**TRABAJO PRÁCTICO:**

**Proceso Unificado**

1) Las tres características principales del Proceso Unificado son:

* Está dirigido por los casos de uso:
* Desde la especificación hasta el mantenimiento.
* Se centra en la arquitectura:
* La arquitectura es prioritaria desde el principio hasta el final.
* Se facilita el refinamiento progresivo de la arquitectura.
* Iterativo e incremental
* El trabajo se divide en iteraciones pequeñas en función de la importancia de los casos de uso y el análisis de riesgos.

2) Un “Caso de Desarrollo” es un componente fundamental que define las actividades específicas necesarias para llevar a cabo el desarrollo de un software. Estos casos detallan cada paso a seguir para implementar, verificar y gestionar los distintos elementos de software que permitirán cumplir con los requisitos identificados en los Casos de Uso. En este sentido, los Casos de Desarrollo guían a los desarrolladores en la ejecución de cada funcionalidad, proporcionando instrucciones específicas y criterios de éxito que aseguran que el producto final cumpla con las expectativas. A su vez, cada Caso de Desarrollo cubre aspectos como la especificación de tareas, la asignación de recursos y la definición de pruebas y métricas para evaluar el avance.

Por otro lado, definir estos casos al principio de un proyecto ayuda a identificar posibles riesgos y problemas antes de que se conviertan en obstáculos. Esto permite que el equipo tome medidas preventivas, reduciendo la probabilidad de errores o retrabajos costosos en fases posteriores del proyecto.

3) La fase de Inicio (Inception)

Al comenzar un proyecto hay que contestar algunas preguntas:

* ¿Cuál es la visión del sistema?
* ¿Es viable?
* ¿Se puede comprar o hay que fabricar el sistema?
* ¿Cuánto va a costar?
* Y finalmente, ¿Seguimos adelante o paramos?

El objetivo es desarrollar el análisis de negocio hasta el punto necesario para la puesta en marcha del proyecto. Para ello, es necesario:

* Delimitar el alcance y objetivos del proyecto
* Definir la funcionalidad y capacidades del producto
* Tener una idea de la arquitectura (arquitectura candidata)
* Reducir los riesgos cuanto antes
* Hacer estimaciones iniciales de costos.

4) Proceso Iterativo e Incremental

La característica fundamental de UP es el Proceso Iterativo e Incremental:

* Se basa en la ampliación y el refinamiento del sistema.
* Es una serie de desarrollos cortos (mini proyectos de 2 a 6 semanas, cada iteración reproduce el ciclo de vida a menor escala).
* No solo se mejora sino que el sistema también crece.
* El resultado de cada iteración es un sistema ejecutable (aunque sea incompleto y no esté listo para su ejecutable).
* Un sistema instalable requiere varias iteraciones:
* Evolución de prototipos ejecutables
* Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes
* Concepto de “time-boxing”: cada iteración debe tener una duración fija (el máximo son 6 meses)
* El progreso es visible

5) En el Proceso Unificado (UP), las etapas de Ingeniería y Producción representan fases avanzadas del ciclo de vida del proyecto de software, cada una con objetivos, enfoques y dinámicas de equipo diferentes.

La etapa de **Ingeniería** se centra en el desarrollo y estabilización del sistema, incluyendo la validación de que el software cumple con los requisitos funcionales y no funcionales especificados. Las actividades en esta etapa incluyen la integración de componentes, la validación del sistema mediante pruebas exhaustivas y la revisión de la arquitectura para asegurar su solidez y capacidad de soportar el despliegue. Dentro de esta etapa, se pueden identificar principalmente dos fases: la de Construcción, donde el sistema toma forma con cada iteración de desarrollo, y la de Prueba o Validación, que asegura la calidad final del producto antes de pasar a Producción.

La etapa de **Producción**, en cambio, se enfoca en la entrega, despliegue y mantenimiento del sistema. Su objetivo es garantizar que el sistema funcione adecuadamente en un entorno real y que esté listo para el uso final por parte de los usuarios. Aquí se incluyen las fases de Despliegue y Mantenimiento, que implican el soporte técnico, la capacitación de usuarios y la supervisión de la estabilidad del sistema en su entorno operativo. Esta etapa también incluye ajustes y correcciones menores en respuesta a problemas reportados o necesidades de optimización.

En términos de **predictibilidad**, la etapa de Ingeniería tiende a ser menos predecible debido a la naturaleza del desarrollo y las pruebas. Los equipos pueden encontrarse con problemas inesperados en la integración de componentes, conflictos de diseño o dificultades para cumplir con los requisitos de calidad. Esto hace que la etapa de Ingeniería requiera flexibilidad en la gestión del tiempo y los recursos, ya que la resolución de problemas puede alargar las iteraciones.

Por otro lado, la etapa de Producción suele ser más predecible. Dado que el sistema ya está desarrollado y probado, las actividades principales son el despliegue y el soporte, tareas que generalmente se planifican con mayor precisión. Aunque pueden surgir problemas, estos tienden a ser menores y están relacionados con ajustes específicos, sin necesidad de hacer cambios profundos en el software.

En cuanto al **tamaño del equipo**, la etapa de Ingeniería generalmente requiere un equipo más grande y especializado, ya que incluye desarrolladores, arquitectos, ingenieros de pruebas y posiblemente expertos en integración para manejar la complejidad de la construcción del sistema. En la etapa de Producción, el equipo suele ser más reducido y especializado en soporte técnico y mantenimiento, aunque puede ampliarse temporalmente en caso de que surjan problemas importantes.

6) En el Proceso Unificado (UP), las "cuatro P" fundamentales son:

* **Personas:** Son los miembros del equipo, cada uno con roles específicos y esenciales para el éxito del proyecto. La comunicación y colaboración son claves para un buen resultado.
* **Proyecto:** Implica la planificación y gestión de recursos, tiempo y presupuesto para cumplir los objetivos, adaptándose a cambios e imprevistos cuando sea necesario.
* **Producto:** Es el software final y todos los artefactos creados durante el desarrollo. El producto mejora en cada iteración hasta cumplir con los requisitos del cliente.
* **Proceso:** Estructura el desarrollo de software mediante actividades y roles en etapas y disciplinas iterativas, garantizando un avance controlado y adaptable.

Estas "P" funcionan de manera interdependiente para asegurar la calidad y el éxito del desarrollo de software.

7) El modelo de arquitectura 4+1 es una metodología para representar la arquitectura de sistemas complejos desde diferentes perspectivas. Utiliza cinco vistas (de ahí el "+1") para asegurar que cada aspecto relevante del sistema sea analizado adecuadamente, abordando tanto los requisitos funcionales como los no funcionales. Las cinco vistas son:

* **Vista Lógica:** Describe la funcionalidad del sistema desde la perspectiva del usuario. Representa cómo se organizan los elementos principales para satisfacer los requisitos funcionales. Se enfoca en los casos de uso y el flujo de información.
* **Vista de Realización** (también llamada vista de implementación): Se centra en la organización estática del software en su desarrollo, como módulos, bibliotecas y componentes. Ayuda a los desarrolladores a entender cómo el código se organizará en paquetes o módulos.
* **Vista de Procesos:** Muestra el comportamiento dinámico del sistema, principalmente cómo los diferentes procesos o tareas se comunican y sincronizan en tiempo de ejecución. Es fundamental para garantizar que se cumplan requisitos no funcionales como la concurrencia y la escalabilidad.
* **Vista de Despliegue:** Representa la disposición física de los componentes en hardware. Esta vista es importante para considerar los aspectos de distribución, rendimiento y disponibilidad del sistema en el entorno real.
* **Vista de Casos de Uso:** Integra las cuatro vistas anteriores mediante escenarios específicos o casos de uso que ilustran cómo se comporta el sistema en diferentes situaciones. Sirve para verificar y validar que el diseño cumple con los requisitos funcionales y no funcionales.

8) Criterios de evaluación de la fase de Elaboración

Al comienzo de la fase de elaboración:

* Se planifica la fase y se forma el equipo
* Se establecen criterios de evaluación que habrá que cumplir al final:
* Respecto a los requisitos:
* ¿Se han identificado? ¿Se han detallado lo suficiente?
* En cuanto a la arquitectura:
* ¿Satisface los requisitos? ¿Es robusta?
* Los riesgos:
* ¿Se han eliminado los críticos? ¿Se ha completado la lista?
* Evaluar el proyecto:
* ¿Se puede fijar un precio y una fecha de entrega?

9) En el Proceso Unificado, un **artefacto** es cualquier documento, modelo o elemento de software que se crea, modifica y utiliza durante el ciclo de vida del proyecto para definir, visualizar, construir y probar el sistema. Los artefactos ayudan a capturar y comunicar distintos aspectos del sistema, sirviendo de referencia y guía en cada fase del desarrollo (inicio, elaboración, construcción y transición). Los cinco tipos diferentes de artefactos son:

* **Modelo de Casos de Uso**: Define los requisitos funcionales del sistema desde la perspectiva de los usuarios. Contiene los casos de uso y los actores que interactúan con el sistema, ayudando a capturar los escenarios que el sistema debe soportar.
* **Modelo de Diseño:** Describe cómo el sistema se estructurará internamente para satisfacer los requisitos funcionales. Este modelo incluye diagramas de clases, componentes y otros elementos de diseño que guían la implementación del sistema.
* **Modelo de Implementación:** Es un conjunto de componentes de software organizados en paquetes que representan el código fuente del sistema. Este modelo estructura la lógica de negocio en módulos, facilitando el desarrollo y el mantenimiento.
* **Plan de Pruebas:** Documenta la estrategia y los casos de prueba para asegurar que el sistema cumpla con los requisitos y funciones correctamente. Incluye los casos de prueba específicos, el enfoque de pruebas y los criterios de aceptación para verificar la calidad del sistema.
* **Modelo de Despliegue:** Describe cómo se distribuirán y desplegarán los componentes del sistema en el hardware real, incluyendo servidores y redes. Este modelo garantiza que el sistema esté preparado para operar en el entorno final y sea escalable, seguro y eficiente.

10) En el Proceso Unificado, la gestión del riesgo es una actividad fundamental y continua que se realiza a lo largo de todo el ciclo de vida del proyecto. Su propósito es identificar, analizar, y mitigar los riesgos que puedan afectar el éxito del proyecto, desde la planificación inicial hasta la entrega final. La gestión del riesgo en el Proceso Unificado se lleva a cabo en todas sus fases (inicio, elaboración, construcción y transición), con un enfoque más profundo en las primeras etapas, cuando aún es posible hacer ajustes importantes con menor impacto en costos y cronograma.

Algunos tipos de Riesgos en el Proceso Unificado son:

* **Riesgos Técnicos:** Relacionados con la tecnología o la arquitectura del sistema, como problemas con la elección de tecnologías, dificultad para integrar sistemas o el uso de tecnologías nuevas o inestables.
* **Riesgos de Requerimientos:** Asociados a la posibilidad de cambios en los requisitos o a la falta de claridad en estos. Incluye la inestabilidad de los requisitos o su falta de alineación con las expectativas del cliente.
* **Riesgos de Cronograma:** Peligros relacionados con la gestión del tiempo, como plazos ajustados, estimaciones de tiempo inexactas o dependencias que retrasan el desarrollo.
* **Riesgos de Personal:** Implican problemas de disponibilidad o habilidades del equipo, como carencias en competencias necesarias o la rotación de personal clave.
* **Riesgos de Calidad:** Incluyen riesgos asociados a la falta de control de calidad, que pueden derivar en problemas de rendimiento, seguridad o estabilidad.

Abordaje de los Riesgos en Cada Fase del Proceso Unificado:

1. **Fase de Inicio:** En esta fase, el equipo identifica y documenta los riesgos iniciales, centrándose en aquellos que podrían afectar la viabilidad del proyecto. Se evalúan los riesgos relacionados con los requisitos, la tecnología y los recursos necesarios. Aquí, se define una primera estrategia de mitigación para los riesgos críticos y se establece un plan de contingencia.
2. **Fase de Elaboración:** Esta fase tiene un enfoque fuerte en la mitigación de riesgos técnicos y de requisitos. Se realizan prototipos y pruebas de concepto para validar decisiones arquitectónicas y tecnológicas. Los requisitos también se detallan y priorizan para reducir la ambigüedad y los posibles cambios futuros. Se ajustan los planes de mitigación y se revisan los riesgos según el aprendizaje obtenido.
3. **Fase de Construcción:** Durante la construcción, la atención se centra en los riesgos de cronograma y calidad. La gestión del riesgo aquí busca asegurar que el desarrollo avance de acuerdo con los tiempos previstos y que se mantenga la calidad del producto. Se implementan estrategias de monitoreo y control para evitar desviaciones de cronograma, y se aplican pruebas continuas para reducir los riesgos de calidad.
4. **Fase de Transición:** En la fase de transición, los riesgos asociados a la implementación y la aceptación del sistema son críticos. Se gestionan los riesgos relacionados con el despliegue, la capacitación de usuarios y el soporte. Aquí, los planes de contingencia están listos para cualquier problema que surja durante la implementación del sistema en su entorno real.