

Ingeniería de Requisitos

Sesión 13

Trazabilidad de Requisitos

Prof. Ciro Rodriguez
crodriguezro@unmsm.edu.pe

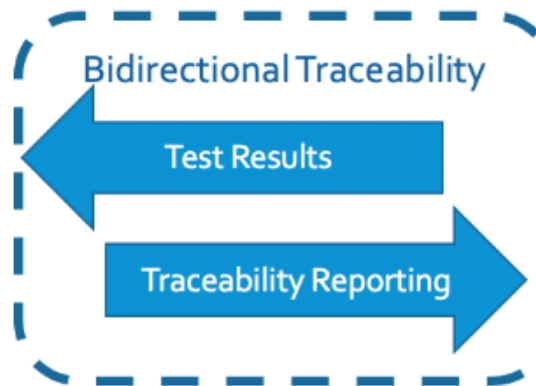
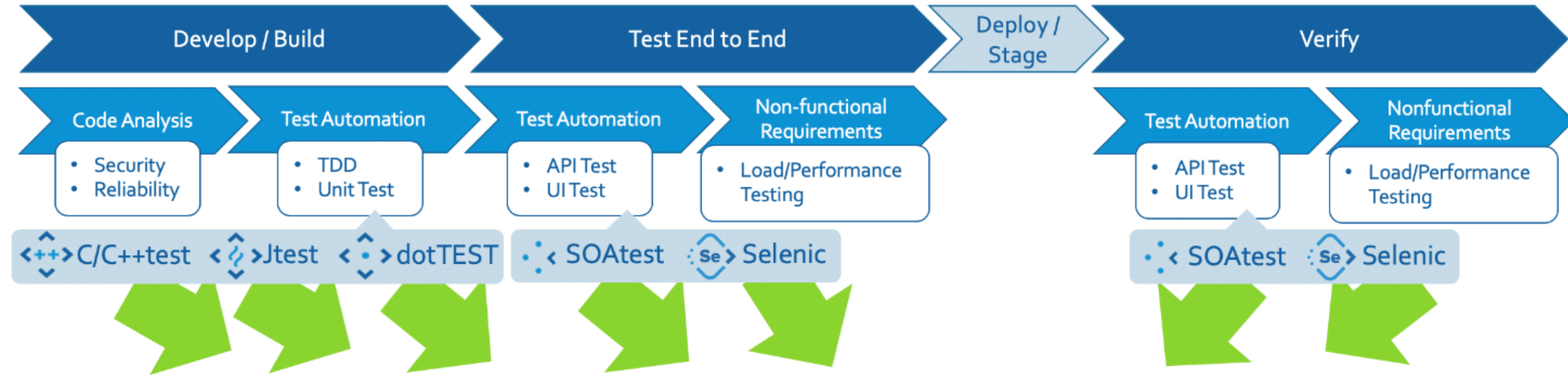
Tema: Trazabilidad de Requisitos

- Conceptos y Beneficios de la Trazabilidad de Requisitos
- Técnicas y Herramientas para Establecer la Trazabilidad de Requisitos
- Mantenimiento y Evolución de la Trazabilidad de Requisitos
- Herramientas y Automatización de la Trazabilidad de Requisitos

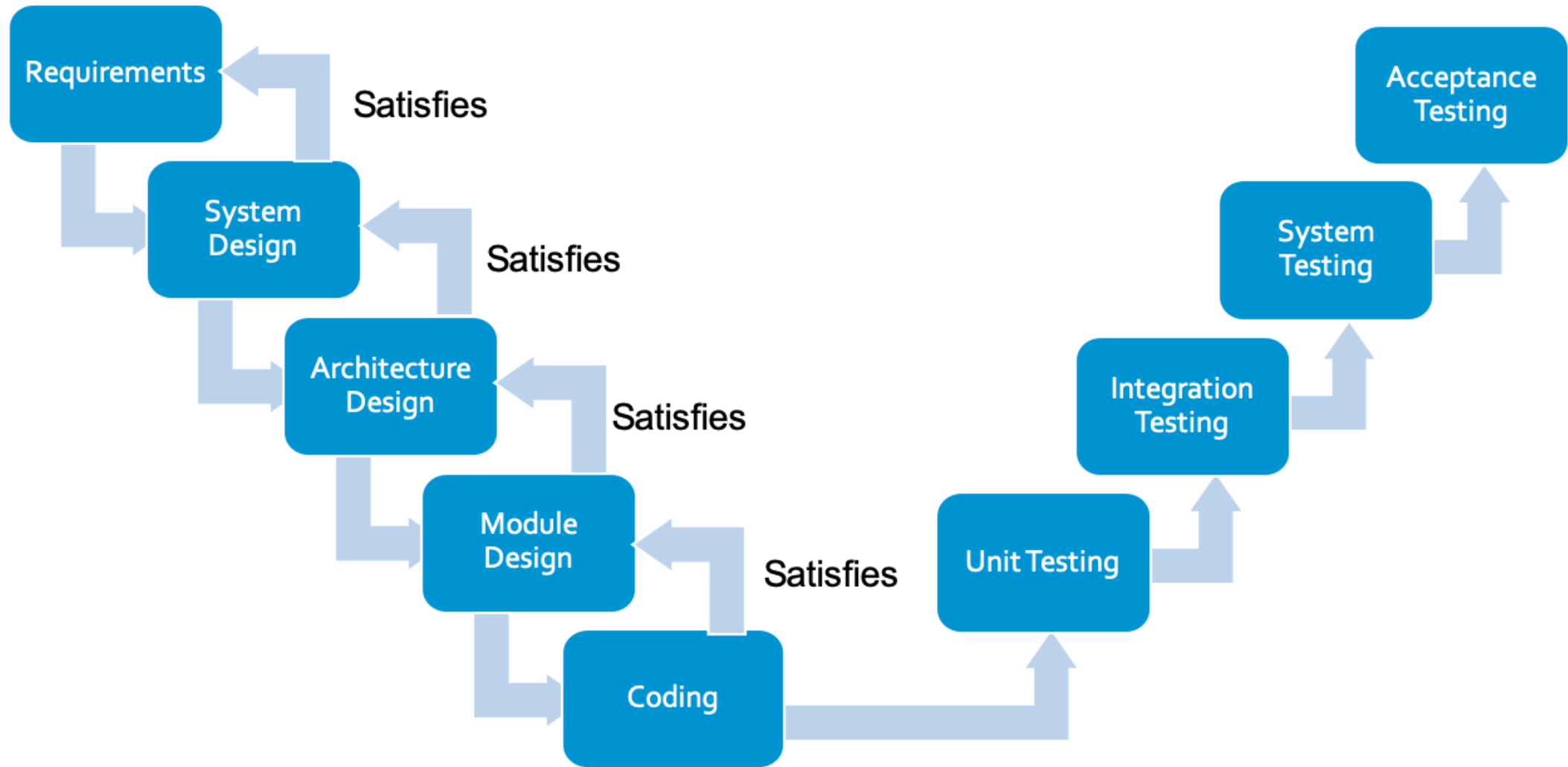
Trazabilidad de Requisitos

La trazabilidad de requisitos es el proceso de **documentar y seguir la relación de los requisitos a lo largo de todo el ciclo de vida del proyecto**. Este seguimiento **asegura que todos los requerimientos se implementen correctamente y se validen de acuerdo con las expectativas y objetivos del proyecto**.

Trazabilidad de Requisitos

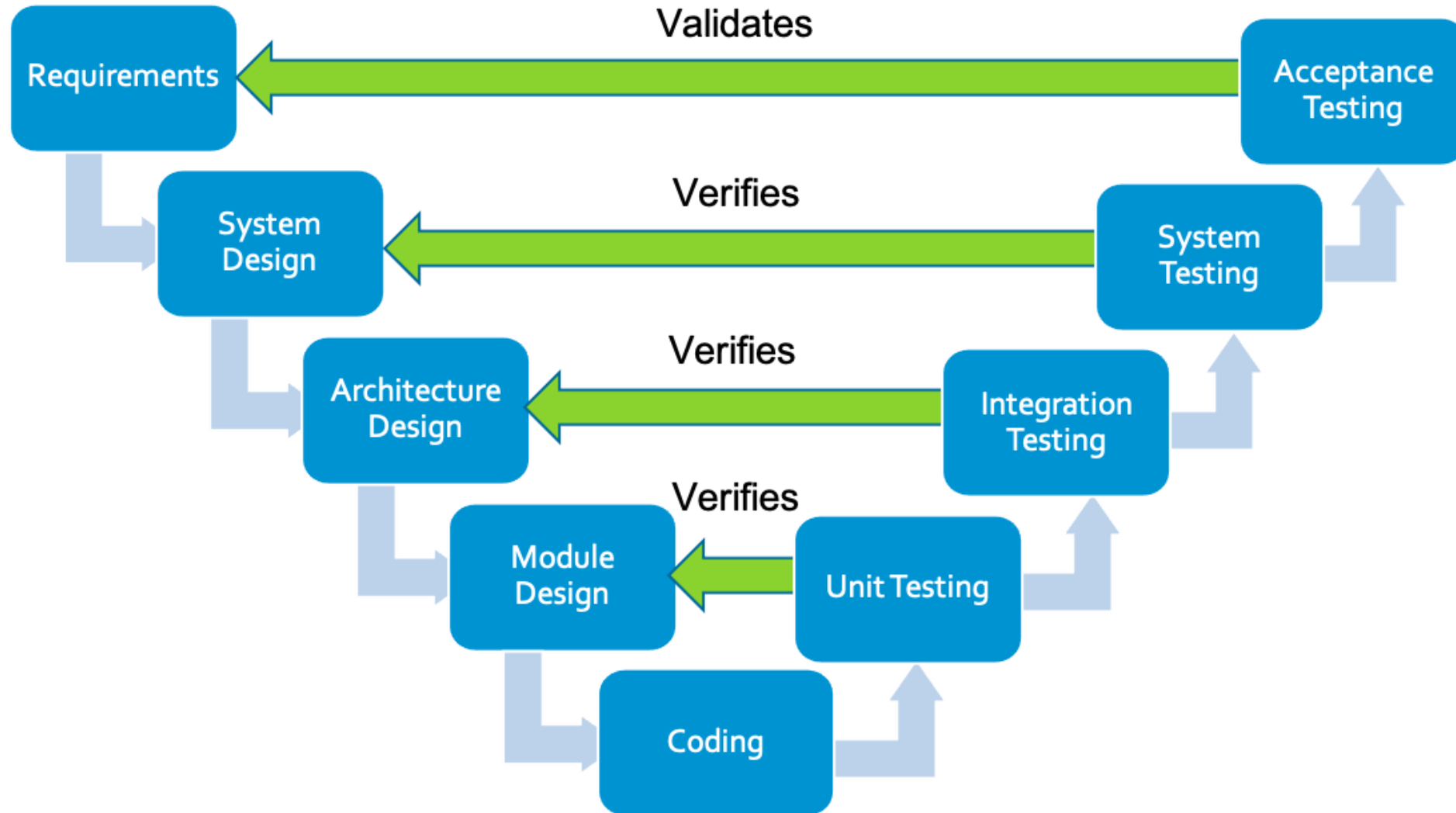


Trazabilidad de Requisitos



El diagrama V clásico muestra cómo avanza y retrocede la trazabilidad en cada fase del desarrollo.

Trazabilidad de Requisitos



La otra parte importante de la trazabilidad de los requisitos son las pruebas de verificación y validación para demostrar la implementación de la especificación desde la fase de diseño correspondiente. La validación generalmente ocurre al final del ciclo de desarrollo durante la prueba de aceptación final con el cliente.

Conceptos y Beneficios de la Trazabilidad de Requisitos

1. Conceptos de Trazabilidad de Requisitos:

Es el seguimiento de la relación entre los requisitos iniciales del proyecto y las diferentes etapas del **desarrollo, prueba, implementación y mantenimiento**. Se utiliza para garantizar que los requisitos se cumplan adecuadamente y cualquier cambio en los mismos se gestione de manera efectiva.

Trazabilidad Directa (Forward Traceability):

- Rastrea la progresión de los requisitos desde su definición hasta su implementación en el sistema final.
- **Ejemplo:** Un requisito de performance requiere no más de 2 segundos de tiempo de respuesta. Trazabilidad directa permitirá verificar que el requisito se ha mantenido consistentemente durante el diseño, implementación y pruebas.

Trazabilidad Inversa (Backward Traceability):

- Hace el seguimiento desde los productos desarrollados y la implementación final hacia los requisitos iniciales para asegurar que cada componente esté alineado con los requerimientos.
- **Ejemplo:** Revisar un caso de prueba que verifica si cumple con el requisito original especificado.

Trazabilidad de Paso Completo (End-to-End Traceability):

- Combina ambos tipos de trazabilidad, asegurando un ciclo de retroalimentación completo desde los requisitos iniciales, pasando por implementaciones y pruebas, hasta el producto final y su validación, y viceversa.

Conceptos y Beneficios de la Trazabilidad de Requisitos

2. Beneficios de la Trazabilidad de Requisitos:

- **Gestión de Cambios:**
 - Facilita la evaluación del impacto de los cambios en los requisitos sobre el proyecto completo, permitiendo decisiones informadas y estratégicas.
- **Cumplimiento Normativo:**
 - Ayuda a garantizar el cumplimiento con normativas y estándares, especialmente en industrias reguladas como salud y finanzas.
- **Aseguramiento de Calidad:**
 - Asegura que todos los requisitos son abordados y validados, reduciendo el riesgo de errores y omisiones.
- **Mejora de la Comunicación:**
 - Clarifica la relación y responsabilidades entre diferentes equipos y stakeholders, mejorando la comprensión y la colaboración.
- **Documentación Integral:**
 - Proporciona una documentación detallada que puede ser utilizada para futuros proyectos, auditorías, y mejora continua.

Técnicas y Herramientas para Establecer la Trazabilidad de Requisitos

1. Técnicas para Establecer la Trazabilidad de Requisitos:

- **Matrices de Trazabilidad:**

Tablas que muestran las relaciones entre requisitos y otros artefactos del proyecto (diseños, casos de prueba, código, etc.).

Facilitan la visualización y gestión de las relaciones.

Ejemplo: Matrices de trazabilidad en Excel o en herramientas como JIRA.

- **Historias de Usuarios y Casos de Uso:**

Relatan historias de usuario y casos de uso detallados que pueden vincularse claramente a los RF y RNF.

Crear historias y casos claramente vinculados a los requisitos para asegurar cobertura total de aspectos.

Ejemplo: Historias de usuarios en JIRA.

Técnicas y Herramientas para Establecer la Trazabilidad de Requisitos

1. Técnicas para Establecer la Trazabilidad de Requisitos:

- **Diagramas UML:**

Uso de diagramas de actividades, casos de uso, secuencia y clases para mostrar la relación entre requisitos y diseños.

Permite visualización gráfica de cómo los requisitos se integran en el sistema.

Ejemplo: Diagrama de casos de uso en **Lucidchart**.

- **Trazabilidad Bidireccional:**

Implementación de trazabilidad tanto desde los requisitos hacia el desarrollo como desde los desarrollos hacia los requisitos.

Uso simultáneo de trazabilidad directa e inversa para un control completo de los cambios y su impacto.

Ejemplo: Vincular casos de prueba en **HP ALM**.

Mantenimiento y Evolución de la Trazabilidad de Requisitos

1. Mantenimiento de la Trazabilidad:

Actualización Continua:

- Mantener las matrices de trazabilidad y las relaciones entre los requisitos y artefactos del proyecto actualizadas. Crítico cada vez que se introduce un cambio en el sistema.
- **Ejemplo:** Actualizar documentos en **Confluence** y tickets en **JIRA** con cambio aprobado.

Revisiones Periódicas:

- Realizar revisiones regulares de la trazabilidad, por ejemplo, al final de cada sprint en metodologías ágiles, para asegurar que todas las relaciones están al día y correctas.
- **Ejemplo:** Revisión mensual de matrices de trazabilidad por parte del equipo de QA.

Validación Regular:

- Validar que todos los requisitos todavía se reflejan correctamente en los artefactos relacionados, especialmente después de cambios importantes.
- **Ejemplo:** Validar mediante revisiones de código y pruebas de regresión que los requisitos no se han comprometido.

Mantenimiento y Evolución de la Trazabilidad de Requisitos

2. Evolución de la Trazabilidad:

Gestión de la Evolución de Requisitos:

- El reconocimiento de que los requisitos pueden evolucionar requiere ajustar la trazabilidad. Actualizar las relaciones y matrices según los cambios en los requisitos.
- **Ejemplo:** Transformar Req y ajustar la trazabilidad después del feedback del cliente.

Adaptación a Proyectos Ágiles:

- Integrar prácticas ágiles para adaptar la trazabilidad a un entorno de desarrollo iterativo, permitiendo flexibilidad y rápida adaptación.
- **Ejemplo:** Implementar seguimiento diario de la evolución del requisito en historias de usuarios durante el sprint.

Herramientas de Automatización:

- Utilizar herramientas que permitan una actualización automática o semi-automática de la trazabilidad para asegurar continuidad y reducir el esfuerzo manual.
- **Ejemplo:** Uso de scripts para extraer información de JIRA y actualizar automáticamente las matrices de trazabilidad en Confluence.

Herramientas y Automatización de la Trazabilidad de Requisitos

1. Herramientas de Trazabilidad:

JIRA con Confluence:

- Integración para gestionar requisitos, tareas y documentación, permitiendo trazabilidad automática entre tickets de trabajo y artefactos documentados.

IBM Rational DOORS:

- Herramienta robusta para la gestión de requisitos con capacidades avanzadas de trazabilidad y soporte para grandes proyectos complejos.

Helix ALM:

- Solución integrada para la trazabilidad de requisitos y la gestión de pruebas y defectos.

ReqView:

- Permite gestionar la trazabilidad de requisitos de forma eficiente y ligera, particularmente útil para proyectos medianos y pequeños.

Herramientas y Automatización de la Trazabilidad de Requisitos

2. Automatización de Trazabilidad:

Scripts y APIs:

- Utilización de scripts y APIs para automatizar el mantenimiento de matrices de trazabilidad, facilitando la incorporación de cambios.
- **Ejemplo:** Python scripts que recolectan datos de JIRA y actualizan tablas en Confluence automáticamente.

Herramientas de Integración Continua (CI):

- Jenkins, GitLab CI/CD para automatizar la validación de trazabilidad cada vez que se realiza un commit o despliegue.
- **Ejemplo:** Scripts en Jenkins que validan la relación entre códigos, pruebas y requisitos en cada integración.

Plugins y Add-ons:

- Utilización de plugins específicos en herramientas como JIRA para facilitar la trazabilidad automática y mantener actualizados los cambios.
- **Ejemplo:** 'Jira Portfolio' mapea visualmente la relación trabajo, requisitos y entregables

Caso Práctico: Trazabilidad en el Sistema de Diagnóstico Médico Asistido por IA (SDMIA)

Al desarrollar un Sistema de Diagnóstico Médico Asistido por IA (SDMIA), se necesita asegurar que todos los requisitos, desde los iniciales hasta los de cumplimiento normativo, se implementen, validen y mantengan adecuadamente.

- **Ejemplo:** Python scripts que recolectan datos de JIRA y actualizan tablas en Confluence automáticamente.

Herramientas de Integración Continua (CI):

- Jenkins, GitLab CI/CD para automatizar la validación de trazabilidad cada vez que se realiza un commit o despliegue.
- **Ejemplo:** Scripts en Jenkins que validan la relación entre códigos, pruebas y requisitos en cada integración.

Plugins y Add-ons:

- Utilización de plugins específicos en herramientas como JIRA para facilitar la trazabilidad automática y mantener actualizados los cambios.
- **Ejemplo:** 'Jira Portfolio' mapea visualmente la relación trabajo, requisitos y entregables

La trazabilidad de requisitos es crucial para el éxito de proyectos complejos como el Sistema de Diagnóstico Médico Asistido por IA (SDMIA).

Al establecer una estructura clara y utilizar herramientas y técnicas adecuadas, los estudiantes pueden asegurar una excelente gestión y validación de requisitos, garantizando que el sistema cumpla con todas las expectativas y normativas existentes.

Esto permite una adaptación eficiente a los cambios y mejora la calidad y coherencia de cada componente del proyecto.

Ingeniería de Requisitos

Sesión 14

Validación de Requisitos

Prof. Ciro Rodriguez
crodriguezro@unmsm.edu.pe

Tema: Trazabilidad de Requisitos

- Conceptos y Técnicas de Validación de Requisitos
- Planificación y Ejecución de Actividades de Validación de Requisitos
- Análisis e Interpretación de Resultados de Validación de Requisitos
- Seguimiento y Seguimiento de Acciones de Validación de Requisitos

Validación de Requisitos

La validación de requisitos es un proceso esencial en el desarrollo de cualquier sistema o proyecto y tiene como objetivo **asegurar que los requisitos definidos cumplen con las expectativas y necesidades de los usuarios y stakeholders.**

Conceptos Técnicas de Validación de Requisitos

1. Conceptos y Validacion de Requisitos:

La validación de requisitos es el proceso de confirmar que los requisitos especificados son correctos y completos, y que el sistema que se desarrollará satisfará dichos requisitos.

Diferencia con Verificación: La verificación asegura que el producto se está construyendo correctamente (**conforme a las especificaciones**), mientras que la validación **confirma** que se está construyendo el producto correcto (**satisface las necesidades del usuario**).

2. Técnicas de Validación de Requisitos:

Revisiones de Requisitos:

- Descripción: Reuniones formales e informales en las que los requisitos son revisados y discutidos para identificar errores o incompletitudes.
- Métodos: Inspecciones, walkthroughs, y revisiones por pares.

Prototipos:

- Descripción: Creación de versiones tempranas del sistema o sus componentes que permiten a los stakeholders interactuar con el sistema y validar requisitos.
- Métodos: Prototipos de alta fidelidad, wireframes y maquetas interactivas.

Conceptos Técnicas de Validación de Requisitos

2. Técnicas de Validación de Requisitos:

Pruebas de Aceptación del Usuario (UAT):

- Descripción: Pruebas realizadas por los usuarios finales para validar que el sistema cumple con los requisitos y es adecuado para su uso.

Modelos y Simulaciones:

- Descripción: Uso de modelos y simulaciones para validar el comportamiento del sistema conforme a los requisitos.
- Métodos: Diagramas UML, simulaciones de casos de uso.

Estudios de Caso y Entrevistas:

- Descripción: Recopilación de experiencias y feedback directo de usuarios y stakeholders para validar requisitos.

Análisis de Escenarios:

- Descripción: Evaluación de cómo el sistema satisface los requisitos bajo diferentes condiciones y escenarios de uso.

Planificación y Ejecución de Actividades de Validación de Requisitos

1. Planificación de la Validación:

Identificación de Requisitos Críticos:

- Priorizar los requisitos en función de su importancia y riesgos asociados.

Definición de Metodología:

- Elegir las técnicas de validación adecuadas (revisiones, prototipos, UAT) para los diferentes tipos de requisitos.

Asignación de Recursos:

- Determinar los recursos necesarios (personas, herramientas, tiempo) para llevar a cabo las actividades de validación.

Creación de un Plan de Validación:

- Documento que incluya un calendario de actividades, responsabilidades y métodos de validación.

Planificación y Ejecución de Actividades de Validación de Requisitos

2. Ejecución de la Validación:

Revisiones de Requisitos:

- Realizar revisiones de documentos de requisitos con la participación de stakeholders clave.

Desarrollo de Prototipos:

- Crear prototipos funcionales y/o no funcionales para evaluar y validar los requisitos con los usuarios.

Pruebas de Aceptación del Usuario:

- Conducir UAT con usuarios finales en escenarios reales de uso.

Simulaciones y Modelos:

- Utilizar modelos y simulaciones para probar que el comportamiento del sistema cumple con los requisitos especificados.

Planificación y Ejecución de Actividades de Validación de Requisitos

3. Documentación de Resultados:

Registro de Observaciones:

- Documentar todas las observaciones y hallazgos realizados durante las actividades de validación.

Reporte de Resultados:

- Crear informes detallados que describan los resultados de las validaciones, incluyendo cualquier requisito que no se haya cumplido.

Análisis e Interpretación de Resultados de Validación de Requisitos

1. Análisis de Resultados de Validación:

Revisión de Resultados:

- Analizar los resultados obtenidos de las revisiones de requisitos, prototipos, UAT y modelos.

Identificación de Problemas:

- Detectar cualquier requisito que no se haya validado correctamente o que muestre inconsistencias.

Clasificación de Problemas:

- Clasificar los problemas en categorías tales como errores, omisiones, ambigüedades y necesidades adicionales.

Análisis e Interpretación de Resultados de Validación de Requisitos

2. Interpretación de Resultados:

Evaluación del Impacto:

- Evaluar el impacto de los problemas identificados en el proyecto general, tiempo y costo.

Priorización de Acciones:

- Ordenar los problemas identificados en términos de urgencia y criticidad para la resolución.

Decisión sobre Correcciones:

- Decidir qué acciones tomar para corregir los problemas identificados y asegurar el cumplimiento de los requisitos.

3. Informe de Análisis:

Contenido del Informe:

- Incluir un resumen de los resultados de la validación, análisis detallado de problemas y recomendaciones para corregirlos.

Ejemplo:

Seguimiento y Monitoreo de Acciones de Validación de Requisitos

1. Seguimiento de Acciones de Validación:

Plan de Acción:

- Desarrollar un plan de acción para abordar los problemas identificados en la fase de análisis.

Asignación de Responsabilidades:

- Asignar responsabilidades claras para la resolución de cada problema identificado.

Calendario de Seguimiento:

- Establecer un calendario para realizar seguimientos periódicos de las acciones correctivas.

2. Monitoreo del Progreso:

Reuniones de Seguimiento:

- Organizar reuniones regulares para revisar el progreso de las acciones correctivas.

Actualización del Estado:

- Mantener actualizado el estado de las acciones pendientes y completadas.

Seguimiento y Monitoreo de Acciones de Validación de Requisitos

Revisiones de Calidad:

- Realizar revisiones de calidad para asegurar que las acciones correctivas implementadas cumplen con los requisitos de validación.

3. Documentación del Seguimiento:

Registro de Cambios:

- Documentar cualquier cambio realizado como resultado de las actividades de seguimiento y monitoreo.

Informe de Seguimiento:

- Crear informes de seguimiento detallados que describan el progreso, los problemas resueltos y cualquier desviación del plan original.

iiiA trabajar!!!