

UNIVERSIDAD DE CASTILLA-LA MANCHA
ESCUELA SUPERIOR DE INFORMÁTICA



Asignatura: Ingeniería del Software II

Grupo de Titulación (24/25): C02

Titulación: Grado en Ingeniería Informática

Fecha: 15/11/2024

TRABAJO TEÓRICO

Grupo Trabajo	C02
Componentes	1. Sandra Rodríguez Sánchez-Gil 2. Steven Tipantuña Aquieta 3. Raúl Mata Muela 4. Álvaro Mozos Olmo 5. Alonso Zamora Zamora 6. Manuel Villanueva Alises

ÍNDICE

1. Realización de la planificación	3
1.1 Explicación de prioridades:	3
2.Tabla de planificación:	4
2.1 Iteración 0 (It0).....	4
2.2 Iteración 1 y 2 (It1 - It2)	5
2.3 Iteración 3 y 4 (It3 - It4)	5
2.4 Iteración 5 y 6 (It5 - It6)	5
2.5 Iteración 7 y 8 (It7 - It8)	6
2.6 Iteración 9 y 10 (It9 - It10).....	6
2.7 Iteración 11 y 12 (It11 - It12).....	6
2.8 Iteración 13 y 14 (It13 - It14).....	7
2.9 Iteración 15 y 16 (It15 - It16).....	7
2.10 Iteración 17 y total de horas por trabajador (It17 - total)	7
3. Fases:	8
4. Calendario:	8
5. Gestión de Calidad	10
5.1.1. Mantenibilidad:.....	10
5.1.2. Disponible (subcaracterística de Fiabilidad):	10
5.1.3. Seguridad:	10
5.2. Especificar evaluación	11
5.3. Diseñar evaluación	11
5.4. Llevar a cabo la evaluación.....	12
5.5. Finalizar la evaluación (Informe de calidad)	13
6. Gestión de Configuración	14
6.1. Introducción.....	14
6.2. Criterios para la Identificación de los Elementos de Configuración	14
6.3. Limitaciones y Suposiciones	14
6.4. Responsabilidades y Autoridades del Plan	15
6.5. Organización del Proyecto	15
6.6. Responsabilidades de CM	15
6.7. Políticas Aplicables, Directivas y Procedimientos	15
6.8. Actividades Planificadas, Agenda y Recursos	15
6.9. Mantenimiento del Plan de CM	15

1. Realización de la planificación

La planificación de los casos de uso se ha realizado considerando la funcionalidad y la prioridad de cada caso. A continuación, dejamos una tabla que resume todo lo mencionado:

Caso de Uso	Funcionalidad	Prioridad
CDU 1	Elaborar mapa de activos de datos	1
CDU 2	Documentar procesos de negocio	3
CDU 3	Identificar y documentar requisitos de datos	3
CDU 4	Identificar responsables de proyectos de datos	4
CDU 5	Clasificar datos (confidenciales, no confidenciales, etc.)	1
CDU 6	Definir dominios de datos	2
CDU 7	Gestionar y compartir permisos de uso de datos	1
CDU 8	Crear metadatos de negocio, técnicos y operativos	2
CDU 9	Sistema de alertas y avisos	5
CDU 10	Descubrir datos en repositorios	3
CDU 11	Describir datos a nivel conceptual, lógico y físico	3
CDU 12	Describir ciclo de vida del dato (linaje)	4
CDU 13	Cliente	1
CDU 14	Servidor	1
CDU 15	Multidispositivo	1
CDU 16	Realizar evaluaciones de calidad de datos	4

1.1 Explicación de prioridades:

A continuación, explicaremos brevemente la asignación de prioridades para cada caso de uso, según su relevancia para el funcionamiento del sistema.

Prioridad 1: Son casos de uso esenciales e imprescindibles para el funcionamiento básico y seguro del sistema.

Prioridad 2: Casos de uso más importantes que harán una mejor organización de los datos.

Prioridad 3: Son casos de uso relevante para mejorar la documentación y estructura. Son útiles, pero no esenciales.

Prioridad 4: Casos de uso secundarios que implementarán detalles para una mejor experiencia del usuario, el impacto en el funcionamiento es mínimo.

Prioridad 5: Son casos de uso opcionales que mejoran la experiencia de usuario con el sistema, como alertas o avisos.

El orden que hemos seguido de prioridades para hacer la tabla de planificación es el siguiente:

Iteración	It1	It2	It3	It4	It5	It6	It7	It8	It9	It10	It11	It12	It13	It14	It15	It16
Caso de Uso	CDU1	CDU5	CDU7	CdU13	CdU14	Cdu15	CDU6	CDU8	CDU2	CDU3	CDU10	CDU11	CDU4	CDU12	CDU16	CDU9

2.Tabla de planificación:

En esta sección presentaremos la planificación detallada del proyecto, desglosada en iteraciones. Cada iteración incluye las tareas asignadas a los miembros del equipo, junto con la distribución de responsabilidades y la carga horaria estimada.

2.1 Iteración 0 (It0)

	It0				
	R	A	D	I	T
Sandra (requisitos y análisis)	2	1			
Alonso(requisitos y análisis)	1	3			
Raúl(diseñador e implementador)			2	1	
Steven(diseñador e implementador)			2	3	
Alvaro(testers)					
Manuel(testers)					
Coste	1.020 €				
Horas	15				

2.2 Iteración 1 y 2 (It1 - It2)

It1					It2				
R	A	D	I	T	R	A	D	I	T
3						2			
1	2				2				
		4	2				3	2	
		2	2				2	3	
				1					2
				1					2
1.100 €					910 €				
18					18				

2.3 Iteración 3 y 4 (It3 - It4)

It3					It4				
R	A	D	I	T	R	A	D	I	T
3	1				1	1			
2	2				2				
		3	2				3	3	
		2	3				3	4	
				2					2
				2					2
1.300 €					1.040 €				
22					21				

2.4 Iteración 5 y 6 (It5 - It6)

It5					It6				
R	A	D	I	T	R	A	D	I	T
2	1				1	4			
1	2				1	1			
		3	2				3	3	
		2	3				4	2	
				1					3
				2					3
1.080 €					1.340 €				
19					25				

2.5 Iteración 7 y 8 (It7 - It8)

It7					It8				
R	A	D	I	T	R	A	D	I	T
1	3				2	2			
1	2					2			
		4	1				4		
		3	3				2	3	
				2					2
				1					2
1.250 €					1.090 €				
21					19				

2.6 Iteración 9 y 10 (It9 - It10)

It9					It10				
R	A	D	I	T	R	A	D	I	T
2	1				2				
	3				1	3			
		3	2				5	2	
		4	2				3	4	
									3
1.100 €					1.290 €				
17					23				

2.7 Iteración 11 y 12 (It11 - It12)

It11					It12				
R	A	D	I	T	R	A	D	I	T
1	1				2	2			
	2				2	4			
		3	5				2	4	
		3	2				2	4	
				2					1
				2					2
1.020 €					1.480 €				
21					25				

2.8 Iteración 13 y 14 (It13 - It14)

It13					It14				
R	A	D	I	T	R	A	D	I	T
1	3				2	4			
2	1				2	3			
		1					4	2	
		2	3				3	3	
				4					1
				1					2
1.030 €					1.660 €				
18					26				

2.9 Iteración 15 y 16 (It15 - It16)

It15					It16				
R	A	D	I	T	R	A	D	I	T
2					3	1			
	3				4				
		3	2				4	1	
		2	3				1	3	
				3					3
				2					2
1.020 €					1.310 €				
20					22				

2.10 Iteración 17 y total de horas por trabajador (It17 - total)

It17					Total
R	A	D	I	T	
3	4				64
5	3				63
		3	1		92
		3	1		96
				5	34
				4	33
2.030 €					22.070 €
32					382

3. Fases:

La siguiente tabla muestra la planificación por fases del proyecto, incluyendo costo, horas y duración de cada iteración. Con el fin de visualizar y gestionar los recursos del proyecto.

FASE	INICIO	ELABORACIÓN	CONSTRUCCIÓN															TRANSICIÓN
ITERACION	It0	It1	It2	It3	It4	It5	It6	It7	It8	It9	It10	It11	It12	It13	It14	It15	It16	It17
COSTE	1.020	1.100	910	1.300	1.040	1.080	1.340	1.250	1.090	1.100	1.290	1.020	1.480	1.030	1.660	1.020	1.310	2.030
HORAS	15	18	18	22	21	19	25	21	19	17	23	21	25	18	26	20	22	32
AGENDA	2 días	3 días	3 días	3 días	3 días	3 días	4 días	3 días	3 días	3 días	3 días	3 días	4 días	3 días	4 días	3 días	3 días	5 días

4. Calendario:

El calendario detalla las fechas de ejecución de cada iteración durante noviembre, diciembre y enero permitiendo un control eficiente del cronograma del proyecto.

NOVIEMBRE						
LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES	SÁBADO	DOMINGO
1	It0 2	It0 3	It1 4	It1 5	6	7
It1 8	It2 9	It2 10	It2 11	It3 12	13	14
It3 15	It3 16	It4 17	It4 18	It4 19	20	21
It5 22	It5 23	It5 24	It6 25	It6 26	27	28
It6 29	It6 30					

DICIEMBRE						
LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES	SÁBADO	DOMINGO
		It7 ₁	It7 ₂	It7 ₃	4	5
6	It8 ₇	It8 ₈	It8 ₉	It9 ₁₀	11	12
It9 ₁₃	It9 ₁₄	It10 ₁₅	It10 ₁₆	It10 ₁₇	18	19
It11 ₂₀	It11 ₂₁	It11 ₂₂	It12 ₂₃	24	25	26
It12 ₂₇	It12 ₂₈	It12 ₂₉	It13 ₃₀	31		

ENERO						
LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES	SÁBADO	DOMINGO
					1	2
It13 ₃	It13 ₄	It14 ₅	6	It14 ₇	8	9
It14 ₁₀	It14 ₁₁	It15 ₁₂	It15 ₁₃	It15 ₁₄	15	16
It16 ₁₇	It16 ₁₈	It16 ₁₉	It17 ₂₀	It17 ₂₁	22	23
It17 ₂₄	It17 ₂₅	It17 ₂₆	27	28	29	30
31						

5. Gestión de Calidad

En este documento se discutirán tres características de calidad que deberían ser implementadas en desarrollo como parte de requisitos.

A continuación, nombrare características que son críticas y que deben ser adoptadas en las fases de requisitos y diseño basándome en el sentido de nuestro proyecto y su uso futuro:

5.1.1. Mantenibilidad:

- **Porque:** MDX debe ser mantenible con el paso del tiempo, y tiene que ser efectivo y eficiente su mantenimiento, y poder modificarse añadiendo nuevas funcionalidades, corrección de errores o ajustes sin afectar la experiencia al usuario
- **Implementación:** Separar todo en módulos diferentes y que no se afecten entre sí. Además del uso del versionado en Git y diferenciar todo en ramas para poder hacer cambios sin afectar al sistema principal.

5.1.2. Disponible (subcaracterística de Fiabilidad):

- **Porque:** MDX tiene que estar siempre operativa debido que gestionara datos muy importantes para empresas y no puede pasar mucho tiempo actualizándose o en mantenimiento por errores.
- **Implementación:** Esto se puede evitar con una arquitectura de cliente-servidor distribuida que si un elemento falla el sistema pueda seguir funcionando o que se avise del fallo.

5.1.3. Seguridad:

- **Porque:** La aplicación MDX maneja información muy sensible que no puede ser vista por todo el mundo, o se debe diferenciar entre rangos de la empresa que pueden ver esa información.
- **Implementación:** Crear un sistema de acceso muy restringido que diferencie entre rangos de la empresa y que oculte información sensible a usuarios que no puedan verla.

En este apartado completaremos con las conclusiones sacadas de la reunión del 10/11/24 sobre como añadir estas características, con estos pasos:

5.2. Especificar evaluación

Para este paso se definirán cinco usuarios que necesitarán de estas características de calidad y en que peso o porcentaje les importara o afectara al usar MDX.

Los usuarios están definidos ya, para los que ya están especificados los requisitos funcionales, pero en este apartado se verá el peso de cada uno para estos **requisitos no funcionales de calidad**, así como luego veremos en que porcentaje los satisface y por último haremos una media sobre el peso y la puntuación de satisfacción para ver que usuario se ve más beneficiado por estas características de calidad.

5.3. Diseñar evaluación

Mostraremos los usuarios en una tabla para cada fila y cada columna los requisitos de calidad y el porcentaje que afecta a cada usuario:

Tipo de Usuario	Mantenibilidad	Disponibilidad	Seguridad
Propietario de datos	20%	40%	40%
Implicado en datos	30%	30%	40%
Gestor de seguridad	10%	30%	60%
Gestor de calidad	20%	50%	30%
Usuario de metadatos	40%	40%	20%

Después de analizar como cada requisito afecta a cada usuario le daremos un valor o puntuación a como de bien se adecua a cada usuario de MDX:

Tipo de Usuario	Mantenibilidad	Disponibilidad	Seguridad
Propietario de datos	20% 80	40% 85	40% 90

Implicado en datos	30% 75	30% 80	40% 85
Gestor de seguridad	10% 70	30% 75	60% 95
Gestor de calidad	20% 85	50% 90	30% 80
Usuario de metadatos	40% 90	40% 85	20% 75

5.4. Llevar a cabo la evaluación

Por último, una tabla donde combinaremos los porcentajes de los pesos y las puntuaciones de cada usuario y obtendremos la media ponderada:

Tipo de Usuario	Mantenibilidad	Disponibilidad	Seguridad	Media Ponderada
Propietario de datos	20% 80	40% 85	40% 90	85.5
Implicado en datos	30% 75	30% 80	40% 80	80.5
Gestor de seguridad	10% 70	30% 75	60% 75	83.5
Gestor de calidad	20% 88	50% 92	30% 85	88.6

Usuario de metadatos	40% 90	40% 85	20% 85	84.0
-----------------------------	-----------	-----------	-----------	-------------

Para el cálculo de la media ponderada por ejemplo de propietario de datos:

- Mantenibilidad: $20\% \times 80 = 16$
- Disponibilidad: $40\% \times 85 = 34$
- Seguridad: $40\% \times 90 = 36$

Media Ponderada: $16 + 34 + 36 = 85.5$

Como conclusión de esta primera evaluación hemos obtenido o sacado en claro que el **gestor de calidad** es el que más beneficiado se ve por estos requisitos no funcionales de calidad.

5.5. Finalizar la evaluación (Informe de calidad)

A continuación, realizaremos un informe de calidad que enviaríamos como solicitud de certificación para que nuestro producto sea sellado como un producto de calidad.

Hemos decidido que para cada característica de calidad crearíamos un requisito no funcional que se añadiría a los requisitos en la fase de desarrollo y son estos:

5.5.1. Mantenibilidad

- El sistema debe estar diseñado en módulos independientes con bajo acoplamiento y alta cohesión, de forma que las modificaciones en un módulo no afecten el funcionamiento de otros módulos.
- Es decir, un sistema basado en interfaces que manejen las dependencias entre los submódulos.

5.5.2 Disponibilidad

- MDX debe tener una disponibilidad del 100% durante el horario laboral 8:00 a 20:00 por ejemplo, con un tiempo de inactividad máximo de 10 minutos al mes para mantenimiento programado.

5.5.3 Seguridad

- El sistema debe implementar un control de acceso basado en roles, permitiendo visualizar la información sensible solo a los usuarios con los

permisos necesarios.

- Puede ser un sistema de doble factor con aplicaciones como authenticador que mande un código a el móvil personal de cada usuario.

6. Gestión de Configuración

6.1. Introducción

- Propósito del Plan: Establecer una guía para la gestión de configuración del sistema Metadata Management and Explorer (MDX), asegurando la integridad del producto, facilitando el control de cambios y optimizando el control del sistema a lo largo de todo su ciclo de vida.
- Alcance del Plan: Este plan abarca desde la identificación, control hasta la auditoría de los elementos de configuración del sistema MDX, iniciando en su fase de desarrollo hasta su mantenimiento y con el tiempo, la retirada.
- Relación con la Organización y Otros Proyectos: El plan de CM está alineado con los estándares de la IEEE 828 y los objetivos específicos del proyecto del trabajo de teoría.
- Términos Claves y Referencias: Se aplican términos como baseline, branching, release, y configuración de auditorías.

6.2. Criterios para la Identificación de los Elementos de Configuración

- Se aplicará CM a todos los componentes del sistema, como documentos de requisitos, código fuente, bases de datos de metadatos, scripts, y manuales de usuario.
- Los elementos de configuración serán identificados y para un control efectivo.

6.3. Limitaciones y Suposiciones

- Limitaciones: Recursos económicos limitados por parte del cliente y un presupuesto fijo para el desarrollo.
- Suposiciones: El equipo estará compuesto por 6 personas (2 especialistas en requisitos y analistas, 2 diseñadores e implementadores y 2 testers), con un enfoque multidispositivo para cumplir con los requisitos del cliente.

6.4. Responsabilidades y Autoridades del Plan

- Autoridades: La coordinadora (equivalente al Configuration Management Authority en el estándar) será la encargada de evaluar y aprobar cambios en la configuración.
- Responsables: El equipo de trabajo es responsable de seguir el plan y asegurar la correcta implementación de las actividades de CM.

6.5. Organización del Proyecto

- El equipo se organiza de acuerdo con los roles descritos en el documento de planificación, con la asignación de horas y costes detallada en un Excel específico para la gestión de recursos.

6.6. Responsabilidades de CM

- Control de versiones: Se utilizará el sistema GitFlow para el control de ramas y versiones del código.
- Auditoría: Se realizarán reuniones de configuración antes de cada liberación para la comprobación de satisfacer los requisitos preestablecidos.

6.7. Políticas Aplicables, Directivas y Procedimientos

- Procedimientos basados en el estándar IEEE 828-2012 para la auditoría, control de cambios y liberación de versiones.
- Uso de Semantic Versioning para el etiquetado y control de versiones.

6.8. Actividades Planificadas, Agenda y Recursos

- Agenda: El proyecto comienza con la Iteración 0, centrada en la planificación y definición de requisitos, seguida por el desarrollo incremental en iteraciones posteriores.
- Recursos de CM: Nuestra principal herramienta será Git, para el control de versiones, la documentación del proyecto y la misma organización de este.

6.9. Mantenimiento del Plan de CM

- Actualizaciones continuas del plan según posibles cambios en los requisitos y durante cada iteración. Se establecerán revisiones periódicas del plan para adaptarse a nuevas necesidades del proyecto.
- El plan nos guiará al inicio de cada nueva fase para asegurar la integridad y viabilidad de los cambios aplicados.