



# TEMA 8: El proceso de ingeniería de requisitos en el ciclo global del software

## Ingeniería de Requisitos

Raquel Martínez España

Grado en Ingeniería Informática



# Introducción

---

- ¿Qué ocurre con los requisitos cuando tengo metodologías ágiles?
- ¿Qué es una metodología ágil?
  - Metodologías que dan soporte a los valores incluidos en el Manifiesto Ágil.
  - Iterativos de ciclo corto y flexibles.
  - Se centran más en el código que en la documentación, debido a:
    - Son más adaptativos que predictivos: poca planificación.
    - Orientados a las personas: confían en la experiencia, competencia y colaboración



# Metodologías ágiles

---

- Alternativa a las metodologías tradicionales
- Valores del Manifiesto Ágil
  - Valorar a los individuos y su interacción **sobre** los procesos y las herramientas
  - Valorar software que funciona que **sobre** documentación exhaustiva
  - Valorar la colaboración con el cliente **sobre** la negociación contractual
  - Valorar la respuesta al cambio **sobre** el seguimiento de un plan

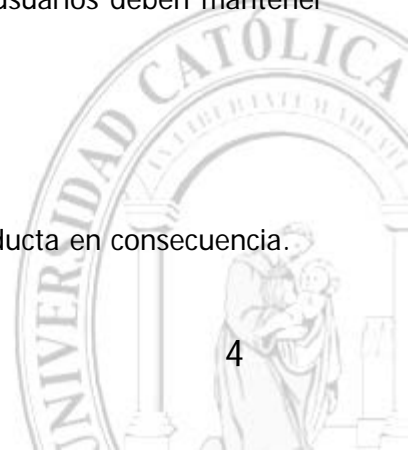


# Metodologías ágiles

---

- Principios Ágiles

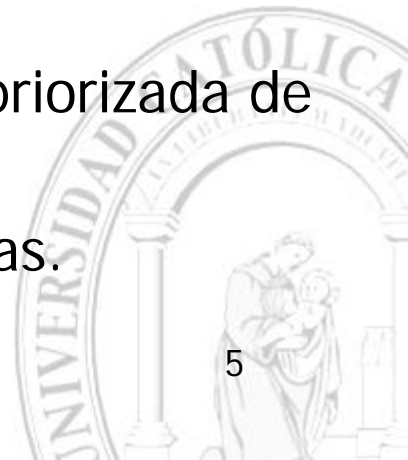
1. Nuestra **principal prioridad es satisfacer al cliente** a través de la entrega temprana y continua de software de valor.
2. Son bienvenidos los **requisitos cambiantes**, incluso si llegan tarde al desarrollo. Los procesos ágiles se doblan al cambio como ventaja competitiva para el cliente.
3. **Entregar con frecuencia software** que funcione, en periodos de un par de semanas hasta un par de meses, con preferencia en los periodos breves.
4. Las **personas del negocio y los desarrolladores deben trabajar juntos** de forma cotidiana a través del proyecto.
5. Construcción de proyectos en torno a individuos motivados, dándoles la oportunidad y el respaldo que necesitan y procurándoles confianza para que realicen la tarea.
6. La forma más eficiente y efectiva de comunicar información de ida y vuelta dentro de un equipo de desarrollo es mediante la conversación cara a cara.
7. El software que funciona es la principal medida del progreso.
8. Los procesos ágiles promueven el desarrollo sostenido. Los patrocinadores, desarrolladores y usuarios deben mantener un ritmo constante de forma indefinida.
9. La atención continua a la **excelencia técnica** enaltece la agilidad.
10. La simplicidad como arte de maximizar la cantidad de trabajo que no se hace, es esencial.
11. Las mejores arquitecturas, requisitos y diseños emergen de equipos que se auto-organizan.
12. En intervalos regulares, el equipo reflexiona sobre la forma de ser más efectivo y ajusta su conducta en consecuencia.



# Metodologías ágiles

---

- Extreme Programming (XP)
  - No habla explícitamente de técnicas de IR.
  - Usa entrevistas, brainstorming, priorización e historias de usuario.
- Modelado Ágil (AM)
  - Idem XP.
  - Diferencia entre los modelos informales y la documentación.
- Scrum
  - Los requisitos se recolectan en el backlog (lista priorizada de características a implementar).
  - En cada sprint se implementan las más prioritarias.



# Metodologías ágiles vs Ingeniería de requisitos

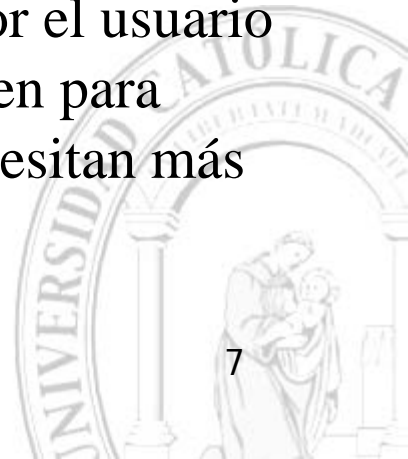
---

- “Captura de requisitos”
  - Implicación de los usuarios:
    - Las metodologías ágiles suponen “cliente ideal”.
    - La IR tradicional evitan esta suposición con técnicas para confrontar y validar.
  - Entrevistas: es la principal técnica de educación de las metodologías ágiles.
  - Priorización: es una práctica, donde se implementan primero los requisitos más valiosos para el cliente. Se reprioriza muy frecuentemente: adaptación al entorno.

# Metodologías ágiles vs Ingeniería de requisitos

---

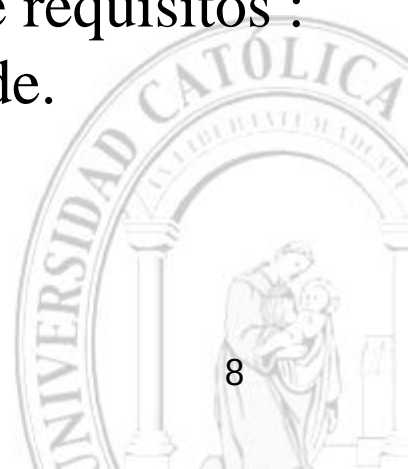
- “Especificación de requisitos”
  - Reuniones de análisis son frecuentes durante todo el proceso. Pero la documentación es menos rígida.
  - Las técnicas de modelado a diferencia de la IR tradicional, en la metodología ágil sólo se usan para facilitar la comprensión.
  - No se pretende generar una documentación completa del proyecto.
  - Se deja mucha libertad al equipo de desarrollo.
  - Las historias de usuario es una de las técnicas más usadas en las metodologías ágiles. Estas historias están escritas por el usuario en su lenguaje, no contienen aspectos técnicos. Sirven para estimar tiempos de desarrollo (casos de uso), no necesitan más detalle.



# Metodologías ágiles vs Ingeniería de requisitos

---

- “Validación de requisitos”
  - Usan reuniones de revisión para la validación de requisitos.
- “Gestión de requisitos”
  - No es posible la trazabilidad total de los requisitos como en la ingeniería de requisitos tradicional.
  - Sin embargo, proporcionan una buena base: backlog, lista de características, ...
  - Aproximación poco detallada a la captura de requisitos : se omiten detalles para completarlos más tarde.



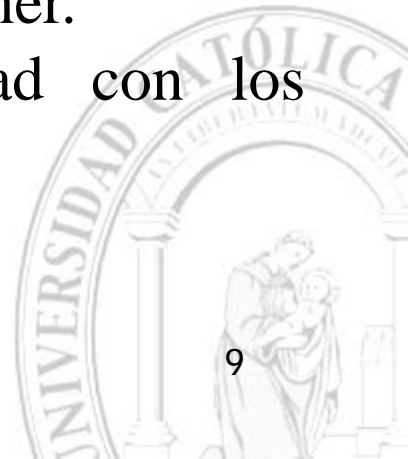


# Metodologías ágiles vs Ingeniería de requisitos

---

- Resumen:

- Las fases de la ingeniería de requisitos no están fácilmente separables en las metodologías ágiles.
- Las técnicas utilizadas para el desarrollo de el proceso de ingeniería de requisitos se encuentran vagamente descritas ya que se confía mucho en el equipo de desarrollo.
- La documentación generada es poco extensa, es bastante compacta por lo tanto más sencilla de mantener.
- Pueden aparecer problema de trazabilidad con los requisitos.



# Metodologías ágiles vs Ingeniería de requisitos

---

- Ahora vamos a estudiar brevemente las características más esenciales de la metodología Scrum, una de las metodologías ágiles más conocidas actualmente:
  - ¿Qué es Scrum?
  - Roles en Scrum
  - Artefactos
  - Eventos



# SCRUM

---

- Metodología ágil.
- Scrum es una metodología, "framework", conjunto de buenas prácticas, para la gestión de proyectos.
- Iteración y entregas incrementales.



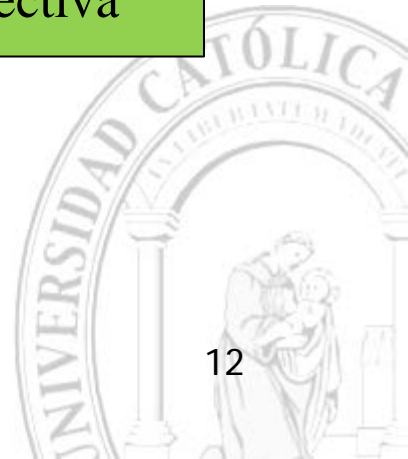
# Elementos de Scrum

---

Roles
<ul style="list-style-type: none"><li>- Product Owner</li><li>- Equipo de desarrollo</li><li>- Scrum Master</li></ul>

Artefactos
<ul style="list-style-type: none"><li>- Historias de Usuario</li><li>- Product Backlog</li><li>- Sprint Backlog</li></ul>

Eventos
<ul style="list-style-type: none"><li>- Sprint</li><li>- Planificación del Sprint</li><li>- Revisión del Sprint</li><li>- Retrospectiva</li></ul>



# Roles

---

- **Product Owner**

*Gestionar las necesidades que serán satisfechas por el proyecto y asegurar el valor del trabajo que el equipo lleva a cabo.*

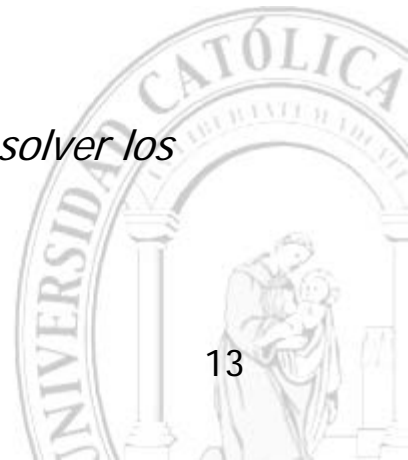
- Recoger y tener claras las necesidades de la aplicación.
- Fijar criterios de aceptación para cada historia de usuario.

- **Equipo de desarrollo**

- Formado por los desarrolladores, equipo ágil auto-organizado, multifuncional, autónomo.

- **Scrum Master**

*Se centra en el proceso de Scrum, actúa como mentor y ayuda a resolver los impedimentos que vayan surgiendo.*



# Artefactos

---

- Historias de Usuario

Son, principalmente, **lo que el cliente o el usuario quiere que se implemente**. Son una descripción breve de una funcionalidad software tal y como la percibe el usuario.

*Como [rol del beneficiario], quiero [objetivo], para poder [beneficio]*

**Como cliente quiero  
poder añadir elementos  
al carrito para poder  
revisarlos antes de  
comprar**

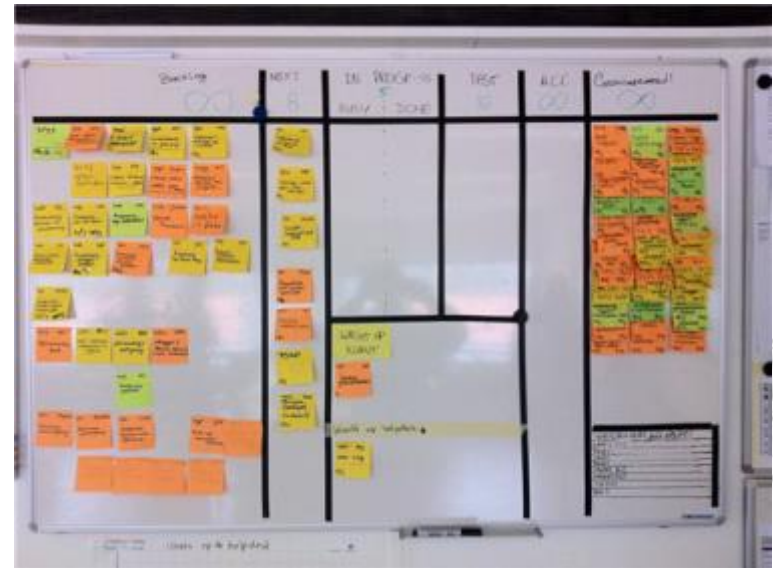
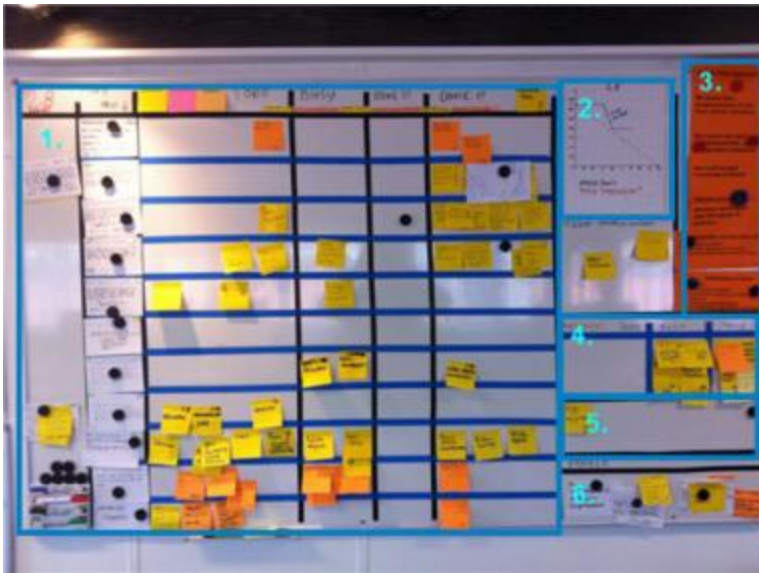


# Artefactos

---

- **Product Backlog**

Listado **ordenado** y **priorizado** con todo aquello que es necesario añadir al producto.



# Artefactos

---

- Product Backlog
  - Estimar Historias de Usuario
    - 1, 2, 3, 5, 8, etc.
    - S, M, L, etc.
  - Priorizar Historias de Usuario
    - MoSCoW
      - M (Must)
      - S (Should)
      - C (Could)
      - W (Won' t)





# Artefactos

---

- **Sprint Backlog**
  - Subconjunto seleccionado del Product Backlog para el sprint.
  - Los elementos del Product Backlog se suelen dividir en tareas
  - **Tablero Scrum:**

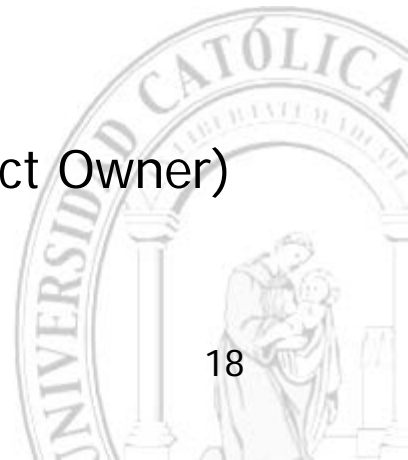
Story	To Do	In Progress	Done
Story A		Task	Task
Story B	Task	Task	Task
Story C		Task	Task



# Eventos

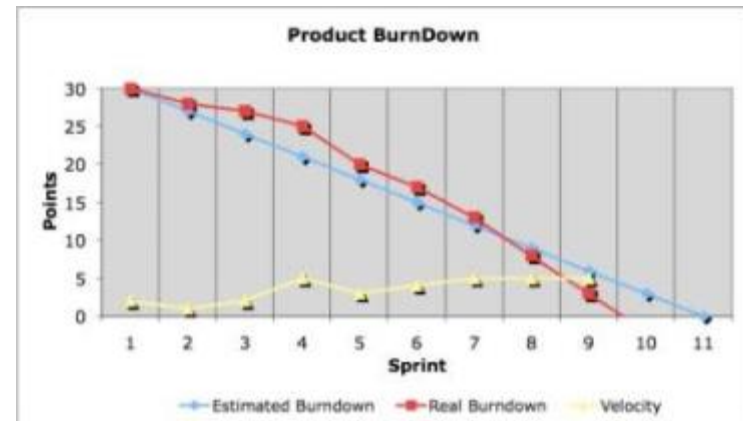
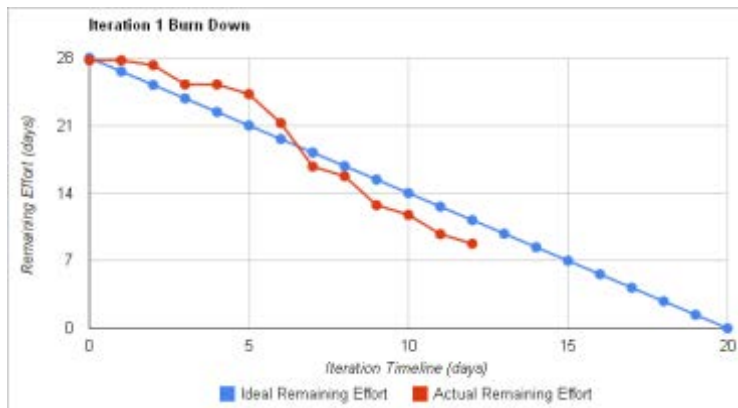
---

- **Sprint**
  - 1 a 4 semanas
  - Producto software potencialmente entregable (prototipo operativo)
- **Planificación del Sprint**
  - Primer día del sprint
  - Product Owner, Scrum Master y el equipo de desarrollo.
- **Daily meeting**
  - No más de 15 minutos.
  - Scrum Master y el equipo de desarrollo (opc: Product Owner)



# Eventos

- Revisión del sprint
  - No más de 4 horas
  - Product Owner, Scrum Master y el equipo de desarrollo.
  - BurnDown Chart:



- Retrospectiva
  - Aspectos + y – del sprint
  - Product Owner y equipo de desarrollo

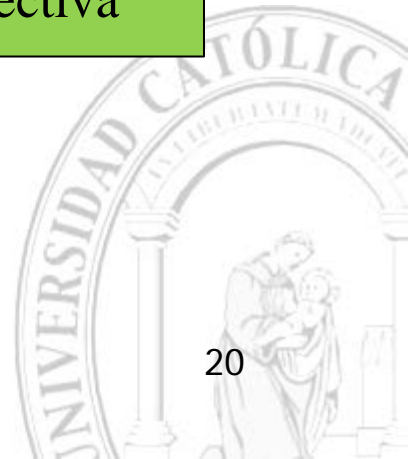
# Elementos de Scrum

---

Roles
<ul style="list-style-type: none"><li>- Product Owner</li><li>- Equipo de desarrollo</li><li>- Scrum Master</li></ul>

Artefactos
<ul style="list-style-type: none"><li>- Historias de Usuario</li><li>- Product Backlog</li><li>- Sprint Backlog</li></ul>

Eventos
<ul style="list-style-type: none"><li>- Sprint</li><li>- Planificación del Sprint</li><li>- Revisión del Sprint</li><li>- Retrospectiva</li></ul>



# Scrum vs. Requisitos

---

## *¿Ingeniería de Requisitos?*



# Scrum vs. Requisitos

---

*Javier Garzas indica que las historias de usuario no son requisitos, que no hay requisitos, otros autores consideran que las historias de usuario pueden ser los requisitos (menos detallados) de las metodologías ágiles .*



# La seguridad de los sistemas

---

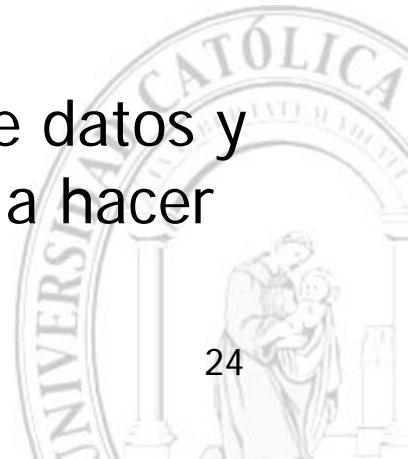
*La seguridad de los datos se debe de tratar, estudiar y establecer desde el mismo momento en que se recogen los requisitos del sistema.*



# La seguridad desde el inicio

---

- La protección de datos tema crítico
- Nuevo reglamento:
  - Ya no se requiere declarar ficheros que describen tratamientos ni tener un documento de seguridad sino que apela a los principios de "Seguridad por defecto y por diseño".
- Se propone como obligatorio hacer un análisis de impacto en la privacidad (PIA)
  - PIA es una análisis funcional de los tipos de datos y las operaciones de tratamiento que se van a hacer con ellos → identificar riesgos potenciales.





# La seguridad desde el inicio

---

- La Agencia Española de Protección de Datos ya ha dado las pautas sobre qué hay que completar en estos estudios. El enlace a la guía de la Agencia Española de Protección de Datos con la guía para hacer PIAs es este:

[https://www.agpd.es/portalwebAGPD/canaldocumentacion/publicaciones/common/Guias/Guia\\_EIPD.pdf](https://www.agpd.es/portalwebAGPD/canaldocumentacion/publicaciones/common/Guias/Guia_EIPD.pdf)



# La seguridad desde el inicio

---

- El nuevo reglamento parte de la unión europea:
  - Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo, de 27 de abril de 2016, relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que se deroga la Directiva 95/46/CE (Reglamento general de protección de datos). Artículo 25 y 35

[https://www.agpd.es/portaIwebAGPD/canaIdocumentacion/legislacion/union\\_europea/reglamentos/common/pdfs/Reglamento\\_UE\\_2016-679\\_Proteccion\\_datos\\_DOUE.pdf](https://www.agpd.es/portaIwebAGPD/canaIdocumentacion/legislacion/union_europea/reglamentos/common/pdfs/Reglamento_UE_2016-679_Proteccion_datos_DOUE.pdf)



# La seguridad desde el inicio

---

- ¿Qué afecta a los requisitos?
  - Deben de contemplarse en los requisitos los aspectos de seguridad correspondiente a la protección de datos e incluirse si fuera necesario el análisis de un PIA para validar el nivel de seguridad necesario en la aplicación.
  - **Requisitos de seguridad indispensables cuando se van a guardar datos personales.**

