



# Metodologías ágiles en práctica. Problemas y soluciones

Gustavo Cebrián García  
Scrum Master Certified (Mike Cohn)





# Metodologías Ágiles → Puntos a tratar

- Ciclo de Vida Nuevo. Explicar el cambio al manager.
- Scrum. Marco de mejora.
- Manifiesto Ágil
- Principios
- Conceptos



# Como le planteamos el cambio a un manager

- En los proyectos hay muchos **cambios de prioridades**
- No se vas a poder transmitir todos los conocimientos en unos folios siguiendo metodología en cascada, y habrá muchos **cambios de alcance**
- Vas a ver la **velocidad de desarrollo**, que aumentará con el conocimiento del negocio por parte del equipo
- Vemos la dimensión del proyecto y **formamos un equipo y un coste**
- Costes, Alcance, Tiempo. El Alcance es la variable.
- Si que podemos dar estimaciones realistas, pero no muy alejadas en el tiempo
- Necesitamos un **equipo muy involucrado** por parte del cliente
- Hacer ese documento de **1000 páginas ya es un error** ( ya podrías haber empezado) y ni siquiera es ético. Es más, pueden engañarte fácilmente, confía en nosotros!



# Scrum como marco de mejora

- Anteproyecto Prince2
- Product Owner
- Scrum Master (Agile Couch)
- Equipo. ( Equilibrio entre Testers y Developers)
  
- Sprint Planning
- Daily Stand up
- Demonstration
- Retrospective



# Individualidades e interacciones...

- Se utilizan herramientas y procesos, **pero un proceso se automatiza cuando da valor y es muy repetitivo**. Hay procesos que no añaden valor, y estos se suplen con interacciones humanas, más rápidas.
- Hay veces que no es bueno utilizar todo tipo de herramientas.
- Ejemplo, bolígrafo y papel es muy rápido en muchas ocasiones ( con su respectiva fotografía )
- El miembro del equipo no es un programador, es un **desarrollador**. Tiene que saber del negocio, este es el valor. En EEUU los desarrolladores están más valorados.



# Software que funciona...

- ... frente documentación comprensible
- Nivel de documentación. Se optan por equipos multidisciplinares, en negocio y en programming. Mantener la documentación es muy muy difícil. Hacer un buen manual es necesario muchas veces, y linkar el código con la documentación es útil.



# Colaboración del cliente...

- ... frente a la negociación de un contrato
- 
- -Eso ya lo dijimos.
- -Es lo que está en la Excel
- -Hay que entrenar muy bien al usuario/Cliente
- -Si el equipo es de 7, aparte del product owner, 2 personas
- -Hacer al cliente entender el burn-down
- -Hacerles entender las historias de usuario, condiciones de aceptación
- -Tienen que entender los cambios de alcance y los requisitos nuevos
- -Colaboración significa hacer testing, pensar en nuevos casos, definiciones con ejemplos...



# Respuesta al cambio...

- ... frente al seguimiento de un planning estricto
- -Quitamos estas HUs y ponemos estas HUs. Esto es el mejor indicador ágil.
- -Largo plazo. En software, estimar más de 2 o tres meses no da valor.
- -Para estimar se tiene que ver la velocidad ( Después de 3,4 Sprints Estimar para máximo 12 (3 a 6 meses) ) Depende del tamaño del equipo...
- -Estima todo el equipo



# Principios Ágiles I

1. Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.
  - -Es bueno que **no se haga todo lo planeado en un Sprint**. Otras veces se hará más.
  - -El product Owner y clientes **tienen que ir** a las reuniones de presentación.



# Principios Ágiles II

2. Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.
  - El cliente tiene que **entender lo que es un cambio**, y también el desarrollador. Es un cambio de mentalidad. Si hay cambio de alcance, hay cambio de estimación. Mostrar "Historias de usuario"



# Principios Ágiles III

3. Entregamos frecuentemente software que funcione, desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre una entrega y la siguiente. Tamaño del Sprint. 2 semanas, 3 semanas.
4. Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.
  - El mejor desarrollador es el que intenta entender el negocio, pregunta, indaga...



# Principios Ágiles IV

5. Construimos proyectos en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir el trabajo. Goleman. Inteligencia emocional. Abrir la mente. Pensar en vuestra carrera.
6. El diálogo cara a cara es el método más eficiente y efectivo para comunicar información al equipo de desarrollo y dentro de él. 6 personas, con dos proyectos, o 3 personas para cada proyecto.
  - -Si una persona se va, el conocimiento tiene que estar compartido. XP y Pair programming.



# Principios Ágiles V

7. La medida principal de progreso es el software que funciona. **Software medido en puntos.**
  
7. Los procesos ágiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener un ritmo constante. Software medido en puntos
  - **Hay que saber decir NO.**



# Principios Ágiles VI

9. La atención continua a la excelencia técnica y al buen diseño aumentan la agilidad. -Repasar todas las prácticas técnicas XP, Bob uncle, patrones de diseño, etc
10. La simplicidad (arte de maximizar la cantidad de trabajo no realizado) es esencial. Retrospectivas

- .

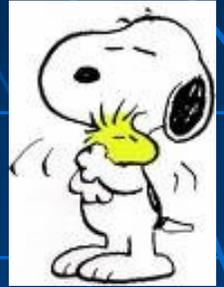


# Principios Ágiles VII

11. Las mejores arquitecturas, requisitos y diseños surgen de los equipos de trabajo organizados por sí mismos. Objetivos comunes, ayudas entre compañeros. Product Owner, que ayude. Que ayude equipo de personas del cliente
11. En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento. Retrospectivas.



# Conceptos Ágiles I

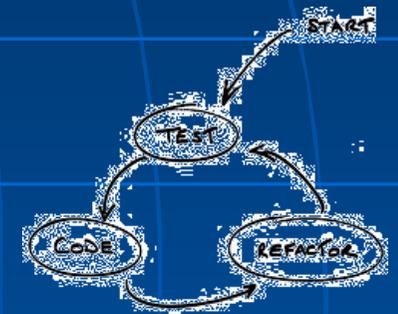


- Historias de usuario
  - Una ventanita, no es una historia de usuario
  - Entre 3 y 10 días.
  - Hamburguesa.
  - Definición con ejemplos.



# Conceptos Ágiles II

- Burn down de sprint
- Burn down de versión





# Conceptos Ágiles III

- Prácticas XP
- Kanban
- Pomodoro, GTD.
- Wiki de proyecto. Herramientas como Rally y RedMine.
- Convencer al manager de vuestras habilidades técnicas. Automatización de pruebas. El Tester va teniendo menos trabajo.





- ¿Preguntas...?
- Gustavo Cebrian Garcia
- [g.cebrian.garcia@gmail.com](mailto:g.cebrian.garcia@gmail.com)

