



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE MÁSTER

Equipo o Sin Equipo. Estudio comparativo del impacto de los equipos de desarrollo en el ciclo de vida integral de las aplicaciones

Máster Universitario en Dirección TIC para la Defensa

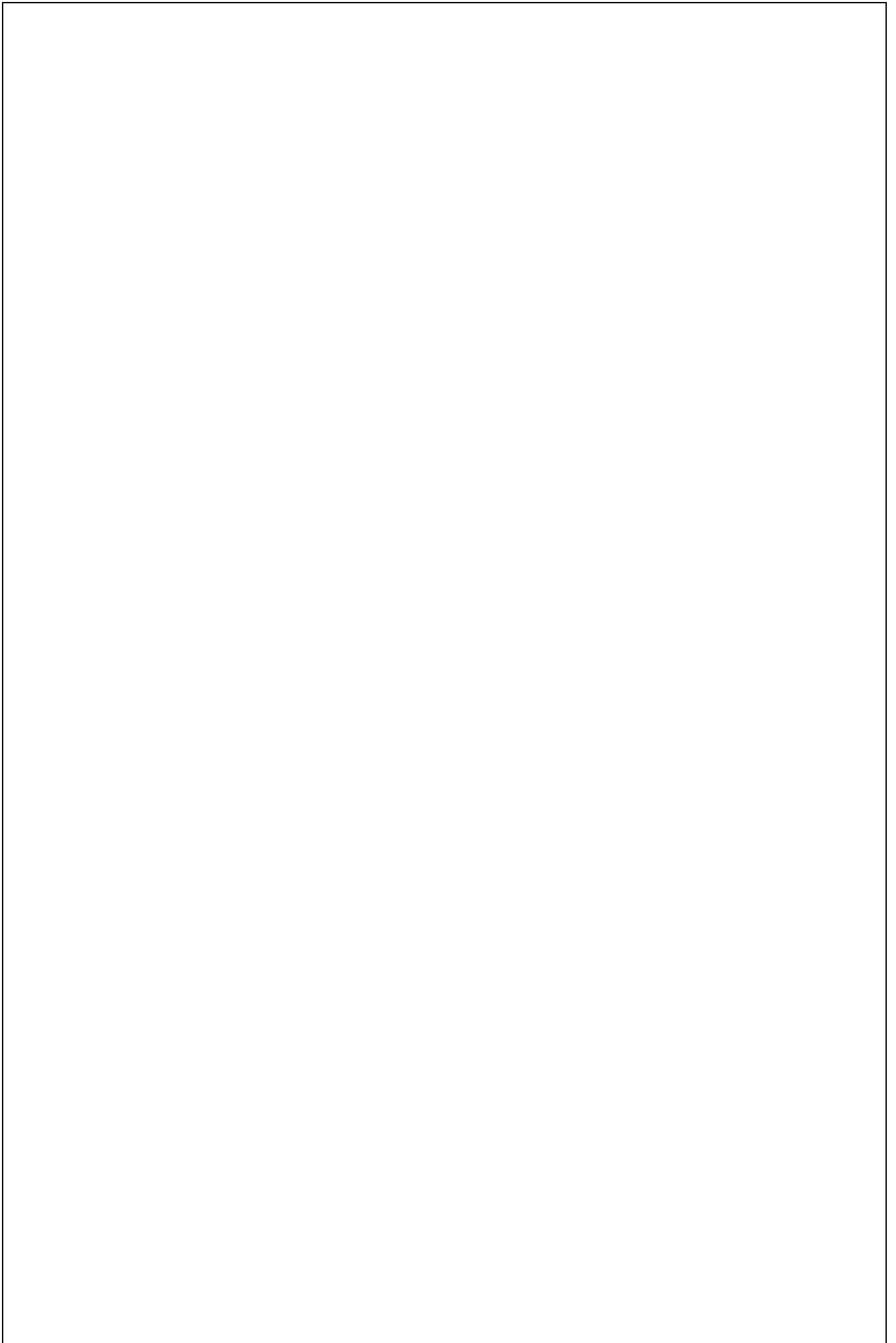
ALUMNO: Juan Carlos Gutiérrez Palomo

DIRECTORES: Miguel Ángel Ares Tarrío

José González Coma

CURSO ACADÉMICO: 2024-2025

Universida_{de}Vigo





Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE MÁSTER

Equipo o Sin Equipo. Estudio comparativo del impacto de los equipos de desarrollo en el ciclo de vida integral de las aplicaciones

Máster Universitario en Dirección TIC para la Defensa
Especialidad de Sistemas y Tecnologías de la Información

Universida_{de}Vigo

RESUMEN

Vivimos en una sociedad en constante cambio y evolución en todas las disciplinas humanas. La evolución de los lenguajes de programación no se queda atrás y, por ende, el desarrollo de software. Esta constante innovación supone un avance muy significativo en disciplinas como medicina, ingeniería, física, computación. En este contexto, una parte importante del software es desarrollado por equipos externos, suponiendo que la externalización en el desarrollo de aplicaciones es sinónimo de buenos resultados, ya que crea valor para el cliente.

Sin embargo, no siempre se tiene en cuenta los riesgos inherentes que puede traer consigo esta elección, ni la pérdida que supone, en términos no cuantitativos, de habilidades y capacidades como son el talento, la experiencia y el conocimiento en el desarrollo de aplicaciones. Como consecuencia, el proceso de desarrollo de aplicaciones presenta un dilema: decidir entre desarrollar software internamente o externalizar este proceso.

Por un lado, disponer de un equipo de desarrollo propio, supone un valor añadido ya que garantiza el conocimiento y las habilidades mencionadas y permite que los equipos puedan adaptarse más rápidamente a los cambios y a las necesidades del mercado. Por otro lado, la externalización, puede provocar una falta de resiliencia organizacional debido a la gran dependencia de los servicios proporcionados por los proveedores. Esto puede convertir una organización en una entidad cautiva.

Para analizar esta disyuntiva, se requiere de un análisis comparativo de ambos equipos, acotado el estudio solamente al resultado del desarrollo de software y excluyendo factores como los lenguajes empleados, metodologías, infraestructura, entre otros.

Este trabajo de fin de máster (TFM) pretende analizar y comparar el impacto de los equipos de desarrollo interno frente a los externos en el ciclo de vida integral de las aplicaciones, estudiando si la presencia de un equipo interno mejora el proceso de desarrollo, las capacidades operativas y el aseguramiento de la calidad a través de buenas prácticas ofreciendo una visión razonada que ayude a las organizaciones a tomar decisiones apropiadas y adecuadas a sus intereses.

PALABRAS CLAVE

Desarrollo de aplicaciones, In-house, Externalización, Outsourcing, Ciclo de vida integral de las aplicaciones, SDLC.

AGRADECIMIENTOS

A mis hijos, Susana y Carlos, mi ángel y mi luchador, porque cada día aprendo de ellos.

A mis padres por todo su esfuerzo y dedicación.

A mis amigos de siempre, Yoyi y Silvia, que siempre han estado a pesar de las adversidades.

A D. Miguel Ángel Ares Tarrío y D. José González Coma por su orientación y apoyo.

Y, por último y no menos importante, al Ejército del Aire y del Espacio, por todo lo que me ha dado.

Tabla de contenido

Índice de Figuras	3
Índice de Tablas.....	4
1 Introducción.....	6
1.1 Desafíos Actuales.....	6
1.2 Objetivos	7
2 Estado del Arte	9
2.1 Base de Conocimiento	9
2.1.1 Hardware.....	9
2.1.2 Software	10
2.1.3 Tipos de Software	11
2.1.3.1 Según su función.....	11
2.1.3.2 Según su licencia	12
2.1.3.3 Según su uso	13
2.1.3.4 Según su plataforma.....	13
2.1.4 Requisitos para el Desarrollo de Software.....	15
2.1.4.1 Perfiles Necesarios.....	15
2.1.4.2 Ciclo de vida del Desarrollo de Software (SDLC)	19
2.1.4.3 Entornos de Despliegue	23
2.1.4.4 Mantenimiento	26
2.1.4.5 Aseguramiento de la Calidad	27
3 Equipo o Sin Equipo. Estudio Comparativo para el Desarrollo de aplicaciones	29
3.1 Elementos del Estudio.....	29
3.1.1 Equipo Interno (In-house).....	29
3.1.1.1 Tipos de Equipos Internos (<i>In-house</i>).....	29
3.1.1.2 Ventajas e Inconvenientes	30
3.1.2 Equipo Externo (Outsourcing).....	30
3.1.2.1 Tipos de Equipos Externos (<i>Outsourcing</i>).....	30
3.1.2.2 Ventajas e Inconvenientes	31
3.1.3 Elementos Análogos	31
3.1.3.1 Perfiles	31
3.1.3.2 Tecnologías	31
3.1.3.3 Metodologías de Trabajo	32
3.2 Metodología de Investigación	35
3.2.1 Criterios Comunes	35
3.2.2 Métricas	37

3.2.2.1 Métricas Cuantitativas	37
3.2.2.2 Métricas Cualitativas	38
3.2.3 Selección de Casos de Uso	40
3.2.3.1 Indicadores Cuantitativos	41
3.2.3.2 Indicadores Cualitativos	48
4 Resultados	49
4.1 Análisis comparativo de Resultados	49
4.1.1 Resumen Indicadores Cuantitativos y Cualitativos	49
4.1.2 Indicadores Evaluación del Ciclo de Vida del Proyecto (SDLC).....	50
4.1.3 Indicadores Operativos	51
4.1.4 Indicadores de Calidad del Código Fuente (SonarQube)	53
4.1.5 Indicadores Cualitativos	55
5 Conclusiones y Líneas futuras.....	57
5.1 Conclusiones	57
5.2 Líneas futuras	58
6 Bibliografía.....	59
Anexo I: Métricas Cuantitativas.....	63
A1-1 INVENTIC	63
A1-2 IDMANAGE.....	68
A1-3 SANIGEST	73
A1-4 CARGONET	78
A1-5 PLANEA.....	83
A1-6 ECOGEST	88
A1-7 MONITORA.....	93
A1-8 UNITPLAN.....	98
A1-9 SENSORPRO	103
A1-10 GESLOG	108
Anexo II: Métricas Cualitativas	113
A2-1 EQUIPO INTERNO (<i>IN-HOUSE</i>).....	113
A2-2 EQUIPO EXTERNO (<i>OUTSOURCING</i>).....	118
Anexo III: Métricas Cuantitativas (Complementarias)	125
Anexo IV: Métricas Cualitativas (Complementarias)	134

ÍNDICE DE FIGURAS

Figura 1-1 Entorno VUCA + H, aptitudes para mitigar el cambio [2].....	6
Figura 1-2 Entorno BANI [4]	7
Figura 2-1 Línea de tiempo Hardware [6]	9
Figura 2-2 Línea de tiempo de Lenguajes de programación [8]	11
Figura 2-3 Clasificación del software según su función (elaboración propia).....	12
Figura 2-4 Clasificación software según su licencia [10]	13
Figura 2-5 Ingresos generados por software [11].....	15
Figura 2-6 Fases del SDLC (elaboración propia).....	20
Figura 2-7 Estimación de tiempo en las fases del SDLC (elaboración propia).....	21
Figura 2-8 Entornos diferenciados [20].....	26
Figura 2-9 Capacidades del desarrollo de aplicaciones [22]	28
Figura 3-1 Metodología en Cascada [27]	32
Figura 3-2 Metodología en V [27].....	33
Figura 3-3 Metodología en Espiral [27]	33
Figura 3-4 Metodología SCRUM [27]	34
Figura 3-5 Metodología KANBAN [27]	34
Figura 3-6 Metodología <i>Extreme Programming</i> [27]	35
Figura 3-7 Interfaz Herramienta Jira (elaboración propia).....	41
Figura 3-8 Interfaz Herramienta SonarQube (elaboración propia)	41
Figura 4-1 Interpretación Indicadores Evaluación del Ciclo de Vida del Proyecto (SDLC) (elaboración propia)	51
Figura 4-2 Interpretación Indicadores Operativos (elaboración propia)	52
Figura 4-3 Interpretación Indicadores de Calidad del Código Fuente (SonarQube) (elaboración propia)	54
Figura 4-4 Interpretación Indicadores Cualitativos (elaboración propia)	56

ÍNDICE DE TABLAS

Tabla 2-1 Perfiles necesarios para desarrollar software (elaboración propia) [14].....	19
Tabla 2-2 Estimación de tiempo empleado y perfiles que intervienen en las fases del SDLC [15]	21
Tabla 2-3 Ventajas e Inconvenientes del entorno de desarrollo (elaboración propia) [17].....	24
Tabla 2-4 Ventajas e Inconvenientes del entorno de preproducción (elaboración propia) [18]	25
Tabla 2-5 Ventajas e Inconvenientes del entorno de producción (elaboración propia) [19].....	26
Tabla 3-1 Ventajas e Inconvenientes de un equipo interno (<i>in-house</i>) [24].....	30
Tabla 3-2 Ventajas e Inconvenientes de un equipo externo (<i>outsourcing</i>) [24].....	31
Tabla 3-3 Proyectos comparados por pares según criterios comunes (elaboración propia).....	40
Tabla 3-4 Indicadores para la Evaluación del Ciclo de Vida del Proyecto (SDLC) (elaboración propia)	45
Tabla 3-5 Relación entre Indicadores Evaluación del Ciclo de Vida del Proyecto (SDLC) e Indicadores Operativos (elaboración propia)	46
Tabla 3-6 Indicadores Operativos (elaboración propia).....	47
Tabla 3-7 Indicadores de Calidad del Código Fuente (SonarQube) (elaboración propia)	48
Tabla 4-1 Resumen Indicadores del Equipo Interno (elaboración propia)	50
Tabla 4-2 Resumen Indicadores del Equipo Externo (elaboración propia)	50
Tabla 4-3 Resumen Indicadores Evaluación del Ciclo de Vida del Proyecto (SDLC) (elaboración propia)	50
Tabla 4-4 Resumen Indicadores Operativos (elaboración propia).....	52
Tabla 4-5 Resumen Indicadores de Calidad del Código Fuente (SonarQube) (elaboración propia)	54
Tabla 4-6 Resumen Indicadores Cualitativos (elaboración propia)	55
Tabla A1-1 Métricas Cuantitativas InvenTIC (elaboración propia).....	67
Tabla A1-2 Métricas Cuantitativas IDManage (elaboración propia)	72
Tabla A1-3 Métricas Cuantitativas SaniGest (elaboración propia).....	77
Tabla A1-4 Métricas Cuantitativas CargoNet (elaboración propia).....	82
Tabla A1-5 Métricas Cuantitativas PlanEA (elaboración propia).....	87
Tabla A1-6 Métricas Cuantitativas EcoGest (elaboración propia).....	92
Tabla A1-7 Métricas Cuantitativas Monitora (elaboración propia)	97
Tabla A1-8 Métricas Cuantitativas UnitPlan (elaboración propia).....	102
Tabla A1-9 Métricas Cuantitativas SensorPro (elaboración propia).....	107
Tabla A1-10 Métricas Cuantitativas GesLog (elaboración propia)	112
Tabla A2-1 Métricas Cualitativas Equipo Interno (<i>In-house</i>) (elaboración propia)	117
Tabla A2-2 Métricas Cualitativas Equipo Externo (<i>Outsourcing</i>) (elaboración propia)	122
Tabla A3-1 Funcionalidad [34]	126

Tabla A3-2 Seguridad [34]	127
Tabla A3-3 Disponibilidad [34]	128
Tabla A3-4 Fiabilidad [34]	128
Tabla A3-5 Usabilidad [34]	129
Tabla A3-6 Mantenibilidad [34]	130
Tabla A3-7 Compatibilidad [34]	131
Tabla A3-8 Portabilidad [34]	132
Tabla A3-9 Proceso de Desarrollo e Impacto del Software [34]	133
Tabla A4-1 Talento [35]	135
Tabla A4-2 Experiencia [35]	135
Tabla A4-3 Conocimiento [35]	136
Tabla A4-4 Resolución de Problemas Complejos [35]	137
Tabla A4-5 Pensamiento Crítico [35]	137
Tabla A4-6 Adaptabilidad [35]	138
Tabla A4-7 Comunicación Efectiva [35]	138

1 INTRODUCCIÓN

1.1 Desafíos Actuales

En esta sociedad, donde los cambios son continuos y variables, especialmente en el ámbito de las Telecomunicaciones y Sistemas de Información (TI), las organizaciones se enfrentan a numerosos desafíos. Estos desafíos están definidos por el entorno VUCA+H (*Volatility* (Volátil), *Uncertainty* (Incierto), *Complexity* (Complejo), *Ambiguity* (Ambiguo), *Hyperconnectivity* (Hiperconectado)), lo que puede tener un impacto muy significativo en el desarrollo de aplicaciones, como, por ejemplo, la adaptabilidad y la flexibilidad frente a cambios rápidos e impredecibles.

Este concepto nacido en los años 90 hace referencia a un marco inseguro y cambiante de una organización caracterizado por cambios rápidos e impredecibles, múltiples factores para la toma de decisiones e incertidumbre en situaciones no definidas. Aparte, se añade la complejidad en entornos hiperconectados. Este entorno puede representar una serie de retos, pero también pueden convertirse en una oportunidad si se gestiona adecuadamente. Un claro ejemplo de entorno VUCA + H fue la pandemia de COVID-19. [1]

En la siguiente figura se muestra características del entorno VUCA + H junto con las aptitudes necesarias para mitigar las mismas:



Figura 1-1 Entorno VUCA + H, aptitudes para mitigar el cambio [2]

A su vez, y como evolución al entorno de VUCA+H, aparece el concepto de BANI (*Brittle* (Frágil), *Anxious* (Ansioso), *Non linear* (No lineal) e *Incomprehensible* (Incompresible)), que introduce elementos emocionales y psicológicos que influyen en la forma en que personas y organizaciones

responden a estos desafíos. Estos factores aumentan la necesidad de mantener y desarrollar internamente las capacidades de un equipo interno (*in-house*), ya que una dependencia excesiva puede convertir a una organización en una entidad cautiva de empresas proveedoras de software pudiendo provocar una falta de resiliencia organizacional. El ejemplo del COVID-19 también se puede incluir en este entorno. [3]

La siguiente figura refleja las características del entorno BANI desde el punto de vista individual y por grupo:

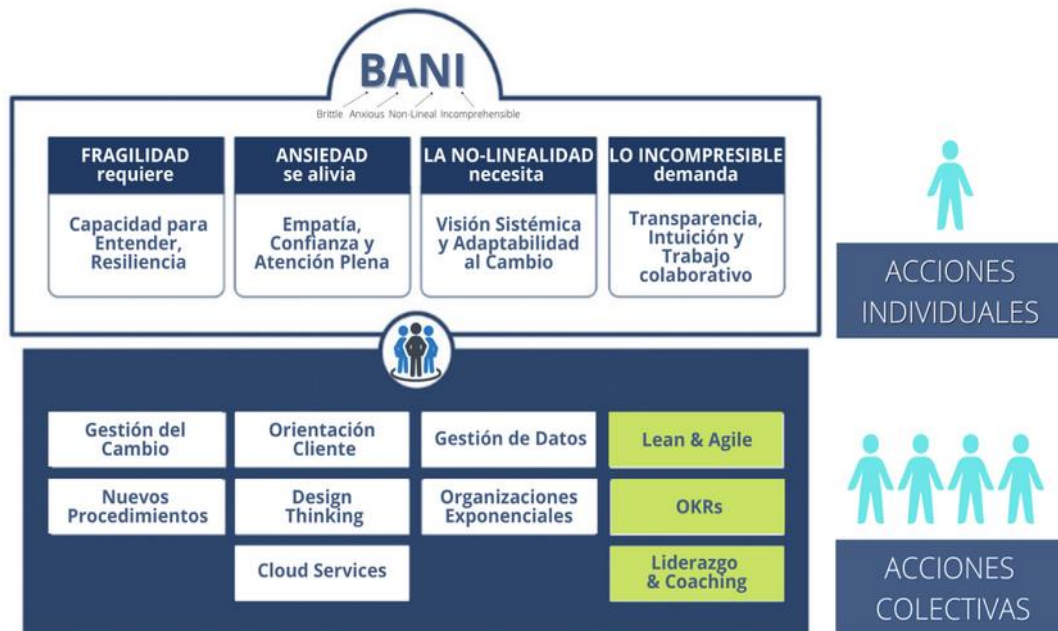


Figura 1-2 Entorno BANI [4]

Por ello, el desarrollo de aplicaciones está sujeto a muchos desafíos a la hora de evaluar la elección que deben considerarse para decidir sobre la necesidad o no de disponer de un equipo de desarrollo interno o externo.

1.2 Objetivos

Este trabajo fin de máster (TFM) está motivado por la necesidad de disponer de datos concretos y realistas que determinen si es más beneficioso disponer de un equipo de desarrollo interno (*in-house*) o uno externo (*outsourcing*) para el desarrollo de aplicaciones.

Un equipo interno, *In-house*, se compone de personal técnico contratado directamente por la organización. Este grupo crea, mantiene y mejora las herramientas de software específico en la propia organización. Como valor aporta un amplio conocimiento de los procesos internos y mayor seguridad de la información.

Por otra parte, el equipo externo, o externalización (*outsourcing*), implica que una organización delega una parte de su actividad a terceros. Esto permite que la organización se pueda centrar y mejorar en ciertas funciones aumentando en eficacia y ahorrando tiempo.

Para lograr el objetivo, se determinan las siguientes líneas de actuación:

1.- Investigar la literatura existente sobre el desarrollo del software y el trabajo en equipo. Para ello, este TFM se ha apoyado en el estudio y análisis del estado del arte relativo a todos los elementos necesarios para el desarrollo de software.

2.- Asimismo, para recabar la información necesaria, se plantean las siguientes premisas:

- Indicadores de evaluación del ciclo de vida de los desarrollos propuestos, contemplando aspectos como el tiempo empleado, el coste, la duración estimada, la duración real en comparación con lo previsto, etc. El análisis de estos resultados pretende determinar el grado de evaluación del ciclo de vida de los proyectos como punto de partida para llevar a cabo una planificación integral durante todo el ciclo de vida del software (SDLC). La obtención, recopilación y evaluación de estos resultados han sido realizados a través de la herramienta Jira, en base al Sistema de Gestión de Calidad (SGC) implantado en el Centro de Informática de Gestión (CIGES) y de acuerdo con la normativa Publicación Española de Calidad (PECAL 2110¹ y 2210²).

- Indicadores que evalúen la calidad operativa del proyecto para el aseguramiento de la calidad. Con estas métricas se pretende reflejar aquellos parámetros como son: Funcionalidad, Seguridad, Disponibilidad, Fiabilidad, Usabilidad y Mantenibilidad, que son necesarios para la implementación del desarrollo del software desde todos los puntos de vista operativos del software. La obtención, recopilación y evaluación de estos indicadores han sido realizados a través de la herramienta Jira, en base al Sistema de Gestión de Calidad (SGC) implantado en el Centro de Informática de Gestión (CIGES) y de acuerdo con la normativa Publicación Española de Calidad (PECAL 2110 y 2210).

- Indicadores de calidad del código fuente. Estos indicadores pone en relieve la aplicación de buenas prácticas en el desarrollo de software ya que lo que se aspira es que el código fuente, con los siguientes parámetros (cobertura, comentarios, líneas duplicadas, deuda técnica y vulnerabilidades), ponga de manifiesto que está bien desarrollado, documentado y con buena cobertura. La obtención, recopilación y evaluación de estos indicadores han sido realizados a través de la herramienta SonarQube, en base al Sistema de Gestión de Calidad (SGC) implantado en el Centro de Informática de Gestión (CIGES) y de acuerdo con la normativa Publicación Española de Calidad (PECAL 2110 y 2210).

- Indicadores cualitativos. Se pretende ampliar estas comparativas dando valor a aspectos como el talento, la experiencia y el conocimiento que, aun no siendo cuantificables, son aspectos críticos para el éxito de un equipo de desarrolladores, pudiendo ser medidos cualitativamente según observaciones, entrevistas y evaluaciones. Se reseña, como ejemplo, que la capacidad de innovación o la adaptabilidad ante cambios tecnológicos puede influir enormemente en los cometidos del equipo desarrollador.

3.- La combinación de datos cuantitativos y cualitativos ofrecerá una visión integral sobre la necesidad de disponer de un equipo propio o la externalización en el desarrollo de aplicativos. La obtención, recopilación, integración y evaluación de las métricas cuantitativas han sido realizadas mediante la herramienta de Jira (gestión de proyectos) y de SonarQube (análisis de vulnerabilidades), mientras que las métricas cualitativas se ha llevado a cabo mediante un cuestionario que ha sido distribuido a gestores / analistas por cada uno de los proyectos indicados en los casos de uso.

4.- El principal objetivo de este TFM es informar a los Mandos para orientar la estrategia a seguir en la toma de decisiones, según las conclusiones derivadas de este estudio. Estas premisas orientarán el trabajo comparativo para la obtención de conclusiones significativas sobre la conveniencia o no de disponer de equipos de desarrollo propios.

¹ PECAL 2110: Requisitos OTAN de aseguramiento de la calidad para el diseño, el desarrollo y producción.

² PECAL 2210: Requisitos OTAN de aseguramiento de la calidad del Software, suplementarios a la PECAL 2110.

2 ESTADO DEL ARTE

2.1 Base de Conocimiento

Actualmente, muchas personas disponen de varios dispositivos tecnológicos como nunca antes se había visto. Estos dispositivos se han convertido en una gran herramienta de trabajo permitiendo realizar tareas repetitivas de manera sencilla y con poco esfuerzo, además de su empleo en nuestro ámbito personal como medios de comunicación y diversión.

A continuación, se expone información básica sobre los elementos que intervienen en el desarrollo de software:

2.1.1 Hardware

Si retrocedemos a 1960, hace su aparición uno de los primeros ordenadores con circuitos integrados, el cuál era capaz de realizar los cálculos necesarios de orientación y navegación necesarios para que el hombre pudiera llegar a la Luna en 1969. Esta hazaña representó un hito en la evolución en el desarrollo del hardware, como en los procesadores (la ley de Moore predice que el número de transistores se duplica aproximadamente cada dos años), la memoria principal, los dispositivos de entrada y salida, el almacenamiento, etc.

Como precursores de la evolución del hardware destaca a Alan Turing, cuyo trabajo en la teoría de la computación aportó las bases para los ordenadores modernos, y John Von Neumann, quien desarrolló la arquitectura física básica de la mayoría de los ordenadores actuales. [5]

La siguiente figura muestra, de manera gráfica, las generaciones de hardware:

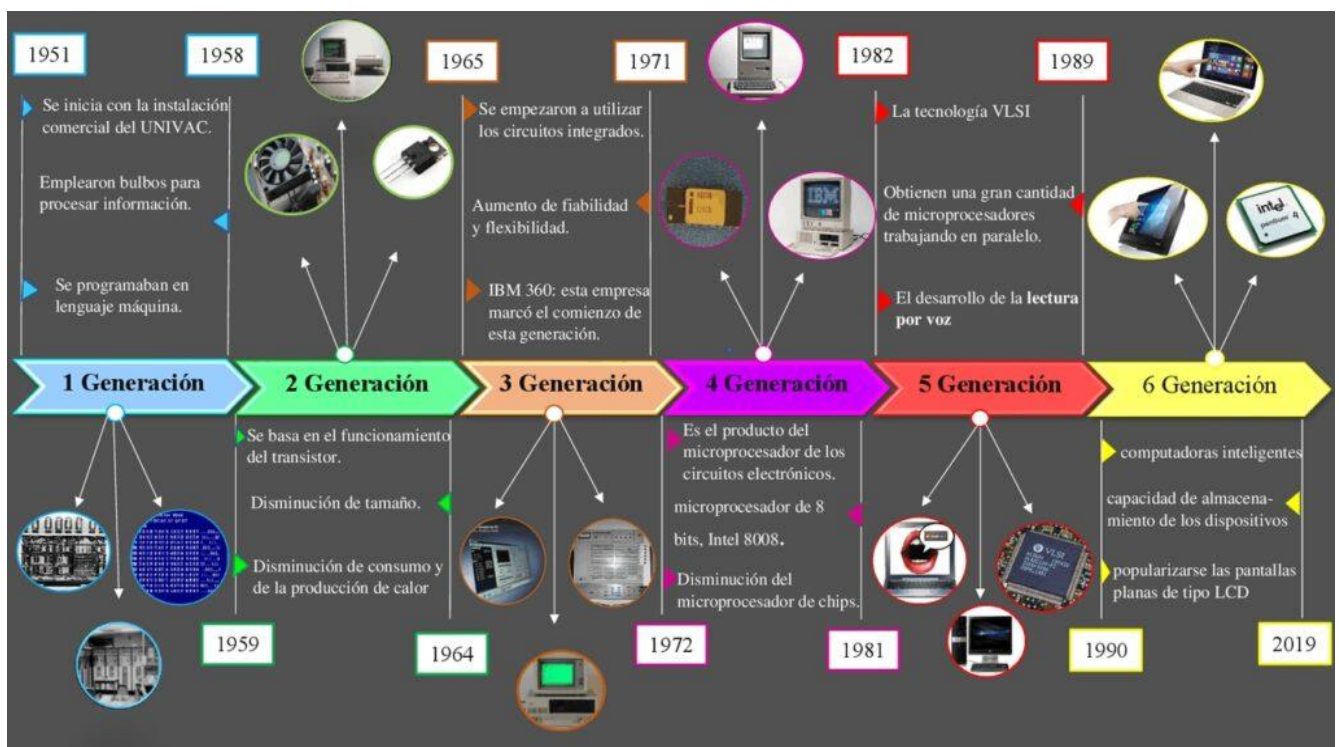


Figura 2-1 Línea de tiempo Hardware [6]

2.1.2 Software

Sin embargo, el hardware, la parte física de un sistema informático, debe ser completado por el software, ya que este dispone de las instrucciones para que un ordenador realice tareas complejas y específicas. Sin software, todos los dispositivos carecerían de utilidad.

A la par que el hardware, el software también experimentó una evolución en los lenguajes de programación, destacando en sus inicios los siguientes:

- FORTRAN (*FORmula TRANslation*): diseñado para cálculos numéricos, aplicaciones científicas e ingeniería por John Backus en 1957 para equipos IBM.

- COBOL (*COmmon Business-Oriented Language*): uso en aplicaciones comerciales y empresariales desarrollado por Grace Hopper en 1960 para en IBM.

En 1965, se produjo un punto crítico en el desarrollo de software. Esto era debido a los problemas derivados de la poca eficacia en el desarrollo de software y proyectos cada vez más difíciles de gestionar, que generaban como resultando la aparición de sobrecostos y retrasos. Este suceso se conoció como “la crisis del software”.

Para corregir estos problemas, surgió una nueva disciplina denominada “Ingeniería de Software” (1968), cuya metodología incluía la creación de software confiable y de calidad.

Aunque los lenguajes de programación siempre han tenido el componente de ser un lenguaje solamente entendible como lenguaje-maquina, este entorno ha evolucionado a una programación más comprensible para los humanos. Ejemplos de ello es el surgimiento de los nuevos lenguajes de programación como:

- C: lenguaje empleado para sistemas operativos. Fue diseñado por Dennis Ritchie entre 1969 y 1972 para Laboratorios Bell.

- BASIC (*Beginners' All-purpose Symbolic Instruction Code*): utilizado para facilitar la programación en ordenadores a estudiantes, fue desarrollado por John George Kemeny, Thomas Eugene Kurtz y Mary Kenneth Keller en 1964 para la Universidad Privada Dartmouth College (New Hampshire, EEUU).

- PASCAL: usado para enseñar en la universidad a los estudiantes de programación en la Universidad Técnica de Zúrich (Suiza), fue desarrollado por Niklaus Wirth en 1970.

Estos lenguajes evolucionaron hacia los lenguajes de programación orientados a objetos, que actualmente conocemos como:

- C++: es un lenguaje de programación compilado, multiparadigma, desarrollado por Bjarne Stroustrup en 1980 y empleado en Laboratorios AT&T.

- JAVA: es un lenguaje de programación orientado a objetos, portable entre diferentes sistemas operativos y relativamente fácil de usar. Fue creado por James Gosling en 1995 para Sun Microsystems.

- Python: es un lenguaje muy utilizado en aplicaciones web, desarrollo de software, BIG DATA y *Machine Learning* (ML). Fue desarrollado por Guido Van Rossum en 1991 y empleado en el Centrum Wiskunde & Informatica (CWI) de Ámsterdam (Países Bajos). [7]

La siguiente figura, muestra los lenguajes de programación conforme a su evolución:



Figura 2-2 Línea de tiempo de Lenguajes de programación [8]

2.1.3 Tipos de Software

Hasta ahora, se han mencionado principalmente los lenguajes de programación como parte de software. Sin embargo, dentro del software existen diferentes criterios de clasificación dependiendo de su función, licencias o uso, pudiendo algunas aplicaciones pertenecer a varios tipos.

A continuación, se enumeran los tipos de software:

2.1.3.1 Según su función

La característica de este tipo de software es cómo interactúa el software con el hardware y el usuario y qué tipos de tareas puede ejecutar.

A continuación, se presenta su clasificación:

- Software de Sistema: se encarga de administrar los recursos del sistema, como sistemas operativos, controladores de dispositivo, firmware, entre otros. Su objetivo es proporcionar funcionalidad a los usuarios y a otros programas.

- Software de Programación: estos programas disponen de herramientas y librerías para el desarrollo de software de aplicación, para el desarrollo del código fuente (GitHub, Eclipse, Visual Studio Code).

- Software de Aplicación: programas usado por los usuarios como navegadores web, suites ofimáticas, entre otros. [9]

La figura siguiente muestra la clasificación el software en función de las tareas que desempeñan:

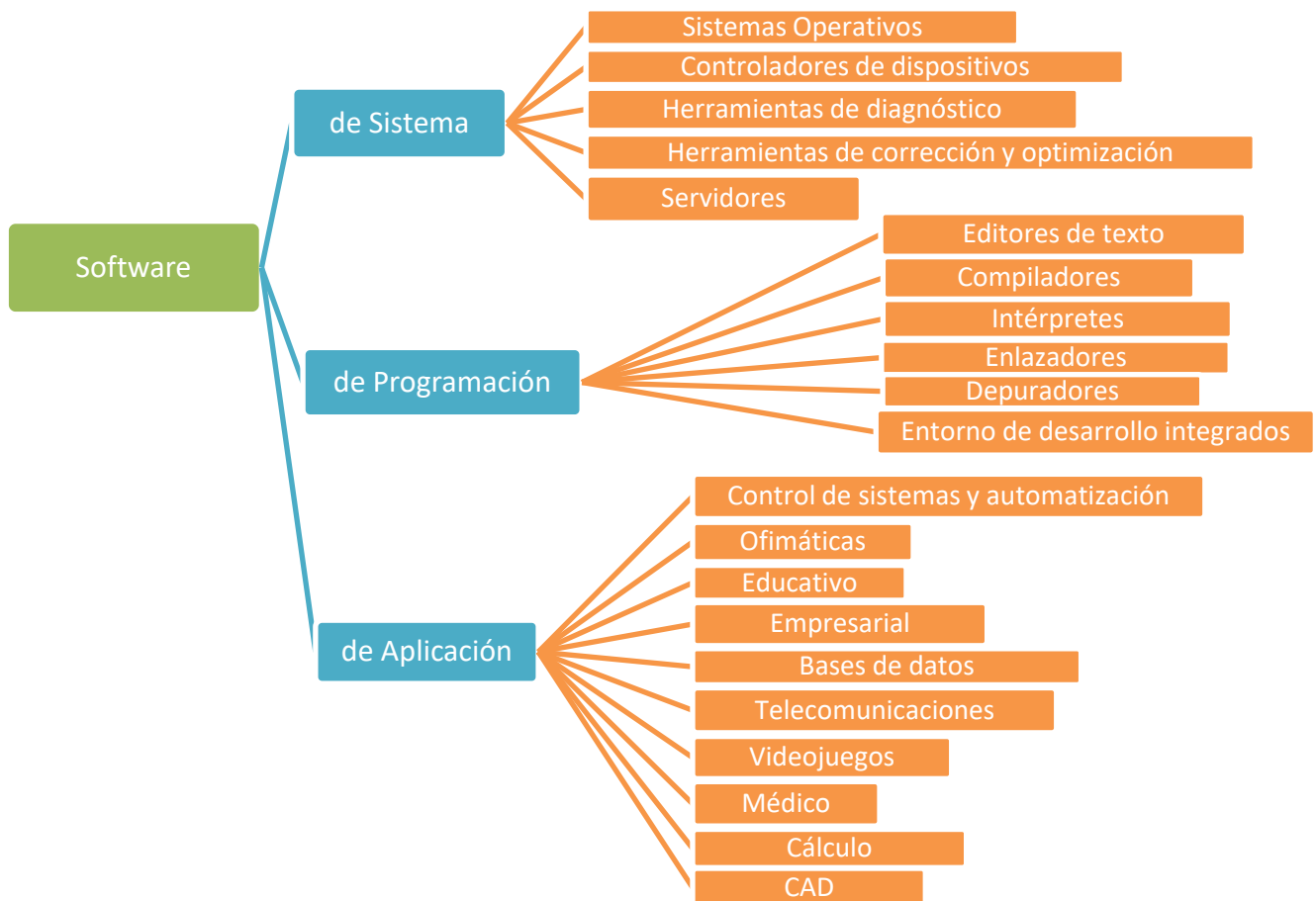


Figura 2-3 Clasificación del software según su función (elaboración propia)

2.1.3.2 Según su licencia

Se refiere a las condiciones y término en los que el software puede ser utilizado, distribuido y modificado.

A continuación, se enumeran la clasificación según su licencia:

- Software de Código Abierto: el código fuente se encuentra disponible para el público, pudiendo ser modificado y distribuido (Linux, Apache).
- Software Propietario: software que está desarrollado y distribuido bajo licencia. No se puede acceder a su código fuente (Microsoft Office, Adobe Photoshop).
- *Freeware*: software gratuito donde el creador mantiene los derechos de autor (Skype, GIMP).
- *Shareware*: software gratuito durante un período de prueba. Tras finaliza, se debe pagar para continuar utilizándolo (WinRAR). [\[9\]](#)

La siguiente figura representa los tipos de licencias de software y las diferentes variantes que existen acorde al tipo de software licenciado:

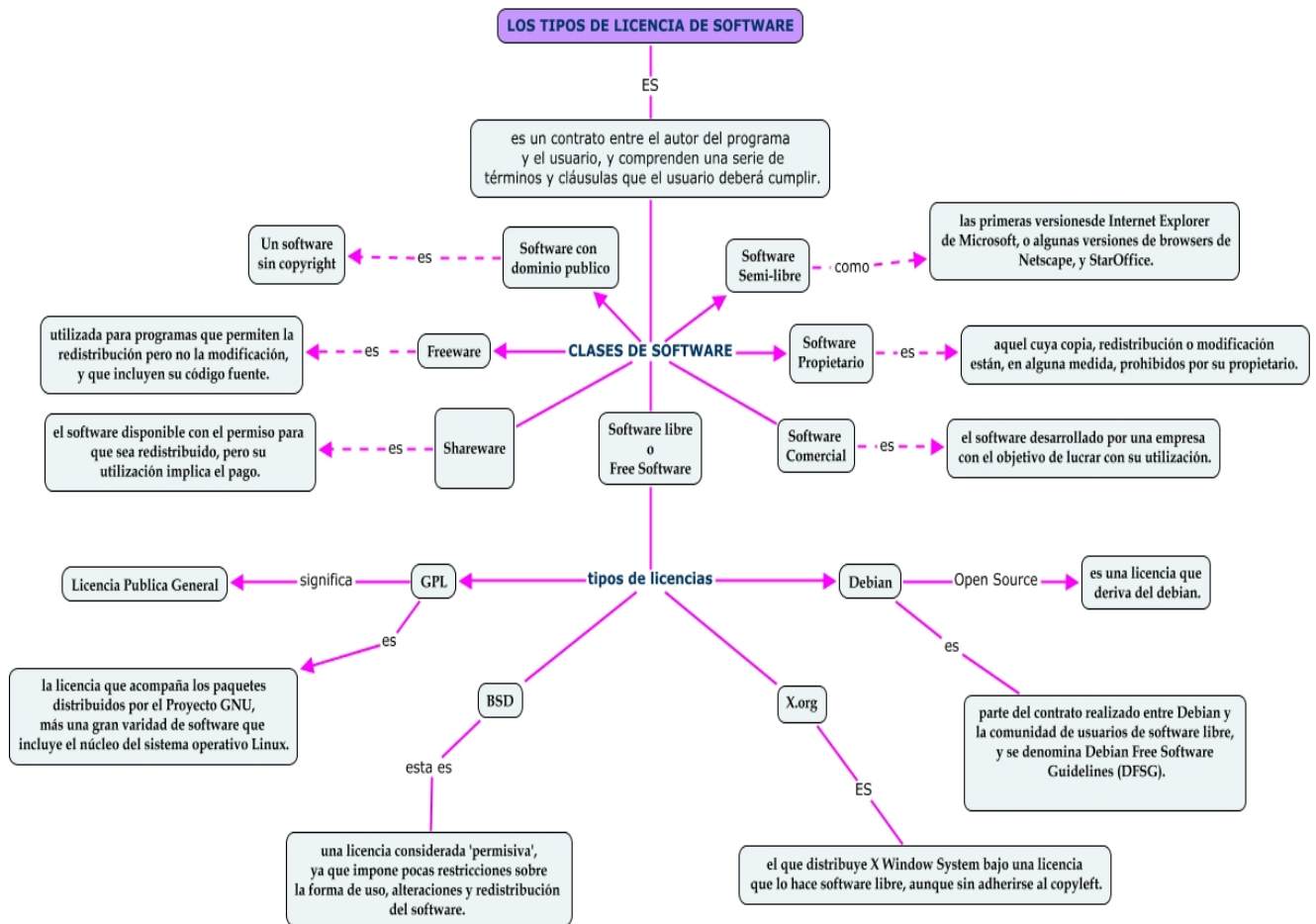


Figura 2-4 Clasificación software según su licencia [10]

2.1.3.3 Según su uso

Se denomina software de uso en referencia al contexto o contorno en el que se utiliza, lo que puede influir en sus funcionalidades. Las categorías más comunes son:

- Software de Productividad: desarrollado para tareas productivas (Google Workspace, Trello).
- Software de Entretenimiento: creado para ocio y entretenimiento (VLC, Spotify).
- Software Educativo: utilizado para la enseñanza (Moodle, Duolingo).
- Software Empresarial: utilizado para la gestión y operaciones empresariales (Oracle ERP, HubSpot).
- Software Científico y de Ingeniería: creado para investigación y análisis científicos e ingeniería (MatLab, AutoCAD). [9]

2.1.3.4 Según su plataforma

Esta clasificación se refiere al entorno donde el software puede ser ejecutado. Las categorías principales son:

- Software de Escritorio: software instalado en ordenadores de sobremesa o portátiles (Microsoft Office, Adobe Photoshop).

- Software Web: software al que se puede acceder a través de un navegador web sin requerir instalación en local (Google Docs).

- Software Móvil: diseñado para dispositivos móviles como tablets y smartphones (WhatsApp, Twitter).

- Software de Servidores: el uso de este software es proporcionar servicios a dispositivos de red y a otros ordenadores como MySQL, Postgre SQL, etc. [\[9\]](#)

Llegados a este punto y valorando lo que supone el desarrollo de software a nivel mundial, las estadísticas reflejan un crecimiento exponencial que alcanzarán los 305.000 millones de dólares para 2025. Este crecimiento presenta diversos retos, entre los cuales destacan:

- Tecnologías 5G: su uso empieza a estar muy extendido en sectores como la sanidad, la logística y la automatización. Su principal objetivo es el desarrollo de aplicaciones más rápidas, escalables y con mayor capacidad de procesamiento de datos.

- Internet de las Cosas (*Internet of Things*): proporciona el desarrollo de aplicaciones para conectar todo tipo de dispositivos por medio de sensores. Se utiliza en áreas como el hogar inteligente, agricultura e industria.

- Realidad Aumentada (AR) y Realidad Virtual (VR): su uso es mayoritariamente en simuladores y entretenimiento. Estas tecnologías proporcionan un entorno inmersivo donde se mezcla la realidad con experiencias virtuales.

- *Blockchain*: su uso permite garantizar la seguridad, trazabilidad y transparencia en las transacciones y procesos empresariales.

- Inteligencia Artificial (IA) y Aprendizaje Automático (ML): estos conceptos permiten funcionalidades diversas, como *chatbots*, reconocimiento de imágenes y detección de rostros.

- *Chatbots*: son aplicaciones destinadas a proporcionar ayuda a usuarios basadas en inteligencia artificial.

- Wearables: dispositivos que ofrecen datos biométricos de usuarios ofreciendo información adicional al estado de salud.

- Monedero móvil y *Contactless*: aplicaciones que integran funcionalidades de pago móvil, facilitando transacciones financieras.

- Aplicaciones mediante apps y móvil: aplicaciones que integran funcionalidades para el aprendizaje.

- *Low-code / No-code*: aunque ambos términos difieren en el desarrollo del software, el objetivo es la creación de software con un mínimo de programación manual y sin necesidad de disponer de muchos conocimientos de programación.

La siguiente figura representa los ingresos generados en el desarrollo de software, integrando las siguientes ramas; software de productividad, software empresarial, software de infraestructuras de sistemas y software de desarrollo de aplicaciones.

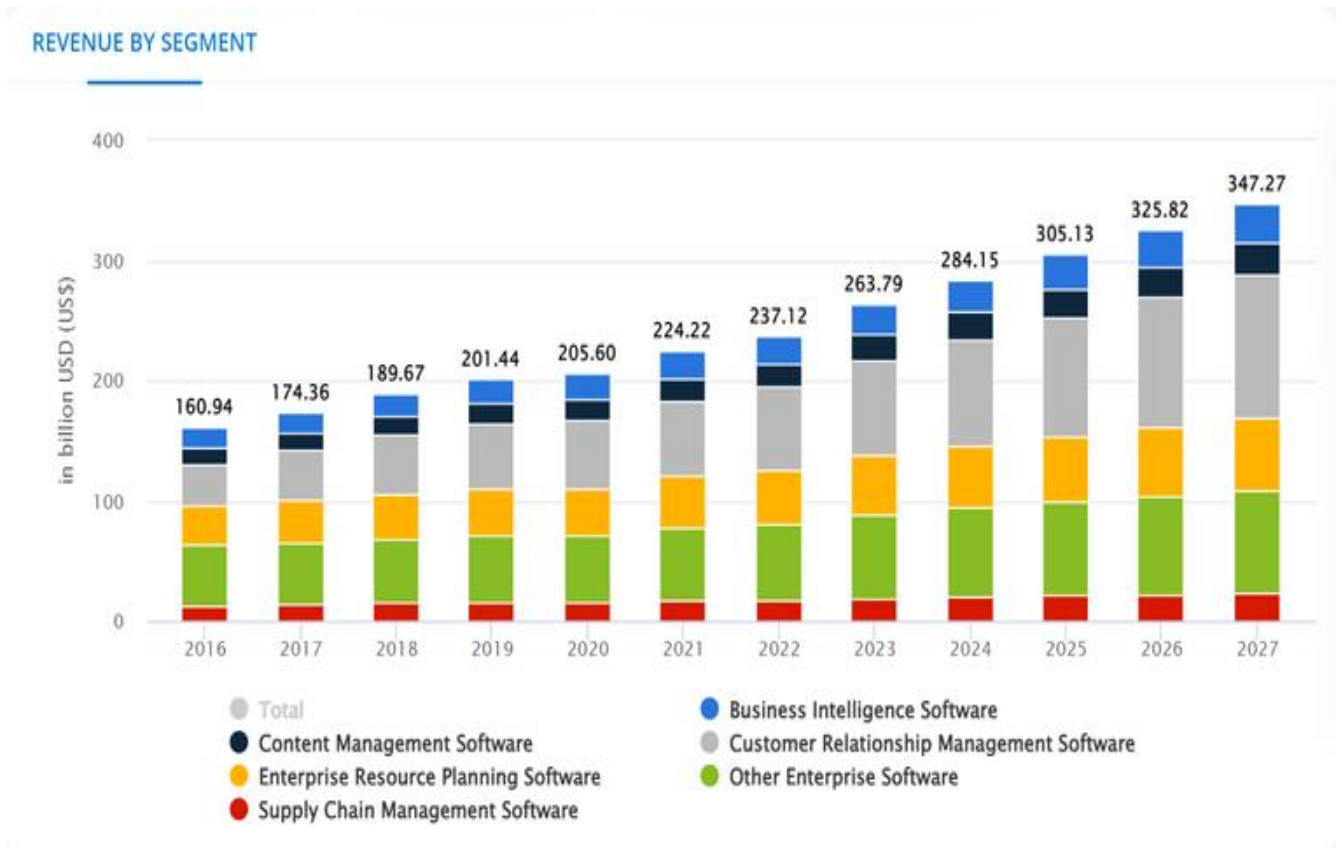


Figura 2-5 Ingresos generados por software [11]

2.1.4 Requisitos para el Desarrollo de Software

En este contexto, el desarrollo de aplicaciones se ha convertido en un pilar esencial tanto para usuarios como para organizaciones y empresas ya que aporta valor como resultado de su eficiencia, adaptación a los cambios que se producen, acceso a la información en cualquier momento y lugar, innovación, ...

2.1.4.1 Perfiles Necesarios

Dentro de los requisitos necesarios para desarrollar software se encuentra un activo esencial: un equipo técnico con perfiles diversos que realizan el desarrollo del software.

A continuación, se exponen los perfiles junto con las responsabilidades y habilidades necesarias para el desarrollo de software:

Perfil	Responsabilidades	Habilidades
Director de Proyecto	Planifica y organiza el proyecto y define los objetivos. Coordina recursos. Evalúa y gestiona los riesgos y las posibles amenazas. Control económico del proyecto, asegurando su cumplimiento. Supervisa la calidad del desarrollo.	Experiencia en gestión de proyectos y conocimiento de metodologías de desarrollo de software. Habilidades comunicativas. Liderazgo. Persona analítica, proactiva, con iniciativa y adaptación al cambio.

Jefe de Proyecto	Desarrolla un plan detallado para el proyecto. Asigna tareas y responsabilidades. Asegura y gestiona la disponibilidad de los recursos necesarios. Monitoriza el progreso del proyecto y toma medidas para su cumplimiento.	Habilidades técnicas. Planificación y organización. Liderazgo. Trabajo en equipo, comunicación y colaboración. Resolución de problemas. Negociación. Adaptabilidad y flexibilidad. Resolución de problemas y gestión de riesgos.
Gestor de Proyecto (Project Manager)	Planifica, programa, supervisa y controla el proyecto de principio a fin. Responsable de conducir el proyecto y a todo el equipo de forma efectiva. Responsable de documentar todos los aspectos del proyecto. Informa del progreso del proyecto y de cualquier problema que surja. Planifica los recursos. Gestiona el tiempo y la programación del proyecto.	Experiencia en gestión de proyectos y conocimiento de metodologías de desarrollo de software. Amplio conocimiento técnico. Responsables de la calidad y entrega del producto. Actitud positiva y orientada a los resultados. Liderazgo. Habilidades de comunicación y negociación. Detecta problemas que surgen y encuentra la manera de resolverlos.
Product Owner (Metodología Ágil)	Responsable del producto final. Responsable de la coordinación entre otras las partes intervinientes (equipo de Scrum, los usuarios finales, etc.). Responsable de alcanzar las funcionalidades deseadas.	Conocimiento de metodologías de desarrollo de software (metodologías ágiles). Liderazgo. Habilidades de comunicación y organización. Persona con visión estratégica, analítica, proactiva, con iniciativa y adaptación al cambio. Resolución de problemas.
Scrum Master (Metodología Ágil)	Responsable de establecer la metodología Scrum y mantener a los miembros del equipo en sus principios y prácticas de Scrum. Asistir al dueño del producto incorporando técnicas para la mejora de procesos.	Domino de la metodología Scrum. Conocimientos técnicos. Liderazgo. Habilidades comunicativas. Orientación al equipo y resolución de conflictos.
Analista de Proyecto	Analiza el proyecto para identificar mejoras, problemas y oportunidades de crecimiento. Evalúa la viabilidad del proyecto. Encargado de trasladar al equipo	Conocimiento de las mejores prácticas de desarrollo de software. Conocimiento de bases de datos. Persona analítica con iniciativa y visión de futuro.

	desarrollador las necesidades del cliente. Analiza el impacto de nuevas funcionalidades en un proyecto. Responsable de la documentación del proyecto.	Capacidad de organización. Toma de decisiones. Resolución de problemas.
Business Analyst (Metodología Ágil)	Colabora con el <i>Product Owner</i> para la alineación de los objetivos	Habilidades de análisis de requisitos. Análisis de datos y negocio. Persigue la mejora continua.
Arquitecto de Software	Define la arquitectura del sistema, componentes, interfaces y comunicación entre ellos. Diseñar y documentar los modelos y estándares de desarrollo que se utilizarán en el proyecto. Responsable de examinar y comprobar el código fuente para garantizar su calidad. Identificar y mitigar los riesgos técnicos y las posibles limitaciones del sistema.	Amplios conocimientos técnicos (diseño, programación, arquitectura de software y aplicaciones). Capacidad de liderazgo y dotes organizativas. Resolución de problemas. Habilidades de comunicación. Actualización continua de conocimientos.
Desarrollador Front-End³	Responsable de la relación del aplicativo con los usuarios. Desarrolla la interfaz del usuario. Mantiene la funcionalidad, accesibilidad y estética con los usuarios.	Conocimiento de diseño y lenguajes de programación. Habilidad para crear interfaces responsivas y accesibles. Creatividad. Resolución de problemas.
Desarrollador Back-End⁴	Diseñar, crear y mantener las bases de datos y servidores. Codifica y mantiene el código del lado del servidor. Mantiene que diferentes aplicaciones se comuniquen entre sí a través de interfaz (APIs⁵) Responsable de la eficiencia y rendimiento.	Amplios conocimientos técnicos (lenguajes de programación, en diseño, bases de datos y servidores). Capacidad de comunicación y trabajo en equipo. Resolución de problemas. Adaptable y creativo.

³ *Front-end*: capa que se encuentra por encima del back-end, que incluye menús desplegables, imágenes, elementos gráficos, etc. En resumen, es la interfaz para que el usuario interactúe con el software.

⁴ *Back-end*: capa compuesta por lenguajes de programación que determinan la lógica del software. En esta capa los usuarios no tienen acceso.

⁵ APIs, *Application Programming Interface*: pieza de código que permite a diferentes aplicaciones comunicarse entre sí y compartir información y funcionalidades, sin necesidad de saber cómo están implementados.

	Actualiza, comprueba y corrige el código fuente. También implementa nuevas funcionalidades.	
Programador Senior	Diseña y desarrolla código complejo. Prioriza, asigna y ejecuta tareas de codificación. Asegura que el código cumpla con los estándares de calidad y mejores prácticas. Ejecuta pruebas funcionales y depura código cuando detecta un error. Guía y supervisa a programadores junior.	Conocimiento avanzado del desarrollo del software y sus tecnologías. Experiencia y dominio de lenguajes de programación. Buena capacidad de trabajo en grupo, comunicación y colaboración. Habilidades para resolver problemas.
Programador Junior	Responsable de escribir y mantener el código. Corrige errores menores de código. Participa en la ejecución y documentación de las pruebas.	Conocimiento básico de lenguajes de programación, bbdd y SO. Capacidad de aprender nuevas tecnologías. Capacidad de trabajar en equipo y resolución de problemas.
Diseñador UX⁶/UI⁷ (específico para desarrollo web)	Diseño de componentes UI, garantizando la usabilidad según preferencias y necesidades de los usuarios. Analizar los requisitos y especificaciones de diseño, resolver problemas de UX e idear soluciones. Realizar análisis de métricas del producto para optimizar su diseño.	Conocimientos de diseño. Capacidad analítica y creativa. Empatía y conocimiento del usuario. Trabajo en equipo. Pensamiento crítico y resolución de problemas. Adaptabilidad, comunicación y presentación.
Técnico en Seguridad	Responsable de diseñar, implantar y evaluar soluciones de seguridad al software. Identifica y mitiga vulnerabilidades.	Conocimientos técnicos avanzados (pruebas de penetración y análisis forense). Capacidad de resolución de problemas. Capacidad de trabajo en equipo.
Tester de Calidad (QA)	Responsable de llevar a cabo planes de pruebas y casos de uso para evaluar el software. Identificar y documentar los defectos y problemas de software.	Conocimiento en metodologías, herramientas y prueba de software (unitarias, integración, etc.). Gran capacidad analítica y resolución de problemas.

⁶ UI, *User Interface*, Interfaz de Usuario: la idea de UI es guiar al usuario por la aplicación durante el tiempo que se está usando, encargándose del aspecto visual de la interfaz. [\[12\]](#)

⁷ UX, *User Experience*, Experiencia de Usuario: término que determina cómo se siente una persona mientras usa cualquier producto y la sensación después de haberlo usado, garantizando que la navegación y acciones que realiza el usuario estén en consonancia. [\[12\]](#)

	Optimiza procesos y metodologías de prueba. Realiza pruebas de rendimiento, evaluando su respuesta, estabilidad y escalabilidad.	Capacidad de comunicación y trabajo en equipo.
Ingeniero DevOps⁸	Responsable de configurar y definir las herramientas de desarrollo, de pruebas y de automatización. Dispone de técnicas de solución de problemas y corrección de errores de código. Trabaja en formas de automatizar y mejorar los procesos de desarrollo.	Conocimiento avanzado del desarrollo de software (CI/CD⁹). Conocimiento de lenguajes de programación y en administración de sistemas. Trabajar en equipo y comunicación.
Administrador de Bases de Datos (DBA)	Diseña, mantiene e implementa las bases de datos. Responsable de la seguridad de las bases de datos. Garantiza los datos almacenados y su recuperación.	Amplios conocimientos en la administración, lenguajes de consulta y tecnologías de bases de datos. Habilidad para resolución de problemas. Trabajar en equipo y comunicación.

Tabla 2-1 Perfiles necesarios para desarrollar software (elaboración propia) [14]

2.1.4.2 Ciclo de vida del Desarrollo de Software (SDLC)

El ciclo de vida del desarrollo de software (SDLC, *Systems Development Life Cycle*) está definido por un marco de referencia que describe la vida de un aplicativo desde su inicio hasta su finalización. Este ciclo de vida está dividido en fases donde están incluidos los procesos, actividades y tareas. El objetivo del SDLC es proporcionar una mayor visibilidad del proyecto a las partes implicadas, permitiendo una estimación y una planificación eficiente. Además, deberán cumplir con las expectativas y funcionalidades requeridas por el cliente, minimizando los riesgos que pudieran producirse.

Dependiendo del enfoque utilizado, el ciclo de vida de un desarrollo consta de 6 o 7 fases. La siguiente figura representa las fases del SDLC, con los puntos más importantes para cada una de ellas. Posteriormente, se expondrá cada una de las fases con sus características más significativas.

⁸ DevOps: término acuñado de la combinación de desarrollo (DEvelopment) y operaciones (OPerationS). Permite alinear diferentes equipos que no son de desarrollo pero que tiene un impacto significativo en el desarrollo de aplicaciones (administradores de sistemas, equipos de seguridad, ...). [13]

⁹ Herramientas CI/CD: herramientas para la integración y despliegue continuo basado en mejorar, automatizar y agilizar el ciclo de vida de desarrollo del software (SDLC).

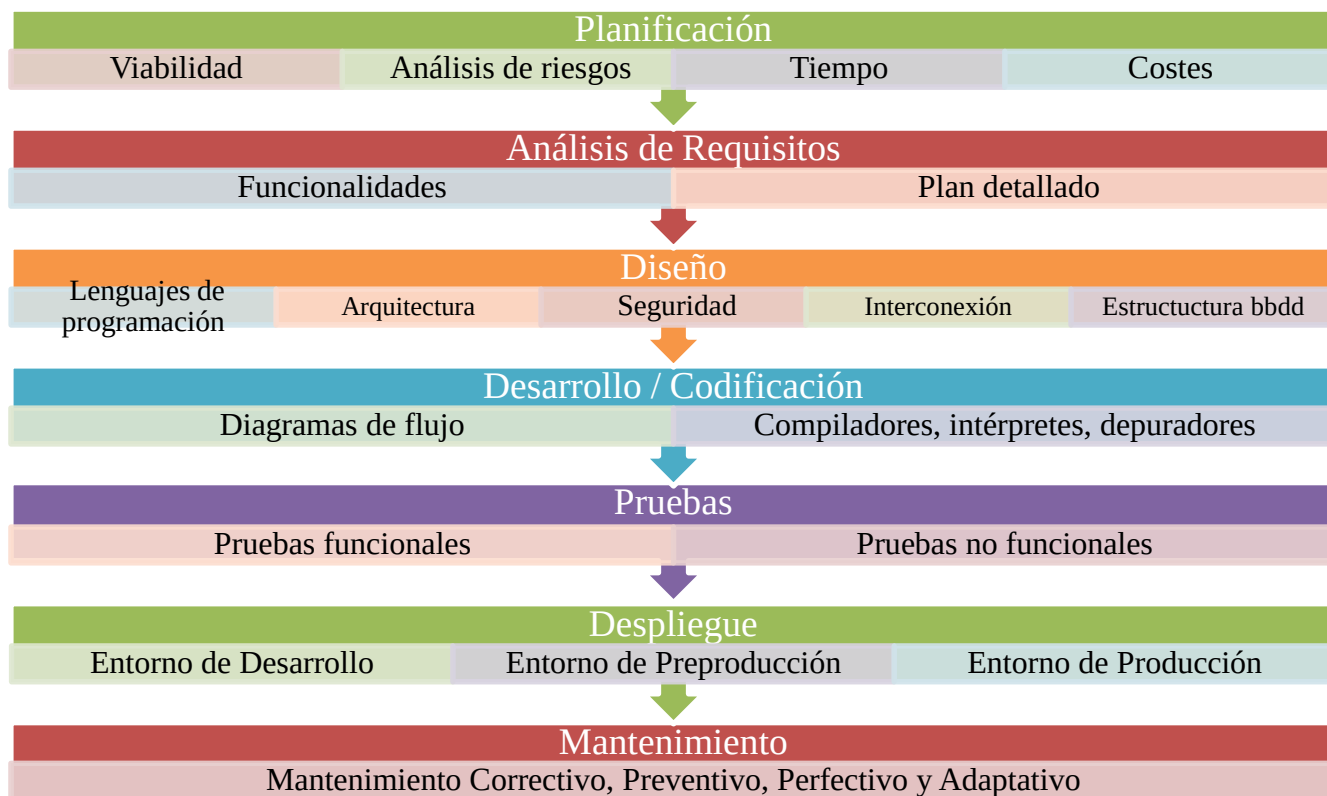


Figura 2-6 Fases del SDLC (elaboración propia)

Mediante la siguiente tabla y gráfico se muestran datos sobre la estimación de tiempo empleado en el SDLC, los roles necesarios, los porcentajes de participación y la tendencia por fase.

Para su interpretación, se observa que la fase de desarrollo y pruebas son las fases que necesitan más tiempo para llevarse a cabo. A su vez, la fase de mantenimiento presenta la mayor variabilidad puesto que la cantidad de recursos y esfuerzos necesarios pueden variar según las circunstancias del proyecto.

Por otro lado, la columna “Tendencia” representa el valor medio de participación de los diferentes roles involucrados en cada fase del proyecto. Esto traducido a valores, sugiere que a mayor valor podría existir una mayor probabilidad de que el porcentaje medio se incremente, o sea, determina cuánta fluctuación o cambio se puede esperar en el porcentaje de participación en cada fase del proyecto.

$$Tendencia = \frac{(\text{valor medio} - \text{porcentaje mínimo})}{(\text{porcentaje máximo} - \text{porcentaje mínimo})}$$

Fase	Roles involucrados	Porcentaje mínimo	Porcentaje máximo	Valor medio	Tendencia
Planificación	Director de Proyecto, Jefe de Proyecto, Gestor de Proyecto.	10	15	12,5	0,5
Análisis	Gestor de Proyecto, Analista de Proyecto.	20	25	22,5	0,25
Diseño	Analista de Proyecto, Programador <i>Frontend</i> , Programador <i>Backend</i> .	15	20	17,5	0,33

Desarrollo	Programador Senior, Programador Junior, Programador <i>Backend</i> .	30	40	35	0,33
Pruebas	Tester de Calidad (QA), Programador <i>Backend</i> .	20	25	22,5	0,25
Despliegue	Ingeniero <i>DevOps</i> , Programador <i>Backend</i> .	5	10	7,5	1
Mantenimiento	Ingeniero <i>DevOps</i> , Programador Senior, Tester de Calidad (QA).	0	100	50	1

Tabla 2-2 Estimación de tiempo empleado y perfiles que intervienen en las fases del SDLC [15]

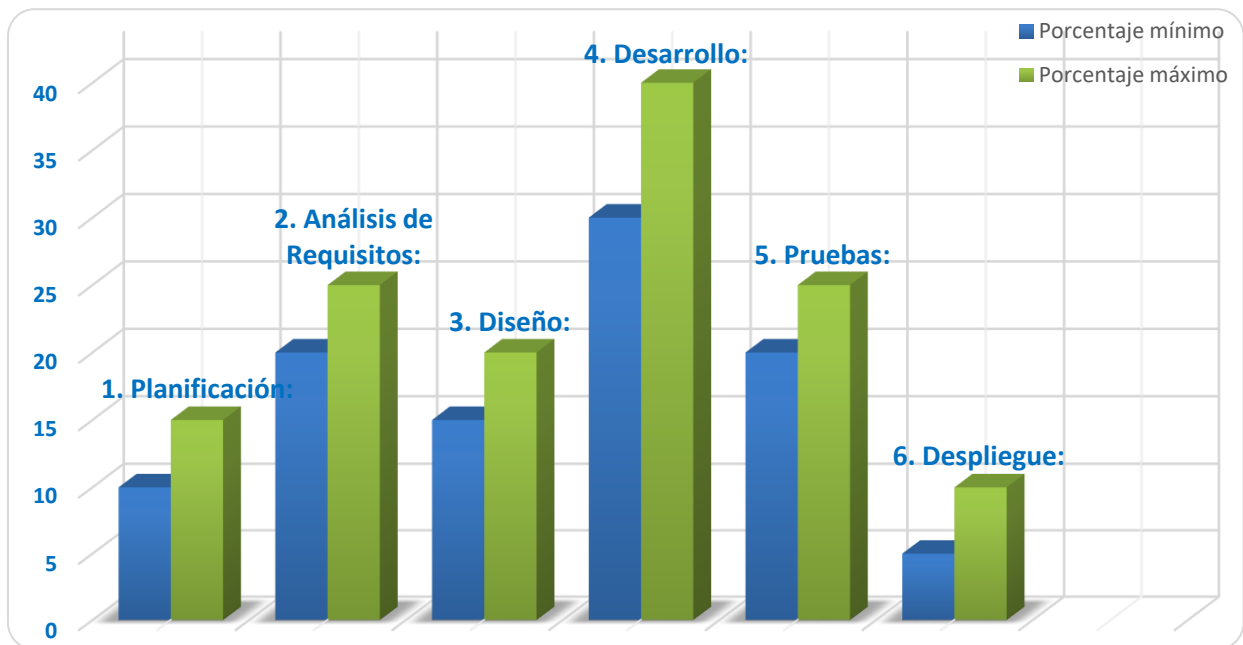


Figura 2-7 Estimación de tiempo en las fases del SDLC (elaboración propia)

A continuación, se amplía las características principales de cada una de las fases del SDLC:

- Planificación

En esta fase, los responsables del proyecto establecen los objetivos mediante:

- Estudio de viabilidad: en este estudio se incluyen aspectos técnicos, financieros y legales acorde a los recursos disponibles.
- Análisis de riesgos: al inicio del proyecto, se estima la probabilidad de riesgo de cada activo y el impacto que puede tener en la organización.
- En el inicio de un proyecto también se determina la estimación del tiempo y costes, una programación de las diferentes fases y los perfiles técnicos que intervienen.

- Análisis de Requisitos

Esta etapa se caracteriza por la recopilación y definición de los requisitos específicos de cliente para determinar las funcionalidades necesarias. En esta fase, el objetivo es la optimización de costes, el

cálculo del tiempo en su realización y la planificación de cómo se va desarrollar. Debe existir una estrecha colaboración entre el cliente y el equipo desarrollador para la comprensión de las necesidades, expectativas y funcionalidades.

- **Diseño**

Una vez se dispone de información precisa sobre las funcionalidades que debe tener el proyecto, se estudian diferentes alternativas (los lenguajes de programación, la arquitectura, características de seguridad a emplear, la interconexión con otros sistemas, etc.), para el desarrollo, la estructura de la base de datos, la lógica del negocio y la interfaz del usuario. El objetivo es definir cómo se diseñará, estructurará y visualizará el aplicativo.

- **Desarrollo / Codificación**

En esta fase se escribe el código fuente según las especificaciones del diseño. Para facilitar la distribución entre el grupo desarrollador, se utilizan los diagramas de flujo que garantizan la organización del proyecto. Además, se utilizan una variedad de herramientas como compiladores, intérpretes y depuradores. Así mismo, el objetivo final es disponer de un producto mínimo viable (PMV) para su implementación.

Este PMV es la versión de prueba de un nuevo producto que incluye las funcionalidades básicas necesarias para el cliente. Los requisitos que deben cumplir un PMV son, el diseño, la usabilidad, la funcionalidad y la fiabilidad.

- **Pruebas**

En esta fase determina si cumple con los requisitos del cliente y se comprueba, por medio de test, si el software contiene defectos, carencias, riesgos y errores.

Entre los diferentes test, se distinguen:

- Pruebas Funcionales

Se utilizan para garantizar que las características y funcionalidades del software se comportan según lo esperado. Entre estas pruebas están:

- Test unitarios: consistentes en probar métodos y funciones individuales de las clases, componentes o módulos que usa el software, comprobando que los parámetros que devuelven son los correctos.

- Test de integración: en estas pruebas se verifican que los módulos o servicios mencionados anteriormente funcionan coordinadamente.

- Test de sistemas / funcionales: estos test son realizados en el entorno de preproducción con el código fuente compilado en una versión estable para comprobar las especificaciones del software, el nivel de seguridad del sistema, la resiliencia y la recuperación ante situaciones anormales.

- Test de validación: pruebas realizadas para comprobar el aseguramiento de la calidad del producto. Normalmente estas pruebas son realizadas por el usuario y por personal técnico ajeno al equipo desarrollador.

- Pruebas No Funcionales

Estas pruebas permiten mejorar la experiencia del usuario que incluyen:

- Pruebas de rendimiento: estas pruebas valoran el rendimiento del software, los tiempos de respuesta y la carga de trabajo. Con ello se consigue identificar los intervalos de cuellos de botellas y picos de tráfico.

- Pruebas de carga: comprueba el rendimiento de software cuando existe una alta carga de trabajo.
- Pruebas de seguridad: estas pruebas tienen como objetivo que el software esté libre de amenazas y vulnerabilidades.
- Prueba de compatibilidad: evalúa el software en relación a si es adaptable a otros sistemas, dispositivos, etc.
- Prueba de confiabilidad: comprueban si para una tarea específica y un período determinado, la capacidad del software.
- Prueba de usabilidad: comprueba si el usuario puede acceder con facilidad, si el software es amigable y funcional.

Estas pruebas hacen que el software sea estable. Como resultado se compila una nueva versión para despliegue. Es un ciclo continuo por lo que todo se prueba hasta que las funcionalidades son viables. Las pruebas normalmente son realizadas por testers y por equipos para aseguramiento de la calidad (QA). [\[16\]](#)

2.1.4.3 Entornos de Despliegue

Llegados a este punto, el equipo desarrollador determina cuándo se pondrá en producción y cuándo se entregará al usuario final. Para ello se evalúa la integración, la portabilidad, la integración, la adaptabilidad en el nuevo entorno de despliegue. Para llevarlo a cabo se requiere de una buena coordinación y una colaboración de técnicos especializados en cada una de las áreas para el despliegue con éxito. Como buenas prácticas, cada entorno debe disponer de un análisis de vulnerabilidades para evitar accesos no autorizados, modificación o eliminación de carpetas o documentos, etc.

Para el desarrollo de aplicaciones existen varios entornos diferenciados:

- Entorno de Desarrollo

Este es el entorno donde el grupo de desarrollo emplea software específico y librerías para la elaboración del código fuente, el cual facilita la iniciativa, experimentación e iteración dentro del equipo desarrollador. El objetivo principal es reducir las incidencias que puedan producirse para cuando el software sea migrado a entornos posteriores. Tiene la ventaja que permite interactuar desde el inicio del desarrollo entre clientes y desarrolladores.

Entre las características que debe disponer este entorno, destacan:

- Software de Desarrollo: el uso de herramientas en este entorno simula todo el ecosistema necesario para el desarrollo. Para ello, el uso de entornos de desarrollo integrados (IDEs, ayuda a los programadores a desarrollar el código fuente de manera eficiente), los sistemas de control de versiones y los compiladores (convierten el código fuente en código máquina).
- Librerías y Dependencias: uno de los elementos más importantes es disponer de librerías y dependencias para el correcto funcionamiento del código fuente. Esto incluye frameworks (marcos de trabajo) y paquetes de software. Estos elementos ofrecen componentes y funcionalidades predefinidas, con código reutilizable y la integración para que el software funcione correctamente.
- Simulación de entornos externos: el entorno de desarrollo puede disponer simulaciones o entornos de prueba externos para comprobar y verificar servicios externos como bases de datos, APIs. Con ello se consigue probar el código en escenarios que imitan un escenario real.
- Configuración flexible: este entorno suele ser muy configurable debido a que tiene que adaptarse a las necesidades específicas de cada desarrollador. Para ello también puede incluir configuraciones personalizadas de herramientas, scripts de automatización y ajustes del IDE. [\[16\]](#)

Así mismo, disponer de un entorno de desarrollo tiene las siguientes ventajas e inconvenientes:

Ventajas	Inconvenientes
Iteración rápida: los desarrolladores pueden resolver rápidamente los problemas que surgen ya que pueden probar y modificar el código rápidamente.	Diferencia con entornos posteriores: las configuraciones de los diferentes entornos pueden ser diferentes lo que puede dificultar el despliegue del desarrollo.
Colaboración: las herramientas expuestas proporcionan que los equipos trabajen colaborativamente y de forma efectiva.	
Pruebas iniciales: antes de su puesta en producción se realizan pruebas unitarias, de integración y de sistema para determinar si existen errores y corregirlos.	Riesgo de fragmentación: debido a que múltiples desarrolladores están trabajando a la vez en diferentes módulos puede existir el riesgo de fragmentación del código.
Prototipo: el entorno de desarrollo permite desarrollar y probar modificaciones en los proyectos modificados sin afectar al código que se encuentra desplegado en el entorno de producción.	

Tabla 2-3 Ventajas e Inconvenientes del entorno de desarrollo (elaboración propia) [\[17\]](#)

○ Entorno de Preproducción (pruebas)

El entorno de preproducción, conocido también como *staging*, es un entorno creado a imagen y semejanza del entorno de producción. Con ello se persigue que, al disponer de la misma configuración, no se produzcan fallos una vez sea desplegado en producción.

Como características del entorno de preproducción, se menciona:

- Sin cambios frecuentes: este entorno, al estar separado del entorno de desarrollo, donde los programadores realizan el trabajo diario, no suele haber cambios frecuentes en el código, por lo que mantiene la estabilidad para la realización de pruebas.
- Pruebas de integración completa: como el entorno de preproducción es un entorno de prueba, se realizan las pruebas necesarias (pruebas de integración, pruebas de rendimiento, pruebas de carga, pruebas de seguridad y pruebas de usabilidad) para identificar errores.
- Concurrencia de usuarios: en ese entorno también se prueba la concurrencia, comportamiento y funcionalidad del software por parte de usuarios reales.
- Pruebas de despliegue: en este entorno también se simula las pruebas de despliegue en el entorno de producción mediante la ejecución de scripts, actualizaciones del sistema y procesos de rollback (revertir un conjunto de operaciones a un estado anterior) para asegurar que el despliegue se realice sin errores. [\[16\]](#)

A continuación, se muestran las ventajas e inconvenientes de un entorno de preproducción:

Ventajas	Inconvenientes
Detección temprana de problemas: permite detectar y corregir problemas al ser una réplica del entorno de producción.	Coste adicional: al disponer de dos entornos iguales, preproducción y producción, el mantenimiento de este entorno puede ser costoso porque tiene que disponer de los mismos elementos que el entorno de producción (hardware, software, tiempo y recursos humanos para su gestión).

Pruebas realistas: al existir gran similitud con la configuración del entorno de producción, las pruebas que se realizan, puede determinar el rendimiento y la estabilidad del software.

Reducción de riesgos: con las pruebas mencionadas anteriormente, se pueden minimizar los fallos críticos de cara a su despliegue en producción.

Validación de la configuración: en este entorno se comprueban que las configuraciones del sistema, redes y aplicaciones funcionen correctamente. Su objetivo es evitar incompatibilidades.

Complejidad en la gestión: la gestión puede volverse compleja al disponer de sincronización entre el entorno de preproducción y producción.

Posibles incompatibilidades: la incompatibilidad entre entornos puede representar un problema en el despliegue en producción. Esto es debido a que ambos tienen que ser exactamente iguales y disponer de la configuración, datos.

Falsos positivos / falsos negativos: el entorno de preproducción y producción deben disponer de la misma configuración, bbdd, etc. Pudiendo producirse falsos positivos si ambos entornos no son “espejos” uno de otro.

Tabla 2-4 Ventajas e Inconvenientes del entorno de preproducción (elaboración propia) [18]

○ Entorno de Producción

Entorno donde finalmente se ejecuta el software y donde acceden los usuarios finales. Tal y como se ha indicado anteriormente, tanto el entorno de preproducción como el entorno de producción deben ser una réplica uno de otro.

Algunas de las características del entorno de producción son:

- Estabilidad y Fiabilidad: el entorno debe estar diseñado para ser fiable y estable.
- Acceso a usuarios finales: en este entorno se despliega el software de manera funcional y operativo. El software desplegado debe proporcionar la configuración necesaria (configuración de servidores, de redes, bases de datos, etc.) para manejar el tráfico de usuarios y soportar la carga de trabajo necesaria sin comprometer el rendimiento.
- Seguridad: la seguridad es un objetivo esencial en el entorno de producción ya que debe protegerse contra ataques externos y garantizar la privacidad y la integridad de los datos de los usuarios. Para ello se implementan medidas de seguridad robustas, como firewalls, cifrado y controles de acceso.
- Herramientas de monitoreo y mantenimiento: al ser los servidores de producción un recurso crítico dispone de herramientas de monitorización para detectar y resolver problemas rápidamente. Asimismo, también disponen de herramientas que proporcionan el mantenimiento adecuado, aparte de las actualizaciones y parches que se instalan para minimizar riesgos. [16]

En el siguiente cuadro se muestran las siguientes ventajas e inconvenientes del entorno de producción:

Ventajas	Inconvenientes
Operación en condiciones reales: permite que los usuarios ejecuten un software operativo, pudiendo determinar el rendimiento y su funcionalidad.	Riesgo de fallos críticos: cualquier problema en el entorno de producción puede afectar a todos los usuarios finales. Una buena gestión de riesgos y sus posibles contingencias son esenciales para minimizar estos impactos.

Feedback del usuario: los usuarios pueden identificar área de mejoras comunicándolo al equipo desarrollador.

Estabilidad y Fiabilidad: estas capacidades son inherentes a la ejecución de un aplicativo por lo que son necesarias para que el software funcione de manera continua y confiable.

Valor de negocio: el software desarrollado genera valor real para la empresa o los clientes.

Mantenimiento: el coste de un entorno de producción es muy costoso debido a todos los elementos que lo componen (el hardware, el software, las licencias, el personal de soporte y las medidas de seguridad).

Implementación de cambios, seguridad y privacidad: aparte de la necesidad de requerir tiempo para su instalación, hay que ser muy cuidadoso con los cambios que se puedan llevar a cabo en un entorno de producción.

Escalabilidad: una de las características de las que debe prever el desarrollo de software es el aumento de usuarios.

Tabla 2-5 Ventajas e Inconvenientes del entorno de producción (elaboración propia) [19]

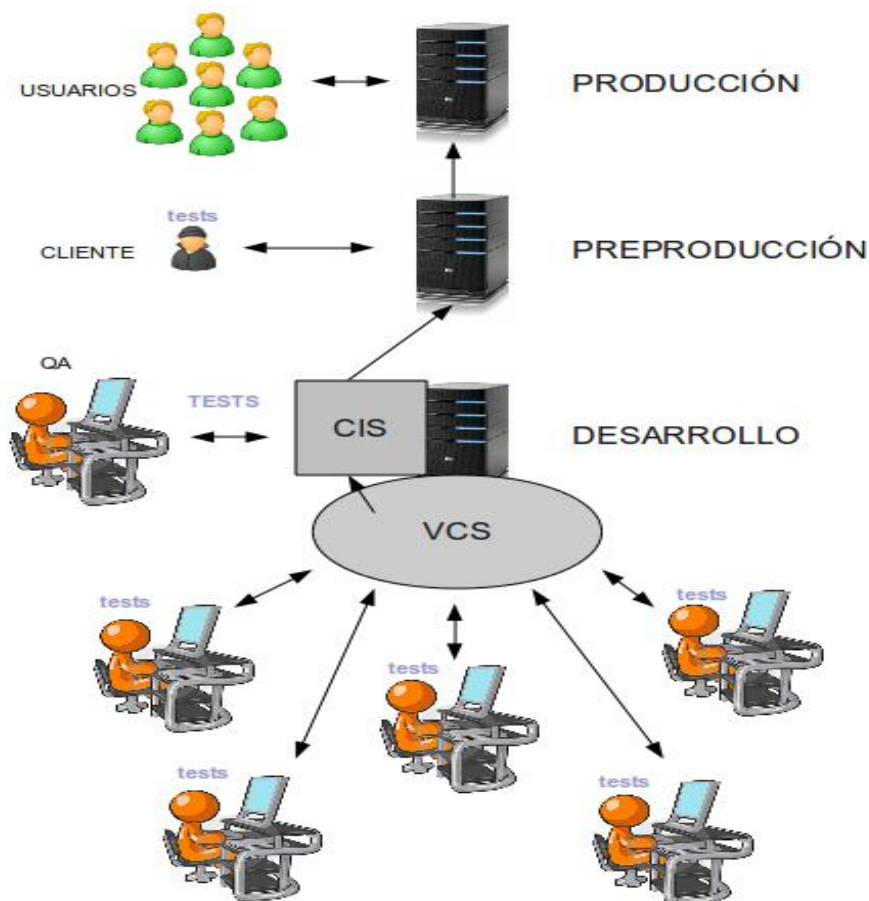


Figura 2-8 Entornos diferenciados [20]

2.1.4.4 Mantenimiento

Por último, aunque no menos importante, el mantenimiento del software es necesario para conseguir el éxito de cualquier desarrollo. El objetivo principal de un mantenimiento es la corrección de errores, implementación de nuevas funcionalidades y adaptación a cambios de entorno.

Existen diferentes tipos de mantenimiento, entre los que se encuentran:

- Mantenimiento correctivo: aparecen cuando existen fallos o errores en la lógica del software.
- Mantenimiento preventivo: con este mantenimiento se pretende evitar el deterioro del software. Normalmente surgen como pequeños fallos, pero si no se corrigen se puedan convertir en grandes en el futuro.
- Mantenimiento perfectivo o evolutivo: es produce cuando el cliente tiene la necesidad de nuevas características o requisitos para mejorar la aplicación. También se utiliza para eliminar todas aquellas funcionalidades que son irrelevantes o no efectivas.
- Mantenimiento adaptativo: depende de las tecnologías, arquitecturas, políticas, reglas, ... relacionadas con el software. Cuando se producen estos cambios, el software desarrollado se tiene que adecuar también a estos requisitos tecnológicos para su buen funcionamiento.

2.1.4.5 Aseguramiento de la Calidad

Continuando con los requisitos necesarios para el desarrollo del software, la calidad es un factor esencial. Para ello, se emplean términos como *Quality Assurance* (QA) y *Quality Control* (QC), que determinan qué requisitos debe cumplir un software para que se considere de calidad.

Como aclaración, el término QA es un conjunto de actividades y procesos para el asegurar que el producto final sea de calidad y cumpla los requerimientos del cliente. Estos requisitos son satisfechos mediante el cumplimiento de los estándares de calidad establecidos, procedimientos para el control de calidad, las inspecciones y pruebas, el seguimiento y la gestión de la documentación. Entre estos estándares se encuentran, ISO/IEC 25010:2011 sobre la Calidad del Producto de Software, ISO/IEC 12207 sobre el Proceso de Ciclo de Vida de Software, ISO/IEC 27001 sobre la Gestión de Seguridad de la Información, entre otras. Estas actividades y procesos incluyen al *Quality Control* (QC).

Por otro lado, el término QC, se refiere a los procesos de QA y trata de identificar y corregir cualquier desviación o no conformidad de los estándares de un producto.

Ambos términos, QA y QC, están interrelacionados, aunque la diferencia radica que QA es de carácter preventivo cuyo objetivo es evitar los errores antes de que ocurran y por su parte, QC, es de naturaleza reactiva y tiene la finalidad de comprobar si el producto final cumple con los requisitos.

Actualmente, las organizaciones son muy conscientes de asegurar la calidad del producto. Con ello se garantiza que el software cumpla con los requisitos, funcionalidades y seguridad para el cliente. Ello se traduce en un ahorro significativo de costes y de tiempo de desarrollo. [\[21\]](#)

La siguiente figura representa las características necesarias para desarrollar software junto con las capacidades para llevarlo a cabo:



Figura 2-9 Capacidades del desarrollo de aplicaciones [22]

Así mismo, uno de los aspectos más importantes del aseguramiento de la calidad es que permiten a las organizaciones identificar áreas de mejora, tomar decisiones acordes a las métricas indicadas y dar valor a los esfuerzos para su logro. Todo ello para evitar los problemas que dieron lugar a “la crisis del software”.

3 EQUIPO O SIN EQUIPO. ESTUDIO COMPARATIVO PARA EL DESARROLLO DE APLICACIONES

3.1 Elementos del Estudio

Este estudio comparativo tiene como finalidad identificar y definir una serie de elementos que permiten evaluar las ventajas, desventajas, riesgos y beneficios de cada uno de los modelos que se exponen.

Sin embargo, se señala que estas métricas no reflejan un correcto resultado final ya que existen otras habilidades y capacidades cualitativas, como son el talento, la experiencia y el conocimiento, entre otras, que deben ser tenidas en cuenta para este estudio.

Para ello, se determinan las características propias de cada modelo de equipo:

3.1.1 Equipo Interno (*In-house*)

En un equipo interno (*in-house*), los desarrolladores están contratados directamente por la organización, en este caso, las métricas están referenciadas a las empleadas en el Ejército del Aire y del Espacio (en adelante, EA).

Este equipo permite crear, desarrollar y mantener aplicaciones de gestión específicas durante todo su ciclo de vida (SDLC). Llamamos aplicaciones específicas a aquel software que dispone de unas funcionalidades y unas necesidades muy concretas en el Ejército del Aire y del Espacio (ej.: gestión de aeronaves, gestión de radiobalizas, etc.).

3.1.1.1 Tipos de Equipos Internos (*In-house*)

Según las necesidades y recursos de una organización, los equipos internos pueden adoptar la siguiente clasificación:

- *In-house* completo: el equipo interno es responsable de todo el ciclo de vida (SDLC).
- *In-house* parcial: el equipo interno se dedica a las tareas más críticas o estratégicas subcontratando otras fases de desarrollo a terceros.
- *In-house* centralizado: el equipo interno desarrolla todo el software, dando soporte a toda la organización.
- *In-house* distribuido: el desarrollo se realiza entre equipos internos ubicados en varias secciones/unidades.
- *In-house* híbrido: este tipo combina el desarrollo *in-house* con el *outsourcing*, permitiendo que el equipo interno controle el proyecto y se subcontraten áreas específicas del ciclo de vida del aplicativo (diseño, mantenimiento). [\[23\]](#)

3.1.1.2 Ventajas e Inconvenientes

A continuación, se enumeran las ventajas e inconvenientes al contar con un equipo interno:

Ventajas	Inconvenientes
Extenso conocimiento técnico de todas las tareas y funcionalidades del software. Ello facilita una respuesta rápida ante nuevos cambios, nuevas funcionalidades o errores y vulnerabilidades del software.	Disponer de un equipo interno conlleva gastos importantes, requiriendo una fuerte inversión inicial como es el alquiler de oficinas, hardware/software y otros como salarios para los desarrolladores.
La organización dispone de control total de todos los desarrollos permitiendo el aseguramiento de la Calidad del producto final.	Para que la tecnología empleada no se vuelva obsoleta, el cambio a nuevas tecnologías comprende la inversión en la actualización y la formación del equipo.
Mayor nivel de seguridad ya que los datos y la propiedad intelectual (código fuente) no es compartida con personal externo.	La retención del talento, rotación voluntaria y satisfacción laboral son factores críticos que pueden afectar al equipo interno.
Alineación con los intereses, valores y cultura de la organización, lo que favorece compromiso a largo plazo.	La capacidad limitada en cuanto a recursos humanos y el desarrollo de múltiples proyectos, puede ralentizar el desarrollo de nuevas iniciativas y mejoras.
Mejora la comunicación, colaboración e integración entre sus integrantes y en la organización.	Ante un aumento en la demanda, el equipo interno no puede asumir esa sobrecarga pudiendo afectar a la productividad.

Tabla 3-1 Ventajas e Inconvenientes de un equipo interno (*in-house*) [24]

3.1.2 Equipo Externo (*Outsourcing*)

Por otro lado, el desarrollo de software se realiza con un equipo mediante la subcontratación de proveedores externos (*outsourcing*), que proporcionan el servicio para el desarrollo de aplicaciones.

3.1.2.1 Tipos de Equipos Externos (*Outsourcing*)

Al igual que los equipos internos, los equipos externos disponen de una clasificación acorde a las necesidades requeridas, siendo:

- *Co-sourcing*: en este tipo de *outsourcing*, la organización y el proveedor externo comparten las responsabilidades y los riesgos derivados del desarrollo del software.

- *Off-sit*: esta modalidad establece que el desarrollo se realiza en las instalaciones de la empresa subcontratada.

- *Off-shoring*: estos servicios son subcontratados a una empresa extranjera, normalmente porque ofrecen menores costes.

- *In-house* externalizado: los servicios externalizados se realizan en la empresa contratante, aunque son ejecutados por una empresa externa. [25]

3.1.2.2 Ventajas e Inconvenientes

A continuación, se enumeran las ventajas e inconvenientes al contar con un equipo externo:

Ventajas	Inconvenientes
Ahorro de costes ya que al contratar a un proveedor externo, la organización puede reducir costes en infraestructura, personal.	Al externalizar funciones a terceros, se pueden perder cierto control sobre funcionalidades.
Mayor eficiencia debido a que al externalizar funciones no esenciales, la organización puede centrarse en otras actividades más productivas.	Puede haber problemas de comunicación entre la organización y el proveedor.
Acceso a nuevas tecnologías y conocimientos que de lo contrario no podría ser.	Riesgo de seguridad en los datos, teniendo que asegurar que el proveedor disponga de medidas de seguridad adecuadas.
Flexibilidad y escalabilidad ya que la organización puede ajustar los recursos según las necesidades y no tienen que preocuparse por personal adicional cuando hay poca demanda.	Puede conllevar la dependencia excesiva del proveedor. La organización debe disponer de un plan de contingencia en caso de que el proveedor no esté disponible.

Tabla 3-2 Ventajas e Inconvenientes de un equipo externo (*outsourcing*) [\[24\]](#)

3.1.3 Elementos Análogos

Para este estudio, se considera que existen elementos análogos a ambos equipos, como son:

3.1.3.1 Perfiles

El desarrollo de aplicativos, requiere de perfiles técnicos muy especializados y con diferentes capacidades y habilidades. Normalmente, en el desarrollo de proyecto suelen coincidir perfiles determinados con son el director de proyectos, jefe de proyectos, arquitecto de software, gestor de proyectos, analista, programadores (*front-end*, *back-end* y *senior/junior*), *testers* y equipo de calidad (QA). Sin embargo, dependiendo del tipo de desarrollo y de la metodología a emplear pueden existir perfiles muy específicos como pueden ser el de *Product Owner*, *Scrum Master*, Diseñador UX/UI o Técnico en seguridad.

Las capacidades y habilidades de los perfiles mencionados se citan en el [punto 2.1.4.1](#).

3.1.3.2 Tecnologías

Las tecnologías empleadas en cada uno de los grupos son prácticamente análogas, aunque dependerán de las necesidades funcionales de la organización, de los perfiles técnicos requeridos y de la infraestructura de despliegue necesaria para los diferentes entornos. A continuación, se enumeran las tecnologías más usadas:

- Lenguajes de programación: Java, Python, JavaScript, TypeScript.
- Herramientas de gestión de proyectos: Jira, Trello, Asana.
- Control de versiones: GitHub, GitLab, Bitbucket.
- Entornos de desarrollos integrados: Visual Studio Code, IntelliJ IDEA, Eclipse.

- Bases de datos: MySQL, PostgreSQL, MongoDB.
- Herramientas CI/CD: Jenkins, CircleCI, GitLab CI.
- Infraestructura en la nube: Amazon Web Services (AWS), Google Cloud Platform (GCP), Microsoft Azure. [\[26\]](#)

3.1.3.3 Metodologías de Trabajo

Cada metodología de trabajo empleado en el desarrollo de aplicaciones va en función de las necesidades de la organización, el valor de la calidad, la rapidez, la innovación, ...

Las siguientes metodologías de trabajo se pueden agrupar de la siguiente forma:

- Metodologías Convencionales:

- Desarrollo en cascada (Waterfall), este modelo está basado en el desarrollo secuencial entre fases, siendo que una nueva fase comienza cuando se ha completado la anterior. Cada proyecto pasa por las diferentes fases, tal y como se explicó en el [punto 2.1.4.2](#).

Al igual que todas las metodologías presenta pros y contras siendo las primeras cuando el proyecto tiene un objetivo bien definido y no hay restricción de presupuesto ni de tiempo. En contra, es complicado volver a una fase una vez finalizada y el aseguramiento de la calidad se produce en la última fase del desarrollo del software. [\[16\]](#)

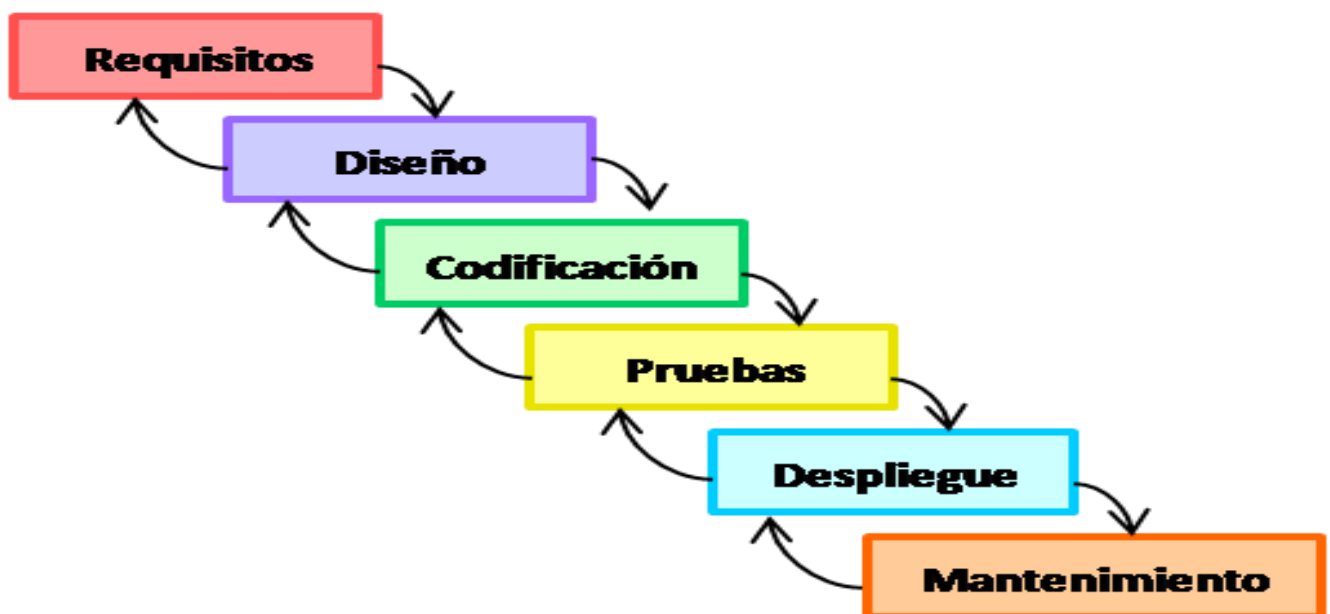


Figura 3-1 Metodología en Cascada [\[27\]](#)

- Desarrollo en V, este modelo representa las fases del ciclo de vida del desarrollo de software integrando en cada fase la verificación y validación por medio de test. Presenta la ventaja que se minimizan riesgos en el proyecto y los errores se pueden corregir sin tener que llegar a la etapa final y, por el contrario, dispone de poca flexibilidad para los cambios y es más costoso debido a la realización de pruebas y verificación para cada fase. [\[16\]](#)



Figura 3-2 Metodología en V [27]

○ Desarrollo en espiral, consiste en que las actividades se crean en espiral y se llevan a cabo según prioridades en función del análisis de riesgos. Permite una total flexibilidad a los cambios y el control de los riesgos. Sin embargo, no es recomendable en proyectos a corto plazo por requerir técnicos para el análisis de riesgos y, a medida que aumentan las espirales, el proyecto se hace más costoso. [16]

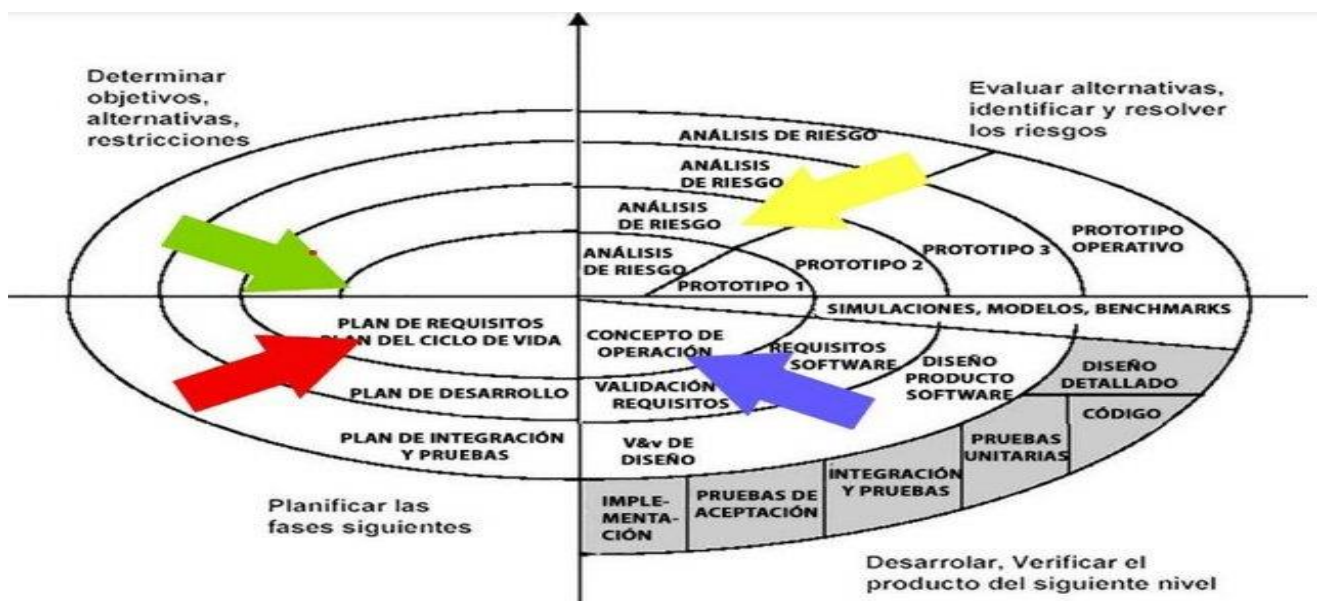


Figura 3-3 Metodología en Espiral [27]

- Metodologías Ágiles:

○ SCRUM, este modelo está basado en la flexibilidad y la pronta entrega del producto. Para ello, las diferentes iteraciones se dividen en *sprints* o ciclos cortos de desarrollo. Esta metodología destaca para proyectos creativos e innovadores por las entregas frecuentes e incrementales donde el cliente puede comprobar la evolución de las funcionalidades solicitadas. En contra, los equipos pueden desviarse fácilmente de las tareas del proyecto y requiere un alto nivel de compromiso de todo el equipo. [16]

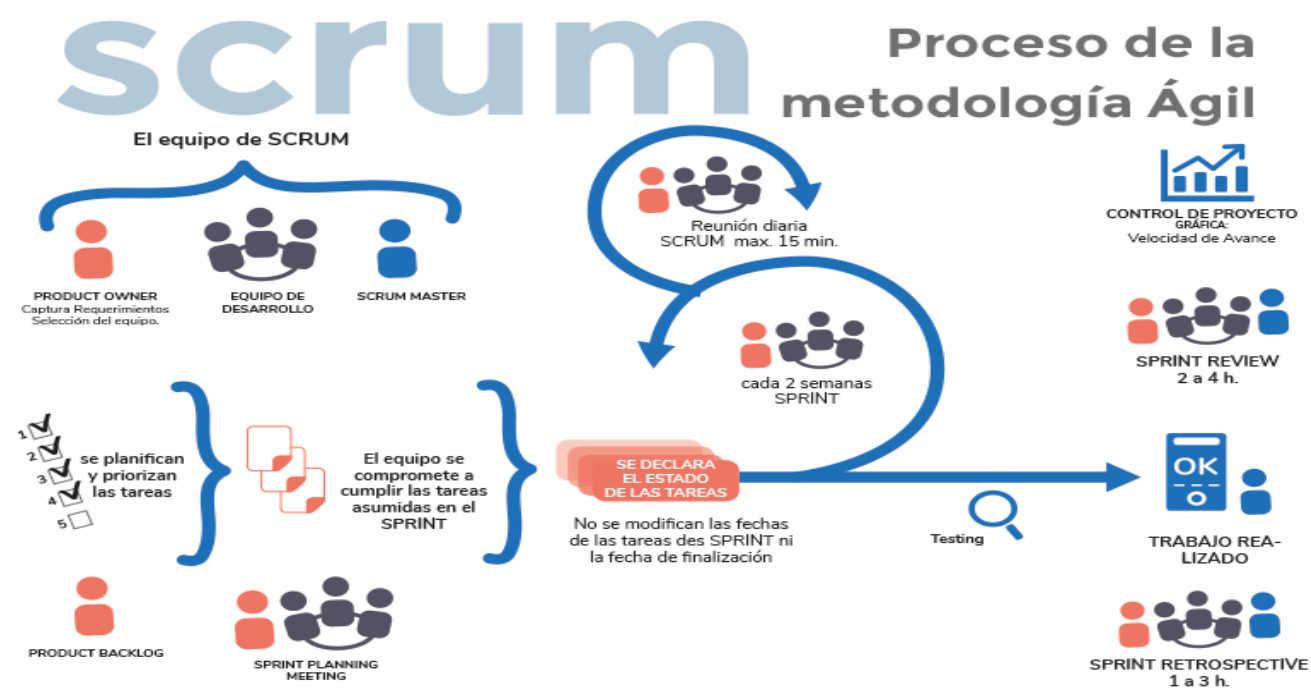


Figura 3-4 Metodología SCRUM [27]

○ KANBAN, marco ágil de trabajo utilizado en el desarrollo de software DevOps basado en la gestión de proyectos que permite a los equipos visualizar los flujos de trabajo y la carga de trabajo. Como ventaja permite que los equipos se centren en tareas específicas mejorando la calidad y proporciona el estado de los proyectos en tiempo real. En cuanto a sus desventajas, no establece claramente la prioridad de las tareas y no puede adaptarse fácilmente a los cambios. [16]

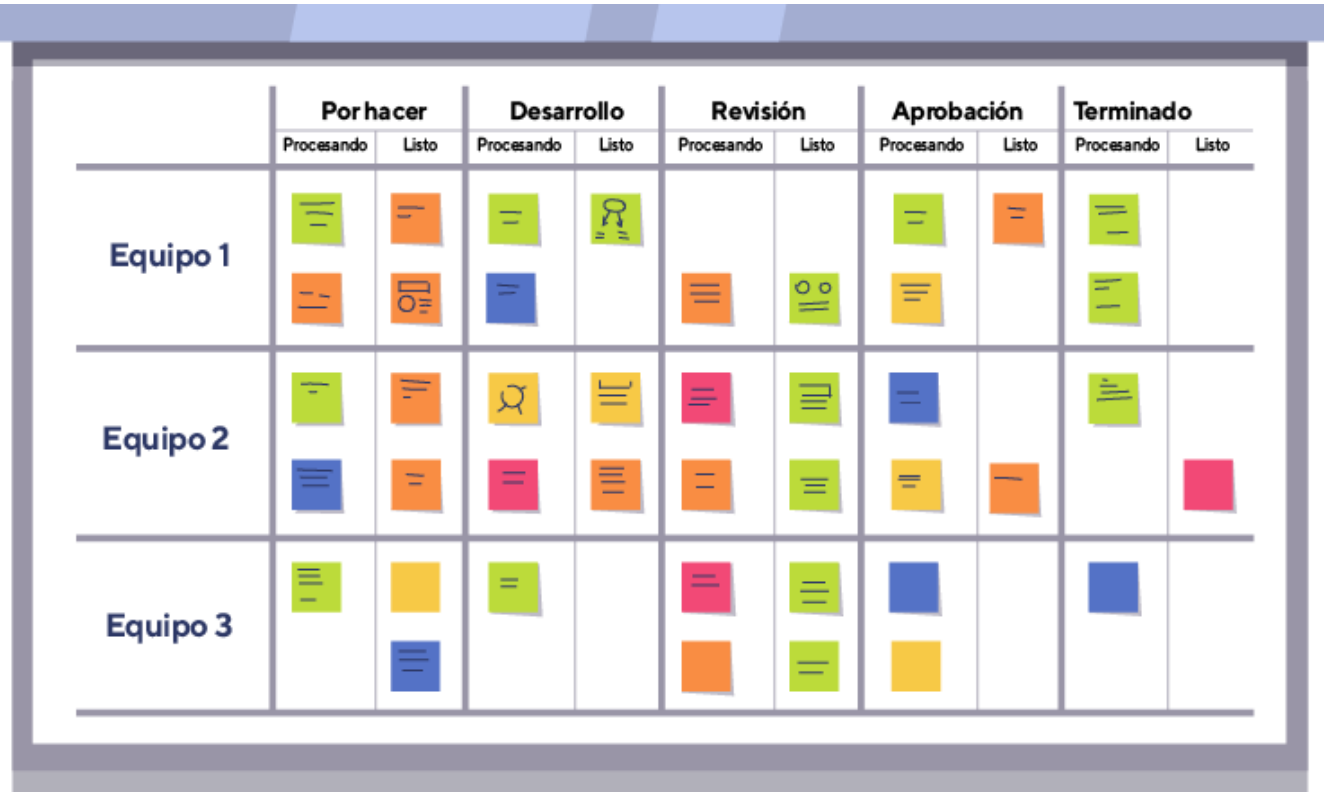


Figura 3-5 Metodología KANBAN [27]

○ Programación extrema (XP), es una metodología de desarrollo ágil que permite desarrollar y gestionar los proyectos con eficiencia, flexibilidad y control. Se basa en la comunicación, la reutilización de código. Sin embargo, requiere mucho tiempo de desarrollo y es relativamente caro. [16]

PROGRAMACIÓN EXTREMA (XP)

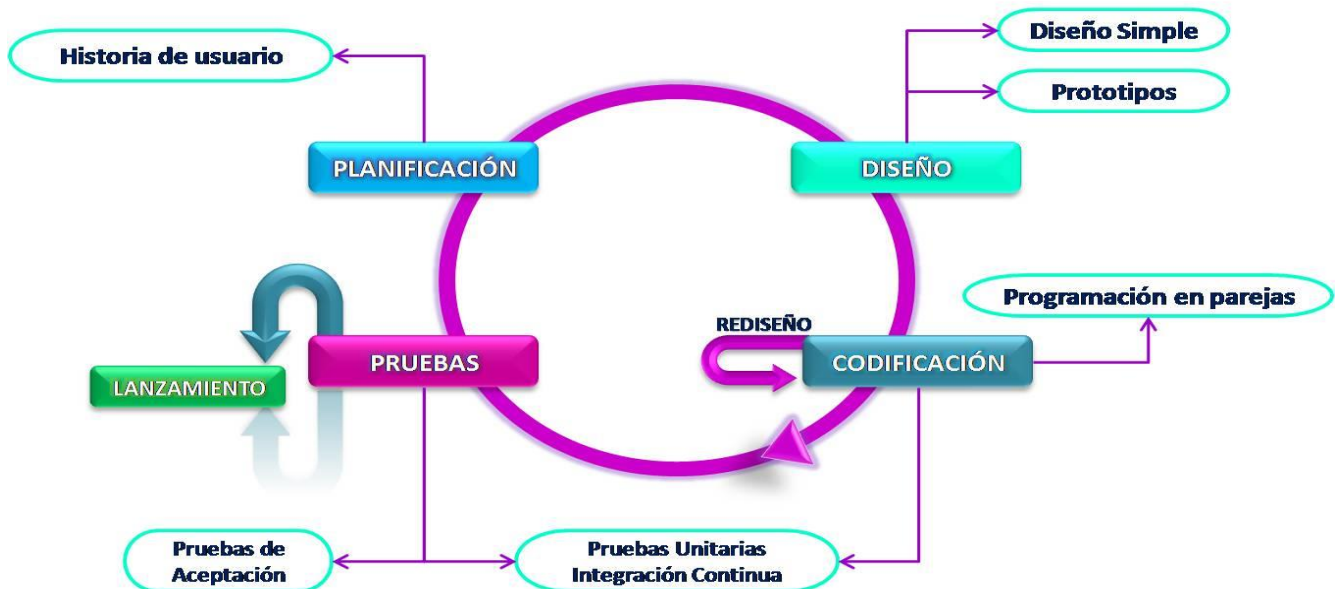


Figura 3-6 Metodología Extreme Programming [27]

3.2 Metodología de Investigación

Para llevar a cabo la comparativa, se han seleccionado cinco (5) proyectos del equipo interno (*in-house*) y cinco (5) proyectos del equipo externo (*outsourcing*), haciendo un total de diez (10) proyectos.

Es importante señalar que, aunque la comparativa es a nivel general, los proyectos están organizados por pares, es decir, cada proyecto del equipo interno tiene su equivalente en el equipo externo, acorde a los criterios comunes que se exponen a continuación.

También remarcar que en las métricas están excluidos todos aquellos elementos que no son propios del ciclo de vida del desarrollo de software, como es la infraestructura, dispositivos y servicios, la formación del personal para la adquisición de conocimientos técnicos, entre otros.

3.2.1 Criterios Comunes

Los criterios que se exponen a continuación están basados en elementos que garantizan que ambos equipos han desarrollado aplicaciones bajo condiciones similares. Estos criterios permiten realizar comprobar la eficiencia y eficacia en el desarrollo de software:

- **En el desarrollo de aplicaciones**, los equipos internos como los equipos externos disponen de su propia infraestructura y tecnología, eso permite el aprovechamiento de sus propios recursos y entornos. Sin embargo, no se consideran relevantes la tecnología utilizada ni las metodologías adoptadas, ya que los parámetros valorados se centran en el tiempo empleado en su ciclo de vida, el coste, el número de errores o incidencias, entre otras. Además, se puntualiza que ambos equipos, antes del despliegue en el entorno de preproducción, confluyen en un único punto en el EA, el Grupo de Desarrollo del Centro de Informática de Gestión (CIGES), quien se encarga de cotejar, documentar, validar y asegurar el correcto

despliegue de las aplicaciones en entorno de preproducción y de producción en la Infraestructura Integral de Información para la Defensa (I3D)¹⁰.

- **Los proyectos son seleccionados según sus flujos de trabajo:** Un flujo de trabajo, *workflow*, se emplea para gestionar procesos y tareas repetitivas con un determinado orden. Su principal objetivo es reducir el tiempo de trabajo debido a la automatización de procesos repetitivos, minimizar los errores ya que identifican cuellos de botella, redundancias o errores de proceso y optimizar el uso de los recursos. Cada grupo se compara bajo un mismo marco organizativo y operativo. Al seleccionar los proyectos con flujos de trabajos similares, se garantiza la eficacia y la capacidad de respuesta de ambos equipos. [28]

- **Análoga estructura en cuanto a complejidad técnica:** los proyectos seleccionados para ambos equipos presentan una dificultad similar. Ello asegura que los esfuerzos sean equivalentes. La complejidad técnica relaciona el número de tareas de cada proyecto, pero también su diversidad y especialización para acometerlas. La complejidad técnica también está relacionada con la interdependencia de tareas ya que una tarea puede depender de otras, lo que aumenta su planificación y ejecución (si la implementación del *back-end* depende de tareas que se tienen que llevar a cabo en el front-end, la coordinación es esencial). Al comparar la complejidad técnica de los equipos se garantiza que los proyectos resuelvan desafíos similares. [29]

- **Funcionalidades similares:** las funcionalidades determinan las características y capacidades del software, aportando valor, usabilidad y ahorro de tiempo al automatizar procesos repetitivos. Hay que tener en cuenta que las funcionalidades también permiten evaluar la capacidad de cada equipo para gestionar el ciclo de vida del desarrollo de software (SDLC). Al asegurar que los equipos trabajan en proyectos con funcionalidades similares, se evalúa la capacidad para plasmar los requisitos de un cliente en un proyecto, aumentando la eficiencia y generando ventajas competitivas para el cliente.

- **El número de roles y/o usuarios es equivalente** para cada aplicativo es un factor clave ya que tiene un impacto directo en la complejidad en el desarrollo de software. Un mayor número de usuarios comprende mayor volumen de datos, necesidad de escalabilidad, un imprescindible grado de seguridad y un constante mantenimiento. Además, el número de usuarios también afecta a los recursos necesarios como infraestructura y el soporte y atención al usuario. Al ser un número de usuarios equivalente para cada proyecto, se garantiza que ambos equipos trabajen con las mismas condiciones.

- **Herramientas comunes para la gestión y desarrollo de software:**

- Jira: herramienta que permite la gestión de los proyectos, organizando y llevando a cabo el seguimiento de tareas, errores, incidencias, flujos de trabajo, procesos. Debido a su capacidad para capturar datos detallados sobre tareas, cumplimiento de plazos y resolución de problemas, Jira proporciona elementos para realizar la comparación de manera cuantitativa. [31]

- SonarQube: herramienta que proporciona el seguimiento y detección de errores y vulnerabilidades de seguridad en el análisis estático del código fuente mediante:

- Reglas de codificación: son filtros para generar problemas en el código fuente. Existen cuatro tipos de reglas; *Code Smells* (dimensión de mantenibilidad), *Error* (dimensión de confiabilidad), *Vulnerability* (dimensión de seguridad) y *Access Point* (dimensión de seguridad).

- Cobertura: analiza en estático el código fuente a fin de que los desarrolladores escriban código más limpio y seguro.

¹⁰la Infraestructura Integral de Información para la Defensa (I3D), infraestructura tecnológica única, de carácter estratégico del Ministerio de Defensa. Está diseñada para proporcionar servicios CIS/TIC a las Fuerzas Armadas. [30]

- Comentarios: permite analizar cuantas líneas de código tiene comentarios sobre cómo están designadas las variables, clases. En resumen, su función es que el desarrollo pueda ser mantenido por otros desarrolladores diferentes de los que lo realizaron.
- Código duplicado: este parámetro evalúa las líneas de código que están duplicadas.

Este software es esencial para la fase de testing y auditoría dentro del ciclo de vida de una aplicación. [\[32\]](#)

3.2.2 Métricas

Para poder evaluar el rendimiento de los equipos internos y externos, se han identificado una amplia variedad de métricas con diferentes alcances, enfoques y fórmulas que permiten medir aspectos como coste, tiempo de respuesta, la velocidad de desarrollo, el número de errores, incidencias, etc. Estas métricas aportan datos para cada una de las fases del ciclo de vida del desarrollo de un aplicativo (SDLC).

Sin embargo, ha sido necesario realizar una selección de las métricas que proporcionen los datos más relevantes. Ello permitirá disponer de una visión representativa para este estudio. Además, como complemento a las métricas evaluadas, se han incluido en el [Anexo III](#) y en el [Anexo IV](#), otras métricas que podrían ser de utilidad en estudios futuros.

3.2.2.1 Métricas Cuantitativas

A continuación, se describen los conceptos cuantitativos relacionados con el software, siendo éstos, los parámetros utilizados en la comparativa entre equipos internos (*in-house*) y externos (*outsourcing*):

- **Funcionalidad:** es la capacidad del software que determina las tareas y funciones específicas para lo que fue diseñado.

Se consideran las siguientes características para asegurar una buena funcionalidad:

- Exactitud: si el software ha sido diseñado para realizar las tareas correctamente.
- Interoperabilidad: si existe interacción con otros sistemas de manera eficiente.
- Buenas prácticas: si se le aplican los estándares desarrollo y de buenas prácticas para su desarrollo.

- **Seguridad:** hace referencia a la necesidad de proteger el software de amenazas potenciales (accesos no autorizados, inyección de código malicioso).

Los aspectos importantes de la seguridad son:

- Confidencialidad: el desarrollo del software debe contener de protección de la información ante accesos no autorizados.
- Integridad: disponer que los datos no puedan ser alterados.
- Autenticación y autorización: estos aspectos hacen referencia a que los usuarios autorizados solo, y solo ellos, puedan acceder a la información.

- **Disponibilidad:** significa disponer de los datos cuando se requiera.

Una alta disponibilidad, se define por los siguientes factores:

- Tiempo de actividad: tiempo de funcionamiento correcto del software.

- Tolerancia a fallo: capacidad de resiliencia ante problemas o fallos.

- **Fiabilidad:** la fiabilidad consiste en la capacidad para que el software pueda funcionar sin fallos o interrupciones durante un período de tiempo.

Para garantizar la fiabilidad, se debe considerar:

- Tolerancia a fallo: capacidad de resiliencia ante problemas o fallos.
- Estabilidad: implica si el software puede funcionar ante fallos.
- Resiliencia: capacidad para que, ante fallos o errores, pueda funcionar de manera normal.

- **Usabilidad:** es la capacidad que determina la forma en que el usuario interactúa con el software.

La usabilidad debe incluir:

- Facilidad de aprendizaje: en cuanto tiempo el usuario está familiarizado con el software.
- Eficiencia: velocidad en la que un usuario puede realizar tareas una vez aprende su manejo.
- Prevención de errores: capacidad para minimizar los errores de usuarios o ayudar a corregirlos rápidamente.
- Satisfacción: grado de satisfacción del usuario con el software.

- **Mantenibilidad:** es la capacidad de la que dispone un software para poder ser modificado, implementar funcionalidades o mejorar su rendimiento. Es buena práctica disponer de la documentación ya que permite a los programadores que no participaron en su desarrollo poder continuar desarrollando las funcionalidades.

Para ello, se deben tener en cuenta los siguientes factores:

- Modificabilidad: facilidad para modificar y actualizar el software.
- Corrección de errores: tiempo en encontrar y corregir errores de software.
- Pruebas: facilidad para probar los cambios realizados.
- Reusabilidad: reutilización de elementos del software en otros desarrollos de software.

Todos estos conceptos son indicadores que determinan si el software cumple apropiadamente con sus funciones, si es seguro, eficiente, fácil de usar y sostenible a lo largo del tiempo. Cuando estos factores se cumplen determina que el desarrollo de software es de alta calidad. [\[16\]](#)

3.2.2.2 Métricas Cualitativas

A continuación, se describen los conceptos cualitativos relacionados con el software, para la comparativa entre equipos internos (*in-house*) y externos (*outsourcing*):

- **Talento:** hace referencia a un conjunto de habilidades, conocimiento, capacidades y aptitudes para realizar de actividades complejas. Los equipos de desarrollo con talento son creativos, resolutivos, metódicos y dados a la innovación.

Particularidades del talento:

- Innato o desarrollado: el talento puede provenir de modo innato o bien puede ser desarrollado mediante esfuerzo y experiencia.
- Potencial: el talento puede convertirse en una gran ventaja si es gestionado correctamente.
- Diversidad de talento: los equipos suelen disponer de talentos diferentes individuales.

- **Experiencia:** se entiende como los conocimientos y habilidades que se han adquirido a lo largo del tiempo mediante éxitos y fracaso.

Los aspectos importantes de la experiencia son:

- Duración: es el tiempo dedicado a acumular experiencia.
- Profundidad: en la experiencia importa no solamente el tiempo en alcanzarlo sino también las situaciones a las que se ha enfrentado una persona.
- Aplicabilidad: la experiencia es especialmente útil cuando es aplicada a nuevas situaciones y entornos.

- **Conocimiento:** consiste en la comprensión teórica y práctica de un tema. Se basa en el estudio, observación e investigación.

Se distinguen los siguientes factores:

- Teórico y práctico: el conocimiento se puede adquirir a través del estudio y mediante la implicación directamente en tareas y proyectos.
- Compartir y enseñar: la efectividad del equipo se incrementa cuando el conocimiento se comparte de manera eficiente.

- **Resolución de Problemas Complejos:** habilidad para analizar situaciones complicadas y encontrar la solución.

Los aspectos importantes en la resolución de problemas complejos son:

- Análisis crítico: si el problema se descompone en partes más pequeñas y manejables será más fácil su resolución.
- Creatividad: adoptar otros puntos de vista puede inspirar soluciones.
- Iteración: la resolución de problemas complejos conlleva varios intentos hasta dar con la solución correcta.

- **Pensamiento Crítico:** se trata de evaluar y analizar la información para tomar decisiones.

Aspectos del pensamiento crítico:

- Toma de decisiones: el pensamiento crítico ayuda a tomar decisiones basado en hechos empíricos.
- Curiosidad: hace referencia a la necesidad de aprender y cuestionar lo que se sabe.

- **Adaptabilidad:** la adaptabilidad es la capacidad para adaptarse a nuevas situaciones y entornos.

Factores para la adaptabilidad:

- Resiliencia: ser capaz de recuperación ante cambios inesperados o fracasos.
- Flexibilidad: poder cambiar de perspectiva ante problemas que surjan.
- Aprendizaje: se trata de la facilidad para formarse e instruirse en nuevas habilidades y tecnologías.

- **Comunicación efectiva:** es la habilidad para relacionarse con el equipo. Su importancia reside en escuchar, entender y transmitir ideas de manera clara y comprensible.

Los componentes de la comunicación efectiva son:

- Claridad y precisión: la comunicación en el equipo debe ser clara, evitando ambigüedades.

- Escucha activa: saber escuchar y comprender al interlocutor es importante para la comunicación en el equipo.
- Adaptación al entorno: la comunicación debe acomodarse al entorno según a quien se le esté hablando.

Dado que estas métricas no pueden ser medidas cuantitativamente, se recurre a:

- La observación directa permite valorar cómo los integrantes del equipo aplican el conocimiento de forma efectiva y cómo resuelven problemas complejos en tiempo real.

- Las entrevistas aportan información sobre el pensamiento crítico, adaptabilidad a nuevos desafíos y la habilidad para comunicar ideas y soluciones de manera efectiva.

- y las evaluaciones permiten medir la capacidad de resolución de problemas complejos, el pensamiento crítico, la adaptabilidad con datos sobre cómo los equipos se enfrentan a desafíos técnicos y operativos. [\[33\]](#)

3.2.3 Selección de Casos de Uso

Para llevar a cabo los casos de estudio se han seleccionado por su similitud según los criterios comunes descritos, los siguientes proyectos:

Equipo interno (In-house)	Características	Equipo externo (Outsourcing)	Características
InvenTIC	Gestión de peticiones hardware/software/consumibles. 750 usuarios. 6 roles. 4 funcionalidades.	EcoGest	Sistema de gestión medioambiental y energético. 900 usuarios. 5 roles. 4 funcionalidades.
IDManage	Gestión para la solicitud de tarjetas identificativas. 600 usuarios. 5 roles. 7 funcionalidades.	Monitora	Gestión y control de Centros de personal y material. 400 usuarios. 5 roles. 15 funcionalidades.
SaniGest	Gestión de sanidad. 400 usuarios. 5 roles. 9 funcionalidades.	UnitPlan	Gestión del estado y programación de una Unidad. 300 usuarios. 6 roles. 15 funcionalidades.
CargoNet	Gestión y administración para transporte aéreo de personal y material. 515 usuarios. 7 roles. 5 funcionalidades.	SensorPro	Gestión de sensorización de salas (temperatura, humedad). 350 usuarios. 5 roles. 8 funcionalidades.
PlanEA	Sistema de información para UCOs (Unidades, Centros y Organismos.) 900 usuarios. 8 roles. 8 funcionalidades.	GesLog	Gestión logística de material. 1.300 usuarios. 6 roles. 5 funcionalidades.

Tabla 3-3 Proyectos comparados por pares según criterios comunes (elaboración propia)

3.2.3.1 Indicadores Cuantitativos

Como punto de partida para este estudio, se han desarrollado tres (3) tablas para evaluar las métricas cuantitativas, basándose en:

- indicadores que evalúan el ciclo de vida del proyecto (SDLC),
- indicadores operativos,
- indicadores de calidad del código fuente.

Estos indicadores han sido basados en datos extraídos de la herramienta Jira (las dos primeras tablas) y en la herramienta SonarQube (la tercera tabla).

A continuación, se muestra un ejemplo de cómo se visualizaría la interfaz de la herramienta Jira, utilizada para los indicadores del ciclo de vida e indicadores operativos de los proyectos, y de la herramienta SonarQube, para los indicadores de calidad del código fuente:

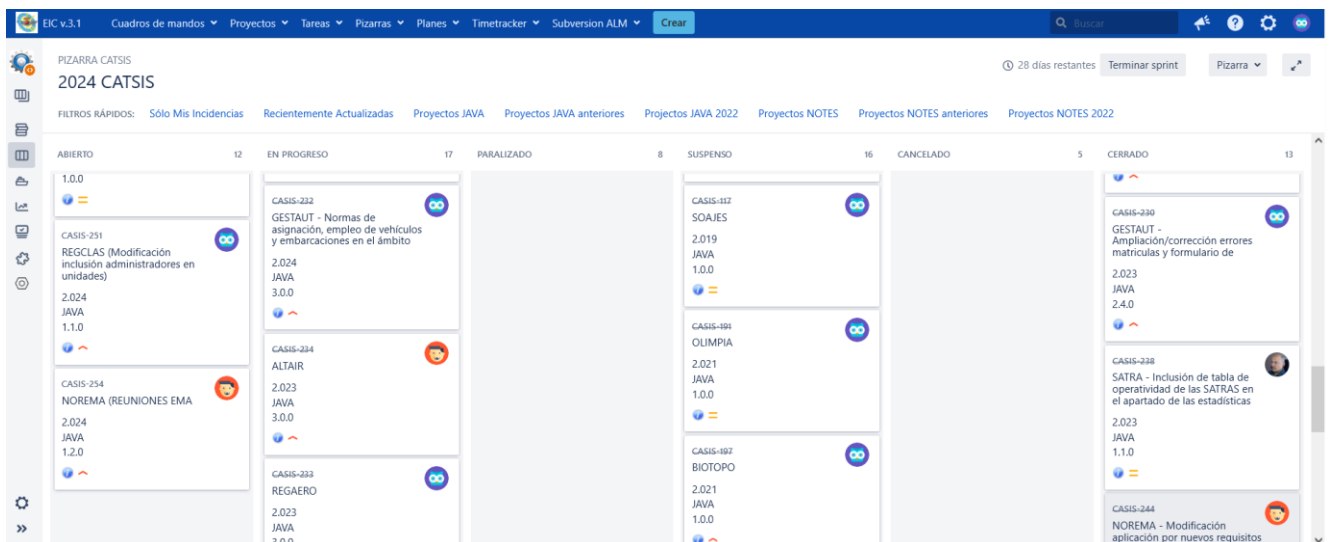


Figura 3-7 Interfaz Herramienta Jira (elaboración propia)

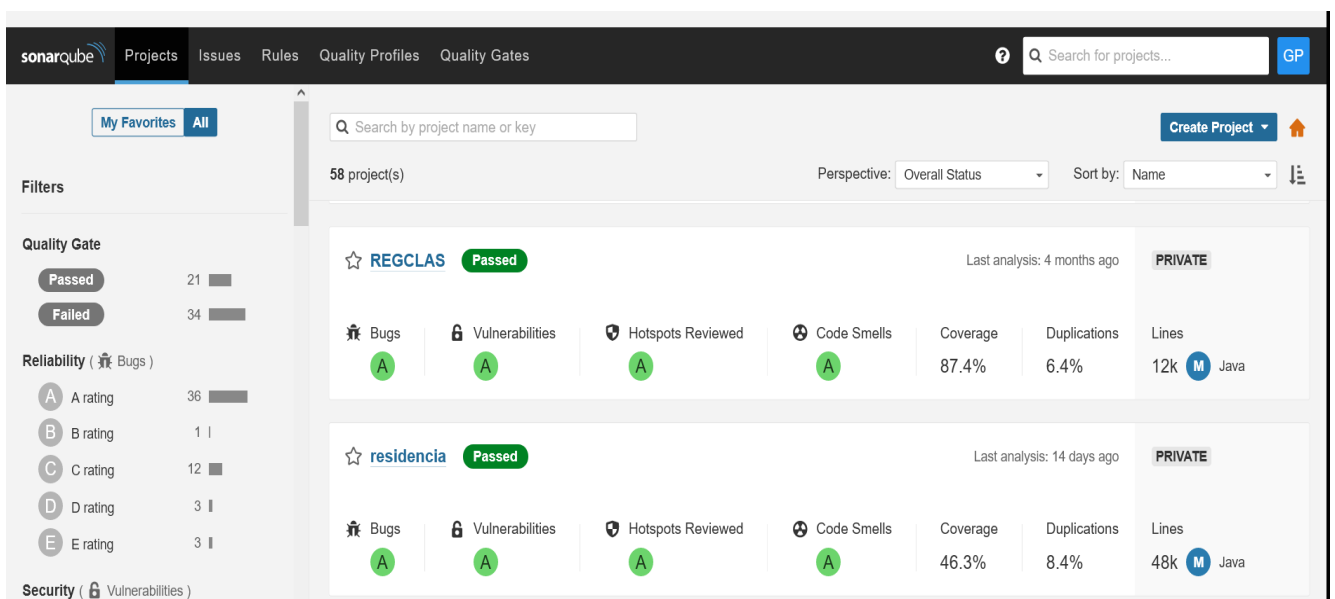


Figura 3-8 Interfaz Herramienta SonarQube (elaboración propia)

Tal como se ha expuesto, la primera tabla tiene por objeto evaluar los diferentes parámetros de cada proyecto durante todo el SDLC ([punto 2.1.4.2](#)). Para ello, los indicadores expuestos permiten conocer el tiempo empleado, el coste, la duración estimada, la duración real en comparación con lo previsto, etc.

Las fases del SDLC consideradas para este estudio incluyen:

- **Petición:** etapa inicial donde se solicita la realización del nuevo desarrollo o mantenimiento de software por parte de un peticionario. En esta fase se analizan los aspectos de tiempos de respuesta y viabilidad técnica.

- **Análisis y Diseño (A+D):** En esta fase se evalúan los requisitos del sistema, la arquitectura a emplear y su planificación. Como indicadores principales se incluye la precisión en el diseño y cumplimiento de plazos.

- **Codificación y Pruebas (C+P):** En esta fase se realiza la compilación del código y las pruebas necesarias para garantizar su funcionamiento. Como aspectos importantes, mencionar la codificación, núm. de errores detectados y los costes.

- **Producción:** una vez desplegado el software, esta etapa tiene como objetivo la identificación de fallos durante la puesta en producción. Por ello, se evalúan los errores de concurrencia, fallos en la lógica del negocio, fallos no detectados anteriormente y, en general, la estabilidad del software.

- **Servicio:** período donde el software ya está operativo. En esta fase se monitorea el rendimiento, tiempos de respuesta y soporte, así como la resolución de incidencias.

Estos parámetros permiten evaluar e identificar la eficacia en cada una de las fases del desarrollo, aparte de identificar posibles áreas de mejoras pudiendo optimizar la gestión de procesos, mejorar la asignación de recursos y ajustar tanto el tiempo como los costes asociados al proyecto.

A continuación, para su interpretación se describen los parámetros según orden alfabético:

Parámetros	Descripción	Obtención	Afecta a
CPE CRC Proceso	Calcula el valor entre las conformidades (CON ¹¹), las no conformidades (NCO) y las no conformidades condicionadas (NCO COND)	Es la información proporcionada por el peticionario (RCON-Revisiones Conjuntas).	Fiabilidad, Funcionalidad y Mantenibilidad
$CRC = \frac{(CON)}{\sum (CON + NCO + NCO\ COND)}$			

¹¹ Conformidad (CON): es el cumplimiento de requisitos, especificaciones y funcionalidades determinado por el cliente para satisfacer sus necesidades.

Demora del Usuario	Demoras en el proyecto no atribuidas al equipo de desarrollo. <i>Identifica factores externos de retraso</i>	Métricas obtenidas para cada una de las fases en relación a las fechas.	Fiabilidad
DUP (Duración Prevista)	Duración esperada para completar el proyecto. $DUP = Fecha Prevista Fin - Fecha Prevista Inicio$	Métricas obtenidas para cada una de las fases en relación a las fechas.	Disponibilidad y Fiabilidad
DUR (Duración Real)	Duración efectiva en días una vez que el proyecto finalizó. $DUR = Fecha Real Fin - Fecha Real Inicio$	Métricas obtenidas para cada una de las fases en relación a las fechas.	Fiabilidad y Mantenibilidad
Esfuerzo (HH-Hora Hombre)	Medición de las horas-hombre dedicadas al proyecto. $Esfuerzo = \frac{Coste Total}{Coste HH}$	Coste total entre número de perfiles técnicos.	Fiabilidad y Mantenibilidad
NCP RFC (Solicitud para el cambio, <i>Request For Change</i>) RFC1, RFC2 y RFC3	Calcula si las funcionalidades solicitadas son conformes o no. Complementa a la métrica anterior. Aparece cuando antes de finalizar el desarrollo, el peticionario requiere nuevas funcionalidades $RFC = \frac{\sum(CON)}{\sum(CON + NCO)}$	Métricas obtenidas por cada revisión conjunta (RCON)	Funcionalidad y Mantenibilidad
PDP (Precisión Duración Prevista)	Grado de coincidencia entre la duración prevista y la real.	Métricas obtenidas para cada una de las fases en relación a las fechas.	Fiabilidad y Mantenibilidad

$PDP = \frac{(DUR - Demora\ del\ usuario)}{DUP}$			
PTP (Precisión del Esfuerzo)	Calcula en horas el tiempo previsto entre el trabajo real (TRR), el trabajo previsto (TRP) y el sobreesfuerzo	Métricas obtenidas en cada informe de verificación de cada proceso.	Funcionalidad y Mantenibilidad
$PTP = \frac{(TRR - SE)}{TRP}$			
PRD Complejidad (PFA-Puntos de Función de Albrecht)	La complejidad de cada proyecto comprende el esfuerzo, el tamaño, la duración y el coste del software desde el punto de vista funcional proporcionado al usuario.	A través de tablas que evalúan el esfuerzo, tamaño, la duración y el coste	Fiabilidad y Mantenibilidad
$PFA = \sum (Núm.\ de\ ficheros\ de\ entrada + Núm.\ de\ ficheros\ de\ salida + Núm.\ de\ ficheros^{12} + Núm.\ de\ Interfaces + Tamaño\ del\ proyecto) * Coste\ HH$			
RIM (Riesgos Indicados Materializados)	Calcula el número de riesgos detectados para cada proyecto y riesgos materializados (si ha derivado en un problema)	Mediante informes de vulnerabilidades de código (análisis estático), en el entorno de preproducción (análisis dinámico en PRE) y de producción (análisis dinámico en PRO)	Fiabilidad y Seguridad
$Riesgos = \frac{\sum Núm.\ de\ riesgos\ indicados}{\sum (Núm.\ riesgos\ indicados + Núm.\ de\ riesgos\ materializados)}$			
Sobreesfuerzo (SE)	Calcula en horas la cantidad adicional de trabajo que se realiza por encima del Trabajo Previsto (TRP)	Métricas obtenidas para cada una de las fases en relación a las fechas	Disponibilidad, Fiabilidad, Funcionalidad, Usabilidad y Mantenibilidad
$SE = TRR - TRP$			

¹² Núm. de ficheros: se refiere a cada grupo lógico de datos de usuario o información de control. Están excluidos los ficheros temporales.

TFA (Tiempo en fallos)	Núm. de minutos e donde la aplicación tiene fallos.	Métricas obtenidas en cada informe de verificación de cada proceso.	Fiabilidad y Mantenibilidad
$FIA (Fiabilidad) = \frac{(TOP - TFA)}{TOP}$			
TIN (Tiempo indisponible)	Núm. de minutos en los que el software no está disponible para su uso.	Métricas obtenidas en cada informe de verificación de cada proceso.	Disponibilidad, Mantenibilidad y Seguridad
$TIN = TFA - TMA$ $DIS (Disponibilidad) = \frac{TIN}{TOP}$			
TMA (Tiempo en mantenimiento)	Núm. de minutos dedicados al mantenimiento	Métricas obtenidas en cada informe de verificación de cada proceso.	Mantenibilidad y Usabilidad
$MAN (Mantenibilidad) = \frac{TMA}{TOP}$			
TOP (Tiempo en operación)	Núm. de minutos que la aplicación está operativa sin interrupciones.	Valor obtenido desde que el SW está en operación.	Funcionalidad y Mantenibilidad
$TOP = (Inicio Proceso Operación - Fin Proceso Operación) * 1.440$			
TRP (Trabajo Previsto (horas))	Estimación de horas que se espera invertir en el proyecto.	Métricas obtenidas para cada una de las fases en relación a las fechas.	Disponibilidad, Fiabilidad y Mantenibilidad
<i>Se calcula en base a las horas necesarias para cada perfil del equipo</i>			
TRR Trabajo Real (horas))	Cantidad de horas invertidas en cada una de las fases del ciclo de vida del software (SDLC)	Métricas obtenidas para cada una de las fases en relación a las fechas	Fiabilidad, Funcionalidad, Mantenibilidad y Usabilidad
<i>Se calcula en base a las horas necesarias para cada perfil del equipo</i>			

Tabla 3-4 Indicadores para la Evaluación del Ciclo de Vida del Proyecto (SDLC) (elaboración propia)

Los resultados de estos parámetros se reflejan en un cuadro resumen denominado “Evaluación del Ciclo de Vida del Proyecto (SDLC)”. Este cuadro proporciona datos muy relevantes sobre el SDLC del proyecto. Como ejemplo, consultar [el punto 4.1.1](#)

A diferencia de los indicadores operativos y de calidad del código fuente, la tabla de la Evaluación del Ciclo de Vida del Proyecto, no dispone de mínimos establecidos para determinar un valor concreto. Sin embargo, se toma como referencia el valor de 1, significando que el proyecto ha sido desarrollado según las previsiones y requisitos acordados. Si, por el contrario, está por debajo o por encima de 1, es indicativo de que el proyecto tiene puntos de mejora. Este análisis es esencial para mejorar la planificación y ejecución de futuros desarrollos.

Como ejemplos de los problemas que pudieran derivarse de que el valor no fuera 1, podrían ser:

- Retraso en la entrega del proyecto (PDP por debajo de 1): si este valor se encuentra por debajo es que la duración real ha superado a la duración prevista con lo que puede afectar a la fiabilidad y mantenibilidad ya que el software no estaría disponible a tiempo para la realización de pruebas u operación.

- Sobresfuerzo del equipo (PTD por debajo de 1): este valor implica que el trabajo real fue superior al previsto por lo que el equipo puede originar más errores. Afectaría a las capacidades de funcionalidad y mantenibilidad.

- Tiempo de inactividad (PRD por debajo de 1): si el PRD está por debajo de 1, significaría que el software no está disponible el tiempo necesario para operarlo correctamente, ya que pueden producirse fallos o un mantenimiento excesivo. Ello afectaría a la disponibilidad y a la fiabilidad.

Asimismo, los resultados obtenidos de las seis (6) métricas evaluadas tienen relación con las capacidades que son determinadas en la segunda tabla, estas son:

Evaluación del Ciclo de Vida del Proyecto (SDLC)	Afecta a
PDP (Previsión Duración Prevista)	Fiabilidad y Mantenibilidad
PTD (Precisión Trabajo Previsto)	Funcionalidad y Mantenibilidad
CPE (Conformidad)	Funcionalidad y Mantenibilidad
PRD (Producción)	Disponibilidad y Fiabilidad
NCP (Conformidad Verificaciones)	Mantenibilidad
RIM (Riesgos)	Seguridad

Tabla 3-5 Relación entre Indicadores Evaluación del Ciclo de Vida del Proyecto (SDLC) e Indicadores Operativos (elaboración propia)

La segunda tabla contiene indicadores que estiman la calidad operativa del proyecto, evaluando las capacidades como la funcionalidad, seguridad, disponibilidad, fiabilidad, usabilidad y mantenibilidad. Estos marcadores permiten comprobar si el proyecto cumple con todos los requisitos para el aseguramiento de la calidad una vez que el software ya está en el entorno de producción. Todos ellos están estrechamente relacionados entre sí y pueden influirse mutuamente. Poniendo un ejemplo, una usabilidad deficiente puede influir en la mantenibilidad, pero también en la funcionalidad. Cada uno de estos aspectos es necesario para determinar que el software sea robusto y eficiente.

Seguidamente, se exponen los indicadores operativos con los valores mínimos que deben cumplir el software.

Indicadores Operativos	Obtención	Valores mínimos
Funcionalidad (FUN)	Información proporcionada por el peticionario (RCON-Revisiones Conjuntas).	90%
Seguridad (SEG)	Información extraída del cuestionario completado por el gestor del proyecto.	90%
Disponibilidad (DIS)	$DIS = TIN / TOP$	95%
Fiabilidad (FIA)	$FIA = (TOP - TFA) / TOP$	80%
Usabilidad (USA)	Información extraída del cuestionario completado por el gestor del proyecto.	90%
Mantenibilidad (MAN)	$MAN = (TMA / TOP)$	80%

Tabla 3-6 Indicadores Operativos (elaboración propia)

Si estos indicadores están por debajo de los umbrales marcados, el proyecto podría sufrir diferentes problemas en su operación.

Los problemas que pudieran aparecer, podrían ser:

- Falta de requisitos requeridos (FUN): si el software no proporciona las funcionalidades requeridas puede afectar al usuario pudiendo producirse abandono del producto o la búsqueda de otra solución.

- Fallo de seguridad (SEG): si las métricas estuvieran por debajo del 90% del umbral mínimo, el software podría ser vulnerable a ataques como accesos no autorizados, explotación de vulnerabilidades, etc., comprometiendo la integridad y confidencialidad.

- Baja disponibilidad (DIS): si la disponibilidad del software no cumple con la expectativa mínimo de estar operativo del 95% pudiera ser que el sistema experimentara caídas frecuentes o tiempos de inactividad prolongados.

- Fiabilidad inadecuada (FIA): no disponer de la fiabilidad adecuada puede provocar fallos frecuentes y pérdida de datos.

- Usabilidad mínima (USA): si el software no dispone de una usabilidad mínima, puede afectar a la familiarización en el empleo del software, a la posibilidad de errores por parte de los usuarios y a la insatisfacción del mismo. Todo ello pudiera requerir la asistencia de un soporte técnico que aumentaría el coste y el tiempo necesario para la resolución de los problemas del usuario.

- Dificultad de mantenimiento (MAN): si la mantenibilidad es baja, el software podría generar costes de mantenimiento altos. Asimismo, también puede afectar a un aumento en los tiempos de respuesta.

Por último, la tercera tabla se basa en la calidad del código fuente recopilando los indicadores a través de la herramienta SonarQube. Estos indicadores incluyen aspectos como las líneas de código

duplicadas, cobertura de pruebas, comentarios, deuda técnica y vulnerabilidades y evalúan la calidad del código fuente.

En la tabla, se muestran los umbrales requeridos para el cumplimiento de los indicadores de calidad del código fuente.

Indicadores de Calidad del Código Fuente (SonarQube)	Valores
Líneas duplicadas	< 10%
Cobertura	> 50%
Comentarios	> 40%
Deuda técnica	< 5%
Vulnerabilidades	=< 3

Tabla 3-7 Indicadores de Calidad del Código Fuente (SonarQube) (elaboración propia)

Si estos indicadores están por debajo de los umbrales marcados, el proyecto podría sufrir diferentes problemas en su operación. Estos valores afectan a la efectividad, mantenibilidad, escalabilidad y seguridad. Como ejemplo se menciona:

- Líneas duplicadas: que el código fuente sea <10%, significa que está mal estructurado, lo que puede afectar a la mantenibilidad. A la hora de realizar cambios y actualizaciones puede aumentar el riesgo de errores e inconsistencias.

- Cobertura: disponer de una cobertura de pruebas inferior al 50% reduce la efectividad y la confiabilidad del software ya que no se ha evaluado el código fuente con todas las pruebas necesarias.

- Comentarios: proporciona la información necesaria para que el código sea entendible y pueda mantenerse por quien no lo haya desarrollado.

- Deuda técnica: esta métrica representa el coste que representa reescribir partes del código fuente que no cumplen los estándares adecuados. Afecta a la escalabilidad y mantenibilidad.

- Vulnerabilidades: las vulnerabilidades se clasifican según su impacto, facilidad de explotación y exposición como vulnerabilidades críticas, altas, medias, bajas e informativas. Para este estudio el valor de =<3 corresponde a vulnerabilidades altas/medias.

Los resultados de los parámetros obtenidos junto con los tres (3) cuadros resumen de los diez (10) casos de uso se encuentran en el [Anexo I](#).

3.2.3.2 Indicadores Cualitativos

A continuación, se analizan los conceptos cualitativos básicos como el talento, experiencia, conocimiento, resolución de problemas complejos, pensamiento crítico, adaptabilidad y comunicación efectiva, entre otros. Estos elementos son esenciales para evaluar el rendimiento y eficacia de los equipos pudiendo proporcionar información sobre la capacidad de enfrentarse a situaciones imprevistas, adaptación a entornos cambiantes y a la colaboración en grupo. Estos factores influyen directamente en el éxito y en los objetivos establecidos para un proyecto.

Para obtener estos indicadores, he elaborado un cuestionario de sesenta (60) preguntas con los conceptos mencionados y fue realizado por un total de doce (12) responsables de la gestión de proyectos, entre los que destaca el gestor de proyecto y el analista, por cada uno de los equipos. Lo que se pretende con estos indicadores es la evaluación de los equipos de desarrollo según la percepción de sus responsables.

Aunque la muestra no es extensa, permite examinar los conceptos mencionados pudiendo hacer una valoración e interpretación de los mismos.

En la valoración de estos indicadores, se ha desarrollado un cuadro resumen titulado “Indicadores Cualitativos”, en el que se asigna un peso específico a cada una de las preguntas formuladas, siendo 1 el valor mínimo (muy poco/muy baja) y 5 para el valor máximo (mucho/muy buena). El resultado será la media aritmética de sus respuestas. Esta valoración permitirá la comparación entre el equipo interno (*in-house*) y el externo (*outsourcing*).

El cuestionario y los indicadores cualitativos se encuentran en el [Anexo II](#).

4 RESULTADOS

4.1 Análisis comparativo de Resultados

Una vez realizada la obtención, recopilación, integración y evaluación de las métricas objeto de este estudio, se comparan, analizan e interpretan los datos organizados por cada uno de los cuatro (4) cuadros resúmenes elaborados.

Cada uno de los indicadores cuantitativos y cualitativos tienen relación con los cuadros resumen. Sin embargo, es importante señalar que, una única métrica o indicador no corresponde específicamente a una determinada capacidad. A modo de ejemplo, para determinar la funcionalidad, no solamente se contemplan los indicadores como son CRC Proceso, PTP (Precisión del Esfuerzo), RFC (Solicitud de Cambios), Sobreesfuerzo, TOP (Tiempo en operación) y TRR (Trabajo real), contemplados en este estudio, sino que intervienen muchos otros indicadores que, debido a la falta de métricas adecuadas, no han sido estimadas.

Este enfoque permite una visión general y detallada sobre la viabilidad del proyecto, la capacidad operativa, la calidad del código fuente y los indicadores cualitativos, muchos más subjetivos que el resto de los indicadores, pero no por ello, excluibles de este estudio.

4.1.1 Resumen Indicadores Cuantitativos y Cualitativos

A continuación, se muestra un cuadro resumen general con los indicadores cuantitativos y cualitativos de los equipos internos (in-house) y equipos externos (outsourcing).

RESUMEN PROYECTOS, EQUIPOS INTERNOS (IN-HOUSE)									
INDICADORES CUANTITATIVOS						4) INDICADORES CUALITATIVOS			
1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)					
PDP	1,03	FUN	100,0%	LÍNEAS DUPLICADAS	7,9%	TALENTO	47,20		
PTP	1,01	SEG	96,6%	COBERTURA	41,6%	EXPERIENCIA	46,40		
CPE	1,00	DIS	98,6%	COMENTARIOS	30,6%	CONOCIMIENTO	45,80		
PRD	0,97	FIA	99,5%	DEUDA TÉCNICA	0,4%	RES. PROBLEMAS COMPLEJOS	46,00		
NCP	0,98	USA	93,2%	VULNERABILIDADES	2	PENSAMIENTO CRÍTICO	47,40		
RIM	0,00	MAN	99,1%			ADAPTABILIDAD	46,80		
						COMUNICACIÓN EFECTIVA	47,60		
						CREATIVIDAD	45,00		
						COLABORACIÓN	47,60		
						GESTIÓN DEL TIEMPO	44,80		
						RESPONSABILIDAD	50,80		

ORIENTACIÓN AL DETALLE	46,20
TOTAL	46,80

Tabla 4-1 Resumen Indicadores del Equipo Interno (elaboración propia)

RESUMEN PROYECTOS, EQUIPOS EXTERNOS (OUTSOURCING)									
INDICADORES CUANTITATIVOS						4) INDICADORES CUALITATIVOS			
1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)					
PDP	1,15	FUN	100,0%	LINEAS DUPLICADAS	7,6%	TALENTO	48,00		
PTP	1,11	SEG	94,9%	COBERTURA	45,6%	EXPERIENCIA	46,20		
CPE	1,00	DIS	94,3%	COMENTARIOS	29,0%	CONOCIMIENTO	48,60		
PRD	1,00	FIA	98,1%	DEUDA TÉCNICA	0,1%	RES. PROBLEMAS COMPLEJOS	48,00		
NCP	0,99	USA	95,9%	VULNERABILIDADES	2	PENSAMIENTO CRÍTICO	43,80		
RIM	0,00	MAN	95,2%			ADAPTABILIDAD	49,20		
						COMUNICACIÓN EFECTIVA	47,40		
						CREATIVIDAD	49,20		
						COLABORACIÓN	48,60		
						GESTIÓN DEL TIEMPO	45,60		
						RESPONSABILIDAD	52,20		
						ORIENTACIÓN AL DETALLE	49,20		
						TOTAL	48,00		

Tabla 4-2 Resumen Indicadores del Equipo Externo (elaboración propia)

Partiendo de los datos obtenidos de estos cuadros resumen y para efectuar un mejor análisis e interpretación de los indicadores a comparar, se fracciona la información acorde a las siguientes tablas:

4.1.2 Indicadores Evaluación del Ciclo de Vida del Proyecto (SDLC)

Según el gráfico representado a continuación, se han comparado seis (6) valores obtenidos en relación a métricas durante todo el ciclo de vida del desarrollo (SDLC).

1) Evaluación del Ciclo de Vida del Proyecto (SDLC)	In-House	Outsourcing
PDP	1,03	1,15
PTP	1,01	1,11
CPE	1	1
PRD	0,97	1
NCP	0,98	0,99
RIM	0	0

Tabla 4-3 Resumen Indicadores Evaluación del Ciclo de Vida del Proyecto (SDLC) (elaboración propia)

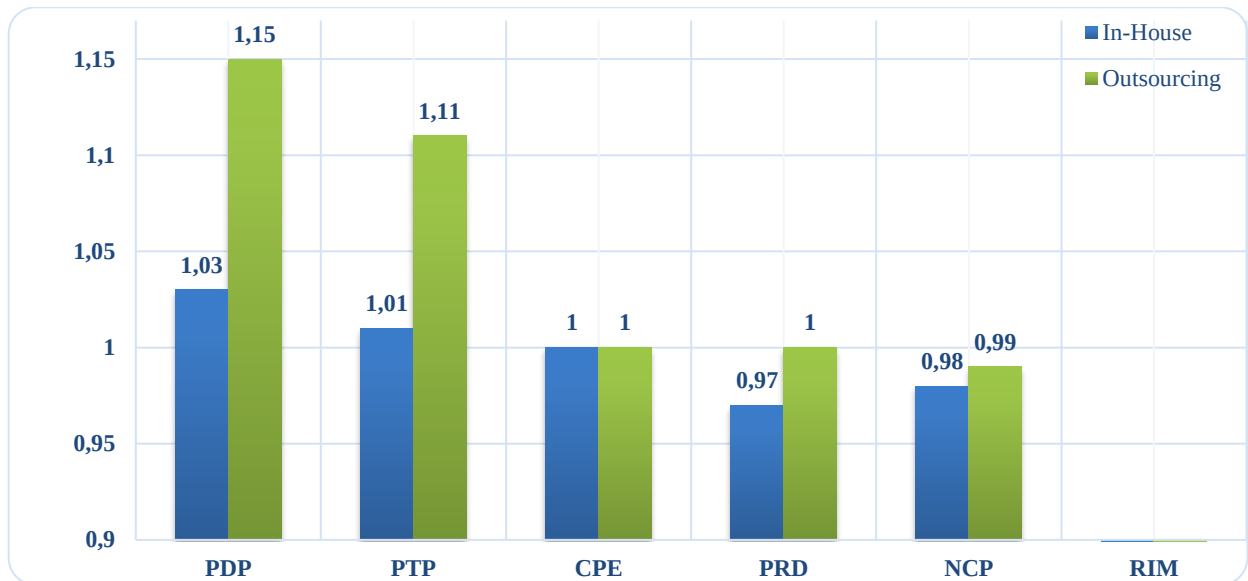


Figura 4-1 Interpretación Indicadores Evaluación del Ciclo de Vida del Proyecto (SDLC) (elaboración propia)

Cabe recordar que los valores que están alrededor de 1, indican un rendimiento equilibrado. A continuación, se interpretan los resultados obtenidos:

- PDP (Precisión Duración Prevista): el equipo interno (*in-house*) está más cerca del valor ideal de 1, con una ligera desviación por encima lo que puede significar que ha empleado mayor tiempo en las fases o un sobreesfuerzo. En cambio, el equipo externo (*outsourcing*) está muy por encima, lo que sugiere una peor precisión en la planificación al emplear mayor tiempo del SDLC.

- PTP (Precisión del Esfuerzo): con una métrica de 1,01 el equipo interno está en el punto ideal, lo que denota un planeamiento efectivo acorde al trabajo real y trabajo previsto. Sin embargo, el equipo externo está por encima de la métrica ideal lo que puede indicar inexactitud en la estimación de esfuerzos.

- CPE (CRC PROCESO): ambos equipos ajustan a 1, lo que quiere decir que los parámetros de las conformidades y no conformidades están acorde con las funcionalidades y los requisitos requeridos por el cliente.

- PRD (Complejidad, puntos de función de Albrecht): ambos equipos están en parámetros cercanos al 1 lo que indica una correcta estimación del esfuerzo y funcionalidades.

- NCP (RFC): ambos equipos se encuentran cercanos al valor de 1, siendo la diferencia mínima entre ambos. Ello sugiere que ambos equipos han gestionado eficazmente las modificaciones solicitadas por el cliente.

4.1.3 Indicadores Operativos

En este segundo cuadro resumen, para mantener las mismas medidas que entre todos los indicadores cuantitativos, se toma como referencia el valor ideal de 1, procediendo a normalizar los valores obtenidos (aplicación de la fórmula de normalización directa).

A continuación, se explica el proceso de normalización aplicado:

- Funcionalidad (FUN): deben ser mayor al 90%. Valores: 90 (valor mínimo) y 100 (valor máximo)..

- Seguridad (SEG): debe ser mayor de 90%. Valores: 90 (valor mínimo) y 100 (valor máximo).
- Disponibilidad (DIS): debe ser mayor de 95%. Valores: 95% (valor mínimo) y 100% (valor máximo).
- Fiabilidad (FIA): debe ser mayor de 80%. Valores: 80% (valor mínimo) y 100% (valor máximo).
- Usabilidad (USA): debe ser mayor de 90%. Valores: 90 (valor mínimo) y 100 (valor máximo).
- Mantenibilidad (MAN): debe ser mayor de 80%. Valores: 80% (valor mínimo) y 100% (valor máximo).

A continuación, se presenta la fórmula utilizada:

$$\text{Fórmula de normalización directa} = \frac{(\text{valor actual} - \text{valor mínimo})}{(\text{valor máximo} - \text{valor mínimo})}$$

Para mayor comprensión, se adjunta en la siguiente tabla la aplicación de la fórmula de normalización directa con los resultados que reflejará el gráfico:

2) Indicadores Operativos	In-House	Outsourcing	Mínimo	Máximo	In-House	Outsourcing
FUN	100%	100%	90%	100%	1	1
SEG	96,6%	94,9%	90%	100%	0,66	0,49
DIS	98,6%	94,3%	95%	100%	0,72	-0,14
FIA	99,5%	98,1%	80%	100%	0,975	0,905
USA	93,2%	95,9%	90%	100%	0,32	0,59
MAN	99,1%	95,2%	80%	100%	0,955	0,76

Tabla 4-4 Resumen Indicadores Operativos (elaboración propia)

A continuación, se expone el gráfico de este segundo cuadro resumen, obteniendo los siguientes valores con respecto a las capacidades operativas para su interpretación:

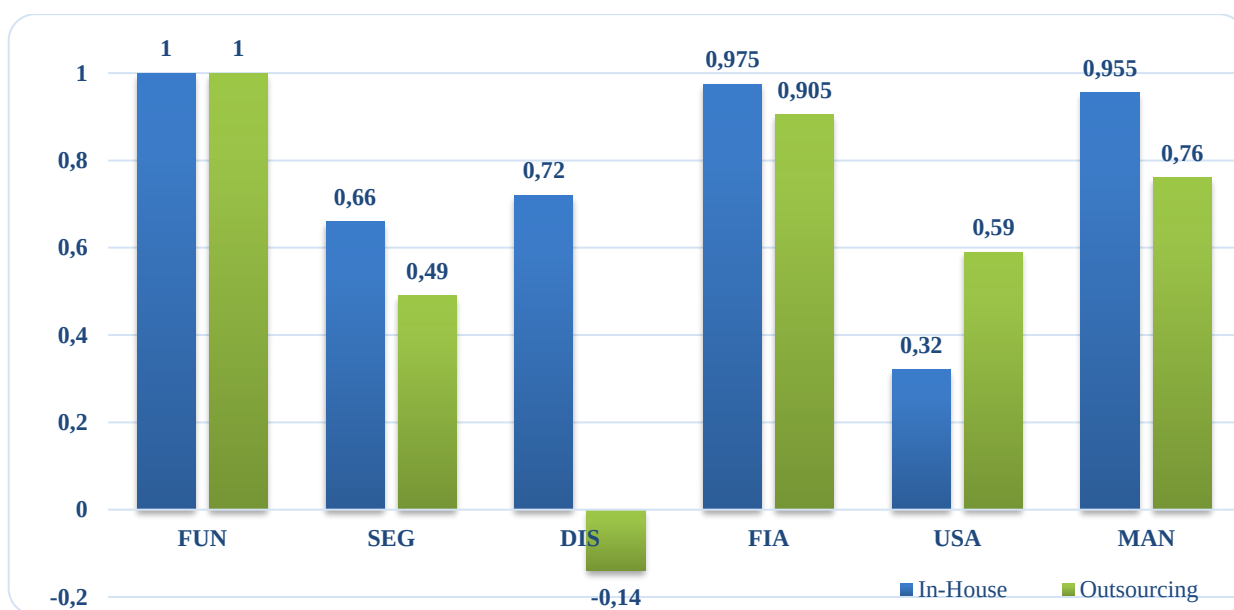


Figura 4-2 Interpretación Indicadores Operativos (elaboración propia)

- Funcionalidad (FUN): ambos equipos consiguen el máximo valor de 1. Esto es debido a que todas las funcionalidades y requisitos del cliente son satisfechos en los nuevos desarrollos. No ocurre así, cuando se trata de mantenimientos donde sí podría variar esos valores.

- Seguridad (SEG): se observa que hay una pequeña diferencia a favor del equipo interno, lo que puede ser debido a la implementación de medidas de seguridad a lo largo del SDLC. Esto no exime para que se sigan implementando la seguridad lo máximo posible. Aunque el equipo externo dispone de un buen nivel de seguridad puede convertirse en relevante si se manejan datos sensibles.

- Disponibilidad (DIS): a pesar de no disponer del valor ideal, el equipo interno está orientado a una alta disponibilidad y acceso de los usuarios al sistema, aunque hay posibilidad de mejora. En cuanto al equipo externo los valores muestran que no cumple con el valor mínimo lo que indica que existe mayor tiempo de inactividad, producido porque el software no está funcionando correctamente o bien carece de resiliencia ante problemas o fallos. Esto afecta negativamente a la experiencia del usuario.

- Fiabilidad (FIA): el equipo interno destaca con un alto grado de fiabilidad lo que denota que tanto el rendimiento como la operación no tienen interrupciones significativas. En cuanto al equipo externo, su rendimiento es inferior al del equipo interno, quizás derivado de interrupciones por fallos o mantenimiento.

- Usabilidad (USA): el equipo interno refleja buen diseño y usable por los usuarios, sin embargo, es inferior al alcanzado por el equipo externo. Para obtener mayor valor, podría mejorar el diseño, uso y personalización para que fuera más amigable para los usuarios.

- Mantenibilidad (MAN): en estos valores destaca con creces el equipo interno ya que tiene optimizado el código, los mantenimientos y las actualizaciones. Aun así, hay posibilidad de mejora continua. Por otro lado, el equipo externo, aunque cumple con parámetros, a futuro pudiera tener problemas para el mantenimiento, traducido a costes altos y tiempo de realización.

4.1.4 Indicadores de Calidad del Código Fuente (SonarQube)

En cuanto al tercer cuadro, se muestran aquellos indicadores que proporcionan datos relevantes que aseguran la calidad del código fuente. Para ello, al igual que en los apartados anteriores, se han normalizado los valores obtenidos al valor ideal de 1.

Para ello, es determinante señalar que para cada indicador:

- Líneas duplicadas: debe ser menor de 10%. Valores: 0% (valor mínimo) y 10% (valor máximo). Se aplica la fórmula de normalización inversa.

- Cobertura: debe ser mayor de 50%. Valores: 50% (valor mínimo) y 100% (valor máximo). Se aplica la fórmula de normalización directa.

- Comentarios: debe ser mayor de 40%. Valores: 40% (valor mínimo) y 100% (valor máximo). Se aplica la fórmula de normalización directa.

- Deuda técnica: debe ser menor de 5%. Valores: 0% (valor mínimo) y 5% (valor máximo). Se aplica fórmula de normalización inversa.

- Vulnerabilidades: deben ser menores o iguales a 3 (≤ 3). Valores: 0 (valor mínimo) y 3 (valor máximo). Se aplica la fórmula de normalización inversa.

A continuación, se presentan las fórmulas utilizadas:

$$\text{Fórmula de normalización directa} = \frac{(\text{valor actual} - \text{valor mínimo})}{(\text{valor máximo} - \text{valor mínimo})}$$

$$\text{Fórmula de normalización inversa} = 1 - \frac{(\text{valor actual} - \text{valor mínimo})}{(\text{valor máximo} - \text{valor mínimo})}$$

Asimismo, es importante destacar que para asegurar la calidad del código fuente, los porcentajes de cobertura y comentarios tienen que ser valores altos ya que reflejan que el código está bien documentado y con buena cobertura en pruebas. Por el contrario, los indicadores de líneas duplicadas, deuda técnica y vulnerabilidades cuanto menor sean sus valores, más óptimo será el código fuente.

Para mayor comprensión, en la siguiente tabla se adjunta la aplicación de la fórmula de normalización directa e inversa:

3) Indicadores de Calidad del Código Fuente (SonarQube)	In-House	Outsourcing	Mínimo	Máximo	In-House	Outsourcing
Líneas duplicadas	7,9%	7,6%	0%	10%	0,21	0,24
Cobertura	41,6%	45,6%	50%	100%	-0,17	-0,09
Comentarios	30,6%	29,0%	40%	100%	-0,16	-0,18
Deuda técnica	0,4%	0,1%	0%	5%	0,92	0,98
Vulnerabilidades	2	2	0%	3%	0,33	0,33

Tabla 4-5 Resumen Indicadores de Calidad del Código Fuente (SonarQube) (elaboración propia)

Dados los valores anteriores, se muestra el siguiente gráfico para su interpretación:

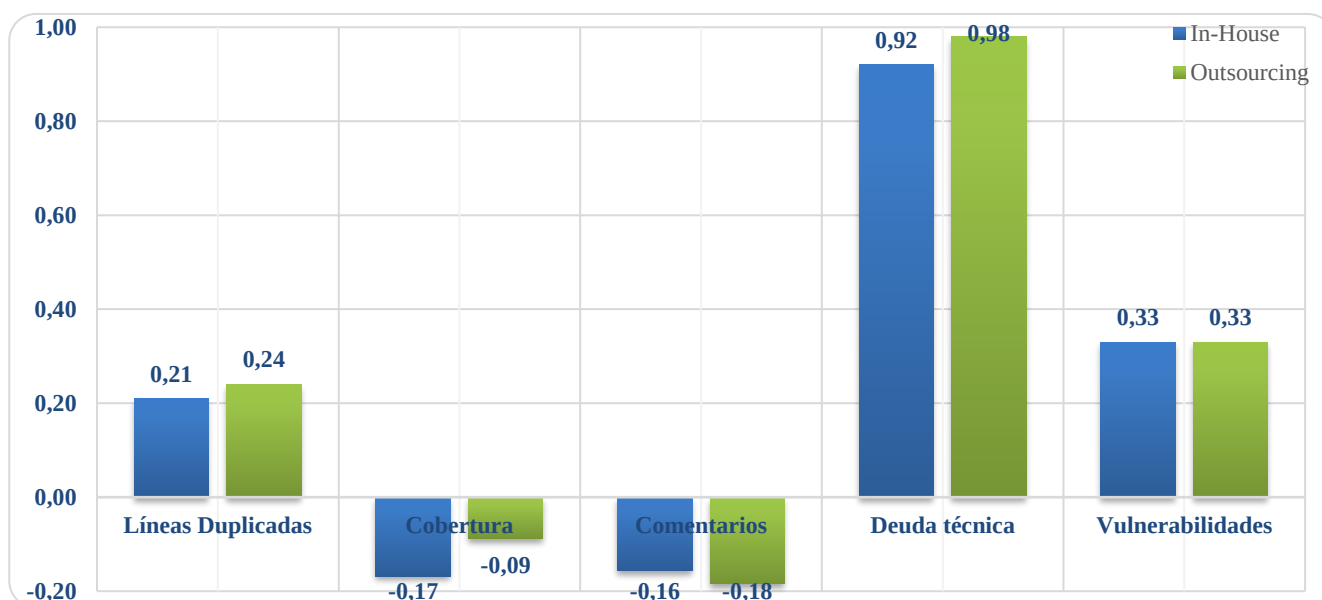


Figura 4-3 Interpretación Indicadores de Calidad del Código Fuente (SonarQube) (elaboración propia)

- Líneas duplicadas: el código fuente de ambos equipos están dentro de los límites requeridos. Sin embargo, los equipos externos tienen una ligera ventaja frente a los equipos internos.

- Cobertura: ambos equipos no alcanzan el valor requerido del 50% de cobertura. Esta baja cobertura se traduce en la necesidad de hacer más pruebas al código para prevenir fallos en el futuro.

- Comentarios: al igual que el anterior indicador, ambos equipos no alcanzan el valor requerido del 40%. Esto implica la dificultad para realizar mantenimientos a futuro, dado que no se ha comentado en el código fuente lo que realiza realmente el software y, es muy probable que tampoco dispusiera de documentación.

- Deuda técnica: el equipo interno y el equipo externo destacan en esta métrica, siendo que asegura un código más fácil de mantener a largo plazo.

- Vulnerabilidades: ambos equipos tienen el mismo valor. Aun así, las medidas de seguridad deben reducirse para que presente el valor más bajo.

4.1.5 Indicadores Cualitativos

Este cuarto cuadro resumen analiza los conceptos cualitativos que son esenciales para evaluar el rendimiento y la eficacia de los equipos de desarrollo.

En este cuestionario de sesenta (60) preguntas se han incluido 12 parámetros como el talento, experiencia, conocimiento, resolución de problemas complejos, pensamiento crítico, adaptabilidad, comunicación efectiva, creatividad, colaboración, gestión del tiempo, responsabilidad y orientación al detalle.

Asimismo, existen otros métodos complementarios como son la observación directa y las entrevistas. Sin embargo, dado que la comparación se realiza una vez que los proyectos confluyen en el Ejército del Aire y del Espacio en el momento del despliegue del software (se conoce familiarmente como “producto de llave en mano”), ambas valoraciones no se han podido realizar dado que los equipos externos están dirigidos por diferentes empresas externas a nuestra organización.

Para mayor comprensión, se adjunta la siguiente tabla con los resultados obtenidos para cada equipo:

4) Indicadores Cualitativos	In-House	Outsourcing
Talento	47,20	48,00
Experiencia	46,40	46,20
Conocimiento	45,80	48,60
Res. Problemas Complejos	46,00	48,00
Pensamiento Critico	47,40	43,80
Adaptabilidad	46,80	49,20
Comunicación Efectiva	47,60	47,40
Creatividad	45,00	49,20
Colaboración	47,60	48,60
Gestión del Tiempo	44,80	45,60
Responsabilidad	50,80	52,20
Orientación al detalle	46,20	49,20
TOTAL	46,80	48,00

Tabla 4-6 Resumen Indicadores Cualitativos (elaboración propia)

Como continuación, a la tabla anterior, se expone el gráfico para su interpretación:

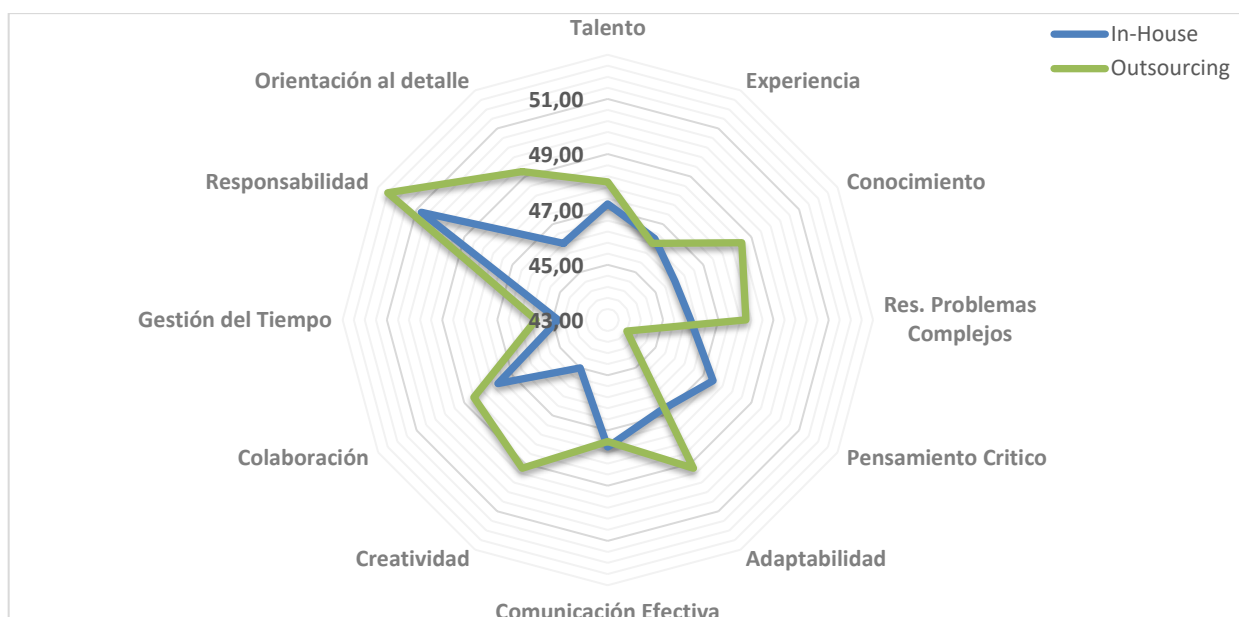


Figura 4-4 Interpretación Indicadores Cualitativos (elaboración propia)

Como resultado de esta evaluación cualitativa, los equipos de desarrollo interno (*in-house*) y externo (*outsourcing*) ofrecen fortalezas y debilidades en distintos indicadores. La media aritmética de las puntuaciones de los ambos equipos, no presenta excesivas variaciones, con un promedio de 46,8 del equipo interno frente a un 48 de promedio del equipo externo. Es, por lo que, dependiendo de las mismas, los equipos pueden orientarse a objetivos y demandas concretos en el desarrollo del software.

En el gráfico se puede observar que el equipo interno sobresale, en pensamiento crítico. Por su parte, el equipo externo destaca en talento, conocimientos, resolución de problemas, adaptabilidad, creatividad, orientación al detalle y responsabilidad. Los parámetros de experiencia, comunicación efectiva y colaboración son bastantes similares en ambos equipos.

En un principio, al observar los resultados, sin llegar a ser una muestra extensa, permitía examinar los indicadores para su valoración con una participación de 24 perfiles. Sin embargo, el cuestionario debería contener de 30 a 50 cuestionarios por grupo. Esta cantidad está basada en un nivel de confianza del 95%, con una desviación estándar estimada de 0,5. Para ello, sería necesario disponer de, al menos, 100 cuestionarios para compensar posibles pérdidas de datos y aumentar la precisión de los resultados.

$$\text{Tamaño mínimo muestra para cada grupo } (n) = 2 \frac{(Z \sigma)^2}{d^2}$$

donde Z es el nivel de confianza (95%), σ es la desviación estándar y d es el margen de error.

Un análisis adicional revela que los valores obtenidos del equipo externo, carecen de puntuaciones en el valor mínimo y en el segundo valor mínimo (Muy poco/Muy baja y Poco/Baja), lo que hace suponer una tendencia a sobreestimar sus capacidades. Es por ello, por lo que los valores obtenidos difieren de la realidad y, por ello, no parecen del todo congruentes ni del todo fiables.

5 CONCLUSIONES Y LÍNEAS FUTURAS

5.1 Conclusiones

El estudio comparativo del impacto de los equipos de desarrollo en el ciclo integral de las aplicaciones se puede resumir en los siguientes puntos:

1. Indicadores Evaluación del Ciclo de Vida del Proyecto (SDLC): los equipos internos destacan frente al equipo externo en una planificación más detallada y un mejor control del esfuerzo (PTP), aunque tienen margen de mejora en la previsión de la complejidad del desarrollo (PRD), obteniendo métricas cercanas al valor 1. En cuanto al equipo externo, dispone de buenas métricas en el CPE y PRD, lo que indica una buena planificación y estimación en la complejidad del software.

2. Indicadores Operativos: aunque los equipos externos aventajan al equipo interno en usabilidad, los equipos internos sobresalen en la mayoría de las capacidades evaluadas como son funcionalidad, seguridad, disponibilidad, fiabilidad y mantenibilidad, lo que les convierte en una buena opción para proyectos a largo plazo y con una estabilidad operativa superior al equipo externo.

3. Indicadores de Calidad del Código Fuente (SonarQube): se ha considerado que otros indicadores claves son los indicadores del código fuente. Si bien, las métricas relativas a la cobertura y los comentarios se encuentran muy cerca de valores mínimos en ambos equipos y tienen que ser optimizadas, los equipos internos desarrollan un código de mayor calidad con menor líneas duplicadas y deuda técnica. Por su parte, los equipos externos, aparte de lo expresado, presentan margen de mejora en la deuda técnica y en las líneas duplicadas.

4. Indicadores Cualitativos: los equipos internos destacan en pensamiento crítico y colaboración, lo que les convierte en una opción adecuada para proyectos que requieren un análisis exhaustivo y buena planificación. Por su parte, los equipos externos disponen de una mayor adaptabilidad y creatividad, lo que les adecúa para los proyectos de innovación continua.

Según este estudio, como resultado de la selección de indicadores cuantitativos y cualitativos, los equipos internos destacan en el desarrollo de software. Esto es debido a varios factores que lo hacen posible, como su extenso conocimiento técnico de funcionalidades del software, que permite comprender en profundidad la lógica del software. Además, disponen de un mayor control y planificación de procesos y tareas, lo que se traduce en un rendimiento óptimo tanto en tiempo como en recursos. Asimismo, una comunicación fluida, facilita la coordinación entre el equipo y ayuda a minimizar errores de interpretación. Otro factor importante es el compromiso a largo plazo o fidelización, ya que se alinean los intereses, valores y cultura de la organización en el equipo desarrollador. Por último, y esencial, la implementación de mayor nivel de seguridad reduce la exposición a vulnerabilidades, accesos a datos sensibles y propiedad intelectual. Por el contrario, la externalización, puede provocar una falta de resiliencia organizacional debido a la gran dependencia de los servicios proporcionados por los proveedores, lo que puede convertir una organización en una entidad cautiva dependiendo de terceros para el soporte, mantenimiento y actualizaciones.

Asimismo, aunque el resultado de la evaluación cualitativa no ha sido lo concluyente que se esperaba. Esto es debido a los índices de participación y a la falta de puntuaciones mínimas en algunas áreas del equipo externo, que son aspectos críticos para el éxito de un equipo de desarrolladores. Aun así, estas capacidades siguen siendo fundamentales, ya que pueden proporcionar una visión más amplia sobre la calidad y el éxito del proyecto, que no se puede adquirir a través de las métricas cuantitativas.

No obstante, parece que no hay un enfoque único para el desarrollo de los proyectos, sino que todo depende de la naturaleza del mismo, pudiendo delimitarlos a proyectos de largo plazo y de proyectos innovadores. En cuanto a los primeros, el equipo idóneo es el equipo interno debido a que requieren alta estabilidad, seguridad y mantenimiento. Además, por las métricas obtenidas, gestionan y destacan en las capacidades operativas y en el aseguramiento de la calidad. Por otro lado, en los proyectos innovadores, con entregas rápidas, los equipos externos pueden aportar más valor que los equipos internos.

5.2 Líneas futuras

Como continuación a las conclusiones y relacionadas con este estudio, se pueden establecer diversas líneas para futuros estudios:

- Colaboración híbrida entre equipos: el estudio podría investigar la colaboración entre equipos internos y externos para aprovechar las fortalezas de cada uno de ellos, minimizando sus debilidades. Esto incluiría cómo se pueden gestionar la comunicación entre equipos, la integración de procesos y conocimientos para alcanzar un rendimiento óptimo.

- Innovación y Creatividad: la innovación y la creatividad son fortalezas de los equipos externos. Estudios futuros podrían centrarse en fomentar estas capacidades en los equipos internos. Esto podría dar lugar a grupos de trabajos colaborativos donde las nuevas tecnologías, metodologías y conocimiento permitan maximizar la eficiencia para el desarrollo de software.

- Inteligencia Artificial y Automatización en la Evaluación de Calidad: en futuros estudios se podría explorar la implantación de la inteligencia artificial (IA) y la automatización en la evaluación de la calidad tanto en las métricas cuantitativas como en las métricas cualitativas. Esta implantación evaluaría parámetros adicionales a los expuestos como: la estimación del proyecto durante su ciclo de vida, la calidad del código, la deuda técnica, etc. en tiempo real. Además, se podrían desarrollar algoritmos predictivos para problemas futuros y evaluar las decisiones tomadas durante todo el ciclo de vida del desarrollo.

- Metodologías Ágiles en Equipos Internos: el estudio podría investigar la efectividad en la implantación de metodologías ágiles en los equipos internos. Esto incluiría el análisis de estas metodologías desde diferentes puntos de vista como la productividad, la calidad del software, la respuesta ante cambios, la satisfacción del cliente, etc. para adaptarlo al equipo interno.

6 BIBLIOGRAFÍA

- [1] Myrian C. González-Rabanal. *Gestión de Crisis y Gestión de Proyectos en un Entorno Volátil, Incierto, Cambiante y Ambíguo (VUCA)*. Editorial Dykinson. 2023.
- [2] «Web de Lelia Zapata» [En línea]. Available: <https://leliazapata.com/liderar-en-un-entorno-vuca> [Último acceso: 08 octubre 2023].
- [3] Gary Hamel with Bill Breen. *The Future of Management*. Harvard Business School Press. 2007.
- [4] «Web de LinkedIn» [En línea]. Available: <https://es.linkedin.com/pulse/el-nuevo-contexto-bani-marcelo-pablo-mercs> [Último acceso: 09 noviembre 2022].
- [5] Arthur W. Burks. *The First Computers: History and Architectures*. MIT Press. 2000.
- [6] «Web de Línea de Tiempo» [En línea]. Available: <https://lineadeltiempo.net/linea-del-tiempo-de-las-computadoras-2022-2023/> [Último acceso: 09 julio 2023].
- [7] Jean E. Sammet. *Programming Languages: History and Fundamentals*. Prentice Hall. 1969.
- [8] «Web Revista ON OFF» [En línea]. Available: <https://www.revistaonoff.es/lenguaje-de-programacion-una-historia-con-mas-de-200-anos/> [Último acceso: 01 septiembre 2022].
- [9] Andrew S. Tanenbaum. *Modern Operating Systems*. Pearson. 2001.
- [10] «Web Necesarioparaproba» [En línea]. Available: <http://necesarioparaprobar.blogspot.com/2018/11/tipos-de-licencias-de-software.html> [Último acceso: 01 septiembre 2022].
- [11] «Web de Kinsta» [En línea]. Available: <https://kinsta.com/es/estadisticas-ingenieria-software/> [Último acceso: 07 noviembre 2018]
- [12] «Web de Normas ISO» [En línea]. Available: <https://normasiso.org/norma-iso-25010/> [Último acceso: 24 septiembre 2023]
- [13] «Web de Atlassian» [En línea]. Available: <https://www.atlassian.com/es/devops> [Último acceso: 14 septiembre 2024]
- [14] Steve McConnell. *Software Project Survival Guide*. Microsoft Press. 1997.
- [15] Roger S. Pressman. *Software Engineering: A Practitioner's Approach*. McGraw-Hill. 2014.
- [16] Ian Sommerville. *Software Engineering*. Pearson. 2015.
- [17] Martin Fowler. *Refactoring: Improving the Design of Existing Code*. Addison-Wesley. 1999.
- [18] Kim D. B. y A. T. G. *Software Deployment: Principles, Processes and Best Practices*. Wiley. 2019.
- [19] Roy A. Miller. *Software Development: A Unified Framework*. Wiley. 1999.

- [20] «Web de Drupal Groups» [En línea]. Available: <https://groups.drupal.org/node/141589> [Último acceso: 12 abril 2011]
- [21] Alan C. Gillies. *Software Quality: A Guide to the SQA Process*. Springer. 2005.
- [22] <https://www.testingit.com.mx/blog/atributos-de-calidad-de-software>
- [23] Harold Kerzner. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*. Wiley. 2017.
- [24] David I. Cleland-Huang y otros. *Software Project Management in a Changing World*. Springer. 2015.
- [25] Ravi K. P. Sinha y Preeti Sinha. *Global Outsourcing: Executing an Onshore, Nearshore, and Offshore Strategy*. Prentice Hall. 2018.
- [26] Martin Fowler. *Patterns of Enterprise Application Architecture*. Addison-Wesley. 2002.
- [27] «Web de Cynoteck» [En línea]. Available: <https://cynoteck.com/es/blog-post/top-software-development-models-to-choose-from/> [Último acceso: 09 mayo 2022]
- [28] «Web de IBM» [En línea]. Available: <https://www.ibm.com/es-es/topics/workflow> [Último acceso: 23 mayo 2024]
- [29] «Web de Bscdesigner» [En línea]. Available: <https://bscdesigner.com/es/metricas-de-complejidad.htm> [Último acceso: 01 julio 2020]
- [30] «Web de CESTIC» [En línea]. Available: <https://www.defensa.gob.es/cectic/noticias/2023/listado/noticia1.html> [Último acceso: 12 julio 2023]
- [31] «Web de Wikipedia» [En línea]. Available: <https://es.wikipedia.org/wiki/Jira> [Último acceso: 05 agosto 2024]
- [32] «Web de Sentries» [En línea]. Available: <https://sentries.io/blog/que-es-sonarqube/> [Último acceso: 15 diciembre 2021]
- [33] Peter G. Northouse. *Leadership: Theory and Practice*. Sage Publications. 2018.
- [34] Beverly Baker. *Software Engineering Metrics: An Overview*. Wiley. 2006
- [35] Fenton, N. E., & Pfleeger, S. L. *Software Metrics: A Rigorous & Practical Approach*. PWS Publishing Company. 1997

ANEXO I: MÉTRICAS CUANTITATIVAS

A1-1 INVENTIC

PDP - PRECISIÓN DE LA DURACIÓN					PTP - PRECISIÓN DEL ESFUERZO				CPE - CONFORMIDAD					PRD - PRODUCCIÓN	
	DUP [días] Duración Prevista	DUR [días] Duración Real	Demora del Usuario	PDP Precisión Duración	TRP [horas] Trabajo Previsto	TRR [horas] Trabajo Real	Sobreesfuerzo	PTP Prec. Trabajo Previsto		CON	NCO	NCO COND.	CRC PROCESO	Complejidad (PFA)	Esfuerzo (H)
PETICIÓN	15,00	18,00	0,00	1,20	149,50	65,50	0,00	0,44	PETICIÓN	1	0	0	1,00	2.845	3.026,89
A+D	309,00	307,00	0,00	0,99	1.184,50	1.023,00	0,00	0,86	A+D	4	0	0	1,00	0,94	
C+P	315,00	318,00	0,00	1,01	4.156,00	4.187,00	0,00	1,01	C+P	1	0	0	1,00		
PROD.	243,00	244,00	0,00	1,00	310,00	323,00	0,00	1,04	PROD.	2	0	0	1,00		
SERVICIO	96,00	39,00	0,00	0,41	1.998,00	2.018,00	0,00	1,01	SERVICIO	1	0	0	1,00		
Proyecto	978,00	926,00	0,00	0,95	7.798,00	7.616,50	0,00	0,98	Proyecto	9	0	0	1,00		

RFC - CONFORMIDAD DE LA VERIFICACIÓN									
RFC	RFC0		RFC1		RFC2		RFC3		TOTAL
METRICA	CON	NCO	CON	NCO	CON	NCO	CON	NCO	
PETICIÓN	17	0	0	0					1,00
A+D	17	0	0	0					1,00
C+P	25	0	27	0					1,00
PROD.	0	0	12	0					1,00
SERVICIO	0	0	9	0					1,00
TOTAL	59	0	48	0	0	0	0	0	1,00

RIM - RIESGOS	
RIESGOS INDICADOS	RIESGOS MATERIALIZADOS
2	0
RIM	0,00%

1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)	
PDP (Previsión Duración Prevista)	0,95
PTP (Precisión Trabajo Previsto)	0,98
CPE (Conformidad)	1,00
PRD (Producción)	0,94
NCP (Conformidad Verificaciones)	1,00
RIM (Riesgos)	0,00

USABILIDAD			
ADECUACIÓN A LA TAREA	SI	NO	N/A
¿Los pasos que se dan en la aplicación son iguales u optimizan los que el usuario realizaba antes de la mecanización?	X		
¿Está bien definida la jerarquía de opciones de los menús de la aplicación?	X		
¿La aplicación permite reducir el tiempo de realización de la tarea?	X		
¿La presentación de información es la oportuna, evitando una carga de trabajo mental innecesaria?	X		
¿Es la interfaz compatible con los documentos fuente?	X		
¿Se utilizan valores por defecto o aquellos más significativos según la tarea?	X		
TOTAL	6	0	0
CARÁCTER AUTODESCRIPTIVO			
¿La aplicación te informa de todas las pantallas por las que has pasado?		X	
¿Tienes posibilidad de acceder a las pantallas visitadas desde cualquier punto?		X	
¿Ofrece la aplicación información o ayuda contextual?		X	
¿Informa la aplicación de dónde estás en cada pantalla y lo que espera?	X		
¿Los elementos del menú están claramente identificados?	X		
TOTAL	2	3	0
CONFORMIDAD CON LAS EXPECTATIVAS DEL USUARIO			
¿Se adecúa la terminología y el vocabulario a los conocimientos técnicos del usuario?	X		
¿Recibe el usuario información útil y efectiva de las acciones que se está llevando a cabo?	X		
Cuando hay una desviación en el tiempo de respuesta esperado por el usuario, ¿se le informa adecuadamente?	X		
¿El comportamiento y apariencia de un diálogo es consistente dentro de las tareas y entre tareas similares?	X		
¿Tienen los mensajes presentados al usuario un estilo constructivo y objetivo?	X		
¿Se sigue una tabulación lógica entre campos de datos secuenciales?	X		
TOTAL	6	0	0
ADECUACIÓN DE APRENDIZAJE			
¿Dispone el usuario del acceso a las reglas y/o tutoriales del programa?	X		
¿La ayuda consigue una comprensión conceptual del sistema por parte del usuario?	X		
¿El diálogo ofrece información tanto de resultados finales como intermedios?	X		
¿Puede el usuario probar y explorar los pasos del diálogo sin consecuencias negativas?	X		
¿Se establecen configuraciones por defecto para facilitar el uso a personal con conocimientos mínimos?	X		
TOTAL	5	0	0
CONTROL POR PARTE DEL USUARIO			
¿Controla el usuario la forma de continuar con el diálogo?	X		
¿Es posible deshacer las operaciones de una tarea cuando su contexto y utilización lo permiten?	X		

Ante la presentación de un gran volumen de datos, ¿controla el usuario la forma en que se presentan?	X		
¿Puede un usuario modificar los valores que vienen establecidos por defecto?	X		
Cuando se han modificado datos, ¿siguen disponibles los datos originales si lo exige la tarea?	X		
TOTAL	5	0	0
TOLERANCIA A ERRORES			
¿Ayuda el sistema al usuario a detectar o prevenir errores en la entrada?	X		
Cuando aparece un error, ¿se le da al usuario información para facilitar su corrección?	X		
Ante una validación, ¿se posiciona el cursor en la posición donde es necesaria la corrección?	X		
¿La verificación y validación de datos se realizan antes de que el sistema procese la entrada?	X		
Si de una acción del usuario se derivan consecuencias graves, ¿informa la aplicación de ello antes de ejecutarla?	X		
TOTAL	5	0	0
PERSONALIZACIÓN DEL DIÁLOGO			
¿La cantidad de información de apoyo se puede modificar de acuerdo con el nivel de conocimientos del usuario?	X		
¿Puede el usuario seleccionar el formato y tipo de los datos, tanto de entrada como de salida?	X		
¿Puede un usuario añadir o reorganizar distintas funcionalidades para adaptarlo a sus necesidades?			X
¿La personalización es reversible permitiendo volver a la configuración original?			X
TOTAL	2	0	2
TOTAL	31	3	US
	91,18%		

SEGURIDAD DE ACCESO			
Modo de Acceso	SI	NO	N/A
¿Se utiliza un sistema de acceso mediante identificador y contraseña?	X		
¿Es necesario ser usuario de Windows para acceder a la aplicación?	X		
¿Es obligatorio el uso de la infraestructura de clave pública PKI para acceder a la aplicación?			X
¿Hay restricciones de acceso por IP/MAC?		X	
¿Se utiliza una tabla de códigos para acceder a la aplicación?	X		
¿Se utiliza identificación por datos biométricos?			X
¿Se registran los intentos de acceso no autorizados?	X		
TOTAL	4	1	2
Restricciones de Seguridad			
¿Se bloquea la aplicación al introducir la contraseña erróneamente un número determinado de veces?	X		

¿Se bloquea automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se elimina automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se genera la primera contraseña suministrada mediante un proceso aleatorio o pseudo-aleatorio?	X		
¿Hay restricciones de acceso por horario?			X
TOTAL	4	0	1
Registros de seguridad			
¿Queda registro de los accesos de cada usuario (fecha/hora)?	X		
¿Queda registro de la duración de la sesión de cada usuario?	X		
¿Queda registro de la manipulación de los datos?	X		
¿Existe un registro de incidencias?	X		
¿Se realizan copias de seguridad de los registros de accesos?	X		
¿Se guardan los registros de acceso en un entorno seguro?	X		
¿Existe la figura de responsable del fichero?	X		
TOTAL	7	0	0
Contraseñas			
¿Es obligatorio modificar la contraseña proporcionada inicialmente?	X		
¿Se obliga a emplear claves de acceso fuertes (longitud mínima y máxima, que contengan números y letras en mayúsculas??	X		
¿Al cambiar la contraseña es obligatorio que la nueva contraseña guarde ciertas diferencias con la antigua (no poder usar secuencias de caracteres utilizados en la antigua)?	X		
¿Hay restricciones en los contenidos de las contraseñas (Ej.: No se puede incluir el nombre del usuario o su DNI en la contraseña)?	X		
¿Si se compromete una contraseña, se compromete el sistema?	X		
¿Las contraseñas han de cambiarse cada cierto tiempo?	X		
TOTAL	6	0	0
Gestión de Usuarios			
¿Existe Una jerarquía de roles y funcionalidades del sistema a la que pertenezcan los usuarios del mismo?	X		
¿Existe un módulo de gestión de usuarios/contraseñas para crearlos, modificarlos o eliminarlos?	X		
¿Existe la figura de un administrador de la seguridad del sistema?	X		
TOTAL	3	0	0
Seguridad de los Datos			
¿Se registran las acciones de modificaciones de datos llevadas a cabo en el sistema?	X		
¿Se cifran los datos almacenados?	X		

¿Se cifran las transmisiones de datos?	X		
¿Se han definido procedimiento para realizar copias de seguridad y verificarlas?	X		
¿Se clasifican los datos según la Ley orgánica de protección de Datos?	X		
TOTAL	5	0	0
Seguridad del sistema			
¿Se ha realizado un análisis de riesgos de seguridad del sistema?	X		
¿En el diseño de las interfaces de usuario se ha contemplado el nivel de conocimiento de los usuarios de forma que se reduzca el riesgo de comprometer el sistema por un mal uso de la aplicación?	X		
¿Existen procedimientos para aplicar regularmente los parches de seguridad a los servidores involucrados en el sistema?	X		
¿Existen políticas de seguridad de red (uso de firewalls, proxys, etc.) para limitar los ataques externos al sistema?	X		
¿Los equipos donde se instala la aplicación cuentan con antivirus actualizado?	X		
¿En los equipos donde se instala la aplicación está restringida la instalación de software externo?	X		
TOTAL	6	0	0

TOTAL	35	1	3
	97,22%		

MANTENIBILIDAD		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)	
TOP (Minutos en Operación)	56.160,00	FUNCIONALIDAD	100,00%	LINEAS DUPLICADAS	4,30%
TMA (Minutos en Mantenimiento)	600,00	USABILIDAD	91,18%	COBERTURA	41,80%
TFA (Minutos en Fallos)	300,00	SEGURIDAD	97,22%	COMENTARIOS	30,70%
TIN (Minutos Indisponible)	900,00	MANTENIBILIDAD	98,93%	DEUDA TÉCNICA	0,10%
		FIABILIDAD	99,47%	VULNERABILIDADES	2
		DISPONIBILIDAD	98,40%		

Tabla A1-1 Métricas Cuantitativas InvenTIC (elaboración propia)

A1-2 IDMANAGE

PDP - PRECISIÓN DE LA DURACIÓN					PTP - PRECISIÓN DEL ESFUERZO				CPE - CONFORMIDAD					PRD - PRODUCCIÓN	
	DUP [días] Duración Prevista	DUR [días] Duración Real	Demora del Usuario	PDP Precisión Duración Prevista	TRP [horas] Trabajo Previsto	TRR [horas] Trabajo Real	Sobreesfuerzo	PTP Prec. Trabajo Previsto		CON	NCO	NCO COND.	CRC PROCESO	Complejidad (PFA)	Esfuerzo (H)
PETICIÓN	15,00	25,00	11,00	0,93	88,00	55,50	0,00	0,63						1.945	2.106
A+D	741,00	830,00	0,00	1,12	2.106,00	1.264,00	0,00	0,60		5	0	0	1,00	0,92	
C+P	835,00	830,00	0,00	0,99	3.606,00	4.591,00	0,00	1,27							
PROD.	102,00	112,00	0,00	1,10	72,00	54,00	0,00	0,75		1	0	0	1,00		
SERVICIO	112,00	117,00	0,00	1,04	7,00	7,00	0,00	1,00							
Proyecto	1.805,00	1.914,00	11,00	1,05	5.879,00	5.971,50	0,00	1,02		6	0	0	1,00		

RFC - CONFORMIDAD DE LA VERIFICACIÓN									
RFC	RFC0		RFC1		RFC2		RFC3		TOTAL
METRICA	CON	NCO	CON	NCO	CON	NCO	CON	NCO	
PETICIÓN	17	1							0,94
A+D	16	1							0,94
C+P	22	1							0,96
PROD.	11	0							1,00
SERVICIO	9	0							1,00
TOTAL	75	3	0	0	0	0	0	0	0,96

RIM - RIESGOS	
RIESGOS INDICADOS	RIESGOS MATERIALIZADOS
3	0
RIM	0,00%

1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)	
PDP (Previsión Duración Prevista)	1,05
PTP (Precisión Trabajo Previsto)	1,02
CPE (Conformidad)	1,00
PRD (Producción)	0,92
NCP (Conformidad Verificaciones)	0,96
RIM (Riesgos)	0,00

USABILIDAD			
ADECUACIÓN A LA TAREA			
¿Los pasos que se dan en la aplicación son iguales u optimizan los que el usuario realizaba antes de la mecanización?	SI	NO	N/A
¿Está bien definida la jerarquía de opciones de los menús de la aplicación?	X		

¿La aplicación permite reducir el tiempo de realización de la tarea?	X		
¿La presentación de información es la oportuna, evitando una carga de trabajo mental innecesaria?	X		
¿Es la interfaz compatible con los documentos fuente?	X		
¿Se utilizan valores por defecto o aquellos más significativos según la tarea?	X		
TOTAL	6	0	0
CARÁCTER AUTODESCRIPTIVO			
¿La aplicación te informa de todas las pantallas por las que has pasado?		X	
¿Tienes posibilidad de acceder a las pantallas visitadas desde cualquier punto?		X	
¿Ofrece la aplicación información o ayuda contextual?		X	
¿Informa la aplicación de dónde estás en cada pantalla y lo que espera?	X		
¿Los elementos del menú están claramente identificados?	X		
TOTAL	2	3	0
CONFORMIDAD CON LAS EXPECTATIVAS DEL USUARIO			
¿Se adecúa la terminología y el vocabulario a los conocimientos técnicos del usuario?	X		
¿Recibe el usuario información útil y efectiva de las acciones que se está llevando a cabo?	X		
Cuando hay una desviación en el tiempo de respuesta esperado por el usuario, ¿se le informa adecuadamente?	X		
¿El comportamiento y apariencia de un diálogo es consistente dentro de las tareas y entre tareas similares?	X		
¿Tienen los mensajes presentados al usuario un estilo constructivo y objetivo?	X		
¿Se sigue una tabulación lógica entre campos de datos secuenciales?	X		
TOTAL	6	0	0
ADECUACIÓN DE APRENDIZAJE			
¿Dispone el usuario del acceso a las reglas y/o tutoriales del programa?	X		
¿La ayuda consigue una comprensión conceptual del sistema por parte del usuario?	X		
¿El diálogo ofrece información tanto de resultados finales como intermedios?	X		
¿Puede el usuario probar y explorar los pasos del diálogo sin consecuencias negativas?	X		
¿Se establecen configuraciones por defecto para facilitar el uso a personal con conocimientos mínimos?	X		
TOTAL	5	0	0
CONTROL POR PARTE DEL USUARIO			
¿Controla el usuario la forma de continuar con el diálogo?	X		
¿Es posible deshacer las operaciones de una tarea cuando su contexto y utilización lo permiten?	X		
Ante la presentación de un gran volumen de datos, ¿controla el usuario la forma en que se presentan?	X		
¿Puede un usuario modificar los valores que vienen establecidos por defecto?	X		
Cuando se han modificado datos, ¿siguen disponibles los datos originales si lo exige la tarea?	X		

TOTAL	5	0	0
TOLERANCIA A ERRORES			
¿Ayuda el sistema al usuario a detectar o prevenir errores en la entrada?	X		
Cuando aparece un error, ¿se le da al usuario información para facilitar su corrección?	X		
Ante una validación, ¿se posiciona el cursor en la posición donde es necesaria la corrección?	X		
¿La verificación y validación de datos se realizan antes de que el sistema procese la entrada?	X		
Si de una acción del usuario se derivan consecuencias graves, ¿informa la aplicación de ello antes de ejecutarla?	X		
TOTAL	5	0	0
PERSONALIZACIÓN DEL DIÁLOGO			
¿La cantidad de información de apoyo se puede modificar de acuerdo con el nivel de conocimientos del usuario?	X		
¿Puede el usuario seleccionar el formato y tipo de los datos, tanto de entrada como de salida?	X		
¿Puede un usuario añadir o reorganizar distintas funcionalidades para adaptarlo a sus necesidades?			X
¿La personalización es reversible permitiendo volver a la configuración original?			X
TOTAL	2	0	2

TOTAL	31	3	US
	91,18%		

SEGURIDAD DE ACCESO			
Modo de Acceso			
¿Se utiliza un sistema de acceso mediante identificador y contraseña?	X		
¿Es necesario ser usuario de Windows para acceder a la aplicación?	X		
¿Es obligatorio el uso de la infraestructura de clave pública PKI para acceder a la aplicación?			X
¿Hay restricciones de acceso por IP/MAC?		X	
¿Se utiliza una tabla de códigos para acceder a la aplicación?	X		
¿Se utiliza identificación por datos biométricos?			X
¿Se registran los intentos de acceso no autorizados?	X		
TOTAL	4	1	2
Restricciones de Seguridad			
¿Se bloquea la aplicación al introducir la contraseña erróneamente un número determinado de veces?	X		
¿Se bloquea automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se elimina automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se genera la primera contraseña suministrada mediante un proceso aleatorio o pseudo-aleatorio?	X		

¿Hay restricciones de acceso por horario?			X
TOTAL	4	0	1
Registros de seguridad			
¿Queda registro de los accesos de cada usuario (fecha/hora)?	X		
¿Queda registro de la duración de la sesión de cada usuario?	X		
¿Queda registro de la manipulación de los datos?	X		
¿Existe un registro de incidencias?	X		
¿Se realizan copias de seguridad de los registros de accesos?	X		
¿Se guardan los registros de acceso en un entorno seguro?	X		
¿Existe la figura de responsable del fichero?	X		
TOTAL	7	0	0
Contraseñas			
¿Es obligatorio modificar la contraseña proporcionada inicialmente?	X		
¿Se obliga a emplear claves de acceso fuertes (longitud mínima y máxima, que contengan números y letras en mayúsculas??	X		
¿Al cambiar la contraseña es obligatorio que la nueva contraseña guarde ciertas diferencias con la antigua (no poder usar secuencias de caracteres utilizados en la antigua)?	X		
¿Hay restricciones en los contenidos de las contraseñas (Ej.: No se puede incluir el nombre del usuario o su DNI en la contraseña)?	X		
¿Si se compromete una contraseña, se compromete el sistema?	X		
¿Las contraseñas han de cambiarse cada cierto tiempo?	X		
TOTAL	6	0	0
Gestión de Usuarios			
¿Existe Una jerarquía de roles y funcionalidades del sistema a la que pertenezcan los usuarios del mismo?	X		
¿Existe un módulo de gestión de usuarios/contraseñas para crearlos, modificarlos o eliminarlos?	X		
¿Existe la figura de un administrador de la seguridad del sistema?	X		
TOTAL	3	0	0
Seguridad de los Datos			
¿Se registran las acciones de modificaciones de datos llevadas a cabo en el sistema?	X		
¿Se cifran los datos almacenados?	X		
¿Se cifran las transmisiones de datos?	X		
¿Se han definido procedimiento para realizar copias de seguridad y verificarlas?	X		
¿Se clasifican los datos según la Ley orgánica de protección de Datos?	X		

TOTAL	5	0	0
Seguridad del sistema			
¿Se ha realizado un análisis de riesgos de seguridad del sistema?	X		
¿En el diseño de las interfaces de usuario se ha contemplado el nivel de conocimiento de los usuarios de forma que se reduzca el riesgo de comprometer el sistema por un mal uso de la aplicación?	X		
¿Existen procedimientos para aplicar regularmente los parches de seguridad a los es involucrados en el sistema?	X		
¿Existen políticas de seguridad de red (uso de firewalls, proxys, etc.) para limitar los ataques externos al sistema?	X		
¿Los equipos donde se instala la aplicación cuentan con antivirus actualizado?	X		
¿En los equipos donde se instala la aplicación está restringida la instalación de software externo?	X		
TOTAL	6	0	0
TOTAL	35	1	3
	97,22%		

MANTENIBILIDAD		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)	
TOP (Minutos en Operación)	168.480,00	FUNCIONALIDAD	100,00%	LINEAS DUPLICADAS	2,51%
TMA (Minutos en Mantenimiento)	720,00	USABILIDAD	91,18%	COBERTURA	46,60%
TFA (Minutos en Fallos)	328,00	SEGURIDAD	97,22%	COMENTARIOS	33,10%
TIN (Minutos Indisponible)	1.048,00	MANTENIBILIDAD	99,57%	DEUDA TÉCNICA	0,80%
		FIABILIDAD	99,81%	VULNERABILIDADES	3
		DISPONIBILIDAD	99,38%		

Tabla A1-2 Métricas Cuantitativas IDManage (elaboración propia)

A1-3 SANIGEST

PDP - PRECISIÓN DE LA DURACIÓN					PTP - PRECISIÓN DEL ESFUERZO				CPE - CONFORMIDAD					PRD - PRODUCCIÓN	
	DUP [días] Duración Prevista	DUR [días] Duración Real	Demora del Usuario	PDP Precisión Duración Prevista	TRP [horas] Trabajo Previsto	TRR [horas] Trabajo Real	Sobreesfuerzo	PTP Prec. Trabajo Previsto		CON	NCO	NCO COND.	CRC PROCESO	Complejidad (PFA)	Esfuerzo (H)
PETICIÓN	45,00	139,00	0,00	3,09	80,50	80,50	0,00	1,00	PETICIÓN	2	0	0	1,00	9.124	9.393,75
A+D	954,00	954,00	0,00	1,00	5.937,00	6.703,50	0,00	1,13	A+D	4	0	0	1,00	0,97	
C+P	779,00	1194,00	0,00	1,53	18.018,20	23.100,60	0,00	1,12	C+P	3	0	0	1,00		
PROD.	150,00	65,00	0,00	0,43	1.652,00	1.811,00	0,00	1,10	PROD.	1	0	0	1,00		
SERVICIO	370,00	69,00	0,00	0,19	360,00	386,00	0,00	1,07	SERVICIO	0	0	0	1,00		
Proyecto	2.298,00	2.421,00	0,00	1,05	26.047,70	32.081,60	0,00	1,12	Proyecto	10	0	0	1,00		

RFC - CONFORMIDAD DE LA VERIFICACIÓN										RIM - RIESGOS		1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)	
RFC	RFC0		RFC1		RFC2		RFC3		TOTAL	RIESGOS INDICADOS	RIESGOS MATERIALIZADOS	PDP (Previsión Duración Prevista)	1,05
METRICA	CON	NCO	CON	NCO	CON	NCO	CON	NCO		2	0	PTP (Precisión Trabajo Previsto)	1,12
PETICIÓN	15	1							0,94	RIM	0,00%	CPE (Conformidad)	1,00
A+D	15	2							0,88			PRD (Producción)	0,97
C+P	23	0							1,00			NCP (Conformidad Verificaciones)	0,96
PROD.	15	0							1,00			RIM (Riesgos)	0,00
SERVICIO	12	0							1,00				
TOTAL	80	2	0	0	0	0	0	0	0,96				

USABILIDAD			
ADECUACIÓN A LA TAREA	SI	NO	N/A
¿Los pasos que se dan en la aplicación son iguales u optimizan los que el usuario realizaba antes de la mecanización?	X		
¿Está bien definida la jerarquía de opciones de los menús de la aplicación?	X		
¿La aplicación permite reducir el tiempo de realización de la tarea?	X		
¿La presentación de información es la oportuna, evitando una carga de trabajo mental innecesaria?	X		
¿Es la interfaz compatible con los documentos fuente?	X		
¿Se utilizan valores por defecto o aquellos más significativos según la tarea?	X		
TOTAL	6	0	0
CARÁCTER AUTODESCRIPTIVO			
¿La aplicación te informa de todas las pantallas por las que has pasado?	X		
¿Tienes posibilidad de acceder a las pantallas visitadas desde cualquier punto?	X		
¿Ofrece la aplicación información o ayuda contextual?	X		
¿Informa la aplicación de donde estas en cada pantalla y lo que espera?	X		
¿Los elementos del menú están claramente identificados?	X		
TOTAL	5	0	0
CONFORMIDAD CON LAS EXPECTATIVAS DEL USUARIO			
¿Se adecúa la terminología y el vocabulario a los conocimientos técnicos del usuario?	X		
¿Recibe el usuario información útil y efectiva de las acciones que se está llevando a cabo?	X		
Cuando hay una desviación en el tiempo de respuesta esperado por el usuario, ¿se le informa adecuadamente?	X		
¿El comportamiento y apariencia de un diálogo es consistente dentro de las tareas y entre tareas similares?	X		
¿Tienen los mensajes presentados al usuario un estilo constructivo y objetivo?	X		
¿Se sigue una tabulación lógica entre campos de datos secuenciales?	X		
TOTAL	6	0	0
ADECUACIÓN DE APRENDIZAJE			
¿Dispone el usuario del acceso a las reglas y/o tutoriales del programa?	X		
¿La ayuda consigue una comprensión conceptual del sistema por parte del usuario?	X		
¿El diálogo ofrece información tanto de resultados finales como intermedios?	X		
¿Puede el usuario probar y explorar los pasos del diálogo sin consecuencias negativas?	X		
¿Se establecen configuraciones por defecto para facilitar el uso a personal con conocimientos mínimos?	X		
TOTAL	5	0	0
CONTROL POR PARTE DEL USUARIO			
¿Controla el usuario la forma de continuar con el diálogo?	X		

¿Es posible deshacer las operaciones de una tarea cuando su contexto y utilización lo permiten?	X		
Ante la presentación de un gran volumen de datos, ¿controla el usuario la forma en que se presentan?	X		
¿Puede un usuario modificar los valores que vienen establecidos por defecto?	X		
Cuando se han modificado datos, ¿siguen disponibles los datos originales si lo exige la tarea?	X		
TOTAL	5	0	0
TOLERANCIA A ERRORES			
¿Ayuda el sistema al usuario a detectar o prevenir errores en la entrada?	X		
Cuando aparece un error, ¿se le da al usuario información para facilitar su corrección?	X		
Ante una validación, ¿se posiciona el cursor en la posición donde es necesaria la corrección?		X	
¿La verificación y validación de datos se realizan antes de que el sistema procese la entrada?	X		
Si de una acción del usuario se derivan consecuencias graves, ¿informa la aplicación de ello antes de ejecutarla?	X		
TOTAL	4	1	0
PERSONALIZACIÓN DEL DIÁLOGO			
¿La cantidad de información de apoyo se puede modificar de acuerdo con el nivel de conocimientos del usuario?	X		
¿Puede el usuario seleccionar el formato y tipo de los datos, tanto de entrada como de salida?		X	
¿Puede un usuario añadir o reorganizar distintas funcionalidades para adaptarlo a sus necesidades?	X		
¿La personalización es reversible permitiendo volver a la configuración original?	X		
TOTAL	3	1	0

TOTAL	34	2	US
	94,44%		

SEGURIDAD DE ACCESO			
Modo de Acceso	SI	NO	N/A
¿Se utiliza un sistema de acceso mediante identificador y contraseña?	X		
¿Es necesario ser usuario de Windows para acceder a la aplicación?	X		
¿Es obligatorio el uso de la infraestructura de clave pública PKI para acceder a la aplicación?			X
¿Hay restricciones de acceso por IP/MAC?		X	
¿Se utiliza una tabla de códigos para acceder a la aplicación?	X		
¿Se utiliza identificación por datos biométricos?			X
¿Se registran los intentos de acceso no autorizados?	X		
TOTAL	4	1	2
Restricciones de Seguridad			

¿Se bloquea la aplicación al introducir la contraseña erróneamente un número determinado de veces?	X		
¿Se bloquea automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se elimina automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se genera la primera contraseña suministrada mediante un proceso aleatorio o pseudo-aleatorio?		X	
¿Hay restricciones de acceso por horario?		X	
TOTAL	3	2	0
Registros de seguridad			
¿Queda registro de los accesos de cada usuario (fecha/hora)?	X		
¿Queda registro de la duración de la sesión de cada usuario?	X		
¿Queda registro de la manipulación de los datos?	X		
¿Existe un registro de incidencias?	X		
¿Se realizan copias de seguridad de los registros de accesos?	X		
¿Se guardan los registros de acceso en un entorno seguro?	X		
¿Existe la figura de responsable del fichero?			X
TOTAL	6	0	1
Contraseñas			
¿Es obligatorio modificar la contraseña proporcionada inicialmente?	X		
¿Se obliga a emplear claves de acceso fuertes (longitud mínima y máxima, que contengan números y letras en mayúsculas??	X		
¿Al cambiar la contraseña es obligatorio que la nueva contraseña guarde ciertas diferencias con la antigua (no poder usar secuencias de caracteres utilizados en la antigua)?	X		
¿Hay restricciones en los contenidos de las contraseñas (Ej.: No se puede incluir el nombre del usuario o su DNI en la contraseña)?	X		
¿Si se compromete una contraseña, se compromete el sistema?	X		
¿Las contraseñas han de cambiarse cada cierto tiempo?	X		
TOTAL	6	0	0
Gestión de Usuarios			
¿Existe Una jerarquía de roles y funcionalidades del sistema a la que pertenezcan los usuarios del mismo?	X		
¿Existe un módulo de gestión de usuarios/contraseñas para crearlos, modificarlos o eliminarlos?	X		
¿Existe la figura de un administrador de la seguridad del sistema?	X		
TOTAL	3	0	0
Seguridad de los Datos			
¿Se registran las acciones de modificaciones de datos llevadas a cabo en el sistema?	X		

¿Se cifran los datos almacenados?	X		
¿Se cifran las transmisiones de datos?	X		
¿Se han definido procedimiento para realizar copias de seguridad y verificarlas?	X		
¿Se clasifican los datos según la Ley orgánica de protección de Datos?	X		
TOTAL	5	0	0
Seguridad del sistema			
¿Se ha realizado un análisis de riesgos de seguridad del sistema?	X		
¿En el diseño de las interfaces de usuario se ha contemplado el nivel de conocimiento de los usuarios de forma que se reduzca el riesgo de comprometer el sistema por un mal uso de la aplicación?	X		
¿Existen procedimientos para aplicar regularmente los parches de seguridad a los servidores involucrados en el sistema?	X		
¿Existen políticas de seguridad de red (uso de firewalls, proxys, etc.) para limitar los ataques externos al sistema?	X		
¿Los equipos donde se instala la aplicación cuentan con antivirus actualizado?	X		
¿En los equipos donde se instala la aplicación está restringida la instalación de software externo?	X		
TOTAL	6	0	0
TOTAL	33	3	3
	91,67%		

MANTENIBILIDAD		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)	
TOP (Minutos en Operación)	99.360,00	FUNCIONALIDAD	100,00%	LINEAS DUPLICADAS	7,21%
TMA (Minutos en Mantenimiento)	2.945, 00	USABILIDAD	94,44%	COBERTURA	44,40%
TFA (Minutos en Fallos)	145,00	SEGURIDAD	91,67%	COMENTARIOS	36,21%
TIN (Minutos Indisponible)	3.090,00	MANTENIBILIDAD	97,04%	DEUDA TÉCNICA	0,80%
		FIABILIDAD	99,85%	VULNERABILIDADES	2
		DISPONIBILIDAD	96,89%		

Tabla A1-3 Métricas Cuantitativas SaniGest (elaboración propia)

A1-4 CARGONET

PDP - PRECISIÓN DE LA DURACIÓN					PTP - PRECISIÓN DEL ESFUERZO				CPE - CONFORMIDAD					PRD - PRODUCCIÓN	
	DUP [días] Duración Prevista	DUR [días] Duración Real	Demora del Usuario	PDP Precisión Duración	TRP [horas] Trabajo Previsto	TRR [horas] Trabajo Real	Sobreesfuerzo	PTP Prec. Trabajo Previsto		CON	NCO	NCO COND.	CRC PROCESO	Complejidad (PFA)	Esfuerzo (H)
PETICIÓN	35,00	30,00	29,00	0,03	39,00	33,00	0,00	1,00	PETICIÓN	0	0	0	1,00	624	619.17
A+D	115,00	129,00	38,00	0,79	465,00	478,00	0,00	1,03	A+D	1	0	0	1,00	1,01	
C+P	133,00	159,00	0,00	1,20	2.354,00	3.205,00	0,00	1,08	C+P	2	0	0	1,00		
PROD.	40,00	60,00	0,00	1,50	331,00	362,00	0,00	1,09	PROD.	0	0	0	1,00		
SERVICIO	30,00	62,00	0,00	2,07	2.200,00	2.167,00	0,00	0,99	SERVICIO	0	0	0	1,00		
Proyecto	353,00	440,00	67,00	1,06	5.989,00	6.251,00	0,00	1,04	Proyecto	3	0	0	1,00		

RFC - CONFORMIDAD DE LA VERIFICACIÓN									
RFC	RFC0		RFC1		RFC2		RFC3		TOTAL
METRICA	CON	NCO	CON	NCO	CON	NCO	CON	NCO	
PETICIÓN	16	0							1,00
A+D	16	0							1,00
C+P	23	0							1,00
PROD.	11	0							1,00
SERVICIO	12	0							1,00
TOTAL	78	0	0	0	0	0	0	0	1,00

RIM - RIESGOS	
RIESGOS INDICADOS	RIESGOS MATERIALIZADOS
1	0
RIM	0,00%

1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)	
PDP (Previsión Duración Prevista)	1,06
PTP (Precisión Trabajo Previsto)	1,04
CPE (Conformidad)	1,00
PRD (Producción)	1,01
NCP (Conformidad Verificaciones)	1,00
RIM (Riesgos)	0,00

USABILIDAD			
ADECUACIÓN A LA TAREA	SI	NO	N/A
¿Los pasos que se dan en la aplicación son iguales u optimizan los que el usuario realizaba antes de la mecanización?	X		
¿Está bien definida la jerarquía de opciones de los menús de la aplicación?	X		
¿La aplicación permite reducir el tiempo de realización de la tarea?	X		
¿La presentación de información es la oportuna, evitando una carga de trabajo mental innecesaria?	X		
¿Es la interfaz compatible con los documentos fuente?	X		
¿Se utilizan valores por defecto o aquellos más significativos según la tarea?	X		
TOTAL	6	0	0
CARÁCTER AUTODESCRIPTIVO			
¿La aplicación te informa de todas las pantallas por las que has pasado?	X		
¿Tienes posibilidad de acceder a las pantallas visitadas desde cualquier punto?	X		
¿Ofrece la aplicación información o ayuda contextual?	X		
¿Informa la aplicación de donde estas en cada pantalla y lo que espera?	X		
¿Los elementos del menú están claramente identificados?	X		
TOTAL	5	0	0
CONFORMIDAD CON LAS EXPECTATIVAS DEL USUARIO			
¿Se adecúa la terminología y el vocabulario a los conocimientos técnicos del usuario?	X		
¿Recibe el usuario información útil y efectiva de las acciones que se está llevando a cabo?	X		
Cuando hay una desviación en el tiempo de respuesta esperado por el usuario, ¿se le informa adecuadamente?	X		
¿El comportamiento y apariencia de un diálogo es consistente dentro de las tareas y entre tareas similares?	X		
¿Tienen los mensajes presentados al usuario un estilo constructivo y objetivo?	X		
¿Se sigue una tabulación lógica entre campos de datos secuenciales?	X		
TOTAL	6	0	0
ADECUACIÓN DE APRENDIZAJE			
¿Dispone el usuario del acceso a las reglas y/o tutoriales del programa?	X		
¿La ayuda consigue una comprensión conceptual del sistema por parte del usuario?	X		
¿El diálogo ofrece información tanto de resultados finales como intermedios?	X		
¿Puede el usuario probar y explorar los pasos del diálogo sin consecuencias negativas?	X		
¿Se establecen configuraciones por defecto para facilitar el uso a personal con conocimientos mínimos?	X		
TOTAL	5	0	0
CONTROL POR PARTE DEL USUARIO			
¿Controla el usuario la forma de continuar con el diálogo?	X		

¿Es posible deshacer las operaciones de una tarea cuando su contexto y utilización lo permiten?	X		
Ante la presentación de un gran volumen de datos, ¿controla el usuario la forma en que se presentan?	X		
¿Puede un usuario modificar los valores que vienen establecidos por defecto?	X		
Cuando se han modificado datos, ¿siguen disponibles los datos originales si lo exige la tarea?	X		
TOTAL	5	0	0
TOLERANCIA A ERRORES			
¿Ayuda el sistema al usuario a detectar o prevenir errores en la entrada?	X		
Cuando aparece un error, ¿se le da al usuario información para facilitar su corrección?	X		
Ante una validación, ¿se posiciona el cursor en la posición donde es necesaria la corrección?	X		
¿La verificación y validación de datos se realizan antes de que el sistema procese la entrada?	X		
Si de una acción del usuario se derivan consecuencias graves, ¿informa la aplicación de ello antes de ejecutarla?	X		
TOTAL	5	0	0
PERSONALIZACIÓN DEL DIÁLOGO			
¿La cantidad de información de apoyo se puede modificar de acuerdo con el nivel de conocimientos del usuario?	X		
¿Puede el usuario seleccionar el formato y tipo de los datos, tanto de entrada como de salida?		X	
¿Puede un usuario añadir o reorganizar distintas funcionalidades para adaptarlo a sus necesidades?			X
¿La personalización es reversible permitiendo volver a la configuración original?	X		
TOTAL	2	1	1
TOTAL	34	1	US
97,14%			

SEGURIDAD DE ACCESO			
Modo de Acceso	SI	NO	N/A
¿Se utiliza un sistema de acceso mediante identificador y contraseña?	X		
¿Es necesario ser usuario de Windows para acceder a la aplicación?	X		
¿Es obligatorio el uso del uso de la infraestructura de clave pública PKI para acceder a la aplicación?			X
¿Hay restricciones de acceso por IP/MAC?			X
¿Se utiliza una tabla de códigos para acceder a la aplicación?			X
¿Se utiliza identificación por datos biométricos?			X
¿Se registran los intentos de acceso no autorizados?			X
TOTAL	2	0	5
Restricciones de Seguridad			

¿Se bloquea la aplicación al introducir la contraseña erróneamente un número determinado de veces?			X
¿Se bloquea automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?			X
¿Se elimina automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?			X
¿Se genera la primera contraseña suministrada mediante un proceso aleatorio o pseudo-aleatorio?			X
¿Hay restricciones de acceso por horario?			X
TOTAL	0	0	5
Registros de seguridad			
¿Queda registro de los accesos de cada usuario (fecha/hora)?			X
¿Queda registro de la duración de la sesión de cada usuario?			X
¿Queda registro de la manipulación de los datos?	X		
¿Existe un registro de incidencias?	X		
¿Se realizan copias de seguridad de los registros de accesos?			X
¿Se guardan los registros de acceso en un entorno seguro?			X
¿Existe la figura de responsable del fichero?	X		
TOTAL	3	0	4
Contraseñas			
¿Es obligatorio modificar la contraseña proporcionada inicialmente?			X
¿Se obliga a emplear claves de acceso fuertes (longitud mínima y máxima, que contengan números y letras en mayúsculas)?	X		
¿Al cambiar la contraseña es obligatorio que la nueva contraseña guarde ciertas diferencias con la antigua (no poder usar secuencias de caracteres utilizados en la antigua)?	X		
¿Hay restricciones en los contenidos de las contraseñas (Ej.: No se puede incluir el nombre del usuario o su DNI en la contraseña)?			X
¿Si se compromete una contraseña, se compromete el sistema?		X	
¿Las contraseñas han de cambiarse cada cierto tiempo?	X		
TOTAL	3	1	2
Gestión de Usuarios			
¿Existe Una jerarquía de roles y funcionalidades del sistema a la que pertenezcan los usuarios del mismo?	X		
¿Existe un módulo de gestión de usuarios/contraseñas para crearlos, modificarlos o eliminarlos?			X
¿Existe la figura de un administrador de la seguridad del sistema?	X		
TOTAL	2	0	1
Seguridad de los Datos			
¿Se registran las acciones de modificaciones de datos llevadas a cabo en el sistema?	X		

¿Se cifran los datos almacenados?			X
¿Se cifran las transmisiones de datos?	X		
¿Se han definido procedimiento para realizar copias de seguridad y verificarlas?	X		
¿Se clasifican los datos según la Ley orgánica de protección de Datos?	X		
TOTAL	4	0	1
Seguridad del sistema			
¿Se ha realizado un análisis de riesgos de seguridad del sistema?	X		
¿En el diseño de las interfaces de usuario se ha contemplado el nivel de conocimiento de los usuarios de forma que se reduzca el riesgo de comprometer el sistema por un mal uso de la aplicación?	X		
¿Existen procedimientos para aplicar regularmente los parches de seguridad a los servidores involucrados en el sistema?	X		
¿Existen políticas de seguridad de red (uso de firewalls, proxys, etc.) para limitar los ataques externos al sistema?	X		
¿Los equipos donde se instala la aplicación cuentan con antivirus actualizado?	X		
¿En los equipos donde se instala la aplicación está restringida la instalación de software externo?	X		
TOTAL	6	0	0

TOTAL	20	1	18
	95,24%		

MANTENIBILIDAD		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)	
TOP (Minutos en Operación)	88.260,00	FUNCIONALIDAD	100,00%	LINEAS DUPLICADAS	3,50%
TMA (Minutos en Mantenimiento)	1.257,00	USABILIDAD	97,14%	COBERTURA	63,70%
TFA (Minutos en Fallos)	397,00	SEGURIDAD	95,24%	COMENTARIOS	30,10%
TIN (Minutos Indisponible)	1.654,00	MANTENIBILIDAD	98,58%	DEUDA TÉCNICA	0,20%
		FIABILIDAD	99,55%	VULNERABILIDADES	1
		DISPONIBILIDAD	98,13%		

Tabla A1-4 Métricas Cuantitativas CargoNet (elaboración propia)

A1-5 PLANEÁ

PDP - PRECISIÓN DE LA DURACIÓN					PTP - PRECISIÓN DEL ESFUERZO				CPE - CONFORMIDAD					PRD - PRODUCCIÓN	
	DUP [días] Duración Prevista	DUR [días] Duración Real	Demora del Usuario	PDP Precisión Duración Prevista	TRP [horas] Trabajo Previsto	TRR [horas] Trabajo Real	Sobreesfuerzo	PTP Prec. Trabajo Previsto		CON	NCO	NCO COND.	CRC PROCESO	Complejidad (PFA)	Esfuerzo (H)
PETICIÓN	35,00	23,00	0,00	0,66	45,50	26,50	0,00	0,58	PETICIÓN	5	0	0	1,00	503	509.62
A+D	235,00	249,00	0,00	1,06	359,50	362,50	0,00	1,01	A+D	0	0	0		0,99	
C+P	660,00	681,00	0,00	1,03	1.416,00	1.241,00	0,00	0,88	C+P	0	0	0			
PROD.	32,00	37,00	0,00	1,16	35,00	40,00	0,00	1,14	PROD.	0	0	0			
SERVICIO	30,00	30,00	0,00	1,00	17,00	17,00	0,00	1,00	SERVICIO	0	0	0			
Proyecto	992,00	1.020,00	0,00	1,03	1.873,00	1.687,00	0,00	0,90	Proyecto	5	0	0	1,00		

RFC - CONFORMIDAD DE LA VERIFICACIÓN										RIM - RIESGOS		1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)	
RFC	RFC0		RFC1		RFC2		RFC3		TOTAL	RIESGOS INDICADOS	RIESGOS MATERIALIZADOS	PDP (Previsión Duración Prevista)	1,03
METRICA	CON	NCO	CON	NCO	CON	NCO	CON	NCO		2	0	PTP (Precisión Trabajo Previsto)	0,90
PETICIÓN	16	0							1,00	RIM	0,00%	CPE (Conformidad)	1,00
A+D	14	0							1,00			PRD (Producción)	0,99
C+P	20	1							1,00			NCP (Conformidad Verificaciones)	0,99
PROD.	12	0							1,00			RIM (Riesgos)	2
SERVICIO	10	0							1,00				
TOTAL	72	1							0,99				

USABILIDAD			
ADECUACIÓN A LA TAREA	SI	NO	N/A
¿Los pasos que se dan en la aplicación son iguales u optimizan los que el usuario realizaba antes de la mecanización?	X		
¿Está bien definida la jerarquía de opciones de los menús de la aplicación?	X		
¿La aplicación permite reducir el tiempo de realización de la tarea?	X		
¿La presentación de información es la oportuna, evitando una carga de trabajo mental innecesaria?	X		
¿Es la interfaz compatible con los documentos fuente?	X		
¿Se utilizan valores por defecto o aquellos más significativos según la tarea?	X		
TOTAL	6	0	0
CARÁCTER AUTODESCRIPTIVO			
¿La aplicación te informa de todas las pantallas por las que has pasado?	X		
¿Tienes posibilidad de acceder a las pantallas visitadas desde cualquier punto?	X		
¿Ofrece la aplicación información o ayuda contextual?	X		
¿Informa la aplicación de donde estas en cada pantalla y lo que espera?	X		
¿Los elementos del menú están claramente identificados?	X		
TOTAL	5	0	0
CONFORMIDAD CON LAS EXPECTATIVAS DEL USUARIO			
¿Se adecúa la terminología y el vocabulario a los conocimientos técnicos del usuario?	X		
¿Recibe el usuario información útil y efectiva de las acciones que se está llevando a cabo?	X		
Cuando hay una desviación en el tiempo de respuesta esperado por el usuario, ¿se le informa adecuadamente?	X		
¿El comportamiento y apariencia de un diálogo es consistente dentro de las tareas y entre tareas similares?	X		
¿Tienen los mensajes presentados al usuario un estilo constructivo y objetivo?	X		
¿Se sigue una tabulación lógica entre campos de datos secuenciales?	X		
TOTAL	6	0	0
ADECUACIÓN DE APRENDIZAJE			
¿Dispone el usuario del acceso a las reglas y/o tutoriales del programa?	X		
¿La ayuda consigue una comprensión conceptual del sistema por parte del usuario?	X		
¿El diálogo ofrece información tanto de resultados finales como intermedios?	X		
¿Puede el usuario probar y explorar los pasos del diálogo sin consecuencias negativas?	X		
¿Se establecen configuraciones por defecto para facilitar el uso a personal con conocimientos mínimos?	X		
TOTAL	5	0	0
CONTROL POR PARTE DEL USUARIO			
¿Controla el usuario la forma de continuar con el diálogo?	X		

¿Es posible deshacer las operaciones de una tarea cuando su contexto y utilización lo permiten?	X		
Ante la presentación de un gran volumen de datos, ¿controla el usuario la forma en que se presentan?	X		
¿Puede un usuario modificar los valores que vienen establecidos por defecto?	X		
Cuando se han modificado datos, ¿siguen disponibles los datos originales si lo exige la tarea?	X		
TOTAL	5	0	0
TOLERANCIA A ERRORES			
¿Ayuda el sistema al usuario a detectar o prevenir errores en la entrada?	X		
Cuando aparece un error, ¿se le da al usuario información para facilitar su corrección?	X		
Ante una validación, ¿se posiciona el cursor en la posición donde es necesaria la corrección?	X		
¿La verificación y validación de datos se realizan antes de que el sistema procese la entrada?	X		
Si de una acción del usuario se derivan consecuencias graves, ¿informa la aplicación de ello antes de ejecutarla?	X		
TOTAL	5	0	0
PERSONALIZACIÓN DEL DIÁLOGO			
¿La cantidad de información de apoyo se puede modificar de acuerdo con el nivel de conocimientos del usuario?	X		
¿Puede el usuario seleccionar el formato y tipo de los datos, tanto de entrada como de salida?		X	
¿Puede un usuario añadir o reorganizar distintas funcionalidades para adaptarlo a sus necesidades?			X
¿La personalización es reversible permitiendo volver a la configuración original?	X		
TOTAL	2	1	1

TOTAL	34	1	US
	97,14%		

SEGURIDAD DE ACCESO			
Modo de Acceso	SI	NO	N/A
¿Se utiliza un sistema de acceso mediante identificador y contraseña?	X		
¿Es necesario ser usuario de Windows para acceder a la aplicación?	X		
¿Es obligatorio el uso del uso de la infraestructura de clave pública PKI para acceder a la aplicación?			X
¿Hay restricciones de acceso por IP/MAC?			X
¿Se utiliza una tabla de códigos para acceder a la aplicación?			X
¿Se utiliza identificación por datos biométricos?			X
¿Se registran los intentos de acceso no autorizados?			X
TOTAL	2	0	5
Restricciones de Seguridad			

¿Se bloquea la aplicación al introducir la contraseña erróneamente un número determinado de veces?			X
¿Se bloquea automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?			X
¿Se elimina automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?			X
¿Se genera la primera contraseña suministrada mediante un proceso aleatorio o pseudo-aleatorio?			X
¿Hay restricciones de acceso por horario?			X
TOTAL	0	0	5
Registros de seguridad			
¿Queda registro de los accesos de cada usuario (fecha/hora)?			X
¿Queda registro de la duración de la sesión de cada usuario?			X
¿Queda registro de la manipulación de los datos?	X		
¿Existe un registro de incidencias?	X		
¿Se realizan copias de seguridad de los registros de accesos?			X
¿Se guardan los registros de acceso en un entorno seguro?			X
¿Existe la figura de responsable del fichero?	X		
TOTAL	3	0	4
Contraseñas			
¿Es obligatorio modificar la contraseña proporcionada inicialmente?			X
¿Se obliga a emplear claves de acceso fuertes (longitud mínima y máxima, que contengan números y letras en mayúsculas??	X		
¿Al cambiar la contraseña es obligatorio que la nueva contraseña guarde ciertas diferencias con la antigua (no poder usar secuencias de caracteres utilizados en la antigua)?	X		
¿Hay restricciones en los contenidos de las contraseñas (Ej.: No se puede incluir el nombre del usuario o su DNI en la contraseña)?			X
¿Si se compromete una contraseña, se compromete el sistema?		X	
¿Las contraseñas han de cambiarse cada cierto tiempo?	X		
TOTAL	3	1	2
Gestión de Usuarios			
¿Existe Una jerarquía de roles y funcionalidades del sistema a la que pertenezcan los usuarios del mismo?	X		
¿Existe un módulo de gestión de usuarios/contraseñas para crearlos, modificarlos o eliminarlos?			X
¿Existe la figura de un administrador de la seguridad del sistema?	X		
TOTAL	2	0	1
Seguridad de los Datos			
¿Se registran las acciones de modificaciones de datos llevadas a cabo en el sistema?	X		

¿Se cifran los datos almacenados?			X
¿Se cifran las transmisiones de datos?	X		
¿Se han definido procedimiento para realizar copias de seguridad y verificarlas?	X		
¿Se clasifican los datos según la Ley orgánica de protección de Datos?	X		
TOTAL	4	0	1
Seguridad del sistema			
¿Se ha realizado un análisis de riesgos de seguridad del sistema?	X		
¿En el diseño de las interfaces de usuario se ha contemplado el nivel de conocimiento de los usuarios de forma que se reduzca el riesgo de comprometer el sistema por un mal uso de la aplicación?	X		
¿Existen procedimientos para aplicar regularmente los parches de seguridad a los servidores involucrados en el sistema?	X		
¿Existen políticas de seguridad de red (uso de firewalls, proxys, etc.) para limitar los ataques externos al sistema?	X		
¿Los equipos donde se instala la aplicación cuentan con antivirus actualizado?	X		
¿En los equipos donde se instala la aplicación está restringida la instalación de software externo?	X		
TOTAL	6	0	0
TOTAL	20	1	18
	95,24%		

MANTENIBILIDAD		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)	
TOP (Minutos en Operación)	43.620,00	FUNCIONALIDAD	100%	LINEAS DUPLICADAS	21,80%
TMA (Minutos en Mantenimiento)	489,00	USABILIDAD	97,14%	COBERTURA	11,31%
TFA (Minutos en Fallos)	401,00	SEGURIDAD	95,24%	COMENTARIOS	23,06%
TIN (Minutos Indisponible)	890,00	MANTENIBILIDAD	98,88%	DEUDA TÉCNICA	0,30%
		FIABILIDAD	99,08%	VULNERABILIDADES	2
		DISPONIBILIDAD	97,96%		

Tabla A1-5 Métricas Cuantitativas PlanEA (elaboración propia)

A1-6 EcoGEST

PDP - PRECISIÓN DE LA DURACIÓN					PTP - PRECISIÓN DEL ESFUERZO				CPE - CONFORMIDAD					PRD - PRODUCCIÓN	
	DUP [días] Duración Prevista	DUR [días] Duración Real	Demora del Usuario	PDP Precisión Duración Prevista	TRP [horas] Trabajo Previsto	TRR [horas] Trabajo Real	Sobreesfuerzo	PTP Prec. Trabajo Previsto		CON	NCO	NCO COND.	CRC PROCESO	Complejidad (PFA)	Esfuerzo (H)
PETICIÓN	1.017,00	985,00	0,00	0,97	60,50	60,00	0,00	0,99	PETICIÓN	0	0	0		7.427	7.750
A+D	279,00	283,00	0,00	1,01	727,00	718,60	0,00	0,99	A+D	1	0	0	1,00	0,96	
C+P	241,00	350,00	0,00	1,45	3.537,20	3.599,80	0,00	1,02	C+P	1	0	0	1,00		
PROD.	220,00	426,00	0,00	1,94	131,50	129,00	0,00	0,98	PROD.	0	0	0			
SERVICIO	35,00	32,00	0,00	0,91	537,50	754,21	0,00	1,40	SERVICIO	0	0	0			
Proyecto	1.792,00	2.076,00	0,00	1,16	1.456,50	1.661,81	0,00	1,14	Proyecto	2	0	0	1,00		

RFC - CONFORMIDAD DE LA VERIFICACIÓN									
RFC	RFC0		RFC1		RFC2		RFC3		TOTAL
METRICA	CON	NCO	CON	NCO	CON	NCO	CON	NCO	
PETICIÓN	13	0							1,00
A+D	17	0							1,00
C+P	25	1							0,96
PROD.	0	0							
SERVICIO	0	0							
TOTAL	55	1	0	0	0	0	0	0	0,98

RIM - RIESGOS	
RIESGOS INDICADOS	RIESGOS MATERIALIZADOS
3	0
RIM	0,00%

1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)	
PDP (Previsión Duración Prevista)	1,16
PTP (Precisión Trabajo Previsto)	1,14
CPE (Conformidad)	1,00
PRD (Producción)	0,96
NCP (Conformidad Verificaciones)	0,98
RIM (Riesgos)	0,00

USABILIDAD

ADECUACIÓN A LA TAREA	SI	NO	N/A
¿Los pasos que se dan en la aplicación son iguales u optimizan los que el usuario realizaba antes de la mecanización?	X		
¿Está bien definida la jerarquía de opciones de los menús de la aplicación?	X		
¿La aplicación permite reducir el tiempo de realización de la tarea?	X		
¿La presentación de información es la oportuna, evitando una carga de trabajo mental innecesaria?	X		
¿Es la interfaz compatible con los documentos fuente?	X		
¿Se utilizan valores por defecto o aquellos más significativos según la tarea?	X		
TOTAL	6	0	0
CARÁCTER AUTODESCRIPTIVO			
¿La aplicación te informa de todas las pantallas por las que has pasado?	X		
¿Tienes posibilidad de acceder a las pantallas visitadas desde cualquier punto?	X		
¿Ofrece la aplicación información o ayuda contextual?	X		
¿Informa la aplicación de dónde estás en cada pantalla y lo que espera?	X		
¿Los elementos del menú están claramente identificados?	X		
TOTAL	5	0	0
CONFORMIDAD CON LAS EXPECTATIVAS DEL USUARIO			
¿Se adecúa la terminología y el vocabulario a los conocimientos técnicos del usuario?	X		
¿Recibe el usuario información útil y efectiva de las acciones que se está llevando a cabo?	X		
Cuando hay una desviación en el tiempo de respuesta esperado por el usuario, ¿se le informa adecuadamente?		X	
¿El comportamiento y apariencia de un diálogo es consistente dentro de las tareas y entre tareas similares?	X		
¿Tienen los mensajes presentados al usuario un estilo constructivo y objetivo?	X		
¿Se sigue una tabulación lógica entre campos de datos secuenciales?	X		
TOTAL	5	1	0
ADECUACIÓN DE APRENDIZAJE			
¿Dispone el usuario del acceso a las reglas y/o tutoriales del programa?	X		
¿La ayuda consigue una comprensión conceptual del sistema por parte del usuario?	X		
¿El diálogo ofrece información tanto de resultados finales como intermedios?	X		
¿Puede el usuario probar y explorar los pasos del diálogo sin consecuencias negativas?	X		
¿Se establecen configuraciones por defecto para facilitar el uso a personal con conocimientos mínimos?	X		
TOTAL	5	0	0
CONTROL POR PARTE DEL USUARIO			
¿Controla el usuario la forma de continuar con el diálogo?	X		
¿Es posible deshacer las operaciones de una tarea cuando su contexto y utilización lo permiten?	X		

Ante la presentación de un gran volumen de datos, ¿controla el usuario la forma en que se presentan?	X		
¿Puede un usuario modificar los valores que vienen establecidos por defecto?	X		
Cuando se han modificado datos, ¿siguen disponibles los datos originales si lo exige la tarea?	X		
TOTAL	5	0	0
TOLERANCIA A ERRORES			
¿Ayuda el sistema al usuario a detectar o prevenir errores en la entrada?	X		
Cuando aparece un error, ¿se le da al usuario información para facilitar su corrección?	X		
Ante una validación, ¿se posiciona el cursor en la posición donde es necesaria la corrección?	X		
¿La verificación y validación de datos se realizan antes de que el sistema procese la entrada?	X		
Si de una acción del usuario se derivan consecuencias graves, ¿informa la aplicación de ello antes de ejecutarla?	X		
TOTAL	5	0	0
PERSONALIZACIÓN DEL DIÁLOGO			
¿La cantidad de información de apoyo se puede modificar de acuerdo con el nivel de conocimientos del usuario?	X		
¿Puede el usuario seleccionar el formato y tipo de los datos, tanto de entrada como de salida?	X		
¿Puede un usuario añadir o reorganizar distintas funcionalidades para adaptarlo a sus necesidades?			X
¿La personalización es reversible permitiendo volver a la configuración original?			X
TOTAL	2	0	2
TOTAL	33	1	US
		97,06%	

SEGURIDAD DE ACCESO			
Modo de Acceso	SI	NO	N/A
¿Se utiliza un sistema de acceso mediante identificador y contraseña?	X		
¿Es necesario ser usuario de Windows para acceder a la aplicación?	X		
¿Es obligatorio el uso de la infraestructura de clave pública PKI para acceder a la aplicación?		X	
¿Hay restricciones de acceso por IP/MAC?			X
¿Se utiliza una tabla de códigos para acceder a la aplicación?	X		
¿Se utiliza identificación por datos biométricos?			X
¿Se registran los intentos de acceso no autorizados?	X		
TOTAL	4	1	2
Restricciones de Seguridad			
¿Se bloquea la aplicación al introducir la contraseña erróneamente un número determinado de veces?	X		

¿Se bloquea automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se elimina automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se genera la primera contraseña suministrada mediante un proceso aleatorio o pseudo-aleatorio?		X	
¿Hay restricciones de acceso por horario?			X
TOTAL	3	1	1
Registros de seguridad			
¿Queda registro de los accesos de cada usuario (fecha/hora)?	X		
¿Queda registro de la duración de la sesión de cada usuario?	X		
¿Queda registro de la manipulación de los datos?	X		
¿Existe un registro de incidencias?	X		
¿Se realizan copias de seguridad de los registros de accesos?	X		
¿Se guardan los registros de acceso en un entorno seguro?	X		
¿Existe la figura de responsable del fichero?			X
TOTAL	6	0	1
Contraseñas			
¿Es obligatorio modificar la contraseña proporcionada inicialmente?	X		
¿Se obliga a emplear claves de acceso fuertes (longitud mínima y máxima, que contengan números y letras en mayúsculas??	X		
¿Al cambiar la contraseña es obligatorio que la nueva contraseña guarde ciertas diferencias con la antigua (no poder usar secuencias de caracteres utilizados en la antigua)?	X		
¿Hay restricciones en los contenidos de las contraseñas (Ej.: No se puede incluir el nombre del usuario o su DNI en la contraseña)?	X		
¿Si se compromete una contraseña, se compromete el sistema?	X		
¿Las contraseñas han de cambiarse cada cierto tiempo?	X		
TOTAL	6	0	0
Gestión de Usuarios			
¿Existe Una jerarquía de roles y funcionalidades del sistema a la que pertenezcan los usuarios del mismo?	X		
¿Existe un módulo de gestión de usuarios/contraseñas para crearlos, modificarlos o eliminarlos?	X		
¿Existe la figura de un administrador de la seguridad del sistema?	X		
TOTAL	3	0	0
Seguridad de los Datos			
¿Se registran las acciones de modificaciones de datos llevadas a cabo en el sistema?	X		
¿Se cifran los datos almacenados?	X		

¿Se cifran las transmisiones de datos?	X		
¿Se han definido procedimiento para realizar copias de seguridad y verificarlas?	X		
¿Se clasifican los datos según la Ley orgánica de protección de Datos?	X		
TOTAL	5	0	0
Seguridad del sistema			
¿Se ha realizado un análisis de riesgos de seguridad del sistema?	X		
¿En el diseño de las interfaces de usuario se ha contemplado el nivel de conocimiento de los usuarios de forma que se reduzca el riesgo de comprometer el sistema por un mal uso de la aplicación?	X		
¿Existen procedimientos para aplicar regularmente los parches de seguridad a los servidores involucrados en el sistema?	X		
¿Existen políticas de seguridad de red (uso de firewalls, proxys, etc.) para limitar los ataques externos al sistema?	X		
¿Los equipos donde se instala la aplicación cuentan con antivirus actualizado?	X		
¿En los equipos donde se instala la aplicación está restringida la instalación de software externo?	X		
TOTAL	6	0	0
TOTAL	33	2	4
	94,29%		

MANTENIBILIDAD		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)	
TOP (Minutos en Operación)	45.252,60	FUNCIONALIDAD	100%	LINEAS DUPLICADAS	11,20%
TMA (Minutos en Mantenimiento)	2.629,00	USABILIDAD	97,06%	COBERTURA	40,00%
TFA (Minutos en Fallos)	1.241,00	SEGURIDAD	94,29%	COMENTARIOS	23,00%
TIN (Minutos Indisponible)	3.870,00	MANTENIBILIDAD	91,45%	DEUDA TÉCNICA	0,00%
		FIABILIDAD	97,26%	VULNERABILIDADES	3
		DISPONIBILIDAD	93,66%		

Tabla A1-6 Métricas Cuantitativas EcoGest (elaboración propia)

A1-7 MONITORA

PDP - PRECISIÓN DE LA DURACIÓN					PTP - PRECISIÓN DEL ESFUERZO				CPE - CONFORMIDAD					PRD - PRODUCCIÓN	
	DUP [días] Duración Prevista	DUR [días] Duración Real	Demora del Usuario	PDP Precisión Duración Prevista	TRP [horas] Trabajo Previsto	TRR [horas] Trabajo Real	Sobreesfuerzo	PTP Prec. Trabajo Previsto		CON	NCO	NCO COND.	CRC PROCESO	Complejidad (PFA)	Esfuerzo (H)
PETICIÓN	90,00	92,00	0,00	1,02	9,00	9,00	0,00	1,00	PETICIÓN	2	0	0	1,00	5.968	6.388
A+D	309,00	426,00	0,00	1,38	1.434,10	1.433,40	0,00	1,00	A+D	0	0	0		0,93	
C+P	396,00	403,00	0,00	1,02	6.505,00	6.851,60	0,00	1,02	C+P	1	0	0	1,00		
PROD.	169,00	241,00	0,00	1,43	249,00	307,00	0,00	1,23	PROD.	0	0	0			
SERVICIO	151,00	161,00	0,00	1,07	4,50	5,88	0,00	1,31	SERVICIO	0	0	0			
Proyecto	1.115,00	1.323,00	0,00	1,19	8.202,10	8.606,88	0,00	1,05	Proyecto	3	0	0	1,00		

RFC - CONFORMIDAD DE LA VERIFICACIÓN										RIM - RIESGOS		1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)	
RFC	RFC0		RFC1		RFC2		RFC3		TOTAL	RIESGOS INDICADOS	RIESGOS MATERIALIZADOS	PDP (Previsión Duración Prevista)	1,19
METRICA	CON	NCO	CON	NCO	CON	NCO	CON	NCO		1	0	PTP (Precisión Trabajo Previsto)	1,05
PETICIÓN	15	0							1,00	RIM	0,00%	CPE (Conformidad)	1,00
A+D	0	0							1,00			PRD (Producción)	0,93
C+P	17	0							1,00			NCP (Conformidad Verificaciones)	1,00
PROD.	21	0							1,00			RIM (Riesgos)	0,00
SERVICIO	14	0							1,00				
TOTAL	67	0	0	0	0	0	0	0	1,00				

USABILIDAD			
ADECUACIÓN A LA TAREA	SI	NO	N/A
¿Los pasos que se dan en la aplicación son iguales u optimizan los que el usuario realizaba antes de la mecanización?	X		
¿Está bien definida la jerarquía de opciones de los menús de la aplicación?	X		
¿La aplicación permite reducir el tiempo de realización de la tarea?	X		
¿La presentación de información es la oportuna, evitando una carga de trabajo mental innecesaria?	X		
¿Es la interfaz compatible con los documentos fuente?	X		
¿Se utilizan valores por defecto o aquellos más significativos según la tarea?	X		
TOTAL	6	0	0
CARÁCTER AUTODESCRIPTIVO			
¿La aplicación te informa de todas las pantallas por las que has pasado?			X
¿Tienes posibilidad de acceder a las pantallas visitadas desde cualquier punto?	X		
¿Ofrece la aplicación información o ayuda contextual?	X		
¿Informa la aplicación de donde estas en cada pantalla y lo que espera?	X		
¿Los elementos del menú están claramente identificados?	X		
TOTAL	4	0	1
CONFORMIDAD CON LAS EXPECTATIVAS DEL USUARIO			
¿Se adecúa la terminología y el vocabulario a los conocimientos técnicos del usuario?	X		
¿Recibe el usuario información útil y efectiva de las acciones que se está llevando a cabo?	X		
Cuando hay una desviación en el tiempo de respuesta esperado por el usuario, ¿se le informa adecuadamente?	X		
¿El comportamiento y apariencia de un diálogo es consistente dentro de las tareas y entre tareas similares?	X		
¿Tienen los mensajes presentados al usuario un estilo constructivo y objetivo?			X
¿Se sigue una tabulación lógica entre campos de datos secuenciales?	X		
TOTAL	5	0	1
ADECUACIÓN DE APRENDIZAJE			
¿Dispone el usuario del acceso a las reglas y/o tutoriales del programa?	x		
¿La ayuda consigue una comprensión conceptual del sistema por parte del usuario?	X		
¿El diálogo ofrece información tanto de resultados finales como intermedios?			X
¿Puede el usuario probar y explorar los pasos del diálogo sin consecuencias negativas?			X
¿Se establecen configuraciones por defecto para facilitar el uso a personal con conocimientos mínimos?	X		
TOTAL	3	0	2
CONTROL POR PARTE DEL USUARIO			
¿Controla el usuario la forma de continuar con el diálogo?	X		

¿Es posible deshacer las operaciones de una tarea cuando su contexto y utilización lo permiten?	X		
Ante la presentación de un gran volumen de datos, ¿controla el usuario la forma en que se presentan?	X		
¿Puede un usuario modificar los valores que vienen establecidos por defecto?		X	
Cuando se han modificado datos, ¿siguen disponibles los datos originales si lo exige la tarea?	X		
TOTAL	4	1	0
TOLERANCIA A ERRORES			
¿Ayuda el sistema al usuario a detectar o prevenir errores en la entrada?	X		
Cuando aparece un error, ¿se le da al usuario información para facilitar su corrección?			X
Ante una validación, ¿se posiciona el cursor en la posición donde es necesaria la corrección?	X		
¿La verificación y validación de datos se realizan antes de que el sistema procese la entrada?	X		
Si de una acción del usuario se derivan consecuencias graves, ¿informa la aplicación de ello antes de ejecutarla?	X		
TOTAL	4	0	1
PERSONALIZACIÓN DEL DIÁLOGO			
¿La cantidad de información de apoyo se puede modificar de acuerdo con el nivel de conocimientos del usuario?	X		
¿Puede el usuario seleccionar el formato y tipo de los datos, tanto de entrada como de salida?	X		
¿Puede un usuario añadir o reorganizar distintas funcionalidades para adaptarlo a sus necesidades?	X		
¿La personalización es reversible permitiendo volver a la configuración original?	X		
TOTAL	4	0	0

TOTAL	30	1	US
	96,77%		

SEGURIDAD DE ACCESO			
Modo de Acceso	SI	NO	N/A
¿Se utiliza un sistema de acceso mediante identificador y contraseña?	X		
¿Es necesario ser usuario de Windows para acceder a la aplicación?	X		
¿Es obligatorio el uso del uso de la infraestructura de clave pública PKI para acceder a la aplicación?			X
¿Hay restricciones de acceso por IP/MAC?			X
¿Se utiliza una tabla de códigos para acceder a la aplicación?		X	
¿Se utiliza identificación por datos biométricos?		X	
¿Se registran los intentos de acceso no autorizados?	X		
TOTAL	3	2	2
Restricciones de Seguridad			

¿Se bloquea la aplicación al introducir la contraseña erróneamente un número determinado de veces?	X		
¿Se bloquea automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se elimina automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se genera la primera contraseña suministrada mediante un proceso aleatorio o pseudo-aleatorio?	X		
¿Hay restricciones de acceso por horario?			X
TOTAL	4	0	1
Registros de seguridad			
¿Queda registro de los accesos de cada usuario (fecha/hora)?	X		
¿Queda registro de la duración de la sesión de cada usuario?	X		
¿Queda registro de la manipulación de los datos?	X		
¿Existe un registro de incidencias?	X		
¿Se realizan copias de seguridad de los registros de accesos?	X		
¿Se guardan los registros de acceso en un entorno seguro?	X		
¿Existe la figura de responsable del fichero?	X		
TOTAL	7	0	0
Contraseñas			
¿Es obligatorio modificar la contraseña proporcionada inicialmente?	x		
¿Se obliga a emplear claves de acceso fuertes (longitud mínima y máxima, que contengan números y letras en mayúsculas??	x		
¿Al cambiar la contraseña es obligatorio que la nueva contraseña guarde ciertas diferencias con la antigua (no poder usar secuencias de caracteres utilizados en la antigua)?	X		
¿Hay restricciones en los contenidos de las contraseñas (Ej: No se puede incluir el nombre del usuario o su DNI en la contraseña)?	X		
¿Si se compromete una contraseña, se compromete el sistema?	X		
¿Las contraseñas han de cambiarse cada cierto tiempo?	X		
TOTAL	6	0	0
Gestión de Usuarios			
¿Existe Una jerarquía de roles y funcionalidades del sistema a la que pertenezcan los usuarios del mismo?	X		
¿Existe un módulo de gestión de usuarios/contraseñas para crearlos, modificarlos o eliminarlos?	X		
¿Existe la figura de un administrador de la seguridad del sistema?	X		
TOTAL	3	0	0
Seguridad de los Datos			
¿Se registran las acciones de modificaciones de datos llevadas a cabo en el sistema?	X		
¿Se cifran los datos almacenados?	X		

¿Se cifran las transmisiones de datos?	X		
¿Se han definido procedimiento para realizar copias de seguridad y verificarlas?	X		
¿Se clasifican los datos según la Ley orgánica de protección de Datos?	X		
TOTAL	5	0	0
Seguridad del sistema			
¿Se ha realizado un análisis de riesgos de seguridad del sistema?	X		
¿En el diseño de las interfaces de usuario se ha contemplado el nivel de conocimiento de los usuarios de forma que se reduzca el riesgo de comprometer el sistema por un mal uso de la aplicación?	X		
¿Existen procedimientos para aplicar regularmente los parches de seguridad a los servidores involucrados en el sistema?	X		
¿Existen políticas de seguridad de red (uso de firewalls, proxys, etc.) para limitar los ataques externos al sistema?	X		
¿Los equipos donde se instala la aplicación cuentan con antivirus actualizado?	X		
¿En los equipos donde se instala la aplicación está restringida la instalación de software externo?	X		
TOTAL	6	0	0
TOTAL	34	2	3
	94,44%		

MANTENIBILIDAD		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)	
TOP (Minutos en Operación)	231.400,00	FUNCIONALIDAD	100%	LINEAS DUPLICADAS	8,20%
TMA (Minutos en Mantenimiento)	7.854,00	USABILIDAD	96,77%	COBERTURA	46,70%
TFA (Minutos en Fallos)	2.674,00	SEGURIDAD	94,44%	COMENTARIOS	30,50%
TIN (Minutos Indisponible)	10.528,00	MANTENIBILIDAD	96,61%	DEUDA TÉCNICA	0,00%
		FIABILIDAD	98,85%	VULNERABILIDADES	1
		DISPONIBILIDAD	95,46%		

Tabla A1-7 Métricas Cuantitativas Monitora (elaboración propia)

A1-8 UNITPLAN

PDP - PRECISIÓN DE LA DURACIÓN					PTP - PRECISIÓN DEL ESFUERZO				CPE - CONFORMIDAD					PRD - PRODUCCIÓN	
	DUP [días] Duración Prevista	DUR [días] Duración Real	Demora del Usuario	PDP Precisión Duración Prevista	TRP [horas] Trabajo Previsto	TRR [horas] Trabajo Real	Sobreesfuerzo	PTP Prec. Trabajo Previsto		CON	NCO	NCO COND.	CRC PROCESO	Complejidad (PFA)	Esfuerzo (H)
PETICIÓN	20,00	25,00	0,00	1,25	99,00	125,10	0,00	1,26	PETICIÓN	0	0	0		6.002	5.625
A+D	411,00	498,00	0,00	1,21	658,00	1.197,10	0,00	1,82	A+D	0	0	0		1,07	
C+P	488,00	579,00	0,00	1,19	3.564,50	3.954,80	0,00	1,11	C+P	1	0	0	1,00		
PROD.	124,00	140,00	0,00	1,13	130,00	110,00	0,00	0,85	PROD.	0	0	0			
SERVICIO	187,00	135,00	0,00	0,72	80,00	80,00	0,00	1,00	SERVICIO	0	0	0			
Proyecto	1.230,00	1.377,00	0,00	1,12	4.531,50	5.467,00	0,00	1,21	Proyecto	1	0	0	1,00		

RFC - CONFORMIDAD DE LA VERIFICACIÓN									
RFC	RFC0		RFC1		RFC2		RFC3		TOTAL
METRICA	CON	NCO	CON	NCO	CON	NCO	CON	NCO	
PETICIÓN	16	0							1,00
A+D	17	0							1,00
C+P	20	0							1,00
PROD.	0	0							
SERVICIO	0	0							
TOTAL	53	0	0	0	0	0	0	0	1,00

RIM - RIESGOS	
RIESGOS INDICADOS	RIESGOS MATERIALIZADOS
2	0
RIM	0,00%

1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)	
PDP (Previsión Duración Prevista)	1,12
PTP (Precisión Trabajo Previsto)	1,21
CPE (Conformidad)	1,00
PRD (Producción)	1,07
NCP (Conformidad Verificaciones)	1,00
RIM (Riesgos)	0,00

USABILIDAD

ADECUACIÓN A LA TAREA	SI	NO	N/A
¿Los pasos que se dan en la aplicación son iguales u optimizan los que el usuario realizaba antes de la mecanización?	X		
¿Está bien definida la jerarquía de opciones de los menús de la aplicación?	X		
¿La aplicación permite reducir el tiempo de realización de la tarea?			X
¿La presentación de información es la oportuna, evitando una carga de trabajo mental innecesaria?	X		
¿Es la interfaz compatible con los documentos fuente?	X		
¿Se utilizan valores por defecto o aquellos más significativos según la tarea?	X		
TOTAL	5	0	1
CARÁCTER AUTODESCRIPTIVO			
¿La aplicación te informa de todas las pantallas por las que has pasado?			X
¿Tienes posibilidad de acceder a las pantallas visitadas desde cualquier punto?		X	
¿Ofrece la aplicación información o ayuda contextual?	X		
¿Informa la aplicación de dónde estás en cada pantalla y lo que espera?			X
¿Los elementos del menú están claramente identificados?	X		
TOTAL	2	1	2
CONFORMIDAD CON LAS EXPECTATIVAS DEL USUARIO			
¿Se adecúa la terminología y el vocabulario a los conocimientos técnicos del usuario?	X		
¿Recibe el usuario información útil y efectiva de las acciones que se está llevando a cabo?	X		
Cuando hay una desviación en el tiempo de respuesta esperado por el usuario, ¿se le informa adecuadamente?			X
¿El comportamiento y apariencia de un diálogo es consistente dentro de las tareas y entre tareas similares?	X		
¿Tienen los mensajes presentados al usuario un estilo constructivo y objetivo?	X		
¿Se sigue una tabulación lógica entre campos de datos secuenciales?	X		
TOTAL	5	0	1
ADECUACIÓN DE APRENDIZAJE			
¿Dispone el usuario del acceso a las reglas y/o tutoriales del programa?	X		
¿La ayuda consigue una comprensión conceptual del sistema por parte del usuario?	X		
¿El diálogo ofrece información tanto de resultados finales como intermedios?	X		
¿Puede el usuario probar y explorar los pasos del diálogo sin consecuencias negativas?			X
¿Se establecen configuraciones por defecto para facilitar el uso a personal con conocimientos mínimos?	X		
TOTAL	4	0	1
CONTROL POR PARTE DEL USUARIO			
¿Controla el usuario la forma de continuar con el diálogo?	X		
¿Es posible deshacer las operaciones de una tarea cuando su contexto y utilización lo permiten?	X		

Ante la presentación de un gran volumen de datos, ¿controla el usuario la forma en que se presentan?			X
¿Puede un usuario modificar los valores que vienen establecidos por defecto?	X		
Cuando se han modificado datos, ¿siguen disponibles los datos originales si lo exige la tarea?	X		
TOTAL	4	0	1
TOLERANCIA A ERRORES			
¿Ayuda el sistema al usuario a detectar o prevenir errores en la entrada?			X
Cuando aparece un error, ¿se le da al usuario información para facilitar su corrección?	X		
Ante una validación, ¿se posiciona el cursor en la posición donde es necesaria la corrección?	X		
¿La verificación y validación de datos se realizan antes de que el sistema procese la entrada?	X		
Si de una acción del usuario se derivan consecuencias graves, ¿informa la aplicación de ello antes de ejecutarla?	X		
TOTAL	4	0	1
PERSONALIZACIÓN DEL DIÁLOGO			
¿La cantidad de información de apoyo se puede modificar de acuerdo con el nivel de conocimientos del usuario?	X		
¿Puede el usuario seleccionar el formato y tipo de los datos, tanto de entrada como de salida?	X		
¿Puede un usuario añadir o reorganizar distintas funcionalidades para adaptarlo a sus necesidades?			X
¿La personalización es reversible permitiendo volver a la configuración original?			X
TOTAL	2	0	2
TOTAL	26	1	US
	96,30%		

SEGURIDAD DE ACCESO			
Modo de Acceso	SI	NO	N/A
¿Se utiliza un sistema de acceso mediante identificador y contraseña?	X		
¿Es necesario ser usuario de Windows para acceder a la aplicación?	X		
¿Es obligatorio el uso de la infraestructura de clave pública PKI para acceder a la aplicación?		X	
¿Hay restricciones de acceso por IP/MAC?			X
¿Se utiliza una tabla de códigos para acceder a la aplicación?		X	
¿Se utiliza identificación por datos biométricos?			X
¿Se registran los intentos de acceso no autorizados?	X		
TOTAL	3	2	2
Restricciones de Seguridad			
¿Se bloquea la aplicación al introducir la contraseña erróneamente un número determinado de veces?	X		

¿Se bloquea automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se elimina automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se genera la primera contraseña suministrada mediante un proceso aleatorio o pseudo-aleatorio?	X		
¿Hay restricciones de acceso por horario?	X		
TOTAL	5	0	0
Registros de seguridad			
¿Queda registro de los accesos de cada usuario (fecha/hora)?	X		
¿Queda registro de la duración de la sesión de cada usuario?	X		
¿Queda registro de la manipulación de los datos?	X		
¿Existe un registro de incidencias?	X		
¿Se realizan copias de seguridad de los registros de accesos?			X
¿Se guardan los registros de acceso en un entorno seguro?	X		
¿Existe la figura de responsable del fichero?	X		
TOTAL	6	0	1
Contraseñas			
¿Es obligatorio modificar la contraseña proporcionada inicialmente?			X
¿Se obliga a emplear claves de acceso fuertes (longitud mínima y máxima, que contengan números y letras en mayúsculas??	X		
¿Al cambiar la contraseña es obligatorio que la nueva contraseña guarde ciertas diferencias con la antigua (no poder usar secuencias de caracteres utilizados en la antigua)?	X		
¿Hay restricciones en los contenidos de las contraseñas (Ej.: No se puede incluir el nombre del usuario o su DNI en la contraseña)?	X		
¿Si se compromete una contraseña, se compromete el sistema?	X		
¿Las contraseñas han de cambiarse cada cierto tiempo?	X		
TOTAL	5	0	1
Gestión de Usuarios			
¿Existe una jerarquía de roles y funcionalidades del sistema a la que pertenezcan los usuarios del mismo?	X		
¿Existe un módulo de gestión de usuarios/contraseñas para crearlos, modificarlos o eliminarlos?	X		
¿Existe la figura de un administrador de la seguridad del sistema?	X		
TOTAL	3	0	0
Seguridad de los Datos			
¿Se registran las acciones de modificaciones de datos llevadas a cabo en el sistema?	X		
¿Se cifran los datos almacenados?			X

¿Se cifran las transmisiones de datos?			X
¿Se han definido procedimiento para realizar copias de seguridad y verificarlas?	X		
¿Se clasifican los datos según la Ley orgánica de protección de Datos?	X		
TOTAL	3	0	2
Seguridad del sistema			
¿Se ha realizado un análisis de riesgos de seguridad del sistema?	X		
¿En el diseño de las interfaces de usuario se ha contemplado el nivel de conocimiento de los usuarios de forma que se reduzca el riesgo de comprometer el sistema por un mal uso de la aplicación?			X
¿Existen procedimientos para aplicar regularmente los parches de seguridad a los servidores involucrados en el sistema?	X		
¿Existen políticas de seguridad de red (uso de firewalls, proxys, etc.) para limitar los ataques externos al sistema?	X		
¿Los equipos donde se instala la aplicación cuentan con antivirus actualizado?	X		
¿En los equipos donde se instala la aplicación está restringida la instalación de software externo?	X		
TOTAL	5	0	1
TOTAL	30	2	7
	93,75%		

MANTENIBILIDAD		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)	
TOP (Minutos en Operación)	194.400,00	FUNCIONALIDAD	100%	LINEAS DUPLICADAS	8,63%
TMA (Minutos en Mantenimiento)	6.245,00	USABILIDAD	96,30%	COBERTURA	43,08%
TFA (Minutos en Fallos)	3.654,00	SEGURIDAD	93,75%	COMENTARIOS	30,50%
TIN (Minutos Indisponible)	9.899,00	MANTENIBILIDAD	96,79%	DEUDA TÉCNICA	0,00%
		FIABILIDAD	98,12%	VULNERABILIDADES	2
		DISPONIBILIDAD	94,91%		

Tabla A1-8 Métricas Cuantitativas UnitPlan (elaboración propia)

A1-9 SENSORPRO

PDP - PRECISIÓN DE LA DURACIÓN					PTP - PRECISIÓN DEL ESFUERZO				CPE - CONFORMIDAD					PRD - PRODUCCIÓN	
	DUP [días] Duración Prevista	DUR [días] Duración Real	Demora del Usuario	PDP Precisión Duración	TRP [horas] Trabajo Previsto	TRR [horas] Trabajo Real	Sobreesfuerzo	PTP Prec. Trabajo Previsto		CON	NCO	NCO COND.	CRC PROCESO	Complejidad (PFA)	Esfuerzo (H)
PETICIÓN	55,00	52,00	0,00	0,95	64,00	67,00	0,00	1,05		0	0	0		3.412	3.357,5
A+D	368,00	368,00	0,00	1,10	1.006,50	954,50	0,00	0,95		1	0	0	1,00	1,01	
C+P	669,00	669,00	0,00	1,00	6.506,00	7.008,00	0,00	1,08		1	0	0	1,00		
PROD.	20,00	14,00	0,00	0,70	116,50	123,50	0,00	1,06		0	0	0			
SERVICIO	66,00	95,00	0,00	1,44	448,20	448,20	0,00	1,00		0	0	0			
Proyecto	1.145,00	1.198,00	0,00	1,05	8.141,20	8.601,20	0,00	1,06			0	0	1,00		

RFC - CONFORMIDAD DE LA VERIFICACIÓN									
RFC	RFC0		RFC1		RFC2		RFC3		TOTAL
METRICA	CON	NCO	CON	NCO	CON	NCO	CON	NCO	
PETICIÓN	14	0							1,00
A+D	16	0							1,00
C+P	25	0							1,00
PROD.	12	0							1,00
SERVICIO	9	0							1,00
TOTAL	76	0	0	0	0	0	0	0	1,00

RIM - RIESGOS	
RIESGOS INDICADOS	RIESGOS MATERIALIZADOS
1	0
RIM	0,00%

1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)	
PDP (Previsión Duración Prevista)	1,05
PTP (Precisión Trabajo Previsto)	1,06
CPE (Conformidad)	1,00
PRD (Producción)	1,01
NCP (Conformidad Verificaciones)	1,00
RIM (Riesgos)	0,00

USABILIDAD			
ADECUACIÓN A LA TAREA	SI	NO	N/A
¿Los pasos que se dan en la aplicación son iguales u optimizan los que el usuario realizaba antes de la mecanización?	X		
¿Está bien definida la jerarquía de opciones de los menús de la aplicación?	X		
¿La aplicación permite reducir el tiempo de realización de la tarea?	X		
¿La presentación de información es la oportuna, evitando una carga de trabajo mental innecesaria?	X		
¿Es la interfaz compatible con los documentos fuente?	X		
¿Se utilizan valores por defecto o aquellos más significativos según la tarea?	X		
TOTAL	6	0	0
CARÁCTER AUTODESCRIPTIVO			
¿La aplicación te informa de todas las pantallas por las que has pasado?	X		
¿Tienes posibilidad de acceder a las pantallas visitadas desde cualquier punto?	X		
¿Ofrece la aplicación información o ayuda contextual?	X		
¿Informa la aplicación de donde estas en cada pantalla y lo que espera?	X		
¿Los elementos del menú están claramente identificados?	X		
TOTAL	5	0	0
CONFORMIDAD CON LAS EXPECTATIVAS DEL USUARIO			
¿Se adecúa la terminología y el vocabulario a los conocimientos técnicos del usuario?	X		
¿Recibe el usuario información útil y efectiva de las acciones que se está llevando a cabo?	X		
Cuando hay una desviación en el tiempo de respuesta esperado por el usuario, ¿se le informa adecuadamente?	X		
¿El comportamiento y apariencia de un diálogo es consistente dentro de las tareas y entre tareas similares?	X		
¿Tienen los mensajes presentados al usuario un estilo constructivo y objetivo?	X		
¿Se sigue una tabulación lógica entre campos de datos secuenciales?	X		
TOTAL	6	0	0
ADECUACIÓN DE APRENDIZAJE			
¿Dispone el usuario del acceso a las reglas y/o tutoriales del programa?	X		
¿La ayuda consigue una comprensión conceptual del sistema por parte del usuario?	X		
¿El diálogo ofrece información tanto de resultados finales como intermedios?	X		
¿Puede el usuario probar y explorar los pasos del diálogo sin consecuencias negativas?	X		
¿Se establecen configuraciones por defecto para facilitar el uso a personal con conocimientos mínimos?	X		
TOTAL	5	0	0
CONTROL POR PARTE DEL USUARIO			
¿Controla el usuario la forma de continuar con el diálogo?	X		

¿Es posible deshacer las operaciones de una tarea cuando su contexto y utilización lo permiten?	X		
Ante la presentación de un gran volumen de datos, ¿controla el usuario la forma en que se presentan?	X		
¿Puede un usuario modificar los valores que vienen establecidos por defecto?	X		
Cuando se han modificado datos, ¿siguen disponibles los datos originales si lo exige la tarea?	X		
TOTAL	5	0	0
TOLERANCIA A ERRORES			
¿Ayuda el sistema al usuario a detectar o prevenir errores en la entrada?	X		
Cuando aparece un error, ¿se le da al usuario información para facilitar su corrección?	X		
Ante una validación, ¿se posiciona el cursor en la posición donde es necesaria la corrección?	X		
¿La verificación y validación de datos se realizan antes de que el sistema procese la entrada?	X		
Si de una acción del usuario se derivan consecuencias graves, ¿informa la aplicación de ello antes de ejecutarla?	X		
TOTAL	5	0	0
PERSONALIZACIÓN DEL DIÁLOGO			
¿La cantidad de información de apoyo se puede modificar de acuerdo con el nivel de conocimientos del usuario?	X		
¿Puede el usuario seleccionar el formato y tipo de los datos, tanto de entrada como de salida?		X	
¿Puede un usuario añadir o reorganizar distintas funcionalidades para adaptarlo a sus necesidades?			X
¿La personalización es reversible permitiendo volver a la configuración original?	X		
TOTAL	2	1	1

TOTAL	34	1	US
	97,14%		

SEGURIDAD DE ACCESO			
Modo de Acceso	SI	NO	N/A
¿Se utiliza un sistema de acceso mediante identificador y contraseña?	X		
¿Es necesario ser usuario de Windows para acceder a la aplicación?	X		
¿Es obligatorio el uso de la infraestructura de clave pública PKI para acceder a la aplicación?			X
¿Hay restricciones de acceso por IP/MAC?			X
¿Se utiliza una tabla de códigos para acceder a la aplicación?			X
¿Se utiliza identificación por datos biométricos?			X
¿Se registran los intentos de acceso no autorizados?			X
TOTAL	2	0	5
Restricciones de Seguridad			

¿Se bloquea la aplicación al introducir la contraseña erróneamente un número determinado de veces?			X
¿Se bloquea automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?			X
¿Se elimina automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?			X
¿Se genera la primera contraseña suministrada mediante un proceso aleatorio o pseudo-aleatorio?			X
¿Hay restricciones de acceso por horario?			X
TOTAL	0	0	5
Registros de seguridad			
¿Queda registro de los accesos de cada usuario (fecha/hora)?			X
¿Queda registro de la duración de la sesión de cada usuario?			X
¿Queda registro de la manipulación de los datos?	X		
¿Existe un registro de incidencias?	X		
¿Se realizan copias de seguridad de los registros de accesos?			X
¿Se guardan los registros de acceso en un entorno seguro?			X
¿Existe la figura de responsable del fichero?	X		
TOTAL	3	0	4
Contraseñas			
¿Es obligatorio modificar la contraseña proporcionada inicialmente?			X
¿Se obliga a emplear claves de acceso fuertes (longitud mínima y máxima, que contengan números y letras en mayúsculas??	X		
¿Al cambiar la contraseña es obligatorio que la nueva contraseña guarde ciertas diferencias con la antigua (no poder usar secuencias de caracteres utilizados en la antigua)?	X		
¿Hay restricciones en los contenidos de las contraseñas (Ej.: No se puede incluir el nombre del usuario o su DNI en la contraseña)?			X
¿Si se compromete una contraseña, se compromete el sistema?		X	
¿Las contraseñas han de cambiarse cada cierto tiempo?	X		
TOTAL	3	1	2
Gestión de Usuarios			
¿Existe Una jerarquía de roles y funcionalidades del sistema a la que pertenezcan los usuarios del mismo?	X		
¿Existe un módulo de gestión de usuarios/contraseñas para crearlos, modificarlos o eliminarlos?			X
¿Existe la figura de un administrador de la seguridad del sistema?	X		
TOTAL	2	0	1
Seguridad de los Datos			
¿Se registran las acciones de modificaciones de datos llevadas a cabo en el sistema?	X		

¿Se cifran los datos almacenados?			X
¿Se cifran las transmisiones de datos?	X		
¿Se han definido procedimiento para realizar copias de seguridad y verificarlas?	X		
¿Se clasifican los datos según la Ley orgánica de protección de Datos?	X		
TOTAL	4	0	1
Seguridad del sistema			
¿Se ha realizado un análisis de riesgos de seguridad del sistema?	X		
¿En el diseño de las interfaces de usuario se ha contemplado el nivel de conocimiento de los usuarios de forma que se reduzca el riesgo de comprometer el sistema por un mal uso de la aplicación?	X		
¿Existen procedimientos para aplicar regularmente los parches de seguridad a los servidores involucrados en el sistema?	X		
¿Existen políticas de seguridad de red (uso de firewalls, proxys, etc.) para limitar los ataques externos al sistema?	X		
¿Los equipos donde se instala la aplicación cuentan con antivirus actualizado?	X		
¿En los equipos donde se instala la aplicación está restringida la instalación de software externo?	X		
TOTAL	6	0	0
TOTAL	20	1	18
	95,24%		

MANTENIBILIDAD		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)	
TOP (Minutos en Operación)	137.220,00	FUNCIONALIDAD	100%	LINEAS DUPLICADAS	2,40%
TMA (Minutos en Mantenimiento)	2.154,00	USABILIDAD	97,14%	COBERTURA	57,20%
TFA (Minutos en Fallos)	1.328,00	SEGURIDAD	95,24%	COMENTARIOS	30,20%
TIN (Minutos Indisponible)	3.482,00	MANTENIBILIDAD	98,43%	DEUDA TÉCNICA	0,00%
		FIABILIDAD	99,03%	VULNERABILIDADES	1
		DISPONIBILIDAD	94,46%		

Tabla A1-9 Métricas Cuantitativas SensorPro (elaboración propia)

A1-10 GesLOG

PDP - PRECISIÓN DE LA DURACIÓN					PTP - PRECISIÓN DEL ESFUERZO				CPE - CONFORMIDAD					PRD - PRODUCCIÓN	
	DUP [días] Duración Prevista	DUR [días] Duración Real	Demora del Usuario	PDP Precisión Duración Prevista	TRP [horas] Trabajo Previsto	TRR [horas] Trabajo Real	Sobreesfuerzo	PTP Prec. Trabajo Previsto		CON	NCO	NCO COND.	CRC PROCESO	Complejidad (PFA)	Esfuerzo (H)
PETICIÓN	72,00	189,00	0,00	2,63	157,25	168,25	0,00	1,07	PETICIÓN	0	0	0		20.149	19.358,12
A+D	1.234,00	1.324,00	0,00	1,07	1.701,00	1.448,00	0,00	0,85	A+D	3	0	0	1,00	1,04	
C+P	1.305,00	1.329,00	0,00	1,02	8.533,00	9.591,00	0,00	1,12	C+P	3	0	0	1,00		
PROD.	17,00	52,00	0,00	3,06	12,70	8,60	0,00	0,68	PROD.	0	0	0			
SERVICIO	16,00	217,00	0,00	13,56	259,00	378,00	0,00	1,46	SERVICIO	0	0	0			
Proyecto	2.644,00	3.111,00	0,00	1,18	10.662,95	11.593,85	0,00	1,09	Proyecto	6	0	0	1,00		

RFC - CONFORMIDAD DE LA VERIFICACIÓN									
RFC	RFC0		RFC1		RFC2		RFC3		TOTAL
METRICA	CON	NCO	CON	NCO	CON	NCO	CON	NCO	
PETICIÓN	17	0	0	0					1,00
A+D	0	0	17	2					0,89
C+P	0	0	21	0					1,00
PROD.	0	0	14	0					1,00
SERVICIO	0	0	10	0					1,00
TOTAL	17	0	62	2					0,98

RIM - RIESGOS	
RIESGOS INDICADOS	RIESGOS MATERIALIZADOS
2	0
RIM	0,00%

1) EVALUACIÓN DEL CICLO DE VIDA DEL PROYECTO (SDLC)	
PDP (Previsión Duración Prevista)	1,18
PTP (Precisión Trabajo Previsto)	1,09
CPE (Conformidad)	1,00
PRD (Producción)	1,04
NCP (Conformidad Verificaciones)	0,98
RIM (Riesgos)	0,00

USABILIDAD

ADECUACIÓN A LA TAREA	SI	NO	N/A
¿Los pasos que se dan en la aplicación son iguales u optimizan los que el usuario realizaba antes de la mecanización?			X
¿Está bien definida la jerarquía de opciones de los menús de la aplicación?		X	
¿La aplicación permite reducir el tiempo de realización de la tarea?	X		
¿La presentación de información es la oportuna, evitando una carga de trabajo mental innecesaria?			X
¿Es la interfaz compatible con los documentos fuente?	X		
¿Se utilizan valores por defecto o aquellos más significativos según la tarea?	X		
TOTAL	3	1	2
CARÁCTER AUTODESCRIPTIVO			
¿La aplicación te informa de todas las pantallas por las que has pasado?			X
¿Tienes posibilidad de acceder a las pantallas visitadas desde cualquier punto?	X		
¿Ofrece la aplicación información o ayuda contextual?	X		
¿Informa la aplicación de dónde estás en cada pantalla y lo que espera?			X
¿Los elementos del menú están claramente identificados?	X		
TOTAL	3	0	2
CONFORMIDAD CON LAS EXPECTATIVAS DEL USUARIO			
¿Se adecúa la terminología y el vocabulario a los conocimientos técnicos del usuario?	X		
¿Recibe el usuario información útil y efectiva de las acciones que se está llevando a cabo?	X		
Cuando hay una desviación en el tiempo de respuesta esperado por el usuario, ¿se le informa adecuadamente?	X		
¿El comportamiento y apariencia de un diálogo es consistente dentro de las tareas y entre tareas similares?	X		
¿Tienen los mensajes presentados al usuario un estilo constructivo y objetivo?			X
¿Se sigue una tabulación lógica entre campos de datos secuenciales?	X		
TOTAL	5	0	1
ADECUACIÓN DE APRENDIZAJE			
¿Dispone el usuario del acceso a las reglas y/o tutoriales del programa?	X		
¿La ayuda consigue una comprensión conceptual del sistema por parte del usuario?	X		
¿El diálogo ofrece información tanto de resultados finales como intermedios?			X
¿Puede el usuario probar y explorar los pasos del diálogo sin consecuencias negativas?		X	
¿Se establecen configuraciones por defecto para facilitar el uso a personal con conocimientos mínimos?	X		
TOTAL	3	1	1
CONTROL POR PARTE DEL USUARIO			
¿Controla el usuario la forma de continuar con el diálogo?			X
¿Es posible deshacer las operaciones de una tarea cuando su contexto y utilización lo permiten?	X		

Ante la presentación de un gran volumen de datos, ¿controla el usuario la forma en que se presentan?	X		
¿Puede un usuario modificar los valores que vienen establecidos por defecto?			X
Cuando se han modificado datos, ¿siguen disponibles los datos originales si lo exige la tarea?	X		
TOTAL	3	0	2
TOLERANCIA A ERRORES			
¿Ayuda el sistema al usuario a detectar o prevenir errores en la entrada?	X		
Cuando aparece un error, ¿se le da al usuario información para facilitar su corrección?	X		
Ante una validación, ¿se posiciona el cursor en la posición donde es necesaria la corrección?	X		
¿La verificación y validación de datos se realizan antes de que el sistema procese la entrada?	X		
Si de una acción del usuario se derivan consecuencias graves, ¿informa la aplicación de ello antes de ejecutarla?	X		
TOTAL	5	0	0
PERSONALIZACIÓN DEL DIÁLOGO			
¿La cantidad de información de apoyo se puede modificar de acuerdo con el nivel de conocimientos del usuario?	X		
¿Puede el usuario seleccionar el formato y tipo de los datos, tanto de entrada como de salida?	X		
¿Puede un usuario añadir o reorganizar distintas funcionalidades para adaptarlo a sus necesidades?			X
¿La personalización es reversible permitiendo volver a la configuración original?			X
TOTAL	2	0	2
TOTAL	24	2	US
		92,31%	

SEGURIDAD DE ACCESO			
Modo de Acceso	SI	NO	N/A
¿Se utiliza un sistema de acceso mediante identificador y contraseña?	X		
¿Es necesario ser usuario de Windows para acceder a la aplicación?	X		
¿Es obligatorio el uso de la infraestructura de clave pública PKI para acceder a la aplicación?			X
¿Hay restricciones de acceso por IP/MAC?			X
¿Se utiliza una tabla de códigos para acceder a la aplicación?			X
¿Se utiliza identificación por datos biométricos?			X
¿Se registran los intentos de acceso no autorizados?	X		
TOTAL	3	0	4
Restricciones de Seguridad			
¿Se bloquea la aplicación al introducir la contraseña erróneamente un número determinado de veces?	X		

¿Se bloquea automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?	X		
¿Se elimina automáticamente la cuenta de un usuario si no se utiliza durante un periodo de tiempo determinado?			X
¿Se genera la primera contraseña suministrada mediante un proceso aleatorio o pseudo-aleatorio?		X	
¿Hay restricciones de acceso por horario?			X
TOTAL	2	1	2
Registros de seguridad			
¿Queda registro de los accesos de cada usuario (fecha/hora)?	X		
¿Queda registro de la duración de la sesión de cada usuario?	X		
¿Queda registro de la manipulación de los datos?	X		
¿Existe un registro de incidencias?	X		
¿Se realizan copias de seguridad de los registros de accesos?	X		
¿Se guardan los registros de acceso en un entorno seguro?	X		
¿Existe la figura de responsable del fichero?	X		
TOTAL	7	0	0
Contraseñas			
¿Es obligatorio modificar la contraseña proporcionada inicialmente?			X
¿Se obliga a emplear claves de acceso fuertes (longitud mínima y máxima, que contengan números y letras en mayúsculas??	X		
¿Al cambiar la contraseña es obligatorio que la nueva contraseña guarde ciertas diferencias con la antigua (no poder usar secuencias de caracteres utilizados en la antigua)?	X		
¿Hay restricciones en los contenidos de las contraseñas (Ej.: No se puede incluir el nombre del usuario o su DNI en la contraseña)?	X		
¿Si se compromete una contraseña, se compromete el sistema?			X
¿Las contraseñas han de cambiarse cada cierto tiempo?	X		
TOTAL	4	0	2
Gestión de Usuarios			
¿Existe Una jerarquía de roles y funcionalidades del sistema a la que pertenezcan los usuarios del mismo?	X		
¿Existe un módulo de gestión de usuarios/contraseñas para crearlos, modificarlos o eliminarlos?	X		
¿Existe la figura de un administrador de la seguridad del sistema?	X		
TOTAL	3	0	0
Seguridad de los Datos			
¿Se registran las acciones de modificaciones de datos llevadas a cabo en el sistema?	X		
¿Se cifran los datos almacenados?			X

¿Se cifran las transmisiones de datos?	X		
¿Se han definido procedimiento para realizar copias de seguridad y verificarlas?	X		
¿Se clasifican los datos según la Ley orgánica de protección de Datos?	X		
TOTAL	4	0	1
Seguridad del sistema			
¿Se ha realizado un análisis de riesgos de seguridad del sistema?	X		
¿En el diseño de las interfaces de usuario se ha contemplado el nivel de conocimiento de los usuarios de forma que se reduzca el riesgo de comprometer el sistema por un mal uso de la aplicación?	X		
¿Existen procedimientos para aplicar regularmente los parches de seguridad a los servidores involucrados en el sistema?	X		
¿Existen políticas de seguridad de red (uso de firewalls, proxys, etc.) para limitar los ataques externos al sistema?	X		
¿Los equipos donde se instala la aplicación cuentan con antivirus actualizado?	X		
¿En los equipos donde se instala la aplicación está restringida la instalación de software externo?	X		
TOTAL	6	0	0
TOTAL	29	1	9
	96,67%		

MANTENIBILIDAD		2) INDICADORES OPERATIVOS		3) INDICADORES DE CALIDAD DEL CÓDIGO FUENTE (SONARQUBE)	
TOP (Minutos en Operación)	44.460,00	FUNCIONALIDAD	100%	LINEAS DUPLICADAS	7,80%
TMA (Minutos en Mantenimiento)	3.257,00	USABILIDAD	92,31%	COBERTURA	40,90%
TFA (Minutos en Fallos)	1.141,00	SEGURIDAD	96,97%	COMENTARIOS	30,90%
TIN (Minutos Indisponible)	4.398,00	MANTENIBILIDAD	92,70%	DEUDA TÉCNICA	0,30%
		FIABILIDAD	97,44%	VULNERABILIDADES	2
		DISPONIBILIDAD	90,15%		

Tabla A1-10 Métricas Cuantitativas GesLog (elaboración propia)

ANEXO II: MÉTRICAS CUALITATIVAS

A2-1 EQUIPO INTERNO (*IN-HOUSE*)

Métricas Cualitativas	Cuestiones	Muy poco / Muy baja	Poco / Baja	Correcto	Bastante buena	Mucho / Muy buena	Total	Media
Talento	En el puesto actual ¿estás motivado para desarrollar tus habilidades?	0	0	3	7	2	47	47,2
	¿Aprendes nuevas habilidades rápidamente?	0	0	5	4	3	46	
	Cuando se presentan problemas, ¿eres innovador?	0	1	4	3	4	46	
	Ante nuevas competencias ¿te adaptas fácilmente?	0	1	1	7	3	48	
	En el puesto actual, ¿crees que tu potencial está siendo aprovechado?	0	0	4	3	5	49	
Experiencia	Con tu experiencia ¿eres capaz de enseñar a otros?	0	2	6	2	2	40	46,4
	Según tu trayectoria profesional, expón la calidad de la experiencia acumulada	0	2	3	4	3	44	
	¿Crees que tu experiencia te ayuda a solventar problemas actuales?	0	0	0	11	1	49	
	Ante problemas futuros, ¿crees que tu experiencia te permitirá anticiparte?	0	0	2	6	4	50	
	Con situaciones difíciles actuales, ¿tu equipo cuenta con tu experiencia?	0	0	1	9	2	49	

Conocimiento	En el trabajo, ¿crees que tu conocimiento es efectivo?	0	0	1	10	1	48	45,8
	¿Crees que dominas las herramientas en tu trabajo?	0	1	3	5	3	46	
	¿Tu conocimiento es aprovechado por el equipo en su conjunto?	0	0	3	9	0	45	
	¿Cómo valoras tu capacidad para transmitir conocimiento a otros?	0	1	2	7	2	46	
	Con el conocimiento del que dispones ¿tu equipo e consulta frecuentemente?	0	0	6	4	2	44	
Resolución de Problemas Complejos	¿Cuál es la frecuencia con la que aparecen problemas complejos en tu trabajo?	0	2	7	1	2	39	46
	Para la resolución de problemas, ¿cómo evalúas tu capacidad para dividirlo en partes manejables?	0	0	2	7	3	49	
	¿Eres capaz de identificar la causa de un problema complejo rápidamente?	0	0	3	7	2	47	
	Ante problemas críticos, ¿tu equipo cuenta contigo?	0	0	0	9	3	51	
	Antes de tomar una decisión, ¿analizas el problema desde varias perspectivas?	0	0	5	6	1	44	
Pensamiento Crítico	Antes de aceptar una solución ante un problema, ¿evalúas la información?	1	1	4	4	2	41	47,4
	Ante una decisión importante, ¿reflexionas sus consecuencias?	0	0	0	5	7	55	

	Ante el planteamiento del equipo, ¿expones una crítica constructiva?	0	1	1	9	1	46	
	Ante una decisión importante, ¿consideras todos los puntos de vista?	0	0	4	6	2	46	
	En la toma de decisiones, ¿utilizas el pensamiento crítico?	0	0	1	9	2	49	
Adaptabilidad	¿Te adaptas a nuevas tecnologías y herramientas fácilmente?	0	0	5	6	1	44	46,8
	¿Eres flexible cuando cambian las condiciones del trabajo manteniendo el rendimiento?	0	1	2	9	0	44	
	Ante nuevas prioridades o proyectos, ¿cómo te adaptas?	0	1	6	2	3	43	
	¿El equipo se adaptable fácilmente?	0	0	2	9	1	47	
	¿Qué capacidad tienes para asumir nuevas responsabilidades cuando se requiere?	0	0	0	4	8	56	
Comunicación Efectiva	¿Expones tus ideas de manera clara y efectiva?	0	0	3	6	3	48	47,6
	¿Escuchas activamente a los demás?	0	0	5	6	1	44	
	¿Eres capaz de transmitir mensajes complejos a tu equipo de forma sencilla?	0	0	3	8	1	46	
	¿Eres transparente en la relación con tus colegas?	1	0	1	1	9	53	
	¿Tu equipo comprende exactamente los mensajes que transmites?	0	0	3	7	2	47	

Creatividad	¿Puedes proponer soluciones creativas al equipo?	0	1	1	2	8	53	45
	En tu equipo, ¿se estiman tus ideas innovadoras?	0	0	3	7	2	47	
	Frente a retos técnicos, ¿cómo valoras tu capacidad de innovación?	0	0	6	5	1	43	
	¿Tu equipo genera ideas innovadoras?	0	2	2	7	1	43	
	¿Te gusta experimentar con nuevas tecnologías?	0	2	7	1	2	39	
Colaboración	¿Tu equipo trabaja de forma colaborativa?	0	0	2	8	2	48	47,6
	¿Es fácil trabajar con otros equipos?	0	0	1	9	2	49	
	¿Valora el equipo la participación de todos sus miembros?	0	0	0	11	1	49	
	¿Es efectiva la comunicación entre los miembros del equipo?	0	1	1	10	0	45	
	¿Colaboras con compañeros en problemas complejos?	1	0	1	7	3	47	
Gestión del tiempo	¿Organizas tu tiempo de forma efectiva?	1	0	3	6	2	44	44,8
	¿Gestionas tus prioridades de tareas correctamente?	1	1	1	1	8	50	
	Dentro de los tiempos acordados, ¿gestionas bien el tiempo?	0	1	2	6	3	47	
	Si hay interrupciones, ¿te resulta fácil gestionar sin afectar a la productividad?	2	1	2	6	1	39	
	¿Planificas eficientemente tus tareas diarias?	1	1	4	1	5	44	

Responsabilidad	¿Cómo es tu nivel de compromiso?	0	0	1	2	9	56	50,8
	¿Cumples los objetivos sin supervisión?	0	1	1	5	5	50	
	¿Revisas tu trabajo antes de su entrega?	0	1	0	6	5	51	
	¿Tienes la responsabilidad de mantener documentado y organizado el código?	0	1	4	6	1	43	
	¿Asumes tareas adicionales para cumplir con los objetivos del equipo?	0	0	0	6	6	54	
Orientación al detalle	¿Te aseguras que el código cumple con buenas prácticas antes de entregarlo?	0	0	2	10	0	46	46,2
	¿Tienes capacidad para identificar errores o fallos fácilmente?	0	0	5	4	3	46	
	¿Aseguras la calidad del código?	0	0	7	2	3	44	
	¿Eres detallista en la documentación y tareas que realizas?	1	1	3	4	3	43	
	¿Eres meticuloso cuando implementas nuevas funcionalidades?	0	1	0	5	6	52	
		9	29	160	349	173	2808	46,8
		1%	4%	22%	48%	24%		

Tabla A2-1 Métricas Cualitativas Equipo Interno (*In-house*) (elaboración propia)

A2-2 EQUIPO EXTERNO (OUTSOURCING)

Equipo Externo (Outsourcing)								
Métricas Cualitativas	Cuestiones	Muy poco / Muy baja	Poco / Baja	Correcto	Bastante buena	Mucho / Muy buena	Total	Media
Talento	En el puesto actual ¿estás motivado para desarrollar tus habilidades?	0	0	0	12	0	48	48
	¿Aprendes nuevas habilidades rápidamente?	0	0	0	9	3	51	
	Cuando se presentan problemas, ¿eres innovador?	0	0	3	9	0	45	
	Ante nuevas competencias ¿te adaptas fácilmente?	0	0	0	9	3	51	
	En el puesto actual, ¿crees que tu potencial está siendo aprovechado?	0	0	3	9	0	45	
Experiencia	Con tu experiencia ¿eres capaz de enseñar a otros?	0	0	2	8	2	48	46,2
	Según tu trayectoria profesional, expón la calidad de la experiencia acumulada	0	0	0	12	0	48	
	¿Crees que tu experiencia te ayuda a solventar problemas actuales?	0	0	3	9	0	45	
	Ante problemas futuros, ¿crees que tu experiencia te permitirá anticiparte?	0	0	8	2	2	42	
	Con situaciones difíciles actuales, ¿tu equipo cuenta con tu experiencia?	0	0	6	0	6	48	
Conocimiento	En el trabajo, ¿crees que tu conocimiento es efectivo?	0	0	0	12	0	48	48,6
	¿Crees que dominas las herramientas en tu trabajo?	0	0	0	12	0	48	

	¿Tu conocimiento es aprovechado por el equipo en su conjunto?	0	0	3	9	0	45	
	¿Cómo valoras tu capacidad para transmitir conocimiento a otros?	0	0	2	2	8	54	
	Con el conocimiento del que dispones ¿tu equipo te consulta frecuentemente?	0	0	2	8	2	48	
Resolución de Problemas Complejos	¿Cuál es la frecuencia con la que aparecen problemas complejos en tu trabajo?	0	0	2	2	8	54	48
	Para la resolución de problemas, ¿cómo evalúas tu capacidad para dividirlo en partes manejables?	0	0	0	9	3	51	
	¿Eres capaz de identificar la causa de un problema complejo rápidamente?	0	0	3	9	0	45	
	Ante problemas críticos, ¿tu equipo cuenta contigo?	0	0	6	0	6	48	
	Antes de tomar una decisión, ¿analizas el problema desde varias perspectivas?	0	0	6	6	0	42	
Pensamiento Crítico	Antes de aceptar una solución ante un problema, ¿evalúas la información?	0	0	3	9	0	45	43,8
	Ante una decisión importante, ¿reflexionas sus consecuencias?	0	0	3	9	0	45	
	Ante el planteamiento del equipo, ¿expones una crítica constructiva?	0	0	8	2	2	42	
	Ante una decisión importante, ¿consideras todos los puntos de vista?	0	0	3	9	0	45	
	En la toma de decisiones, ¿utilizas el pensamiento crítico?	0	0	6	6	0	42	

Adaptabilidad	¿Te adaptas a nuevas tecnologías y herramientas fácilmente?	0	0	0	9	3	51	49,2
	¿Eres flexible cuando cambian las condiciones del trabajo manteniendo el rendimiento?	0	0	3	9	0	45	
	Ante nuevas prioridades o proyectos, ¿cómo te adaptas?	0	0	2	8	2	48	
	¿El equipo se adaptable fácilmente?	0	0	2	2	8	54	
	¿Qué capacidad tienes para asumir nuevas responsabilidades cuando se requiere?	0	0	0	12	0	48	
Comunicación Efectiva	¿Expones tus ideas de manera clara y efectiva?	0	0	2	8	2	48	47,4
	¿Escuchas activamente a los demás?	0	0	3	9	0	45	
	¿Eres capaz de transmitir mensajes complejos a tu equipo de forma sencilla?	0	0	9	0	3	42	
	¿Eres transparente en la relación con tus colegas?	0	0	3	0	9	54	
	¿Tu equipo comprende exactamente los mensajes que transmites?	0	0	2	8	2	48	
Creatividad	¿Puedes proponer soluciones creativas al equipo?	0	0	0	3	9	57	49,2
	En tu equipo, ¿se estiman tus ideas innovadoras?	0	0	0	12	0	48	
	Frente a retos técnicos, ¿cómo valoras tu capacidad de innovación?	0	0	3	9	0	45	
	¿Tu equipo genera ideas innovadoras?	0	0	3	9	0	45	

	¿Te gusta experimentar con nuevas tecnologías?	0	0	0	9	3	51	
Colaboración	¿Tu equipo trabaja de forma colaborativa?	0	0	0	9	3	51	48,6
	¿Es fácil trabajar con otros equipos?	0	0	6	6	0	42	
	¿Valora el equipo la participación de todos sus miembros?	0	0	0	12	0	48	
	¿Es efectiva la comunicación entre los miembros del equipo?	0	0	2	8	2	48	
	¿Colaboras con compañeros en problemas complejos?	0	0	2	2	8	54	
Gestión del tiempo	¿Organizas tu tiempo de forma efectiva?	0	0	0	12	0	48	45,6
	¿Gestionas tus prioridades de tareas correctamente?	0	0	3	9	0	45	
	Dentro de los tiempos acordados, ¿gestionas bien el tiempo?	0	0	0	12	0	48	
	Si hay interrupciones, ¿te resulta fácil gestionar sin afectar a la productividad?	0	0	6	6	0	42	
	¿Planificas eficientemente tus tareas diarias?	0	0	3	9	0	45	
Responsabilidad	¿Cómo es tu nivel de compromiso?	0	0	0	6	6	54	52,2
	¿Cumples los objetivos sin supervisión?	0	0	2	8	2	48	
	¿Revisas tu trabajo antes de su entrega?	0	0	0	0	12	60	

	¿Tienes la responsabilidad de mantener documentado y organizado el código?	0	0	0	9	3	51	
	¿Asumes tareas adicionales para cumplir con los objetivos del equipo?	0	0	0	12	0	48	
Orientación al detalle	¿Te aseguras que el código cumple con buenas prácticas antes de entregarlo?	0	0	0	6	6	54	49,2
	¿Tienes capacidad para identificar errores o fallos fácilmente?	0	0	8	2	2	42	
	¿Aseguras la calidad del código?	0	0	2	2	8	54	
	¿Eres detallista en la documentación y tareas que realizas?	0	0	2	8	2	48	
	¿Eres meticulouso cuando implementas nuevas funcionalidades?	0	0	0	12	0	48	
TOTALES		0	0	140	440	140	2880	48
PORCENTAJES TOTALES		0%	0%	19%	61%	19%		

Tabla A2-2 Métricas Cualitativas Equipo Externo (*Outsourcing*) (elaboración propia)

ANEXO III: MÉTRICAS CUANTITATIVAS (COMPLEMENTARIAS)

Para lograr esta capacidad, es indispensable disponer de métricas que permitan medir de manera objetiva y cuantitativa las características del software. A continuación, se examinan las capacidades clave del software junto con las métricas asociadas para su evaluación. Aunque existen numerosas métricas disponibles, se reflejan las más relevantes:

Funcionalidad

Se entiende como funcionalidad, la capacidad del software para ser empleado en tareas y funciones específicas para las que fue diseñado.

Métricas	Alcance	Fórmula
Cobertura de requisitos	Mide el porcentaje que han sido verificados mediante pruebas	$\text{Cobertura de requisitos (\%)} = (\text{N}^\circ \text{ requisitos implementados} / \text{N}^\circ \text{ requisitos totales}) * 100$
Cobertura de pruebas	Porcentaje de código ejecutado durante las pruebas	$\text{Cobertura de pruebas (\%)} = \text{N}^\circ \text{ líneas de código cubiertas por prueba} / \text{N}^\circ \text{ total de líneas de código}$
Tasa de éxito de pruebas (<i>Test Past Rate</i>)	Mide el porcentaje de pruebas pasadas con éxito	$\text{Tasa de éxito de pruebas (\%)} = (\text{N}^\circ \text{ pruebas exitosas} / \text{N}^\circ \text{ total de pruebas ejecutadas}) * 100$
Tasa de defectos funcionales	Número de defectos localizados en las funcionalidades principales durante las pruebas unitarias de código	$\text{Tasa de defectos funcionales} = \text{N}^\circ \text{ de defectos funcionales} / \text{Tamaño total de funcionalidades}$
Tiempo de respuesta	Indica el tiempo que tarda el software en responder a una solicitud	$\text{Tiempo de respuesta} = \text{Tiempo transcurrido desde la solicitud}$
Tasa de transferencia efectiva (<i>Throughput</i> ¹³)	Mide la cantidad de trabajo que el software puede realizar en un tiempo determinado (rendimiento)	$\text{Rendimiento} = \text{N}^\circ \text{ total de transacciones procesadas} / \text{Unidad de tiempo}$
Tasa de errores de tiempo de ejecución	Muestra la frecuencia de errores durante la ejecución del software	$\text{Tasa de errores de ejecución} = (\text{N}^\circ \text{ de errores de ejecución} / \text{N}^\circ \text{ total de ejecuciones}) * 100$

¹³ Tasa de transferencia efectiva, *throughput*, es el volumen de información a través de un sistema. Es particularmente significativo en almacenamiento de información y sistemas de recuperación de información. Aparte, también se define como la velocidad real a través de una red telemática. [\[53\]](#)

Satisfacción del usuario	Evalúa si el software cumple con las expectativas funcionales	Satisfacción del usuario (%) = $(\text{N}^{\circ} \text{ respuestas positivas} / \text{N}^{\circ} \text{ total de encuestas}) * 100$
Coste total	Coste total del proyecto incluyendo salarios, costes asociados, ...	Inversión total requerida para completar el desarrollo, incluyendo salarios, tarifas de proveedores y otros costos relacionados.
Tiempo de ciclo (<i>Cycle Time</i>)	Tiempo de una tarea entre fases	Tiempo de ciclo = Fecha de finalización – Fecha de inicio
Tiempo de resolución (<i>Resolution time</i>)	Tiempo en que una tarea cambia de estado a “Resuelto”	Tiempo de resolución = Fecha de resolución – Fecha de creación
Porcentaje de funcionalidades completadas vs planificadas	Porcentaje de funcionalidades completadas vs planificadas	Porcentaje de funcionalidades completadas (%) = $((\text{N}^{\circ} \text{ de funcionalidades completadas} / \text{N}^{\circ} \text{ de funcionalidades planificadas}) * 100$
Velocidad de desarrollo (<i>Velocity</i>)	Velocidad de desarrollo	Velocidad de desarrollo = Puntos de historia completados por sprint

Tabla A3-1 Funcionalidad [\[34\]](#)

Seguridad

Es la capacidad necesaria para proteger el software de amenazas potenciales (accesos no autorizados, inyección de código malicioso, ...).

Métricas	Alcance	Fórmula
Número de vulnerabilidades descubiertas	Número de vulnerabilidades de seguridad identificadas en las pruebas o auditorias	Número de vulnerabilidades descubiertas = Total de vulnerabilidades identificadas
Tasa de incidentes de seguridad	Número de incidentes de seguridad en un tiempo determinado cuando el aplicativo está en producción	Tasa de incidentes de seguridad = $\text{Número de incidentes} / \text{Tiempo en producción}$
Tiempo para resolver vulnerabilidades	Tiempo necesario para corregir una vulnerabilidad desde que se descubre hasta su resolución	Tiempo de resolución de vulnerabilidades = $\text{Tiempo total desde la detección hasta la resolución de la vulnerabilidad} / \text{N}^{\circ} \text{ total de vulnerabilidades}$
Cobertura de seguridad	Porcentaje de componentes del software que ha sido sometido a pruebas de seguridad	Cobertura de seguridad = $\text{N}^{\circ} \text{ de elementos cubiertos por pruebas de seguridad} / \text{N}^{\circ} \text{ total de elementos del sistema}$

Aseguramiento de la Calidad	Nº de errores críticos y problemas de rendimiento en los primeros 6 meses después de su despliegue en producción. (encontrados durante o después del desarrollo)	Porcentaje de errores críticos = $(\text{Nº de errores críticos en los primeros 6 meses} / \text{Nº total de errores}) * 100$
Frecuencia de Parcheo de Seguridad	Cada cuánto tiempo se aplican parches de seguridad al aplicativo	Frecuencia de Parcheo de Seguridad = $\text{Nº de parches aplicados} / \text{Unidad de tiempo}$
Impacto de Vulnerabilidades Críticas	Categorización de vulnerabilidades por nivel de impacto	Impacto de Vulnerabilidades Críticas = $\text{Nº de vulnerabilidades críticas} * \text{Impacto promedio por vulnerabilidad}$
Tiempo promedio de detección de vulnerabilidades	Tiempo que se tarda en detectar una vulnerabilidad	Tiempo promedio de detección = $\text{Tiempo total desde la aparición hasta la detección} / \text{Nº total de vulnerabilidades detectadas}$

Tabla A3-2 Seguridad [34]

Disponibilidad

Es la capacidad del software para estar operativo y accesible a los usuarios.

Métricas	Alcance	Fórmula
Disponibilidad	Tiempo en el que el software está disponible en relación al tiempo total.	Disponibilidad (%) = $(\text{Tiempo operativo} / \text{Tiempo total}) * 100$
Tiempo entre fallos (MTBF- <i>Mean Time Between Failures</i>)	Promedio de tiempo entre fallos	MTBF = $\text{Tiempo total operativo} / \text{Número de fallos}$
Tiempo para reparar errores o incidencias (MTTR - <i>Mean Time To Repair</i>)	Tiempo promedio de resolución de errores o incidencias.	MTTR = $\text{Tiempo total de reparación} / \text{Nº total de fallos}$
Disponibilidad operativa	Disponibilidad del software en un periodo de tiempo determinado.	Disponibilidad operativa = $\text{MTBF} / (\text{MTBF} + \text{MTTR})$
Tiempo de inactividad (<i>Downtime</i>)	Tiempo de inactividad cuando se produce un fallo o incidencia.	<i>Downtime</i> = $\text{Tiempo total} - \text{Tiempo operativo}$
Frecuencia de fallos	Número de fallos e incidentes producidos en un período de tiempo determinado.	Frecuencia de fallos = $\text{Número de fallos} / \text{Tiempo total}$
Tasa de fallos por hora	Determina el número de fallos en un período determinado.	Tasa de fallos por hora = $\text{Número de fallos} / \text{Horas de operación}$

Tiempo medio de respuesta	Calcula el tiempo medio de respuesta.	Tiempo medio de respuesta = Tiempo total de respuesta / Número de solicitudes
Tiempo de recuperación (RTO)	Tiempo máximo permitido para que el aplicativo vuelva a estar operativo.	No dispone de fórmula. Se determina en los acuerdos de nivel de servicio (SLA).

Tabla A3-3 Disponibilidad [34]

Fiabilidad

La fiabilidad consiste en la capacidad para que **el software pueda** funcionar sin fallos o interrupciones durante un período de tiempo. Los aspectos a considerar entre los requisitos de fiabilidad software son:

Métricas	Alcance	Fórmula
Tiempo medio entre fallos (MTBF)	Tiempo promedio que el software opera sin fallar	$MTBF = \text{Tiempo total de operación} / \text{Número total de fallos}$
Tiempo promedio de recuperación (MTTR)	Tiempo promedio necesario para recuperar el sistema tras un fallo	$MTTR = \text{Tiempo total de inactividad} / \text{Nº total de fallos}$
Disponibilidad del sistema	Mide el porcentaje de tiempo en el que el sistema está operativo y disponible	$\text{Disponibilidad del sistema} = (\text{MTBF} / \text{MTBF} + \text{MTTR}) * 100$
Tasa de fallos	Mide la frecuencia con la que ocurren fallos en el software durante un período de tiempo	$\text{Tasa de fallos} = \text{Nº de fallos} / \text{Unidad de tiempo}$
Tasa de criticidad del fallo	Mide la proporción de fallos que se consideran críticos según su impacto y gravedad en el sistema	$\text{Tasa de criticidad del fallo} = \text{Nº fallos críticos} / \text{Nº total de fallos}$
Número de Incidentes por Tipo	Tipos de incidentes según el tipo de error	$\text{Nº de Incidentes por Tipo} = \text{Nº de incidentes de cada tipo}$
Porcentaje de Fallos Recurrentes	Fallos que se repiten después de su resolución	$\text{Porcentaje de Fallos Recurrentes} = (\text{Nº de fallos recurrentes} / \text{Nº total de fallos}) * 100$
Tiempo Promedio de Resolución	Tiempo medio necesario para resolver un fallo desde su inicio hasta su resolución	$\text{Tiempo Promedio de Resolución} = \text{Tiempo total para resolver fallos} / \text{Nº total de fallos resueltos}$
Tasa de Cumplimiento de SLAs	Porcentaje de fallos del tiempo acordado en los SLAs	$\text{Tasa de Cumplimiento de SLAs} = (\text{Nº de fallos resueltos dentro del SLA} / \text{Nº total de fallos}) * 100$
Frecuencia de Actualización de Parches	Frecuencia con la que se actualizan parches o actualizaciones	$\text{Frecuencia de Actualización de Parches} = \text{Nº de parches aplicados} / \text{Unidad de tiempo}$

Tabla A3-4 Fiabilidad [34]

Usabilidad

La usabilidad hace referencia al aspecto que incluye la forma en que el usuario interactúa con la aplicación.

Métricas	Alcance	Fórmula
Tiempo de aprendizaje	Promedio de tiempo que los usuarios nuevos necesitan para adecuarse a las tareas específicas de manera efectiva	$\text{Tiempo de aprendizaje} = \frac{\text{Tiempo total necesario para que un usuario se adecue al software}}{\text{Nº total de usuarios}}$
Tiempo para completar tareas	Mide el tiempo que un usuario tarda en completar tareas específicas	$\text{Tiempo en completar tareas} = \frac{\text{Tiempo total empleado por los usuarios en completar tareas}}{\text{Nº total de usuarios}}$
Tasa de error del usuario	Frecuencia con la que los usuarios cometen errores durante el uso del software	$\text{Tasa de error del usuario} = \frac{\text{Nº de errores cometidos}}{\text{Nº total de acciones realizadas}}$
Satisfacción del usuario	Mide la percepción del usuario sobre la facilidad en su uso	$\text{Satisfacción del usuario (\%)} = \left(\frac{\text{Respuestas positivas}}{\text{Total de encuestas}} \right) * 100$
Tiempo de error	Promedio que tarda un usuario en corregir un error que cometió	$\text{Tiempo de error} = \frac{\text{Tiempo total para corregir errores}}{\text{Nº de errores corregidos}}$
Facilidad de aprendizaje	Rapidez con que los nuevos usuarios completan tareas sin ayuda	$\text{Facilidad de aprendizaje} = \frac{\text{Nº de tareas completadas sin ayuda}}{\text{Nº total de tareas}}$
Tasa de abandono de tarea	Frecuencia en que los usuarios abandonan tareas sin hacer	$\text{Tasa de abandono de tarea} = \frac{\text{Nº de tareas abandonadas}}{\text{Nº total de tareas iniciadas}}$
Número de revisiones requeridas	Nº de veces que una tarea necesita ser revisada antes de completarse	$\text{Número de revisiones} = \frac{\text{Nº total de revisiones}}{\text{Nº de tareas o proyectos}}$

Tabla A3-5 Usabilidad [34]

Mantenibilidad

Es la capacidad de la que dispone un software para poder ser modificado, implementar funcionalidades o mejorar su rendimiento. Es buena práctica disponer de la documentación ya que permite a los programadores que no participaron en su desarrollo poder continuar desarrollando las funcionalidades.

Métricas	Alcance	Fórmula
Complejidad ciclomática	Medida que determina cómo de complejo es el código y que influye en la facilidad de su mantenimiento	$\text{Complejidad ciclomática} = \text{Nº de rutas lineales independientes en el código fuente}$

Tasa defectos tras modificaciones	Número de defectos introducidos durante el mantenimiento	Tasa de defectos tras modificaciones = $\text{N}^\circ \text{ de defectos introducidos} / \text{N}^\circ \text{ total de modificaciones}$
Densidad de defectos post-entrega	Mide la cantidad de defectos encontrados después de la entrega del software	Densidad de defectos post-entrega = $\text{N}^\circ \text{ total de defectos encontrados} / \text{Tamaño del software en KLOC}$
Propagación de cambios	Mide el impacto que tiene un cambio en el código sobre otros módulos	Propagación de cambios = $(\text{N}^\circ \text{ de módulos afectados por el cambio} / \text{N}^\circ \text{ total de módulos}) * 100$
Relación comentarios / código	Mide la proporción de comentarios en el código	Relación comentarios / código = $(\text{N}^\circ \text{ de líneas de comentarios} / \text{N}^\circ \text{ total de líneas de código}) * 100$
Duplicación de código	Mide el porcentaje de código duplicado dentro del software	Duplicación de código = $(\text{N}^\circ \text{ de líneas de código duplicadas} / \text{N}^\circ \text{ total de líneas de código}) * 100$
Tiempo medio de resolución de defectos	Tiempo promedio para resolver defectos encontrados durante el mantenimiento	Tiempo Medio de Resolución = $\text{Tiempo total para resolver defectos} / \text{N}^\circ \text{ total de defectos resueltos}$
Porcentaje de funcionalidades modificadas	Porcentaje de funcionalidades modificadas durante el mantenimiento	Porcentaje de Funcionalidades Modificadas = $(\text{N}^\circ \text{ de funcionalidades modificadas} / \text{N}^\circ \text{ total de funcionalidades}) * 100$
Tasa de reutilización de código	Porcentaje de código reutilizado	Tasa de Reutilización de Código = $(\text{N}^\circ \text{ de líneas de código reutilizadas} / \text{N}^\circ \text{ total de líneas de código}) * 100$
Tiempo de implementación de mejoras	Tiempo necesario para implementar mejoras en el software	Tiempo de Implementación de Mejoras = $\text{Tiempo total para implementar mejoras} / \text{N}^\circ \text{ total de mejoras implementadas}$
Eficiencia de refactorización	Efectividad de mejora en la calidad del código	Eficiencia de Refactorización = $(\text{Mejora en la calidad del código después de refactorización} / \text{N}^\circ \text{ de líneas refactorizadas}) * 100$

Tabla A3-6 Mantenibilidad [34]

Compatibilidad

La compatibilidad es una característica que indica que el software pueda funcionar correctamente en diferentes entornos, sistemas operativos, navegadores o dispositivos.

Métricas	Alcance	Fórmula
----------	---------	---------

Tasa de fallos en distintos entornos	Números de fallos al ejecutar el software en diferentes configuraciones de hardware o software	Tasa de fallos en distintos entornos = $(\text{N}^\circ \text{ de fallos} / \text{N}^\circ \text{ total de entornos probados}) * 100$
Cobertura de pruebas de compatibilidad	Muestra el porcentaje en diferentes entornos, sistemas operativos, navegadores, dispositivos, ... donde se ha probado el desarrollo con éxito	Cobertura de pruebas de compatibilidad (%) = $(\text{N}^\circ \text{ entornos probados con éxito} / \text{N}^\circ \text{ total de entornos}) * 100$
Compatibilidad entre versiones	Mide la capacidad de software para funcionar correctamente en diferentes versiones	Compatibilidad entre versiones = $\text{N}^\circ \text{ de versiones de software ejecutadas correctamente} / \text{N}^\circ \text{ total de versiones}$
Esfuerzo de migración	Mide el tiempo y los recursos necesarios para migrar el software a otro entorno	Esfuerzo de migración = $\text{Tiempo y recursos invertidos en la migración} / \text{Tamaño del software en KLOC}^{14}$
Tasa de cobertura de dispositivos [63]	Porcentaje de dispositivos (móviles, tabletas, PCs) donde se ha probado con éxito	Tasa de cobertura de dispositivos = $(\text{N}^\circ \text{ de dispositivos probados con éxito} / \text{N}^\circ \text{ total de dispositivos objetivo}) * 100$
Tasa de compatibilidad con navegadores [64]	Porcentaje de compatibilidad con diferentes navegadores	Tasa de compatibilidad con navegadores = $(\text{N}^\circ \text{ de navegadores con pruebas exitosas} / \text{N}^\circ \text{ total de navegadores probados}) * 100$
Tasa de compatibilidad de sistemas operativos	Porcentaje de compatibilidad en diferentes sistemas operativos	Tasa de compatibilidad de sistemas operativos = $(\text{N}^\circ \text{ de sistemas operativos probados con éxito} / \text{N}^\circ \text{ total de sistemas operativos objetivo}) * 100$
Tasa de interoperabilidad	Porcentaje de compatibilidad con otros sistemas o plataformas externas, como APIs ...	Tasa de interoperabilidad = $(\text{N}^\circ \text{ de integraciones funcionales} / \text{N}^\circ \text{ total de integraciones requeridas}) * 100$
Tasa de fallos de instalación	Cantidad de fallos que ocurren durante la instalación	Tasa de fallos de instalación = $(\text{N}^\circ \text{ de fallos durante la instalación} / \text{N}^\circ \text{ total de intentos de instalación}) * 100$

Tabla A3-7 Compatibilidad [34]

Portabilidad

Es la capacidad del software para que pueda ser migrado y funcionar en diferentes entornos/plataformas sin necesidad de modificaciones significativas.

Métricas	Alcance	Fórmula
Tasa de éxito de migraciones	Porcentaje de migraciones con éxito a otros entornos/plataformas sin la necesidad de modificaciones significativas	Tasa de éxito de migraciones (%) = $(\text{N}^\circ \text{ migraciones exitosas} / \text{N}^\circ \text{ total de migraciones}) * 100$

¹⁴ Miles de líneas de código, *thousands of lines of code* (KLOC), es la medida expresada en miles, que se utiliza para medir la complejidad, el tamaño del software y evaluar el esfuerzo requerido, la productividad y la calidad del código desarrollado. [54]

Tiempo de adaptación a nuevas plataformas	Tiempo requerido en adaptar y hacer que el software funcione en una nueva plataforma	Tiempo de adaptación a nuevas plataformas = Tiempo requerido para adaptar / N° total de plataformas
N° de entornos soportados	Indica cuántos entornos diferentes de software puede operar sin requerir modificaciones	N° de entornos soportados = Cantidad de plataformas o sistemas operativos donde el software funciona
Esfuerzo de migración	Mide la cantidad de recursos para adaptar el software a nuevos entornos	Esfuerzo de migración = Tiempo y recursos invertidos en la migración / Tamaño del software en KLOC
Tasa de compatibilidad entre versiones	Capacidad de operar en diferentes versiones de un entorno sin modificaciones	Tasa de compatibilidad entre versiones = (N° de versiones soportadas / N° total de versiones) * 100
Coste de adaptación a nuevas plataformas	Coste total asociado a la adaptación de nuevas plataformas	Costo de adaptación = Coste total de adaptar a nuevas plataformas / N° de plataformas adaptadas
Tasa de Migración de Datos	Proporción de datos migrados con éxito	Tasa de migración de datos = (N° de datos migrados exitosamente / N° total de datos migrados) * 100

Tabla A3-8 Portabilidad [\[34\]](#)

Proceso de Desarrollo e Impacto del Software

Métricas	Alcance	Fórmula
Tiempo de desarrollo	Duración total del proyecto, desde su inicio hasta su finalización.	Tiempo de desarrollo = Fecha de entrega – Fecha de inicio
Velocidad de implementación	Mide el tiempo promedio por iteración.	Velocidad de implementación = Fecha de lanzamiento – Fecha de inicio) / N° iteraciones
Adopción de usuario	Porcentaje en donde los usuarios se familiarizan con la nueva aplicación	Tasa de adopción = (N° de nuevos usuarios / N° de usuarios objetivo) * 100
Tasa de retención de usuarios	Porcentaje que indica el n° de usuarios que siguen utilizando la aplicación después del período inicial	Tasa de retención = (N° de usuarios activos al final del periodo / N° de usuarios al inicio del período) * 100
Satisfacción del usuario	Grado de satisfacción de los usuarios a través de encuestas y valoraciones	Puntuación de satisfacción = $\sum$ Puntuaciones de encuestas / N° de encuestas

Tasa de uso de funcionalidades	Frecuencia de uso de diferentes funcionalidades	Tasa de uso de funcionalidades = $(\text{N}^{\circ} \text{ de usos de una funcionalidad} / \text{N}^{\circ} \text{ total de usos}) * 100$
Tasa de innovación	Cantidad de mejoras implementadas en un período de tiempo	Tasa de innovación = $\text{N}^{\circ} \text{ de nuevas características} / \text{Período de tiempo}$
Cumplimiento de plazos	Tiempo utilizado para cumplir los plazos establecidos	Cumplimiento de plazos = $(\text{N}^{\circ} \text{ de entregas a tiempo} / \text{N}^{\circ} \text{ total de entregas}) * 100$
Tasa de crecimiento de usuarios	Cantidad del aumento de usuarios en un periodo concreto	Tasa de crecimiento de Usuarios = $(\text{N}^{\circ} \text{ de nuevos usuarios} / \text{N}^{\circ} \text{ de usuarios iniciales}) * 100$
Tasa de eficiencia de desarrollo	Relación entre tiempo invertido en el desarrollo y los resultados obtenidos	Tasa de eficiencia = $\text{N}^{\circ} \text{ de funcionalidades entregadas} / \text{Tiempo total de desarrollo}$
Coste de implementación por funcionalidad	Coste medio de cada funcionalidad	Coste de implementación por funcionalidad = $\text{Coste total del proyecto} / \text{N}^{\circ} \text{ funcionalidades implementadas}$
Número de iteraciones	Nº de versiones antes de la entrega final	Nº de iteraciones = $\text{N}^{\circ} \text{ total de versiones entregadas durante el desarrollo}$
Tiempo de respuesta de implementación	Tiempo medio necesario para implementar cambios	Tiempo de respuesta = $\text{Tiempo total para implementar cambios} / \text{N}^{\circ} \text{ de cambios implementados}$
Porcentaje de funcionalidades planificadas vs entregadas	Porcentaje de funcionalidades implementadas completas	Porcentaje de funcionalidades entregadas = $(\text{N}^{\circ} \text{ de funcionalidades entregadas} / \text{N}^{\circ} \text{ de funcionalidades planificadas}) * 100$
Tasa de defectos funcionales	Nº de defectos funcionales entregados	Tasa de defectos funcionales = $\text{N}^{\circ} \text{ de defectos funcionales} / \text{Tamaño total de funcionalidades}$
Índice de complejidad del proyecto	Complejidad del proyecto	Índice de complejidad = $\text{N}^{\circ} \text{ total de requerimientos} / \text{N}^{\circ} \text{ de iteraciones}$

Tabla A3-9 Proceso de Desarrollo e Impacto del Software [\[34\]](#)

ANEXO IV: MÉTRICAS CUALITATIVAS (COMPLEMENTARIAS)

Son elementales para evaluar aspectos como el talento, el conocimiento, la experiencia, la capacidad de resolución de problemas complejos, el pensamiento crítico¹⁵, la adaptabilidad y la comunicación efectiva.

Talento

Es la capacidad o habilidad de una persona para aprender o ejecutar una determinada tarea. Los equipos de desarrollo con talento son creativos, resolutivos ante problemas complejos y dados a la innovación.

Métricas	Alcance	Fórmula
Evaluación de desempeño	Promedio de las calificaciones recibidas en evaluaciones de desempeño (ej. calidad del código, creatividad y resolución de problemas).	$\text{Puntuación de desempeño} = \frac{\sum \text{Calificaciones de evaluación por pares}}{\text{Nº de evaluaciones}}$
Tiempo de resolución de problemas	Promedio en el que un desarrollador resuelve un problema	$\text{Promedio de tiempo de resolución} = \frac{\sum \text{Tiempo empleado para resolver problemas}}{\text{Nº de problemas resueltos}}$
Tasa de retención del talento	Porcentaje de empleados que permanecen en la empresa en un período de tiempo determinado	$\text{Tasa de retención} = \left(\frac{\text{Nº de empleados actuales}}{\text{Nº de empleados al inicio del periodo}} \right) \times 100$
Tiempo de productividad inicial	Tiempo que tarda en obtener el máximo nivel un nuevo empleado	$\text{Tiempo de productividad} = (\text{Días desde la incorporación hasta la máxima productividad alcanzada})$
Capacitación y mejora de habilidades	Progreso de habilidades después de la adecuación al entorno	$\text{Porcentaje de mejora} = \frac{(\text{Habilidad posterior a la capacitación} - \text{Habilidad previa a la capacitación})}{\text{Habilidad previa}} \times 100$
Índice de satisfacción laboral	Satisfacción laboral entre los empleados	$\text{Índice de satisfacción laboral} = \frac{\sum \text{Puntuaciones de satisfacción}}{\text{Nº de empleados}}$
Rotación voluntaria	Porcentaje de empleados que voluntariamente renuncian a la empresa	$\text{Tasa de rotación voluntaria} = \left(\frac{\text{Nº de empleados que}}{\text{Nº de empleados al inicio del periodo}} \right) \times 100$

¹⁵ Pensamiento crítico: proceso de analizar, evaluar y sintetizar información para formar juicios o tomar decisiones. Implica habilidades como la resolución de problemas, el razonamiento lógico y la creatividad. [\[78\]](#)

renuncian voluntariamente / N°
total de empleados) x 100

Tabla A4-1 Talento [35]

Experiencia

Se entiende como los conocimientos y habilidades que se han adquirido a lo largo del tiempo mediante éxitos y fracaso.

Métricas	Alcance	Fórmula
Años de experiencia	N° años que la persona ha trabajado como desarrollador	Años de experiencia = Fecha actual – Fecha de inicio en el campo
Complejidad de proyectos anteriores	Calificación del desarrollador del proyecto según complejidad	Índice de complejidad = $\sum$ Puntuación de complejidad de proyectos / N° proyectos
Número de tecnologías dominadas	Lenguajes de programación, frameworks o herramientas dominadas por los desarrolladores	N° de tecnologías dominadas = $\sum$ Tecnologías en las que el desarrollador tiene competencia comprobada
Tasa de éxito en los proyectos	Porcentaje de proyectos en los que el desarrollador ha cumplido plazo y objetivos del proyecto	Tasa de éxito en proyectos = (Proyectos exitosos / Total de proyectos) x 100
Aptitud para el trabajo en equipo	Encuestas sobre las capacidades del desarrollador para trabajar en equipo	Índice de aptitud para trabajo en equipo = $\sum$ Evaluaciones de equipo / N° de evaluaciones
Velocidad promedio de entrega	Tiempo medio en que un desarrollador entregue tareas dentro de un proyecto	Velocidad de entrega = $\sum$ Tiempo de entrega de tareas / N° de tareas

Tabla A4-2 Experiencia [35]

Conocimiento

Consiste en la comprensión teórica y práctica de un tema. Se basa en el estudio, observación e investigación.

Métricas	Alcance	Fórmula
Dominio de herramientas y tecnologías	Pruebas técnicas para evaluar el conocimiento en herramientas y tecnología	Puntuación de competencia = (Puntuación en pruebas técnicas / Puntuación máxima) x 100
Capacidad de documentación	Promedio de calificaciones en referencia a la documentación técnica (clara, exhaustiva y útil)	Puntuación de documentación n = $\sum$ Calificaciones de calidad de documentación / N° de evaluaciones

Actualización de conocimientos	Frecuencia con el que el desarrollador adquiere nuevas habilidades	Frecuencia de actualización = N° de cursos completados / N° de años
Certificaciones técnicas	N° de certificaciones obtenidas en el desarrollo	N° de certificaciones obtenidas = $\sum$ Certificaciones relacionadas con la tecnología
Capacidad de bugs complejos	Habilidad para identificar y solucionar problemas complejos de código	Puntuación de resolución de bugs = $\sum$ Bugs resueltos / N° total de bugs complejos
Dominio de patrones de diseño	Grado de conocimiento del diseño en el desarrollo de aplicaciones	Puntuación de dominio de patrones = $\sum$ Proyectos con patrones aplicados / N° total de proyectos
Conocimiento de arquitectura de software	Nivel de conocimientos de arquitectura de software	Puntuación de arquitectura = (Puntuación en evaluaciones de arquitectura / Puntuación máxima) $\times$ 100

Tabla A4-3 Conocimiento [35]

Capacidad de resolución de problemas complejos

Habilidad para analizar situaciones complicadas y encontrar la solución.

Métricas	Alcance	Fórmula
Tasa de resolución de problemas	Cuántos problemas se resuelve en un periodo de tiempo	Tasa de resolución = N° de problemas complejos resueltos / Total de problemas complejos planteados
Tiempo de resolución de problemas	Promedio que tarda el programador en resolver problemas complejos	Tiempo promedio = $\sum$ Tiempo para resolver problemas / N° de problemas resueltos
Número de problemas complejos resueltos	Número de problemas complejos resueltos en un período de tiempo	N° de problemas resueltos = $\sum$ Problemas complejos solucionados
Satisfacción del cliente con la resolución	Grado de satisfacción del cliente en la solución proporcionada	Satisfacción = $\sum$ Calificaciones de clientes / N° de soluciones entregadas
Frecuencia de problemas repetitivos	Problemas complejos que se repiten con frecuencia	Frecuencia de problemas repetitivos = N° de problemas recurrentes / N° total de problemas

Innovación de soluciones	Originalidad y creatividad en la resolución de problemas	$\text{Innovación} = \frac{\sum \text{Evaluaciones de creatividad de soluciones}}{\text{Nº de problemas evaluados}}$
--------------------------	--	--

Tabla A4-4 Resolución de Problemas Complejos [35]

Pensamiento crítico

Es el proceso de analizar, evaluar y sintetizar información para formar juicios o tomar decisiones.

Métricas	Alcance	Fórmula
Evaluación de toma de decisiones	Cantidad de decisiones basadas en datos y análisis	$\text{Evaluación} = \frac{\sum \text{Evaluaciones de decisiones}}{\text{Nº de decisiones tomadas}}$
Nº de alternativas evaluadas	Numero de alternativas consideradas antes de tomar una decisión	$\text{Nº de alternativas} = \frac{\sum \text{Alternativas evaluadas}}{\text{Nº de decisiones tomadas}}$
Cantidad de análisis crítico	Calidad de análisis crítico realizado en la toma de decisiones	$\text{Calidad} = \frac{\sum \text{Evaluaciones del análisis crítico}}{\text{Nº de análisis realizados}}$
Impacto de las decisiones	Evaluación del impacto de las decisiones en el proyecto	$\text{Impacto} = \frac{\sum \text{Resultados de decisiones}}{\text{Nº de decisiones evaluadas}}$
Revisión post-decisional	Frecuencia y calidad de la revisión de decisiones después de su implementación.	$\text{Revisión} = \frac{\sum \text{Evaluaciones post-decisionales}}{\text{Nº de decisiones implementadas}}$
Tasa de errores en decisiones	Porcentaje de errores cometidos en decisiones basadas en análisis crítico.	$\text{Tasa de errores} = \left(\frac{\text{Errores en decisiones}}{\text{Total de decisiones}} \right) \times 100$
Documentación de la toma de decisiones	Documentación donde se refleja el proceso de la toma de decisiones	$\text{Puntuación de documentación} = \frac{\sum \text{Calificaciones de documentación del proceso}}{\text{Nº de decisiones documentadas}}$

Tabla A4-5 Pensamiento Crítico [35]

Adaptabilidad

La adaptabilidad es la capacidad para amoldarse a las nuevas situaciones y entornos.

Métricas	Alcance	Fórmula
Velocidad de adaptación a nuevas tecnologías	Velocidad que un desarrollador se adapta a una nueva tecnología	$\text{Velocidad de adaptación} = \frac{\sum \text{Tiempo para adaptarse a nuevas tecnologías}}{\text{Nº de tecnologías nuevas}}$

Número de herramientas dominadas	Cantidad de herramientas nuevas dominadas en un período de tiempo	$\text{Nº de herramientas dominadas} = \frac{\sum \text{Herramientas nuevas dominadas}}{\text{Nº total de herramientas aprendidas}}$
Feedback de adaptación	Capacidad de adaptación por parte de colegas y supervisores	$\text{Feedback} = \frac{\sum \text{Evaluaciones de adaptación}}{\text{Nº de evaluaciones realizada}}$
Participación en proyectos diversos	Nº de proyectos con diferentes tecnologías o metodologías en los que se ha participado	$\text{Nº de proyectos diversos} = \frac{\sum \text{Proyectos con diferentes enfoques}}{\text{Nº total de proyectos}}$
Capacidad para aprender errores	Habilidad para aprender y adaptarse después de cometer errores	$\text{Aprendizaje de errores} = \frac{\sum \text{Errores aprendidos}}{\text{Nº de errores cometidos}}$
Resiliencia en cambios de proyectos	Capacidad para mantenerse efectivo durante cambios significativos en el proyecto.	$\text{Resiliencia} = \frac{\sum \text{Evaluaciones de resiliencia durante cambios}}{\text{Nº de cambios significativos}}$
Capacidad para manejar múltiples tareas	Habilidad para manejar múltiples tareas o proyectos a la vez	$\text{Capacidad de manejo} = \frac{\sum \text{Número de tareas o proyectos gestionados simultáneamente}}{\text{Nº de tareas o proyectos asumidos}}$

Tabla A4-6 Adaptabilidad [35]

Comunicación efectiva

Es la habilidad para escuchar, entender a los miembros del equipo, transmitir ideas, instrucción o sentimiento de manera clara y comprensible.

Métricas	Alcance	Fórmula
Claridad en la comunicación técnica	Evaluación de la claridad de la comunicación técnica en documentos y presentaciones.	$\text{Claridad} = \frac{\sum \text{Evaluaciones de claridad}}{\text{Nº de comunicaciones técnicas}}$
Efectividad en la comunicación entre equipos	Capacidad para comunicarse con otros equipos	$\text{Efectividad} = \frac{\sum \text{Evaluaciones interdepartamentales}}{\text{Nº de comunicaciones interdepartamentales}}$
Habilidad para presentar ideas Presentaciones ppt, ...	Evalúa la habilidad para presentar ideas	$\text{Habilidad de presentación} = \frac{\sum \text{Evaluaciones de presentación}}{\text{Nº de presentaciones realizadas}}$

Tabla A4-7 Comunicación Efectiva [35]