

Curso de Ingeniería de Software

Unidad 5

Requerimientos de software

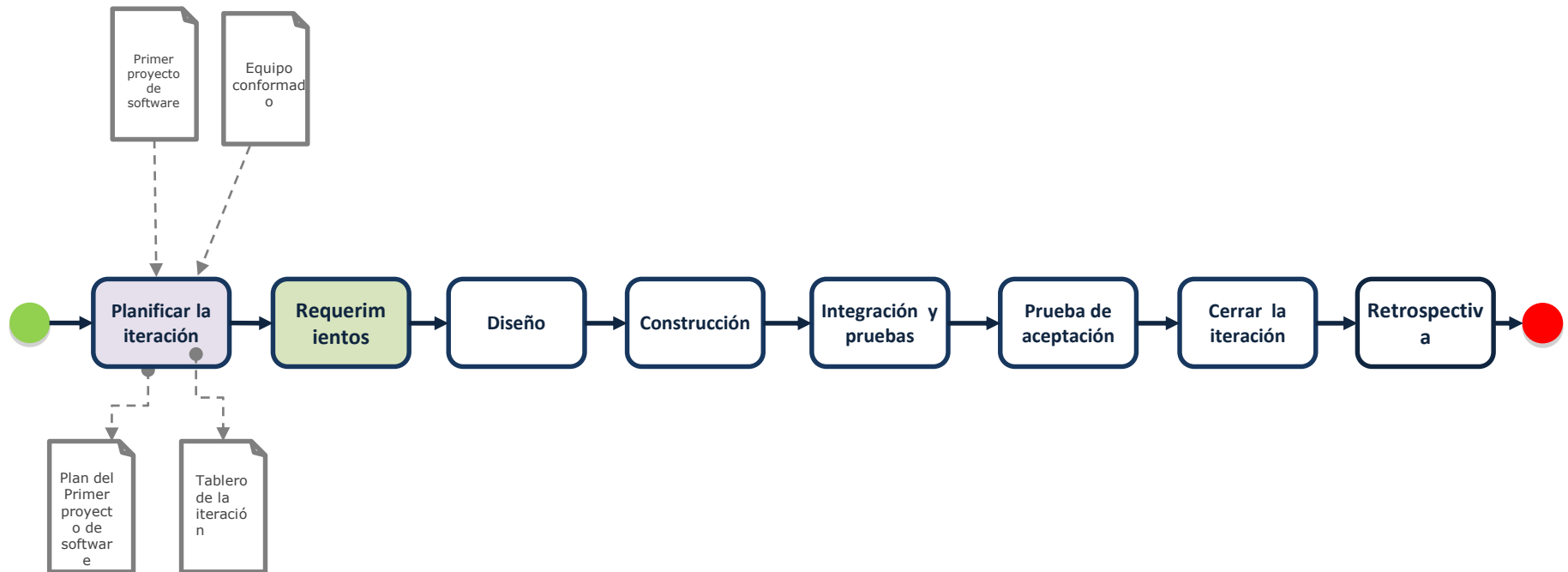
Guadalupe Ibargüengoitia

Hanna Oktaba

Entradas a esta unidad

- **Condiciones**
 - Proyecto planificado
-
- **Productos de trabajo**
 - *Plan del proyecto*
 - *Planteamiento de necesidades*
 - *Plantilla Especificación de Requerimientos de Software*
 - *Repositorio compartido con la carpeta para la iteración*
 - *Tablero de la iteración*

Actividades de la iteración



¿Qué es un requerimiento?

- "Los requerimientos de software expresan las **necesidades y restricciones** que debe tener un producto de software para que contribuya a la solución de un problema en el mundo real" (SWEBOK, 2014).
-
- “Los requerimientos de un producto de software son las descripciones de lo que hará, los servicios que proporcionará y las restricciones de su operación” (Sommerville I., 2011).
-
- Un requerimiento o necesidad es lo que el cliente o un usuario desean que haga el software para resolver un problema

¿Cómo se expresan los requerimientos?

- Lenguaje natural – texto
- Forma gráfica – diagramas
- Prototipo de interfaz de usuario – pantallas

Objetivos de la especificación de requerimientos

- Entender el problema a resolver
- Construir un modelo de los requerimientos
- Obtener criterios para verificar y validar el producto de software terminado

Calidad de un producto de software

- La calidad de un producto de software se define como “capacidad de un producto de software de **satisfacer las necesidades** expresas e implícitas en **condiciones específicas**” (SWEBOK, 2014).
- Al definir los requerimientos se busca **claridad, precisión y no ambigüedad** tanto para el desarrollador y para el cliente.

¿Cómo asegurar la calidad de la especificación de los requerimientos?

- El cliente pueda evaluar la calidad de software, es revisando y avalando la especificación de los requerimientos.
- La validación final, de que el software cumple con los requerimientos especificados, se hace hasta que el software esté construido.
- Los requerimientos cambian constantemente

Tipos de requerimientos

- Funcionales
 - Son los servicios, la forma en que el software debe reaccionar a entradas particulares y cómo debe comportarse en situaciones particulares
- No-Funcionales
 - Son restricciones que los servicios y funciones de software deben cumplir

Requerimientos No-Funcionales

- **Las necesidades de la interfaz externa:** hardware y software, comunicaciones, facilidades de uso requeridas por los usuarios
- **Cualidades del software :** confiabilidad, eficiencia, seguridad, compatibilidad, portabilidad, mantenimiento
- **Restricciones del diseño:** formatos, lenguajes, estándares, herramientas
- **Otros:** tipo de base de datos, instalación, capacitación

Actividades para definir los requerimientos

- **Obtener** – Entender (desarrollador/cliente)
- **Analizar** (desarrollador)
- **Especificar** – Documentar texto, diagramas, prototipos (desarrollador)
- **Validar** (por el cliente)
- **Administrar** – Cambios a requerimientos (desarrollador)

Casos de prueba

- Para todos los **casos de uso** se definen **casos de prueba**
- Los **casos de prueba** se aplicarán cuando el **caso de uso ya esté implementado** para asegurar el cumplimiento de los requerimientos especificados
- Los **casos de prueba** confirman la **verificabilidad** de los requerimientos

Método Inicial de Desarrollo de Software

- Práctica de desarrollo PD1

Requerimientos de Software

Práctica de desarrollo de software

PA1	Práctica	
Requerimientos de software		
Objetivos		
<ul style="list-style-type: none">Comprender y especificar los requerimientos del sistema de software, basándose en el <i>Planteamiento de necesidades</i>.		
Entrada		Resultado
Condiciones <ul style="list-style-type: none">Proyecto planificado Productos de trabajo <ul style="list-style-type: none"><i>Plan del proyecto</i><i>Planteamiento de necesidades</i><i>Plantilla Especificación de Requerimientos de Software</i><i>Repositorio compartido con la carpeta para la iteración</i><i>Tablero de la iteración</i>		Condiciones <ul style="list-style-type: none">Requerimientos entendidos Productos de trabajo <ul style="list-style-type: none"><i>Especificación de Requerimientos de Software</i>

Curso de Ingeniería de Software

Práctica de desarrollo 1

Requerimientos de Software

Actividades

1. Entender el *Planteamiento de necesidades* para la iteración (TR1)
2. Analizar y especificar los requerimientos funcionales (TR2, TR3, TR4, TR5)
3. Especificar los requerimientos no funcionales (TR6)
4. Documentar la *Especificación de Requerimientos de Software* (TR7)

Herramientas

- Herramienta para hacer diagramas de UML
- Herramienta para diseñar el prototipo de interfaz de usuario

A1. Entender el *Planteamiento de necesidades* para la iteración

- A fin de entender el problema para poder especificar los requerimientos se pueden llevar a cabo estas actividades:
 - Hacer **entrevistas** con posibles usuarios
 - Aplicar **cuestionarios** a los usuarios
 - **Observar** a los futuros usuarios al realizar las tareas que apoyará el software
 - **Revisar documentos o sistemas** ya existentes que se pretenden mejorar
 - Hacer un **glosario de términos**.

Técnica R1. Entender el problema

- Revisar documento de *Planteamiento de Necesidades*
- Hacer el glosario de términos

Ver en la Plantilla de Requerimientos

Técnica R1. Entender el problema

- Glosario de términos
 - Para que todos los involucrados puedan comunicarse más fácilmente, se recomienda construir un *Glosario de términos* para establecer un *vocabulario común*.
 - El glosario de términos es un pequeño *diccionario*, donde se registran los términos importantes para entender el problema y su significado.
 - El glosario se puede ir *actualizando a lo largo* del proyecto.

Glosario de términos

Término	Definición
Actor	Aquel que realiza la acción que define el caso de uso.
Amigo	Usuario a quien es sugerido otro Usuario para chatear por tener gustos en común.
Campo	Recurso que almacena un dato específico.
Caso de Uso	Forma de identificar una funcionalidad bien definida en el sistema.
Chatear	Intercambio de mensajes privados entre dos Usuarios.
Formulario	Documento diseñado para que el usuario introduzca datos estructurados en las zonas correspondientes, para ser almacenados y procesados posteriormente.
Perfil	Registro de la información personal de un Usuario.
Sistema	El producto de software en funcionamiento.

- El **Responsable de la colaboración**, pone una tarjeta en el *Tablero* en la columna de Por Hacer, con la actividad de **Hacer el *Glosario de términos*** y lo asigna al **Responsable técnico**.
- El **Responsable Técnico** mueve la tarjeta a la columna de *Haciendo* y dirige la discusión para hacer el *Glosario de términos*. Al terminar el glosario, mueve la tarjeta a la columna *Hecho*
- El **Responsable de calidad** se asegura que se ponga el glosario en la plantilla personalizada de *Especificación de Requerimientos de Software*.

A2. Analizar y especificar los requerimientos funcionales

- Los casos de uso permiten de manera simple **definir** como los usuarios **utilizarán el software**.
- Servirán de hilo conductor para todo el proceso de desarrollo en el ***Método Inicial de Desarrollo de Software***.

Técnica TR2. Hacer el *Diagrama de Casos de Uso para la Iteración*

- Para iniciar una iteración los desarrolladores deben de **revisar el *Plan de Proyecto***, que contiene el *Diagrama General de Casos de Uso de todo el sistema*.
- Todos participan en la selección de **los casos de uso para la iteración** y **hacen un diagrama general con el subconjunto** de casos de uso que se desarrollarán en esa iteración.
- Se **asigna al menos un caso de uso** a cada desarrollador.

- El **Responsable de la colaboración**, pone una tarjeta en el Tablero en la columna de *Por Hacer*, con la actividad de Hacer el diagrama de casos de uso de la iteración y lo asigna al **Responsable técnico**.
- El **Responsable técnico** mueve la tarjeta a la columna de *Haciendo*, revisa con todo el equipo el *Diagrama General de Casos de Uso de todo el sistema* para identificar los casos de uso seleccionados para la iteración y dibujan el diagrama general de casos de uso de la iteración.
- Al terminar el diagrama, **Responsable técnico** mueve la tarjeta a la columna *Hecho*.
- El **Responsable de calidad** asegura que se integre el diagrama en la **plantilla personalizada de Especificación de Requerimientos de Software**.

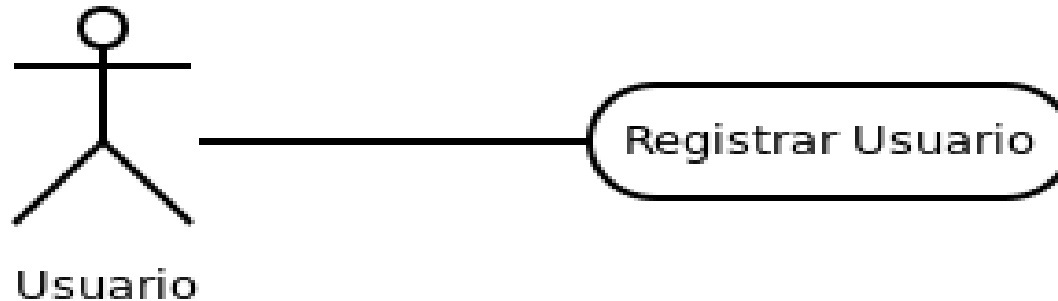
Técnica R3. Detallar los casos de uso (1)

- **Caso de uso “Nombre”**: El nombre deberá ser un verbo en infinitivo representativo de la funcionalidad del caso de uso (el mismo que en el diagrama general).
- **Diagrama del caso de uso**: dibujo del o los actores y el caso de uso en cuestión.
- **Descripción**: Texto breve describiendo la función que representa el caso de uso.
- **Precondiciones**: El estado en el que se encuentra el sistema para que el actor (o los actores) pueda iniciar el caso de uso.

Técnica R3. Detallar los casos de uso (2)

- **Flujo normal de eventos:** Tabla que describe el flujo de interacciones esperadas (*el camino feliz*) entre el actor y el sistema durante el caso de uso.
- **Flujo alternativo de eventos:** Tabla que describe el flujo de interacciones alternativas del camino normal en el caso de uso.
- **Flujo excepcional de eventos:** Tabla con las acciones que ocurren en situaciones anormales o excepcionales.
- **Poscondiciones:** Define el estado en el que se encuentra el sistema después de la terminación exitosa del caso de uso.

Caso de uso Registrar Usuario



Descripción: Permite al Usuario registrar sus datos en el sistema para poder hacer uso de éste.

Precondiciones: El Usuario no registrado está viendo la página principal de la red social con un botón de registro.

Flujo normal de eventos

Pasol	Acción Actor	Paso	Acción Sistema
1	Usuario presiona el botón de registrar.	2	Despliega campos para capturar correo electrónico y dos veces la contraseña.
3	Usuario llena los campos del formulario con sus datos. Presiona el botón de enviar.	4	Valida si el correo tiene @ y si las contraseñas son idénticas. Si es así, presenta formulario adicional para agregar mas datos: sexo, fecha de nacimiento, foto (opcional) y pasatiempos.
5	El Usuario llena los datos adicionales. Presiona el botón de enviar.	6	El sistema registra los datos del Usuario y despliega la página del inicio de Usuario registrado.

Flujos alternativos

ID	Acción Actor	Acción Sistema
A1	El Usuario ingresa un correo sin @ y/o contraseñas idénticas.	Informa al Usuario que no los datos son incorrectos y le pide que vuelva a introducirlos.
A2	El Usuario omite llenar algún campo.	Informa al Usuario que todos los campos son obligatorios y le pide que los llene para poder terminar el proceso.

Flujo excepcional de eventos

ID	Nombre	Acción Sistema
E1	Ocorre un error con el servidor al momento de intentar guardar el registro.	Aborta el proceso, informa que hubo un problema y sugiere que intente mas tarde.

Poscondiciones: El Usuario queda registrado en el sistema y queda desplegada la página de inicio del Usuario. En caso del flujo excepcional el Usuario queda en la página principal sin cumplir con el registro.

- Cada miembro del equipo escoge al menos un caso de uso, dentro del alcance de la iteración representado en el *Diagrama de Casos de Uso* para la Iteración, para detallarlo.
- El **Responsable de la colaboración**, pone una tarjeta en el **Tablero** en la columna de *Por Hacer*, con la actividad de *Detallar el caso de uso* para cada caso de uso de la iteración y se lo asigna al miembro del equipo que lo detallará.
- Cada miembro del equipo al iniciar el detalle de su caso de uso, mueve su tarjeta del tablero a la columna *Haciendo* y usa el formato propuesto para detallar el caso de uso del que es responsable. Al terminar de llenar todos los campos, mueve su tarjeta a la columna de *Hecho*.
- El **Responsable de calidad** asegura que se pongan los detalles de todos los casos de uso en la plantilla personalizada de *Especificación de Requerimientos de Software*.

Técnica R4. Definición del prototipo de interfaz de usuario

- Un **prototipo de interfaz de usuario** es una representación **inicial de las pantallas** que el software mostrará al usuario.
- Estas **pantallas** muestran la **distribución de diferentes elementos** que permitirán al usuario interaccionar con el sistema pero que **no ofrecen ninguna funcionalidad implementada**.

Prototipo de interfaz

- Para diseñar las pantallas se puede usar desde **papel y lápiz**, código **HTML**, o una **herramienta** que lo facilite*
- Ver la plantilla para el prototipo de interfaz humana.

* Pencil Project <http://pencil.evolus.vn/>



Nombre de Usuario:	Contraseña:
--------------------	-------------

Ver Postulados Postularse a Vacante Alta Vacante Modificar Vacante Eliminar Vacante Ver Vacantes

Alta de Vacante

Datos de la Vacante

Nombre: _____

Requisitos:

Descripción:

Sueldo: Horario:

Contacto:

Datos de la Empresa

NT 1 Correo

- El **Responsable de la colaboración**, pone una tarjeta en el *Tablero* en la columna de *Por Hacer*, con la actividad de *Definir el prototipo* para cada caso de uso de la iteración y se lo asigna al miembro del equipo que definirá la interfaz.
- Todo el equipo se pone de acuerdo en lineamientos para el diseño de la interfaz de usuario y escoge la manera para realizarlo.
- Cada miembro del equipo define el prototipo de interfaz que corresponde al detalle de su caso de uso para los flujos normal, alternativos y excepcionales.
- Cada miembro del equipo al iniciar la actividad, mueve su tarjeta del tablero a la columna *Haciendo* y usa el formato propuesto para definir el prototipo para el caso de uso del que es responsable. Al terminar la actividad, mueve su tarjeta a la columna de *Hecho*.
- El **Responsable de calidad** se asegura que los prototipos se integren en la plantilla personalizada de *Especificación de Requerimientos de software*.

Técnica R5: Casos de prueba para los requerimientos funcionales

- Los casos de prueba para cada caso de uso se documentan en una tabla.
- Un caso de prueba identifica las entradas al caso de uso y los resultados esperados
- Se consideran entradas para flujos normales, alternativos y excepcionales.

Ver en la plantilla de Requerimientos de Software como se documentan los casos de prueba

Caso de prueba 1 (flujo normal)

Entradas	Resultado esperado
luisoto976@gmail.com fpEo34Sw fpEo34Sw masculino 25/09/1990 perfjl.jpg música futbol cocinar	Se espera que se registre exitosamente el Usuario nuevo.

Caso de prueba 2 (flujo alternativo)

Entradas	Resultado esperado
luisoto976@gmail.com fpEo34Sw q1u5r4kw3e3 masculino 25/09/1990 perfjl.jpg música futbol cocinar	Se despliega un aviso al Usuario de que la confirmación de contraseña no es correcta.

Caso de prueba 3 (flujo excepcional)

Entradas	Resultado esperado
Antes de enviar los datos completos se pierde la conexión a la base de datos.	Se despliega un aviso al usuario que hubo una falla y que intente más tarde.

- El **Responsable de la colaboración**, pone una tarjeta en el Tablero en la columna de *Por Hacer*, con la actividad de Definir los casos de prueba para cada caso de uso de la iteración y se lo asigna al miembro del equipo que definirá sus casos de prueba.
- Cada miembro del equipo define los casos de prueba que corresponden al detalle de su caso de uso en los flujos normales, alternativos y excepcionales.
- Cada miembro del equipo al iniciar la actividad, mueve su tarjeta del tablero a la columna *Haciendo* y usa el formato propuesto para definir los casos de prueba para el caso de uso de los que es responsable. Al terminar la actividad, mueve su tarjeta a la columna de *Hecho*.
- El **Responsable de calidad** se asegura que se integren los casos de prueba en la plantilla personalizada de *Especificación de Requerimientos de software*.

A3. Especificar los requerimientos no funcionales.

- Cuando ya se tienen entendidas las funcionalidades que deberá ofrecer el sistema de software, se especifican las **características o requerimientos no funcionales** que deberá cumplir.

Requerimientos no funcionales

Interfaz con el usuario

La interfaz debe tener un diseño similar al de las redes sociales más populares.

Disponibilidad

Se espera que el sistema tenga una disponibilidad 24/7.

Eficiencia

Se espera un tiempo de respuesta de a lo más de tres segundos en cada petición.

Seguridad

El sistema tiene que cifrar las contraseñas, para evitar exponer información en caso de que la base de datos se comprometa

Mantenimiento

El sistema tiene que ser modular para poder facilitar el mantenimiento del mismo. Se tiene que generar la documentación en UML y el código debe tener comentarios de acuerdo a los estándares para poder brindar soporte a los desarrolladores en caso de mantenimiento.

Portabilidad

El sistema debe de construirse con Java, poder correrlo que cualquier ambiente que pueda usar este lenguaje.

Restricciones de diseño y construcción

Se solicita que la interfaz tenga un diseño parecido a las redes sociales más populares, además que sea adaptativo y responsivo.

- El **Responsable de la colaboración**, pone una tarjeta en el Tablero en la columna de *Por Hacer*, con la actividad de Especificar los requerimientos no funcionales y la asigna al **Responsable técnico**.
- Todos los miembros del equipo revisan el *Planteamiento de Necesidades* para la Iteración e identifican los requerimientos no funcionales en la sección de Restricciones.
- El **Responsable Técnico** redacta el texto con los requerimiento no funcionales.
- El **Responsable de calidad** se asegura que se integren los requerimientos no funcionales en la plantilla personalizada de *Especificación de Requerimientos de Software*.

A4. Documentar la Especificación de Requerimientos de Software

- Una vez aplicadas todas las técnicas anteriores, se han generado una serie de resultados que al juntarlos conforman la *Especificación de Requerimientos de Software*.

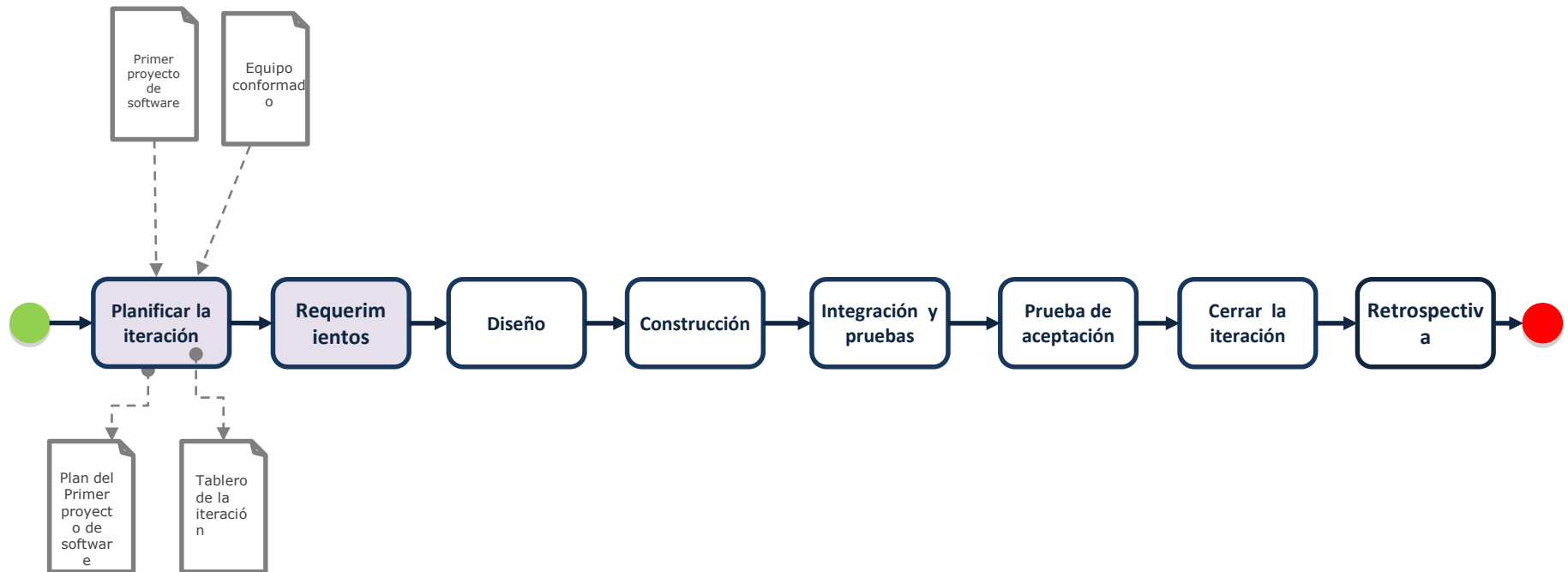
- **TR7. Documentar la *Especificación de Requerimientos de Software***

-

- La labor del **Responsable de calidad** es asegurar que fueron integrados todos los elementos generados en esta práctica en la plantilla personalizada del documento *Especificación de Requerimientos de Software*.

- El **Responsable de Calidad** asegura la integración de todos los elementos en el documento de *Especificación de Requerimientos de Software* siguiendo la plantilla.

Actividades de la iteración



Resultados de esta unidad

- **Condiciones**
 - Requerimientos de software entendidos
- **Productos de trabajo**
 - *Especificación de Requerimientos de Software* que incluye:
 - Diagrama general de casos de uso de la iteración
 - Glosario de términos
 - Detalle de los casos de uso
 - Prototipo de interfaz de usuario
 - Casos de prueba para los casos de uso
 - Requerimientos no funcionales

Bibliografía

- ISO/IEC25010. (2011). *Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models*. ISO.
- Jacobson I., C. M. (1992). *Object-Oriented Software Engineering. A Use Case Driven Approach*. Addison-Wesley.
- Sommerville I. (2011). *Software Engineering*. Pearson.
- SWEBOK. (2014). *Guide to the Software Engineering Body of Knowledge v3.0*. IEEE Computer Society.