




UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

El Ciclo de Vida del Software

*Grupo de Ingeniería del Software y Bases de Datos
Departamento de Lenguajes y Sistemas Informáticos
Universidad de Sevilla
septiembre 2016*



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

El Ciclo de Vida del Software

- Objetivos de este tema
 - Conocer el concepto de **ciclo de vida** del software.
 - Conocer los ciclos de vida más **habituales**.
 - Conocer cómo plantean el ciclo de vida las propuestas **metodológicas** más relevantes.

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles



5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería Inversa


9. Reingeniería del software



septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

1



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

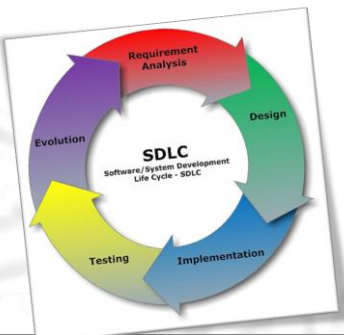
7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software


- ¿Qué es el ciclo de vida del software?
 - Un marco de referencia que contiene los **procesos**, las **actividades** y las **tareas** involucradas en el desarrollo, la operación y el mantenimiento de un **producto** software, abarcando la vida del sistema desde su definición hasta su retirada (ISO 12207).



septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

2



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

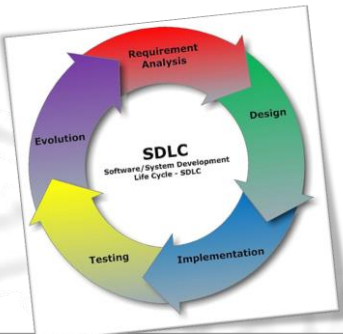
7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- ¿Qué es el ciclo de vida del software?
 - El ciclo de vida de un proyecto especifica el enfoque general del desarrollo, indicando los **procesos**, **actividades** y **tareas** que se van a realizar y en qué orden, y los **productos** que se van a generar, los que se van a entregar al cliente y en qué orden se van a entregar.



septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

3

IISSI

2

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- Ciclo de vida clásico (en cascada)
 - Cada fase comienza cuando termina la anterior.
 - Asume que se conocen todos los requisitos.
 - Se tarda mucho en disponer del software.
 - Es mejor que no seguir ningún ciclo de vida.
 - Es el más fácil de planificar, es el ciclo ideal.

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

4

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- ¿Por qué evolutivo?
 - Cuanto mayor es un proyecto, menor es la probabilidad de éxito (informes CHAOS).
 - Obtener todos los requisitos al comienzo de un proyecto es prácticamente imposible porque las necesidades de clientes y usuarios evolucionan durante el desarrollo.
 - Ciclos *requisitos-desarrollo-evaluación*
 - El resultado de la evaluación permite evolucionar hacia la siguiente versión.

Dr. Barry Boehm
Creador del modelo espiral

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

5

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

Requisitos

Análisis
Diseño
Implementación
Pruebas

1

Versión 1
Funcionalidad 1

Análisis
Diseño
Implementación
Pruebas

1 2

Versión 2
Funcionalidad 2

Análisis
Diseño
Implementación
Pruebas

1 2 3

Versión 3
Funcionalidad 3

Requisitos

Requisitos

El Ciclo de Vida del Software

- Ciclo de vida incremental (I)
 - Repetición de varios ciclos de vida en cascada.
 - Se suele aplicar a desarrollos de gran tamaño.
 - Al final de cada ciclo se entrega una versión parcial del software incrementada con cierta funcionalidad nueva respecto a las anteriores.

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

6

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

Requisitos

Análisis
Diseño
Implementación
Pruebas

1

Versión 1
Funcionalidad 1

Análisis
Diseño
Implementación
Pruebas

1 2

Versión 2
Funcionalidad 2

Análisis
Diseño
Implementación
Pruebas

1 2 3

Versión 3
Funcionalidad 3

Requisitos

Requisitos


El Ciclo de Vida del Software

- Ciclo de vida incremental (II)
 - Los ciclos se repiten hasta obtener un producto completo.
 - Aunque no completamente, los usuarios disponen antes del software y pueden sugerir mejoras (nuevos requisitos).

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

7



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- Ciclo de vida iterativo (I)
 - Repetición de varios ciclos de vida en cascada.
 - Se suele aplicar a desarrollos en los que los requisitos no están claros.
 - Al final de cada ciclo se entrega una versión completa del software mejorada respecto a la anterior.

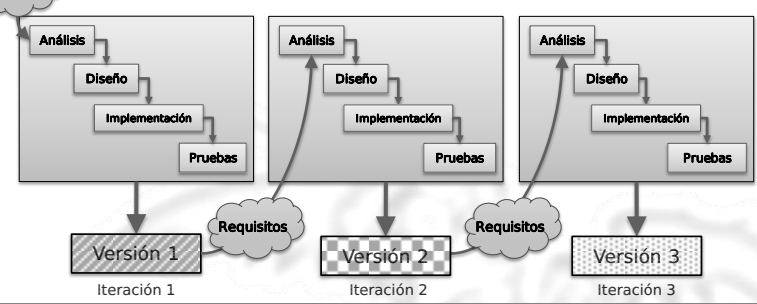



Diagram illustrating the Iterative Cycle I (Ciclo de vida iterativo I). The cycle consists of three iterations (Iteración 1, Iteración 2, Iteración 3). Each iteration follows a waterfall model: **Análisis** → **Diseño** → **Implementación** → **Pruebas**. The output of each iteration is a version of the software: **Versión 1** (shaded), **Versión 2** (checkered), and **Versión 3** (dotted). A cloud labeled **Requisitos** is shown above the first iteration, indicating the starting point for the first cycle.

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

8



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- Ciclo de vida iterativo (II)
 - Las primeras versiones pueden ser prototipos que se desechan posteriormente.
 - Los ciclos se repiten hasta obtener un producto satisfactorio.
 - Los usuarios deben evaluar el producto en cada iteración y proponer mejoras.

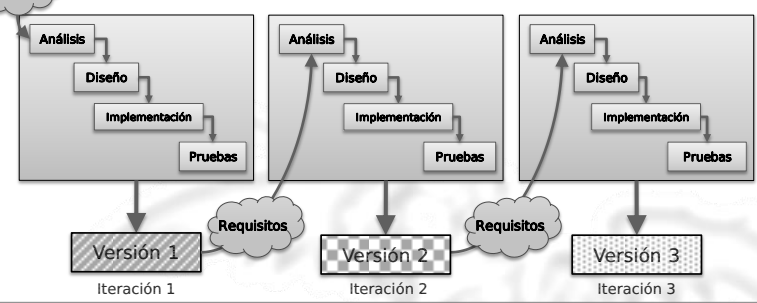


Diagram illustrating the Iterative Cycle II (Ciclo de vida iterativo II). The cycle consists of three iterations (Iteración 1, Iteración 2, Iteración 3). Each iteration follows a waterfall model: **Análisis** → **Diseño** → **Implementación** → **Pruebas**. The output of each iteration is a version of the software: **Versión 1** (shaded), **Versión 2** (checkered), and **Versión 3** (dotted). A cloud labeled **Requisitos** is shown above the first iteration, indicating the starting point for the first cycle.

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

9

IISSI

5

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- Incremental frente a iterativo

Don't Know What I Want, But I Know How to Get It
Jeff Patton, 2008. <http://jpattonassociates.com/>

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

10

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- Sobre los prototipos...

- Su uso no es exclusivo del ciclo de vida iterativo.
- Se pueden usar como una herramienta para obtener y validar los requisitos de clientes y usuarios en cualquier ciclo de vida.
- Lo habitual es usar **prototipos de interfaz de usuario**, que pueden reutilizarse (ejecutables) o desecharse (usualmente en papel, denominados **mockups**).

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

11

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- Sobre los prototipos...
 - Siempre se debe evaluar si el **esfuerzo** de desarrollo del prototipo merece la pena.
 - Es fundamental la implicación de los **usuarios**.
 - Siempre se debe tener en cuenta que el prototipo no es el producto final, ya que su **calidad** no suele ser la necesaria.
 - Otro tipo de prototipos, los **funcionales**, se utilizan para evaluar diferentes algoritmos antes de tomar decisiones de diseño.

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

12

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software


- Ciclo de vida en los métodos ágiles
 - Son ciclos de vida evolutivos con **iteraciones de corta duración** (2 semanas a 2 meses) para favorecer la comunicación con clientes y usuarios.
 - En cada iteración se incorporan nuevas peticiones de clientes y usuarios (requisitos).

```
graph TD
    subgraph Iteración_1 [Iteración 1]
        A1[Análisis] --> D1[Diseño]
        D1 --> I1[Implementación]
        I1 --> P1[Pruebas]
    end
    subgraph Iteración_2 [Iteración 2]
        A2[Análisis] --> D2[Diseño]
        D2 --> I2[Implementación]
        I2 --> P2[Pruebas]
    end
    subgraph Iteración_3 [Iteración 3]
        A3[Análisis] --> D3[Diseño]
        D3 --> I3[Implementación]
        I3 --> P3[Pruebas]
    end
    R1((Requisitos)) --> A1
    P1 --> V1[Versión 1]
    V1 --> R2((Requisitos))
    R2 --> A2
    P2 --> V2[Versión 2]
    V2 --> R3((Requisitos))
    R3 --> A3
    P3 --> V3[Versión 3]
```

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

13



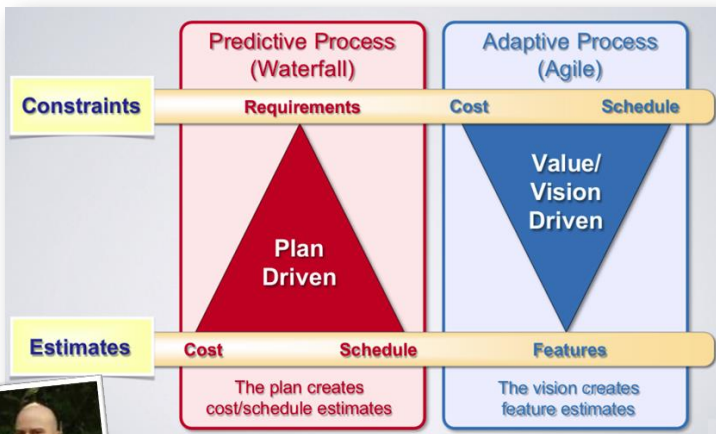
UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

- Concepto de ciclo de vida
- Ciclo de vida clásico
- Ciclos de vida evolutivos
- Ciclos de vida ágiles
- Ciclo de vida del Proceso Unificado
- Ciclo de vida en Métrica 3
- Pruebas en el ciclo de vida
- Ingeniería inversa
- Reingeniería del software

El Ciclo de Vida del Software

- Ciclo de vida en los métodos ágiles
 - Desarrollo ágil frente a tradicional




The diagram compares two software development processes: Predictive Process (Waterfall) and Adaptive Process (Agile).

Predictive Process (Waterfall):

- Top:** Predictive Process (Waterfall)
- Left:** Constraints
- Right:** Requirements
- Center:** Plan Driven (represented by a red triangle pointing up)
- Bottom:** Estimates
- Bottom Text:** The plan creates cost/schedule estimates

Adaptive Process (Agile):

- Top:** Adaptive Process (Agile)
- Left:** Cost
- Right:** Schedule
- Center:** Value/ Vision Driven (represented by a blue triangle pointing down)
- Bottom:** Features
- Bottom Text:** The vision creates feature estimates




Waterfall, Agile & the "Triple Constraint"
Tom Sylvester, 2013. <http://tom-sylvester.com/>

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

14



UNIVERSIDAD DE SEVILLA

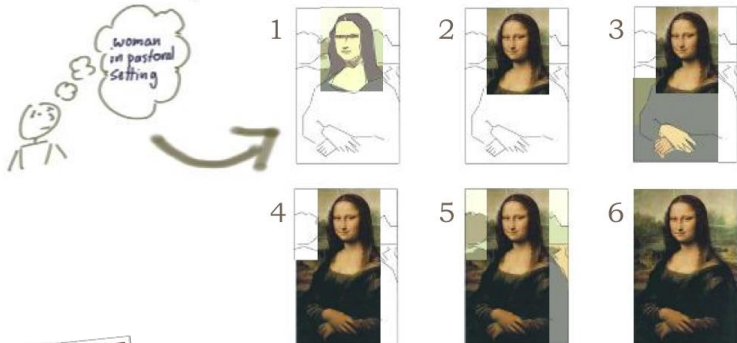
Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos


El Ciclo de Vida del Software

- Ciclo de vida en los métodos ágiles
 - Iterativo e incremental a la vez

1. Concepto de ciclo de vida
2. Ciclo de vida clásico
3. Ciclos de vida evolutivos
4. Ciclos de vida ágiles
5. Ciclo de vida del Proceso Unificado
6. Ciclo de vida en Métrica 3
7. Pruebas en el ciclo de vida
8. Ingeniería inversa
9. Reingeniería del software



The diagram illustrates the iterative and incremental software development cycle using the Mona Lisa painting. It shows 6 stages of the painting being completed in small increments, with a thought bubble indicating "woman in pastoral setting".




Revisiting the Iterative Incremental Mona Lisa
 Steven Thomas, 2012. <http://itsadeliverything.com>

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

15



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

• El manifiesto ágil de 2001...

Manifiesto for Agile Software Dev.

AGILE

• INDIVIDUALS AND INTERACTIONS OVER PROCESSES AND TOOLS

• WORKING SOFTWARE OVER COMPREHENSIVE DOCUMENTATION


• CUSTOMER COLLABORATION OVER CONTRACT NEGOTIATION

• RESPONDING TO CHANGE OVER FOLLOWING A PLAN

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

16



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

• El manifiesto ágil de 2001...

– Individuos e interacciones sobre procesos y herramientas

– Software que funcione sobre documentación detallada

– Colaboración con el cliente sobre negociación de contratos

– Respuesta al cambio sobre seguimiento de un plan

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

17

IISSI

9

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- El manifiesto ágil de 2001...
 - Los métodos ágiles **no** son escribir código desde el primer momento sin planificar ni documentar.

Panel 1: "WE'RE GOING TO TRY SOMETHING CALLED AGILE PROGRAMMING."
Panel 2: "THAT MEANS NO MORE PLANNING AND NO MORE DOCUMENTATION. JUST START WRITING CODE AND COMPLAINING."
Panel 3: "I'M GLAD IT HAS A NAME." "THAT WAS YOUR TRAINING."

Fuente: <http://dilbert.com/strips/2007-11-26>

Manifiesto for Agile Software Dev
AGILE
• INDIVIDUALS AND INTERACTIONS OVER PROCESSES AND TOOLS
• WORKING SOFTWARE OVER COMPREHENSIVE DOCUMENTATION
• COLLABORATION OVER CONTRACT NEGOTIATION
• RESPONDING TO CHANGE OVER FOLLOWING A PLAN

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

18

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software


- Individuos e interacciones sobre procesos y herramientas
 - La gente es el principal factor de éxito de un proyecto 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.
 - Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.

Manifiesto for Agile Software Dev
AGILE
• INDIVIDUALS AND INTERACTIONS OVER PROCESSES AND TOOLS
• WORKING SOFTWARE OVER COMPREHENSIVE DOCUMENTATION
• COLLABORATION OVER CONTRACT NEGOTIATION
• RESPONDING TO CHANGE OVER FOLLOWING A PLAN

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

19



UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática


Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles



5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3



7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software


- Software que funcione sobre documentación detallada
 - La regla a seguir es “*no producir documentos a menos que sean necesarios de forma inmediata para tomar un decisión importante*”.
 - Estos documentos deben ser cortos y centrarse en lo fundamental.



septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

20



UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática


Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles



5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3



7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- Colaboración con el cliente frente a negociación de contratos
 - Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo.
 - Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.



septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

21

IISSI

11

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

• Respuesta al cambio sobre seguimiento de un plan

– La habilidad de responder a los cambios que puedan surgir a los largo del proyecto (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.

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

22

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

• Desarrollo ágil vs. desarrollo tradicional

Ágil	Tradicional
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparados para cambios durante el proyecto	Presenta cierta resistencia al cambio
Impuesta internamente	Impuesta externamente
Proceso menos controlado, con pocos principios	Proceso muy controlado, con numerosas políticas y normas
El contrato es flexible	Contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa formalmente en reuniones
Equipos pequeños y/o en contacto físico	Grupos grandes y/o distribuidos
Pocos artefactos	Numerosos artefactos
Pocos roles	Numerosos roles
Menor énfasis en la arquitectura	Arquitectura y modelos fundamentales

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

23

IISSI

12

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- Técnicas de apoyo para los métodos ágiles
 - Refactorización (*refactoring*)
 - Mejoras sobre el código fuente sin cambiar su funcionalidad.
 - Pruebas automáticas
 - Pruebas programadas en lugar de realizadas a mano.
 - Integración continua
 - Automatización de la compilación y ejecución de pruebas automáticas.
 - Gestión de configuración
 - Especialmente diseñada para apoyar la interacción y la integración continua.

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

24

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- Metodología *Scrum*
 - Metodología ágil más **usada** actualmente.
 - Se basa en iteraciones de 30 días, *sprints*.
 - Durante un sprint:
 - Se produce código potencialmente entregable.
 - No se admiten cambios ni de requisitos ni de miembros del equipo de desarrollo

<http://heliosobjects.com/2014/02/20/requirements-management-in-agile-projects/>

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

25

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

ScrumAlliance
transforming the world of work

- Metodología *Scrum*
 - **Agile meeting:** reuniones cortas y frecuentes donde cada miembro del equipo expone:
 - Qué ha hecho desde la última reunión
 - Qué problemas ha tenido
 - Qué va a desarrollar hasta la próxima reunión
 - **Backlog:** lista priorizada de tareas
 - Reemplaza a los diagramas de Gantt
 - **Backlogs** de **producto** y **backlogs** de **iteración**

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

26

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

Modelo de procesos de *Scrum*

The diagram illustrates the Scrum process. It starts with a 'Product Backlog' (represented by a stack of blocks) which feeds into a 'Sprint Backlog' (represented by a stack of blocks). The 'Sprint Backlog' leads into a 'Sprint' cycle, which is represented by a large circular arrow. Inside the 'Sprint' cycle, there is a smaller circular arrow labeled 'Daily Scrum' with '24 hours' written inside. The 'Sprint' cycle is labeled '1-4 weeks'. The output of the 'Sprint' is a 'Working Increment of the product' (represented by a blue pyramid). A 'ScrumAlliance' logo is present in the bottom right corner of the slide content.

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

27

IISSI

14

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida de Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

• Ciclo de vida del Proceso Unificado

– Proceso iterativo e incremental propuesto por los creadores de UML.

– Define 6 fases: *inicio*, *elaboración*, *construcción*, *transición*, *producción* y *retirada*.

Disciplines

Business Modeling

Requirements

Analysis & Design

Implementation

Test

Deployment

Configuration & Change Management

Project Management

Environment

Operations & Support

Enterprise Management

Phases

Inception

Elaboration

Construction

Transition

Production

Retirement

Iterations

Initial

Elab #1

Elab #2

Const #1

Const #2

Const #N

Tran #1

Tran #2

Production

Ret #1

Ret #2

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

28

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida de Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

• Ciclo de vida del Proceso Unificado

– En cada fase se producen una o más iteraciones y se obtiene una versión evaluable del software.

Desarrollo

Phases

Inception

Elaboration

Construction

Transition

Production

Retirement

Iterations

Initial

Elab #1

Elab #2

Const #1

Const #2

Const #N

Tran #1

Tran #2

Production

Ret #1

Ret #2


septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

29

IISSI


15



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

1. Concepto de ciclo de vida
2. Ciclo de vida clásico
3. Ciclos de vida evolutivos
4. Ciclos de vida ágiles
5. Ciclo de vida del Proceso Unificado
6. Ciclo de vida en Métrica 3
7. Pruebas en el ciclo de vida
8. Ingeniería inversa
9. Reingeniería del software

El Ciclo de Vida del Software




- Ciclo de vida en Métrica 3
 - Metodología oficial de las Administraciones Públicas en España.
 - Métrica 3 permite aplicar diferentes ciclos de vida.
 - Sus procesos básicos son:
 - Plan de Sistemas de Información (PSI)
 - Desarrollo de Sistemas de Información
 - Estudio de Viabilidad del Sistema (EVS)
 - Análisis del Sistema de Información (ASI)
 - Diseño del Sistema de Información (DSI)
 - Construcción del Sistema de Información (CSI)
 - Implantación y Aceptación del Sistema (IAS)
 - Mantenimiento de Sistemas de Información (MSI)

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información


30



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

1. Concepto de ciclo de vida
2. Ciclo de vida clásico
3. Ciclos de vida evolutivos
4. Ciclos de vida ágiles
5. Ciclo de vida del Proceso Unificado
6. Ciclo de vida en Métrica 3
7. Pruebas en el ciclo de vida
8. Ingeniería inversa
9. Reingeniería del software

El Ciclo de Vida del Software



- Ciclo de vida en Métrica 3
 - También incluye procesos de apoyo
 - Gestión de proyectos
 - Seguridad
 - Gestión de la Configuración
 - Aseguramiento de la Calidad
 - Puede descargarse desde <http://administracionelectronica.gob.es>

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

31

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- El modelo en V
 - Asocia un tipo de pruebas a cada producto de cada fase según su nivel de abstracción.

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

32

Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3

7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- Ingeniería inversa
 - A veces es necesario mantener sistemas heredados (**legacy systems**) de los que no se dispone de documentación.
 - Consiste en analizar el resultado de una fase del desarrollo de software para obtener el resultado de la anterior, normalmente analizar el código para obtener el diseño.

septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

33

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Concepto de ciclo de vida

2. Ciclo de vida clásico

3. Ciclos de vida evolutivos

4. Ciclos de vida ágiles

5. Ciclo de vida del Proceso Unificado

6. Ciclo de vida en Métrica 3


7. Pruebas en el ciclo de vida

8. Ingeniería inversa

9. Reingeniería del software

El Ciclo de Vida del Software

- Reingeniería
 - La reingeniería utiliza la información obtenida por la ingeniería inversa para aplicar cualquier tipo de mantenimiento.
 - El mantenimiento preventivo del **efecto 2000** ha sido el mayor esfuerzo de ingeniería inversa y reingeniería en la historia de la Ingeniería del Software hasta la fecha.



septiembre 2016

Introducción a la Ingeniería del Software y los Sistemas de Información

34