



**Administración de proyectos**  
**Metodologías ágiles**

Módulo:	Administración de proyectos informáticos
Alumno:	Cristopher Andrés Pérez Vargas
Profesor:	José Rubén Hernández Bello
Fecha entrega:	01/10/2018

Índice

**Introducción..... 1**

**Metodologías ágiles..... 2**

**Metodología ágil: Scrum ..... 2**

**Metodología ágil: Kanban ..... 3**

**Metodología ágil: Scrumban ..... 4**

**Metodología ágil: Lean development..... 5**

**Metodología ágil: Feature driven development..... 6**

**Metodología ágil: Test driven development..... 7**

**Conclusión ..... 8**

## Introducción

En la sociedad actual, todo proyecto informático que se busca llevar a cabo, debe pasar por un proceso en cual se delimitan todo lo que abordará el mismo, la rapidez con la cual se trabajará, el orden de las etapas, quienes trabajarán en el mismo, entre otras; y es así como surge una herramienta que engloba todo lo anterior: la metodología de desarrollo de software.

Las metodologías de desarrollo de software tienen como objetivo presentar un conjunto de técnicas tradicionales y modernas de modelado de sistemas que permitan desarrollar software de calidad, incluyendo pautas de construcción y criterios de comparación de modelos de sistemas. Si el término se complica un poco, se puede decir que una metodología, valga la redundancia, es el método con el cual se llevará a cabo el proceso de desarrollo de software.

Del párrafo anterior, se puede sacar en claro tres conceptos que están bajo la metodología y los cuales son: estructurar, planear y controlar el proceso. Estructurar hace referencia al marco en el cuál se trabajará o bajo qué estándares estaremos, planear será el cómo se distribuirá el tiempo, quiénes participarán, en cuánto tiempo se tendrá listo y claro, un control que se deberá llevar cabo para poder contrastar que lo que estamos haciendo esté o no correcto, si es lo que se pidió en un principio o que si se cumplen con lo solicitado.

Si bien existen las metodologías clásicas que son rígidas, es decir, una vez que se comienza con el proyecto, no habrán pausas ni cambios en el diseño del proyecto hasta el final, y están las metodologías ágiles que se caracterizan por ser flexibles, por tener alta relación con el cliente y de poder ir variando al pasar el tiempo, ajustándose a las necesidades cambiantes del cliente. En ésta ocasión, el presente informe tratará de estas últimas metodologías.

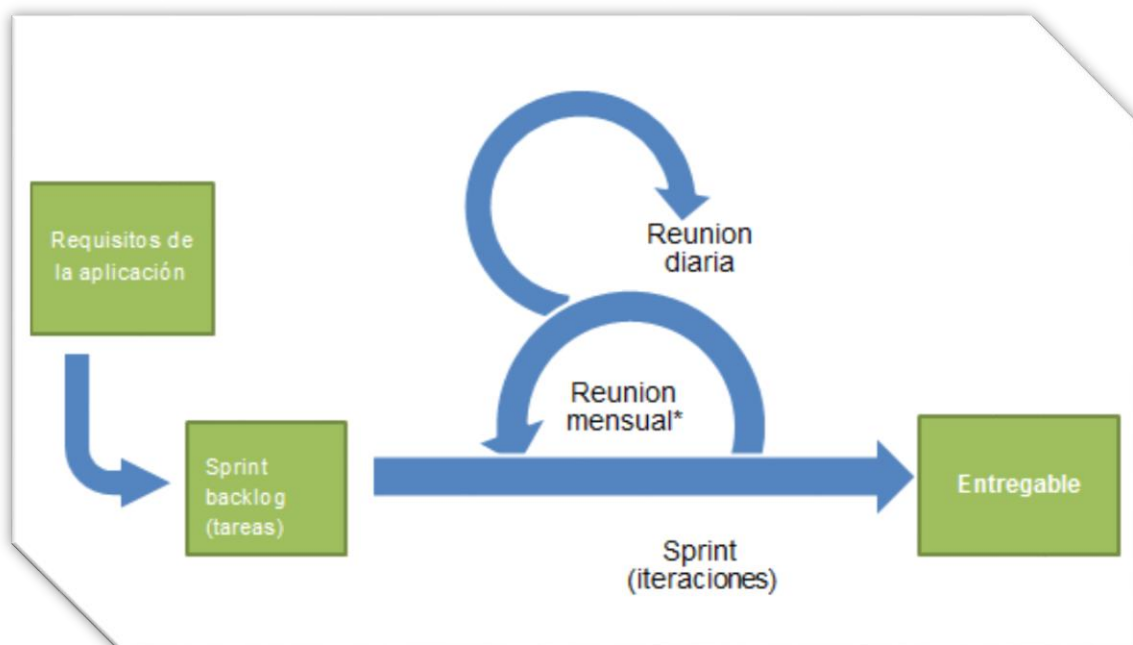
## Metodologías ágiles

Enfocadas en la toma de decisiones, donde los requisitos y soluciones evolucionan con el tiempo según la necesidad del proyecto. Así el trabajo es realizado mediante la colaboración de equipos auto-organizados y multidisciplinarios, inmersos en un proceso compartido de toma de decisiones a corto plazo en donde se evidencian pequeños ciclos de vida que se denominan iteraciones. Mencionar que cada iteración del ciclo de vida incluye: planificación, análisis de requisitos, diseño, codificación, pruebas y documentación.

### Metodología ágil: Scrum

La metodología scrum es usada en equipos que trabajan en proyectos complejos; una metodología de trabajo ágil que tiene como finalidad la entrega de valor en períodos cortos de tiempo, basada tres pilares: la transparencia, inspección y adaptación.

- Transparencia: todos los miembros del equipo tienen conocimiento de qué ocurre en el proyecto y cómo ocurre. Lo anterior hace que haya una visión global y compartida del proyecto en sí.
- Inspección: inspeccionar el progreso para detectar posibles problemas por parte de los miembros del proyecto.
- Adaptación: si hay algo que cambiar, se cambia. El equipo se ajusta para conseguir el objetivo de la iteración.



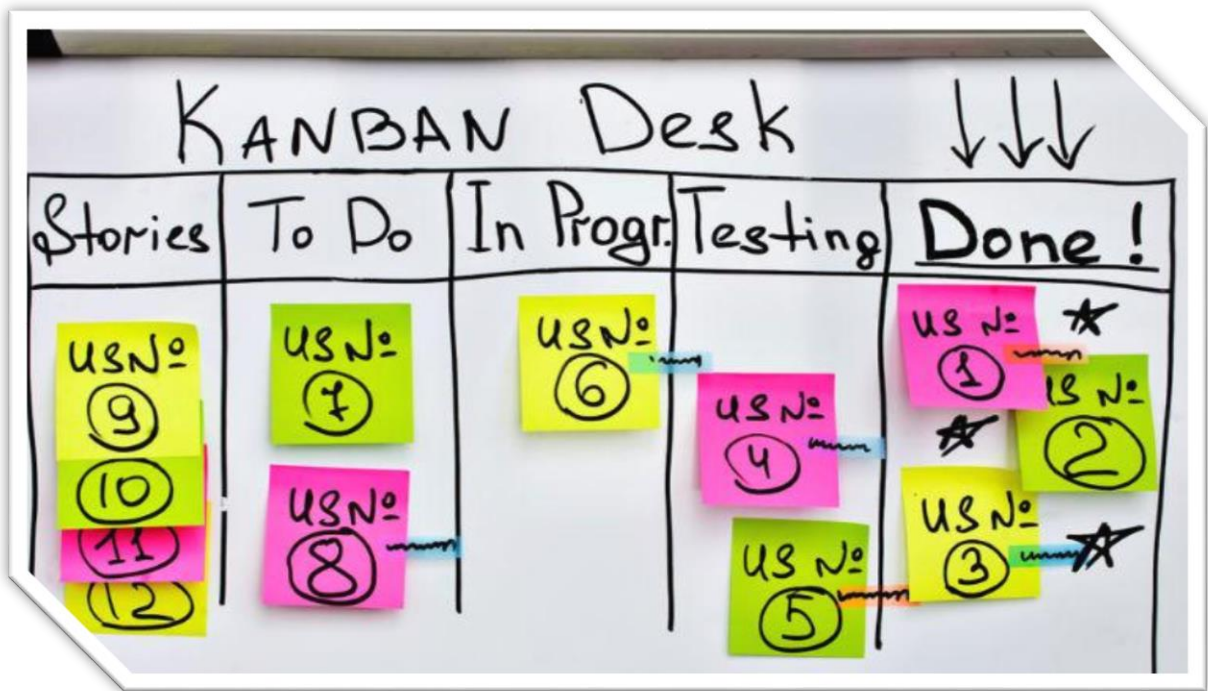
**Metodología ágil: Kanban**

Primero que nada, Kanban significa tarjetas visuales, por lo cual, ya se nos entrega una referencia del tipo de metodología y su funcionamiento basado en tarjetas con las distintas actividades que se deben llevar a cabo.

- **Objetivos:** se marcan a largo plazo, con la idea de que todos los miembros del equipo los tengan en mente. Es una columna opcional, no siempre está presente.
- **Pendiente:** esta columna engloba las tareas pendientes que se pueden afrontar de forma inmediata. En el lugar más alto de esa columna colocaremos la tarea pendiente que tiene la máxima prioridad, y en cuanto empecemos, la pasaremos a las siguientes columnas.
- **Desarrollo:** en este espacio situamos la tarea hasta que la terminemos. Si algo falla, regresa a la columna anterior.
- **Prueba:** comprobamos que todo funciona bien. En función de ese examen, la tarea avanza en el tablón o retrocede.
- **Hecho:** cuando ya no tenemos que preocuparnos más de algo porque hemos terminado la tarea.

Las principales ventajas de esta metodología es que es muy fácil de utilizar, actualizar y asumir por parte del equipo. Además, destaca por ser una técnica de gestión de las tareas muy visual, que permite ver a golpe de vista el estado de los proyectos, así como también pautar el desarrollo del trabajo de manera efectiva

Por último, hay que decir que la labor del gestor es básicamente determinar las tareas que hay que hacer y cambiar su prioridad en función de los acontecimientos ya que toda la cadena de trabajo está a la vista de todos, y si hay atascos queda claro dónde se producen.



Metodología ágil: Scrumban

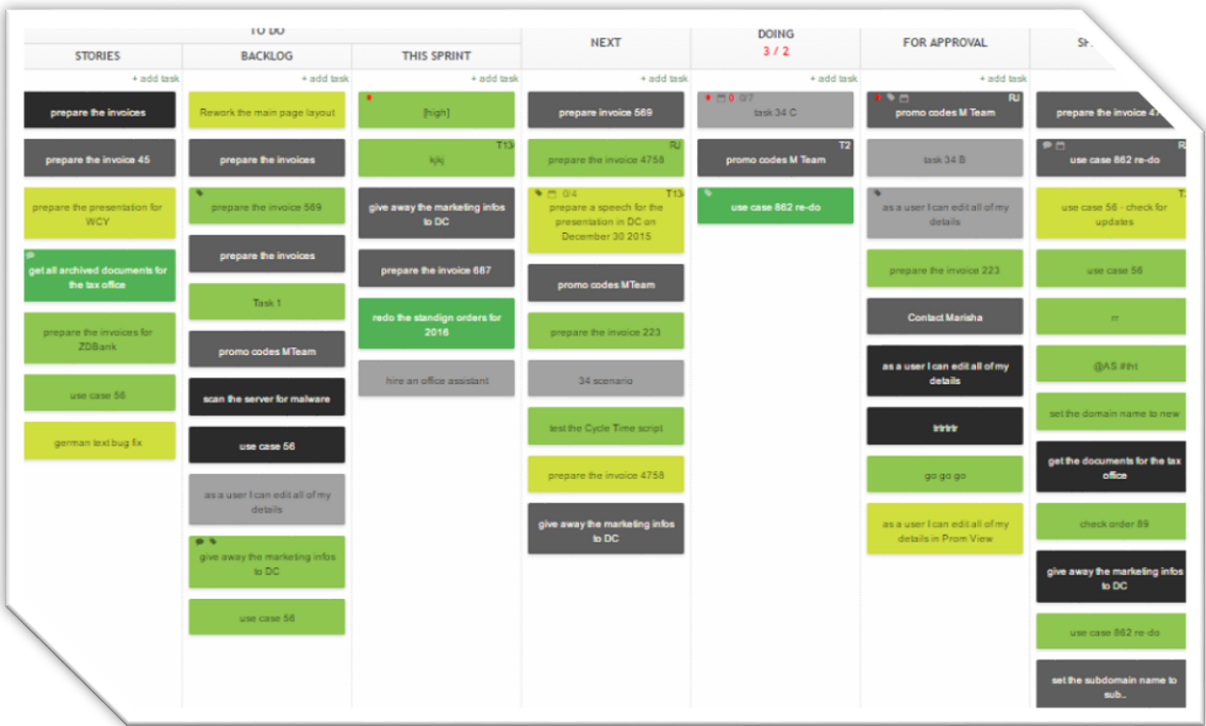
Como su nombre lo dice, la metodología ágil scrumban reúne las mejores características de ambas metodologías (scrum y kanban). Scrumban se centra en hacer planes de iteración a intervalos regulares, sincronizados con revisión y retrospectiva, pero el objetivo de la planificación es llenar los espacios disponibles, no llenar todos los espacios, y no determinar el número de espacios.

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.

Las ventajas más representativas de dicha metodología son las siguientes:

- Calidad
- Just-in-time (justo a tiempo), decisiones y hechos justo cuando se necesitan.
- Plazo de ejecución corto.
- Mejora continua.
- Minimizar todo lo que no le entrega valor al cliente.

Por último, mencionar que el tablero de scrumban te proporciona una excelente visión del flujo de trabajo de un proceso. Informa del número de cosas en las que el equipo está trabajando actualmente, así como del número de tareas ya finalizadas. Aumenta la rendición de cuentas, la responsabilidad, la comunicación y los resultados de rendimiento.



## Metodología ágil: Lean development

Lean development se constituye principalmente por un equipo fuerte y altamente preparado capaz de llevar a cabo cualquier tarea en poco tiempo, logrando todo a la eficacia y la cohesión de los componentes del equipo y obviando los procesos y la burocracia que conlleva normalmente el tener un sistema de producción preestablecido.

La filosofía lean dicta que debes tener un equipo muy preparado, muy motivado y muy unido, tener un alto grado de compromiso y, sobre todo, cuánto está aprendiendo tu equipo. Comprendiendo esta práctica desde un punto de vista empresarial, deberás hacer una inversión fuerte al principio para poder sostener a un equipo que recién se está conociendo y constituyéndose, pero a medio plazo estos costes se reducirán y la productividad subirá, previendo a la larga un escenario en el que los costes de producción se mantienen estables y la productividad del equipo es extraordinaria debido a toda la experiencia que han ido obteniendo durante todo el tiempo.

Los principios en los que se basa dicha metodología son los siguientes:

- Eliminar todo lo que no brinde valor al cliente.
- El aprendizaje es fundamental para el grupo.
- Las decisiones para lo último.
- Entregar lo más rápido posible.
- Capacitación al grupo de trabajo.
- Todos los componentes forman uno solo, buscando el bien en común.
- Ver todo junto, no por separado.

Por último, las ventajas que se pueden apreciar son que poseen un sólido grupo de programadores que son capaces de analizar la situación, tomar decisiones correctas y llevarlas a cabo a una velocidad fuera de lo normal.



### Metodología ágil: Feature driven development

Feature driven development o simplemente FDD, es una metodología ágil diseñada para el desarrollo de software, basada en la calidad y el monitoreo constante del proyecto. Dicha metodología se enfoca en iteraciones cortas, que permiten entregas tangibles del producto en un periodo corto de tiempo, de como máximo dos semanas.

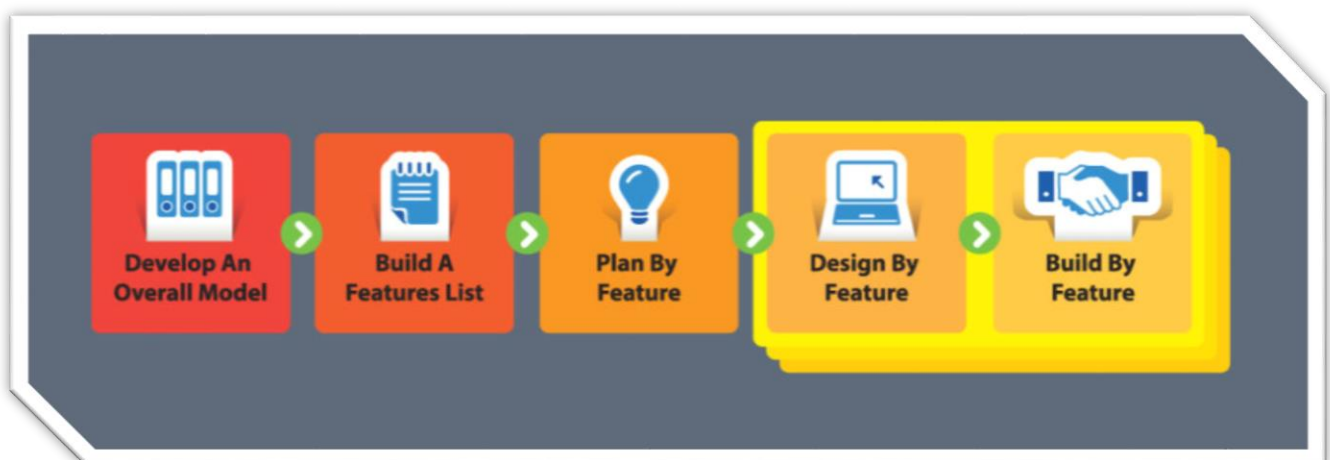
Se definen principalmente cinco pasos:

- Desarrollar el modelo global: al inicio del desarrollo se construye un modelo teniendo en cuenta la visión, el contexto y los requisitos que debe tener el sistema a construir.
- Construir una lista de características: se crea una lista que resuma las funcionalidades que debe tener el sistema, cuya lista es evaluada por el cliente.
- Planificar: se procede a ordenar los conjuntos de funcionalidades conforme a su prioridad y dependencia, y se asigna a los programadores jefes.
- Diseñar: se procede a diseñar y construir la funcionalidad mediante un proceso iterativo, decidiendo que funcionalidad se van a realizar en cada iteración.
- Construir: se lleva a cabo la construcción total del proyecto.

Mencionar que los 3 primeros pueden considerarse la “iteración cero”, aunque en FDD no le llaman así, y los consideran “procesos iniciales”.

Entre las ventajas que más se destacan tenemos:

- Cada componente del producto final ha sido probado y satisface los requerimientos.
- Rápida respuesta a cambios de requisitos a lo largo del desarrollo.
- Entrega continua y en plazos cortos de software funcional.
- Trabajo en conjunto entre el cliente y el equipo de desarrollo.
- Minimizar los costos frente a cambios tanto repentinos como avisados.





### Metodología ágil: Test driven development

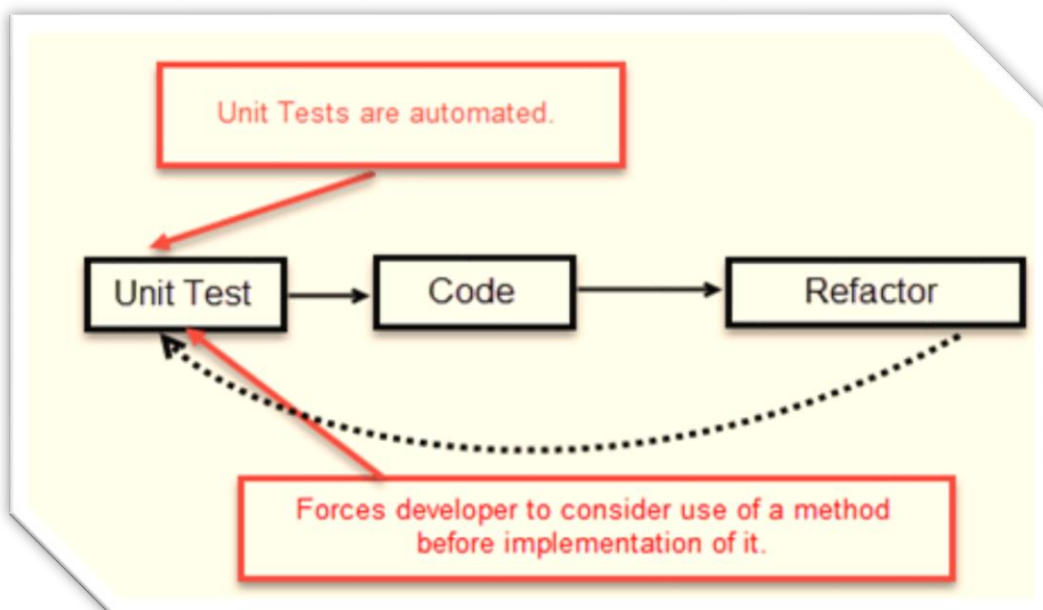
Test driven development o solo TDD, es una práctica de programación que consiste en escribir primero las pruebas (generalmente unitarias), después escribir el código fuente que pase la prueba satisfactoriamente y, por último, re factorizar el código escrito con lo que logra obtener un código más robusto, más seguro y una mayor rapidez en el desarrollo del sistema.

Estos tests también nos ayudan cuando trabajamos en equipo, ya que si disponemos de una buena batería de test unitarios y funcionales, será infinitamente más fácil explicar el proyecto a un compañero: solo tendrá que leer los test para entender realmente lo que está haciendo la aplicación. Otra característica importante es que nos permiten evolucionar nuestras aplicaciones de forma segura. Al modificar o añadir algo nuevo, nos ofrecen la certeza de no haber roto ninguna funcionalidad o, en caso contrario, nos avisan de que nuestros cambios han alterado el funcionamiento de la aplicación de manera inesperada.

Los pasos a seguir en dicha metodología son los siguientes:

- Agregar una prueba.
- Ejecutar todas las pruebas y ver si falla la nueva prueba.
- Escribir el código.
- Ejecutar pruebas.
- Re factorizar el código.

Una vez realizado el proceso, se debe comenzar con otra nueva prueba, así el ciclo se repite para impulsar la funcionalidad.



## Conclusión

Como se pudo apreciar, las metodologías ágiles se caracterizan por entregas rápidas, eliminación de todo lo que no aporte valor al cliente, estar abierto a los cambios y a una mejora continua del proceso en sí. Al caracterizarse por ser muy flexibles, esto arroja como resultado una estructura muy débil, en donde reina la poca exactitud en la planificación del proyecto, desde los plazos de entrega hasta los presupuestos. Como no existe un plan concreto, todo parece flotar en el vacío aunque todo dependerá de su implementación, sea buena o sea mala.

También hay que mencionar que más allá de las dos metodologías más conocidas como lo son Scrum y Kanban, hay otras que mantienen la esencia en lo que ser ágil se refiere pero con pequeñas modificaciones para sacar el máximo provecho del proceso en sí, como lo es scrumban con su sistema de tabla en donde todo el equipo puede ver el desarrollo del sistema o TDD con su pensamiento de desarrollar un plan de pruebas al comienzo de cada iteración para poder minimizar el impacto que puedan tener errores a futuro. Es así como las metodologías pueden tener pequeñas diferencias pero no por eso dejan de ser de la misma línea, simplemente es que cada una de ellas se aplica a un proyecto en particular, ya que cada una posee sus propias ventajas y desventajas; todo ya depende del grupo de trabajo y de lo que tiene como prioridad.

De ser posible, las mejores variantes y las más llamativas son TDD por su sistema de pruebas primero y lean development ya que se centra en el grupo de programadores, en obtener experiencia desde 0 e ir aumentando los conocimientos con cada trabajo.