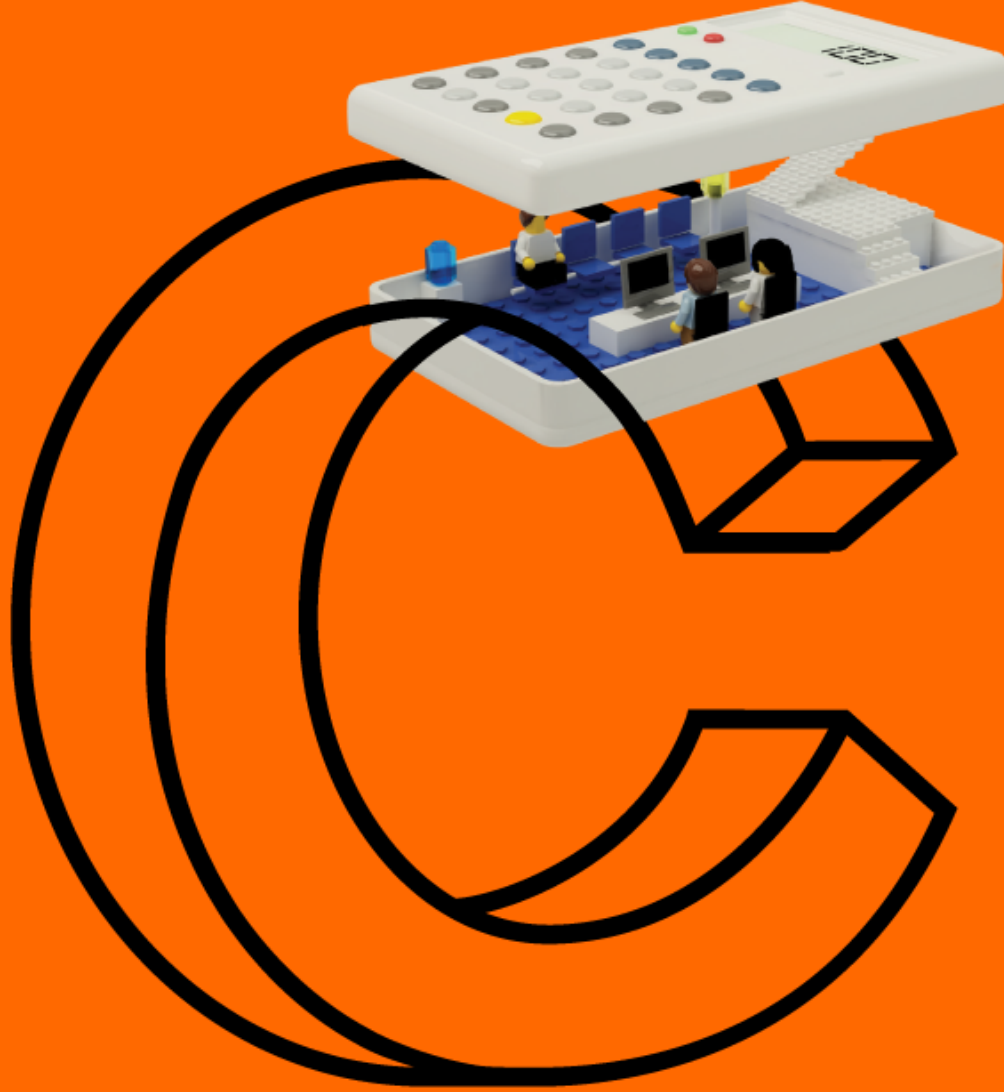


Análisis y Diseño de Sistemas Orientado a Objetos

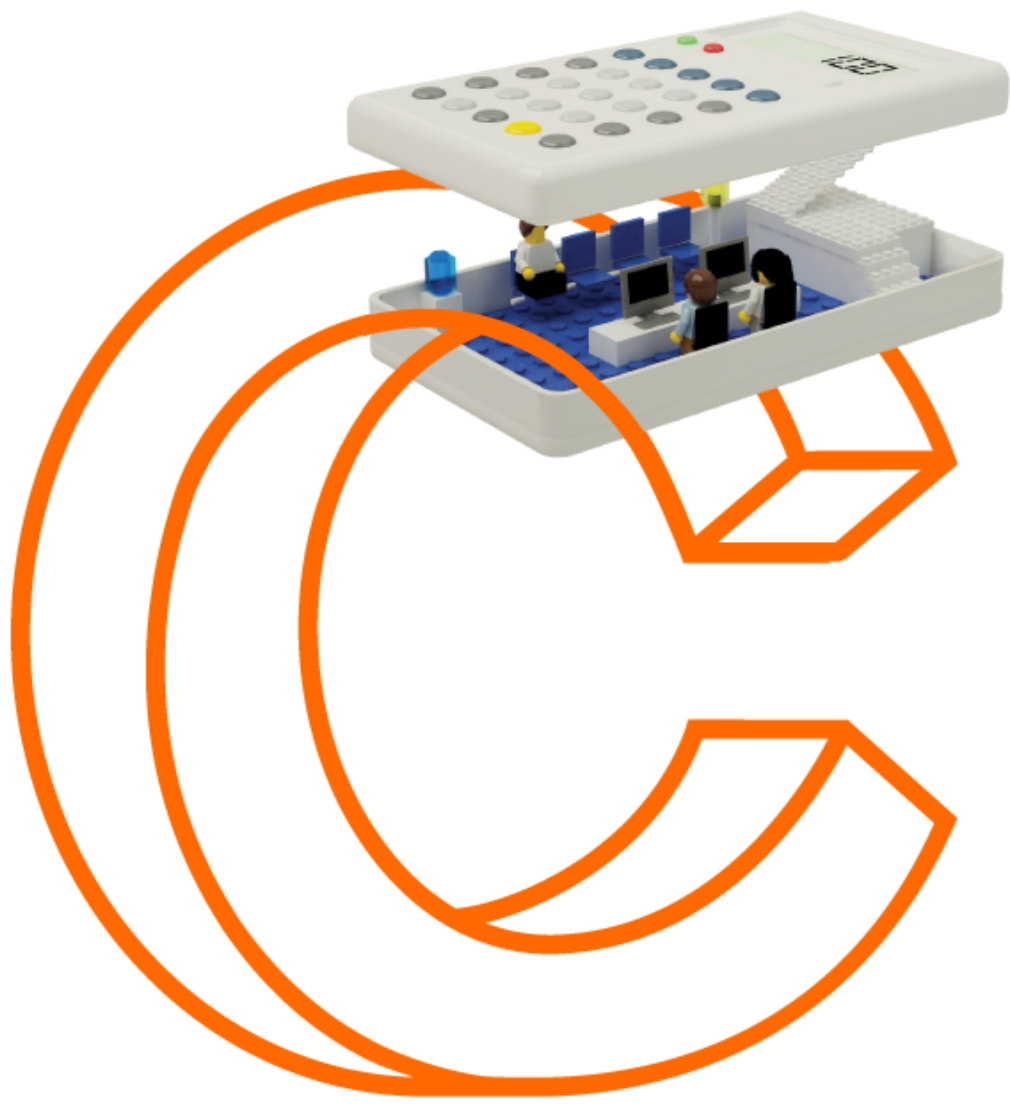


Mg. Daniel Arias, PMP™ y Scaled Scrum Master™

Metodología de desarrollo de software

- Metodología Ágiles

Semana 3



Semana 3: Metodología de desarrollo de software - Metodología Agiles

Logro de aprendizaje

- Framework Scrum
- XP
- Kanban
- Scrumban
- Lean

Actividades

Actividad 03:

- Elabora y presenta un Cuadro Comparativo de cada metodología Agil



Aprendemos:

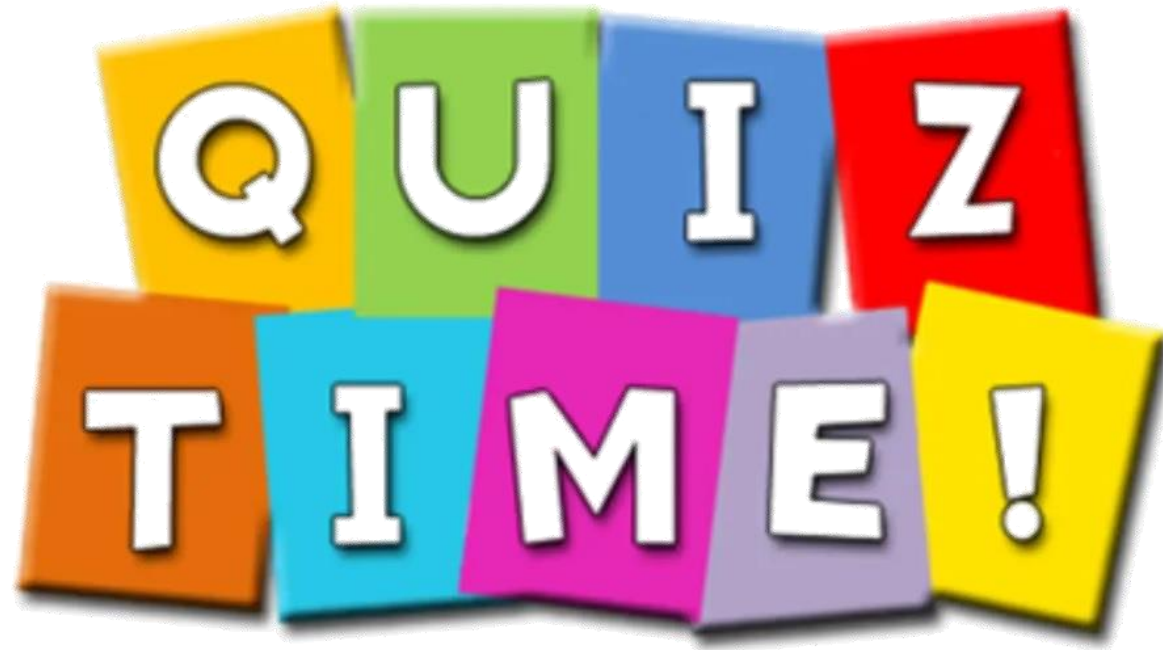
¿Qué aprendimos la clase pasada?

Herramienta Case

- Concepto
- Objetivo
- Tipos



Conversemos

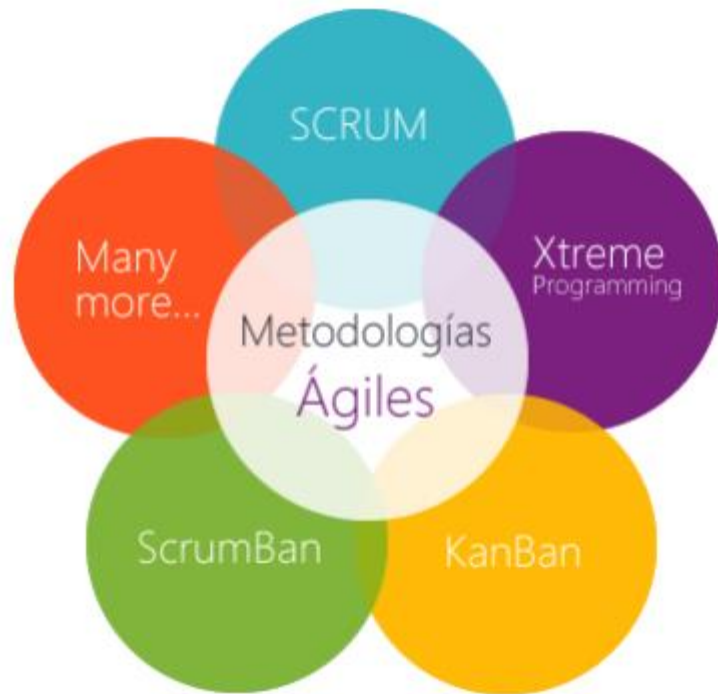


<http://www.menti.com>



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Qué son las metodologías ágiles y su aplicación en el mercado actual

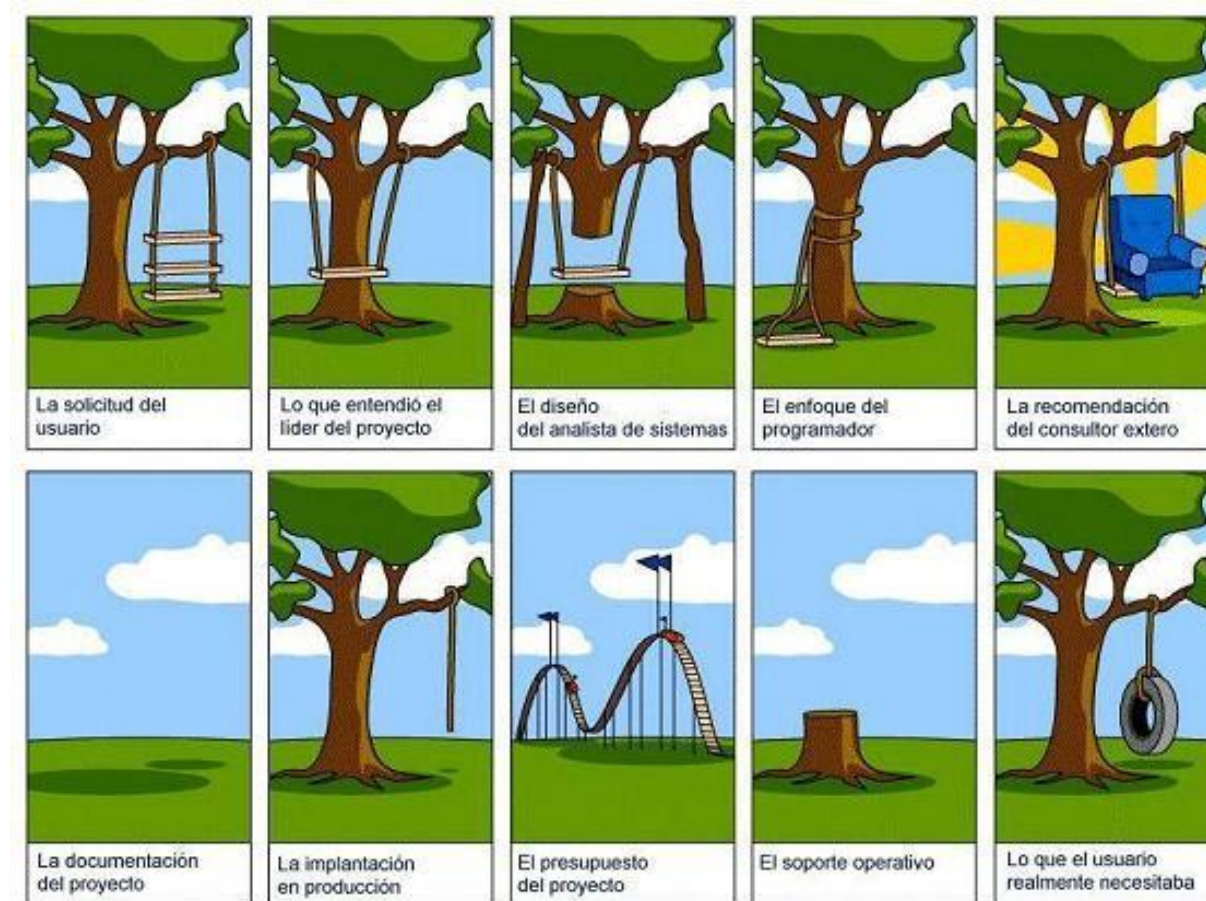


NO soy un gurú de las metodologías ágiles, sólo me adapto.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Ciclo de Vida “tradicional” o en cascada



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Llega el cambio

El negocio cambia. La tecnología cambia. El equipo cambia.

El problema no es el cambio en sí, porque el cambio va a ocurrir; el problema es la incapacidad para adaptarse al cambio cuando llega.

eXtreme Programming

Estamos descubriendo mejores maneras
de hacer software

Firmantes del Manifiesto Ágil 2001



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Metodología Ágil

La metodología Agile es una fórmula para el desarrollo de proyectos que necesitan **rapidez y flexibilidad para adecuarse a las necesidades del cliente**, y siempre está enfocada a mejorar resultados. Aplicar el enfoque Agile a los procesos internos de una empresa es mucho más simple de lo que parece, toda la filosofía viene explicada en el manifiesto ágil que crearon sus fundadores.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Metodología Ágil

En la metodología Agile no se planifica ni se diseña el producto por adelantado, **evoluciona** basándose en bucles de retroalimentación. Se trabaja por períodos (denominados sprints), que pueden ser de dos o cuatro semanas, y durante ese tiempo cada miembro del equipo debe ejecutar una serie de tareas. Al final del sprint, se entregan los avances y se comienza de nuevo el proceso. De este modo, el cliente **va recibiendo poco a poco novedades** y puede ir **marcando prioridades o introduciendo cambios**.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Beneficios de trabajar en Agile

1. **Permite poner el foco en las personas.** En muchas empresas se da más importancia a los procesos y las herramientas que a las mismas personas. Sin embargo, tener empleados motivados, en un entorno adecuado y que sientan que son apoyados redunda en unos mejores resultados.
2. **Fomenta la responsabilidad dentro del equipo y aumenta la autonomía y la transparencia.** El trabajo colaborativo, la fluidez en la comunicación de los equipos y la participación equitativa de todas las partes hacia un objetivo común, claro y conciso ayuda a que los equipos sean más eficaces y autónomos.
3. **Permite dimensionar mejor los proyectos minimizando los riesgos.** Las revisiones continuas permiten la adaptación al cambio de una manera más eficiente, que evita sorpresas tanto para el equipo como para el cliente respecto a los tiempos de entrega y los costes.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Beneficios de trabajar en Agile

4. **Facilita la priorización y toma de decisiones.** Al estar todas las personas involucradas con un ritmo de trabajo constante y evitarse las jerarquías, la operativa es más rápida maximizándose los resultados buscando la eficiencia y optimización.
5. **Mejora la experiencia del cliente.** A través de feedbacks rápidos y precisos, gracias a un modelo de comunicación más colaborativo y cercano, donde el cliente es uno más del equipo, se consigue mayor nivel de satisfacción y ahorro de tiempo.
6. **Entrega de productos y servicios en intervalos con rápida respuesta a los cambios.** La implantación de Agile como metodología proporcionan una estructura flexible que permite entregar diferentes versiones en muy corto plazo de tiempo.
7. **Incrementa el valor añadido y la predictibilidad de los resultados.** Este modelo de trabajo y la cercanía con el cliente permite tener mayor conocimiento del mismo, aportando valor diferencial y evitando desviaciones sobre Plan.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Por ejemplo, si vamos a construir un puente, debemos tener claro todo el diseño antes de empezar a ejecutar, no podemos ir a un modelo de “ir probando” y mientras levantas un pilar decidir con el cliente como quiere seguir.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Valores Ágiles



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Aplicar Agile conlleva cumplir estos 12 principios:

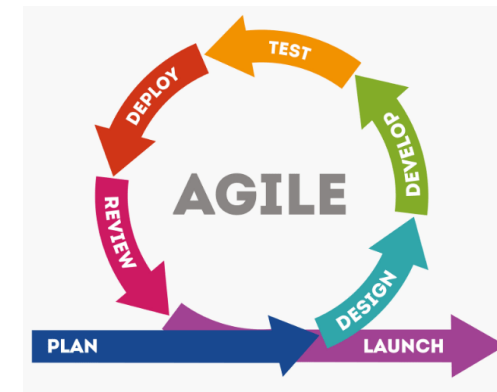
1. La prioridad es que el cliente esté satisfecho y siempre informado del estado del proceso.
2. Los requisitos del proyecto pueden cambiar y no se verá como un problema, sino como una ventaja competitiva.
3. Las entregas se realizan periódicamente y en periodos cortos. La planificación se realizará desde las dos semanas, a los dos meses.
4. El equipo debe trabajar de forma conjunta y coordinada.
5. Es prioritario motivar al equipo, confiar en los miembros y proporcionarles los recursos o apoyos que necesiten.
6. Las reuniones Scrum son el método más efectivo para comunicarse.
7. El éxito depende de si el producto final funciona y es satisfactorio.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Aplicar Agile conlleva cumplir estos 12 principios:

8. Los procesos deben ser sostenibles, tanto en recursos materiales, como en la gestión del tiempo y el ritmo de trabajo.
9. En todo proceso o etapa debe prevalecer la excelencia técnica.
10. Prevalece la ley de la simplicidad: menos, es más.
11. La organización de los equipos es esencial para dar con un buen diseño.
12. Los tiempos para la reflexión y buscar mejoras es necesario e igual que importante que el resto de las fases.

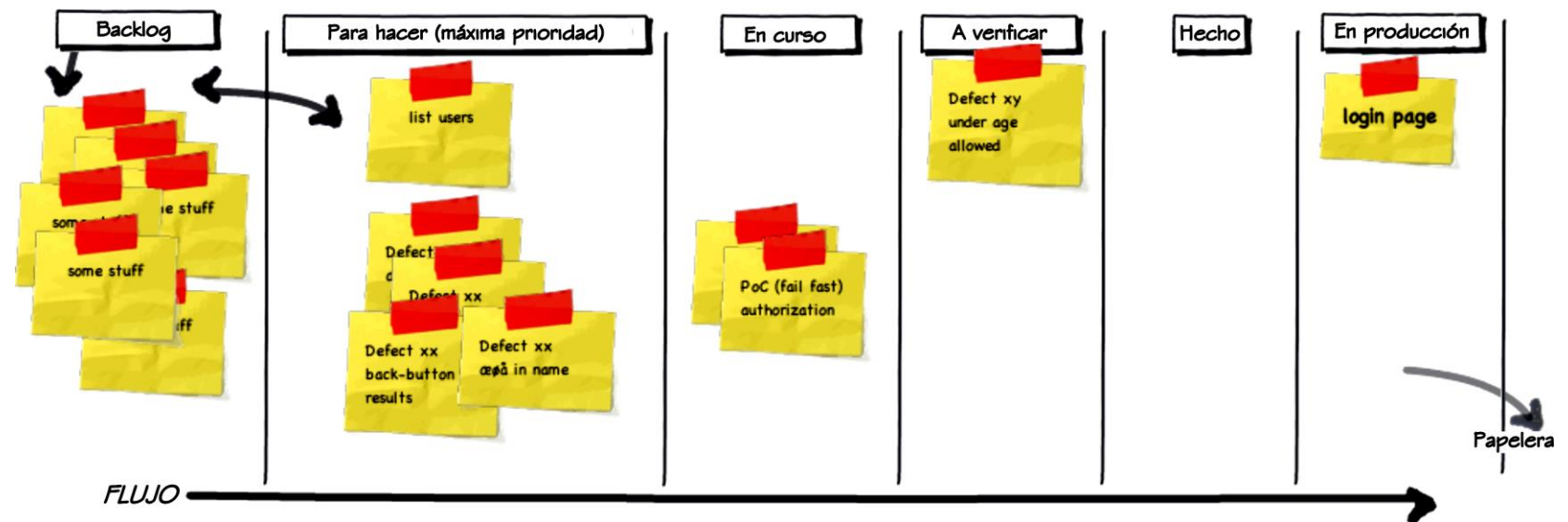


Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Conceptos

- **Tablero Kanban**

La pizarra tiene tantas columnas como estados por los que puede pasar la tarea. Por ejemplo, “en espera de ser desarrollada”, “en análisis”, “en diseño”, etc.



Conceptos

- **Limitar el trabajo en progreso**

Una de las principales ideas del Kanban es que el trabajo en curso (**Work In Progress**) debería estar limitado, es decir, que el número de tareas que se pueden realizar en cada fase debe ser algo conocido.

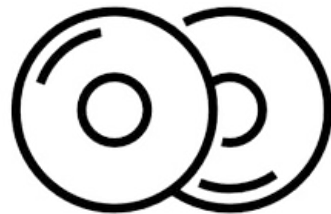
Work in Progress
Limits revisited



Conceptos

- **Minimum Viable Product**

El Minimum Viable Product (MVP) o Producto Mínimo Viable (PMV) es la versión mínima de un producto, tal que nos permita recolectar la mayor cantidad de información de nuestro mercado y clientes con el menor esfuerzo posible. Consiste en hacer foco en las características mínimas y necesarias para que el producto pueda lanzarse al mercado.



Prototype



MVP



Product



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Conceptos

Esto nos permitirá:

- Evitar crear productos que nadie necesita
- Maximizar el aprendizaje por dólar invertido

El MVP es una estrategia de Lean Startup que apunta a acercarnos a nuestros clientes con la menor inversión posible (tiempo/dinero) y con ello determinar si nuestro producto es o no es viable.

You **MUST** do it like this!



You **DON'T** do it like this!



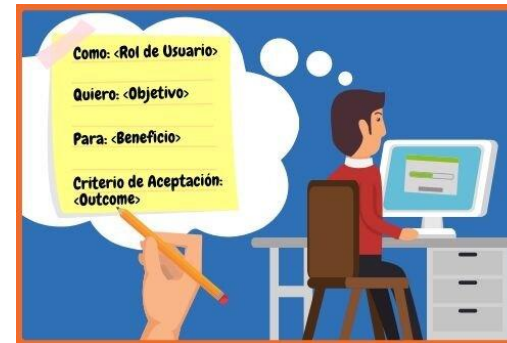
Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Conceptos

- **Historias de Usuario**

Las historias de usuario son descripciones cortas y simples de una característica contada desde la perspectiva de la persona que desea la nueva capacidad, generalmente un usuario o cliente del sistema. Por lo general, siguen una plantilla simple:

Como <Usuario>
Quiero <algún objetivo>
Para que <motivo>



Las historias de usuario a menudo se escriben en fichas o notas adhesivas, se almacenan en una caja y se organizan en paredes o mesas para facilitar la planificación y el debate. Como tal, cambian fuertemente el enfoque de escribir sobre las características a discutir. De hecho, estas discusiones son más importantes que cualquier texto que se escriba.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Conceptos

- **Tarjetas CRC**

Las tarjetas CRC se idearon para enseñar orientación a objetos.

CRC es el acrónimo de “Class Responsibility Collaborator”. Una “Clase” representa objetos similares, la “Responsabilidad” es algo que una clase conoce o hace y un “Colaborador” es otra clase con la que interactúa para poder cumplir sus responsabilidades.

Para obtener las clases se utilizan tarjetas, cada tarjeta será una clase. Cada tarjeta tiene escrito el nombre de la clase su responsabilidad y colaboraciones.

Cliente

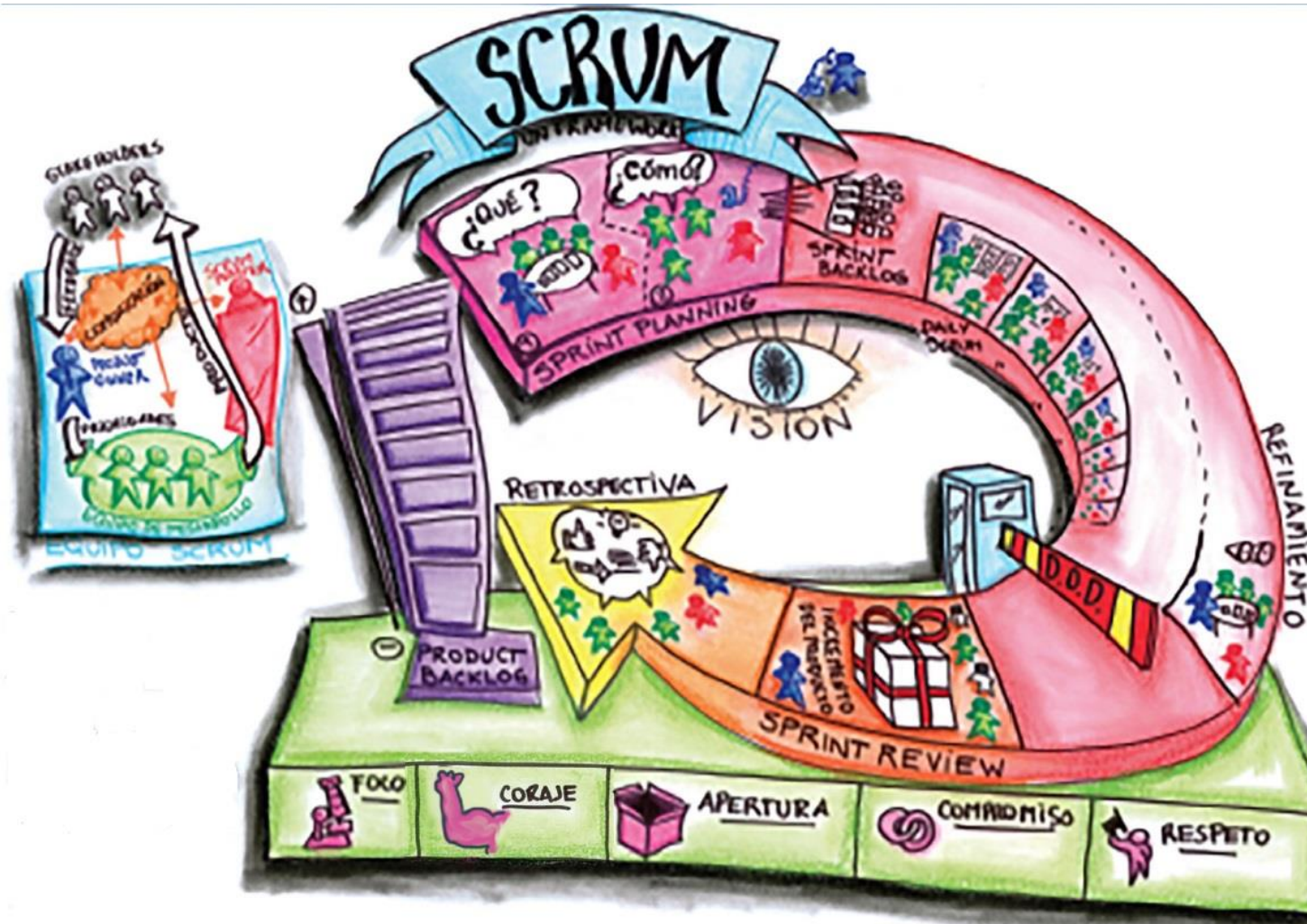
Nombre: Cliente	Id Cliente	Tipo Publico
Descripción En esta clase se ingresan datos personales e incluso se podrán editar o eliminar.	Casos de usos Asociados Administrador, Mascota, Negocio	
Responsabilidades Su Responsabilidad en el sistema es muy importante ya que es aquí donde empieza la función del sistema, El cliente es el que hace que este sistema tenga una función	Colaboradores Administrador	

Atributos (tipo): Privados

Relaciones: Asociación, Generalización, Agregación



Aprendemos: Metodología de desarrollo de software – Metodología Ágil



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Framework Scrum

- Scrum es un marco de trabajo utilizado para desarrollar productos complejos.
- Scrum es un conjunto de buenas prácticas y reglas que proveen la estructura necesaria para ayudar al equipo a reducir la complejidad técnica, comercial e interpersonal del desarrollo del software en su conjunto y que, también capacita a equipos auto organizados para alcanzar sus objetivos.
- El enfoque de Scrum para el desarrollo de productos es iterativo e incremental con múltiples bucles de retroalimentación.

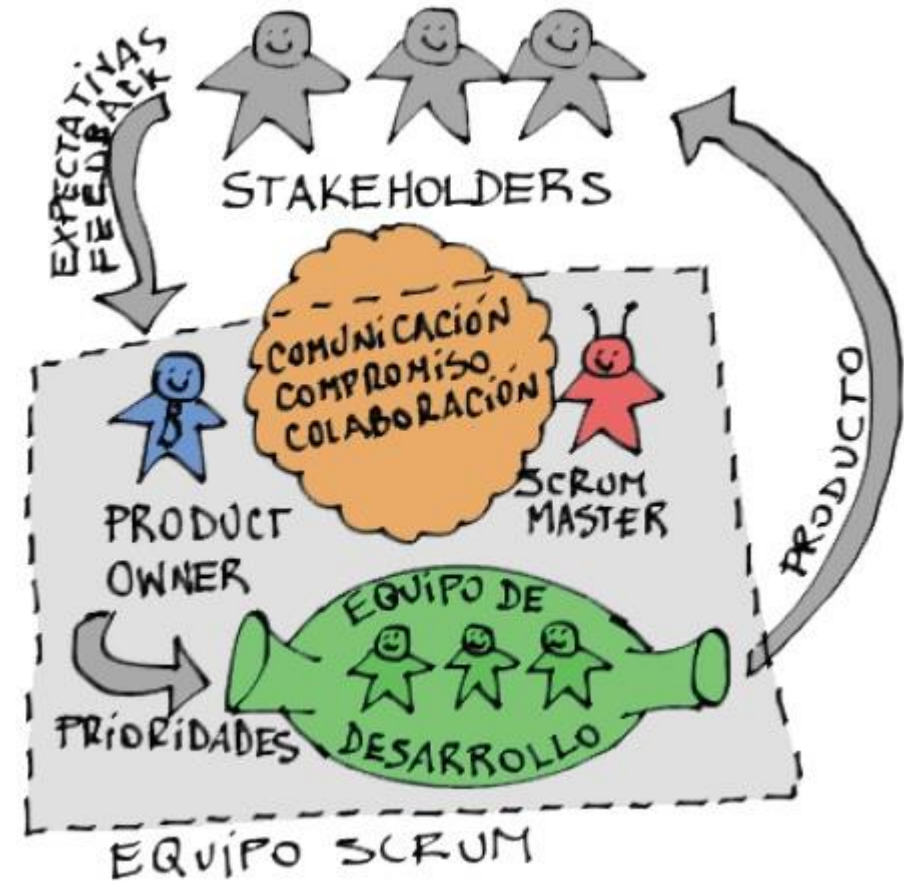


Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Personas y roles en SCRUM

Un Equipo Scrum se compone de tres roles:

- Product Owner,
- Equipo de Desarrollo y
- Scrum Master.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Product Owner

El Product Owner es la persona responsable de maximizar el valor, tanto del producto como del trabajo realizado por el Equipo de Desarrollo.



Su trabajo se refleja, principalmente, en la gestión del Product Backlog. Esto incluye:

- Asegurar la claridad de los ítems del Product Backlog
- Asegurar la visibilidad, transparencia y comprensión del mismo por parte de todos los involucrados



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Product Owner (Cont.)

- Ordenar los Ítems del Product Backlog para lograr los objetivos de la mejor manera posible
- Asegurar que los miembros del Equipo de Desarrollo comprenden los Ítems del Product Backlog al nivel que sea requerido

El PO puede delegar este trabajo en el Equipo de Desarrollo, sin embargo, el PO sigue siendo el responsable.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Equipo de Desarrollo

El Equipo de Desarrollo está formado por todos los individuos necesarios para la construcción de un incremento “terminado” de producto al final de cada Sprint. Es el único responsable por la construcción del producto.



Solo el PO tiene la autoridad para establecer el conjunto de requerimientos para el Equipo de Desarrollo. Es el Equipo de Desarrollo quien descompondrá esos requerimientos en tareas, como parte de su auto-organización.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Equipo de Desarrollo (Cont.)

El Equipo de Desarrollo es auto-organizado. Esto significa que no existe un líder externo que asigne las tareas ni que determine la forma en la que serán resueltos los problemas. Nadie, ni siquiera el SM, tiene autoridad para decirle al ED la forma en la que debe hacer su trabajo. Es el mismo equipo quien determina la forma en que realizará el trabajo y cómo resolverá cada problemática que se presente. La contención de esta auto-organización está dada por el objetivo a cumplir: entregar un incremento de producto “terminado” al final de cada Sprint, que represente los requerimientos comprometidos al principio del Sprint.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Equipo de Desarrollo (Cont.)

El ED es multifuncional. Posee todas las habilidades necesarias para construir un incremento de producto “terminado”. Dentro del equipo de desarrollo no existen especialistas exclusivos, sino más bien individuos generalistas con capacidades especiales. Lo que se espera de un miembro del ED es que no solo realice las tareas en las cuales se especializa sino también todo lo que esté a su alcance para colaborar con el éxito del equipo.

En el ED no existen títulos ni rangos jerárquicos: **todos son desarrolladores**, sin importar el tipo de trabajo que realice.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Equipo de Desarrollo (Cont.)

Tampoco hay sub-equipos, al margen de la cantidad de dominios de actividades que sea necesario realizar (ejemplo: prueba, análisis, arquitectura).

Aunque los miembros del ED tengan habilidades enfocadas en diferentes aspectos de la construcción del producto, la responsabilidad corresponde al equipo como un todo.

El tamaño del Equipo de Desarrollo debe ser lo suficientemente pequeño para mantenerse ágil y lo suficientemente grande para construir un incremento de producto significativo en cada Sprint.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Equipo de Desarrollo (Cont.)

El número buscado es de no menos de tres y no más de nueve personas. Con menos de tres personas, son altas las posibilidades de no contar con todas las habilidades necesarias. Con más de nueve personas el esfuerzo de coordinación se vuelve grande y complejo para un proceso de control empírico como Scrum. El ED tiene tres responsabilidades tan fundamentales como indelegables:

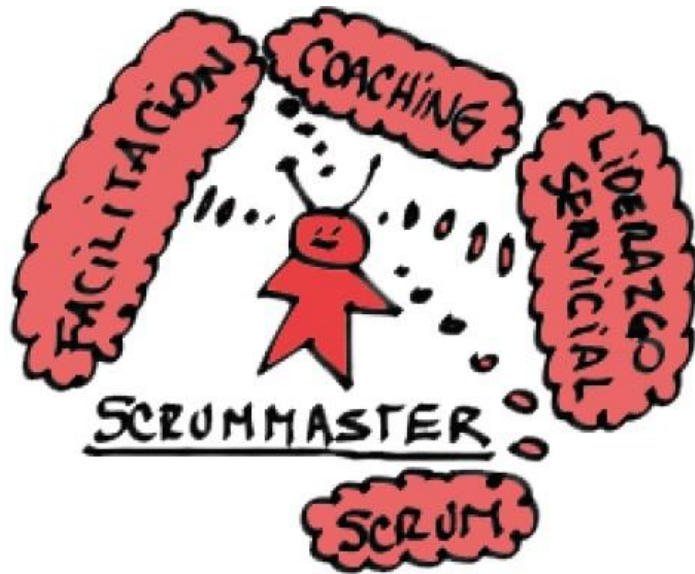
- Proveer las estimaciones de cuánto esfuerzo será requerido para cada una de las características del producto.
- Comprometerse al comienzo de cada Sprint a construir un determinado incremento de producto para cumplir con el objetivo buscado.
- Es responsable por la entrega del producto terminado al finalizar cada Sprint.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Scrum Master

La principal función del Scrum Master es asegurar el entendimiento y seguimiento de Scrum tanto por el Equipo Scrum como por los profesionales con los que este equipo se relaciona.



Al mismo tiempo, el SM es el Coach del equipo y es quien lo ayuda a alcanzar su máximo nivel de productividad posible.

Tomando algunas referencias de Leonardo Wolk¹ podemos decir que el SM, en tanto que coach, es un *líder, facilitador, provocador, detective y soplador de brasas*.

¹ Leonardo Wolk, *Coaching – El arte de soplar brasas*, 2003, p. 22-23



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Scrum Master (Cont.)

Líder por ser un ejemplo a seguir, **facilitador** por fomentar contextos de apertura y discusión donde todos pueden expresar sus opiniones y lograr consensos comunes, **provocador** por desafiar las estructuras rígidas y las antiguas concepciones sobre cómo deben hacerse las cosas, **detective** por involucrarse activamente en la búsqueda e identificación de indicios y pistas en la narrativa del equipo y los individuos y finalmente, **soplador de brasas**, *“un socio facilitador del aprendizaje, que acompaña al otro en una búsqueda de su capacidad de aprender para generar nuevas respuestas”*.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Scrum Master (Cont.)

Se espera, además, que el SM acompañe al equipo de trabajo en su día a día y garantice que todos, incluyendo al PO, comprendan y utilicen Scrum de forma correcta.

Las responsabilidades principales del SM son:

- Velar por el **correcto empleo** y **evolución** de Scrum
- Facilitar el **uso de Scrum**.
- Asegurar que el equipo de desarrollo sea **multi-funcional** y **auto-organizado**.
- **Proteger** al equipo de desarrollo de distracciones y trabas externas al proyecto



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Scrum Master (Cont.)

- Detectar, monitorear y **facilitar la remoción de los impedimentos** que puedan surgir con respecto al proyecto y a la metodología
- Asegurar la **cooperación** y **comunicación** dentro del equipo
- Acompañar al PO de forma que éste aprenda y pueda:
 - Gestionar el Product Backlog de una forma eficiente
 - Maximizar el valor mediante la priorización
 - Entender y practicar la agilidad
 - Comprender la planificación del producto en un contexto empírico



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Scrum Master (Cont.)

- Acompañar al ED de forma que éste aprenda y pueda:
 - Crear productos de alto valor para el negocio
 - Remover impedimentos que estén a su alcance
 - Enfocarse en su trabajo durante el Sprint

Además de estas cuestiones, el SM debe detectar **problemas** y **conflictos interpersonales** dentro del equipo de trabajo. Para respetar la filosofía auto-organizativa del equipo, lo ideal es que el equipo mismo sea quien resuelva estas cuestiones. En el caso de no poder hacerlo, deberá involucrarse al SM y eventualmente a niveles más altos de la gerencia.

El Scrum Master es un Líder Facilitador



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Valores de Scrum

- **Enfoque:** Enfocarse en entregar primero al cliente un producto con las características de mayor valor.
- **Apertura:** Estar abierto a escuchar los feedbacks de todas las partes interesadas en el proyecto con información sobre el trabajo.
- **Respeto:** Respeto por las personas y su libertad de expresión (Scrum enfatiza el respeto por las personas. El mayor beneficiario de esto son los miembros del equipo auto organizado de desarrollo, que ayudan a construir y entregar un producto de software de calidad).
- **Coraje:** Enfrentar el statu quo, el mando de arriba hacia abajo y la jerarquía.
- **Compromiso:** Compromiso con el cliente a través de productos valiosos y de calidad.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

¿Qué hace que Scrum sea exitoso?

- Scrum es simple de entender y proporciona la estructura necesaria para hacer frente a los retos que se presentan durante el desarrollo de software.
- Proviene de equipos auto organizados, compuestos por personas reales y no de una hipótesis o investigación académica.
- Scrum ayuda a los equipos a mejorar su capacidad de adaptación a los rápidos cambios del entorno empresarial.
- Es especialmente adecuado para ser aplicado en entornos de desarrollo de productos complejos y de alto riesgo.
- Ayuda a mejorar el ROI.
- Mejora la alineación del desarrollo de producto con las necesidades del cliente.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

¿Qué hace que Scrum sea exitoso?

- Optimiza la traducción de ideas a productos.
- Mediante el control de costos que crea una sutil presión para entregar los productos, Scrum se beneficia a partir del sentido de urgencia de hacer las cosas.
- Scrum consigue que los equipos se centren en ofrecer el mayor valor para el cliente en cada Sprint.
- La mayoría de los elementos de Scrum son linealmente escalables y pueden aplicarse a entornos de desarrollo de productos a gran escala.
- Scrum identifica y elimina los obstáculos para mejorar el ritmo de desarrollo del producto. El Scrum Master se hace responsable de asegurar que los impedimentos sean eliminados.
- Scrum aumenta la velocidad de la toma de decisiones al empoderar al equipo.
- Fomenta equipos auto organizados y multifuncionales.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

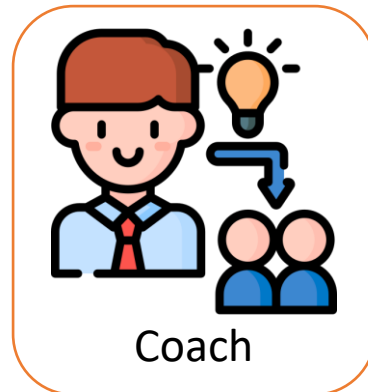
eXtreme Programming (XP)

- La metodología XP o Programación Extrema es una metodología ágil y flexible utilizada para la gestión de proyectos.
- Extreme Programming se centra en potenciar las relaciones interpersonales del equipo de desarrollo como clave del éxito mediante el trabajo en equipo, el aprendizaje continuo y el buen clima de trabajo.
- Esta metodología pone el énfasis en la retroalimentación continua entre cliente y el equipo de desarrollo y es idónea para proyectos con requisitos imprecisos y muy cambiantes.



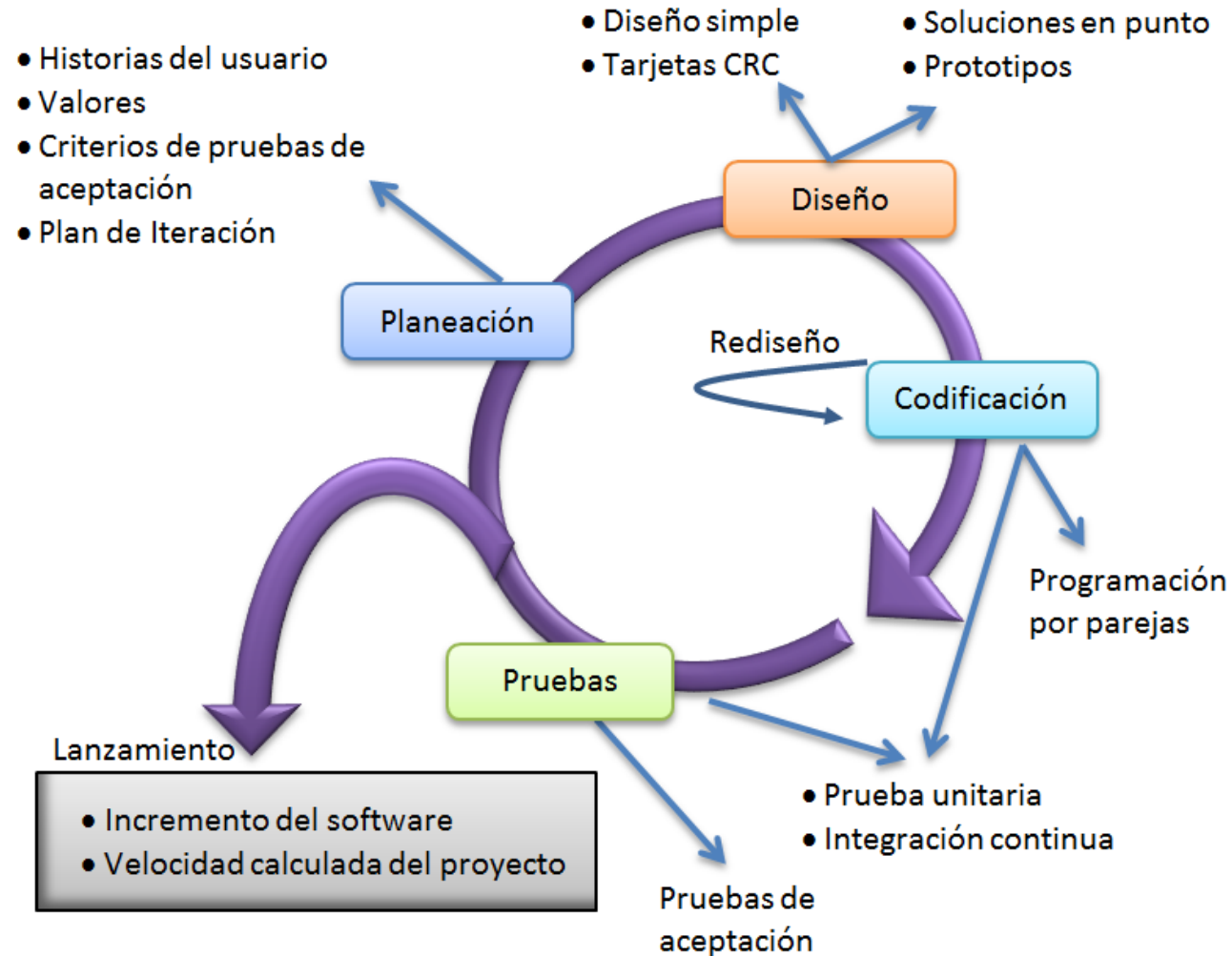
Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Roles en XP



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Ciclo de vida en XP



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

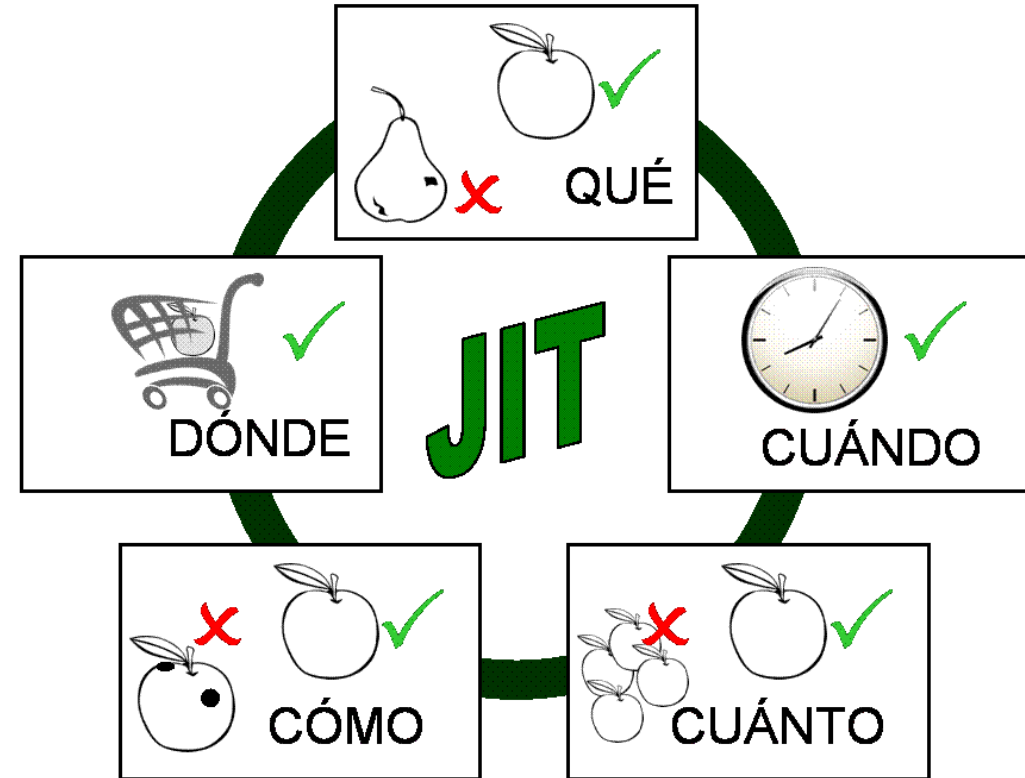
Kanban

- La metodología se basa en el sistema de producción pull, el cual tiene como principio la producción de productos exclusivamente cuando es necesario (Just in Time), por lo que se evitan los desperdicios.
- Con la implementación de la metodología Kanban, se busca la satisfacción del cliente gracias a los tiempos de entrega optimizados y a la mejora de la calidad general del producto finalizado.



¿Qué es?

El Kanban se considera como subsistema del **JIT** ("Just In Time" o justo a tiempo).



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Cómo funciona la metodología Kanban

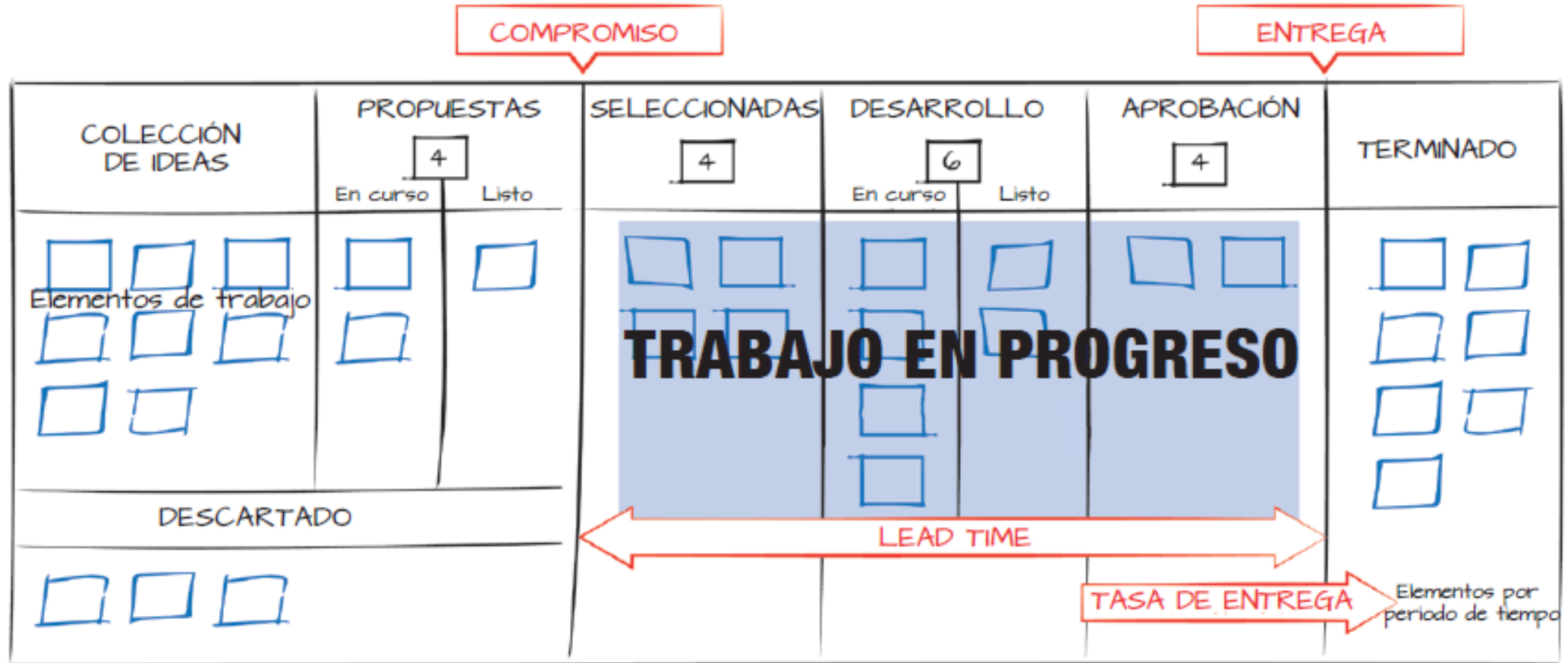
Kanban es una metodología que se basa en elementos visuales como lo son el tablero Kanban y las tarjetas Kanban.

- En un principio, se estudia el proceso, se determinan las etapas por las que atraviesa una actividad o un producto para considerar que está finalizado y se elaboran tarjetas que irán completando estas etapas hasta llegar al final del tablero; el final supone la culminación con éxito de esa actividad que suma a la producción del producto o, incluso, a la fabricación total de este.
- Es una metodología versátil ya que, dependiendo de la construcción de las tarjetas y el tablero, podremos implementarlo tanto en proyectos sencillos, como proyectos más complejos.
- Una parte importante de la implementación de la metodología es la reflexión y análisis de los resultados para proponer mejoras.
- Importante destacar que en la metodología Kanban, buscamos mejoras de forma incremental y no radicales.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Descripción de los sistemas de flujo



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Los cuatro principios básicos de la metodología Kanban

- **Se prioriza la calidad:** Aunque es una metodología que apuesta por mejorar los tiempos de entrega, se fundamenta en hacer bien cada actividad. Es por esto que, la metodología se enfoca en la descripción del proceso y la definición de etapas, además, pone límites de trabajos en cada una de estas etapas para evitar que se acumulen tareas. Cuando las actividades no se acumulan más allá de las capacidades del trabajador, este se dedica a hacer su mejor esfuerzo por entregar un resultado de calidad. Asimismo, la organización permite que los miembros del equipo se concentren en una sola actividad.
- **Se reducen los desperdicios:** No se produce más de lo necesario, ni se agregan características que no sumen valor al producto final. Esto, además de realizar bien las tareas para no tener que repetirlas, supone un ahorro de tiempo y de materia prima, por lo que se reducen los desperdicios. De manera general, con Kanban se produce bajo demanda, por lo que, no se producen productos que no son requeridos.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Los cuatro principios básicos de la metodología Kanban

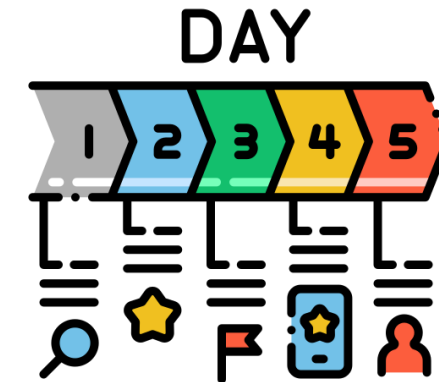
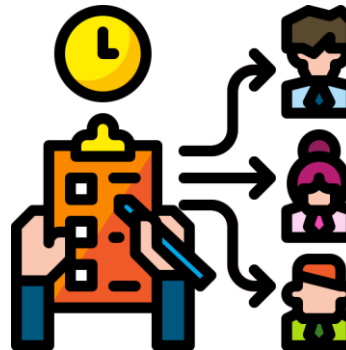
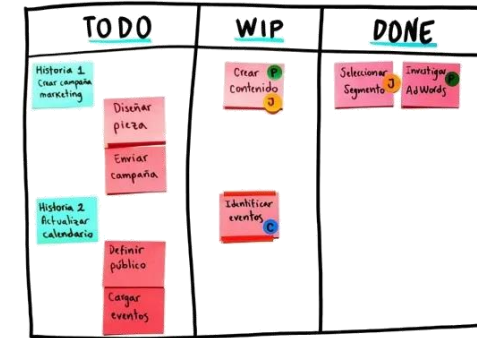
- **Su implementación debe ser flexible:** Parte de las ventajas de la metodología Kanban es que permite que existan variaciones en las actividades con el fin de entregar el producto con las características deseadas. Esto incluye priorizar algunas actividades sobre otras, de acuerdo con su urgencia. Asimismo, producto del análisis del proceso, se pueden introducir cambios incrementales y evolutivos para que este fluya mejor.
- **Se mejora constantemente:** Para Kanban no es importante que se hagan cambios radicales, pero sí que se implementen mejoras cuando exista alguna oportunidad. Incluso acciones tan pequeñas como cambiar los límites de trabajo, pueden causar una mejora en el proceso.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Características de la metodología Kanban

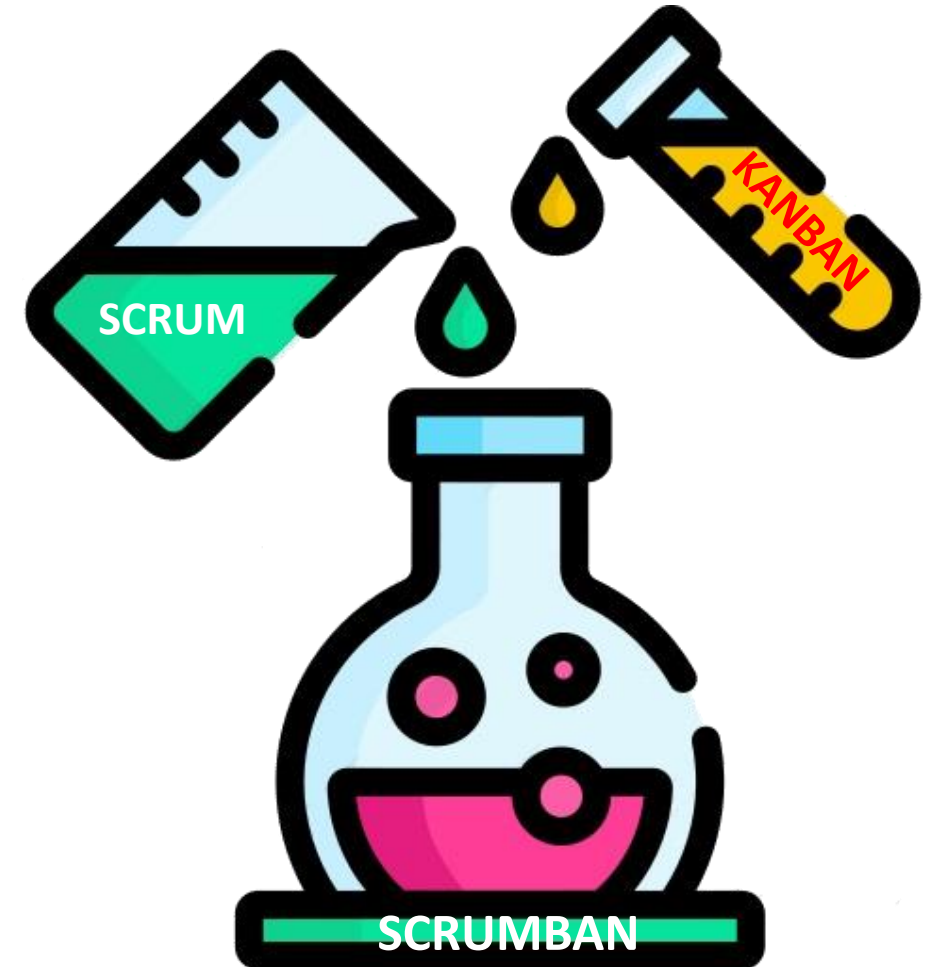
- Es una metodología visual
- Se implementa fácilmente
- Se fundamenta en el trabajo colaborativo
- Evita el multitasking
- Promueve el correcto flujo del proceso



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Scrumban

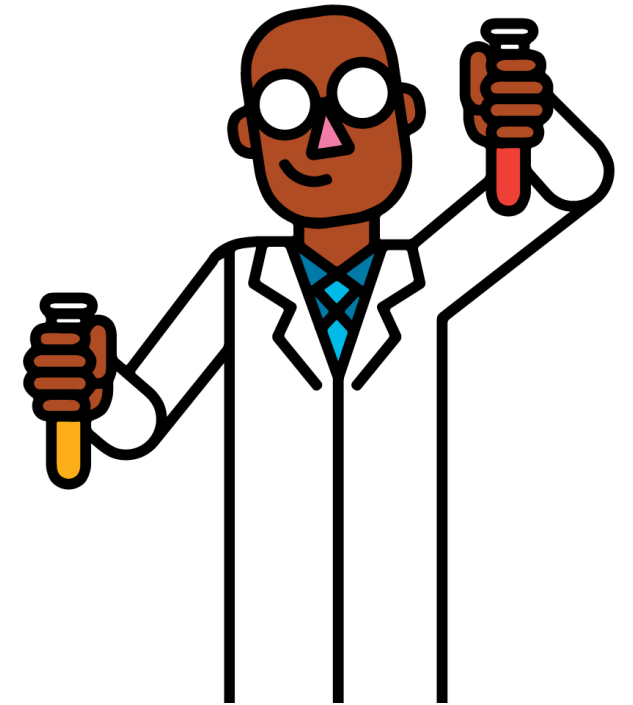
- Es una metodología visual
- Se implementa fácilmente
- Se fundamenta en el trabajo colaborativo
- Evita el multitasking
- Promueve el correcto flujo del proceso



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

¿Por qué es Scrumban una buena elección?

- Nace de la combinación de principios de los métodos ágiles de gestión de proyectos más importantes en la actualidad: Scrum y Kanban.
- Se encarga de combinar aquellos elementos que resultan complementarios
- Se trata de gestionar las tareas previstas con el método Scrum y planificar los errores con el método Kanban.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

¿Cuándo usar Scrumban?

- Gestión de proyectos problemáticos (proyectos con historias de usuarios inesperadas y virus).
- Trabajo dirigido por eventos (apoyo, endurecimiento).
- Mantenimiento de proyectos.
- Desarrollo de nuevos productos (trabajo anterior al desarrollo del sprint o posterior al desarrollo del sprint).
- Mejora continua de la gestión.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Características de Scrumban

- Reduce el tiempo dedicado a planificación y estimación promoviendo el flujo constante de trabajo.
- Desarrolla una cultura de mejora continua, haciendo hincapié en la entrega progresiva sin sobrecargar al equipo de desarrollo.
- Alienta la filosofía de “pensar en grande, actuar en pequeño, fallar rápido, aprender rápidamente”. (Innovación. Competitividad. Flexibilidad y Productividad.)



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Beneficios de Scrumban

- Permite conocer en estado real el proceso de ejecución del proyecto.
- Introduce soluciones oportunas ante eventuales errores.
- Permite un mayor análisis de tareas realizadas.
- Mejora la interacción entre los miembros de un grupo en las reuniones periódicas.
- Aumenta la productividad de proyectos complejos o multiproyectos.
- Favorece una mayor adaptabilidad de las herramientas a las exigencias del proyecto.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Roles de equipo en Scrum

- **SCRUM MASTER:** Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología. Gestiona la reducción de impedimentos del proyecto y trabaja con el product owner.
- **PRODUCT OWNER (PO):** Representante de los accionistas y clientes que usan el software.
- **TEAM:** Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las historias a las que se comprometen al inicio de cada sprint.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Lean

La metodología lean, o también conocida como enfoque lean startup, consiste en mejorar todos los procesos en la creación, puesta en marcha y desarrollo de una empresa. Básicamente, su objetivo es limpiar todas aquellas actividades que no aportan nada y obtener a cambio un producto y experiencia final de calidad para el cliente.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Principios Lean

- **Eliminar el desperdicio:** Todo lo que no añade valor al cliente se considera un desperdicio
- **Ampliar el aprendizaje:** A través de la integración del cliente en el ambiente de desarrollo
- **Decidir en el último momento responsable:** Proporcionando la flexibilidad necesaria para una tardía toma de decisiones.
- **Reaccionar tan rápido como sea posible:** Usando Just In Time, reconociendo las necesidades específicas y el ambiente; se entrega de valor a corto plazo; los equipos se auto - organizan y acuerdan una cadencia de entregas (en períodos cortos).



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Principios Lean

- **Potenciar el equipo:** "Busca gente efectiva y déjalas hacer su trabajo “considerando que las personas necesitan balance en: aceptación, reconocimiento y afecto
- **Crear integridad:**
 - Comprensión del dominio del problema y resolviéndolo al mismo tiempo, no secuencialmente.
 - La información necesaria es producida y entregada al equipo de desarrollo en pequeños lotes.
 - Una de las maneras más saludables hacia una arquitectura integrante es la refactorización del código
 - Automatizar las pruebas, específicamente para la reducción de defectos.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Principios Lean

- **Véase ‘el todo’ tanto como sea posible:** "Piensa en grande, actúa en pequeño equivócate rápido; aprende con rapidez “



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

¿Cuáles son las ventajas del sistema Lean?

- Permite mejorar la productividad gracias a la búsqueda de “cero defectos”.
- Es un método que busca reducir los costos gracias a la optimización de los procesos de producción y la eliminación de gastos innecesarios.
- Mayor satisfacción para el cliente, ya que tienes más tiempo y dinero para mejorar tu producto.
- Disminuye el riesgo para el emprendedor, ya que las decisiones son tomadas en base a datos en mano y no en base a apreciaciones.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

¿Cuáles son las ventajas del sistema Lean?

- Permite contar con un producto mínimo viable que cumple con las necesidades de un cliente en específico.
- Reduce la tasa de fracaso.
- Es una gran forma de obtener datos reales, objetivos y fiables que validen tu hipótesis o idea de negocio; además, el método lean startup permite que puedas descubrir comportamientos de tus clientes que no conocías.



Aprendemos: Metodología de desarrollo de software – Metodología Ágil

Pasos y consejos para seguir el método Lean

- Dar la debida importancia al proyecto de mejora de procesos.
- Garantice el apoyo de los socios y los líderes de la compañía.
- Comience con procesos sencillos, pequeños pasos hacen grandes cambios, opta por lo simple.
- Sé creativo, una de las mayores ventajas del método es que puedes ir mejorando producto hasta hacerlo perfecto y original.
- Involucre a todos los miembros del negocio o proyecto.
- Establece metas claras y bien definidas para tu negocio.
- Determine un espacio físico exclusivo para las reuniones del proyecto.



Verificamos lo aprendido:



<http://www.kahoot.it/>



Verificamos lo aprendido:

- Absolución de Preguntas.
- Conclusiones y Recomendaciones.



Aplicamos lo aprendido: Asignación para la clase siguiente

- Comentar en el Foro



Gracias

