

**PROPUESTA METODOLOGICA PARA LA GENERACION DE
OBJETOS DE APRENDIZAJE USABLES**

**CASO DE ESTUDIO: PROCESO DE ENSEÑANZA-
APRENDIZAJE DEL IDIOMA INGLES PARA ESTUDIANTES
UNIVERSITARIOS**

ANEXOS

UNIVERSIDAD DEL CAUCA

Ricardo Chicangana Solano

Sandro Javier Jaramillo Portillo

UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES
PROGRAMA INGENIERÍA DE SISTEMAS
Grupo de Investigación IDIS
Investigación y Desarrollo en Ingeniería de Software
Popayán
2010

**PROPUESTA METODOLOGICA PARA LA GENERACION DE
OBJETOS DE APRENDIZAJE USABLES**

**CASO DE ESTUDIO: PROCESO DE ENSEÑANZA-
APRENDIZAJE DEL IDIOMA INGLES PARA ESTUDIANTES
UNIVERSITARIOS**



UNIVERSIDAD DEL CAUCA

Ricardo Chicangana Solano

Sandro Javier Jaramillo Portillo

Trabajo de investigación para optar al título de Ingenieros de Sistemas

Director:

Ing. Wilson Libardo Pantoja Yepéz

**UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES
PROGRAMA INGENIERÍA DE SISTEMAS
Grupo de Investigación IDIS
Investigación y Desarrollo en Ingeniería de Software
Popayán
2010**

TABLA DE CONTENIDO

A.	ANEXO TABLAS DE ADAPTACION Y CONFIGURACION DE RUP	1
B.	ANEXO RESEÑA PARA RUP	6
C.	ANEXO MODELOS Y METODOLOGÍAS PARA EL DESARROLLO DE OBJETOS DE APRENDIZAJE	30
D.	ANEXO TÉCNICAS DE USABILIDAD SELECCIONADAS	36
E.	ANEXO ESTÁNDARES REFERENTES LA DISCIPLINA IPO, INGENIERÍA DE SOFTWARE Y DISEÑO CENTRADO EN EL USUARIO	44
F.	ANEXO CASO DE ESTUDIO	51

A. ANEXO TABLAS DE ADAPTACION Y CONFIGURACION DE RUP

A continuación se presentan las tablas detalladas de la evaluación de cada uno de los criterios a cada uno de los elementos del RUP, considerados según el apartado de tailoring de este mismo, como convenientes para su inclusión en proyectos de este tipo.

Tabla 1. Evaluación de elementos correspondientes a la disciplina de modelado de negocios

Actividades	Tareas	Roles	Artefactos relacionados	Roles Asesores	Criterios para Tareas				Criterios para artefactos		Criterios para roles			Ponderado
					AOAS	AUS	COM	COS	NEC	VAL	NECR	RES	ART	
Evaluar el estado del negocio	Captura de vocabulario común del negocio	Analista de procesos de negocio	Glosario del Negocio	Usuario final, cliente										
	Mantener reglas del negocio	Analista de procesos de negocio	Reglas del negocio	Usuario final, cliente										
	Definir la arquitectura del negocio	Analista de procesos de negocio	Documento de la arquitectura del negocio	Usuario final, cliente										
	Evaluar la meta de la organización	Analista de procesos de negocio	Evaluación de la meta de la organización	Usuario final, cliente										
	Configurar y ajustar objetivos	Analista de procesos de negocio	Visión del Negocio	Usuario final, cliente										
	Identificar los objetivos de la organización	Analista de procesos de negocio	Objetivos del Negocio	Usuario final, cliente										
Descripcion de negocio actual	Mantener reglas del negocio	Analista de procesos de negocio	Reglas del negocio	Usuario final, cliente										
	Definir la arquitectura del negocio	Analista de procesos de negocio	Documento de la arquitectura del negocio	Usuario final, cliente										

	Evaluar la meta de la organización	Analista de procesos de negocio	Evaluación de la meta de la organización	Usuario final, cliente										
	Configurar y ajustar objetivos	Analista de procesos de negocio	Visión del Negocio	Usuario final, cliente										
	Identificar los objetivos de la organización	Analista de procesos de negocio	Objetivos del Negocio	Usuario final, cliente										
	Definir actores y casos de uso del negocio	Analista de procesos de negocio	Modelo de casos de uso de negocio	Usuario final, cliente										
	Definir trabajadores y entidades en el negocio	Diseñador de negocio	Modelo de casos de uso de negocio refinado, Modelo de análisis de negocio	Ninguno										
Desarrollar un Modelo de dominio	Captura de vocabulario común del negocio	Analista de procesos de negocio	Glosario del Negocio	Ninguno										
	Mantener reglas del negocio	Analista de procesos de negocio	Reglas del negocio	Ninguno										
	Revisar modelo de analizas de negocios	Revisor Técnico	Registro de revisión	Usuario final, cliente										
	Definir trabajadores y entidades en el negocio	Diseñador de negocio	Modelo de análisis de negocio	Ninguno										

Tabla 2. Evaluación de elementos correspondientes a la disciplina de requerimientos

Actividades	Taras	Roles	Artefactos relacionados	Roles Asesores	Criterios para tareas				Criterios para artefactos		Criterios para roles			Ponderado
					AOAS	AUS	COM	COS	NEC	VAL	NECR	RES	ART	
Análisis del Problema	Captura de vocabulario común del negocio	Analista del Sistema	Glosario	Usuario final, cliente, otro stakeholder										
	Definir actores y casos de uso	Analista del Sistema	Modelo de Casos de uso	Usuario final, cliente, otro stakeholder										
	Desarrollar la visión	Analista del Sistema	Visión	Usuario final, cliente, otro stakeholder										

Comprender Necesidades de los Stakeholders	Captura de vocabulario común del negocio	Analista del Sistema	Glosario	Usuario final, cliente, otro stakeholder																
	Desarrollar la visión	Analista del Sistema	Vision	Usuario final, cliente, otro stakeholder																
	Definir actores y casos de uso	Analista del Sistema	Modelo de Casos de uso	Usuario final, cliente, otro stakeholder																
	Especificar requerimientos de los stakeholders	Analista del Sistema	Requerimientos de los stakeholders, Storyboard	Usuario final, cliente, otro stakeholder																
Definir el Sistema	Desarrollar la visión	Analista del Sistema	Especificaciones suplementarias																	
	Captura de vocabulario común del negocio	Analista del Sistema	Glosario(refinado)																	
	Definir actores y casos de uso	Analista del Sistema	Modelo de casos de uso (refinado)																	
Gestión de Cambios en Requerimientos	Estructurar el modelo de casos de uso	Analista del Sistema	Modelo de casos de uso (reestructurado)	Usuario final, cliente, otro stakeholder																
	Revisión de requerimientos	Revisor Técnico	Resultados de revisión	Usuario final, cliente, otro stakeholder																

Tabla 3. Evaluación de elementos correspondientes a la disciplina de análisis y diseño

Actividades	Tareas	Roles	Artefactos relacionados	Criterios para tareas				Criterios para artefactos		Criterios para roles			Ponderado	
				AOAS	AUS	COM	COS	NEC	VAL	NECR	RES	ART		
Definir una arquitectura candidata	Análisis Arquitectural	Arquitecto	Modelo de diseño, modelo de despliegue, documento de la arquitectura											
	Análisis de Casos de uso	Diseñador	Análisis de realización de casos de uso											
Análisis de comportamiento	Análisis de Casos de uso	Diseñador	Análisis de realización de casos de uso											
	Identificar y diseñar elementos	Arquitecto	Modelo diseño											
	Revisión del diseño	Revisor Técnico	Resultados de revisión											

	Diseño de la interfaz de usuario	Diseñador de la interfaz de usuario	Diseño interfaz de usuario											
Diseño de componentes	Diseño de caso de uso	Diseñador	Realización de caso de uso											
	Diseño de componente	Diseñador	Diseño de Componente											
	Revisión del diseño	Revisor Técnico	Resultados de revisión											

Tabla 4. Evaluación de elementos correspondientes a la disciplina de implementación

Actividades	Tareas	Roles	Artefactos relacionados	Criterios para tareas				Criterios para artefactos		Criterios para roles			Ponderado	
				AOAS	AUS	COM	COS	NEC	VAL	NECR	RES	ART		
Planear la Integración	Planear la integración del sistema	Integrador	Plan de integración											
Implementar Componentes	Implementar componentes	Implementador	Componente implementado											
	Implementar y Ejecutar pruebas de desarrollador	Implementador	Pruebas de desarrollador											
	Revisión de Componente	Revisor Técnico	Resultados de revisión											
Integrar cada subsistema	Implementar y Ejecutar pruebas de desarrollador	Implementador	Resultados de pruebas											
	Integrar subsistema	Integrador	Subsistema											
Integrar el sistema	Integrar sistema	Integrador	Sistema											

Tabla 5. Evaluación de elementos correspondientes a la disciplina de prueba

Actividades	Tareas	Roles	Artefactos relacionados	Criterios para tareas				Criterios para artefactos		Criterios para roles			Ponderado	
				AOAS	AUS	COM	COS	NEC	VAL	NECR	RES	ART		
Definir Misión de la Evaluación	Identificar motivaciones de la prueba	Gestor de Pruebas	Plan de Pruebas											
	Identificar Objetivos de la prueba	Analista de Pruebas	Plan de Pruebas											
Probar y Evaluar	Implementar prueba	Probador	Resultados de la prueba											

	Análisis de fallas encontradas	Probador	Solicitud de cambios										
--	--------------------------------	----------	----------------------	--	--	--	--	--	--	--	--	--	--

Tabla 6. Evaluación de elementos correspondientes a la disciplina de despliegue

Actividades	Tareas	Roles	Artefactos relacionados	Criterios para tareas				Criterios para artefactos		Criterios para roles			Ponderado
				AOAS	AUS	COM	COS	NEC	VAL	NECR	RES	ART	
Plan de Despliegue	Desarrollar plan de despliegue	Gestor de despliegue	Plan de despliegue										
Desarrollo de material de soporte	Desarrollar materiales de soporte	Redactor técnico	Materiales de soporte para usuarios finales										
Gestionar pruebas de aceptación	Ejecución de pruebas en sitio	Probador	Resultados de la prueba										
	Analizar resultados de las pruebas	Analista de Pruebas	Sumario de evaluación de las pruebas										

Tabla 7. Evaluación de elementos correspondientes a la disciplina de gestión de proyecto

Actividades	Tareas	Roles	Artefactos relacionados	Criterios para tareas				Criterios para artefactos		Criterios para roles			Ponderado
				AOAS	AUS	COM	COS	NEC	VAL	NECR	RES	ART	
Evaluar riesgos y alcance del proyecto	Identificar riesgos	Gestor de Proyecto	Lista de riesgos										
Planear iteración	Desarrollar plan de iteración	Gestor de Proyecto	Plan de iteración										
Planear proyecto	Desarrollar plan de control de riesgos	Gestor de Proyecto	Plan de control de riesgos										
	Planear fases e iteraciones	Gestor de Proyecto	Plan de desarrollo de solución										
Gestión de iteración	Gestionar y controlar iteración	Gestor de Proyecto	Aceptación de iteración										
Monitorear y controlar proyecto	Monitorear y controlar estado del proyecto	Gestor de Proyecto	Indicadores del proyecto										

B. ANEXO RESEÑA PARA RUP

Historia

La Figura 1 ilustra la historia de RUP. El antecedente más importante se ubica en 1967 con la Metodología Ericsson (*Ericsson Approach*) elaborada por Ivar Jacobson, una aproximación de desarrollo basada en componentes, que introdujo el concepto de Caso de Uso. Entre los años de 1987 a 1995 Jacobson fundó la compañía *Objectory AB* y lanza el proceso de desarrollo *Objectory* (abreviación de *Object Factory*).

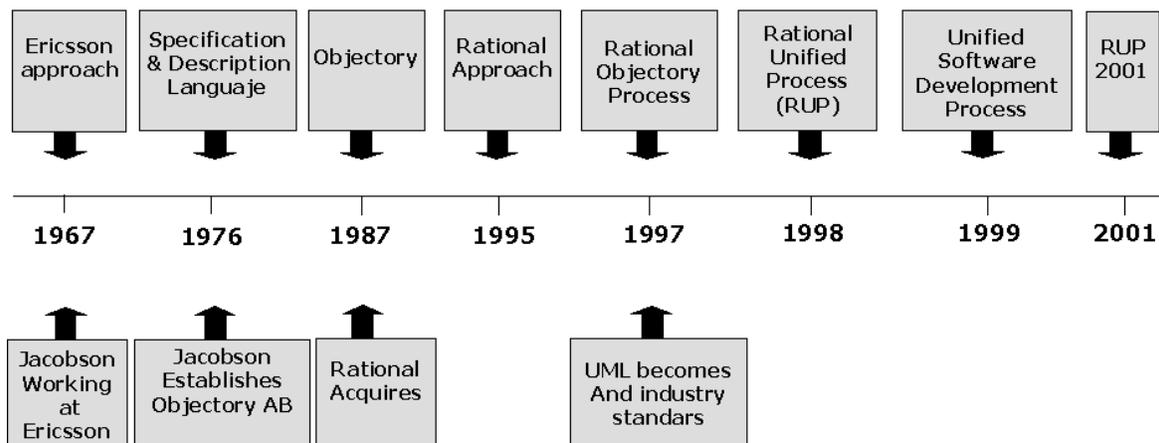


Figura 1. Historia de RUP

Posteriormente en 1995 *Rational Software Corporation* adquiere *Objectory AB* y entre 1995 y 1997 se desarrolla *Rational Objectory Process* (ROP) a partir de *Objectory 3.8* y del Enfoque Rational (*Rational Approach*) adoptando UML como lenguaje de modelado.

Desde ese entonces y a la cabeza de Grady Booch, Ivar Jacobson y James Rumbaugh, Rational Software desarrolló e incorporó diversos elementos para expandir ROP, destacándose especialmente el flujo de trabajo conocido como modelado del negocio. En junio del 1998 se lanza *Rational Unified Process*.

Características esenciales

Los autores de RUP destacan que el proceso de software propuesto por RUP tiene tres características esenciales: está dirigido por los Casos de Uso, está centrado en la arquitectura, y es iterativo e incremental.

Proceso dirigido por Casos de Uso

Según ¹los Casos de Uso son una técnica de captura de requisitos que fuerza a pensar en términos de importancia para el usuario y no sólo en términos de funciones que sería bueno contemplar. Se define un Caso de Uso como un fragmento de funcionalidad del sistema que proporciona al usuario un valor añadido. Los Casos de Uso representan los requisitos funcionales del sistema.

En RUP los Casos de Uso no son sólo una herramienta para especificar los requisitos del sistema. También guían su diseño, implementación y prueba. Los Casos de Uso constituyen un elemento integrador y una guía del trabajo como se muestra en la Figura 2.

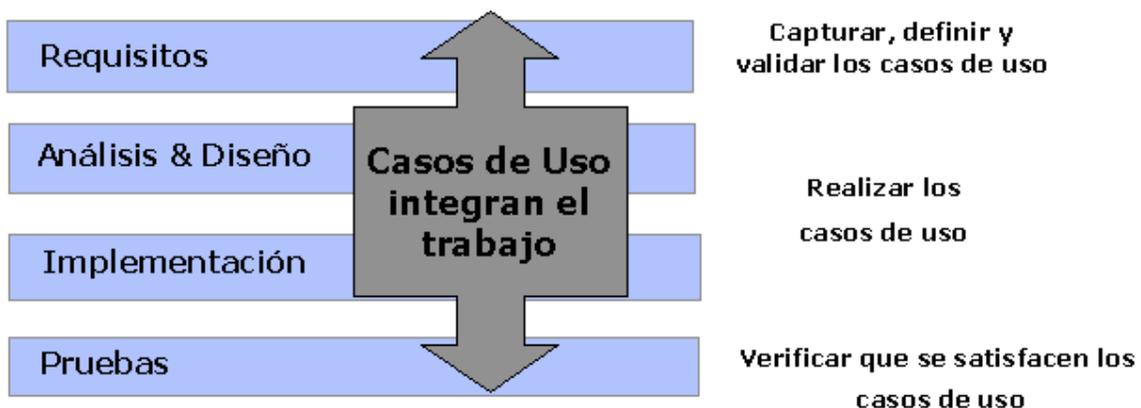


Figura 2. Casos de uso integran el trabajo

Los Casos de Uso no sólo inician el proceso de desarrollo sino que proporcionan un hilo conductor, permitiendo establecer trazabilidad entre los artefactos que son generados en las diferentes actividades del proceso de desarrollo.

Como se muestra en la Figura 3, basándose en los Casos de Uso se crean los modelos de análisis y diseño, luego la implementación que los lleva a cabo, y se verifica que efectivamente el producto implemente adecuadamente cada Caso de Uso. Todos los modelos deben estar sincronizados con el modelo de Casos de Uso.

¹ Kruchten, P., The Rational Unified Process: An Introduction, 2000 Addison Wesley

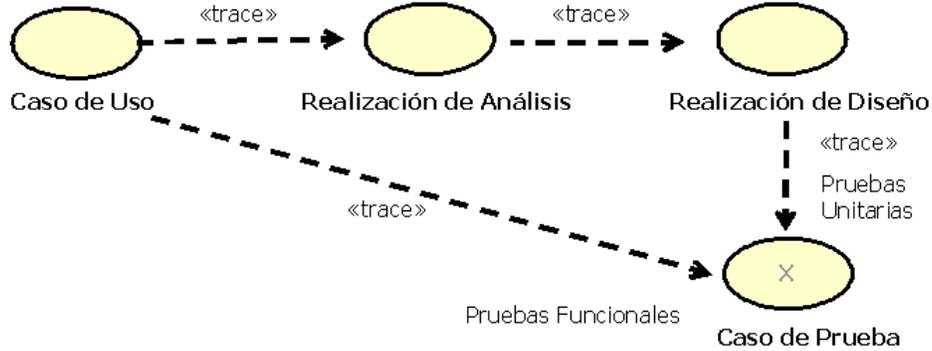


Figura 3. Trazabilidad a partir de los casos de uso

Proceso centrado en la arquitectura

La arquitectura de un sistema es la organización o estructura de sus partes más relevantes, lo que permite tener una visión común entre todos los involucrados (desarrolladores y usuarios) y una perspectiva clara del sistema completo, necesaria para controlar el desarrollo [Kru00].

La arquitectura involucra los aspectos estáticos y dinámicos más significativos del sistema, está relacionada con la toma de decisiones que indican cómo tiene que ser construido el sistema y ayuda a determinar en qué orden. Además la definición de la arquitectura debe tomar en consideración elementos de calidad del sistema, rendimiento, reutilización y capacidad de evolución por lo que debe ser flexible durante todo el proceso de desarrollo. La arquitectura se ve influenciada por la plataforma software, sistema operativo, gestor de bases de datos, protocolos, consideraciones de desarrollo como sistemas heredados. Muchas de estas restricciones constituyen requisitos no funcionales del sistema.

En el caso de RUP además de utilizar los Casos de Uso para guiar el proceso se presta especial atención al establecimiento temprano de una buena arquitectura que no se vea fuertemente impactada ante cambios posteriores durante la construcción y el mantenimiento.

Cada producto tiene tanto una función como una forma. La función corresponde a la funcionalidad reflejada en los Casos de Uso y la forma la proporciona la arquitectura. Existe una interacción entre los Casos de Uso y la arquitectura, los Casos de Uso deben encajar en la arquitectura cuando se llevan a cabo y la arquitectura debe permitir el desarrollo de todos los Casos de Uso requeridos, actualmente y en el futuro. Esto provoca que tanto arquitectura como Casos de Uso deban evolucionar en paralelo durante todo el proceso de desarrollo de software.

En la Figura 4 se ilustra la evolución de la arquitectura durante las fases de RUP. Se tiene una arquitectura más robusta en las fases finales del proyecto. En las fases iniciales lo que se hace es ir consolidando la arquitectura por medio de *baselines* y se va modificando dependiendo de las necesidades del proyecto.

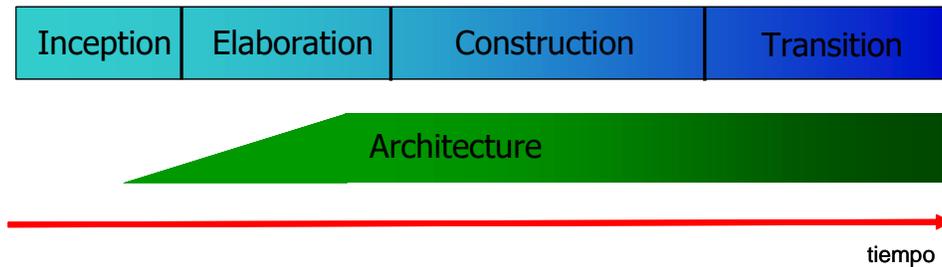


Figura 4. Arquitectura en fases

Es conveniente ver el sistema desde diferentes perspectivas para comprender mejor el diseño por lo que la arquitectura se representa mediante varias vistas que se centran en aspectos concretos del sistema, abstrayéndose de los demás. Para RUP, todas las vistas juntas forman el llamado modelo 4+1 de la arquitectura el cual recibe este nombre porque lo forman las vistas lógica, de implementación, de proceso y de despliegue, más la de Casos de Uso que es la que da cohesión a todas.

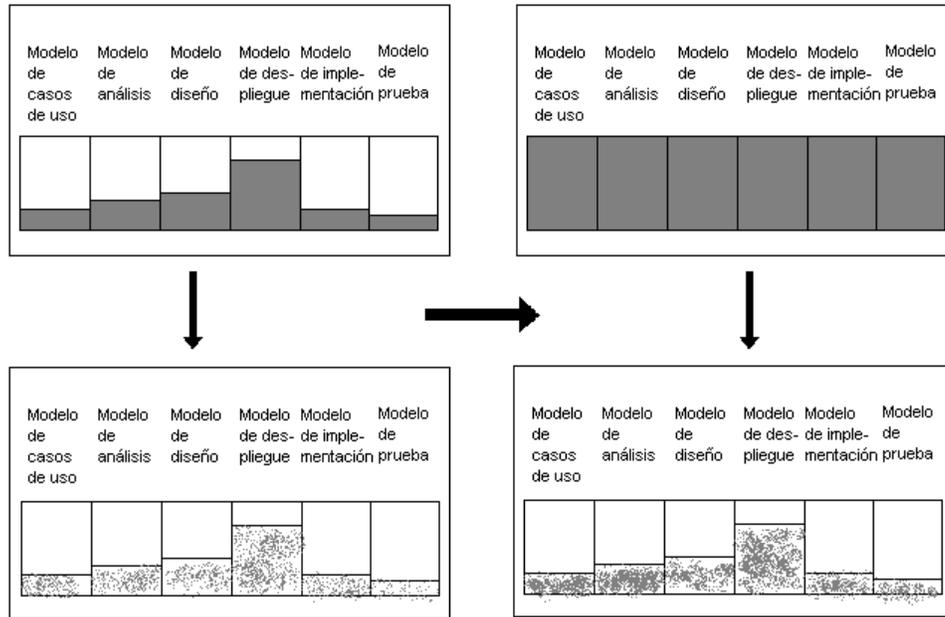


Figura 5. Perspectivas

Al final de la fase de elaboración se obtiene una *baseline*² de la arquitectura donde fueron seleccionados una serie de Casos de Uso arquitectónicamente relevantes (aquellos que ayudan a mitigar los riesgos más importantes, aquellos que son los más importantes para el usuario y aquellos que cubran las funcionalidades significativas)

Como se observa en la Figura 5, durante la construcción los diversos modelos van desarrollándose hasta completarse (según se muestra con las formas rellenas en la esquina superior derecha). La descripción de la arquitectura sin embargo, no debería cambiar significativamente (abajo a la derecha) debido a que la mayor parte de la arquitectura se decidió durante la elaboración. Se incorporan pocos cambios a la arquitectura (indicados con mayor densidad de puntos en la figura inferior derecha)

1.1 Proceso iterativo e incremental

Según³ el equilibrio correcto entre los Casos de Uso y la arquitectura es algo muy parecido al equilibrio de la forma y la función en el desarrollo del producto, lo cual se consigue con el tiempo. Para esto, la estrategia que se propone en RUP es tener un proceso iterativo e incremental en donde el trabajo se divide en partes más pequeñas o mini proyectos. Permitiendo que el equilibrio entre Casos de Uso y arquitectura se vaya logrando durante cada mini proyecto, así durante todo el proceso de desarrollo. Cada mini proyecto se puede ver como una iteración (un recorrido más o

³ Jacoboson, I., Booch, G., Rumbaugh J., El Proceso Unificado de Desarrollo de Software, 2000 Addison Wesley

menos completo a lo largo de todos los flujos de trabajo fundamentales) del cual se obtiene un incremento que produce un crecimiento en el producto.

Una iteración puede realizarse por medio de una cascada como se muestra en la Figura 6. Se pasa por los flujos fundamentales (Requisitos, Análisis, Diseño, Implementación y Pruebas), también existe una planificación de la iteración, un análisis de la iteración y algunas actividades específicas de la iteración. Al finalizar se realiza una integración de los resultados con lo obtenido de las iteraciones anteriores.

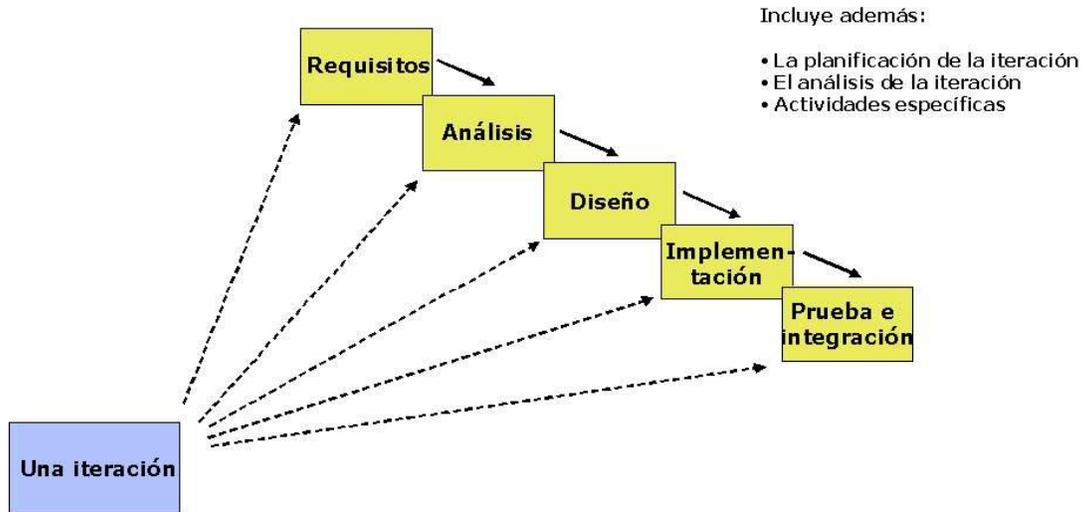


Figura 6. Iteración de RUP

El proceso iterativo e incremental consta de una secuencia de iteraciones. Cada iteración aborda una parte de la funcionalidad total, pasando por todos los flujos de trabajo relevantes y refinando la arquitectura. Cada iteración se analiza cuando termina. Se puede determinar si han aparecido nuevos requisitos o han cambiado los existentes, afectando a las iteraciones siguientes. Durante la planificación de los detalles de la siguiente iteración, el equipo también examina cómo afectarán los riesgos que aún quedan al trabajo en curso. Toda la retroalimentación de la iteración pasada permite reajustar los objetivos para las siguientes iteraciones. Se continúa con esta dinámica hasta que se haya finalizado por completo con la versión actual del producto.

RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades. En la Figura 7 se muestra cómo varía el esfuerzo asociado a las disciplinas según la fase en la que se encuentre el proyecto RUP.

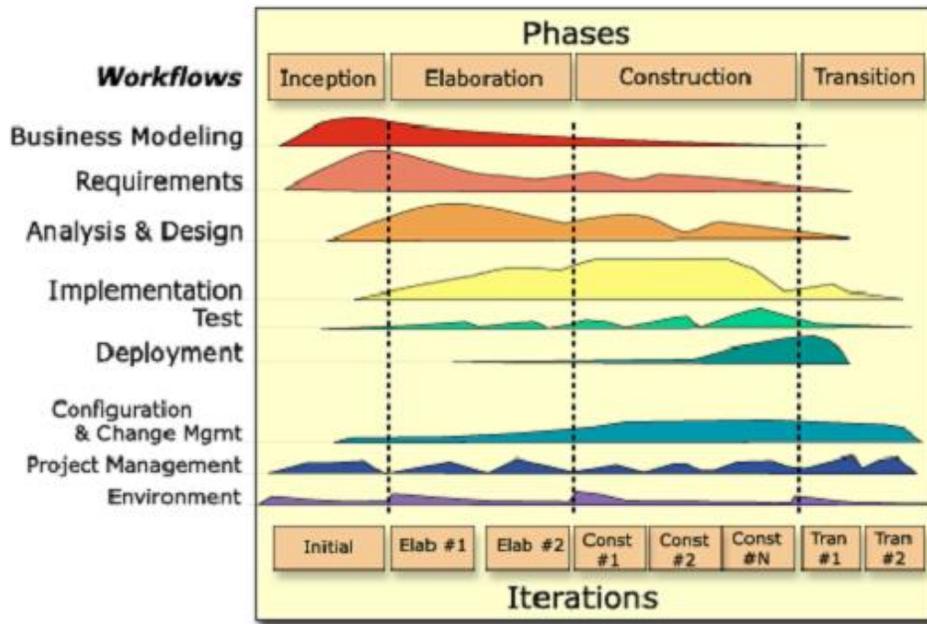


Figura 7. Esfuerzo en actividades según fase del proyecto

Las primeras iteraciones (en las fases de Inicio y Elaboración) se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de una *baseline* de la arquitectura.

Durante la fase de inicio las iteraciones hacen poner mayor énfasis en actividades modelado del negocio y de requisitos.

En la fase de elaboración, las iteraciones se orientan al desarrollo de la *baseline* de la arquitectura, abarcan más los flujos de trabajo de requerimientos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación orientado a la *baseline* de la arquitectura.

En la fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones.

Para cada iteración se selecciona algunos Casos de Uso, se refina su análisis y diseño y se procede a su implementación y pruebas. Se realiza una pequeña cascada para cada ciclo. Se realizan tantas iteraciones hasta que se termine la implementación de la nueva versión del producto.

En la fase de transición se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios.

Como se puede observar en cada fase participan todas las disciplinas, pero que dependiendo de la fase el esfuerzo dedicado a una disciplina varía.

Otras prácticas

RUP identifica 6 *best practices* con las que define una forma efectiva de trabajar para los equipos de desarrollo de software.

Gestión de requisitos

RUP brinda una guía para encontrar, organizar, documentar, y seguir los cambios de los requisitos funcionales y restricciones. Utiliza una notación de Caso de Uso y escenarios para representar los requisitos.

Desarrollo de software iterativo

Desarrollo del producto mediante iteraciones con hitos bien definidos, en las cuales se repiten las actividades pero con distinto énfasis, según la fase del proyecto.

Desarrollo basado en componentes

La creación de sistemas intensivos en software requiere dividir el sistema en componentes con interfaces bien definidas, que posteriormente serán ensamblados para generar el sistema. Esta característica en un proceso de desarrollo permite que el sistema se vaya creando a medida que se obtienen o se desarrollan sus componentes.

Modelado visual (usando UML)

UML es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema software. Es un estándar de la OMG (<http://www.omg.org>). Utilizar herramientas de modelado visual facilita la gestión de dichos modelos, permitiendo ocultar o exponer detalles cuando sea necesario. El modelado visual también ayuda a mantener la consistencia entre los artefactos del sistema: requisitos, diseños e implementaciones. En resumen, el modelado visual ayuda a mejorar la capacidad del equipo para gestionar la complejidad del software.

Verificación continua de la calidad

Es importante que la calidad de todos los artefactos se evalúe en varios puntos durante el proceso de desarrollo, especialmente al final de cada iteración. En esta verificación las pruebas juegan un papel fundamental y se integran a lo largo de todo el proceso. Para todos los artefactos no ejecutables las revisiones e inspecciones también deben ser continuas.

Gestión de los cambios

El cambio es un factor de riesgo crítico en los proyectos de software. Los artefactos software cambian no sólo debido a acciones de mantenimiento posteriores a la entrega del producto, sino que durante el proceso de desarrollo, especialmente importantes por su posible impacto son los cambios en los requisitos. Por otra parte, otro gran desafío que debe abordarse es la construcción de software con la participación de múltiples desarrolladores, posiblemente distribuidos geográficamente, trabajando a la vez en una *release*, y quizás en distintas plataformas. La ausencia de disciplina rápidamente conduciría al caos. La Gestión de Cambios y de Configuración es la disciplina de RUP encargada de este aspecto.

Estructura del proceso

El proceso puede ser descrito en dos dimensiones o ejes

Eje horizontal: Representa el tiempo y es considerado el eje de los aspectos dinámicos del proceso. Indica las características del ciclo de vida del proceso expresado en términos de fases, iteraciones e hitos. Se puede observar en la Figura 8 que RUP consta de cuatro fases: Inicio, Elaboración, Construcción y Transición. Como se mencionó anteriormente cada fase se subdivide a la vez en iteraciones.

Eje vertical: Representa los aspectos estáticos del proceso. Describe el proceso en términos de componentes de proceso, disciplinas, flujos de trabajo, actividades, artefactos y roles.

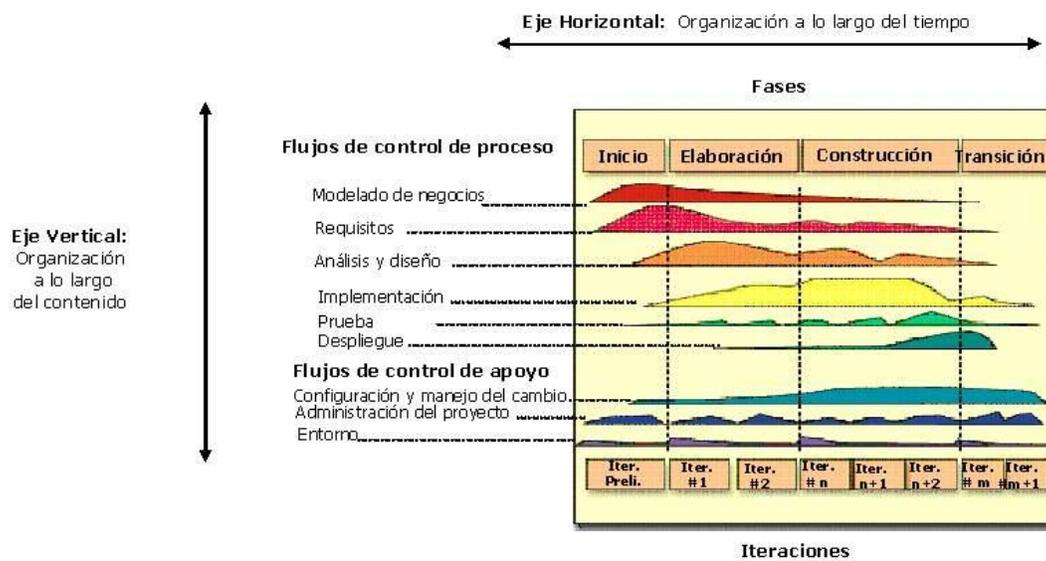


Figura 8. Estructura de RUP

Estructura Dinámica del proceso. Fases e iteraciones

RUP se repite a lo largo de una serie de ciclos que constituyen la vida de un producto. Cada ciclo concluye con una generación del producto para los clientes. Cada ciclo consta de cuatro fases: Inicio, Elaboración, Construcción y Transición. Cada fase se subdivide a la vez en iteraciones, el número de iteraciones en cada fase es variable.

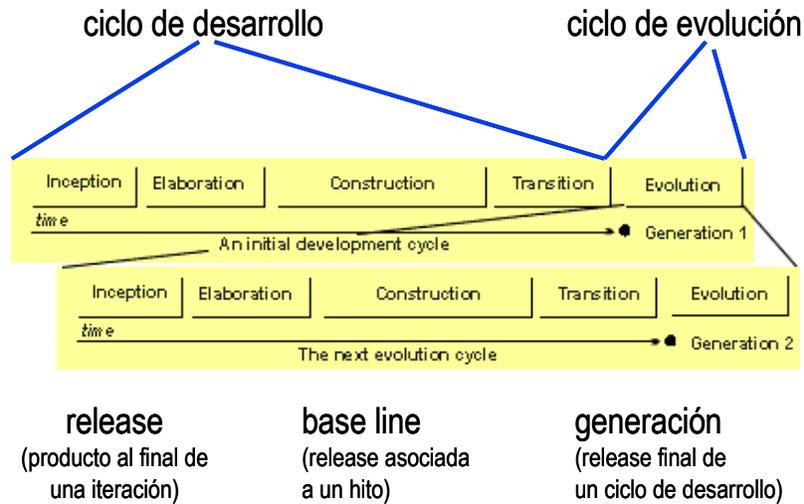


Figura 9. Ciclos

Cada fase se concluye con un hito bien definido, un punto en el tiempo en el cual se deben tomar ciertas decisiones críticas y alcanzar las metas clave antes de pasar a la siguiente fase, ese hito principal de cada fase se compone de hitos menores que podrían ser los criterios aplicables a cada iteración. Los hitos para cada una de las fases son: Inicio - *Lifecycle Objectives*, Elaboración - *Lifecycle Architecture*, Construcción - *Initial Operational Capability*, Transición - *Product Release*. Las fases y sus respectivos hitos se ilustran en la Figura 10.

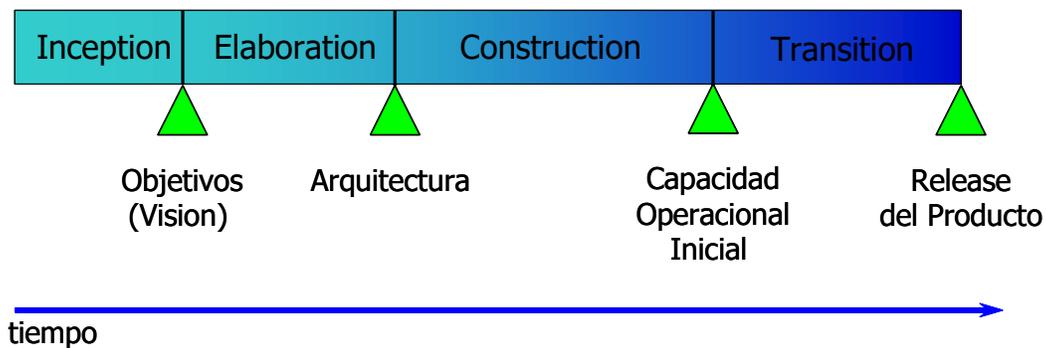


Figura 10. Fases e hitos

La duración y esfuerzo dedicado en cada fase es variable dependiendo de las características del proyecto. Sin embargo, la Figura 11 ilustra porcentajes frecuentes al respecto. Consecuente con el esfuerzo señalado, la Figura 12 ilustra una distribución típica de recursos humanos necesarios a lo largo del proyecto.

Tabla 8. Esfuerzo en cada fase

	Inicio	Elaboración	Construcción	Transición
Esfuerzo	5 %	20 %	65 %	10%
Tiempo Dedicado	10 %	30 %	50 %	10%

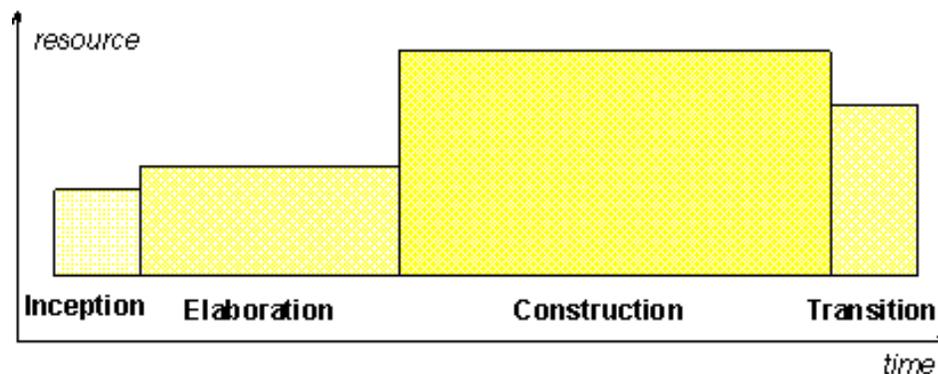


Figura 11. Distribución típica de recursos humanos

Inicio

Durante la fase de inicio se define el modelo del negocio y el alcance del proyecto. Se identifican todos los actores y Casos de Uso, y se diseñan los Casos de Uso más esenciales (aproximadamente el 20% del modelo completo). Se desarrolla, un plan de negocio para determinar qué recursos deben ser asignados al proyecto.

Los objetivos de esta fase son:

- Establecer el ámbito del proyecto y sus límites.
- Encontrar los Casos de Uso críticos del sistema, los escenarios básicos que definen la funcionalidad.
- Mostrar al menos una arquitectura candidata para los escenarios principales.
- Estimar el coste en recursos y tiempo de todo el proyecto.

- Estimar los riesgos, las fuentes de incertidumbre.

Los resultados de la fase de inicio deben ser:

- Un documento de visión: Una visión general de los requerimientos del proyecto, características clave y restricciones principales.
- Modelo inicial de Casos de Uso (10-20% completado).
- Un glosario inicial: Terminología clave del dominio.
- El caso de negocio.
- Lista de riesgos y plan de contingencia.
- Plan del proyecto, mostrando fases e iteraciones.
- Modelo de negocio, si es necesario
- Prototipos exploratorios para probar conceptos o la arquitectura candidata.

Al terminar la fase de inicio se deben comprobar los criterios de evaluación para continuar:

- Todos los interesados en el proyecto coinciden en la definición del ámbito del sistema y las estimaciones de agenda.
- Entendimiento de los requisitos, como evidencia de la fidelidad de los Casos de Uso principales.
- Las estimaciones de tiempo, coste y riesgo son creíbles.
- Comprensión total de cualquier prototipo de la arquitectura desarrollado.
- Los gastos hasta el momento se asemejan a los planeados.

Si el proyecto no pasa estos criterios hay que plantearse abandonarlo o repensarlo profundamente.

Elaboración

El propósito de la fase de elaboración es analizar el dominio del problema, establecer los cimientos de la arquitectura, desarrollar el plan del proyecto y eliminar los mayores riesgos.

En esta fase se construye un prototipo de la arquitectura, que debe evolucionar en iteraciones sucesivas hasta convertirse en el sistema final. Este prototipo debe contener los Casos de Uso críticos identificados en la fase de inicio. También debe demostrarse que se han evitado los riesgos más graves.

Los objetivos de esta fase son:

- Definir, validar y cimentar la arquitectura.
- Completar la visión.
- Crear un plan fiable para la fase de construcción. Este plan puede evolucionar en sucesivas iteraciones. Debe incluir los costes si procede.
- Demostrar que la arquitectura propuesta soportará la visión con un coste razonable y en un tiempo razonable.

Al terminar deben obtenerse los siguientes resultados:

- Un modelo de Casos de Uso completa al menos hasta el 80%: todos los casos y actores identificados, la mayoría de los casos desarrollados.
- Requisitos adicionales que capturan los requisitos no funcionales y cualquier requisito no asociado con un Caso de Uso específico.

- Descripción de la arquitectura software.
- Un prototipo ejecutable de la arquitectura.
- Lista de riesgos y caso de negocio revisados.
- Plan de desarrollo para el proyecto.
- Un caso de desarrollo actualizado que especifica el proceso a seguir.
- Un manual de usuario preliminar (opcional).

En esta fase se debe tratar de abarcar todo el proyecto con la profundidad mínima. Sólo se profundiza en los puntos críticos de la arquitectura o riesgos importantes.

En la fase de elaboración se actualizan todos los productos de la fase de inicio.

Los criterios de evaluación de esta fase son los siguientes:

- La visión del producto es estable.
- La arquitectura es estable.
- Se ha demostrado mediante la ejecución del prototipo que los principales elementos de riesgo han sido abordados y resueltos.
- El plan para la fase de construcción es detallado y preciso. Las estimaciones son creíbles.
- Todos los interesados coinciden en que la visión actual será alcanzada si se siguen los planes actuales en el contexto de la arquitectura actual.
- Los gastos hasta ahora son aceptables, comparados con los previstos.

Si no se superan los criterios de evaluación quizá sea necesario abandonar el proyecto o replanteárselo considerablemente.

Construcción

La finalidad principal de esta fase es alcanzar la capacidad operacional del producto de forma incremental a través de las sucesivas iteraciones. Durante esta fase todos los componentes, características y requisitos deben ser implementados, integrados y probados en su totalidad, obteniendo una versión aceptable del producto.

Los objetivos concretos según incluyen:

- Minimizar los costes de desarrollo mediante la optimización de recursos y evitando el tener que rehacer un trabajo o incluso desecharlo.
- Conseguir una calidad adecuada tan rápido como sea práctico.
- Conseguir versiones funcionales (alfa, beta, y otras versiones de prueba) tan rápido como sea práctico.

Los resultados de la fase de construcción deben ser:

- Modelos Completos (Casos de Uso, Análisis, Diseño, Despliegue e Implementación)
- Arquitectura íntegra (mantenida y mínimamente actualizada)
- Riesgos Presentados Mitigados
- Plan del Proyecto para la fase de Transición.
- Manual Inicial de Usuario (con suficiente detalle)
- Prototipo Operacional – beta
- Caso del Negocio Actualizado

Los criterios de evaluación de esta fase son los siguientes:

- El producto es estable y maduro como para ser entregado a la comunidad de usuario para ser probado.
- Todos los usuarios expertos están listos para la transición en la comunidad de usuarios.
- Son aceptables los gastos actuales versus los gastos planeados.

Transición

La finalidad de la fase de transición es poner el producto en manos de los usuarios finales, para lo que se requiere desarrollar nuevas versiones actualizadas del producto, completar la documentación, entrenar al usuario en el manejo del producto, y en general tareas relacionadas con el ajuste, configuración, instalación y facilidad de uso del producto.

Se citan algunas de las cosas que puede incluir esta fase:

- Prueba de la versión Beta para validar el nuevo sistema frente a las expectativas de los usuarios
- Funcionamiento paralelo con los sistemas legados que están siendo sustituidos por nuestro proyecto.
- Conversión de las bases de datos operacionales.
- Entrenamiento de los usuarios y técnicos de mantenimiento.
- Traspaso del producto a los equipos de marketing, distribución y venta.

Los principales objetivos de esta fase son:

- Conseguir que el usuario se valga por si mismo.
- Un producto final que cumpla los requisitos esperados, que funcione y satisfaga suficientemente al usuario.

Los resultados de la fase de transición son:

- Prototipo Operacional
- Documentos Legales
- Caso del Negocio Completo
- Línea de Base del Producto completa y corregida que incluye todos los modelos del sistema
- Descripción de la Arquitectura completa y corregida
- Las iteraciones de esta fase irán dirigidas normalmente a conseguir una nueva versión.

Los criterios de evaluación de esta fase son los siguientes:

- El usuario se encuentra satisfecho.
- Son aceptables los gastos actuales versus los gastos planificados.

Estructura Estática del proceso. Roles, actividades, artefactos y flujos de trabajo

Un proceso de desarrollo de software define quién hace qué, cómo y cuándo. RUP define cuatro elementos los roles, que responden a la pregunta ¿Quién?, las actividades que responden a la pregunta ¿Cómo?, los productos, que responden a la pregunta ¿Qué? y los flujos de trabajo de las disciplinas que responde a la pregunta ¿Cuándo? (ver Figura 13 y 14).



Figura 12. Roles, actividades, artefactos

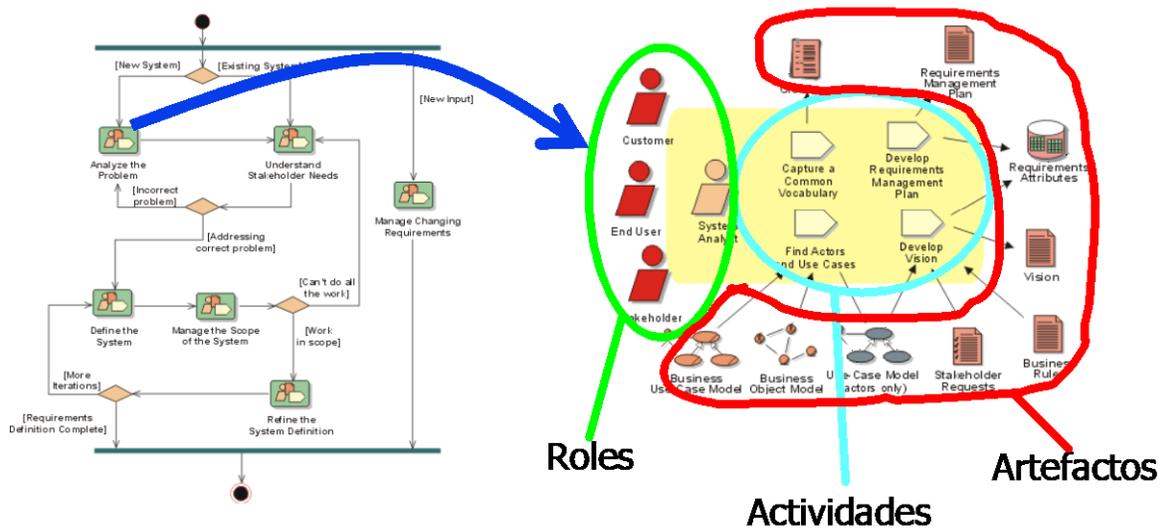


Figura 13. Flujo de trabajo

Roles

Un rol define el comportamiento y responsabilidades de un individuo, o de un grupo de individuos trabajando juntos como un equipo. Una persona puede desempeñar diversos roles, así como un mismo rol puede ser representado por varias personas.

Las responsabilidades de un rol son tanto el llevar a cabo un conjunto de actividades como el ser el dueño de un conjunto de artefactos.

RUP define grupos de roles, agrupados por participación en actividades relacionadas. Estos grupos son:

Analistas:

- Analista de procesos de negocio.
- Diseñador del negocio.
- Analista de sistema.
- Especificador de requisitos.

Desarrolladores:

- Arquitecto de software.
- Diseñador
- Diseñador de interfaz de usuario
- Diseñador de cápsulas.
- Diseñador de base de datos.
- Implementador.
- Integrador.

Gestores:

- Jefe de proyecto
- Jefe de control de cambios.
- Jefe de configuración.
- Jefe de pruebas
- Jefe de despliegue
- Ingeniero de procesos
- Revisor de gestión del proyecto
- Gestor de pruebas.

Apoyo:

- Documentador técnico
- Administrador de sistema
- Especialista en herramientas
- Desarrollador de cursos
- Artista gráfico

Especialista en pruebas:

- Especialista en Pruebas (*tester*)
- Analista de pruebas
- Diseñador de pruebas

Otros roles:

- *Stakeholders.*
- Revisor
- Coordinación de revisiones
- Revisor técnico
- Cualquier rol

Actividades

Una actividad en concreto es una unidad de trabajo que una persona que desempeñe un rol puede ser solicitado a que realice. Las actividades tienen un objetivo concreto, normalmente expresado en términos de crear o actualizar algún producto.

Artefactos

Un producto o artefacto es un trozo de información que es producido, modificado o usado durante el proceso de desarrollo de software. Los productos son los resultados tangibles del proyecto, las cosas que va creando y usando hasta obtener el producto final.

Un artefacto puede ser cualquiera de los siguientes:

- Un documento, como el documento de la arquitectura del software.
- Un modelo, como el modelo de Casos de Uso o el modelo de diseño.
- Un elemento del modelo, un elemento que pertenece a un modelo como una clase, un Caso de Uso o un subsistema.

Flujos de trabajo

Con la enumeración de roles, actividades y artefactos no se define un proceso, necesitamos contar con una secuencia de actividades realizadas por los diferentes roles, así como la relación entre los mismos. Un flujo de trabajo es una relación de actividades que nos producen unos resultados observables. A continuación se dará una explicación de cada flujo de trabajo.

Modelado del negocio

Con este flujo de trabajo pretendemos llegar a un mejor entendimiento de la organización donde se va a implantar el producto.

Los objetivos del modelado de negocio son:

- Entender la estructura y la dinámica de la organización para la cual el sistema va ser desarrollado (organización objetivo).
- Entender el problema actual en la organización objetivo e identificar potenciales mejoras.
- Asegurar que clientes, usuarios finales y desarrolladores tengan un entendimiento común de la organización objetivo.
- Derivar los requisitos del sistema necesarios para apoyar a la organización objetivo.

Para lograr estos objetivos, el modelo de negocio describe como desarrollar una visión de la nueva organización, basado en esta visión se definen procesos, roles y responsabilidades de la organización por medio de un modelo de Casos de Uso del negocio y un Modelo de Objetos del Negocio. Complementario a estos modelos, se desarrollan otras especificaciones tales como un Glosario.

Requisitos

Este es uno de los flujos de trabajo más importantes, porque en él se establece qué tiene que hacer exactamente el sistema que construyamos. En esta línea los requisitos son el contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar los requisitos que especifiquemos.

Los objetivos del flujo de datos Requisitos es:

- Establecer y mantener un acuerdo entre clientes y otros *stakeholders* sobre lo que el sistema podría hacer.
- Proveer a los desarrolladores un mejor entendimiento de los requisitos del sistema.
- Definir el ámbito del sistema.
- Proveer una base para la planeación de los contenidos técnicos de las iteraciones.
- Proveer una base para estimar costos y tiempo de desarrollo del sistema.
- Definir una interfaz de usuarios para el sistema, enfocada a las necesidades y metas del usuario.

Los requisitos se dividen en dos grupos. Los requisitos funcionales representan la funcionalidad del sistema. Se modelan mediante diagramas de Casos de Uso. Los requisitos no funcionales representan aquellos atributos que debe exhibir el sistema, pero que no son una funcionalidad específica. Por ejemplo requisitos de facilidad de uso, fiabilidad, eficiencia, portabilidad, etc.

Para capturar los requisitos es preciso entrevistar a todos los interesados en el proyecto, no sólo a los usuarios finales, y anotar todas sus peticiones. A partir de ellas hay que descubrir lo que necesitan y expresarlo en forma de requisitos.

En este flujo de trabajo, y como parte de los requisitos de facilidad de uso, se diseña la interfaz gráfica de usuario. Para ello habitualmente se construyen prototipos de la interfaz gráfica de usuario que se contrastan con el usuario final.

Análisis y Diseño

El objetivo de este flujo de trabajo es traducir los requisitos a una especificación que describe cómo implementar el sistema.

Los objetivos del análisis y diseño son :

- Transformar los requisitos al diseño del futuro sistema.
- Desarrollar una arquitectura para el sistema.
- Adaptar el diseño para que sea consistente con el entorno de implementación, diseñando para el rendimiento.

El análisis consiste en obtener una visión del sistema que se preocupa de ver qué hace, de modo que sólo se interesa por los requisitos funcionales. Por otro lado el diseño es un refinamiento del

análisis que tiene en cuenta los requisitos no funcionales, en definitiva cómo cumple el sistema sus objetivos.

Al principio de la fase de elaboración hay que definir una arquitectura candidata: crear un esquema inicial de la arquitectura del sistema, identificar clases de análisis y actualizar las realizaciones de los Casos de Uso con las interacciones de las clases de análisis. Durante la fase de elaboración se va refinando esta arquitectura hasta llegar a su forma definitiva. En cada iteración hay que analizar el comportamiento para diseñar componentes. Además si el sistema usará una base de datos, habrá que diseñarla también, obteniendo un modelo de datos.

El resultado final más importante de este flujo de trabajo será el modelo de diseño. Consiste en colaboraciones de clases, que pueden ser agregadas en paquetes y subsistemas.

Otro producto importante de este flujo es la documentación de la arquitectura de software, que captura varias vistas arquitectónicas del sistema.

Implementación

En este flujo de trabajo se implementan las clases y objetos en ficheros fuente, binarios, ejecutables y demás. Además se deben hacer las pruebas de unidad: cada implementador es responsable de probar las unidades que produzca. El resultado final de este flujo de trabajo es un sistema ejecutable.

En cada iteración habrá que hacer lo siguiente:

- Planificar qué subsistemas deben ser implementados y en que orden deben ser integrados, formando el Plan de Integración.
- Cada implementador decide en que orden implementa los elementos del subsistema.
- Si encuentra errores de diseño, los notifica.
- Se prueban los subsistemas individualmente.
- Se integra el sistema siguiendo el plan.

La estructura de todos los elementos implementados forma el modelo de implementación. La integración debe ser incremental, es decir, en cada momento sólo se añade un elemento. De este modo es más fácil localizar fallos y los componentes se prueban más a fondo. En fases tempranas del proceso se pueden implementar prototipos para reducir el riesgo. Su utilidad puede ir desde ver si el sistema es viable desde el principio, probar tecnologías o diseñar la interfaz de usuario. Los prototipos pueden ser exploratorios (desechables) o evolutivos. Estos últimos llegan a transformarse en el sistema final.

Pruebas

Este flujo de trabajo es el encargado de evaluar la calidad del producto que estamos desarrollando, pero no para aceptar o rechazar el producto al final del proceso de desarrollo, sino que debe ir integrado en todo el ciclo de vida.

Esta disciplina brinda soporte a las otras disciplinas. Sus objetivos son :

- Encontrar y documentar defectos en la calidad del software.
- Generalmente asesora sobre la calidad del software percibida.
- Provee la validación de los supuestos realizados en el diseño y especificación de requisitos por medio de demostraciones concretas.

- Verificar las funciones del producto de software según lo diseñado.
- Verificar que los requisitos tengan su apropiada implementación.

Las actividades de este flujo comienzan pronto en el proyecto con el plan de prueba (el cual contiene información sobre los objetivos generales y específicos de las prueba en el proyecto, así como las estrategias y recursos con que se dotará a esta tarea), o incluso antes con alguna evaluación durante la fase de inicio, y continuará durante todo el proyecto.

El desarrollo del flujo de trabajo consistirá en planificar que es lo que hay que probar, diseñar cómo se va a hacer, implementar lo necesario para llevarlos a cabo, ejecutarlos en los niveles necesarios y obtener los resultados, de forma que la información obtenida nos sirva para ir refinando el producto a desarrollar.

Despliegue

El objetivo de este flujo de trabajo es producir con éxito distribuciones del producto y distribuirlo a los usuarios. Las actividades implicadas incluyen:

- Probar el producto en su entorno de ejecución final.
- Empaquetar el software para su distribución.
- Distribuir el software.
- Instalar el software.
- Proveer asistencia y ayuda a los usuarios.
- Formar a los usuarios y al cuerpo de ventas.
- Migrar el software existente o convertir bases de datos.

Este flujo de trabajo se desarrolla con mayor intensidad en la fase de transición, ya que el propósito del flujo es asegurar una aceptación y adaptación sin complicaciones del software por parte de los usuarios. Su ejecución inicia en fases anteriores, para preparar el camino, sobre todo con actividades de planificación, en la elaboración del manual de usuario y tutoriales.

Gestión del proyecto

La Gestión del proyecto es el arte de lograr un balance al gestionar objetivos, riesgos y restricciones para desarrollar un producto que sea acorde a los requisitos de los clientes y los usuarios.

Los objetivos de este flujo de trabajo son:

- Proveer un marco de trabajo para la gestión de proyectos de software intensivos.
- Proveer guías prácticas realizar planeación, contratar personal, ejecutar y monitorear el proyecto.
- Proveer un marco de trabajo para gestionar riesgos.

La planeación de un proyecto posee dos niveles de abstracción: un plan para las fases y un plan para cada iteración.

Configuración y control de cambios

La finalidad de este flujo de trabajo es mantener la integridad de todos los artefactos que se crean en el proceso, así como de mantener información del proceso evolutivo que han seguido.

Entorno

La finalidad de este flujo de trabajo es dar soporte al proyecto con las adecuadas herramientas, procesos y métodos. Brinda una especificación de las herramientas que se van a necesitar en cada momento, así como definir la instancia concreta del proceso que se va a seguir.

En concreto las responsabilidades de este flujo de trabajo incluyen:

- Selección y adquisición de herramientas
- Establecer y configurar las herramientas para que se ajusten a la organización.
- Configuración del proceso.
- Mejora del proceso.
- Servicios técnicos.

El principal artefacto que se usa en este flujo de trabajo es el *caso de desarrollo* que especifica para el proyecto actual en concreto, como se aplicará el proceso, que productos se van a utilizar y como van a ser utilizados. Además se tendrán que definir las guías para los distintos aspectos del proceso, como pueden ser el modelado del negocio y los Casos de Uso, para la interfaz de usuario, el diseño, la programación, el manual de usuario.

Una configuración RUP para proyecto pequeño

En este apartado se describe una posible configuración de RUP para un proyecto pequeño. Por las características del proyecto, se han incluido muy pocos artefactos, roles y actividades de la metodología, manteniendo los más esenciales. Dicha configuración está basada en la siguiente selección de artefactos:

Entregables del proyecto

A continuación se describen brevemente cada uno de los artefactos que se generarán y usarán durante el proyecto.

1. Flujos de Trabajo

Se utilizarán Diagramas de Actividad para modelar los Flujos de Trabajo (workflows) del área problema, tanto los actuales (previos a la implantación de nuevo sistema) como los propuestos, que serán soportados por el sistema desarrollado

2. Características del Producto Software

Es una lista de las características principales del producto, deseables desde una perspectiva de las necesidades del cliente.

3. Glosario

Es un documento que define los principales términos usados en el proyecto. Permite establecer una terminología consensuada.

4. Modelo de Casos de Uso

El modelo de Casos de Uso presenta la funcionalidad del sistema y los actores que hacen uso de ella. Se representa mediante Diagramas de Casos de Uso.

5. Especificaciones de Casos de Uso

Para los casos de uso que lo requieran (cuya funcionalidad no sea evidente o que no baste con una simple descripción narrativa) se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, postcondiciones, flujo de eventos, requisitos no-funcionales asociados.

6. Modelo de Análisis y Diseño

Este modelo establece la realización de los casos de uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación) hacia una de diseño (incluyendo una orientación hacia el entorno de implementación). Está constituido esencialmente por un Diagrama de Clases y algunos Diagramas de Estados para las clases que lo requieran.

7. Modelo Lógico Relacional

Previendo que la persistencia de la información del sistema será soportada por una base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este modelo se utiliza un Diagrama de Tablas donde se muestran las tablas, claves, etc.

8. Modelo de Implementación

Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema.

9. Modelo de Pruebas

Para cada Caso de Uso se establecen pruebas de Aceptación que validarán la correcta implementación del Caso de Uso. Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados.

10. Manual de Instalación

Este documento incluye las instrucciones para realizar la instalación del producto.

11. Material de Usuario

Corresponde a un conjunto de documentos y facilidades de uso del sistema.

12. Producto

Todos los ficheros fuente y ejecutable del producto.

1.2 Esquema de trazabilidad

La Figura 15 ilustra las relaciones de trazabilidad entre artefactos del proyecto, y según la configuración antes mencionada.

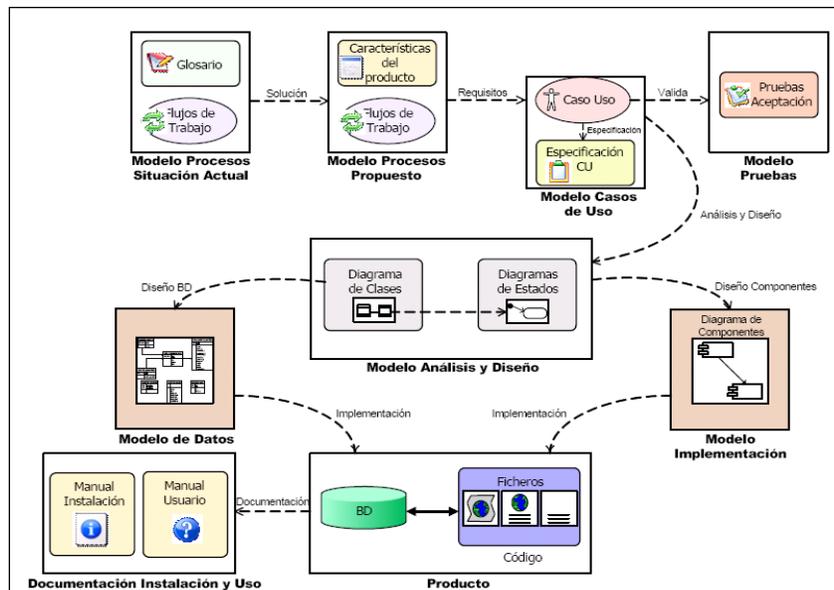


Figura 14. Trazabilidad entre artefactos

Las relaciones de trazabilidad son enlaces entre artefactos que establecen cómo se generan unos a partir de otros. Esto permite por ejemplo asegurar la cobertura de los requisitos o determinar el posible impacto de los cambios. En la Figura 15 se ilustran los modelos y artefactos utilizados, indicando las relaciones de trazabilidad entre ellos, lo cual se resume a continuación:

- Se modelarán los procesos de negocio de la situación actual utilizando Diagramas de Actividad para representar Flujos de Trabajo Actuales. Esto se complementará mediante un Glosario que establecerá la terminología.
- El modelo de procesos de la solución propuesta incluirá Flujos de Trabajo Propuestos junto con una lista de Características del Producto Software.

- Los requisitos serán establecidos mediante un Modelo de Casos de Uso que incluirá Diagramas de Casos de Uso, Prototipos de Interfaces de Usuario y Especificaciones de Casos de Uso.
- El Modelo de Pruebas incluirá las Pruebas de Aceptación establecidas para cada Caso de Uso.
- El Modelo de Análisis y Diseño establecerá el particionamiento interno del sistema. Estará compuesto por un Diagrama de Clases y algunos Diagramas de Estados. Las clases determinarán la estructura y las operaciones necesarias para implementar las funcionalidades descritas en los Casos de Uso. Los Diagramas de Estados detallarán el comportamiento para las clases que lo requieran.
- A partir del Diagrama de Clases, y considerando las clases que requieran persistencia, se derivará el Modelo Lógico Relacional, representado mediante Diagramas de Tablas.
- En el Modelo de Implementación se organizarán las operaciones de las clases en términos de componentes de dicha arquitectura. Esto se representará mediante Diagramas de Componentes.
- La implementación del Modelo Lógico Relacional y de los componentes de la aplicación constituirán el Producto, el cual se complementará con el Manual de Instalación y el Manual de Usuario.

C. ANEXO MODELOS Y METODOLOGÍAS PARA EL DESARROLLO DE OBJETOS DE APRENDIZAJE

A continuación, se describen modelos y metodología para el desarrollo de objetos de aprendizaje, se presentan ventajas y desventajas de las propuestas analizadas.

Proceso Integral del Desarrollo de Objetos de Aprendizaje: Modelo Prescriptivo de Proceso Evolutivo [1]

Proyecto presentado por Rodríguez Verónica y Ayala Gerardo; Laboratorio ICT – Interactive and Cooperative Technologies Universidad de las Américas Puebla México En este proyecto se refleja la importancia de utilizar procesos de ingeniería de software para la construcción de objetos de aprendizaje, elementos que son considerados un producto global que deben crearse bajo un marco de trabajo genérico el cual consta de un conjunto de actividades, acciones y tareas en cada una de sus fases: Comunicación, Plantación, Modelado, Construcción y Despliegue. Los autores concluyen que el éxito de los objetos de aprendizaje depende de la adecuada aplicación del proceso de ingeniería de software y que el modelo planteado debe proporcionar un conjunto de pautas aplicables desde la creación del objeto de aprendizaje hasta su reutilización, sin embargo se hace poco hincapié en los aspectos pedagógicos que deben tener los OA. Los autores afirman en las fases iniciales de este proceso existe una intensa colaboración y comunicación con los clientes por medio de dos actividades: un análisis institucional y una formulación. El *análisis institucional* define el contexto organizativo para el objeto de aprendizaje. Se identifican los participantes (equipo de trabajo multidisciplinario) y se predicen los cambios potenciales en el ambiente educativo o los requisitos de la institución. La *formulación* es una actividad donde se trata de establecer un conjunto común de metas y objetivos para la construcción de los objetos de aprendizaje junto con los requisitos básicos que conducen al desarrollo de un modelo de análisis.

Las actividades convencionales de análisis y diseño de Ingeniería del Software se adaptan al desarrollo de objetos de aprendizaje. Estas actividades se combinan para dar como resultado la actividad de modelados de los objetos de aprendizaje, con el objeto de desarrollar análisis rápidos y modelos de diseño.

Metodología para el desarrollo de Objetos de Aprendizaje usando Patrones ¡Error! No se encuentra el origen de la referencia..

Esta metodología tiene en cuenta otras propuestas y las mejora en el sentido de uso de patrones, además no se considera solo importante la producción de OA, sino también la motivación de la comunidad educativa para que haga uso de los mismos. Este trabajo propone seguir una metodología que guíe paso a paso el proceso de construcción de objetos de aprendizaje, así como la consideración de patrones en su diseño, los cuales facilitan el proceso y permiten la producción de varios OA en serie. Además, plantea la importancia de los objetos en la adquisición de competencias específicas, permitiendo al estudiante ser responsable de su aprendizaje, permite al docente estar capacitado en el uso de nuevas tecnologías, pero para que todo esto sea posible es necesario, primeramente incursionar en su producción.

Esta metodología tiene en cuenta otras propuestas y las mejora en el sentido del uso de patrones y considerando la fase de seguimiento, además no se considera solo importante la producción de OA, sino también la motivación de la comunidad educativa para que los use, del mismo modo el

almacenamiento y la implantación y seguimiento de los objetos, para saber si estos están cumpliendo con su objetivo principal, que es el generar un aprendizaje, así como una determinada competencia. Las fases propuestas por esta metodología son:

Análisis y obtención: importante identificar una competencia a desarrollar, en base a esto se tiene claro que es lo que se va a enseñar, interviene directamente el autor de contenidos.

Diseño: en esta fase se hará uso de los patrones ya elaborados con anterioridad, se seleccionará el patrón que responda a las necesidades de competencia que se pretende desarrollar

Evaluación: en esta fase se evaluará el OA mediante un método de evaluación, en este caso se elaboró un test donde se consideran los siguientes puntos: pertinencia de contenidos, diseño estético y diseño Instruccional.

Implantación: después de pasar un proceso minucioso de evaluación, el OA será almacenado en un repositorio de OA evaluados, y del mismo será integrado en un Sistema de Gestión de Aprendizaje, el cual puede ser propio o comercial, esto, con la finalidad de interactuar con el mismo en un determinado contexto, para hacer uso y re-uso de éste.

Seguimiento del Conocimiento. En la metodología propuesta está es una fase importante también, ya que aquí se evalúa si el objeto de aprendizaje cumplió con su objetivo.

Manual de buenas prácticas para el desarrollo de objetos de aprendizaje [7]

Trabajo desarrollado en Chile por un grupo de profesionales pertenecientes al proyecto FONDEC. Lo que se presenta en este manual es definir conceptos relacionados con OA sus características, estructura y la forma de desarrollarlos. Se presenta de forma clara el contenido y evaluación de los OA en cuanto a la generación de OA se define un proceso lineal que va desde la definición del Objetivo educativo hasta el empaquetamiento del OA. Además de esto hace referencia a la secuenciación de objetos de aprendizaje orientados a un diseño instructivo.

Según el manual de buenas prácticas el primer paso para generar un objeto, es definir e incorporar el objetivo. Para incorporar el objetivo, se debe llenar un formulario con preguntas acerca de las características básicas del objeto, tales como el título, el editor, el desarrollador de contenido, el desarrollador de multimedia, la clasificación temática del contenido, y la fecha de incorporación entre otros, las que formarán parte del catálogo de objetos presente en la plataforma. Posteriormente, el profesor debe desarrollar el contenido en un editor de texto cualquiera, el cual, una vez finalizado, es incorporado secuenciadamente en la plantilla bajo un ambiente de trabajo. Paralelamente, el profesor debe interactuar con un diseñador de multimedia para definir los recursos multimediales (imágenes, animaciones, videos, narración, gráficos, otros) que incorporará al objeto. Una vez completada la plantilla, debe ser incorporada a la plataforma a través de mecanismos simples adjuntado el archivo Flash. La aplicación y la evaluación deberán desarrollarse directamente en la plataforma, para lo cual ésta incorpora herramientas especiales de edición y elaboración de métodos de evaluación. Una vez incorporados todas las secciones del objeto en la plataforma, automáticamente ésta generará el metadato del objeto y los patrones SCORM, empaquetando así el objeto definitivo.

Reusable Learning Object Strategy: Designing and Developing Learning Objects for Multiple Learning Approaches [3]

Proyecto desarrollado desde 1999 hasta la fecha por CISCO Systems. Este proyecto plantea una estrategia que brinda elementos, pautas y fases puntuales para el desarrollo de objetos de aprendizaje teniendo en cuenta los elementos internos de éstos así como el contexto y los objetivos para los que se desarrollan estos objetos. Para esta estrategia una característica

fundamental que deben poseer los objetos de aprendizaje es su reusabilidad y por tanto se definen los conceptos de RLO (Reusable Learning Object) y RIO (Reusable Information Object), conceptos que brindan una idea de lo que el proyecto pretende. Cisco Systems con su estrategia plantea una metodología que se ajusta para ser utilizada en cualquier rama del conocimiento, siendo el objetivo primordial la capacidad de reutilización que tengan los productos obtenidos mediante la aplicación de la estrategia. Cabe recalcar que la estrategia que propone CISCO Systems, es una guía muy adaptable para cualquier ámbito en el que se deseen utilizar objetos de aprendizaje como forma de compartir y coleccionar conocimiento, no obstante la estrategia debido a su gran adaptabilidad propone tareas y procesos de manera muy general por lo que se deben de diseñar y replantear si se desea aplicar esta estrategia para objetivos más específicos

Metodología de desarrollo de objetos de aprendizaje mediante el uso de la cartografía conceptual y células de desarrollo multidisciplinario y multimedia ¡Error! No se encuentra el rigen de la referencia..

Este trabajo propone una metodología para el desarrollo de OA mediante una estrategia pedagógica con un grado de granularidad basado en conceptos (cartografía conceptual), aplicable a la formación de competencias, con equipos multidisciplinarios de producción en diferentes áreas de aplicación y conocimiento en la Universidad Veracruzana en México, para los programas de licenciatura en artes y diplomado en enseñanza superior en su modalidades a distancia y mixto. Las fases de esta metodología son:

Análisis y selección del Concepto: en esta fase se selecciona el área de conocimiento en la cual se producirán los objetos, los equipos académico y tecnológico seleccionan al experto de contenido que apoyará en la creación de dichos objetos. El experto de contenido capacita al equipo de desarrollo en cada concepto.

Preproducción de los objetos de aprendizaje: la preproducción implica la organización, análisis y planeación del contenido del objeto de aprendizaje, en esta fase se crean las relaciones más importantes dentro del desarrollo. Una vez finalizada, se tienen los recursos teóricos necesarios centrados en la planeación.

Producción de los objetos de aprendizaje: la producción es la etapa que lleva más tiempo, en la cual se producen los medios gráficos, electrónicos y audiovisuales que van a contener el objeto.

Postproducción de los objetos de aprendizaje: la postproducción implica la programación electrónica de los materiales y establecer la conducta y comportamiento de los elementos de cada objeto.

Evaluación de los objetos de aprendizaje: en esta fase los equipos pedagógico y tecnológico evaluarán cada uno de los objetos de aprendizaje bajo la perspectiva de su integración al repositorio y su capacidad de reusabilidad y escalabilidad.

Etiquetación y empaquetamiento de los objetos de aprendizaje: en esta fase los objetos producidos se etiquetan con metadatos del estándar SCORM.

Learning Objects Construction Methodology: LOCoMe [5]

Este proyecto fue realizado en la Universidad Central de Venezuela. LOCoMe nace de la necesidad de crear una metodología robusta de construcción de OA, basada en estándares y la metodología Rational Unified Process (RUP), para proveer los mecanismos necesarios y dotar los

niveles idóneos de calidad sistémica al producto obtenido. Consta de cuatro fases iterativas análisis, diseño conceptual, construcción y evaluación pedagógica. El objetivo fundamental de esta investigación es crear conciencia de la importancia del uso de estándares y de metodologías sólidas para la construcción de OA. A continuación se describen cada una de las fases contempladas dentro de la metodología:

Análisis: durante esta fase temprana del desarrollo del objeto de aprendizaje, se procede a establecer la visión, pertinencia, metáforas y características detalladas del objeto a ser construido. Ello permite a su vez establecer claramente cuáles serán los requerimientos perseguidos con el desarrollo del objeto, tanto a nivel conceptual como funcional.

Diseño conceptual: una vez establecido claramente el propósito y los objetivos perseguidos con la construcción del objeto, se procede a establecer la “forma” mediante la cual va a ser desarrollado el mismo. Esta selección resulta independiente de la plataforma de desarrollo a ser empleada en la siguiente fase. Durante esta fase se diseñan modelos informales de datos y de aplicación, que permiten especificar claramente las características y disposición ideal de cada uno de los sub-objetos que conformarían el objeto final, aparte de la selección de la metodología de construcción adecuada. Es importante señalar, que esta etapa de la metodología es independiente del estándar de construcción de OA seleccionado. El objetivo de esta fase es proveer los mecanismos necesarios para que, dentro del ciclo de vida del desarrollo de dicho objeto, se incluyan los esfuerzos necesarios para adaptar el mecanismo de construcción del OA, a las restricciones y características establecidas por el estándar.

Construcción: esta fase contempla la implementación del objeto de aprendizaje. Dentro de la propuesta de metodología, a diferencia de RUP, esta fase abarca dos subfases: Desarrollo de los recursos y adecuación al estándar de OA seleccionado.

Evaluación Pedagógica: como su nombre lo indica, en esta fase se evalúa que el OA, una vez desarrollado, cumpla con el objetivo inicial. Es importante contemplar esta fase adicional ya que es posible que un objeto esté correctamente desarrollado a nivel conceptual y técnico, y que sin embargo, no cubra los objetivos educativos que de él se esperaban, lo cual haría perder el sentido al desarrollo abordado. El objetivo fundamental de esta fase consiste en determinar si las características del enfoque educativo seleccionado se están cumpliendo satisfactoriamente

2.2.6.7 Metodología para elaborar objetos de aprendizaje e integrarlos a un sistema de gestión de aprendizaje.

Esta metodología se basó en el Modelo de Diseño Instruccional ADDIE, recalcan el retomar un Modelo de Diseño Instruccional que guié en la parte pedagogía, presenta fases como análisis y obtención, diseño, desarrollo, evaluación e implantación. Se especifican los productos finales en cada fase, así como los actores que interviene en cada una de ellas. Esta metodología utiliza plantillas para que sirvan de guía en cada fase. Facilitando la obtención de información. A continuación se presenta una breve descripción de cada fase o paso como lo mencionan los autores.

Paso 1. Análisis. Es importante dejar claro que es lo que se quiere enseñar y a quienes, identifique los datos generales del OA.

Paso 2. Obtención del material. Este paso consiste en proveer el material didáctico necesario, para la construcción del OA, el cual puede ser de diversa índole, como por ejemplo: Impresos (textos): libros, enciclopedias, fotocopias, periódicos, documentos, etc;

Paso 3. Digitalizar el material. Este paso solo procede en caso de que el material didáctico, no este previamente digitalizado. El material puede descargarse de Internet o de las fuentes (cd's)

Paso 4. Arme la estructura del OA. Identificando cada uno de sus componentes. La estructura que se indica a continuación es la mínima necesaria, elementos como simulación o interactividad.

Paso 5. Armado. La estructura general del OA, logrado en la fase anterior, es necesario integrarlo en un archivo html

Paso 6. Empaquetar. Mediante un software generador de paquetes SCORM, se procederá a crear y editar el metadato del OA.

Paso 7. Almacenar el OA en un repositorio temporal. Los repositorios de los OA's son aplicaciones que facilitan el almacenamiento, búsqueda, uso y re-uso de los mismos.

Paso 8. Evaluar el OA. En este paso el OA, será evaluado por un grupo de expertos tomando como referencia una serie de indicadores se propone evaluar un OA, bajo las siguientes categorías: categoría didáctico-curricular, categoría técnica-estética, categoría funcional.

Paso 9. Integrar el OA a un sistema de gestión de aprendizaje (SGA). Al integrar el OA a un SGA, se añadirá a un determinado contexto, donde se podrá realizar una explotación adecuada del mismo.

Modelo Instruccional para el Diseño de Objetos de Aprendizaje: Modelo MIDOA [7]

Proyecto presentado por Arturo Barajas y Jaime Muñoz Arteaga de la Universidad Autónoma de Aguascalientes (UAA) México. En este proyecto se manifiesta la necesidad de procesos estructurados para el desarrollo de software educativo y la estandarización de los mismos para garantizar la calidad de los productos, el modelo desarrollado se fundamenta en procesos con ciclos de vida evolutivos los cuales generan prototipos, además de utilizar varias practicas propias de metodologías ágiles específicamente técnicas de programación extrema. Es importante mencionar que en este proyecto se busca crear un puente de comunicación entre la parte pedagógica con la tecnológica lo cual garantice experiencias de aprendizaje mas enriquecedoras. Sin lugar a dudas es un proyecto que se acerca a procesos formales de generación de OA Sin embargo se considera que existen algunas carencias para involucrar de manera específica a los atributos de calidad propios de los OA. Las fases definidas dentro de este modelo son:

Análisis: consiste en analizar el modelo institucional para adaptar las teorías pedagógicas y las taxonomías para producir, así, las competencias requeridas por la institución para alcanzar el modelo deseado.

Diseño: consiste en elaborar el diseño instruccional de los contenidos y las reglas de producción de los objetos con base en el análisis de competencias para garantizar la ergonomía y usabilidad del objeto, y el aprendizaje y los mecanismos de evaluación.

Desarrollo: consiste en producir los objetos de aprendizaje, con base en el diseño instruccional y las reglas de producción, a partir de los contenidos proporcionados por el autor o autores.

Evaluación: consiste en la evaluación de los contenidos del objeto de aprendizaje de acuerdo a la calidad y cantidad de los contenidos.

Utilización: en esta fase el autor y el usuario son quienes usan los objetos de aprendizaje desarrollados y emitirán una “evaluación” en términos de usabilidad, nivel de aprendizaje, calidad de contenidos, entre otros indicadores, de los objetos de aprendizaje. Para el desarrollo de las fases se aplica un proceso en espiral con un mínimo de tres ciclos.

REFERENCIAS

- [1] F. Álvarez, A. Barajas y J Muñoz. (2007). “Modelo Instruccional para el Diseño de Objetos de Aprendizaje: MIDOA”. *Virtual Educa Brasil*, Junio, 2007. [En línea]. Disponible: <http://ihm.ccadet.unam.mx/virtualeduca2007/pdf/164-ABS.pdf>. [Accedido Jun. 10, 2009].
- [2] M. López y J. Medina, “LOCOME: Metodología de Construcción de Objetos de Aprendizaje,” *contenidos digitales Universidad Central de Venezuela*, Julio 2008, pp. 1-4. [En línea] Disponible:http://espacio.uned.es/fez/eserv.php?pid=bibliuned:20062&dsID=LOCOME_Metodolog__a_De_Construcci__n.pdf. [Accedido Abr. 20, 2008].
- [3] F. Álvarez, J. Muñoz y B. Osorio, “Metodología para el desarrollo de objetos de aprendizaje usando patrones,” *Conferencia Latinoamericana de Objetos de Aprendizaje – LACLO*, pp. 2-4, Octubre 2007. [En línea]. Disponible: <http://ava43.files.wordpress.com/2008/07/metodlogia-de-realizar-objetos-de-aprendizaje.pdf>. [Accedido Feb. 25 2009].
- [4] J. Jiménez, R. Parra y D. Reyna, “Metodología de Desarrollo de Objetos de Aprendizaje Mediante el uso de la Cartografía Conceptual y Células de Desarrollo Multidisciplinario y Multimedia,” en *Conferencia Latinoamericana de Objetos de Aprendizaje*, Octubre. 25, 2007, pp. 1-3. [En línea] Disponible: http://www.laclo.espol.edu.ec/index.php?option=com_docman&task=doc_view&gid=20&Itemid=3. [Accedido Nov. 23, 2008].
- [5] José Medina, María López. LOCOME: Metodología De Construcción De Objetos De Aprendizaje Universidad Central de Venezuela [Accedido Jun. 23, 2009].

D. ANEXO TÉCNICAS DE USABILIDAD SELECCIONADAS

En este anexo se brinda una breve descripción de las técnicas IPO seleccionadas, también se expone lo que significa cada categoría a la que pertenecen dichas técnicas. Esto con el fin de continuar cerrando la brecha entre estas las Disciplinas IPO e Ingeniería de Software.

Tabla Técnicas seleccionadas para ser aplicadas en la Metodología Propuesta: MGOAS

	ACTIVIDAD IS General	Actividad IS Especifica	TECNICA IPO
ANALISIS	Educción y Análisis de Requisitos	Análisis de Requisitos	Card Sorting
			Análisis competitivo
			Investigación Contextual
		Análisis de Usuarios	Personas
			Perfiles de Usuario
		Análisis de Tareas	Casos de Uso Esenciales
		Desarrollo del Concepto del Producto	Escenarios y storyboards
			Tormenta de Ideas Visual
		Prototipado	Prototipos de papel
		Especificación de Requisitos	Especificaciones de usabilidad
Validación de Requisitos	Inspecciones colaborativas		
DISEÑO	Diseño de la interacción		Árboles de Menús
			Diagrama de Transición de Estados de la Interfaz
			Mapa de navegación
			Modelado del Contenido de la interfaz

EVALUACION	Evaluación por expertos	Inspecciones
	Test de Usabilidad	Pensar en Voz Alta
		Test de Usabilidad en Laboratorio
		Información post Test
	Estudios de Seguimiento de Sistemas Instalados	Retroalimentación del Usuario
Cuestionarios, Entrevistas y Encuestas		

1. Análisis o Ingeniería de Requisitos

A este grupo están ubicadas una cantidad significativa de técnicas IPO, dentro de la fase de análisis de la IS se considera que hay relación con Usabilidad por medio de los grupos denominados Educcion y análisis de requisitos, la especificación de requisitos y la validación de requisitos.

1.1 Educcion y análisis de requisitos

En la educcion de requisitos se identifican las fuentes de requisitos y se capturan los mismos. El análisis y negociación de requisitos se ocupa de detectar y resolver posibles problemas entre requisitos, delimitar el sistema y cómo debe interactuar con su entorno, y realizar el proceso de transformar los requisitos del sistema a requisitos software.

En IS se considera una tarea importante para el éxito de un proyecto, la técnicas IPO pueden aportar a la fase de educcion la obtención de información y los medios para extraer conocimientos suplementarios a los que se emplean tradicionalmente la Ingeniería de requisitos. Las técnicas de este grupo pueden ampliar el marco de necesidades reales de los usuarios y de las tareas que realizan. En lo concerniente a la negociación de requisitos las técnicas con carácter participativo están principalmente indicadas para enriquecer las labores de negociación con el mencionado componente participativo.

Las técnicas pertenecientes a esta categoría son agrupadas en subcategorías entre ellas tenemos Análisis de Requisitos, Análisis de usuarios, Análisis de tareas, Desarrollo del Concepto del producto, Prototipado

1.1.1 Análisis de requisitos.

En esta etapa se busca identificar requisitos y las relaciones entre ellos las técnicas asociadas a esta fase son Card Sorting, Análisis competitivo e Investigación Contextual.

1.1.2 Card Sorting

Esta técnica permite comprender la representación de información que manejan los usuarios. Consiste en pedir a los usuarios que agrupen una serie de conceptos del dominio, para obtener como resultado una agrupación representativa del modelo del dominio que tiene el usuario en la cabeza. Cada concepto se escribe en una tarjeta, y se pide al usuario que organice las tarjetas en pilas.

1.1.3 Análisis Competitivo

Podemos analizar heurísticamente productos existentes, y podemos realizar test de usabilidad con usuarios. Un producto de la competencia, como ya está implementado totalmente puede ser probado fácilmente. Si vamos a analizar más de un producto de la competencia, podemos realizar un análisis comparativo de las diferentes soluciones para conseguir los objetivos del usuario.

1.1.4 Investigación Contextual

Es una forma de obtención de información que se puede utilizar en la evaluación. Los usuarios y los investigadores participan para identificar y para entender problemas de la utilidad dentro del entorno de trabajo normal del usuario. Con esta técnica se consigue una comprensión mas profunda de las necesidades del usuario que con las tradicionales entrevistas.

1.1.2 Análisis de Usuarios

Esta etapa Se ocupa de identificar cuáles son los usuarios previstos del sistema, y para ellos es preciso obtener toda información sobre sus conocimientos, necesidades y características que sea importante en su interacción con el sistema. Las técnicas consideradas acopladas a esta actividad son: Personas, Perfiles de usuario.

1.1.2.1 Personas

Esta técnica ayuda a sintetizar todos los datos de que se dispongan sobre los usuarios previstos del sistema, en unos usuarios arquetípicos que puedan usarse para alcanzar consenso en el equipo de desarrollo y para centrar las discusiones de diseño. Ayuda también a determinar qué es lo que el producto debe hacer, relacionado con las necesidades a satisfacer, por lo que puede contribuir a todo el proceso de análisis y negociación de requisitos. Al proveer un lenguaje común para referirse a los usuarios concretos del sistema (concreto, frente al término genérico y a menudo equívoco "usuarios del sistema"), ayuda a alcanzar consenso en el equipo de desarrollo. En concreto, evita un problema típico entre un cierto número de desarrolladores. El identificar las capacidades y querencias de los futuros usuarios del sistema con las propias (del desarrollador), conduciendo este tipo de actitud a la producción de sistemas software que únicamente usuarios con marcado perfil tecnológico pueden usar.

1.1.2.2 Perfiles de Usuario

Los perfiles de usuarios describen a los usuarios previstos del sistema, según características psicológicas, de Conocimiento, Experiencia, Características físicas Para cada usuario se incluye una descripción general, una descripción de las características de los usuarios, y un apartado sobre los requisitos de usabilidad para ese tipo de usuario. Esta técnica es cercana a la Ingeniería de Software ya que hace uso de Modelamiento. Además esta actividad es una de las bases del enfoque centrado en el usuario.

1.1.3 Analisis de Tareas

El objetivo del análisis de tareas es obtener descripciones de lo que las personas hacen para realizar las tareas de los cuales se ocupan. Se encuentra una sola técnica para esta categoría: Casos de uso esenciales

1.1.3.1 Casos de Uso Esenciales

En un nivel más alto de abstracción, los casos se definen en términos de intenciones de usuario y responsabilidades del sistema, sin tener en cuenta la tecnología usada y la implementación. Técnica estrechamente relacionada a los casos de Uso de la IS Solo que a un nivel más abstracto su utilización resulta muy valiosa como parte de las actividades de análisis de los desarrollos orientados a Objetos.

1.1.4 Desarrollo del Concepto del Producto

El establecimiento de una visión general del producto cumple un papel muy importante en la negociación de requisitos ya que todas las partes involucradas deben compartir esta visión general del producto a desarrollar. En esta actividad se consideran las normas generales que regirán el funcionamiento del sistema, sus espacios de interacción principales y cómo se trabajará con el mismo. Desarrollar un concepto del producto explícito ayuda a que éste se transmita al usuario para permitirle desarrollar un modelo mental del sistema adecuado. Las técnicas que brinda cumplimiento a estos objetivos son Escenarios y Storyboards, al igual que la técnica Tormenta de Ideas Visual

1.1.4.1 Escenarios y storyboards

Un escenario es una historia de ficción, personalizada con personajes, eventos, productos y entornos, normalmente tipo cómic, que capturan una posible acción significativa. Los storyboards son secuencias de instantáneas que se centran en las principales acciones en una posible situación. Esta técnica tiene un aporte significativo en Usabilidad y el esfuerzo es poco. Esta actividad resulta útil cuando el cliente tiene dificultades para expresar lo que el sistema requiere.

1.1.4.2 Tormenta de Ideas Visual

Es una técnica de bocetos empleada para explorar diseños alternativos. Después de hacer bocetos iniciales, las mejores ideas se pueden desarrollar más en profundidad construyendo representaciones del diseño en cartulina, que pueden ser evaluados por los usuarios. Se puede continuar desarrollando escenarios, software o prototipos en video. Esta técnica pretende obtener la idea del producto a desarrollar

1.1.5 Prototipado

El estándar ISO 13407 define un prototipo como "una representación de todo o parte de un producto o sistema que, aunque limitado de algún modo, puede utilizarse con fines de evaluación

El prototipado está íntimamente ligado al desarrollo iterativo. Para que los prototipos sean efectivos, deberían tener un coste mínimo en términos de recursos y tiempo. La diferencia entre el prototipado propuesto por la IPO y el empleado habitualmente en los desarrollos de la IS se encuentra, de nuevo, en el enfoque. Los prototipos son útiles desde un punto de vista de la usabilidad cuando reflejan principalmente la interacción usuario-sistema, de forma que pueden transmitir cómo va a funcionar el sistema desde el punto de vista del usuario [Preece, 94].

1.1.5.1 Prototipos de papel

El prototipado está estrechamente ligado al desarrollo iterativo. Los prototipos pueden usarse para probar ideas de diseño con los usuarios para obtener su retroalimentación. Con este fin, los prototipos menos costosos (de papel), son suficientes. Los prototipos de papel son útiles para obtener opiniones del usuario mejor que los prototipos que se asemejan al sistema final. Como parte de las reuniones de los requisitos se pueden usar bosquejos a mano o en Computador de la apariencia de la IU. Esto le trasmite la idea al cliente de cual es el sistema a Construir. A pesar de ser prototipos de papel también logran transmitir la interacción del sistema.

1.2 Especificación de Requisitos

Esta actividad se centra en la elaboración de un documento en el cuál se manifiesta los requisitos del sistema, las prioridades del mismo, atributos identificados que debe cumplir el producto. La técnica de usabilidad orientada a estos objetivos es la de: Especificaciones de la usabilidad

1.2.1 Especificaciones de usabilidad

Esta técnica se enmarca dentro de la Especificación de Requisitos la cual tiene que ver con la elaboración de un documento donde se reflejan los requisitos que el sistema debe cumplir, y en particular tiene que ver con la estructura, calidad y verificabilidad de dicho documento, en cuanto a esta técnica Son metas cuantitativas de usabilidad, que se usan como guía para saber cuando una interfaz es suficientemente buena. Las especificaciones pueden ser basadas en medidas objetivas o subjetivas.

Las medidas objetivas se asocian normalmente a tareas de benchmark, mientras que las subjetivas se asocian normalmente con cuestionarios de usuario. Las especificaciones de usabilidad pueden considerarse un subconjunto de los requisitos no funcionales de un sistema.

1.3 Validación de Requisitos

Dentro de esta actividad se busca indagar en los requisitos del sistema y definir si los requisitos esta bien encaminados hacia lo que el usuario espera. Para este caso se tiene la técnica Inspecciones Colaborativas

1.3.1 Inspecciones colaborativas

Es una técnica que brindara soporte a la especificación de requisitos, tienen como finalidad examinar los requisitos para asegurarse de que definen el sistema adecuado. Las inspecciones colaborativas son una variante de las inspecciones de usabilidad, en la que participan todas las partes involucradas. Se trata de un procedimiento estructurado para promover la participación del usuario en el proceso de inspección. Los miembros del equipo de desarrollo que participan en el proceso de inspección tienen prohibido defender, explicar o racionalizar ningún aspecto de su diseño o las decisiones que le han llevado a él. Además, los desarrolladores deberían evitar hacer

promesas implícitas a los usuarios. Las aportaciones de los usuarios deberían tener un peso especial en el proceso de inspección, aunque sin permitir que dicten decisiones de diseño de la interacción.

2. Diseño

Las actividades de diseño software son aquellas en las que los requisitos software son analizados para producir una descripción de la estructura del software que servirá como base para su construcción. Dentro de esta fase se obtendrán modelos mas enfocados y detallados sobre el producto. La igual que una arquitectura inicial de Software. Una categoría que se adecua a estas necesidades es el diseño de la interacción.

2.1 Diseño de la interacción

El Diseño de la Interacción se delega el análisis de los entornos, la interacción de los mismos y su comportamiento. Al incluir el comportamiento conlleva a que se debe regular la interacción entre el usuario y el sistema, y esto se vera reflejado en cambios sobre el diseño que afectan a la estructura interna del sistema. También se incluye el diseño de los componentes visuales que forman la Interfaz Gráfica de Usuario, respecto a la interfaz se trata de una actividad relativamente independiente del resto de actividades de diseño, por lo que corresponde a una categoría propia con actividades de este tipo.

2.1.1 Árboles de Menús

Técnica que se considera similar a las realizadas en Ingeniería de Software ya que requiere de un modelado y organización sistémica. Presenta las relaciones entre los distintos elementos de una jerarquía.

Los árboles del menú son una herramienta de especificación muy útil, puesto que demuestran a usuarios y a otros implicados la funcionalidad completa y detallada del sistema. Para la creación de los árboles de Menú se debe contar con la participación de los Desarrolladores y usuarios.

2.1.2 Diagrama de Transición de Estados de la Interfaz

Los diagramas de Transición de Estados son frecuentes en la IS, por lo que esta técnica esta ligada a esta actividad de la Ingeniería del Software y es mayormente aplicable a productos con IU Graficas y sirve de apoya a la creación de Manuales de usuario. En este tipo de diagramas los nodos representan estados de la interfaz o pantallas, y los arcos representan transiciones de estado basadas en las entradas.

2.1.4 Mapa de Navegación

Técnica que permite la representación de las posibilidades de navegación entre distintos contextos de interacción, por lo que su uso esta indicado varios espacios de interacción y se lograrían apreciar desviaciones de contextos que se podrían analizar y mejorar.

2.1.4 Modelado del Contenido de la interfaz

Permite realizar las tareas de diseño de la interacción con una base grafica que favorece la discusión de alternativas. Esta indicada para IUs basadas en un sistema de ventanas con distintos espacios de interacción. El principal objetivo de esta técnica es crear un puente entre los diseñadores de Interacción de alto nivel con los desarrolladores de la IU. Esta técnica Es una representación abstracta de los contenidos de los distintos espacios de interacción de un sistema.

El contenido se puede modelar por medio de papel (una hoja por cada espacio de interacción) y Post-Its que representan las herramientas y materiales que se van a ofrecer al usuario.

3. Evaluación

En la evaluación se busca principalmente detectar fallas de cualquier tipo en el producto. Ya sea cuando este se ha terminado o cuando esta en proceso de desarrollo

3.1 Evaluación por expertos

3.1.1 Inspecciones

Son similares a las inspecciones de Software que se realizan habitualmente en desarrollos de Software, únicamente cambia el objeto de la inspección que suele ser la IU o un prototipo de la misma. En el desarrollo de software el objetivo de una inspección es encontrar defectos. Las inspecciones de usabilidad tienen por objetivo descubrir defectos de usabilidad.

Hay dos tipos de inspecciones de usabilidad:

- Inspecciones de conformidad (con una guía de estilo, o un estándar): En las inspecciones de conformidad, los participantes inspeccionan el diseño de la interacción del sistema para comprobar que si cumple lo estipulado en determinados estándares o guías de estilo. Todos los participantes deben estar familiarizados con los estándares o guías sobre los que se evalúa.
- Inspecciones de consistencia: En las inspecciones de consistencia, el objetivo consiste en identificar inconsistencias entre contextos de interacción y sus contenidos. El evaluador intenta encontrar inconsistencias en terminología, color, disposición, formatos de entrada y salida de datos, etc. Cuando el producto pertenece a una familia de productos, se juntan representantes de los distintos equipos de diseñadores para inspeccionar la consistencia entre los productos de la familia.

3.2 Test de usabilidad

En los procesos centrados en el usuario estos tipos de técnica son especialmente apropiados, ya que el sistema se prueba con usuarios representativos y se mide la usabilidad con objetividad cuando el usuario realiza tareas habituales con la aplicación, ofreciendo criterios de decisión sobre la calidad del sistema los cuales permiten identificar falencias en Usabilidad. A continuación se presenta algunas técnicas relacionados a Test de Usabilidad.

3.2.1 Pensar en Voz Alta

Técnica que aporta al Test de Usabilidad, se basa en el razonamiento interno del usuario, el cual lo hablara en voz alta para identificar problemas de Usabilidad, esto implica el tener a un participante usando el sistema mientras que piensa en voz alta. Su efectividad está en datos cualitativos y no en medidas de funcionamiento. La idea es conseguir la opinión del usuario mientras usa el sistema para evitar posteriores racionalizaciones.

3.2.2 Test de Usabilidad en Laboratorio

Se pretende realizar test de usabilidad en instalaciones especialmente concebidas para ello, un laboratorio de usabilidad debe proporcionar un ambiente controlado y consistente en el cual

realizar las pruebas. Se debe tener en cuenta que puede resultar costoso crear un entorno para este tipo de pruebas, esto también depende del tipo de proyecto que se llevara a cabo. En este tipo de test, en lugar de llevarle al usuario la interfaz, se trae al usuario a la interfaz. Es decir, los participantes son llevados a unas instalaciones tipo laboratorio de usabilidad donde llevan a cabo las tareas de referencia y todo lo asociado a un test de usabilidad.

3.2.3 Información post Test

Para desligar la experiencia del uso del sistema y la atención que dicho uso requiere del usuario para verbalización de sus acciones esta técnica posibilita que el usuario brinde su Test de Usabilidad una vez su sesión de prueba del sistema ha terminado. La mayor parte de planes de test incluyen una entrevista post-test con cada individuo, Se puede planificar una recogida de información del participante una vez realizado el test de usabilidad.

En esta tipo de entrevistas normalmente se agradece la participación y se le recuerda que su aportación es muy importante y sirve para mejorar el software. Se trata de compaginar un test de usabilidad con medida del rendimiento del usuario (por tanto, con medición del tiempo que emplea en cada tarea), con recogida de información sobre el razonamiento seguido por el usuario. Esta técnica es alternativa a la de Pensar en Voz Alta cuando ésta no se puede aplicar por la medición del rendimiento.

3.3 Estudios de Seguimiento de Sistemas Instalados

Este tipo de técnicas de evaluación se aplica sobre sistemas o prototipos software ya instalados en el entorno previsto de uso. Permite la obtención de información fiable sobre los posibles inconvenientes y dificultades que presenta el producto software.

3.3.1 Retroalimentación Del Usuario

En esta técnica el usuario toma un papel mas activo y es el que toma la iniciativa para acudir a la organización de Desarrollo de software para informar el asunto. La retroalimentación se puede obtener dando a los usuarios acceso a:

- Direcciones especiales de correo electrónico
- Grupos de noticias
- Foros.

Los usuarios pueden de este modo enviar sus quejas o peticiones de cambio o mejora.

3.3.2 Cuestionarios, Entrevistas y Encuestas

Esta técnica proporciona la información relativa a la satisfacción subjetiva del usuario. En las entrevistas flexibles se puede obtener deficiencias de usabilidad del sistema. Son métodos indirectos para estudios de la interfaz de usuario, porque proveen al equipo de desarrollo de la opinión de los usuarios, y no información directa de la interfaz de usuario. Los cuestionarios se pueden enviar por correo ordinario, correo electrónico o con el propio software, las entrevistas se pueden conducir personalmente o por teléfono.

Los cuestionarios sirven para obtener información de un número más amplio de usuarios se pueden elaborar, distribuir y analizar cuestionarios estructurados que incluyen preguntas sobre la usabilidad del producto. Las entrevistas son útiles para obtener las reacciones subjetivas de los usuarios.

E. ANEXO ESTÁNDARES REFERENTES LA DISCIPLINA IPO, INGENIERÍA DE SOFTWARE Y DISEÑO CENTRADO EN EL USUARIO

En esta sección se presenta algunos de los estándares destacados en áreas como la IPO, el DCU y estandartes que integran directa o indirectamente la ingeniería de Software y la Usabilidad. Se consideran algunos enfoques de gran interés y aporte en aras de cumplir objetivos de usabilidad sin descuidar técnicas, Atributos de calidad y procedimientos propios de la ingeniería de Software.

Ciclo de Vida en estrella

Los autores Hix y Hatson plantean un enfoque en estrella el cual es centrado en el usuario, establece actividades IPO que considera principales, no presenta un orden específico y esta es una razón que hace de este enfoque un proceso altamente iterativo. A continuación se presenta en la figura 1 las actividades de este desarrollo. Se observa que las de color gris están relacionadas al desarrollo software mientras que las de color blanco están dirigidas a la interacción y Usabilidad.

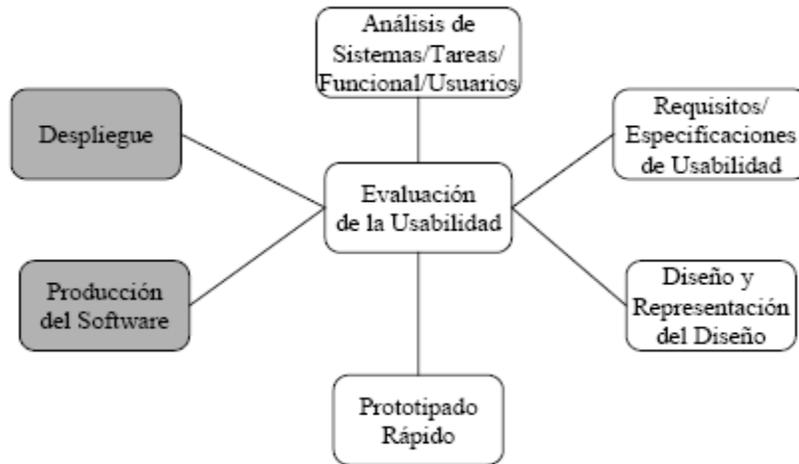


Figura 1. Ciclo de vida en estrella

Se analiza que el ciclo de vida en estrella trata bien los temas de evaluación de la usabilidad a través de evaluación con usuarios reales existe una especificación formal de la IU, y al no detallar suficientemente las actividades referentes al de sistemas, el proceso queda disminuido al querer integrar con otras actividades del desarrollo de Software

Human-centred design processes for interactive systems [1]

El estándar ISO 13407 constituye un marco que sirve de guía para conseguir el Desarrollo de sistemas interactivos usables incorporando el DCU durante el ciclo de vida del desarrollo. El estándar describe las siguientes cuatro actividades que se necesitan desde un principio:

- a) Entender y especificar el contexto de uso.
- b) Especificar los requisitos de los usuarios y organizativos.
- c) Producción de soluciones de diseño.
- d) Evaluar los diseños confrontándolos con los requisitos.

En la Figura 2 se presentan las actividades del proceso de Diseño Centrado en el Usuario según el estándar ISO 13407.

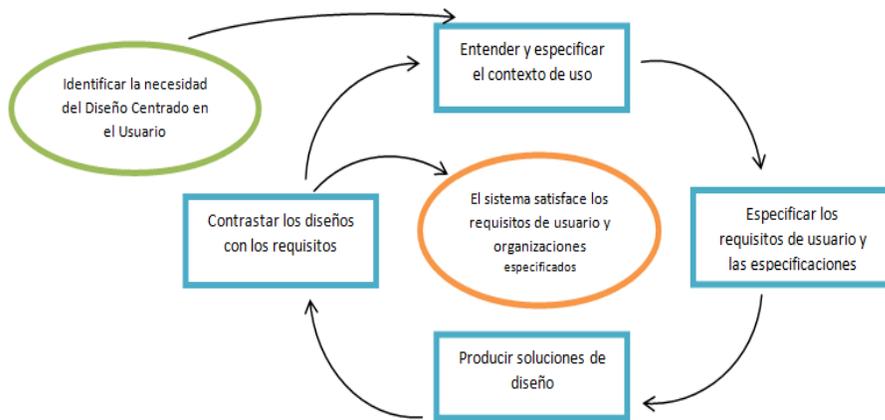


Figura 2. Interdependencia de las actividades de diseño centrado en el humano. Estándar 13407: 1999

- Entender y especificar el contexto de uso. En esta actividad se obtienen las características principales de usuarios, labores y el entorno donde se desarrolla así mismo que su impacto.
- Definir los requisitos de usuario y las especificaciones. en esta fase se debe comprender substancialmente los requerimientos del usuario basados en un DCU se considera que se definen rangos de usuarios, objetivos, prioridades, criterios de diseño.
- Producir soluciones de diseño. Actividad referente a brindar diseños que cumplan los objetivos propuestos por el DCU. Es muy útil diseños basados en simulaciones, modelos, prototipos simples.

- Contrastar los diseños con los requisitos. El objetivo de esta actividad es de este proceso proporcionar retroalimentación de diseño, evaluar si los objetivos han sido alcanzados, validación de campo y monitoreo de largo plazo.

El proceso implica iterar hasta satisfacer los objetivos marcados, para lo que ISO 13407 describe los principios básicos sin estipular métodos específicos. La secuencia de realización o seguimiento de las actividades y el nivel de esfuerzo y detalle apropiado a cada proyecto varía dependiendo del entorno de diseño y el estado del proceso del mismo.

Un inconveniente identificado en este estándar es el poco detalle para guiar a una organización de Software a implementar este enfoque, además no se especifican actividades para todo el proceso de desarrollo.

Estándar ISO/IEC 12207 [2]

Este estándar ofrece un marco de referencia común para los procesos del ciclo de vida del software, con una terminología bien definida a la que puede hacer referencia la industria del software. Agrupa las actividades que se pueden llevar a cabo durante el ciclo de vida del software en cinco procesos principales, ocho procesos de apoyo y cuatro procesos organizativos. Cada proceso del ciclo de vida está dividido en un conjunto de actividades; y cada actividad se subdivide a su vez en un conjunto de tareas.

Una organización dependiendo de sus necesidades, puede seleccionar un subconjunto apropiado para satisfacer dichas necesidades, así pues los procesos pueden ser adaptados a una organización, proyecto o aplicación concreta. El estándar también incluye un proceso que se puede emplear para definir, controlar y mejorar los procesos del ciclo de vida del software.

Diseño centrado en el uso [3]

Este enfoque se suele confundir con el Diseño centrado en el usuario, sin embargo su diferencia obvia es que el Diseño Centrado en el usuario tiene como enfoque al Usuario y el Diseño centrado en el Uso su enfoque radica en el uso que una persona le da al sistema. El Diseño centrado en el Uso no tiene una alta implicación de usuarios, más bien se preocupa por mejorar las herramientas para el cumplimiento de las tareas. Este método se desarrolla en cinco grandes etapas.

- Guías Pragmáticas de diseño
- Un proceso de diseño dirigido por modelos, modelos interrelacionados que permiten especificaciones en varios niveles de abstracción

- Serie de actividades de desarrollo Organizadas
- Mejora Iterativa
- Medidas de Calidad

Analizando las fases y lo que sustentan sus autores: Constantine y LockWood, es una perspectiva del campo de la IPO que tiene sus orígenes y se acerca más al desarrollo de software. Un aporte importante es establecer como pieza clave para la integración entre usabilidad e Ingeniería de software es la utilización de Casos de Uso esenciales, además de estar destinado a todo tipo de desarrolladores los cuales no necesariamente deben ser expertos en Usabilidad. Sin embargo se considera un método difícil de integrar ya que no indica claramente los puntos de integración y se dan recomendaciones generales.

Modelo ISO /TR 8529 [4]

Este estándar es una extensión del modelo ISO 13407, se considera un informe técnico y define que procesos una empresa debe llevar a cabo para lograr metas técnicas que se han propuesto. En la figura 3 se presenta el modelo y sus 7 fases que reflejan iteración e interrelación

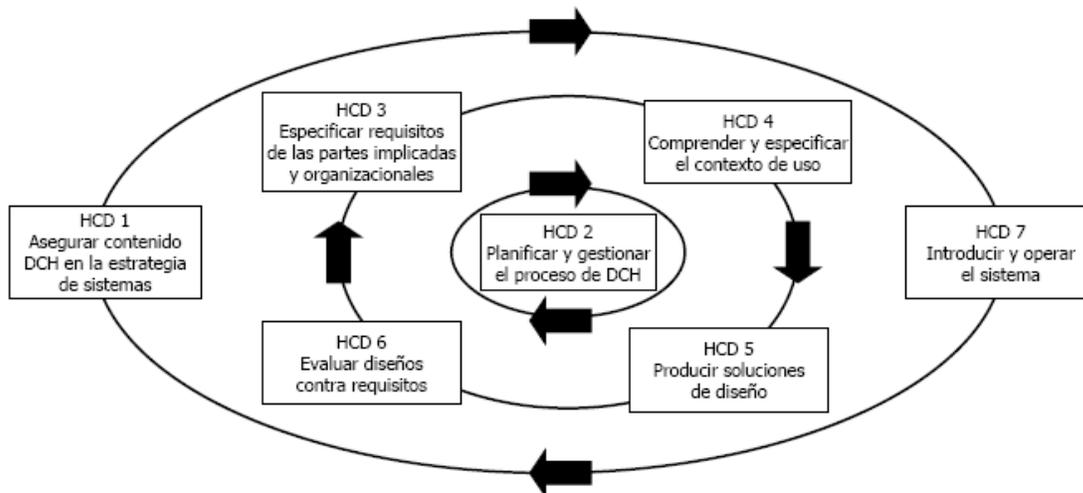


Figura 3. Fases del Modelo ISO 8529

Este proceso indica un grado alto de iteración una razón es que esta siguiendo un enfoque DCU que se deriva del estándar ISO13407. Desafortunadamente el modelo no incluye documentación precisa sobre cómo integrarse con un proceso de desarrollo.

MUSE (Method for usability Engineering)

Este método se centra principalmente en el Diseño, referente a la fase de especificación del diseño. MUSE se divide en tres fases.

- Fase de educación de información y análisis: se ocupa de capturar requisitos y análisis de tareas.
- Fase de síntesis del diseño: referente a un diseño conceptual del sistema objetivo
- Fase de especificación del diseño: centrado en diseño de la IU y definiciones funcionales

Se trata de una propuesta interesante basada en integración de técnicas IPO a partir de un desarrollo estructurado y se ofrecen unas series de técnicas para cada actividad sin embargo no existen indicadores de por que debe aplicarse esa técnica, otra consideración que se hace a este método es la no inclusión de técnicas o fases propias de evaluación.

LUCID (Logical user Centered Interaction Design)

Esta propuesta está desarrollada comercialmente por lo que no está disponible al público. Se trata de un enfoque propietario por lo tanto no existe mucha documentación al respecto. En lo investigado sobre esta metodología se encuentra que utiliza seis fases que se presentan en la figura 4 Actividades de LUCID, y buscan proporcionar un entorno para llevar a cabo actividades de usabilidad, diseño de IU y coordinarlas con prácticas del proceso general de desarrollo.



Figura 4. Actividades de LUCID

A pesar de que el autor presenta LUCID como un marco para integrar la usabilidad en el proceso de desarrollo, no se explica en que se basa para tal afirmación, las etapas de LUCID son fases propias de esta propuesta y no se justifica su relación con etapas de típicas de desarrollo IS, además no se hace referencia como se integrarían las actividades de LUCID con las tradicionales. [1].

Estándar ISO/IEC 9126-1:2001 [5]

Este estándar hace parte del estándar internacional ISO/IEC 9126 (Software engineering - Product quality), pertenece a la parte 1: Calidad del modelo, establece una diferencia entre usabilidad y calidad de uso. En este documento usabilidad se refiere a:

“La capacidad del producto de software de ser comprendido, aprendido, usado y ser atractivo al usuario cuando es utilizado en condiciones específicas.”

La usabilidad forma parte de una serie de componentes internas y externas que dan lugar a que un producto tenga calidad. Las características que en combinación determinan una calidad de uso son: funcionalidad, confiabilidad, eficiencia, capacidad de mantenimiento, portabilidad y usabilidad. Este estándar entrega un marco general para la evaluación de la calidad del software, se destaca de este estándar que es aplicable a cualquier tipo de software.

Estándar ISO/IEC 9241-11 [6]

Este estándar hace referencia a las guías en usabilidad y es la parte 11 del estándar 9241: Requisitos ergonómicos para el trabajo de oficina con terminales de pantalla de visualización. Este estándar establece la definición de usabilidad, relacionada con la ergonomía, la cual se presenta a continuación:

Usabilidad: la medida con la cual un producto puede ser usado por usuarios específicos para lograr los objetivos con efectividad, eficiencia y satisfacción en un contexto de uso.

El estándar ISO 9241-11 expone cómo identificar la información que es necesaria en el momento de especificar o evaluar la usabilidad en términos de medidas de desempeño y la satisfacción del usuario. Brinda orientación sobre la manera de describir el contexto de utilización del producto y las medidas de usabilidad de forma explícita. Incluye una explicación de cómo la usabilidad de un producto puede ser especificada y evaluada como parte de un sistema de calidad. También explica cómo las medidas de rendimiento y la satisfacción del usuario se pueden utilizar para medir cómo el grado de cualquier componente de un sistema de trabajo afecta a la calidad del mismo [23].

En relación a las interfaces multimedia, el estándar ISO 9241-11 [22] expande la definición convencional de la usabilidad vista en cinco componentes:

- La usabilidad operacional: es el sentido convencional de usabilidad que concierne al diseño de rasgos gráficos de la interfaz de usuario como menús, iconos, metáforas y la navegación hipermedia.
- La entrega de la información: constituye un interés principal para cualquier uso intensivo de la información y levanta cuestiones de selección de medios de comunicación, integración de los mismos y el diseño para la atención.

- Estudio y aprendizaje: el entrenamiento y la educación son dos mercados importantes para las tecnologías multimedia, de ahí que el aprendizaje de los productos junto con sus contenidos son atributos de calidad claves.
- Utilidad: en algunas aplicaciones este factor hace referencia a la funcionalidad que apoya las tareas del usuario, en otros la entrega de la información y el estudio, representarán el valor percibido por el usuario.
- Apariencia: sin duda alguna la capacidad de atraer, de seducir o incluso de fascinación de las interfaces multimedia es un aspecto clave de las mismas, que deben ser capaces no sólo de atraer, sino también de motivar a los usuarios y ser fáciles de utilizar y de aprender.

Referencias

[1] Human-centred design processes for interactive systems

http://www.iso.org/iso/catalogue_detail.htm?csnumber=21197

[2] Procesos de ciclo de vida de software. <http://www.bvindicopi.gob.pe/normas/isoiec12207.pdf>

[3] Los elementos de la experiencia de usuario

http://www.jig.net/elements/translations/elements_es.pdf

[4]

http://books.google.com.co/books?id=LoR_qZGX8lgC&pg=PA862&lpg=PA862&dq=ISO+/TR+8529&source=bl&ots=PcNy-cUiJK&sig=zrWxgSUd7NC-bqr94hyXwy2-R9A&hl=es&ei=1SqbTPbzJ8OqIAeg0vXUCg&sa=X&oi=book_result&ct=result&resnum=1&ved=0CBQQ6AEwAA#v=onepage&q=ISO%20%2FTR%208529&f=false

[5] ftp://www.puc-campinas.edu.br/pub/professores/ceatec/tobar/tcc/info_isoiec9126-1.pdf

[6] http://www.usabilitynet.org/tools/r_international.htm#20282

F. ANEXO CASO DE ESTUDIO

DOCUMENTO DE ANÁLISIS DE COMPETENCIAS

Documento de análisis de competencias	
Título del objeto de aprendizaje: Basic level of English language	
Fecha: 2010/02/20	
Responsable: William Jurado	
Objetivos pedagógicos	
<p>-fortalecer las competencias de Writing y Reading, en relación a los temas que se tratan en la unidad uno del libro American Chanel.</p> <p>- aprender a identificar y describir en el idioma ingles: colores, números, países, nacionalidades, horarios, información personal y laboral, además tener la capacidad de describir actividades diarias.</p> <p>-capacidad de lectura y escritura de las siguientes aspectos.</p> <ul style="list-style-type: none"> • Possessive Adjectives • The verb BE • Present simple <p>Prepositions of time</p>	
Estrategias de enseñanza-aprendizaje	
<ul style="list-style-type: none"> - Aprendizaje individual - Aprendizaje interactivo - Aprendizaje por descubrimiento <p>Aprendizaje significativo</p>	
Técnicas de enseñanza - aprendizaje	
<ul style="list-style-type: none"> - Análisis interpretación de lectura - Comprensión y resolución de ejercicios - Ilustraciones - Evaluaciones 	
Información MGOAS	
Disciplina: Modelado de negocio	Numero iteración: 1
Observaciones:	

GLOSARIO DEL NEGOCIO

Glosario del Negocio	
Título del objeto de aprendizaje: Basic level of English language	
Fecha: 2010/02/28	
Responsable(s): Sandro Jaramillo, Usuarios	
Términos comunes	
-Activity Writing - Activity Reading -Grammar -Student -vocabulary -Pronunciation .Usuario -Fase -Unidad -tipo de aprendizaje -learn english	
Información MGOAS	
Disciplina: Modelado de negocio	Numero iteración: 1
Observaciones:	

VISON DEL NEGOCIO

Vision del Negocio	
Titulo del objeto de aprendizaje: Basic level of English language	
Fecha: 2010/01/20	
Responsable(s): Sandro Jaramillo, Ricardo chicangana	
Metas y objetivos de la organizacion	
Trabajar de forma organizada y constructiva en la elaboración de material educativo que apoye el aprendizaje del primer nivel de ingles.	
Fomentar en los estudiantes la motivación por realizar actividades del PFI a través de la Web.	
Recursos utilizados	
-Docentes del PFI	
-Instalaciones y oficinas PFI	
-material de papeleria, impresoras, scanners,	
-sala 4, instalaciones facultad ingeniería electrónica y telecomunicaciones	
Información MGOAS	
Disciplina: Modelado de negocio	Numero iteración: 1
Observaciones:	

REGLAS DEL NEGOCIO

Mantener reglas del negocio	
Titulo del objeto de aprendizaje: Basic level of English language	
Fecha: 2010/02/10	
Responsable(s): Sandro Jaramillo, Ricardo chicangana, Willian Jurado	
Descripción de la reglas del negocio	

-utilizar contenidos desarrollados en el PFI -abarcar solo los temas relacionados a la unidad uno, dos y tres del libro American Channel -orientado a estudiantes del PFI	
Información MGOAS	
Disciplina: Modelado de negocio	Numero iteración: 1,2
Observaciones:	

TECNICA DE USABILIDAD ANÁLISIS COMPETITIVO.

Analisis competitivo	
Titulo del objeto de aprendizaje: Basic level of English language	
Fecha: 2010/03/05	
Responsable(s): Sandro Jaramillo, Ricardo chicangana	
Aplicaciones y centros de enseñanza- aprendizaje del idioma ingles	
Sena Ingles para todos los colombianos Colombia aprende Sitio web www.mansioningles.com	
Falencias encontradas	
<ul style="list-style-type: none"> - No existe sincronización entre el sonido y el texto presentado. - Imágenes sin colores o colores opacos - Links rotos - Ejercicios incompletos - Demasiada información en español - Elementos pesados, es decir consumen ancho de banda y hacen que pierde eficiencia en la utilización de los materiales 	

Elementos a tener en cuenta en el desarrollo del OA	
<p>-Ejercicio interactivo para describir los colores.</p> <p>-Ejercicios de gramatica donde se adiciona información del tema estudiado.</p> <p>-Menus claros y practicos</p>	
Información MGOAS	
Disciplina: Modelado de negocio	Numero iteración: 1
Observaciones:	

ESPECIFICACIONES DE USABILIDAD

ESPECIFICACIONES DE USABILIDAD	
Titulo del objeto de aprendizaje: Basic level of English language	
Fecha: 2010/03/05	
Responsable(s): Sandro Jaramillo, Ricardo chicangana	
Especificaciones de Usabilidad que se pretende cumplir con el OA	
<p>Facilidad de aprendizaje: Cuán fácil es aprender la funcionalidad básica del sistema, como para ser capaz</p> <p>de realizar correctamente la tarea que desea realizar el usuario. Se mide normalmente por el tiempo empleado con el sistema hasta ser capaz de realizar ciertas tareas en menos de un tiempo dado (el tiempo empleado habitualmente por los usuarios expertos). Este atributo es muy importante para usuarios noveles.</p>	

<p>Eficiencia: El número de transacciones por unidad de tiempo que el usuario puede realizar usando el sistema. Lo que se busca es la máxima velocidad de realización de tareas del usuario. Cuanto mayor es la usabilidad de un sistema, más rápido es el usuario al utilizarlo, y el trabajo se realiza con mayor rapidez. Nótese que eficiencia del software en cuanto su velocidad de proceso no implica necesariamente eficiencia del usuario en el sentido en el que aquí se ha descrito.</p>	
<p>Recuerdo en el tiempo: Para usuarios intermitentes (que no utilizan el sistema regularmente) es vital ser capaces de usar el sistema sin tener que aprender cómo funciona partiendo de cero cada vez. Este atributo refleja el recuerdo acerca de cómo funciona el sistema que mantiene el usuario, cuando vuelve a utilizarlo tras un periodo de no utilización.</p>	
<p>Tasa de errores: Este atributo contribuye de forma negativa a la usabilidad de un sistema. Se refiere al número de errores cometidos por el usuario mientras realiza una determinada tarea. Un buen nivel de usabilidad implica una tasa de errores baja. Los errores reducen la eficiencia y satisfacción del usuario, y pueden verse como un fracaso en la transmisión al usuario del modo de hacer las cosas con el sistema.</p>	
<p>Satisfacción: Éste es el atributo más subjetivo. Muestra la impresión subjetiva que el usuario obtiene del sistema.</p>	
<p>Información MGOAS</p>	
<p>Disciplina: Modelado de negocio</p>	<p>Numero iteración: 1,2</p>
<p>Observaciones:</p>	

La metodología propone reglas que deben cumplirse, estas reglas están orientadas a guiar el proceso de desarrollo de OA contribuyendo a que los objeto de aprendizaje se elaboren de manera sistémica y estructurada.

Reglas de producción de OA	
Respectar la estructura estándar de un OA.	MGOAS exige que los objetos de aprendizaje contengan siempre los siguientes componentes. <ul style="list-style-type: none"> - Objetivo de aprendizaje. - Contenido informativo o educativo. - Actividad de aprendizaje - Evaluación. - Metadatos.
Utilizar contenidos de buena calidad.	Esta regla esta orientada a exigir que las Imágenes, videos, sonidos, animaciones, textos sean de calidad aceptables, se considera de calidad aceptable <ul style="list-style-type: none"> - Imágenes claras y a color. - Sonidos y videos: completos y concisos, con formatos reconocidos que sean de fácil reproducción. - Procurar que los textos no sean extensos. Así se evita que los usuarios se abstengan de leer.
Realizar el OA en base a prototipos.	Se complementa con Técnicas IPO prototipos de papel, esta regla de producción se considera importante ya que los prototipos proporcionan un ambiente iterativo identificando falencias, funcionalidades e iteración en cada prototipo.
Diseño consistente	La interfaz grafica no debe cambiar repentinamente es decir las iconos, imágenes, logos, fondos, menús deben ser consistentes en tamaño, colores y ubicación.
Procurar cumplir con las características básicas de los OA.	MGOAS define que los AO desarrollados por medio de la propuesta metodologicas se preocupen por cumplir con las siguientes características. <ul style="list-style-type: none"> - Reutilización - Accesabilidad - Interoperable - Educativo
Evaluar el OA	Se considera que la evaluación de los OA es necesario referirse a las técnicas de Usabilidad que sugiere MGOAS
Diligenciar información de los metadatos	Con respecto a los metadatos no se obliga la utilización para determinados metadatos, pero teniendo en cuenta la regla anterior es necesaria la utilización de metadatos para cumplir características de reutilización y Accesabilidad.
Empaquetamiento bajo el estándar SCORM	SCORM define cómo empaquetar los contenidos para importarse y exportarse entre plataformas, y describe las reglas que un LMS debe seguir a fin de presentar un aprendizaje específico
Definir requerimientos software y hardware	Durante la construcción del objeto de aprendizaje, se deben especificar los requerimientos hardware y software mínimos que se deben cumplir para el correcto funcionamiento de los objetos de aprendizaje
Publicar el OA en un LMS	.básicamente se busca desplegar el OA en el entorno de ejecución, por lo tanto el OA será adicionado como una actividad a desarrollar y evaluar e un curso.
Almacenar el OA en un Repositorio.	Almacenar el OA en un Repositorio.

ESTRUCTURA O MODELO DE NAVEGACIÓN

La estructura o modelo de navegación puede ser: lineal, en árbol o en red, tal como se expone en el apartado 3.7 del estándar ISO/IEC 14915-2 [3], y como se presenta a continuación.

Las estructuras básicas son la base para crear otras estructuras [1].

Estructura lineal: se organizan los elementos en secuencia. Un ejemplo de estructura lineal son los trozos o segmentos de contenidos como se ilustra en la siguiente figura.

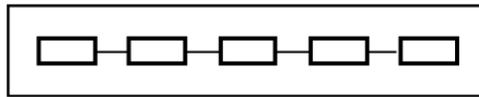


Figura 15. Ejemplo de estructura lineal.

Estructura en árbol: se organiza los elementos de forma jerárquica, donde cada componente es asociado solo con un componente de un nivel superior y puede ser asociado con muchos componentes de un nivel inferior. Un ejemplo de estructura en árbol se ilustra en la siguiente figura.

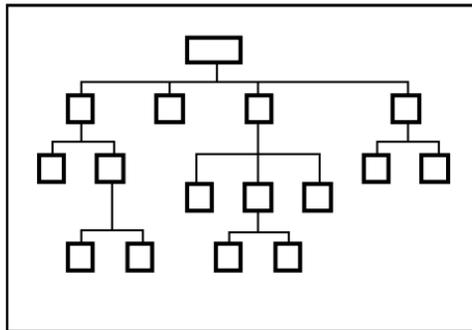


Figura 16. Ejemplo de estructura en árbol.

Estructura en red: se organiza los componentes de manera que cada componente puede ser asociado con múltiples componentes. Un ejemplo es una estructura conectada completamente como se ilustra en la siguiente figura.

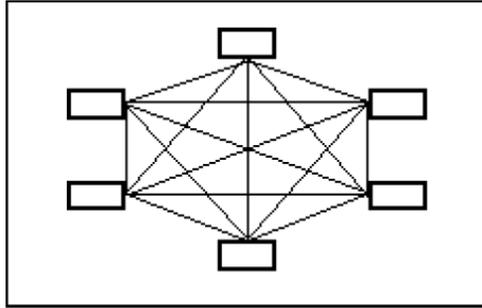


Figura 17. Ejemplo de estructura en red.

[1] International Standards Organization, "Software ergonomics for multimedia user interfaces - Part [2] Multimedia navigation and control", *International Standards Organization*, ISO 14915-2, 2002. [En línea]. Disponible: http://www.iso.org/iso/catalogue_detail.htm?csnumber=28583 [Accedido Jun. 10, 2009].