

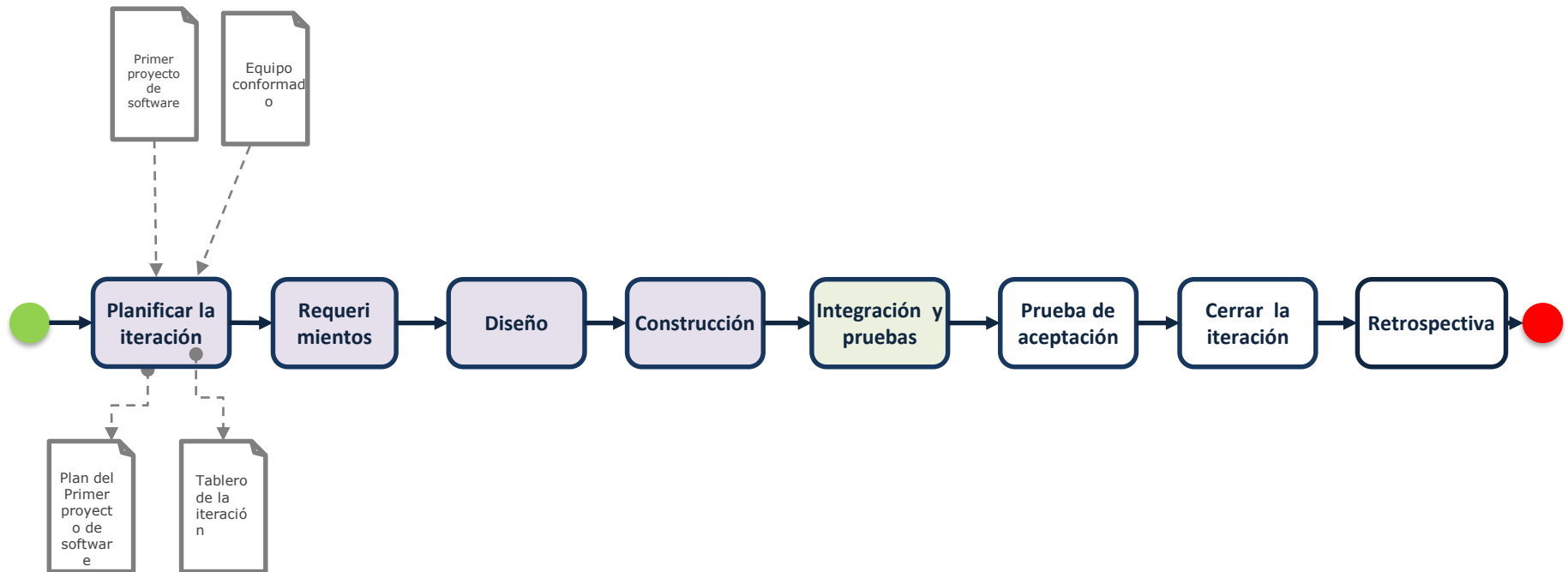
Unidad 8

Integración y pruebas de software

Guadalupe Ibargüengoitia

Hanna Oktaba

Actividades de la iteración



Entrada a esta unidad

- **Condiciones**

- Construcción del software por casos de uso completo

- **Productos de trabajo**

- Código de los componentes del software por caso de uso probados individualmente.
- Plantilla del documento de *Integración y Pruebas de Software*.
- *Repositorio* compartido con la carpeta para la iteración actualizado
- *Tablero* de la iteración actualizado

Integración de software

- El **objetivo** de la integración del software es comprobar que los componentes construidos en la iteración **funcionan bien juntos** y también cuándo se **incorporan al software de iteraciones anteriores**.
- Se llama ***configuración del software*** al conjunto de todos los componentes desarrollados en las iteraciones hasta este momento del proyecto.

Integración continua de software

- En el desarrollo de software iterativo, la integración de software es continua conforme se van generando los componentes.
- El orden de la integración de los componentes se basa principalmente en la arquitectura.
- Lo que implica que conforme se van obteniendo componentes probados de cada nivel de abstracción, se van integrando.(SWEBOK 2014)

Actividades para la integración del software

- Preparar la integración del software
- Realizar la integración del software

Preparar la integración del software

- Para efectuar la integración de la nueva configuración del software se incluyen los componentes realizados en la iteración.
- **Se establecer una estrategia de integración:** en qué orden se van incorporando los códigos de los casos de uso en un solo código.
- **Preparar el ambiente de prueba en alguna computadora del equipo.**
- **Asegurarse que estén todos los componentes en el repositorio del equipo en la última versión aprobada.**

Realizar la integración del software

- Siguiendo la estrategia establecida por el equipo, se van incorporando los casos de uso en una sola configuración del software.

Pruebas del sistema integrado de software

- La prueba de software consiste en la verificación dinámica del comportamiento de un programa en un conjunto finito de casos, adecuadamente seleccionados, contra el comportamiento esperado.
- Permite evaluar la calidad del producto de la iteración y mejorarlo al identificar defectos y problemas.

Pruebas del sistema integrado de software

- En la prueba del sistema integrado de software, conocida también como prueba del sistema, se tratará de comprobar que el sistema integrado de software cumple con todos los requerimientos establecidos tanto los funcionales como los no funcionales.

Aspectos a probar

- **Funcionalidad.** Que cumpla con los requerimientos funcionales especificados.
- **Usabilidad.** Si los usuarios finales lo encuentran útil para apoyarlos en sus actividades y qué tan fácilmente se recupera de errores de usuario.
- **Eficiencia.** La capacidad de respuesta es adecuada.
- **Seguridad.** Cumpla con sus requerimientos de seguridad.
- **Estrés.** Ejecutar el software a su máxima capacidad.
- **Recuperación.** Que tan fácilmente se recupera el software de algún error.

Actividades de pruebas de sistema

- **Preparar las pruebas a realizar.** Decidir qué se probará, en qué orden y preparar el ambiente para efectuar las pruebas.
- **Ejecutar las pruebas.** Se efectúan las pruebas guiadas por **casos de pruebas** y se identifican los defectos.
- **Corregir los defectos.** Se asigna al responsable de corregir los defectos y se asegura que se corrijan.
- **Verificar la corrección de defectos.** Se realizan las pruebas de regresión.

Prueba de regresión

- Se vuelven a aplicar las mismas pruebas para asegurarse que el defecto fue eliminado y que su corrección no ha inyectado otros defectos.
- Las *pruebas de regresión* sirven para detectar defectos al hacer cambios a componentes ya probados o detectar defectos causados por cambios como efectos laterales o por correcciones mal hechas.

Práctica de Integración y Pruebas de Software

Práctica	
Integración y pruebas de software	
Objetivos	
<ul style="list-style-type: none">Integrar los componentes de software contruidos, probar y corregir el software integrado hasta asegurar que cumple con los requerimientos especificados.	
Entrada	Resultado
Condiciones <ul style="list-style-type: none">Construcción del software completo Productos de trabajo <ul style="list-style-type: none">Código de los componentes del software probados individualmente.Documento de <i>Construcción de Software</i>.Plantilla del documento de <i>Integración y Pruebas de Software</i><i>Repositorio</i> compartido con la carpeta para la iteración actualizado<i>Tablero</i> de la iteración actualizado	Condiciones <ul style="list-style-type: none">Integración del software completa y probada Productos de trabajo <ul style="list-style-type: none">Código de los componentes del software probados integradosPrueba del software efectuada.Documento de <i>Integración y Pruebas de Software</i>.<i>Repositorio</i> compartido con la carpeta para la iteración actualizado<i>Tablero</i> de la iteración actualizado

Práctica de Integración y Pruebas de Software

Actividades

1. Integrar el producto de software (TI1)
2. Probar el sistema integrado de software (TI2)
3. Integrar el documento de *Integración y Pruebas de Software* (TI3)

Herramientas

- Herramientas de integración y pruebas

Actividad I1. Integrar el producto de software

- En cada iteración de desarrollo se genera una nueva configuración del software que incluye los componentes de la iteración anterior más los nuevos y/o modificaciones efectuados en esta iteración.

Técnica TI1: Integración de software

1. Establecer una estrategia de integración: en qué orden se van incorporando los códigos de los casos de uso en un solo código.
2. Preparar el ambiente de prueba en alguna computadora del equipo.
3. El **Responsable de la colaboración** deberá revisar que estén todos los componentes en el repositorio del equipo en la última versión aprobada.
4. El **Responsable técnico** coordina las actividades de integración, según esa estrategia, y todo el equipo participa.

1. Establecer una estrategia de integración

- El **Responsable técnico** coordina la discusión de la elección de la estrategia.
- Si la construcción fue a través de casos de uso, se sigue como estrategia el uso del Diagrama de navegación construido en el diseño.
- Se empieza con el primer estado que representa a la pantalla inicial.
- A partir de ese estado inicial se integran los componentes de los casos de uso para los que existen transiciones en el diagrama de navegación uno por uno, compilando y probando esa integración.

2. Integrar

- Se puede revisar el diagrama de navegación e ir ejecutando cada rama del modelo, asegurándose que se abran las pantallas adecuadas y se manden los mensajes de error correspondientes.
- Se usan los casos de prueba que se aplicarán para verificar la correcta interacción entre estos componentes.
- Si se encuentran defectos, se registran y se asigna a cada uno el responsable de su corrección. Después de la corrección se vuelve a probar.

- El **Responsable de calidad** personaliza la plantilla de *Integración y Prueba de Software* (liga a la plantilla de *Integración y Prueba de Software*).
- El **Responsable técnico**, pone una tarjeta en el Tablero en la columna de *Por Hacer*, con la actividad *Integrar el software*, mueve la tarjeta a la columna de *Haciendo* cuando lo está haciendo con todo el equipo y mueve la tarjeta a la columna *Hecho* al terminarlo.
- El **Responsable de la colaboración** deberá revisar que estén todos los componentes en el repositorio del equipo en la última versión aprobada.
- El **Responsable técnico** coordina las actividades de integración y todo el equipo participa.

Actividad I2. Pruebas del sistema integrado de software

- Cuando ya se tiene integrado todo el sistema en la configuración del software, se prueba que cumpla con sus requerimientos funcionales y no funcionales.
- Éste es el objetivo de la prueba del sistema.

Técnica T12: Pruebas del sistema integrado de software

- Se sugiere **revisar casos de prueba** definidos en los Requerimientos y asegurarse que son suficientes para probar los casos de uso en su flujo normal, alternativos y excepcionales. Si falta probar algo, **se definen nuevos casos de prueba**.

Probar el software integrado

- Cada responsable de algún caso de uso inicia las pruebas con sus casos de prueba y registra los defectos.

Registro de defectos

Nombre del caso de uso	Responsable de prueba	Identificador del caso de prueba	Defecto identificado
Total de defectos			#

Corregir los defectos

- Terminadas las pruebas, se decide qué defectos se corregirán. Algunos pueden dejarse para la siguiente iteración, siempre y cuando el software cumpla con los objetivos de esta iteración.
- Cada desarrollador corrige sus defectos y vuelve a probar el software para asegurarse de su corrección (prueba de regresión).

- El **Responsable técnico**, pone una tarjeta en el Tablero en la columna de *Por Hacer*, con la actividad de *Probar el software y corregir los defectos*, participan todos los desarrolladores con sus casos de uso ya integrados.
- El **Desarrollador** mueve la tarjeta a la columna de *Haciendo* cuando lo está haciendo y mueve la tarjeta a la columna *Hecho* al terminarlo.
- El **Responsable de la colaboración** se asegura que se tenga la configuración completa y ya sin defectos.

Actividad I3. Integrar el documento de *Integración y Pruebas del Software*

- Una vez aplicadas todas las técnicas anteriores, se ha integrado el código de todos los casos de uso y se ha probado que funcione bien juntos, según lo especificado en los Requerimientos. Se prueba el funcionamiento de todo el software asegurando que cumple los requerimientos funcionales y no funcionales.

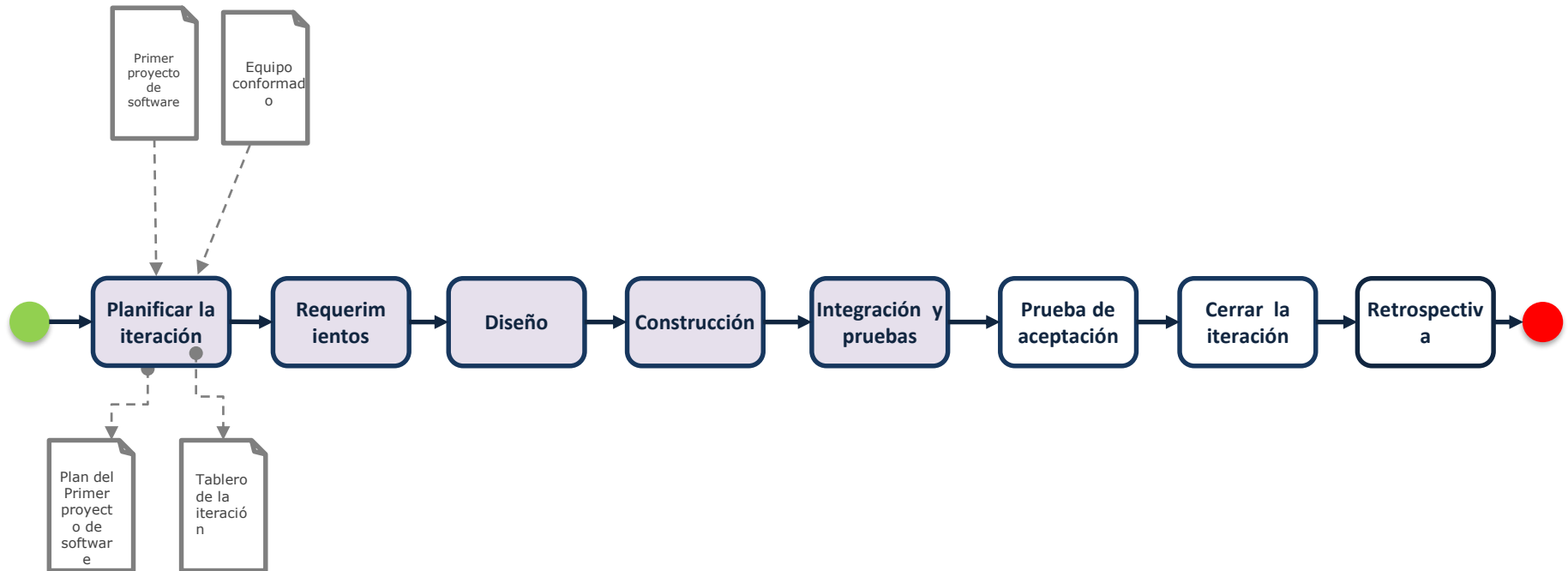
Técnica TI3. Integrar el documento de *Integración y Pruebas del Software*

- El **Responsable de calidad** unifica los productos de las actividades de la práctica en el documento de *Integración y pruebas de software*. Se asegura que el documento de *Integración y Pruebas de Software* está completo según la plantilla personalizada.

Resultado de esta unidad

- **Condiciones**
 - Integración del software completo y probado
- **Productos de trabajo**
 - Código de los componentes del software integrado
 - Prueba de software integrado efectuada
 - Documento de *Integración y Prueba de Software*.
 - *Repositorio* compartido con la carpeta para la iteración actualizado
 - *Tablero* de la iteración actualizado

Actividades de la iteración



Bibliografía

Binder R.V. (2000). *Testing Object-Oriented Systems. Models, patterns and Tools*. Addison Wesley.

Dusrin E., T. G. (2009). *Implementing Automated Software Testing*. Addison Wesley.

ISO/IEC29110. (2011). *29110-5-1-2Software Engineering-lifecycle Profiles for Very Small Entities Management and Engineering Guidde. s.1*. Software Engineering.

Ruiz B. (2013). Marcando la pauta para las pruebas ágiles. *Software Gurú*, no. 38, 18-20.

SWEBOK 3.0. (2014). *Guide to the Software Engineering Body of Knowledge v3.0*. IEEE Computer Society.