo no puede resolver por sí mismo.
- Protege al equipo de interrupciones externas que puedan afectar su compromiso o su productividad.

En esta fase de iteraciones, el cliente junto con el equipo modifican o re planifican los requerimientos (solo si es necesario) con el fin de maximizar la utilidad de lo que se desarrolla.

El último día de la iteración se realiza la reunión de revisiones, esta tiene dos partes:

- I. Demostración (4 horas máximo). El equipo presenta al cliente los requerimientos completados, los resultados y los cambios que hubo en el contexto del proyecto.
- II. Retrospectiva (4 horas máximo). El equipo analiza cómo ha sido la manera de trabajar y cuáles fueron los problemas que a futuro podrían impedirle progresar y así ir mejorando de manera continua la productividad.

4.2.3.2 Kanban

Productividad, eficiencia, optimización, son palabras que suenan a menudo en la mayoría de empresas, objetivos ideales que no siempre se consiguen por no tener un buen sistema de organización o por no aplicarlo correctamente.

En este sentido, existen sistemas de organización enfocados a trabajar en equipo, uno de los puntos que cada vez tienen más importancia, pero del que se sabe todavía muy poco es Kanban, un sistema de trabajo que proporciona un mejor flujo de trabajo al dividir un proceso productivo en varias fases perfectamente delimitadas.

Kanban es una palabra japonesa que significa cartel o panel, elemento clave de este método productivo.

El sistema Kanban surgió en Toyota (fabricante japonés de automóviles) los que se vieron en la necesidad de desarrollar una metodología para organizar mejor la producción de vehículos, los procesos fueron divididos en fases bien delimitadas las que se tenían que completar correctamente para pasar a las siguientes fases, garantizando así un producto de calidad.

De este sistema aplicado a la industria de la automotriz, surgió el método Kanban, ideado por David J. Anderson, que adapta la filosofía original al desarrollo de software, un proceso con muchos puntos en común con la industria automotriz.

El método Kanban en su versión moderna aplicada al software se usó por primera vez en Microsoft, y desde entonces ha sido aplicado en cientos de proyectos de todo el mundo.

Cuatro son las reglas o principios básicos de Kanban:

- I. **Kanban es un método de producción:** No es un sistema que dice cómo hacer tu trabajo. Los equipos saben cómo hacerlo y Kanban ayudará a decidir si lo están haciendo bien o si hay que cambiar algo.
- II. **Acepta el cambio:** Siguiendo el método Kanban, todos los miembros del equipo tienen que estar dispuestos a aplicar cambios constantes para mejorar sus rutinas de trabajo.
- III. **Respetar el proceso en curso, los roles y responsabilidades de cada uno:** En un proyecto de desarrollo de software en el cual un proceso productivo es imprescindible, cada miembro del equipo debe saber qué tiene que hacer y cuáles son sus funciones. Para que el método Kanban funcione esto tiene que estar definido, no se trata de que todos hagan de todo, sino que cada persona sepa qué hacer en el momento adecuado.
- IV. **Liderazgo en todos los niveles:** Tener iniciativa y gestionar correctamente las tareas o al equipo y que cada miembro tenga clara su función y la ejecute correctamente.

Cinco son los elementos que deben darse en un proyecto para que se aplique bien el método Kanban.

- I. **Visualizar el flujo de trabajo:** Parece algo básico, pero no siempre vemos realmente las fases por las que pasa un proyecto ni qué personas trabajan en qué. Esto es muy habitual en grandes empresas, donde el desconocimiento entre el trabajo de otros equipos se ve día a día. El método Kanban recomienda usar un panel con tarjetas (que dan nombre al método) y que definan cada tarea dividiéndola en columnas que indican cada fase del proyecto.
- II. **Limitar el trabajo en curso:** Hacer muchas cosas, pero dejarlas todas a medias no sirve de nada, si se empieza algo debe ser terminado antes de empezar otra tarea, éste es un principio básico del método Kanban y uno de los pilares para que un proyecto funcione.
- III. **Gestión del flujo:** Además de visualizar el flujo de trabajo hay que controlar su funcionamiento, ver en todo momento si las tareas se están ejecutando correctamente o si alguien tiene problemas para solucionarlos.
- IV. **Dejar claras las reglas del proceso:** Para aplicar bien un método hay que entenderlo y comunicarlo con las personas del proyecto, también es importante controlar que las reglas y procesos se estén cumpliendo según lo definido y acordado.
- V. **Mejora en equipo:** Uno de los pilares del método Kanban es la mejora constante. En este sentido, la mejora debe ser acordada en equipo, aportando la experiencia de todos.

Aunque Kanban está pensado para proyectos de software, al ser un método tan simple y con reglas tan sencillas, puede aplicarse a prácticamente a cualquier entorno de trabajo. Evidentemente, un proyecto más mecánico, con reglas muy definidas, admite mejor el método Kanban, pero la clave está en la buena aplicación del método, delimitando bien cada fase, asignando bien los recursos, definiendo bien los roles de

cada miembro del equipo y generar la instancia para la revisión del sistema, aplicando mejoras cuando sean necesarias.

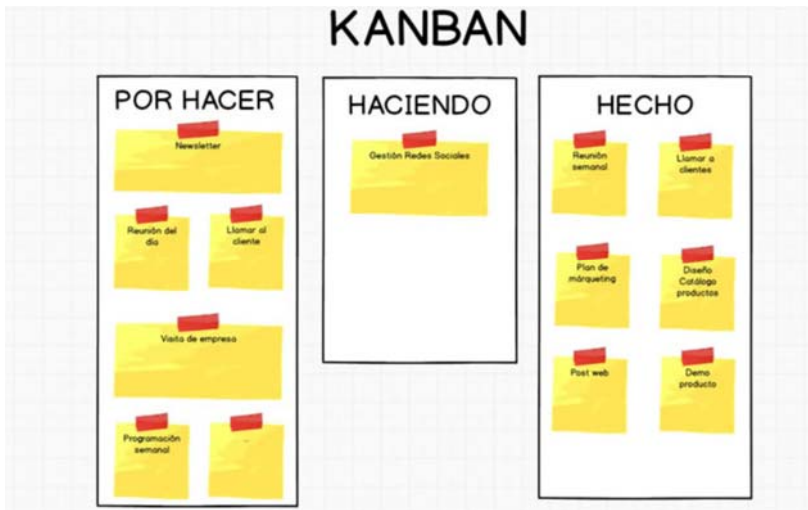


Imagen 6.
Ejemplo de tablero Kanban.

4.2.3.3 Scrumban

La metodología Scrumban nace de la combinación de principios de los métodos ágiles de gestión de proyectos más importantes en la actualidad: Scrum y Kanban. Aunque en principio pueden parecer iguales, las dos estrategias de gestión presentan diferencias en la manera de ejecutar el proyecto.

Es por eso que el novedoso plan Scrumban se encarga de combinar aquellos elementos que resultan complementarios, por ejemplo, una de las combinaciones más usadas en el plano empresarial, es la de gestionar las tareas previstas con el método Scrum y planificar los errores con el método Kanban. Sin embargo, la mezcla de una y otra implica una nueva manera de gestionar los proyectos.

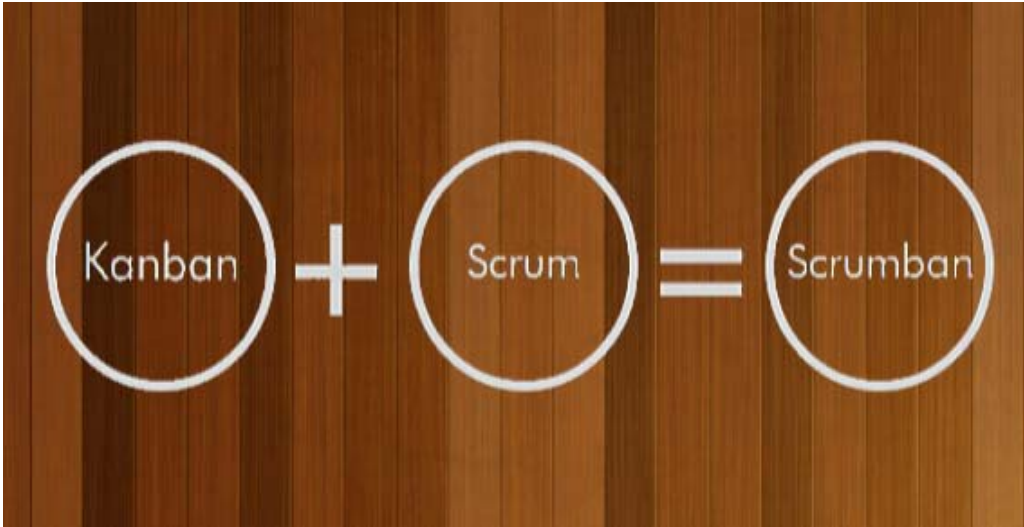


Imagen 7

La Scrumban, el resultado de Kanban + Scrum

Al margen de lo que puede ganar o perder una empresa al aplicar este método, la premisa inicial de Scrumban es emplear lo mejor de Scrum y Kanban. Es decir, aquello que al combinarlo sirva para mejorar la productividad de un plan de proyecto.

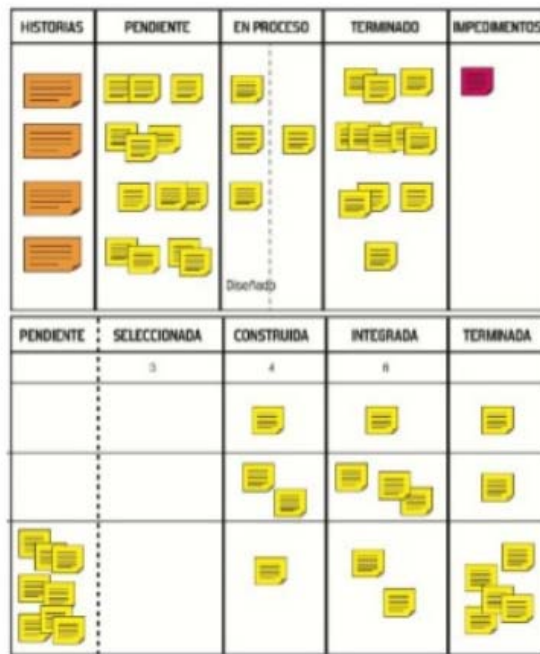
En el terreno de la ejecución, el objetivo de Scrumban es llevar a cabo una secuencia de mejoras evolutivas a través de prácticas eficientes. El flujo de trabajo sigue siendo el mismo de Kanban (etapas relacionadas entre sí), aunque con la inclusión de algunos elementos de Scrum como las reuniones diarias de 15 minutos entre el grupo de trabajo y el gestor o los análisis retrospectivos para incorporar mejoras al proceso. Por ello, existen ciertos proyectos que se adecúan mejor al método mixto. En general, se trata de aquellos con un mayor nivel de complejidad, algunos ejemplos son:

- **Proyectos de mantenimiento:** Aquellos en los que resulta indispensable la presentación de resultados de forma parcial para seguir avanzando.
- **Proyectos en los que los requisitos varíen con frecuencia:** Aquellos en los que el cliente no tiene fijadas las condiciones y expectativas del proyecto y éstas se van introduciendo a lo largo de las distintas etapas.
- **Proyectos en los que surjan errores de ejecución:** Aquellos en los que se deba replantear el método usado y analizar retrospectivamente la evolución de las tareas.

En cuanto al plan de etapas, el mapa o interfaz de la aplicación (en caso de que sea informática) ya no se limita a nombrar las tareas con los rótulos «sin comenzar», «en progreso» o «finalizadas» de Kanban, sino que, tras la revisión de Scrum, añade otras categorías, como por ejemplo «probadas» o «entregadas». Incluso, en algunos casos se recomienda dar una descripción del contexto en el que se han realizado dichas tareas, algo muy propio del método Scrum.

Scrum + Kanban = Scrumban

Organizar la
incertidumbre



+ Predictivo

+ Reactivo

Imagen 8
Ejemplo de tablero Scrumban.

4.2.3.4 XP Programación extrema

La programación extrema XP es posiblemente el método ágil más conocido y ampliamente utilizado, debido a que el enfoque fue desarrollado utilizando las mejores prácticas del desarrollo iterativo y con la participación extrema del cliente.

En la metodología extrema, todos los requerimientos se expresan como escenarios (llamados historias de usuario), los cuales se implementan directamente como una serie de tareas. Los programadores trabajan en parejas y desarrollan pruebas para cada tarea antes de escribir el código, todas las pruebas se deben ejecutar satisfactoriamente cuando el código nuevo se integra al sistema.

El desarrollo incremental se lleva a través de entregas pequeñas y frecuentes del sistema y por medio de un enfoque que sirve para la descripción de requerimientos basado en las historias del cliente o escenarios que pueden ser la base para el proceso de planificación.

La participación del cliente se lleva a cabo a través del compromiso y del tiempo completo del cliente en el equipo de desarrollo. Los colaboradores directos de los clientes participan en el desarrollo y son los responsables de definir las pruebas necesarias que servirán para la aceptación del sistema. El interés de las personas, en vez de los procesos, se lleva a través de la programación en parejas, la propiedad colectiva del código y un proceso de desarrollo sostenible que no implique excesivas jornadas de trabajo. El cambio se lleva a cabo a través de las entregas regulares del sistema, un desarrollo previamente probado y la integración continua.

El mantenimiento se lleva a cabo a través de una recta actualización constante para mejorar la calidad del código y la utilización de diseños sencillos que no prevén cambios futuros en el sistema.

En XP, los clientes están implicados en la especificación y establecimiento de prioridades de los requerimientos del sistema. Dichos requerimientos no se especifican como una lista de funciones requeridas. Más bien, los clientes son parte fundamental

del equipo de desarrollo esto permite que discutan escenarios con todos los miembros del equipo. Desarrollar conjuntamente tarjetas de historia (story card) que recogen las necesidades del cliente. Por ende, el equipo de desarrollo intentará implementar esos escenarios en una entrega futura del software.

La metodología XP ha descartado este principio partiendo del hecho de que diseñar para el cambio es a menudo un esfuerzo inútil. Con frecuencia los cambios previstos nunca se materializan y realmente se efectúan peticiones de cambios completamente diferentes. La metodología extrema aborda este problema sugiriendo que se debe revisar constantemente el software, ya que el equipo de programación busca posibles mejoras y las implementa de forma inmediata así lo que se busca es que siempre sea fácil de entender y cambiar cuando se requiera.

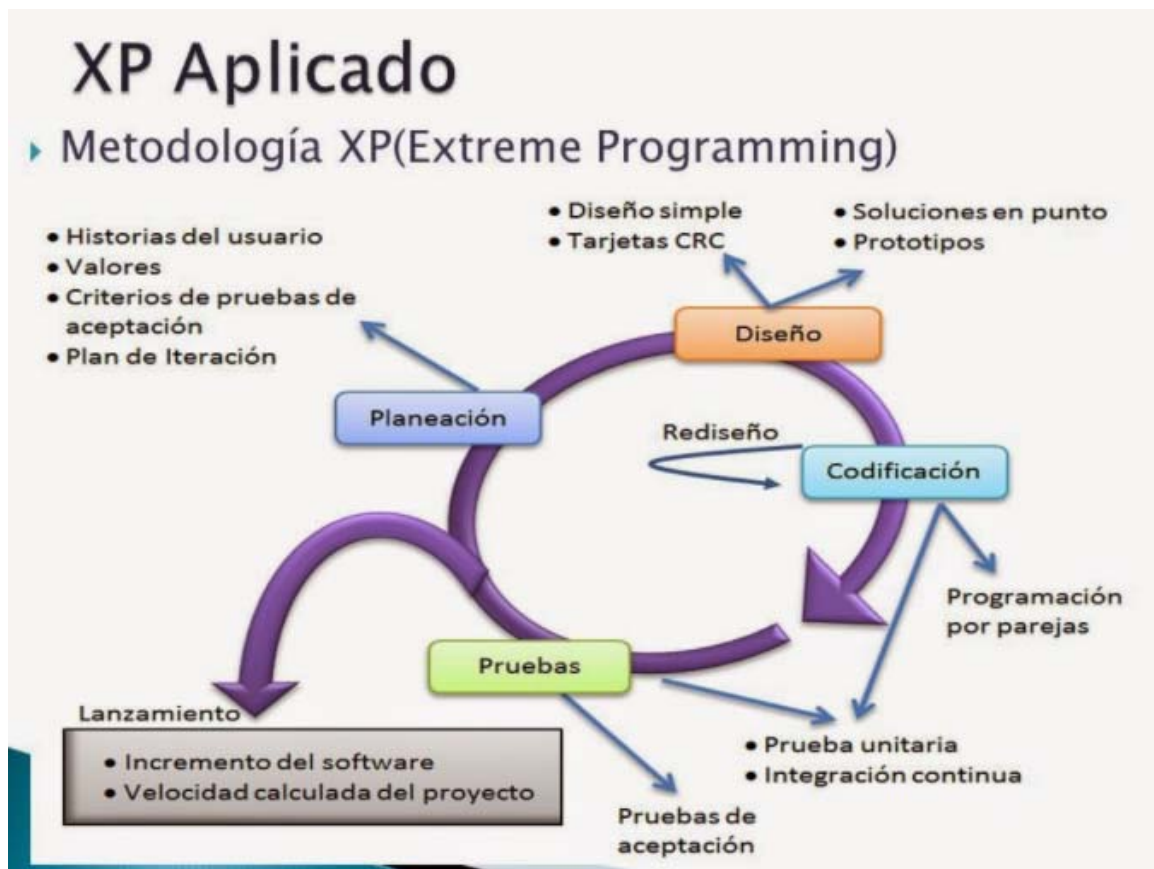


Imagen 9.

Procesos utilizados en la metodología XP

4.2.3.5 **DAS desarrollo adaptativo software**

El desarrollo adaptativo software (DAS) lo propuso Jim Highsmith en 1998 como una técnica para construir software y sistemas complejos. Los apoyos filosóficos del DAS se enfocan en la colaboración humana y la organización propia del equipo. Un enfoque de desarrollo ágil y adaptativo basado en la colaboración es " una fuente de orden en las complejas interacciones entre disciplina e ingeniería". Se define el ciclo de vida DAS, de tres fases principales:

- I. **Especulación:** en esta fase se inicia el proyecto y se conduce el ciclo adaptativo de planeación. Se utiliza información al inicio del proyecto, es decir, el objetivo de del cliente, restricciones del proyecto y los requisitos básicos. Esto permite definir el conjunto de ciclos de lanzamiento que se requerirán para el proyecto.
- II. **Colaboración:** la gente motivada trabaja de una forma que multiplica su talento y sus salidas creativas más allá de sus números absolutos. Este enfoque de colaboración es un tema recurrente en todos los métodos ágiles, pero la cooperación no es fácil. Las personas que trabajan juntas deben confiar entre sí para:
 - Criticar de forma constructiva
 - Ayudar sin resentimientos
 - Trabajar más duro de lo que ya lo hace
 - Tener el conjunto de actitudes para contribuir al trabajo en curso
 - Comunicar los problemas o preocupaciones en una forma que conduzca a la acción efectiva
- III. **Aprendizaje:** como miembros de un equipo de DAS se comienzan a desarrollar los componentes de un ciclo adaptativo, la importancia radica en el aprendizaje y en el progreso a través de un ciclo completo.

- **Grupos enfocados:** El cliente o los usuarios finales proporcionan retroalimentación sobre los avances del software. Esto indica en forma directa la satisfacción o la insatisfacción de las necesidades del negocio.
- **Revisiones técnicas formales:** Los miembros del equipo del DAS revisan los componentes del software desarrollado, mientras mejoran su calidad y su aprendizaje.
- **Post mortem:** El equipo de DAS se vuelve introspectivo al vigilar su propio desempeño y procesos, con el propósito de aprender acerca de su enfoque para después mejorarlo.

Es importante destacar que la filosofía del DAS es meritoria sin importar el modelo del proceso empleado. La dinámica de la organización propia de los equipos, la colaboración interpersonal y el aprendizaje individual conducen a los grupos de proyectos a una mayor posibilidad de éxito.



Imagen 10.

Características y procesos del DAS.

4.2.3.6 *Dynamic Systems Development Method (DSDM)*

El método de desarrollo de sistemas dinámicos (en inglés Dynamic Systems Development Method o DSDM) es un método que provee un framework para el desarrollo ágil de software, apoyado por su continua implicación del usuario en un desarrollo iterativo y creciente que sea sensible a los requerimientos cambiantes, para desarrollar un sistema que reúna las necesidades de la empresa en tiempo y presupuesto.

DSDM fue desarrollado en el Reino Unido en los años 90 por un consorcio de proveedores y de expertos en la materia del desarrollo de sistemas de información (IS). El consorcio de DSDM es una organización no lucrativa y proveedor independiente, que posee y administra el framework. La primera versión fue terminada en enero de 1995 y publicada en febrero de 1995.

Como extensión del Desarrollo rápido de aplicaciones (RAD), DSDM se centra en los proyectos de sistemas de información que son caracterizados por presupuestos y agendas apretadas. DSDM trata los problemas que ocurren con frecuencia en el desarrollo de los sistemas en lo que respecta a pasar sobre los plazos y presupuesto.

DSDM consiste en 3 fases:

- I. Fase del pre-proyecto
- II. Fase del ciclo de vida del proyecto
- III. Fase del post-proyecto.

La fase del ciclo de vida del proyecto se subdivide en distintas etapas:

- estudio de viabilidad.
- estudio de la empresa.
- iteración del modelo funcional.
- diseño e iteración de la estructura.
- implementación.

Hay 9 principios en DSDM, consistentes en cuatro fundamentos y cinco puntos de partida para la estructura del método. Estos principios forman los pilares del desarrollo mediante DSDM.

- I. Involucrar al cliente es la clave para llevar un proyecto eficiente y efectivo, donde ambos, cliente y desarrolladores, comparten un entorno de trabajo para que las decisiones puedan ser tomadas con precisión.
- II. El equipo del proyecto debe tener el poder para tomar decisiones que son importantes para el progreso del proyecto, sin esperar aprobación de niveles superiores.
- III. DSDM se centra en la entrega frecuente de productos, asumiendo que entregar algo temprano es siempre mejor que entregar todo al final, al entregar el producto frecuentemente desde una etapa temprana del proyecto, el producto puede ser verificado y revisado allí donde la documentación de registro y revisión puede ser tomada en cuenta en la siguiente fase o iteración.
- IV. El principal criterio de aceptación de entregables en DSDM reside en entregar un sistema que satisfice las actuales necesidades de negocio. No está dirigida tanto a proporcionar un sistema perfecto que resuelva todas las necesidades posibles del negocio, si no que centra sus esfuerzos en aquellas funcionalidades críticas para alcanzar las metas establecidas en el proyecto/negocio.
- V. El desarrollo es iterativo e incremental, guiado por la realimentación de los usuarios para converger en una solución de negocio precisa.
- VI. Todos los cambios durante el desarrollo son reversibles.
- VII. El alcance de alto nivel y los requerimientos deberían ser en línea base antes de que comience el proyecto.
- VIII. Las pruebas son realizadas durante todo el ciclo vital del proyecto. Esto tiene que hacerse para evitar excesos en los costos de arreglos y mantenimiento del sistema después de la entrega.

- IX. La comunicación y cooperación entre todas las partes interesadas en el proyecto es un prerequisite importante para llevar un proyecto efectivo y eficiente.



Imagen 11.

Procesos utilizados en DSDM

4.2.3.7 Lean Development (LD)

Definida por Bob Charette's a partir de su experiencia en proyectos con la industria japonesa del automóvil, en los años 80 y utilizada en numerosos proyectos de telecomunicaciones en Europa, se plantea que los cambios se consideran riesgos, pero si se manejan adecuadamente se pueden convertir en oportunidades que mejoren la productividad del cliente. Su principal característica es introducir un mecanismo para implementar dichos cambios.

El desarrollo Lean es una adaptación a los entornos de desarrollo de software del método de producción. Se fundamenta principalmente en constituir un equipo fuerte y altamente preparado, capaz de llevar a cabo cualquier tarea en poco tiempo, legando

todo a la eficacia de los integrantes del equipo y obviando los procesos y la burocracia que conlleva normalmente el tener un sistema de producción.

La filosofía Lean dicta que se debe tener un equipo muy preparado, muy motivado y muy unido, los activos más importantes a tener en cuenta cuando se está gestionando un proyecto bajo Lean Development, no son el tiempo o el dinero que estás invirtiendo sino el grado de compromiso, sobre todo, cuánto tu equipo está aprendiendo. Se considera que cuanto más hayan aprendido los miembros del equipo y más unidos se sientan, la cantidad de tiempo y dinero necesario para llevar a cabo los desarrollos será cada vez menor.

Comprendiendo esta práctica desde un punto de vista empresarial, se deberá hacer una inversión fuerte al principio, sostener a un equipo poco motivado y trabajar en mejorar su compromiso con la empresa para dotarlos de experiencia, pero al medio plazo, estos costos se reducirán y la productividad subirá.

Las ventajas que se derivan de tener un equipo muy sólido son entonces evidentes y muy útiles en el mundo de la gestión de proyectos. Dispones de unos programadores que son capaces de analizar la situación, tomar decisiones correctas y llevarlas a cabo a mayor velocidad.

En este proceso pueden ocurrir dos cosas:

- I. Sabes que tu producto final funciona como a los usuarios les gustaría porque ya has probado su comportamiento y operado los cambios oportunos en el propio proceso de desarrollo.
- II. Si el equipo se ha dado cuenta de que tu idea nunca funcionará porque has comprobado que no tiene buena aceptación entre los usuarios y decides abandonar el proyecto. En este caso has ahorrado dinero porque has podido abandonar el proyecto en menos tiempo que si hubieras seguido una metodología de desarrollo tradicional, que necesita llegar hasta el final del proceso para comprobar la efectividad del producto.

Como se puede ver, Lean va más allá de ser solamente un conjunto de técnicas, hoy por hoy es una filosofía de gestión empresarial por sí misma que se ha extendido por el mundo de las compañías dedicadas a las tecnologías.

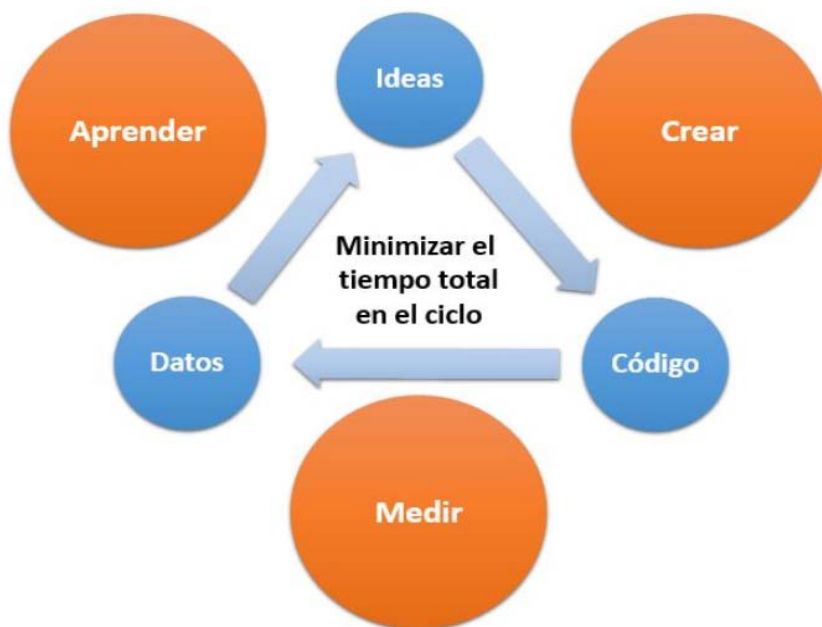


Imagen 12.

Concepto de Lean Development

4.2.4 Herramientas más utilizadas en la gestión ágil

Los cambios en las formas de vida, trabajo y organización han propiciado que las empresas actuales necesiten responder a una demanda constantemente cambiante, con productos novedosos que salgan a la venta en un tiempo récord.

Las tendencias laborales están triunfando con la colaboración, inmediatez y la motivación de los trabajadores con la ayuda de herramientas que facilitan la gestión de proyectos y la calidad de estos.

Existen miles de páginas web, programas y aplicaciones que permiten la gestión de proyectos con métodos ágiles, open source, freemium o de pago.

4.2.4.1 Axosoft (de pago)

Este software cuenta con cuatro módulos: Scrum, Gestor de fallos, Help Desk y Wiki. Optimiza los procesos operativos mediante análisis automatizados, gráficos y un tablero que permite la visualización, edición y difusión de tareas.

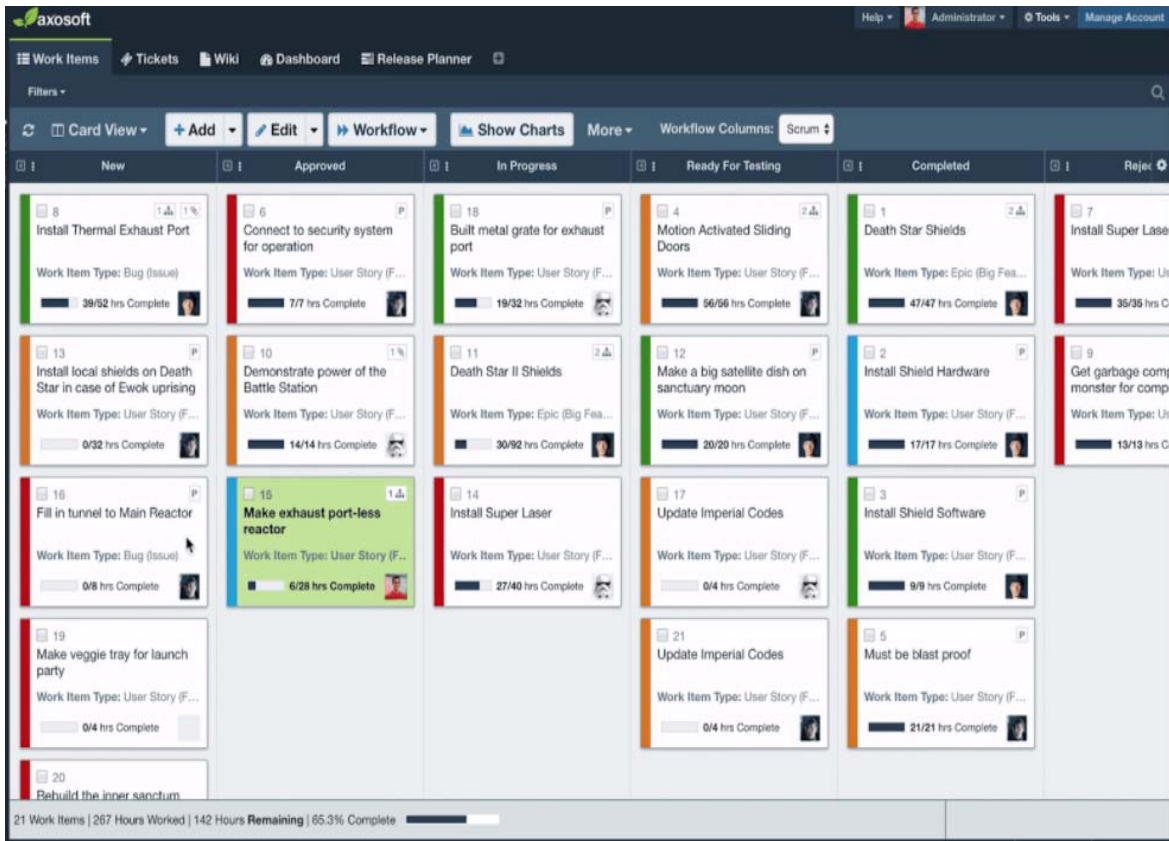


Imagen 13.
Tablero axosoft.

4.2.4.2 Atlassian Jira (de pago)

Este programa es ideal para las empresas desarrolladoras de software, permite organizar las etapas de un proyecto, asignarlas a profesionales y seguir su desarrollo en equipo.

Su interfaz es cómoda e intuitiva porque muestra todas las actividades venideras, en distintos estados como: en desarrollo, finalizadas o en incidencias, desde un mismo módulo principal organizado por columnas.

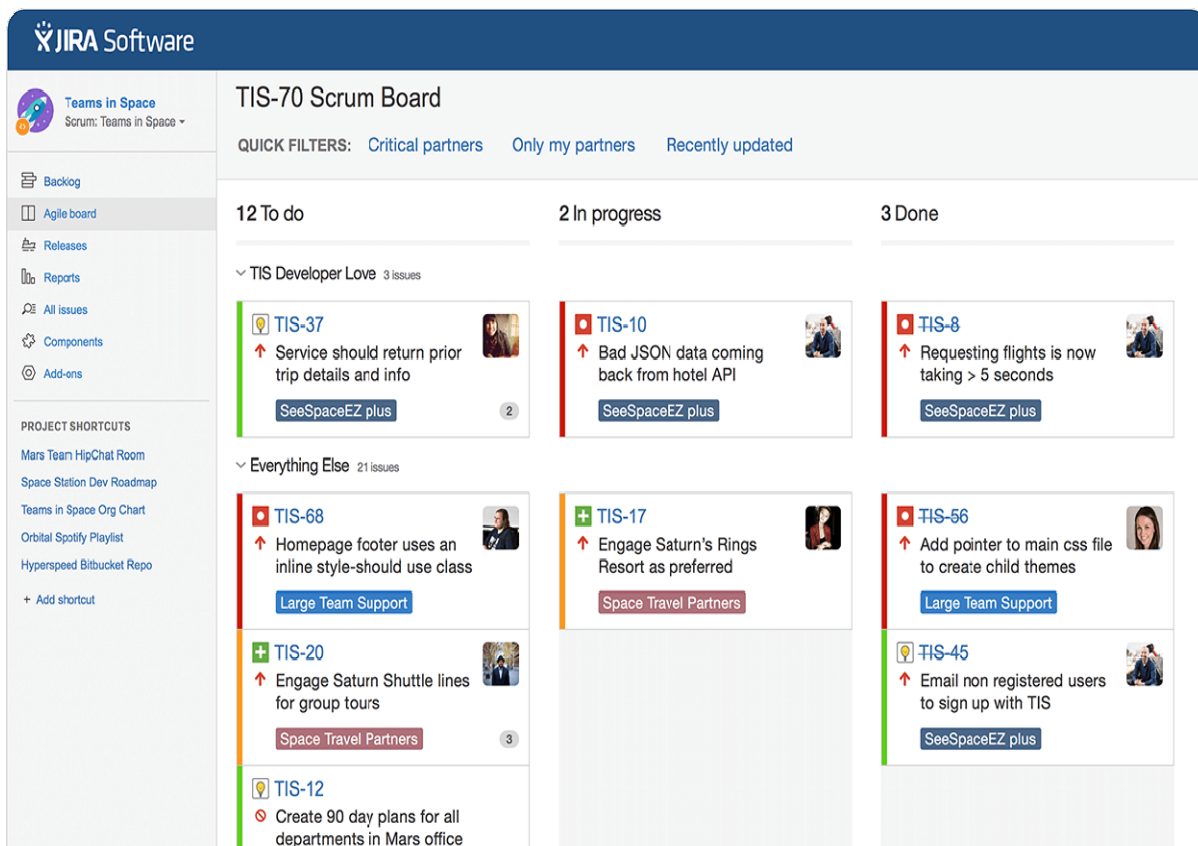


Imagen 14.
Tablero Jira

4.2.4.3 Trello (freemium)

Esta plataforma es extremadamente útil, flexible y visual. Permite organizar temporalmente los flujos de trabajo, a partir de columnas verticales a las que se pueden asignar las distintas etapas del proyecto (pendientes, en curso, finalizadas) se pueden agregar notas, y definir perfiles de determinados trabajadores. También es posible crear

categorías y añadir check-lists para marcar todas las tareas terminadas o por acabar. Trello tiene su aplicación para smartphones y tablets además de permitir el acceso web.

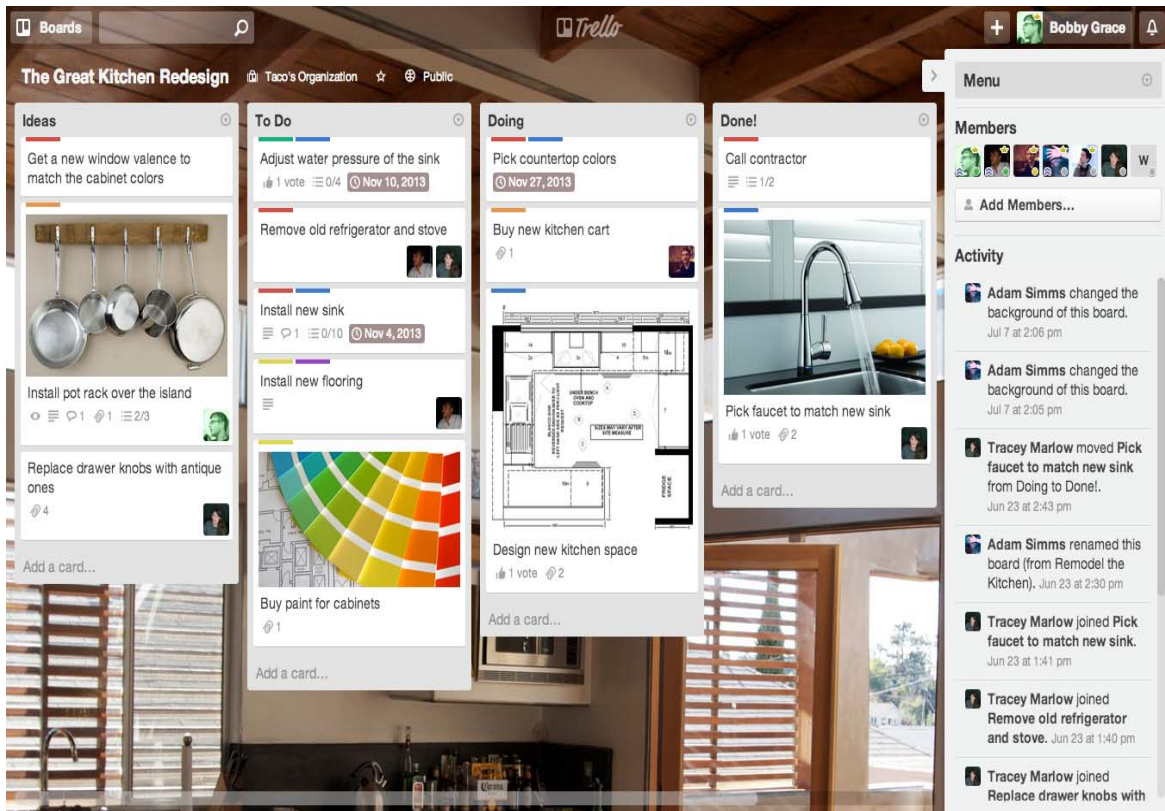


Imagen 15.
Tablero Trello

4.2.4.4 Asana (freemium)

Con más de 400.000 usuarios Asana es uno de los softwares de métodos ágiles más populares del momento. Quizás tenga algo que ver el hecho de que su creador sea Dustin Moskovitz, fundador de Facebook.

Pero lo cierto es que los usuarios tienen muy buena opinión de Asana, su diseño es intuitivo y sus funcionalidades permiten a los profesionales visualizar sus objetivos,

asignarles un tiempo y priorizarlos, recibir actualizaciones y visualizarlo todo a como un calendario.

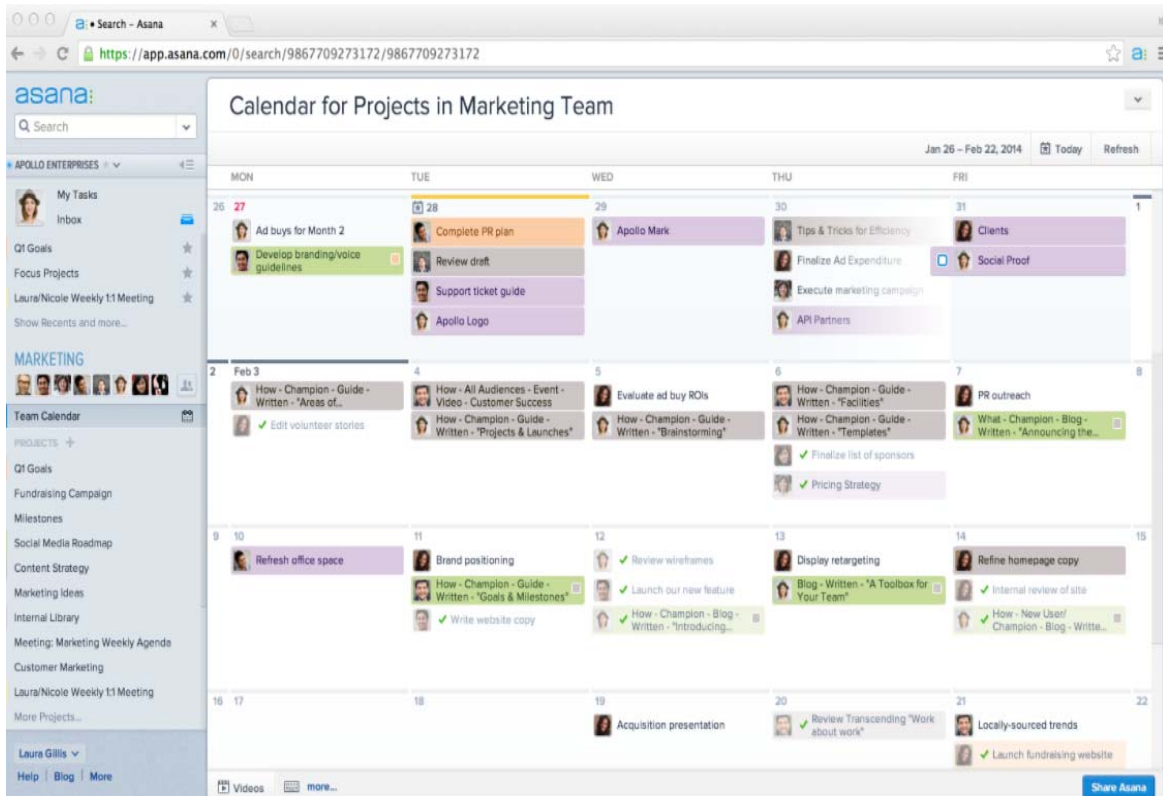


Imagen 16.
Tablero Asana

4.2.4.5 Basecamp (de pago)

Puede que la herramienta de gestión de proyectos con más reputación en todo el mundo sea Basecamp, la que permite editar pizarras de discusión, donde los trabajadores pueden escribir comentarios en tareas específicas, listar de tareas e incluso un centro de intercambio de documentos que puede integrarse fácilmente con el correo.

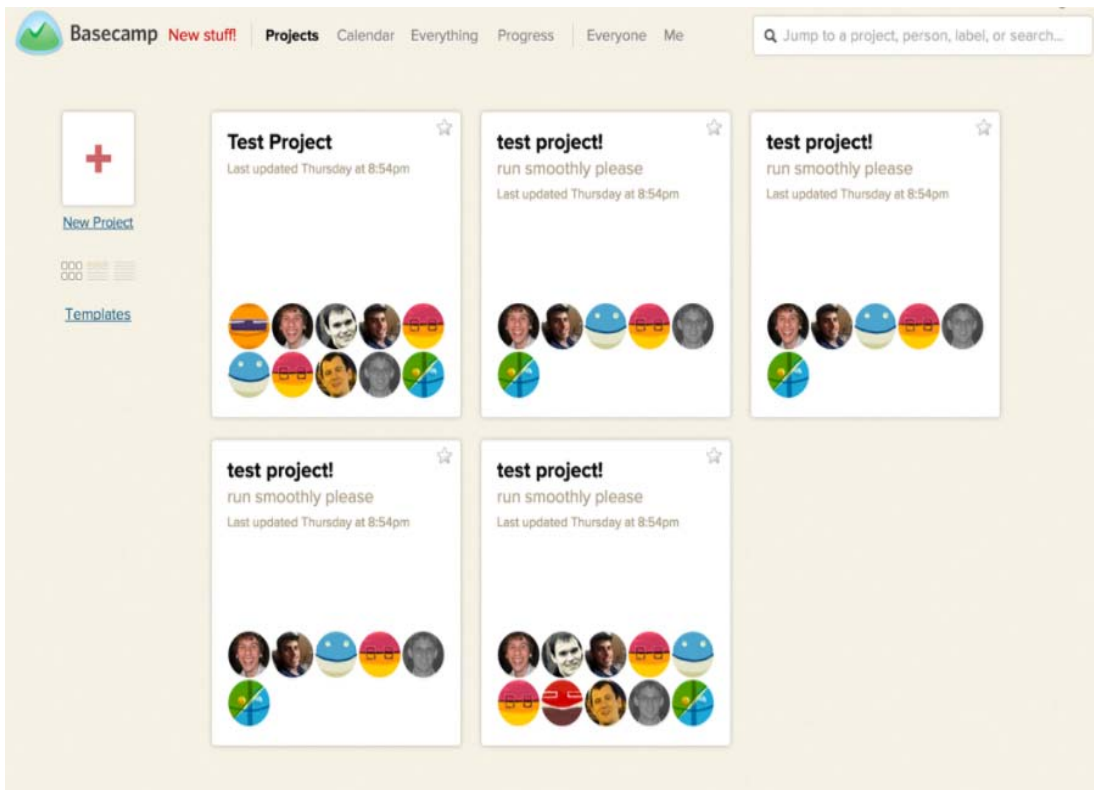


Imagen 17.
Tablero Basecamp

5 Requisitos para implementar QA

5.1 Requisitos Generales

Para comenzar en la implementación de un área de QA, se deben Identificar los procesos del negocio y determinar la secuencia e interacción de los mismos.

Por lo general, las empresas tienen divididos sus procesos por distintas áreas, es por esto que se debe definir un mapa de estos y conocer los distintos flujos, con el fin de identificar donde interactúan, esto es útil para facilitar los diseños de desarrollo y así minimizar los errores en la implementación.

Se deben realizar seguimientos, mediciones, análisis de cada uno de los procesos y monitorear cuales son los puntos críticos, donde se debe tener más cuidado a la hora de hacer modificaciones.

Se deben implementar acciones correctivas y preventivas para alcanzar niveles de maduración que permitan entregar un servicio o un producto de calidad.

Para asegurar una correcta implementación de un modelo de aseguramiento de calidad es necesario considerar los siguientes aspectos:

- Para un buen inicio de un proyecto de implementación de un sistema de aseguramiento de la calidad es imprescindible que la dirección de la empresa esté plenamente implicada y convencida de que la adopción de este sistema será un beneficio para toda la compañía.
- Anuncio a todos los niveles de la organización del compromiso adquirido por la dirección.
- Elección de un responsable de la implementación y mantenimiento del sistema, si se considera necesario, se debe crear un comité de calidad compuesto por varios profesionales con las siguientes tareas:

- Coordinar de principio a fin la implementación del sistema.
- Revisar y aprobar los procedimientos operativos.
- Formar el equipo de trabajo y efectuar revisiones periódicas.

Requisitos de la documentación

Para poder tener procesos de ejecución de pruebas definidos, debemos declarar políticas y objetivos de calidad, es por esto que se deben documentar los procesos en un manual de calidad. En este manual se debe estipular de forma clara y concisa los procedimientos obligatorios y estándares que defina cada organización, los cuales se deben seguir, ejecutar y respetar.

También crear procedimientos de Control de documentos (Mantenciones y actualización de estos), Control de registros, auditorías internas, acciones correctivas y acciones preventivas, con el fin de asegurar la eficacia en la planificación, el funcionamiento y el control de los procesos.

Control de los documentos

Se deben realizar procedimientos documentados de control y mantenciones en el cual se defina: la elaboración, revisión, aprobación, actualización de cambios, disponibilidad de los documentos y control de documentos obsoletos, con el fin de tener respaldos de cada cambio que se le realice.

Compromisos de la dirección

La alta dirección comunica la importancia de satisfacer las necesidades de los clientes, definir políticas de calidad y velar por los entregables.

Se debe establecer una política de calidad y definirla, fijar los objetivos en todo el proceso de desarrollo.

Se deben llevar a cabo revisiones por la dirección la cual asegure que el proceso se ejecute y finalice con todos los estándares y protocolos fijados por la organización, en el cual se especifiquen los plazos y atrasos dentro de cada proyecto, gestionar la disponibilidad de recursos económicos dentro del año, los cuales contemplen los gastos fijos de un proyecto y la cantidad de personas involucradas dentro de los distintos proyectos.

Enfoque en el cliente

Asegurarse que los requisitos del cliente se definan y se cumplan, para lograr aumentar su satisfacción con el producto y así dar una mejor imagen como compañía.

La gerencia debe recordar constantemente a todos los funcionarios la importancia de satisfacer las necesidades de los clientes.

Documentar una metodología específica para lograr aumentar la satisfacción de los clientes, esta debe ser medible para obtener métricas y mejorar constantemente los procesos internos del área de QA.

Políticas de calidad

Es importante que las políticas sean adecuadas al propósito de la organización, se deben definir compromisos de mejoramiento continuo, ser acorde con cada uno de los objetivos de calidad, ser comunicadas y entendidas por todos los miembros de la organización por medio de distintos canales de comunicación.

Será necesario crear un programa de capacitación para asegurar que las políticas de calidad sean entendidas por todas las personas y que se transmita la importancia que tiene la calidad para la compañía y sus clientes.

5.2 Responsabilidades, autoridad y comunicación

Responsabilidad y autoridad

Las responsabilidades se deben definir y se deben comunicar en la organización por medio de los canales de comunicación disponibles.

Cada funcionario debe conocer las funciones de su cargo y saber a quién dirigirse cuando necesite de su ayuda. Para esto, los organigramas deben estar bien definidos y los responsables deben tener la disposición de aclarar dudas y prestar ayuda cuando sea necesario.

Representantes de la dirección

- El representante debe asegurar que se cumplan las normas y estándares definidos por la organización.
- Asignar roles y recursos a los proyectos.
- Gestionar indicadores o métricas con el fin de crear instancias para mejorar continuamente en los procesos de calidad.
- Asegurar que todos los funcionarios sean conscientes de cumplir con los requisitos del cliente y las normas establecidas.
- Asegurar que la comunicación entre las distintas áreas del proyecto y los clientes se efectúen periódicamente.

Comunicación interna

- Se deben establecer procesos de comunicación adecuados.
- Crear formatos de actas de reuniones.
- Políticas de comunicación internas
- Correos corporativos los cuales indiquen los cambios en los procesos y en los sistemas.

Información para las revisiones

- Se debe tener información sobre los resultados de las auditorías.
- Retroalimentación del cliente.
- Desempeño de los procesos y conformidad del producto.
- Estado de las acciones correctivas y preventivas.
- Acciones de seguimiento a las revisiones realizadas por la dirección
- Cambios y recomendaciones para mejoras en los procesos de revisión.

Resultados de las revisiones

- Se deben crear formatos estándares con las métricas esperadas.
- Medir y controlar los atrasos.
- Medir y controlar los resultados esperados.
- Documentar los resultados.

5.3 Gestión de los recursos**Provisión de los recursos**

- Es importante determinar y proporcionar los recursos necesarios para implementar, mantener y mejorar continuamente la eficacia del departamento de calidad, la dirección debe ser capaz de proporcionar los recursos necesarios para que sus funcionarios puedan desempeñar un trabajo eficiente en los plazos estimados.
- Se deben realizar diagnósticos sobre los recursos que la empresa tiene para determinar aquellos que le hacen falta y así poder implementar, mantener y mejorar la calidad de los servicios y proyectos.

Recursos humanos

- Se debe contar con personal competente, con formación, habilidades y experiencia apropiadas para la ejecución de proyectos de tecnología.
- Deben existir procedimientos de selección de personal con las habilidades necesarias para los puestos de trabajos requeridos por la compañía y así contar con un personal capacitado frente a las exigencias de los proyectos.

Competencia y formación.

- Se debe determinar las competencias necesarias del personal.
- Crear criterios de evaluación de competencias.
- Proporcionar formación al personal.
- Evaluar el trabajo y gratificar cuando sea pertinente.
- Se debe crear una política de capacitación.
- Crear una política de integración para los nuevos empleados.

Infraestructura

- Proporcionar y mantener la infraestructura que incluya espacios donde las personas se sientan cómodas.
- Suministrar las herramientas y tecnologías adecuadas para el desarrollo de los proyectos.
- Mantener procedimientos y políticas de mantenimiento el cual especifique cómo mantener en buen estado la infraestructura de la organización.

5.4 Procesos en la ejecución de los proyectos

Identificación y trazabilidad

Es muy importante identificar la trazabilidad de los productos con el objetivo de tener los puntos críticos identificados, ya que facilitará:

- La identificación de los módulos y procesos en lo que se debe tener más cuidado en una revisión.
- Estimar los tiempos de ejecución de las pruebas.
- Clasificar distintas pruebas y categorizarlas por criticidad
- Optimizará la ejecución y tiempos de revisión en los proyectos.

5.5 Seguimientos, medición y propósitos

Satisfacción del cliente

- Determinar y documentar métodos para obtener información relativa a la percepción del cliente con respecto al cumplimiento de sus requisitos.
- Crear un procedimiento, en el cual se especifiquen los aspectos que se van a evaluar, el nivel de satisfacción mínimo requerido y la frecuencia con la cual se lleva a cabo la encuesta.
- Crear un formato que debe ser documentado con los resultados de las mediciones el cual permitirá medir las distintas evaluaciones, estos serán indicadores claves para tomar decisiones de cambios de los procesos internos y así dar un mejor servicio.

Política de calidad

- Se debe implantar una política de calidad como una responsabilidad de la compañía para brindarles a los clientes productos y servicios de calidad.
- La misión como área de QA es asegurar la calidad en los sistemas. Esto está directamente relacionado con el producto o servicio que se les entregará a los usuarios o clientes finales, es por esto que se debe tomar conciencia de la importancia que tiene el área de QA.

6 CONCLUSIONES

Toda organización ha sido creada para obtener ingresos, los cuales son el motor para adquirir bienes. Estos se obtendrán en la medida que se logren vender los productos o servicios que se ofrecen, sin embargo, se requiere que estos llamen la atención y que sean de calidad, es decir los productos y servicios deben poseer características que cumplan con los requisitos y exigencias solicitados por los clientes.

Teniendo en cuenta lo anterior, la implementación de un sistema de gestión de calidad es importante ya que garantiza en un alto porcentaje que las características del producto o servicio cumplan con los requisitos y procesos solicitados por el cliente, evitando al máximo la presencia de inconformidades. Esto implicará una inversión económica por parte de la empresa, sin embargo, si se tiene en cuenta la relación costo/beneficio, serán mayores los beneficios que los costos. Por último, el costo que implica la creación de un área de calidad se verá reflejado en el producto final, generará confianza y credibilidad a los clientes.

En cuanto a las metodologías ágiles podemos concluir que no existe una metodología universal para hacer frente con éxito a cualquier proyecto de desarrollo de software. Toda metodología debe ser adaptada al contexto del proyecto (recursos técnicos y humanos, tiempo de desarrollo, tipo de sistema, etc).

Históricamente, las metodologías tradicionales han intentado abordar la mayor cantidad de situaciones de contexto del proyecto, exigiendo un esfuerzo considerable para ser adaptadas, sobre todo en proyectos pequeños y con requisitos muy cambiantes.

Las metodologías ágiles ofrecen una solución casi a medida para una gran cantidad de proyectos que tienen estas características, una de las cualidades más destacables en una metodología ágil es su sencillez, tanto en su aprendizaje como en su aplicación, reduciéndose así los costos de implantación en un equipo de desarrollo. Esto ha llevado hacia un interés creciente por las metodologías ágiles.

Falta aún un cuerpo de conocimiento consensuado respecto a los aspectos teóricos y prácticos de la utilización de metodologías ágiles, así como una mayor consolidación de

los resultados de aplicación. La actividad de investigación está orientada hacia líneas tales como: métricas y evaluación del proceso, herramientas específicas para apoyar prácticas ágiles, aspectos humanos y de trabajo en equipo. Cualquiera sea la metodología ágil a utilizar existe una condición obligatoria que si bien puede resultar bastante obvia a veces no se logra fácilmente: Todo el equipo debe usar la metodología elegida, no usarla solamente como una serie de prácticas aisladas ya que no tendría ningún valor, además hay que tener presente una serie de inconvenientes y restricciones para su aplicación, tales como:

- Están dirigidas a equipos pequeños o medianos (se sugiere que el tamaño de los equipos se limite de 3 a 20 integrantes como máximo)
- 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 el proceso al fracaso (el clima de trabajo, la colaboración y la relación contractual son claves)
- Uso de tecnologías que no tengan un ciclo rápido de realimentación o que no soporten fácilmente el cambio.
- Falta conocimiento consensuado respecto a los aspectos teóricos y prácticos de la utilización de metodologías ágiles, así como una mayor consolidación de los resultados de aplicación.

Aunque en la actualidad ya existen libros asociados a cada una de las metodologías ágiles existentes, es importante considerar que la práctica hace que las metodologías se adapten mejor al negocio de cada organización.

La calidad en los proyectos, debe ser el objetivo permanente de la organización ya que está directamente relacionada con el producto final que se entrega a los clientes, una buena ejecución de proyecto de calidad o certificación debe contar con las siguientes etapas:

Planificar

- Organización lógica del trabajo
- Identificación del problema y planificación.
- Observaciones y análisis.
- Establecimiento de objetivos a alcanzar.
- Establecimiento de indicadores de control.

Hacer

- Correcta realización de las tareas planificadas
- Preparación exhaustiva y sistemática de lo previsto.
- Aplicación controlada del plan.
- Verificación de la aplicación.

Comprobar

- Comprobación de los logros obtenidos
- Verificación de los resultados de las acciones realizadas.
- Comparación con los objetivos.

Ajustar

- Posibilidad de aprovechar y extender aprendizajes y experiencias adquiridas en otros casos
- Analizar los datos obtenidos.
- Proponer alternativa de mejora.
- Estandarización y consolidación.
- Preparación de la siguiente etapa del plan.

La excelencia ha de alcanzarse mediante un proceso de mejora continua, en todos sus campos, desde las capacidades del personal hasta en la eficiencia de los recursos,

alcanzar los mejores resultados no es labor de un día, es un proceso progresivo en el que no puede haber retrocesos, lo deseable es mejorar día a día, tomarlo como hábito y cultura en la compañía.

Cuando se detecta un problema, la respuesta y solución ha de ser inmediata, no debe haber demoras, de lo contrario se podrían originar consecuencias desastrosas para la compañía y perder clientes.

La Calidad en los sistemas implica tanto la implementación, como el aprendizaje continuo de la organización, el seguimiento de una filosofía de gestión, y la participación activa de todas las personas.

Las empresas no pueden seguir utilizando las malas prácticas de no utilizar plenamente la capacidad intelectual, creativa y la experiencia de su personal, ya pasaron los tiempos en que unos pensaban y otros sólo trabajaban. Se debe cambiar la forma a pensar en “equipos” donde todos tienen el deber de plantear soluciones, así como también dar lo mejor de sí para obtener el éxito de la organización.

Una institución que practica una cultura de calidad tiene la capacidad de trabajar en equipo; sabe planear, controlar y mejorar los procesos con los que se relaciona mediante procedimientos, también tiene una actitud de servicio hacia los clientes tanto internos como externos a quienes siempre trata de satisfacer.

Las culturas no pueden ser evaluadas o comparadas en forma objetiva, una cultura es exitosa siempre que ayude a la sociedad a enfrentar los retos que se le presentan. Por lo tanto, mientras los retos cambien, el patrón cultural tendrá que sufrir modificaciones en lo referente a los hábitos, las prácticas y algunos valores.

El concepto de QA surge por la necesidad de responsabilizar la calidad de un software en un equipo de trabajo encargado de certificar que los requerimientos se cumplan (requerimientos de diseño, performance, de flujos, integración etc.) Todo equipo de desarrollo debería contar con al menos un responsable encargado de asegurar el correcto funcionamiento del software que se desarrolla. No debe interpretarse como alguien que constantemente está corrigiendo lo que no hacen bien los desarrolladores, sino como un facilitador para la realización de pruebas.

Durante la fase de pruebas hay que ser consciente de las expectativas que se persiguen, es decir, qué tipo de requisitos y cómo deben ser implementados, teniendo las ideas claras, definidas y ordenadas por prioridad, solo así se puede comenzar a crear los primeros casos de prueba.

QA debe situarse entre la parte de negocio y la parte técnica, ayudar al negocio a traducir lo que el cliente quiere, por otro lado, validar los criterios de aceptación del software implementados por los desarrolladores.

A la hora de definir un plan de testing debemos recopilar los requisitos funcionales que definen el alcance del proyecto, esto es el núcleo que conforma la parte del negocio que fijamos con el cliente y los usuarios.

Hay que estimar el tiempo en que se realizaran las pruebas, cuando se avanza de forma precipitada es cuando se cometen más fallas en los procesos de revisión, es muy recomendable involucrar a los usuarios claves que han participado en la definición de los requerimientos y mantener feedback frecuentemente.

7 GLOSARIO

Algoritmo

Se denomina algoritmo a un grupo finito de operaciones organizadas de manera lógica y ordenada que permite solucionar un determinado problema, se trata de una serie de instrucciones o reglas establecidas que, por medio de una sucesión de pasos, permiten llegar a un resultado o solución.

Según los expertos en matemática, los algoritmos permiten trabajar a partir de un estado básico o inicial y tras seguir los pasos propuestos, llegar a una solución. Cabe resaltar que, si bien los algoritmos suelen estar asociados al ámbito matemático (ya que permiten, averiguar el cociente entre un par de dígitos o determinar cuál es el máximo común divisor entre dos cifras pertenecientes al grupo de los enteros), aunque no siempre implican la presencia de números.

Los algoritmos se pueden expresar a través de lenguajes de pseudocódigo, el lenguaje natural y también a través de los conocidos diagramas de flujo.

Esta amplitud de significado permite apreciar que no existe una definición formal y única de algoritmo. El término suele ser señalado como el número fijo de pasos necesarios para transformar información de entrada (un problema) en una salida (su solución). De todas formas, algunos algoritmos carecen de final o no resuelven un problema en particular.

Cabe mencionar por último que los algoritmos son muy importantes en la informática ya que permiten representar datos como secuencias de bits. Un programa es un algoritmo que indica a la computadora qué pasos específicos debe seguir para desarrollar una tarea.

Auditoría

Auditoría es un término que puede hacer referencia a tres cosas diferentes pero conectadas entre sí: puede referirse al trabajo que realiza un auditor, a la tarea de estudiar la economía de una empresa, o a la oficina donde se realizan estas tareas (donde trabaja el auditor). La actividad de auditar consiste en realizar un examen de los procesos y de la actividad económica de una organización para confirmar si se ajustan a lo fijado por las leyes o los buenos criterios.

Puede decirse que la auditoría es un tipo de examen o evaluación que se lleva a cabo siguiendo una cierta metodología. Lo habitual es que el auditor no pertenezca a la entidad auditada. Existen grandes firmas dedicadas a las auditorías contables, como Pricewater house Coopers, Deloitte, KPMG y Ernst & Young.

La persona encargada de realizar dicha evaluación recibe el nombre de auditor. Su trabajo implica analizar detenidamente las acciones de la empresa y los documentos donde las mismas han sido registradas y así determinar si las medidas que se han tomado en los diferentes casos son adecuadas y han beneficiado a la compañía.

Una auditoría es una de las formas en las que se pueden aplicar los principios científicos de la contabilidad, donde la verificación de los bienes patrimoniales, la labor y beneficios alcanzados por la empresa son primordiales, pero estos no son lo único importante. La auditoría intenta también brindar pautas que ayuden a los miembros de una empresa a desarrollar adecuadamente sus actividades, evaluándolos, recomendándoles determinadas cosas y revisando detenidamente la labor que cada uno cumple dentro de la organización.

Bugs

Se trata de un concepto utilizado por todos aquellos que tienen conocimientos en el campo de la informática. Esta palabra inglesa, cuya traducción literal es “bicho”, se usa para nombrar a los errores que se producen en un programa informático.

Un bug se genera en la programación o el diseño del software y en algún momento se manifiesta ante el usuario. Algunos bugs habituales son la inclusión de variables que no fueron inicializadas en el momento preciso, la mala indexación de las tablas en una base de datos, la creación de un bucle infinito, el uso de fuentes que resultan difíciles de leer o la elección de colores que confunden a los usuarios.

Lo habitual es que el lenguaje de programación tipifique los posibles bugs para que, de este modo, el programador pueda resolver los errores de una forma sencilla. Por lo general, los lenguajes organizan los bugs según su clasificación ya sea en el momento de ejecución del programa o un error en la compilación.

Check list

Check list o lista de chequeo es un tipo de ayuda de trabajo informativo, se conoce también como: Listas de control u hojas de verificación.

Los check list, como herramienta metodológica están compuesta por una serie de ítems, factores, propiedades, aspectos, componentes, criterios, dimensiones o comportamientos, necesarios de tomarse en cuenta, para realizar una tarea, controlar y evaluar detalladamente el desarrollo de un proyecto, evento, producto o actividad. Dichos componentes se organizan de manera coherente para permitir que se evalúe de manera efectiva, la presencia o ausencia de los elementos individuales enumerados o por porcentaje de cumplimiento u ocurrencia.

En otras palabras, check-list un listado de preguntas, en forma de cuestionario que sirve para verificar el grado de cumplimiento de determinadas reglas o actividades establecidas con un fin determinado. La lista de chequeo es en sí misma, es una ayuda para la memoria, al proporcionar un método para una rápida verificación de los ítems planteados según el tipo de asunto a controlar. Contribuyen a normalizar o estandarizar líneas de acción sistemáticas detallando cada uno de los puntos de actividad o proceso. En síntesis, los check list son dispositivos metodológicos y nemotécnicos, que reducen la complejidad para comprobar solamente los elementos importantes, con ello reducen errores de omisión.

Ciclos de proyecto

Todo proyecto de ingeniería tiene fines ligados a la obtención de un producto, procesos o servicios que son necesarios generar a través de diversas actividades. Algunas de estas actividades pueden agruparse en fases ya que globalmente contribuyen a obtener un producto intermedio, necesario para continuar hacia el producto final y así facilitar la gestión del proyecto. Al conjunto de las fases empleadas se le denomina “ciclo de vida”. Sin embargo, la forma de agrupar las actividades, los objetivos de cada fase y los tipos de productos intermedios, pueden ser muy diferentes dependiendo del tipo de producto o proceso a generar e incluso de las tecnologías utilizadas.

La complejidad de las relaciones entre las distintas actividades crece exponencialmente con el tamaño, con lo que rápidamente se hacen inabordables. De esta forma la división de los proyectos en fases sucesivas es un primer paso para la reducción de su complejidad, tratándose de escoger las partes de manera que sus relaciones entre sí sean lo más simples posibles.

CMMI

CMMI es el acrónimo de Capability Maturity Model Integration y se refiere a los modelos que contienen las mejores prácticas que ayudan a las organizaciones a mejorar sus procesos. Han sido desarrollados por equipos de trabajo formados por especialistas de la industria, el Software Engineering Institute (SEI) que transfirió los derechos al CMMI Institute para su operación y comercialización.

Siendo un modelo refleja una abstracción de la realidad que permite a las organizaciones adoptar prácticas útiles para alcanzar sus objetivos de negocio.

La adecuada interpretación del modelo para cubrir las diferentes situaciones, necesidades y objetivos de una organización son esenciales para lograr los resultados que se quieren. Muchas veces por desconocimiento o por falta de sentido común o criterio, el resultado no es lo esperado. Por ello es importante saber qué es CMMI y qué no esperar de CMMI.

Según el modelo que se utilice se puede obtener el documento con un conjunto de guías que ayudan en:

- Desarrollo y mantenimiento de productos y servicios (CMMI DEV),
- Adquisición de productos y servicios (CMMI ACQ) y
- Establecimiento, entrega y gestión de los servicios (CMMI SVC).

Contiene elementos esenciales de un proceso efectivo y propone una forma de adopción para las organizaciones que permite incrementar la calidad y productividad al tiempo que controla el presupuesto y los compromisos establecidos. Cada una debe interpretar, adoptar y aplicar aquellas prácticas que apoyan el logro de sus objetivos y el cumplimiento de sus necesidades de manera eficiente.

Considera dos enfoques o rutas para adoptar las mejoras y medir el nivel en que han evolucionado y se conocen como representaciones.

- I. En una forma se consideran áreas de proceso de manera individual y se califican en niveles de capacidad de acuerdo con la representación continua.

- II. El otro enfoque considera un conjunto preestablecido de áreas de proceso que constituyen un nivel de madurez y que es la forma de evaluar la representación escalonada o por etapas.

Está estructurado para facilitar su uso en elementos que definen la forma y modo de aplicarlo, considerando los elementos que son obligatorios, sugeridos o el material informativo en las áreas de proceso.

Control de calidad

El control de calidad consiste en la implementación de programas, mecanismos, herramientas y/o técnicas en una empresa para mejorar la calidad de sus productos, servicios y productividad.

El control de la calidad es una estrategia para asegurar el cuidado y mejorar continuamente en la calidad y productos.

Ventajas de establecer procesos de control de calidad

- Muestra el orden, la importancia y la interrelación de los distintos procesos de la empresa.
- Se realiza un seguimiento más detallado de las operaciones.
- Se detectan los problemas antes y se corrigen más fácilmente.
- Se establece un plan de calidad, donde se recopilan datos de los proyectos y procedimientos con el fin de maximizar la calidad de las operaciones y por consiguiente, la satisfacción de los consumidores.

Diagrama de flujo

Se conoce como diagramas de flujo a aquellos gráficos representativos que se utilizan para esquematizar conceptos vinculados a la programación, la economía, los procesos técnicos y casi cualquier temática de análisis.

Los diagramas de flujo son múltiples y diversos, pueden abordar muchos temas distintos de formas muy diferentes.

Comúnmente, se utiliza este tipo de diagramas para detallar el proceso de un algoritmo y la trayectoria de operaciones precisas a través de flechas. Siempre que existe un diagrama de flujo existe un proceso o sistema que pretende ser graficado a través de símbolos visuales que, en vez de términos verbales, simplifican el funcionamiento de dicho proceso y lo hacen más claro y evidente al lector.

Para que el diagrama de flujo tenga sentido, debe existir un camino hacia una solución que parte de un único inicio a un único punto final.

Los símbolos más utilizados en los diagramas de flujo son la flecha (que indica sentido y trayectoria), el rectángulo (representa un evento o proceso), el rombo (una condición), el círculo (un punto de conexión) entre otros.

Encriptación

La etimología de encriptar nos remite a la expresión inglesa to encrypt, a su vez derivada del vocablo griego enkrýptein (que se puede traducir como “ocultar”). La acción de encriptar consiste en copiar un mensaje utilizando una clave.

Cabe recordar que una clave es un código de signos que se interpretan según determinadas correspondencias y reglas. Al encriptar un texto, se apela a un código para que su contenido solo resulte accesible a aquellos que conocen las normas que regulan la clave en cuestión.

Encriptar, en definitiva, consiste en cifrar: es decir, en transcribir un texto en signos (letras, números, etc.) de acuerdo con una determinada clave, de este modo es posible proteger su contenido.

Se denomina criptografía a la técnica y la especialidad consistente en encriptar. Gracias a la criptografía, se puede transformar un texto con una clave de cifrado, volviéndolo incomprensible a quienes no disponen de la clave de descifrado.

Estimar

Estimación (o estimar) es el proceso de encontrar una aproximación sobre una medida, lo que se ha de valorar con algún propósito, es utilizable incluso si los datos de entrada pueden estar incompletos, incierto, o inestables.

La estimación en la gestión de proyectos es el tiempo que emplea un recurso en completar una asignación. El valor es importante para la planificación del proyecto y para el reconocimiento de los ingresos. A corto plazo, las estimaciones ayudan a los jefes de proyecto a adjudicar las horas de trabajo de una forma más eficaz. A largo plazo, los gestores de proyectos podrán comparar las estimaciones con los datos reales, lo cual puede ayudar a producir planificaciones y previsiones más precisas.

Freemium

Freemium, es un término popularizado en 2006 por Fred Wilson de Union Square Ventures. Es un modelo comercial que funciona ofreciendo un juego, producto o un servicio gratuito, mientras que se cobra una tarifa para características avanzadas, funcionalidades, o productos y servicios adicionales. La palabra “freemium” es la combinación de dos aspectos del modelo comercial: “free” y “premium”.

No es un modelo nuevo, pero está adquiriendo popularidad porque algunas empresas se han vuelto rentables con su aplicación, por definición, tener un producto gratis permite conseguir nuevos clientes fácilmente. La economía de Internet hace esto muy atractivo, porque el costo marginal de cada usuario nuevo es muy bajo.

Factores de éxito

- I. Conversión: Uno de los factores clave de este modelo es poder convertir los suscriptores gratuitos en clientes premium. Hacer esto es más fácil cuando la empresa puede controlar sus costos y por ende ofrecer un precio bajo.
- II. Cantidad: Freemium es un juego de números: si solamente el 1% de sus usuarios van a pagar, entonces se necesita tener cantidades muy grandes de usuarios gratuitos para tener utilidades de un 1%.
- III. Valor agregado: Hay que crear servicios y productos adicionales de alto valor agregado para que el usuario gratuito sienta que vale la pena pagar el precio premium. El valor de los servicios debe ser también alto porque en Internet hay mucha competencia para cualquier software que sea gratis.
- IV. Costos bajos: El modelo freemium funciona porque el costo de cada usuario adicional es bajo, por lo que es necesario mantener los costos operacionales bajos.
- V. Análisis: Analizar cuáles son los tipos de usuarios que no pagan y que eventualmente se convierten en pagadores y cuales productos o servicios son adquiridos más a menudo pagando desde un principio. El modelo freemium es un negocio de altos volúmenes y bajos porcentajes entonces pequeñas diferencias porcentuales pueden generar grandes diferencias en utilidades.

Gestión de proyectos

La gestión de proyectos son todas aquellas acciones que se deben realizar para cumplir con una necesidad definida dentro de un periodo de tiempo y en el cual se utilizan recursos, herramientas y personas y que tienen un costo que se debe tener en cuenta cuando se realiza los presupuestos.

El objetivo cuando se hace gestión de proyectos, es que se debe hacer un producto original y que cumpla una necesidad específica, la que debe ser clara. Esta es una de las dificultades de la gestión de proyectos, ya que no suelen haber precedentes en la organización.

Cuando se gestionan los proyectos, se pueden encontrar varios beneficios que ayudarán a la empresa en diferentes ámbitos:

- Posibilita una respuesta rápida a las diferentes demandas del mercado, ya que se maximiza la capacidad de reacción de los empleados, y se consigue más a un menor costo, aunque más allá de cumplir con los objetivos presupuestarios es muy importante aportar valor y calidad.
- Externamente, gracias a la gestión de proyectos se aporta a una mejora en la comunicación de la empresa lo que posibilita fijar objetivos globales, Con una buena gestión de proyectos, se crea una mejor transferencia de información entre los diferentes departamentos, y se marcan prioridades con más facilidad.
- Se aprende de las lecciones del pasado y así se utiliza la experiencia de planificación para usarla en proyectos futuros. Durante las fases de ejecución y desarrollo del proyecto se identificarán errores, los que deben ser corregidos permitiendo así definir procesos más optimizados y crear buenas prácticas las que se podrán reutilizar.
- Se obtiene una correcta información sobre la verdadera capacidad del equipo de trabajo, ya que en la gestión de proyectos se prioriza el trabajo en equipo entre los empleados de la organización. Uno de los principales roles del director de

proyectos es conseguir que las personas trabajen de forma sincronizada y en armonía. Saber identificar los problemas, gestionar los conflictos, negociar, dinamizar la cooperación, estas son las cualidades imprescindibles que debe tener un buen Project manager.

- Asegura calidad, ya que permite proporcionar al cliente un producto que cumple los requisitos de este, y con la adecuación de uso.
- Entrega puntual, exceder el presupuesto cumpliendo con las expectativas de calidad, permiten consolidar la imagen de marca y generar oportunidades de negocio a la organización.

A la hora de gestionar un proyecto, hay que tener en cuenta que este tiene cuatro fases:

- I. Planificación: La planificación es la primera fase en la gestión de un proyecto, y es previa al desarrollo de este. En esta fase se debe definir cuáles serán las actividades que se tendrán que hacer, estimar la duración del proyecto y también se debe indicar cuáles serán los recursos necesarios para poder llevar a cabo con éxito el proyecto.
- II. Programación: Esta es la segunda fase de la gestión de proyectos, y consiste en crear el calendario de ejecución del proyecto, indicando en qué fechas se hará cada parte del proyecto.
- III. Seguimiento y control: Esta fase es la única fase de toda la gestión que se hace durante el proyecto. Consiste en seguir y controlar su ejecución. Es decir, obtener datos de la evolución y tomar las medidas correctivas en caso de que hubiese algún inconveniente.
- IV. Análisis y evolución: Esta fase se ejecuta después de haber terminado el proyecto y se debe medir las diferencias entre lo que se había previsto durante la fase de planificación, y lo que realmente ocurrió. Esta fase es importante también porque permite observar los errores que se han cometido durante todo el proceso, y es donde se aprende de la experiencia.

Help Desk

La tecnología Help Desk (Ayuda de Escritorio) es un conjunto de servicios, que de manera integral, a través de uno o varios medios de contacto, ofrece la posibilidad de gestionar y solucionar todas las posibles incidencias, junto con la atención de requerimientos de las áreas de Tecnologías.

Como su nombre lo dice, es una Mesa de Ayuda, donde se ofrecen Servicios acerca de soporte técnico (bugs, consultas, etc.). Ayuda a incrementar la productividad y aumenta la satisfacción de los usuarios internos y externos.

El analista de Help Desk debe tener habilidades, conocimientos y capacidades, debe usar lógica y razonamiento para identificar las fortalezas y debilidades de soluciones y las posibles alternativas brindadas a los usuarios, sus conocimientos deben ser de software, hardware, comunicaciones, redes, Internet, correo electrónico, temas relacionados con tecnología informática, y capacidades como escuchar y comprender la información y las ideas expuestas en forma oral, aplicar reglas generales a problemas específicos para lograr respuestas con sentido.

ISO 9000

La serie de Normas ISO 9000 son un conjunto de enunciados, los cuales especifican que elementos deben integrar el Sistema de Gestión de la Calidad de una Organización y como deben funcionar en conjunto estos elementos para asegurar la calidad de los bienes y servicios que produce la Organización.

Las normas ISO 9000 han cobrado mayor relevancia internacional en la última década y en la actualidad es utilizada en más de 120 países. Estas normas requieren de sistemas documentados que permitan controlar los procesos que se utilizan para desarrollar y fabricar los productos. Estos tipos de sistemas se fundamentan en la idea de que hay ciertos elementos que todo sistema de calidad debe tener bajo control, con el fin de garantizar que los productos y/o servicios se fabriquen en forma consistente y a tiempo. Las ISO 9000 no definen cómo debe ser un Sistema de Gestión de Calidad de una organización, sino que ofrecen especificaciones de cómo crearlo e implementarlo; éste será diferente en función de las características particulares de la organización y sus procesos.

Las normas se revisan cada 5 años para garantizar la adecuación a las tendencias y dinámica del contexto mundial. En el año 2000 cobraron vigencia los cambios propuestos para las ISO 9000, los que se tradujeron en las actuales Normas ISO 9000 versión 2000.

Las normas ISO 9000:2000 quedaron conformadas por tres grandes apartados:

ISO 9000:2000, sistemas de gestión de calidad: Principios y vocabulario.

ISO 9001:2000, que trata sobre los requisitos de los Sistemas de gestión de calidad.

IT

La Tecnología Informática (IT), según lo definido por la asociación de la Tecnología Informática de América (ITAA), es el estudio, diseño, desarrollo e innovación, puesta en práctica de los sistemas informáticos computarizados, particularmente usos del software y hardware.

La informática abarca muchos aspectos referidos a la tecnología. Su universo puede ser grande, cubriendo muchos campos. Los profesionales realizan una variedad de deberes que se pueden extender desde instalar usos, diseñar redes de computadores hasta bases de datos complejas.

Algunos de los deberes que los profesionales, ingenieros y técnicos en Informática realizan, pueden incluir:

- Administración de datos
- Establecimiento de redes informáticas
- Diseño de las bases de datos
- Diseño del software
- Sistemas de información
- Administración de sistemas

Línea base

Las líneas base o más conocidas por su término en inglés, "baselines", son "una especificación de un producto que se ha revisado formalmente y sobre los que se ha llegado a un acuerdo, y que de ahí en adelante sirve como base para un desarrollo posterior que puede cambiarse solamente a través de procedimientos formales y controles de cambios"

Cuando en un proceso de desarrollo de software se realiza un producto (un documento, código fuente u otros), este producto pasa una etapa de producción en la que se puede modificar sin impedimentos. Sin embargo, una vez terminado este producto, tendrá una revisión formal y se aprobará por el equipo de calidad, en caso de que el cliente lo decida. Una vez que el producto ha sido aprobado, ya no podrá ser modificado de modo informal, sino que se deberá seguir un estricto control de cambios realizados sobre dicho producto para controlar correctamente su evolución.

El uso de las líneas base en un proyecto vendrá dado por la definición de las diferentes especificaciones que se realizarán a lo largo del proyecto (generalmente requisitos de usuario, software, diseño, código fuente, planes, procedimientos, pruebas y funcionamiento) de esta forma controlar cuando se aprueban los distintos productos y así asegurar que se realiza el correspondiente control de cambios que ayudará a mantener la coherencia y calidad de todo el proyecto software.

Mejora continua

La mejora continua, es una filosofía que intenta optimizar y aumentar la calidad de un producto, proceso o servicio. Es mayormente aplicada de forma directa a empresas de fabricación, debido en gran parte a la necesidad constante de minimizar costos de producción obteniendo la misma o mejor calidad del producto, porque como sabemos, los recursos económicos son limitados y en un mundo cada vez más competitivo a nivel de costos, es necesario para una empresa tener algún sistema que le permita mejorar y optimizar continuamente. Principalmente porque si se tiene un sistema de mejora entonces se tiene las siguientes características:

- Un proceso documentado: Esto permite que todas las personas que son partícipes de dicho proceso lo conozcan y todos lo apliquen de la misma manera cada vez.
- Sistemas de medición: Debe permitir determinar si los resultados esperados de cierto proceso se están logrando (indicadores de gestión)
- Participación: de todas o algunas personas relacionadas directamente con el proceso ya que son estas personas las que día a día tienen que lidiar con las virtudes y defectos del mismo.

Viéndolo desde este punto de vista, una de las principales ventajas de tener un sistema establecido de Mejora Continua es que todas las personas que participan en el proceso tienen la capacidad de opinar y proponer mejoras lo que permite que se identifiquen más con su trabajo, además se tiene la garantía que la fuente de información es de primera ya que quien plantea el problema y propone la mejora conoce el proceso y lo realiza todos los días.

Metodología cascada

También conocido como modelo clásico, modelo tradicional o modelo lineal secuencial. El método de cascada es considerado como el enfoque clásico para el ciclo de vida del desarrollo de sistemas, se puede decir que es un método puro que implica un desarrollo rígido. Ésta es una secuencia de actividades (o etapas) que consisten en el análisis de requerimientos, el diseño, la implementación, la integración y las pruebas.

- El análisis de requerimientos consiste en reunir las necesidades del producto y casi siempre su salida es texto.
- El diseño describe la estructura interna del producto y suele representarse con diagramas y texto.
- La implementación significa programación. Producto de esta etapa es el código en cualquier nivel, incluido el producido por sistemas de generación automática.
- La integración es el proceso de ensamblar las partes para completar el producto.

La metodología cascada se caracteriza por ordenar de manera rigurosa las etapas del ciclo de vida del software, dado que el comienzo de cada etapa debe esperar a la finalización de la inmediata anterior. Cuando la revisión determina que el proyecto no está listo para pasar a la siguiente etapa, permanece en la etapa actual hasta que esté preparado y debido a que el proceso está planeado es más fácil determinar costos y los plazos. Este modelo puede ser visto como un modelo con forma de cascada de agua con varios saltos, ya que cada salto representa cada una de las fases del ciclo de vida del proyecto.

Ofuscación

La ofuscación, es un proceso que encubre un pensamiento o un mensaje, haciéndolo difícil de comprender. Esta acepción es frecuente en el ámbito de la informática, con referencia a una modificación que se desarrolla sobre el código fuente de algún software para que resulte más complicado de leer.

Al buscar la ofuscación de un código, el objetivo es dificultar la tarea de los ingenieros a la hora de interpretarlo. Por eso resulta complejo acceder al código original tras un proceso de ofuscación.

Existen diversos motivos para decidir la ofuscación de un programa informático, el más habitual radica en la intención de proteger su código para evitar que otro se lo “robe” y lo utilice por su cuenta sin el crédito correspondiente a su programador. Para ello el creador suele añadir líneas falsas al código o recurrir a un sistema de encriptación que le permita encapsular todo su trabajo a fin de que nadie pueda comprenderlo.

En la era de la piratería y de los duplicados en internet es imprescindible que los creadores busquen nuevas alternativas para proteger sus trabajos e impedir que otros los utilicen en su beneficio sin que ellos lo sepan; para ello la ofuscación es una excelente alternativa, después de haber expuesto su código a la ofuscación, cualquier persona que intente leerlo se verá imposibilitada, ya que se encontrará con un código enrevesado e ilegible. Por eso será difícil que consiga plagiar el software en cuestión.

La ofuscación también es usada por delincuentes que ocultan aplicaciones maliciosas en el código, tal es así que muchos virus son capaces de entrar en nuestros computadores sin problema gracias a esa capacidad de pasar desapercibidos, los creadores, en este caso, se valen de la ofuscación para poner códigos dañinos que afecten el dispositivo en el que sean descargados.

Es importante aclarar que, al igual que ocurre con otras técnicas de protección la ofuscación debe ser realizada con cuidado y siempre realizando una copia de seguridad de todo lo trabajado hasta el momento en el código ya que si existiera un problema que impida volver al código fuente, puede que éste quedara ilegible incluso para el propio creador.

Open Source

La traducción de open source sería código fuente abierto, pero se prefiere utilizar el término en inglés, debido a su gran aceptación.

Open source no sólo significa acceso al código fuente, las condiciones de distribución de un programa open-source deben cumplir con el siguiente criterio:

- Libre redistribución: La licencia no debe restringir a nadie vender o entregar el software como un componente de una distribución de software que contenga programas de distintas fuentes, la licencia no debe requerir ningún tipo de cuota por su venta.
- Código fuente: El programa debe incluir el código fuente, y se debe permitir su distribución tanto como código fuente y compilado, cuando de algún modo no se distribuya el código fuente junto con el producto, deberá proveerse un medio conocido para obtener el código fuente sin cargo, a través de Internet.
El código fuente es la forma preferida en la cual un programador modificará el programa, no se permite el código fuente deliberadamente confundido (ofuscado). Tampoco se permiten formatos intermedios, como la salida de un preprocesador, o de un traductor.
- Trabajos derivados: La licencia debe permitir modificaciones y trabajos derivados, y debe permitir que estos se distribuyan bajo las mismas condiciones de la licencia del software original.
- Integridad del código: La licencia puede restringir la distribución de código fuente modificado sólo si se permite la distribución de "patch files" con el código fuente, con el propósito de modificar el programa en el tiempo de su construcción.
- La licencia debe permitir explícitamente la distribución de software construido en base a código fuente modificado, puede requerir que los trabajos derivados lleven un nombre o número de versión distintos a los del software original.

- No discriminar personas o grupos: La licencia no debe hacer discriminación de personas o grupos de personas.
- No discriminar campos de aplicación: La licencia no debe restringir el uso del programa en un campo específico de aplicación. Por ejemplo, no puede restringir su uso en negocios, o en investigación genética.
- Distribución de la licencia. Los derechos concedidos deben ser aplicados a todas las personas a quienes se redistribuya el programa, sin necesidad de obtener una licencia adicional. La licencia no debe ser específica a un producto, los derechos aplicados a un programa no deben depender de la distribución particular de software.
- Si el programa es extraído de esa distribución y usado o distribuido dentro de las condiciones de la licencia del programa, todas las personas a las que el programa se redistribuya deben tener los mismos derechos que los concedidos en conjunción con la distribución original de software.
- La licencia no debe contaminar otro software: La licencia no debe imponer restricciones sobre otro software que es distribuido junto con él. Por ejemplo, la licencia no debe insistir en que todos los demás programas distribuidos en el mismo medio deben ser software open-source.

Procesos

Un proceso es una secuencia de pasos dispuesta con algún tipo de lógica que se enfoca en lograr algún resultado específico. Los procesos son mecanismos de comportamiento que diseñan los hombres para mejorar la productividad de algo, para establecer un orden o eliminar algún tipo de problema.

El concepto puede emplearse en una amplia variedad de contextos, como por ejemplo en el ámbito jurídico, en el de la informática o en el de la empresa.

En informática, por otro lado, un proceso suele ser el nombre que recibe un programa que está siendo ejecutado en memoria, si vemos el administrador de tareas de Windows, podremos tomar conciencia de muchos de estos programas. En este caso, el nombre de procesos deriva del hecho de que un programa por definición lleva una secuencia de pasos de forma lógica. Muchos de estos procesos son fundamentales para el mantenimiento del sistema operativo, circunstancia que hace que eliminar a alguno pueda tener consecuencias negativas

Finalmente, desde el punto de vista de una empresa, un proceso da cuenta de una serie de acciones que se toman en el aspecto productivo para que la eficiencia sea mayor. En efecto, las empresas buscan continuamente aumentar su rentabilidad produciendo más y bajando sus costos.

Product Owner

El Product Owner, es el representante de todas las personas interesadas en los resultados del proyecto (internas o externas a la organización, promotores del proyecto y usuarios finales, idealmente también debería ser un usuario clave o consumidores finales del producto y actuar como interlocutor único ante el equipo, con autoridad.

El rol del Product Owner, puede venir de parte del cliente o dentro de la empresa misma, generalmente no se aconseja que el Product Owner sea parte también del equipo de desarrollo, debido a que sus intereses se pueden ver enfrentados, pero esto puede variar según el caso.

Las responsabilidades del Product owner (que puede ser interno o externo a la organización) son:

- Ser el representante de todas las personas interesadas en los resultados del proyecto internas o externas a la organización.
- Definir los objetivos del producto o proyecto.
- Dirigir los resultados del proyecto y gestionar los indicadores
- Es el propietario de la planificación del proyecto: crea y mantiene la lista de requerimiento y los prioriza para cubrir los objetivos del producto o proyecto.
- Reparte los objetivos/requerimientos en las iteraciones y establece un calendario de entregas.
- Antes de iniciar cada iteración re-planifica el proyecto en función de los requisitos que aportan más valor
- Colaborar con el equipo para planificar, revisar y dar detalle a los objetivos de cada iteración

Requerimientos:

En ingeniería del software y en el desarrollo de sistemas, un requerimiento es una necesidad documentada sobre el contenido, forma o funcionalidad de un producto o servicio. Los requerimientos son declaraciones que identifican atributos, capacidades, características y cualidades que necesita cumplir un sistema para que tenga valor y utilidad para el usuario.

Etapas de la fase de requerimientos:

- Obtención de requerimientos: búsqueda y obtención de los requerimientos desde los grupos de interés.
- Análisis: comprobación de la consistencia y completitud de los requerimientos
- Verificación: constatación de que los requerimientos especificados son correctos.

Clasificación de los requerimientos:

- Requerimientos funcionales: qué debe hacer el sistema o software.
- Requerimientos no funcionales: cómo debe funcionar el sistema o software (no su implementación), por ej. calidad, rendimiento, facilidad de uso, etc.
- Requerimientos externos: a qué se debe atender el sistema o software con respecto a su entorno, compatibilidad con otros sistemas, adecuación a determinadas leyes, etc.

Características que deberían cumplir los requerimientos:

- Actual: el requerimiento no debe volverse obsoleto con el paso del tiempo.
- Cohesión: el requerimiento debe dirigirse a solo una única cosa.
- Completo: el requerimiento debe estar completamente declarado en un único lugar, sin información faltante.
- Consistente: el requerimiento no debe contradecir ningún otro requerimiento y debe ser completamente consistente con toda la documentación.

- Correcto/necesario: el requerimiento debe cumplir con la necesidad declarada por los interesados en el sistema/software.
- Factible/viable: el requerimiento debe poder ser implementado.
- No ambiguo: el requerimiento debe estar concisamente declarado, debe expresar hechos objetivos, no opiniones subjetivas. debe poder ser interpretado de una única manera.
- Obligatorio: el requerimiento debe representar una característica definida por el grupo interesado en el desarrollo del sistema, su ausencia no puede ser reemplazada.
- Verificable/demostrable: La implementación del requerimiento debe poder ser resuelta en alguno de estos cuatro métodos: inspección, análisis, demostración o prueba.

Balanced Scorecard

Es una herramienta de gestión que traduce la estrategia de la empresa en un conjunto coherente de indicadores. La visión y la estrategia de negocios dictan el camino hacia el que deben encaminarse los esfuerzos individuales y colectivos de una empresa. La definición de estrategias por naturaleza es complicada, pero la implementación de la misma representa el mayor obstáculo en la mayoría de las ocasiones. El reto corresponde en identificar exactamente lo que debe monitorearse para comunicar en todos los niveles de la empresa, y verificar si se están alcanzando las estrategias a través de acciones muy puntuales.

El acceso a los principales almacenes de información brinda la posibilidad de presentar los resultados de desempeño y entender por qué se están dando esos resultados.

El Balanced Scorecard induce una serie de resultados que favorecen la administración de la compañía, pero para lograrlo es necesario implementar una metodología y aplicación para monitorear y analizar los indicadores obtenidos del análisis. Entre otros podemos considerar las siguientes ventajas:

- Alineación de los empleados hacia la visión de la empresa.
- Comunicación hacia todo el personal de los objetivos y su cumplimiento.
- Redefinición de la estrategia en base a resultados.
- Traducción de la visión y estrategias en acción.
- Favorece en el presente la creación de valor futuro.
- Integración de información de diversas áreas de negocio.
- Capacidad de análisis.
- Mejoría en los indicadores financieros.
- Desarrollo laboral de los promotores del proyecto.

Scrum Master

El Scrum Master (SM) o facilitador de proyectos, es la figura que lidera los equipos en la gestión ágil de proyectos. Su misión es que los equipos de trabajo alcancen sus objetivos hasta llegar a la fase de “sprint final”, eliminando cualquier dificultad que puedan encontrar en el camino.

Como facilitador de proyectos, es el encargado de sacar adelante todos aquellos proyectos que utilicen una metodología Scrum: desde la elaboración del Product Backlog (el archivo que recoge las tareas y funciones a desarrollar), y Sprint Backlog (documento que muestra la división de tareas entre los miembros del equipo), el Sprint (en donde se realizan todas las acciones y donde se prueba si las acciones realizadas funcionan) hasta el Burn Down (el análisis y control de las tareas realizadas y todo lo que quedó pendiente).

Muchos Scrum Master han desempeñado con anterioridad el papel de Project Managers, por lo que no solo facilitan las tareas al resto del equipo, sino que en muchas ocasiones también ayuda a encontrar soluciones a los problemas.

Principales características de un Scrum Master:

- Elimina cualquier impedimento con el que se encuentre el equipo para conseguir sus objetivos finales.
- Si los desarrolladores no saben cómo abordar las tareas, el Scrum Master los juntará a todos para explicarles en qué consisten y qué tarea abordará cada uno.
- Facilita la fase de sprint final al equipo. Cuando los miembros del equipo están presentando a los stakeholders el proyecto, evita que no se atasquen en el sprint final, actuando como guía y moderador de las presentaciones.
- Ayuda a llevar a cabo los daily standups, esto es, proporcionar todas las actualizaciones que el equipo necesita para el desarrollo de los proyectos.
- A veces puede realizar las tareas de un agile coach. Es decir, se asegura de que todos los miembros del equipo aprendan y utilicen la metodología adecuada.

- Trabaja codo con codo con el Product Owner y que desde un primer momento define los objetivos del proyecto, detecta riesgos que se puedan detectar durante la fase de sprint y busca actualizaciones para las tareas a desarrollar.
- La figura del Scrum Master refuerza así la idea de que esta metodología mejora el trabajo en equipo, centrando todos los esfuerzos en conseguir un mismo objetivo para satisfacer las necesidades de los stakeholders.

Stakeholders

Stakeholders es una palabra en inglés que, en el ámbito empresarial, significa 'interesado' o 'parte interesada', y que se refiere a todas aquellas personas u organizaciones afectadas por las actividades y las decisiones de una empresa.

En toda organización, además de sus propietarios, participan diversos actores claves y grupos sociales que están constituidos por las personas o entes que, de una manera y otra, tienen interés en el desempeño de una empresa porque están relacionadas, directa e indirectamente, con ella.

En estos grupos podemos contar a los empleados, clientes, proveedores, accionistas, inversores, entes públicos, organizaciones no gubernamentales, sindicatos, organizaciones civiles, la comunidad y la sociedad en general.

El término stakeholder fue acuñado por primera vez por R. Edward Freeman en su libro *Strategic Management: A Stakeholder Approach*, publicado en 1984, en el cual su autor sostenía que estos grupos de interés son un elemento esencial, los que deben ser tomado en cuenta en la planificación estratégica de los negocios.

Así, el éxito o el fracaso de una empresa afecta o concierne no solo a sus dueños, sino que también a los trabajadores, familias, proveedores, competidores, la comunidad donde se encuentra inserta, entre otros.

Existen dos categorías fundamentales de stakeholders.

- I. Los stakeholders primarios, que son aquellos imprescindibles para el funcionamiento de la organización, es decir, todos aquellos que tienen una relación económica directa con la empresa, como los accionistas, los clientes o los trabajadores.
- II. Los stakeholders secundarios, que son aquellos que no participan directamente en las actividades de la empresa, pero que, sin embargo, se ven afectados por ella.

Sprint de entregables:

En Scrum un proyecto se ejecuta en bloques temporales cortos y fijos (iteraciones de un mes natural y hasta de dos semanas).

Cada iteración tiene que proporcionar un resultado completo, un incremento de producto que sea potencialmente entregable, de manera que cuando el cliente solicite un requerimiento sea necesario un esfuerzo mínimo para que el producto esté disponible para ser utilizado. Para ello, durante la iteración el equipo colabora estrechamente y se llevan a cabo las siguientes dinámicas:

- Cada día el equipo realiza una reunión de sincronización, donde cada miembro inspecciona el trabajo de los otros para poder hacer las adaptaciones necesarias, comunica cuales son los impedimentos con que se encuentra
- El Facilitador (Scrum Master) se encarga de que el equipo pueda cumplir con su compromiso y de que no se merme su productividad.
- Elimina los obstáculos que el equipo no puede resolver por sí mismo.
- Protege al equipo de interrupciones externas que puedan afectar su compromiso o su productividad.

BIBLIOGRAFIA

1. Adkins, L. (2013). *Coaching Agile Teams*. Addison-wesley.
2. Agile-Spain, C. (s.f.). *Por Un Scrum Popular*. Obtenido de <https://agile-spain.org/>.
3. Bartolotti, S. (s.f.). *Iebschool*. Obtenido de <http://www.iebschool.com/blog/9-herramientas-para-la-gestion-agil-de-proyectos-agile-scrum/>
4. Eduardo, V. (2010). *Aplicando La Metodología Ágil Lean Start-up*. Editorial académica española.
5. *Experiencia Y Metodologia Para Modelo Desarrollo* . (2002). Popular, s.a.
6. Fernandez, J. C. (s.f.). <https://es.slideshare.net>. Obtenido de slideshare.: <https://es.slideshare.net/jofese/metodologas-giles-y-metodologias-web-apra-el-desarrollo-de-software>
7. Galen, R. (2013). *Scrum Product Ownership*. Llc Rgcg.
8. García Álvarez, C. L. (2012). *Métodos Ágiles y Scrum*. España: Anaya Multimedia.
9. Gómez, J. (s.f.). *Ecured*. Obtenido de https://www.ecured.cu/Metodolog%C3%ADas_Agiles
10. González, J. F. (2013). *Introducción a las metodologías ágiles*. UOC. Obtenido de <https://openlibra.com/es/book/introduccion-a-las-metodologias-agiles>
11. Grifol, D. (s.f.). *metodologias-agiles*. Obtenido de <http://danielgrifol.es/tag/metodologias-agiles/>
12. Henrik, K. (2014). *Scrum and XP from the Trenches*. Enterprise Software Development.
13. scrumalliance. (s.f.). *scrumalliance*. Obtenido de <http://www.scrumalliance.org>
14. Viteri, S. (s.f.). *Blog de Sergio Viteri*. Obtenido de <https://sergioviteri.com/tag/metodologias-agiles/>
15. *Wikipedia*. (s.f.). Obtenido de <http://www.wikipedia.com>