

# Universidad de las Ciencias Informáticas

ERP



**Título:** Componentes para la configuración de la gestión del multilinguaje y la gestión de trazas del sistema Cedrux.

Trabajo de Diploma para optar por el título de  
Ingeniero Informático.

**Autor:** Yormis Corrales Martínez.

**Tutor(es):** Ing. René Lazo Ochoa.  
Ing. Yoandry Morejón Borbón.

Ciudad de la Habana

Junio, 2009

“... lo que da al hombre el poder no es el mero conocimiento que viene del uso de los sentidos, sino, ese otro conocimiento más profundo que se llama Ciencia.”

José Martí

### AGRADECIMIENTOS

Antes que nada a la persona que ha sido el faro guía durante todos estos largos años,

Con su amor y dedicación mi madre Yolanda.

A mi padrastro que es como una madre Norellys, viejo este resultado también es tuyo.

A mi mujer que tanto me ha ayudado, a quien le dedico también este trabajo. Gracias por estar a mi lado todo este tiempo.

A mis tíos Claudio y Jorge, quienes han sido mi segundos padres, apoyo su cariño en todo momento.

A mi amigo y hermano Félix, que me ha ayudado en todos los sentidos de la vida.

A mi tutor René, gracias por enseñarme a ser independiente, y como se desarrolla verdaderamente junto a un equipo.

A mi tutor Borbón, gracias por más que un tutor ser un hermano y amigo.

A mis amigos y compañeros del proyecto, por el apoyo de cada uno de ustedes.

A mis familiares a los que nunca defraudaré jamás.

A todas las personas que me han brindado su apoyo y han creído en mi durante todos estos largos años, de corazón un millón de gracias.

### **DEDICATORIA**

Dedico este trabajo a la Universidad de Ciencias Informáticas, en la cual me he formado como ingeniero y por confiar una vez más en mí.

Al proyecto ERP-CUBA en el cual aprendí muchas cosas del punto de vista laboral así como personal.

A mis padres y tutores que sin ellos no fuese posible llegar al final de esta carrera.

A la revolución y en especial al Comandante en jefe, Fidel Castro Ruz.

### RESUMEN

El uso de los marcos de trabajo, o in inglés, framework, se ha convertido en una de las principales necesidades para el desarrollo de software en sus diversas variantes. Los mismos tienen utilidades principales para la base de desarrollo del proyecto o solución en que estén llevándose a cabo. Son utilizados según las características que destacan y necesidades del implicado.

Muchos de estos frameworks carecen de herramientas que les permitan funcionalidades que no pueden brindarles al proyecto donde están siendo empleados. Como es el caso del Sistema Integral de Gestión Cedrux, el cual se desarrollará con un marco de trabajo que permite gestionar el multilinguaje y la gestión de trazas, pero no cuenta con una herramienta para ello.

Para incrementar la productividad y disminuir los errores en el proceso de configuración de la gestión del multilinguaje o la internalización de variables en la fase de elaboración, y para garantizar la gestión de trazas en la fase de transición del Sistema Integral de Gestión de Entidades (Cedrux), se implementarán dos componentes que formarán parte de su IDE de desarrollo.

Por tal motivo el presente trabajo tiene como objetivo describir el análisis, diseño e implementación de dos componentes, uno para la configuración de la gestión del multilinguaje y otro para la gestión de trazas.

**PALABRAS CLAVES:** ERP, marco de trabajo, IDE, Json, trazas.

### ABSTRACT

The use of frameworks has become one of the key requirements for software development in its different variants. They are useful for the development of projects or solutions. They are used depending on the characteristics and requirements involved.

Many of these frameworks have no tools to provide functionality to the project where they are being used. In the case of the Integral Management System Cedrux, which will develop a framework that allows you to manage multilingual services and the allocation and management of traces, but it does not have a tool for this.

To increase productivity and reduce errors in the setup process of the multi-language or variable internalization in the development phase and to ensure the management of traces in the transition phase of the Integrated System Management Entity (Cedrux), two components will be implemented as part of their development IDE.

Therefore, this paper aims to describe the analysis, design and implementation of two components, one for the setup Multilanguage management and another one for the of trace management.

**KEY WORDS:** ERP, framework, IDE, Json, traces.

## ÍNDICE DE CONTENIDO

<b>INTRODUCCIÓN.....</b>	<b>1</b>
<b>CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN. ....</b>	<b>5</b>
INTRODUCCIÓN .....	5
ANÁLISIS DE LOS PRINCIPALES PROBLEMAS QUE AFECTAN LA PRODUCCIÓN EN LAS LÍNEAS DE TRABAJO DEL ERP, RELACIONADOS CON LA CONFIGURACIÓN DE LA GESTIÓN DEL MULTILENGUAJE Y LA GESTIÓN DE TRAZAS.....	5
ESTUDIO REFERENCIAL.....	9
ESTADO ACTUAL DE DISEÑO DE SOFTWARE. ....	9
JSON: CONFIGURACIÓN DEL MULTILENGUAJE.....	12
ESTADO ACTUAL DE LAS HERRAMIENTAS QUE GARANTIZAN LA CONFIGURACIÓN DE LA GESTIÓN DEL MULTILENGUAJE Y LA GESTIÓN DE TRAZAS EN EL MUNDO, EN LA UNIVERSIDAD Y EN EL SISTEMA INTEGRAL DE GESTIÓN CEDRUX. ....	14
VISTA TECNOLÓGICA DE LOS COMPONENTES DEL IDE PARA EL MARCO DE TRABAJO DE CEDRUX.....	16
<b>CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN.....</b>	<b>32</b>
REQUISITOS .....	32
REQUISITOS FUNCIONALES .....	32
REQUISITOS NO FUNCIONALES.....	34
PROTOTIPO DE INTERFAZ.....	35
DISEÑO DE CLASES .....	42
ARTEFACTOS DE IMPLEMENTACIÓN .....	42
ESQUEMAS DE LA CONFIGURACIÓN DE LOS FICHEROS .JSON.....	43
DIAGRAMA DE COMPONENTES.....	45
CASOS DE PRUEBAS.....	46
CONCLUSIONES .....	61
<b>CAPÍTULO 3: ANÁLISIS DE LA SOLUCIÓN .....</b>	<b>62</b>
INTRODUCCIÓN.....	62

VALORACIÓN DE LA SOLUCIÓN. ....	62
1.1 VALORACIONES APLICADAS A LA SOLUCIÓN PROPUESTA. ....	63
MÉTRICAS QUE EVALÚAN EL DISEÑO, ANÁLISIS DE LOS RESULTADOS. ....	63
2.1 MÉTRICAS DE SOFTWARE. ....	64
2.2 MÉTRICAS ORIENTADAS A OBJETOS. ....	64
CARACTERÍSTICAS DE LAS MÉTRICAS ORIENTADAS A OBJETOS. ....	65
2.3 MÉTRICAS ORIENTADAS A CLASES. ....	66
2.4 MÉTRICAS APLICADAS A LA SOLUCIÓN PROPUESTA. ....	67
2.4.1 RESULTADOS DEL INSTRUMENTO DE EVALUACIÓN DE LA MÉTRICA TAMAÑO DE CLASES (TC). ....	71
2.4.2 RESULTADO DEL INSTRUMENTO DE EVALUACIÓN DE LA MÉTRICA RESPUESTA PARA UNA CLASE (RPC). ....	74
2.4.3 RESULTADOS DEL INSTRUMENTO DE EVALUACIÓN DE LA MÉTRICA ÁRBOL DE PROFUNDIDAD DE HERENCIA (APH). ....	78
2.4.4 RESULTADOS DEL INSTRUMENTO DE EVALUACIÓN DE LA MÉTRICA NÚMERO DE DESCENDIENTES (NDD). ....	79
2.4.5 RESULTADOS DEL INSTRUMENTO DE EVALUACIÓN DE LA MÉTRICA NÚMERO DE OPERACIONES REDEFINIDAS PARA UNA SUB-CLASE (NOR). ....	82
MATRIZ DE CUBRIMIENTO O MATRIZ DE INFERENCIA DE INDICADORES DE CALIDAD. ....	84
CONCLUSIONES. ....	87
<b>CONCLUSIONES GENERALES. ....</b>	<b>88</b>
<b>RECOMENDACIONES. ....</b>	<b>89</b>
<b>BIBLIOGRAFÍA. ....</b>	<b>90</b>
<b>ANEXOS. ....</b>	<b>94</b>
ANEXO 2: REQUISITOS FUNCIONALES DEL COMPONENTE CONFIGURACIÓN DE LA GESTIÓN DEL MULTILENGUAJE. ....	97
ANEXO 3: REQUISITOS FUNCIONALES DEL COMPONENTE GESTIÓN DE TRAZAS. ....	111

---

ANEXO 4: DISEÑO DE CLASES DEL COMPONENTE CONFIGURACIÓN DE LA GESTIÓN DEL MULTILENGUAJE.....	121
ANEXO 5: DIAGRAMA DE CLASES DEL COMPONENTE GESTIÓN DE TRAZAS.....	122
ANEXO 6: DISEÑO DE CASOS DE PRUEBAS DEL COMPONENTE CONFIGURACIÓN DE LA GESTIÓN DEL MULTILENGUAJE. ....	123
ANEXO 7: DISEÑO DE CASOS DE PRUEBAS DEL COMPONENTE GESTIÓN DE TRAZAS.....	172
ANEXO 8: INSTRUMENTO DE MEDICIÓN DE LA MÉTRICA DE TAMAÑO DE CLASES (TC).....	203
ANEXO 10: INSTRUMENTO DE MEDICIÓN DE LA MÉTRICA ÁRBOL DE PROFUNDIDAD DE HERENCIA (APH).....	207
ANEXO 11: INSTRUMENTO DE MEDICIÓN DE LA MÉTRICA NÚMERO DE DESCENDIENTES (NDD).....	208

### ÍNDICE DE TABLAS.

TABLA 1: PERSONAL ENCUESTADO PARA CONOCER LOS CRITERIOS SOBRE LA GESTIÓN MANUAL DE LOS ARCHIVOS DE CONFIGURACIÓN DEL MULTILENGUAJE Y LA REALIZACIÓN MANUAL Y PERSONALMENTE DE LA GESTIÓN DE TRAZAS. ....	6
TABLA 2: RESULTADOS DE LA ENCUESTA PARA CONOCER LOS CRITERIOS SOBRE LA GESTIÓN MANUAL DE LOS ARCHIVOS DE CONFIGURACIÓN DE LA GESTIÓN DEL MULTILENGUAJE Y LA REALIZACIÓN MANUAL Y PERSONALMENTE DE LA GESTIÓN DE TRAZAS. ....	7
TABLA 4: PRINCIPALES ELEMENTOS DE COMPARACIÓN ENTRE LAS PLATAFORMAS. ....	21
TABLA 5: ELEMENTOS DE REUTILIZACIÓN. ....	29
TABLA 6: CONFIGURACIÓN DE LA GESTIÓN DEL MULTILENGUAJE, DCP_CARGAR_PROYECTO_EXISTENTE. ..	49
TABLA 7: CONFIGURACIÓN DE LA GESTIÓN DEL MULTILENGUAJE, DCP_CREAR_NUEVO_PROYECTO. ....	53
TABLA 8: CONFIGURACIÓN DE LA GESTIÓN DEL MULTILENGUAJE, DCP_GUARDAR_PROYECTO_COMO. ....	57
TABLA 9: CONFIGURACIÓN DE LA GESTIÓN DEL MULTILENGUAJE, DCP_GUARDAR_PROYECTO. ....	59
TABLA 10: TAMAÑO DE CLASES (TC).....	68
TABLA 11: RESPUESTA PARA UNA CLASE (RPC). ....	69
TABLA 12: ÁRBOL DE PROFUNDIDAD DE HERENCIA (APH).....	69
TABLA 13: NÚMERO DE DESCENDIENTES (NDD). ....	70
TABLA 14: NÚMERO DE OPERACIONES REDEFINIDAS PARA UNA SUB-CLASE (NOR). ....	70
TABLA 15: RESULTADOS EVALUADOS DE LA RELACIÓN ATRIBUTOS/MÉTRICAS POR CADA COMPONENTE QUE INTEGRAN LA SOLUCIÓN. ....	85
TABLA 16: RANGO DE VALORES PARA LA EVALUACIÓN TÉCNICA DE LOS ATRIBUTOS DE CALIDAD EVALUADOS POR CADA MÉTRICA. ....	85
TABLA 17: RF-1 CREAR NUEVO PROYECTO DEL ESCENARIO GESTIONAR PROYECTO.....	97
TABLA 18: RF-2 CARGAR UN PROYECTO EXISTENTE DEL ESCENARIO GESTIONAR PROYECTO.....	98
TABLA 19: RF-3 GUARDAR PROYECTO COMO DEL ESCENARIO GESTIONAR PROYECTO. ....	99
TABLA 20: RF-4 GUARDAR PROYECTO DEL ESCENARIO GESTIONAR PROYECTO. ....	100
TABLA 21: RF-1 CREAR NUEVO FICHERO .JSON DEL ESCENARIO GESTIONAR FICHERO.....	101
TABLA 22: RF-2 MOSTRAR VISTA PREVIA DEL FICHERO DEL ESCENARIO GESTIONAR FICHERO.....	102
TABLA 23: RF-3 GENERAR FICHERO .JSON DEL ESCENARIO GESTIONAR FICHERO. ....	103
TABLA 24: RF-4 COPIAR FICHERO .JSON A OTRO DIRECTORIO DEL ESCENARIO GESTIONAR FICHERO.....	104
TABLA 25: RF-4 ELIMINAR FICHERO .JSON DEL ESCENARIO GESTIONAR FICHERO. ....	105
TABLA 26: RF-1 ADICIONAR COMPONENTES AL FICHERO .JSON DEL ESCENARIO GESTIONAR COMPONENTE.....	106
TABLA 27: RF-2 EDITAR COMPONENTE AL FICHERO JSON DEL ESCENARIO GESTIONAR COMPONENTE.....	107

TABLA 28: RF-3 ORDENAR O DESORDENAR LOS COMPONENTES DEL FICHERO .JSON DEL ESCENARIO GESTIONAR COMPONENTE. ....	108
TABLA 29: RF-4 VISTA DE COMPONENTE (S) DEL FICHERO .JSON DEL ESCENARIO GESTIONAR COMPONENTE.....	109
TABLA 30: RF-5 ELIMINAR COMPONENTE (S) AL FICHERO .JSON DEL ESCENARIO GESTIONAR COMPONENTE.....	110
TABLA 31: RF-1 CREAR USUARIO DEL ESCENARIO GESTIONAR USUARIO(S) PARA EL SOPORTE.....	111
TABLA 32: RF-2 MODIFICAR USUARIO DEL ESCENARIO GESTIONAR USUARIO(S) PARA EL SOPORTE. ....	112
TABLA 33: RF-3 ELIMINAR USUARIO DEL ESCENARIO GESTIONAR USUARIO(S) PARA EL SOPORTE.....	113
TABLA 34: RF-4 EXPORTAR USUARIO (S) DEL ESCENARIO GESTIONAR USUARIO(S) PARA EL SOPORTE.....	114
TABLA 35: RF-5 ASIGNAR Y NOTIFICAR TRAZA (S) A USUARIO DEL ESCENARIO GESTIONAR USUARIO(S) PARA EL SOPORTE. ....	115
TABLA 36: RF-1 PARÁMETROS DE CONFIGURACIÓN DEL ESCENARIO PARÁMETROS DE CONFIGURACIÓN. ..	116
TABLA 37: RF-1 IMPORTAR TRAZAS DEL ESCENARIO GESTIONAR TRAZAS.....	117
TABLA 38: RF-2 EXPORTAR TRAZAS DEL ESCENARIO GESTIONAR TRAZAS.....	118
TABLA 39: RF-3 ASIGNAR USUARIO A LA (S) TRAZA (S) DEL ESCENARIO GESTIONAR TRAZAS. ....	119
TABLA 40: RF-4 MOSTRAR TRAZAS ASIGNADAS DEL ESCENARIO GESTIONAR TRAZAS. ....	120
TABLA 41: RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA TC Y SU INFLUENCIA EN LOS ATRIBUTOS DE CALIDAD (RESPONSABILIDAD, COMPLEJIDAD DE IMPLEMENTACIÓN Y REUTILIZACIÓN).....	203
TABLA 42: RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA RPC Y SU INFLUENCIA EN LOS ATRIBUTOS DE CALIDAD (ACOPLAMIENTO, COMPLEJIDAD DE MANTENIMIENTO, REUTILIZACIÓN Y CANTIDAD DE PRUEBAS).....	206
TABLA 43: RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA APH. ....	207
TABLA 44: RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA NDD Y SU INFLUENCIA EN LOS ATRIBUTOS DE CALIDAD (REUTILIZACIÓN, ABSTRACCIÓN DEL DISEÑO, NIVEL DE COHESIÓN Y CANTIDAD DE PRUEBAS).....	209
TABLA 45: RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA NOR Y SU INFLUENCIA EN LOS ATRIBUTOS DE CALIDAD (ABSTRACCIÓN DEL DISEÑO, CANTIDAD DE PRUEBAS, COMPLEJIDAD DE MANTENIMIENTO)..	211

### ÍNDICE DE FIGURAS.

FIGURA 1: ARQUITECTURA BASE DEL FRAMEWORK .NET.....	23
FIGURA 2: COMPILADOR DE LA PLATAFORMA .NET.....	23
FIGURA 3: BIBLIOTECA DE CLASES .NET.....	26
FIGURA 4: ESTRUCTURA DEL COMPILADOR DE LA PLATAFORMA .NET.....	27
FIGURA 5: INTERFAZ CORRESPONDIENTE AL RF-1.....	35
FIGURA 6: INTERFAZ CORRESPONDIENTE AL RF-2.....	36
FIGURA 7: INTERFAZ CORRESPONDIENTE A LOS RF-3 Y RF-4.....	36
FIGURA 8: INTERFAZ CORRESPONDIENTE AL RF-1.....	37
FIGURA 9: INTERFAZ CORRESPONDIENTE AL RF-2.....	37
FIGURA 10: INTERFAZ CORRESPONDIENTE A LOS RF-3, RF-4 Y RF-5.....	38
FIGURA 11: INTERFAZ CORRESPONDIENTE A LOS RF-1, RF-2, RF-4 Y RF-5.....	38
FIGURA 12: INTERFAZ CORRESPONDIENTE AL RF-3.....	39
FIGURA 13: INTERFAZ CORRESPONDIENTE AL RF-1.....	39
FIGURA 14: INTERFAZ CORRESPONDIENTE AL RF-2, RF-3, RF-4 Y RF-5.....	40
FIGURA 15: INTERFAZ CORRESPONDIENTE AL RF-1, RF-2 Y RF-3.....	40
FIGURA 16: INTERFAZ CORRESPONDIENTE AL RF-4.....	41
FIGURA 17: INTERFAZ CORRESPONDIENTE AL RF-1.....	41
FIGURA 18: DISEÑO DE CLASES DE LA INTEGRACIÓN DEL COMPONENTE GENERAL.....	42
FIGURA 19: ESQUEMA DEL XML DE FICHEROS .JSON.....	43
FIGURA 20: ESQUEMA DEL XML DE TRAZAS DE RENDIMIENTO.....	43
FIGURA 21: ESQUEMA DEL XML DE TRAZAS DE EXCEPCIÓN-INTEGRACIÓN.....	44
FIGURA 22: DIAGRAMA DE COMPONENTES DEL SISTEMA.....	45
FIGURA 23: REPRESENTACIÓN DE LOS RESULTADOS OBTENIDOS EN EL INSTRUMENTO AGRUPADOS EN LOS INTERVALOS DEFINIDOS.....	71
FIGURA 24: REPRESENTACIÓN EN % DE LOS RESULTADOS OBTENIDOS EN EL INSTRUMENTO AGRUPADOS EN LOS INTERVALOS DEFINIDOS.....	72

FIGURA 25: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA TC EN EL ATRIBUTO RESPONSABILIDAD. ....	72
FIGURA 26: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA TC EN EL ATRIBUTO COMPLEJIDAD DE IMPLEMENTACIÓN. ....	73
FIGURA 27: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA TC EN EL ATRIBUTO REUTILIZACIÓN. ....	73
FIGURA 28: GRÁFICA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA RPC AGRUPADOS POR LA TENDENCIA DE LOS VALORES. ....	74
FIGURA 29: REPRESENTACIÓN EN % DE LOS RESULTADOS OBTENIDOS EN EL INSTRUMENTO AGRUPADOS EN LOS INTERVALOS DEFINIDOS. ....	75
FIGURA 30: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA RPC EN EL ATRIBUTO ACOPLAMIENTO. ....	75
FIGURA 31: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA RPC EN EL ATRIBUTO COMPLEJIDAD DE MANTENIMIENTO. ....	76
FIGURA 32: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA RPC EN EL ATRIBUTO CANTIDAD DE PRUEBAS. ....	76
FIGURA 33: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA RPC EN EL ATRIBUTO REUTILIZACIÓN. ....	77
FIGURA 34: REPRESENTACIÓN EN % DE LOS RESULTADOS OBTENIDOS EN EL INSTRUMENTO AGRUPADOS POR NIVEL. ....	78
FIGURA 35: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA NND EN EL ATRIBUTO REUTILIZACIÓN. ....	79
FIGURA 36: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA NDD EN EL ATRIBUTO ABSTRACCIÓN DE LA CLASE BASE. ....	80
FIGURA 37: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA NDD EN EL ATRIBUTO COHESIÓN DE LA JERARQUÍA DE CLASES. ....	80
FIGURA 38: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA NDD EN EL ATRIBUTO CANTIDAD DE PRUEBAS. ....	81
FIGURA 39: REPRESENTACIÓN EN % DE LOS RESULTADOS OBTENIDOS EN EL INSTRUMENTO AGRUPADOS EN LOS VALORES EXISTENTES. ....	82
FIGURA 40: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA NOR EN EL ATRIBUTO COMPLEJIDAD DEL MANTENIMIENTO. ....	82

FIGURA 41: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA NOR EN EL ATRIBUTO CANTIDAD DE PRUEBAS.....	83
FIGURA 42: REPRESENTACIÓN DE LA INCIDENCIA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA NOR EN EL ATRIBUTO VIOLACIÓN DE LA ABSTRACCIÓN REPRESENTADA POR LA SUPERCLASE. ....	83
FIGURA 47: GRÁFICA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA TC Y SU INFLUENCIA EN LOS ATRIBUTOS DE CALIDAD (RESPONSABILIDAD, COMPLEJIDAD DE IMPLEMENTACIÓN Y REUTILIZACIÓN). ....	204
FIGURA 49: GRÁFICA DE LOS RESULTADOS DE LA EVALUACIÓN DE LA MÉTRICA APH. ....	208
FIGURA 51: REPRESENTACIÓN DE LOS RESULTADOS OBTENIDOS AL APLICAR LOS INSTRUMENTOS QUE EVALÚAN LA MÉTRICA NDD. ....	210
FIGURA 53: REPRESENTACIÓN DE LOS RESULTADOS OBTENIDOS AL APLICAR LOS INSTRUMENTOS QUE EVALÚAN LA MÉTRICA NOR. ....	211

### INTRODUCCIÓN

Actualmente podemos observar como se está produciendo un crecimiento acelerado a nivel empresarial. Esto hace cada vez más complicado poder llevar un control más exacto de los recursos que forman una empresa, que van desde los recursos humanos y financieros, hasta tecnológicos y materiales. Ante este problema en crecimiento, la tecnología de la información nos ofrece una posible solución mediante la implantación de un ERP.

La Planeación de Recursos Empresariales (Enterprise Resource Planning, ERP) es un sistema compuesto por un conjunto de módulos funcionales estándar y que son susceptibles de ser adaptados a las necesidades de cada empresa. (1)

Para llevar a cabo un proyecto como este se necesita un marco de trabajo robusto, configurable, fácil de usar y que cumpla con las necesidades del sistema Cedrux.

Un **marco de trabajo** o, en inglés, framework, es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, un framework puede incluir soporte de programas, bibliotecas y un lenguaje de scripting entre otros softwares para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Un framework representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio. (2)

Para facilitar al desarrollador el trabajo, se necesita de un **entorno de desarrollo integrado** capaz de agilizar el proceso de implementación del sistema.

(Integrated Development Environment) Entorno de desarrollo integrado. Conjunto de programas que se ejecuta a partir de una única interfaz de usuario. Por ejemplo, los lenguajes de programación incluyen a menudo un editor de texto, un compilador y un depurador, los cuales pueden activarse y funcionar a partir de un menú común.

(3)

El marco de trabajo utilizado para el desarrollo del sistema Cedrux permite configurar manualmente el multilinguaje a través de ficheros, lo cual trae como consecuencia un incremento de los errores, además de que el proceso de configuración se hace más lento y engorroso, afectando la productividad y el cumplimiento del cronograma de desarrollo e implantación del sistema; además permite llevar un registro de las trazas del sistema, dando la posibilidad de detectar los problemas generados por el sistema en la fase de transición, permitiendo el monitoreo de los mismos y su gestión posterior; la notificación y asignación de la gestión de trazas se realiza de personalmente, lo que trae consigo que en ocasiones este proceso se torne muy lento y por tanto el mantenimiento del sistema se haga más complejo.

Debido a los problemas anteriores este trabajo se enfocará en la construcción de dos componentes del IDE de desarrollo del marco de trabajo del sistema Cedrux, uno para la configuración de la gestión del multilinguaje y otro para la gestión de trazas.

Del estudio de la situación anterior se decidió darle solución al siguiente problema: ¿Cómo incrementar la productividad y disminuir los errores en el proceso de configuración de la gestión de multilinguaje en la fase de elaboración y disminuir el tiempo de ejecución de la gestión de trazas en la fase de transición?

Trazando como objeto de estudio: Componentes de gestión de multilinguaje y de gestión de trazas en sistemas empresariales.

, y como campo de acción: Informatización de la configuración de componentes de gestión de multilinguaje y gestión de trazas en el sistema empresariales.

Como objetivo general de este trabajo se define: Construir los componentes para la configuración de la gestión del multilinguaje y la gestión de trazas del sistema empresarial Cedrux.

Para dar cumplimiento al objetivo antes planteado se proponen los siguientes objetivos específicos:

- Realizar marco de referencia del estado actual de las herramientas que gestionan la configuración del multilinguaje y gestión de trazas.
- Realizar el análisis de los componentes de configuración de la gestión del multilinguaje y de la gestión de trazas.
- Realizar el diseño de los componentes de configuración de la gestión del multilinguaje y de la gestión de trazas.
- Realizar la implementación y prueba de los componentes de configuración de la gestión del multilinguaje y de la gestión de trazas.
- Realizar los casos de uso de prueba de los componentes de configuración de la gestión del multilinguaje y de la gestión de trazas.

Para validar el presente trabajo se defenderá la siguiente idea: El desarrollo para el sistema Cedrux de un componente para la configuración de la gestión del multilinguaje incrementará la productividad y disminuirá los errores en dicho proceso y de un componente para la gestión de trazas disminuirá el tiempo de su ejecución.

Para realizar una descripción detallada, este documento muestra el contenido de la investigación de la siguiente forma, el trabajo consta de 3 capítulos:

**Capítulo #1:** Se hace énfasis en la fundamentación teórica y marco contextual de la investigación, el estado referencial, estado de reutilización y resultados esperados u objetivos de la solución propuesta.

**Capítulo #2:** Se realiza la captura de requisitos, el diseño del prototipo de Interfaz, el diseño de clases, así como los artefactos generados durante la implementación y los casos de prueba que validan el correcto desarrollo de estos artefactos.

**Capítulo#3:** Se hace una valoración de la solución, matriz de cubrimiento o matriz de inferencia de indicadores de calidad, métricas que evalúan el diseño, análisis de los resultados y finalmente las pruebas de concepto.

### **CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.**

#### **Introducción**

Durante la fase de elaboración del sistema CEDRUX, se lleva a cabo la configuración de la gestión del multilinguaje o internalización de variables. En dicho proceso se originan errores en el momento de configuración de los ficheros json; que no son más que ficheros con un formato de texto completamente independiente para el intercambio de datos. Dicha configuración se realiza manualmente originando en ocasiones que no cumpla con el estándar definido. De la problemática antes mencionada se obtiene como resultado una disminución de la productividad y retrasos en el cronograma del desarrollo y a su vez en el de implantación.

Por otra parte la gestión de trazas se realizan muy lentamente y de forma personal. Por ejemplo, en el momento que se dispara una excepción de integración, para gestionar la solución de las trazas se debe verificar qué subsistema originó dicha traza y luego asignar personalmente al desarrollador del subsistema en cuestión. Como se desarrolla el proceso antes descrito se originan retrasos y por lo tanto demora en el tiempo de solución.

#### **Análisis de los principales problemas que afectan la producción en las líneas de trabajo del ERP, relacionados con la configuración de la gestión del multilinguaje y la gestión de trazas.**

El sistema de Integral de Gestión Empresarial Cedrux se encuentra en pleno desarrollo por lo que carece de una herramienta que garantice la configuración del marco de trabajo del mismo, por tanto, los ficheros de datos encargados de gestionar el multilinguaje son tratados de forma manual e insegura y la gestión de trazas se realiza muy lentamente y de forma personal.

Como parte de la presente investigación, se aplicó una encuesta, (**Anexo1: Plantilla de encuesta realizada a los distintos roles que integran el proyecto**

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

---

**ERP.**), a un grupo de 75 personas del equipo de trabajo del software relacionado (muestra representativa), con el fin de conocer las características, efectos y consecuencias de la gestión manual de los archivos de configuración del multilenguaje, así como, la realización lenta y personalmente la gestión de trazas.

Personal encuestado	No.	%
Jefe de Línea	5	6.66
Analista	8	10.66
Planificador	4	5.33
Arquitecto	11	14.66
Desarrollador	47	62.66
Total	75	100

**Tabla 1: Personal encuestado para conocer los criterios sobre la gestión manual de los archivos de configuración del multilenguaje y la realización manual y personalmente de la gestión de trazas.**

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

<b>Resultados de la encuesta:</b>		
	No.	%
<b>1. Insatisfechos con la configuración manual del multilinguaje.</b>	<b>70</b>	<b>93.33</b>
<b>2. Insatisfechos con la realización lenta y personalmente de la gestión de trazas.</b>	<b>68</b>	<b>90.66</b>
<b>3. Principales dificultades en la configuración de la gestión del multilinguaje:</b>		
<b>Pérdida de tiempo</b>	<b>50</b>	<b>66.66</b>
<b>Introducción de errores.</b>	<b>64</b>	<b>85.33</b>
<b>Incremento del tiempo de máquina.</b>	<b>36</b>	<b>48.00</b>
<b>Baja productividad del equipo de trabajo.</b>	<b>61</b>	<b>81.33</b>
<b>Atraso con el cronograma de entrega.</b>	<b>58</b>	<b>81.69</b>
<b>3. Principales dificultades en la realización de la gestión de trazas. :</b>		
<b>Atraso con el cronograma de solución de las trazas.</b>	<b>65</b>	<b>86.66</b>
<b>Insatisfacción en la realización del proceso.</b>	<b>71</b>	<b>94.66</b>
<b>4. Consideran necesario desarrollar el componente de configuración de la gestión del multilinguaje.</b>	<b>70</b>	<b>93.33</b>
<b>5. Consideran necesario desarrollar el componente de gestión de trazas.</b>	<b>68</b>	<b>90.66</b>

*Tabla 2: Resultados de la encuesta para conocer los criterios sobre la gestión manual de los archivos de configuración de la gestión del multilinguaje y la realización manual y personalmente de la gestión de trazas.*

## **CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.**

---

El análisis de los resultados obtenidos por la encuesta aplicada al personal implicado en el desarrollo del Sistema Integral de Gestión Empresarial Cedrux, refleja el impacto negativo, efecto de la configuración manual de los archivos de la gestión de multilinguaje de Cedrux, de donde se puede concluir que esta práctica causa pérdida de tiempo, que se traduce en retrasos innecesarios en el cumplimiento las tareas y metas programadas para la liberación de componentes y funcionalidades importantes, disminuyendo la eficiencia del trabajo en el sistema, incrementa la introducción de errores en los ficheros de configuración, y genera atrasos en el cronograma de entrega. Por otra parte la realización manual y personalmente de la gestión de trazas trae consigo atraso con el cronograma de solución de las trazas e insatisfacción en la realización del proceso.

En general se llegó a la conclusión la necesidad de desarrollar un componente para la configuración de la gestión del multilinguaje con una aceptación del 93.33% y otro para la gestión de trazas con un de 90.66%.

### **Estudio Referencial**

#### **Estado actual de diseño de software.**

Actualmente el concepto de diseño es orientado a diferentes aristas como la que plantea Grady Bosch” el diseño es el proceso de determinar una implementación efectiva y eficiente que realice las funciones y tenga la información del análisis de dominio”, o la planteada por R.S.Pressman, “el diseño del software se encuentra en el núcleo técnico de la ingeniería de software y se aplica independientemente del modelo de diseño de software que se utilice. Una vez que se analizan y especifican los requisitos del software, el diseño del software es la primera de las tres actividades técnicas-diseño, generación de código y pruebas- que se requieren para construir y verificar el software. Cada actividad transforma la información de manera que dé lugar por último a un software de computadora validado” (PRESSMAN 1998).”El diseño es el lugar en donde se fomentan la calidad en la ingeniería de software. El diseño es la única forma de convertir exactamente los requisitos de un cliente en un producto o sistema de software finalizado. El diseño de software sirve como fundamento para todos los pasos siguientes del soporte del software y de la ingeniería del software. Sin su diseño, corremos el riesgo de construir un sistema inestable, un sistema cuya realidad no puede evaluarse hasta muy avanzado el proceso, sin tiempo suficiente y con mucho dinero gastado” (PRESSMAN 1998). (4).

En el caso de las metodologías, como por ejemplo RUP, los criterios sobre el diseño de sistemas plantean que es el flujo de trabajo donde el arquitecto y los diseñadores de sistema, describen de una forma más concreta y acorde al tipo de tecnología, el cómo serán implementadas las funcionalidades del sistema en construcción, basándose en los tipos de colaboración, la relación que existe entre las clases, y las responsabilidades de cada unidad de código (5). Otra metodología muy utilizada es XP (Extreme Programming), la misma describe el diseño con un enfoque práctico y dinámico para el cual el diseño de sistema es una actividad que construye un proceso de diseño evolutivo que se basa en re facturar un sistema simple en cada

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

---

iteración. Todo el diseño se centra en la iteración actual y no se hace nada anticipadamente para necesidades futuras.

Sin duda alguna uno de los principios de la programación extrema es la simplicidad, de aquí que el diseño no escape a esta filosofía. El diseño debe ser lo más simple posible. El paradigma KISS (“Keep It Small and Simple” para unos o Keep It Simple, Stupid para otros) se lleva hasta las últimas consecuencias; por ejemplo, se hace énfasis en no añadir funcionalidad al diseño del sistema nunca antes de lo necesario, por las sencillas razones de que probablemente ahora mismo no sea lo más prioritario o porque quizás nunca llegue a ser necesaria (FOWLER 2003).

Algunas empresas, como DEADELUS, plantean que el diseño de sistemas se ocupa de desarrollar las directrices propuestas durante el análisis en términos de aquella configuración que tenga más posibilidades de satisfacer los objetivos planteados tanto desde el punto de vista funcional como del no funcional. El proceso de diseño de un sistema complejo se suele realizar de forma descendente: Diseño de alto nivel(o descomposición del sistema a diseñar en subsistemas menos complejos), diseño e implementación de cada uno de los subsistemas, especificación consistente y completa del subsistema de acuerdo con los objetivos establecidos en el análisis, desarrollo según la especificación, prueba, integración de todos los subsistemas, (6) validación del diseño.

Microsoft por su parte plantea en su modelo Microsoft Solution Framework (MSF) para el desarrollo de sistemas de software, que el diseño es una actividad que debe ser realizada con el fin de establecer el comportamiento dinámico del sistema, es decir, cómo debe reaccionar ante los acontecimientos. Para su desarrollo también se utilizan los lenguajes de modelado. El resultado obtenido en la etapa de diseño facilita enormemente la implementación posterior de los sistemas, pues proporciona la estructura básica del sistema y cómo los diferentes componentes actúan y se relacionan entre ellos. (7).

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

---

Partiendo del estado del arte de los conceptos que hoy se manejan en el mundo sobre la importancia y la responsabilidad del “Diseño del Sistema”, planteados tanto por personalidades de la teoría del desarrollo de software, como por metodologías de desarrollo de software, o algunas empresas de desarrollo de software y teniendo en cuenta las características de la solución propuesta, se conceptualiza la actividad de diseño como un proceso y un modelado a la vez, de mucha importancia en la vida de un proyecto de software. Debido a que el proceso de diseño es además un conjunto de pasos repetitivos que permiten al diseñador describir todos los aspectos del sistema a construir, partiendo de las especificaciones que llegan del flujo de análisis; es el flujo que describe de una forma más concreta y teniendo en cuenta la tecnología y el lenguaje de programación que se utiliza, la representación estructural del sistema que se desea construir. A lo largo del diseño se evalúa la calidad del desarrollo del proyecto apoyándonos en un conjunto de revisiones técnicas. El diseño debe implementar todos los requisitos explícitos contenidos en el modelo de análisis y debe acumular todos los requisitos implícitos que desea el cliente. (8). Debe ser una guía que puedan leer y entender los que construyan el código y los que prueban y mantienen el software. El diseño debe proporcionar una completa idea de lo que es el Software, enfocando los dominios de datos, funcional y comportamiento desde el punto de vista de la implementación. En el diseño se deben concretar las representaciones funcionales de interacción entre los siguientes aspectos, Cliente-Sistema, Sistema-Dispositivos, Sistema-Soportes de almacenamiento, y es una base para garantizar un producto acorde con las necesidades del cliente. (9)

### **JSON: Configuración del multilinguaje.**

La **Internacionalización de variables o multilinguaje** es un proceso a través del cual se diseñan productos de software para que puedan adaptarse a diferentes idiomas y regiones sin necesidad de cambios de ingeniería ni cambios en el código. (10)

Json es un acrónimo de "JavaScript Object Notation", es un formato ligero para el intercambio de datos. Es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.

La simplicidad de JSON ha dado lugar a la generalización de su uso. Una de las supuestas ventajas de JSON sobre XML como formato de intercambio de datos en este contexto es que es mucho más sencillo escribir un analizador semántico de JSON.

Por esa razón, JSON se emplea habitualmente en entornos donde el tamaño del flujo de datos entre cliente y servidor es de vital importancia (de aquí su uso por Yahoo, Google, que atienden a millones de usuarios) cuando la fuente de datos es explícitamente de fiar y donde no es importante el no disponer de procesamiento XSLT para manipular los datos en el cliente. (11)

### ¿Por qué usar JSON si el estándar habitual es XML?

JSON tiene algunas ventajas sobre XML que debemos conocer:

La primera es que los datos en JSON ocupan mucho menos que XML, este último añade bastante sobrecarga a los datos que queremos serializar.

Ejemplo de los datos de una persona en XML:

```
<persona>
<nombre>Rodrigo</nombre>
<apellido>Corral</apellido>
<genero>varón</genero>
<edad>29</edad>
</persona>
```

Y estos son los mismos datos en JSON:

```
{
  "nombre" : "Rodrigo",
  "apellido" : "Corral",
  "genero" : "varón",
  "edad" : 29
}
```

Por las ventajas que ofrece este formato ligero para el intercambio de datos, se escogió para llevar a cabo el proceso de internacionalización o multilinguaje en el Sistema Integral de Gestión Cedrux.

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

---

**Estado actual de las herramientas que garantizan la configuración de la gestión del multilinguaje y la gestión de trazas en el mundo, en la universidad y en el Sistema Integral de Gestión Cedrux.**

Actualmente son muchos los framework en PHP que se utilizan para el desarrollo de aplicaciones web. Se estudiaron 5 framework por su impacto y internacionalización de la tecnología PHP en el mundo, de ellos contienen en su arquitectura componentes especializados con niveles considerables de madures para al gestión de estos aspectos arquitectónicos el Zend framework con los componentes ZendLog para la gestión de trazas y ZendTranslator, para la gestión del multilinguaje, la peculiaridad de tratamiento de estas soluciones son basas en ficheros log de texto plano, en uso de xml o en uso de bases de datos, pero este no presenta una aplicativo de interfaz amigable, que permita gestionar y configurar los mismos, en el caso de estos representantes lo usuarios deben hacer uso de las configuraciones modificando personalmente los archivos de configuración en dependencia del adaptador que se desee utilizar. A continuación se especifican:

<b>Adaptador</b>	<b>Archivos de configuración</b>
<b>Cvs</b>	<i>(traducción.csv / traducción.txt)</i>
<b>Gettext</b>	<i>(traducción.mo)</i>
<b>Ini</b>	<i>(traducción.ini)</i>
<b>TBX</b>	<i>(traducción.TBX / traducción.xml)</i>
<b>Tmx</b>	<i>(traducción.tmx / traducción.xml)</i>
<b>Qt</b>	<i>(traducción.ts)</i>
<b>XLIFF</b>	<i>(traducción.XLIFF / traducción.xml)</i>
<b>XmITm</b>	<i>(traducción.xml)</i>

## **CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.**

---

En la Universidad los frameworks anteriores los más usados son el Symfony y Zend. En la actualidad no existen aun herramientas que garanticen estos dos procesos en estos frameworks a través de una herramienta o IDE de desarrollo.

En el Sistema Integral de Gestión Cedrux, el framework utilizado por decisión del proyecto es Zend framework. Debido a la importancia que posee estos dos procesos y la operabilidad que brindan al sistema se hace necesaria la construcción de dos componentes que garanticen la internacionalización de variables o multilenguaje y la gestión de trazas de Cedrux.

### **Vista tecnológica de los componentes del IDE para el marco de trabajo de Cedrux.**

#### **Selección de la plataforma:**

En esta sección se realiza la valoración y evaluación de las plataformas tecnológicas y la selección de la plataforma más óptima teniendo en cuenta las características propias del proyecto, los recursos materiales y financieros y el nivel de conocimiento de los recursos humanos disponibles en esta plataforma.

Debido a las características de la enseñanza y la producción en la universidad, las aplicaciones de escritorio construidas son generadas a partir de dos plataformas principales:

- J2EE
- .net 2.0

Por tanto, debido al estrecho marco de tiempo que caracteriza la entrega del proyecto y a la preparación de los recursos humanos, serán tomadas estas dos plataformas en cuenta para el desarrollo del IDE.



#### **Plataforma J2EE:**

Una plataforma es el entorno de hardware o software en el cual se ejecuta un programa.

A diferencia de las plataformas tradicionales como Linux, Windows y Solaris, la plataforma Java está basada solo en software que se ejecuta sobre otras plataformas basadas en hardware. Por esto la plataforma Java y su software pueden ser utilizados sobre variados sistemas operativos y hardware. La plataforma Java está constituida de tres componentes: el lenguaje, la máquina virtual y las bibliotecas. (12)

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

---

*Java 2 Platform, Enterprise Edition* es una solución dada por *Sun Microsystems* para la creación de servidores de aplicación basado en el lenguaje Java. J2EE es una especificación, un concepto, que define una arquitectura para el desarrollo de aplicaciones basada en componentes que los servidores de aplicación J2EE-compatibles deben cumplir.

J2EE es una buena solución para el desarrollo de aplicaciones empresariales porque permite reducir el tiempo de desarrollo y mejorar la calidad del producto, sin embargo, J2EE no es simple porque consta de variados componentes y unifica una gran cantidad de conceptos. Entre las nuevas componentes destacan servlets y JSP para generar contenidos dinámicos, y EJB por sus posibilidades de encapsular el acceso a una fuente de datos permitiendo el uso objetos de persistentes en el tiempo, centralizar la lógica del negocio y permitir utilizar programación distribuida. Sin embargo, J2EE y sus componentes son relativamente nuevos, con tres años desde la publicación de su primera versión, por lo que no son muy conocidas sus cualidades y objetivos lo que hace necesario un estudio profundo de este. (13)



### **Plataforma .NET 2.0**

.NET es un proyecto de Microsoft para crear una nueva plataforma de desarrollo de software con énfasis en transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones. Basado en ella, la empresa intenta desarrollar una estrategia horizontal que integre todos sus productos, desde el sistema operativo hasta las herramientas de mercado.

.NET podría considerarse una respuesta de Microsoft al creciente mercado de los negocios en entornos Web, como competencia a la plataforma Java de Sun Microsystems y a los diversos framework de desarrollo web basados en PHP. Su propuesta es ofrecer una manera rápida y económica, a la vez que segura y robusta, de desarrollar aplicaciones o como la misma plataforma las denomina, "soluciones" permitiendo una integración más rápida y ágil entre empresas y un acceso más simple y universal a todo tipo de información desde cualquier tipo de dispositivo.

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

---

A fines del 2005 Microsoft lanzó la nueva versión de su plataforma de desarrollo .NET 2.0. Dentro de este paquete de actualizaciones, existen cambios para todo el sistema de desarrollo comprendiendo entorno de desarrollo, aplicaciones de base de datos, lenguajes, frameworks y otras novedades del sistema.

### ***Nuevos entornos de desarrollo:***

- Visual Studio 2005.
- Visual Web Developer 2005 Express

### ***Nuevas Base de Datos:***

- SQL Server 2005
- SQL Server 2005 Express

### ***Nuevos Lenguajes:***

- Visual Basic .NET 7.0
- C# 2.0

### ***Nuevos FrameWorks:***

- .NET Framework 2.0
- .NET Compact Framework 2.0

Al igual que la anterior plataforma, la versión 2.0 mantiene todas las prestaciones que la convirtieron en un producto de excelente calidad para el desarrollo de sistemas empresariales, orientados tanto a aplicaciones de escritorio como a servicios web.

Esta tecnología ofrece soluciones a los problemas de programación actuales, como son la administración de código o la programación para Internet. Para aprovechar al máximo las características de .NET es necesario entender la arquitectura básica en la que está implementada esta tecnología y así beneficiarse de todas las características que ofrece esta nueva plataforma.

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

---

### Selección de las características a evaluar en la selección de la plataforma tecnológica.

- Par 1: Conocimiento de los recursos humanos sobre las características y el manejo de las aplicaciones en la plataforma.
- Par 2: Seguridad de los distintos sistemas. Vista desde todos los niveles de aplicación, datos, lógica e interfaz de usuario.
- Par 3: Gestión de Datos, almacenamiento y servicios.
- Par 4: FrameWork de trabajo con una interfaz amigable, que presente una curva ligera de aprendizaje en vistas de la rapidez de las entregas del producto.
- Par 5: Accesibilidad (De código abierto o cerrado).
- Par 6: Popularidad y desarrollo.
- Par 7: Portabilidad (Sistema operativo con el cual se ejecuta).

### Calificación y cómputo de totales de cada plataforma.

Se lleva a cabo la calificación y cómputo de las plataformas mediante la realización de la tabla de cómputo de los parámetros en la que se asigna una calificación a las características que brinda cada plataforma, permitiendo realizar una comparación entre ellas. Los parámetros están valorados en una escala de valores del 1 al 10 de forma ascendente.

CARACTERÍSTICAS A EVALUAR	.NET	J2EE
Parámetro 1	10*	3
Parámetro 2	5	5
Parámetro 3	5	4
Parámetro 4	10*	4
Parámetro 5	5	4
Parámetro 6	5	4
Parámetro 7	3	5

\* Los elementos marcados poseen características que destacan el nivel por las necesidades del proyecto.

**Tabla 3: Parámetros de importancia para la investigación en la comparación entre plataformas.**

### **Comparación entre plataformas.**

Existen pocas alternativas que compitan directamente con J2EE y la tecnología .NET, debido a que muchos proveedores en vez de crear productos similares a estos, escogen implementar plataformas compatibles. (14)

Microsoft es el único proveedor que ofrece una plataforma para desarrollo de aplicaciones empresariales que compite directamente con J2EE. Su producto es conocido como .NET, anteriormente llamado DNA, que es un modelo de desarrollo para aplicaciones empresariales para la plataforma Windows. Existen analogías directas entre .NET y J2EE como ADO, ActiveX, ASP, COM+ y MTS, para JDBC, applets, JSP, EJB y JTA, respectivamente. Sin embargo, existen diferencias entre .NET y J2EE, siendo la diferencia principal la independencia del lenguaje de .NET v/s independencia del proveedor de J2EE. J2EE es una plataforma independiente de un proveedor, es una aplicación que cumple con los estándares de J2EE puede ser ejecutada en cualquier servidor de aplicaciones J2EE compatible, sin embargo, tiene una limitación, las aplicaciones J2EE deben ser implementadas en Java. (14)

Microsoft .NET es una plataforma independiente del lenguaje, así los desarrolladores pueden elegir el lenguaje y las herramientas a utilizar para implementar aplicaciones .NET.

Entre los lenguajes a elegir se encuentran Visual Basic, Visual C++, Visual J++ y Visual C#, esto se logra transformándolos a un lenguaje intermedio e independiente del lenguaje de programación (IL). Sin embargo, para ser ejecutado se debe utilizar la plataforma Windows y su software como IIS, DTC, OLE DB e Internet Explorer si se utilizan ActiveX.

J2EE es una plataforma más madura, se creó primero y su avance es muy rápido dado los aportes de muchas personas a través del mundo que apoyan el software Java. Por otra parte Microsoft .NET ofrece una solución integrada en la plataforma

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

Windows, donde Microsoft hace el esfuerzo en asegurar que sus productos pueden trabajar juntos sin mayores problemas de configuración.

Dada las distintas tecnologías para desarrollar aplicaciones, es que se han creado métodos para poder relacionarlas. Estas son conocidas como tecnologías de integración de aplicaciones, y están compuestas principalmente por CORBA y Servicios Web. CORBA permite la comunicación de aplicaciones independiente de la arquitectura, el sistema operativo, el lenguaje de programación y el tipo de red utilizada, si bien su ofrecimiento es bueno, no ha logrado posicionarse como la mejor solución existente ya sea por su complejidad o por las pocas implementaciones existentes. Los Servicios Web son un conjunto de tecnologías que utilizan XML para comunicarse e intercambiar información, usando un protocolo de comunicación estándar basado en XML conocido como SOAP. Los Servicios Web son muy recientes, pero han logrado un buen éxito inicial por comunicarse a través de HTTP.

Por tanto una integración de los servicios brindados por el proyecto IDE para el ERP cubano podría comunicarse con cualquier otra plataforma de configuración mediante servicios CORBA y simplemente SOAP.

	.NET	J2EE
Lenguaje de programación	C#...	Java
Plataforma	Windows	...
Máquina virtual	CLR [Common Language Runtime]	JVM [Java Virtual Machine] / JRE [Java Runtime Environment]
Lenguaje interpretado	MSIL	Java bytecodes
Cientes Windows	.NET Forms	Swing
Cientes Web	ASP.NET	JSP / Servlets
Servidores de aplicaciones	...	EJB [Enterprise Java Beans]
Acceso a bases de datos	ADO.NET	JDBC / EJB-SQL
Paso de mensajes	MSMQ	JMS [Java Messaging Service] / Msg EJBs
Integración con sistemas previos	COM TI	JCA [Java Connector Architecture]

**Tabla 4: Principales elementos de comparación entre las plataformas.**

### **Selección de la plataforma apropiada.**

Luego de analizar los resultados de la comparación en base a los parámetros definidos como sobresalientes, se determina como plataforma a utilizar, la plataforma .NET. En la comparación marcan pauta principalmente los parámetros 1 y 4 debido a las características del desarrollo del Sistema Integral de Gestión Cedrux. (15)

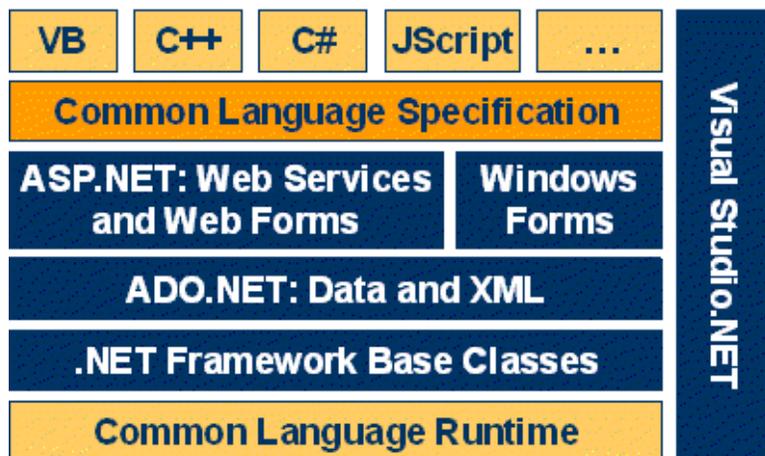
### **Selección del framework y las librerías**

En esta sección se realiza la valoración y evaluación de los frameworks y librerías para la plataforma tecnológica seleccionada y la selección de los frameworks y librerías más óptimas según las características propias del proyecto, los atributos de calidad y las categorías o aspectos arquitectónicamente significativos de la aplicación, los recursos materiales y financieros y el nivel de conocimiento de los recursos humanos disponibles. (16)

### **Framework .NET.**

El framework .NET es la propuesta de Microsoft para el desarrollo de aplicaciones completamente orientadas a objetos, seguras, sencillas de instalar y multiplataforma; no es un lenguaje de programación: es un conjunto de tecnologías de software que permite el desarrollo de aplicaciones de escritorio (Windows), Web(ASP.Net), bibliotecas de código (componentes reutilizables: DLL), dispositivos móviles, controles de usuario, acceso a datos, reportes y mucho más, que se ejecutan bajo el .NET Framework. Es completamente extensible: podemos heredar de sus clases base y modificar su comportamiento o implementar sus interfaces para desarrollar nuevas ideas totalmente originales. (17)

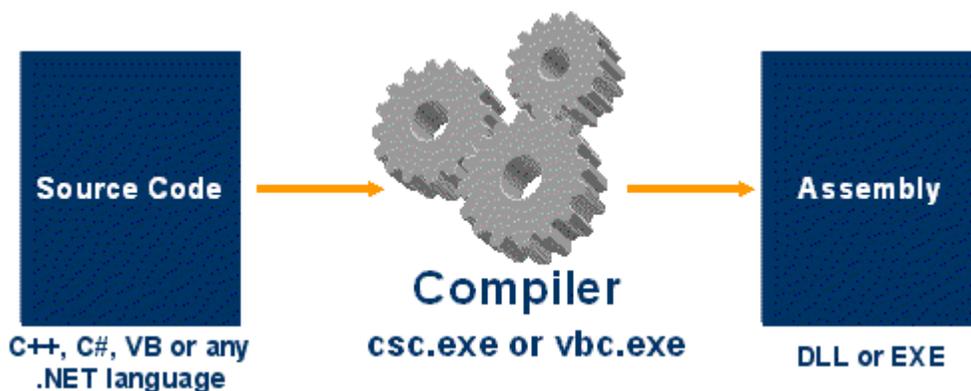
El entorno de programación de la plataforma .NET, *Visual Studio*, es uno de los más potentes en cuanto a facilidad de uso y proporciona una gran cantidad de herramientas para mejorar la productividad del programador. Es independiente del lenguaje de programación, por lo que se pueden realizar componentes en un lenguaje y utilizarlo en otro, sin complicaciones ni pérdida de rendimiento, debido a su sistema de tipos comunes y a su arquitectura de lenguaje intermedio (*MSIL*).



*Figura 1: Arquitectura base del framework .NET.*

### CLR

El **Common Language Runtime (CLR)** es el núcleo de la plataforma .NET ya que es el encargado de gestionar la ejecución de código compilado para la plataforma .NET. Puede asimilarse a la *máquina virtual de Java*. (18)



*Figura 2: Compilador de la plataforma .NET.*

Las dos principales características del CLR son:

- **Ejecución multiplataforma:** El CLR actúa como una máquina virtual, encargándose de ejecutar las aplicaciones diseñadas para la plataforma .NET. Su especificación está abierta, por lo que cualquier plataforma para la que exista una versión del CLR podrá ejecutar cualquier aplicación .NET.

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

---

- **Integración de lenguajes:** Desde cualquier lenguaje para el que exista un compilador que genere código para la plataforma .NET es posible utilizar código generado para la misma usando cualquier otro lenguaje tal y como si de código escrito usando el primero se tratase.

Otras características destacables son:

- **Modelo de programación consistente:** A todos los servicios y facilidades ofrecidos por el CLR se accede de la misma forma: a través de un modelo de programación orientado a objetos.
- **Gestión de memoria:** El CLR incluye un **recolector de basura** que evita que el programador tenga que tener en cuenta cuándo ha de destruir los objetos que dejen de serle útiles. Gracias a este recolector se evitan errores de programación muy comunes como intentos de borrado de objetos inexistentes, agotamiento de memoria por olvido de eliminación de elementos inutilizables o solicitud de acceso a miembros de objetos ya destruidos.
- **Seguridad de tipos:** El CLR facilita la detección de errores de programación difíciles de localizar comprobando que toda conversión de tipos que se realice durante la ejecución de una aplicación .NET se haga de modo que los tipos origen y destino sean compatibles.
- **Aislamiento de procesos:** El CLR asegura que desde el código perteneciente a un determinado proceso no se pueda acceder al código o datos pertenecientes a otro, ni se permite acceder a posiciones arbitrarias de memoria.
- **Tratamiento de excepciones:** En el CLR todos los errores que se puedan producir durante la ejecución de una aplicación se propagan de igual manera: mediante excepciones.
- **Soporte multihilo:** El CLR es capaz de trabajar con aplicaciones divididas en múltiples hilos de ejecución que pueden ir evolucionando por separado en paralelo o intercalándose, según el número de procesadores de la máquina sobre la que se ejecuten. Las aplicaciones pueden lanzar nuevos hilos, destruirlos, suspenderlos por un tiempo o hasta que les llegue una notificación, enviarles notificaciones, sincronizarlos, etc.

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

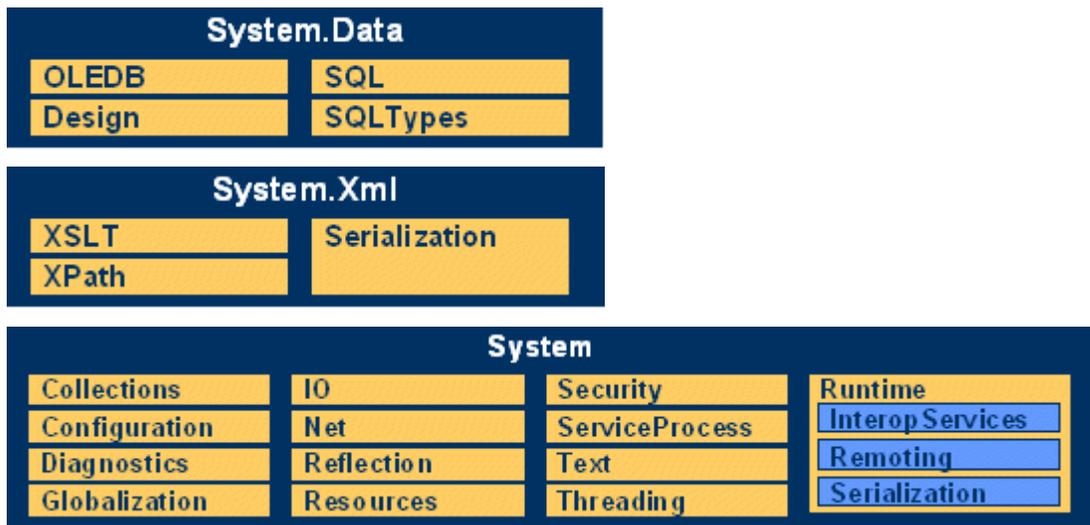
- **Seguridad avanzada:** El CLR proporciona mecanismos para restringir la ejecución de ciertos códigos o los permisos asignados a los mismos según su procedencia o el usuario que los ejecute. (19)

### Ensamblados

Constituyen la unidad lógica de despliegue en la plataforma .NET. Un assembly incluye metadatos acerca de los componentes incluidos en el assembly (versiones, tipos, dependencias...), metadatos acerca de los tipos incluidos (propiedades, atributos, métodos, firmas, clases base...), el código intermedio MSIL (Microsoft Intermediate Language, similar a los bytecodes de Java) y los recursos adicionales que sean necesarios (imágenes, textos...). En definitiva, viene a ser algo parecido a los ficheros JAR (Java archives) de Java.

En la plataforma .NET, una aplicación está formada por uno o varios assemblies. Al poder coexistir distintas versiones de un assembly, se eliminan muchos de los problemas que caracterizan a las aplicaciones típicas de Windows, facilitando el despliegue, actualización y eliminación de aplicaciones. De hecho, una aplicación concreta podría utilizar simultáneamente varias versiones de un assembly.





*Figura 3: Biblioteca de clases .NET.*

### **Desarrollo de aplicaciones. Formularios Windows**

Los formularios Windows están contruidos sobre la base de la plataforma .NET y permiten construir complejas aplicaciones Windows en un entorno de desarrollo visual de aplicaciones (RAD: Rapid Application Development), tal como hasta ahora se venía haciendo con lenguajes del estilo de Visual Basic o Delphi.

### **Formularios web**

Los formularios web, que se construyen con ASP.NET, constituyen la evolución natural y lógica de ASP. Siguiendo el mismo estilo que su antecesor (editar una página y listo), ASP.NET permite utilizar controles complejos, facilita la gestión de sesiones, permite separar la interfaz de la lógica interna, elimina la distinción entre ASP e ISAPI y nos permite emplear cualquier lenguaje de programación que esté soportado por la plataforma .NET.

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

### Soporte para múltiples lenguajes

Para permitir el desarrollo de componentes utilizando múltiples lenguajes de programación, la plataforma .NET establece un sistema de tipos común (CTS: Common Type System) y una especificación que permite que puedan interactuar fragmentos de código escritos en distintos lenguajes (CLS: Common Language Specification).

La plataforma .NET permite utilizar una amplia gama de lenguajes de programación, como es el caso de:

- C#: Un nuevo lenguaje creado para la plataforma .NET. Se puede considerar una versión "segura" de C++ y viene a ser un híbrido entre Java (de Sun) y Delphi (de Borland). Es un lenguaje de programación orientado a objetos que pretende facilitar el desarrollo de componentes de software robustos y duraderos que preserven la inversión realizada en su desarrollo.
- Visual Basic .NET: Moderniza y simplifica el lenguaje de programación Visual Basic, con algunas novedades sintácticas, herencia simple, tratamiento de hebras y manejo de excepciones.

### Ejecución de código

Para que un lenguaje de programación sea soportado por la plataforma .NET, ha de existir un compilador que traduzca de este lenguaje a MSIL ("managed code"). A la hora de ejecutar el código intermedio este es siempre compilado a código nativo.

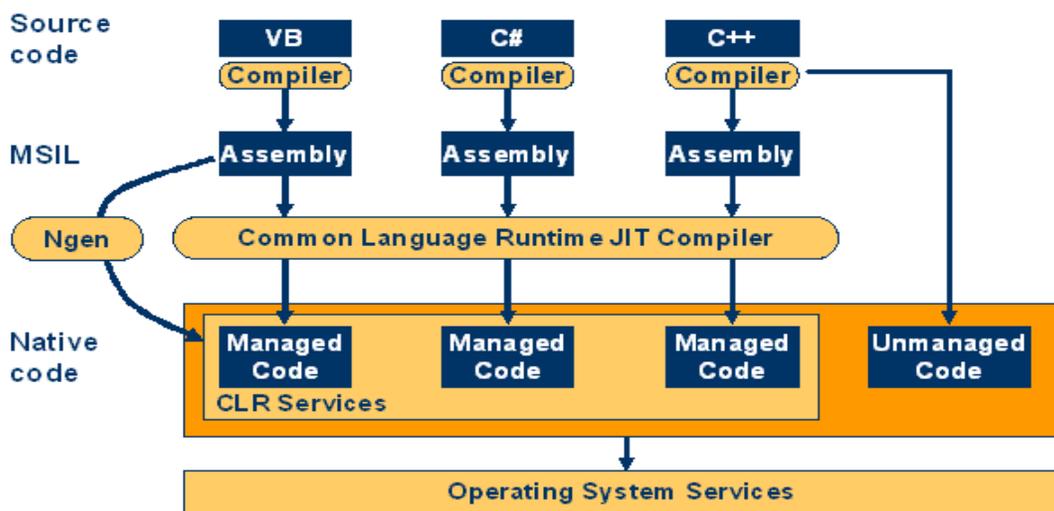


Figura 4: Estructura del compilador de la plataforma .NET.

### Elementos de reutilización

Reutilizar, ya sean elementos de diseño o código, disminuye el tiempo de desarrollo y abarata los costos de producción, a lo que se le podría añadir que permite concentrarse en el dominio del problema y no en problemas que ya se encuentran evaluados, analizados e incluso solucionados por otras personas. Un elemento importante para lograr efectividad y buen impacto organizativo en el diseño, consiste en institucionalizar la reutilización. Una idea que podría apoyarse es la de incentivar la búsqueda de oportunidades para reutilizar desde una perspectiva coordinada. (20)

La reutilización efectiva del diseño se alcanza mejor asignándoles esta tarea a personas en específico, los que serían los responsables de garantizar la actividad concretamente además de extender el análisis hasta el impacto de dicha reutilización en la fase de implementación del sistema. (21).

Para nuestra investigación la planificación de la reutilización se realizó mediante las estrategias que propone RUP en sus fases y flujos de trabajo, específicamente diseño e implementación.

Aunque si bien no es el único elemento que debe tener un buen diseño, la reutilización y más aún, la reutilización institucionalizada, es uno de los más importantes atractivos del proceso de diseño a nivel industrial. (22)

## CAPÍTULO 1: DISEÑO Y PLANIFICACIÓN DE LA SOLUCIÓN.

Proceso	Reutilización	Código, Contexto, Ideas	Objetivo	Por Ciento
<b>Construcción de los componentes.</b>	Sí	Framework 2.0	Utilizar los artefactos que brinda para una mejor implementación.	100%
<b>Construcción de Interfaz Visual.</b>	Sí	SharpLibrary.dll y MagicLibrary.dll	Mantener una Configuración estandarizada o modular de los componentes del framework.	65%
<b>Generación de documentos pdf</b>	Sí	IttextSharp	Generar documentos pdf.	75%
<b>Configuración de los XML</b>	Sí	System.XML	Leer y Persistir en los XML.	80%

**Tabla 5: Elementos de reutilización.**

### **Resultados esperados.**

Como conclusión de los resultados esperados, se obtuvo un componente, el cual posibilita un mejor aprovechamiento del tiempo de desarrollo. Resultado que responde a las problemáticas existentes y expuestas en este trabajo acerca de CEDRUX. El mismo consta de varios componentes los cuales están integrados dentro de una aplicación permitiendo al usuario realizar el uso de sus diversas funcionalidades. En este caso los componentes que brinda dicha aplicación son: Configuración de la gestión del multilinguaje que consta de 2727 líneas de código y que se encuentra en su versión 1.0 y el componente Gestión de trazas que consta de 2088 líneas de código y que se encuentra en su versión 1.0, constituyendo un total de 4815 de líneas de código que conforman a una parte de la aplicación.

Además brindan documentos que permiten conocer los requisitos funcionales y no funcionales más importantes que brindan cada uno de los componentes, así como los casos de prueba, el cual el componente de Configuración de la gestión del multilinguaje consta de 61 páginas. Por su parte el componente Gestión de trazas consta de 30 páginas.

Estos documentos se realizaron con el objetivo de brindar una mejor explicación del funcionamiento de los componentes así como un mejor entendimiento y operatividad.

### **Conclusiones**

En este capítulo se realizó un análisis de los principales problemas relacionados con la configuración manual del multilinguaje y la realización lenta y personalmente de la gestión de trazas en las distintas líneas de desarrollo y módulos del Sistema Integral de Gestión Cedrux, que afectan el rendimiento de la producción. Se realizó un cuestionario a una muestra significativa de los trabajadores del proyecto la cual reflejó la necesidad de desarrollar el componente para la configuración de la gestión del multilinguaje. Se efectuó un estudio del estado actual del diseño de software. Además se incluye la vista tecnológica de estos componentes que formaran parte del IDE para el marco de trabajo de Cedrux a través de la cual se toma como decisión desarrollar el componente usando la plataforma .NET y el framework que propone la misma. Finalmente se muestran los elementos que fueron reutilizados en el desarrollo del componente.

### CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

#### Introducción

En este capítulo se hace un levantamiento de los requisitos funcionales y no funcionales de los componentes desarrollados. También una muestra del prototipo de interfaz y el diseño de clases de los mismos, y además los artefactos de implementación y los casos de prueba realizados para lograr una mejor comprensión de la funcionalidad e integralidad de los componentes.

#### Requisitos

##### Requisitos funcionales

En el (

Anexo1: Plantilla de encuesta realizada a los distintos roles que integran **el proyecto ERP**.

Encuesta para la valoración de la configuración manual de los archivos de configuración de la gestión del multilinguaje y la realización lenta y personalmente de la gestión de trazas del Sistema Integral de Gestión Cedrux.

Estimado \_\_\_\_\_ usuario:  
Con el objetivo de ayudarlos a hacer más eficiente su trabajo, estamos realizando un diagnóstico sobre los problemas relacionados con configuración de los archivos de configuración de la gestión del multilinguaje y el proceso gestión de trazas del Sistema Integral de Gestión Cedrux. Le pedimos responda con la mayor veracidad el siguiente cuestionario:

**1. Señale el rol que ocupa dentro de su equipo de trabajo.**

- Jefe de Línea
- Analista
- Planificador
- Arquitecto
- Desarrollador
- Otro (Por favor especifique)

2. Diga si usted ha utilizado los archivos de configuración de la gestión del multilinguaje. (json)

- Si       No

3. Diga si usted ha participado en el proceso de gestión de trazas en su línea.

- Si       No

4. ¿Se encuentra satisfecho con la configuración manual de la gestión del multilinguaje?

- Si       No

5. ¿Se encuentra satisfecho con la realización de forma personal y lentamente de la gestión de trazas?

**Si**       **No**

**6. Señale las dificultades encontradas durante la configuración manual de de la gestión del multilinguaje.**

- Perdida de tiempo.**
- Introducción de errores.**
- Incremento del tiempo de máquina.**
- Baja productividad del equipo de trabajo.**
- Disminución de la moral del equipo de trabajo.**
- Otro (Por favor especifique)**

**7. Señale las dificultades encontradas durante la realización de forma personal y lentamente de la gestión de trazas**

- Atraso con el cronograma de solución de las trazas.**
- Insatisfacción en la realización del proceso.**
- Otro (Por favor especifique)**

8. ¿Considera necesario el desarrollo de un componente para la configuración de los archivos que gestionan el multilinguaje del Sistema Integral de Gestión Cedrux?

Si  No

9. ¿Considera necesario el desarrollo de un componente para gestión de trazas del Sistema Integral de Gestión Cedrux?

Si  No

Anexo 2: Requisitos funcionales del componente configuración de la gestión del multilinguaje.) se muestra una descripción detallada de los requisitos funcionales del componente configuración de la gestión del multilinguaje. Los mismos están organizados por escenarios para una mejor comprensión.

**a. Escenario: Gestionar proyecto.**

- ✓ RF -1: Crear nuevo proyecto.
- ✓ RF -2 Cargar un proyecto existente.
- ✓ RF -3: Guardar Proyecto como...
- ✓ RF -4: Guardar Proyecto.

**b. Escenario: Gestionar fichero .json.**

- ✓ RF -1: Crear nuevo fichero .json.
- ✓ RF -2: Mostrar vista previa del fichero .json.
- ✓ RF -3: Generar fichero .json.

- ✓ RF -4: Copiar fichero .json a otro directorio.
- ✓ RF -5: Eliminar fichero .json.
- c. Escenario: Gestionar componente (s).**
- ✓ RF -1: Adicionar componente (s) al fichero .json.
- ✓ RF -2: Editar componente al fichero json.
- ✓ RF- 3: Ordenar o desordenar los componentes del fichero .json.
- ✓ RF- 4: Vista de componente (s) del fichero .json.
- ✓ RF -5: Eliminar componente (s) al fichero .json.

En el (

### Anexo1: Plantilla de encuesta realizada a los distintos **roles que integran el proyecto ERP.**

Encuesta para la valoración de la configuración manual de los archivos de configuración de la gestión del multilinguaje y la realización lenta y personalmente de la gestión de trazas del Sistema Integral de Gestión Cedrux.

Estimado usuario:  
Con el objetivo de ayudarlos a hacer más eficiente su trabajo, estamos realizando un diagnóstico sobre los problemas relacionados con configuración de los archivos de configuración de la gestión del multilinguaje y el proceso gestión de trazas del Sistema Integral de Gestión Cedrux. Le pedimos responda con la mayor veracidad el siguiente cuestionario:

#### 1. Señale el rol que ocupa dentro de su equipo de trabajo.

Jefe de Línea

- Analista**
- Planificador**
- Arquitecto**
- Desarrollador**
- Otro (Por favor especifique)**

**2. Diga si usted ha utilizado los archivos de configuración de la gestión del multilinguaje. (json)**

- Si**       **No**

**3. Diga si usted ha participado en el proceso de gestión de trazas en su línea.**

- Si**       **No**

**4. ¿Se encuentra satisfecho con la configuración manual de la gestión del multilinguaje?**

- Si**       **No**

**5. ¿Se encuentra satisfecho con la realización de forma personal y lentamente de la gestión de trazas?**

- Si**       **No**

**6. Señale las dificultades encontradas durante la configuración**

manual de de la gestión del multilinguaje.

- Perdida de tiempo.
- Introducción de errores.
- Incremento del tiempo de máquina.
- Baja productividad del equipo de trabajo.
- Disminución de la moral del equipo de trabajo.
- Otro (Por favor especifique)

7. Señale las dificultades encontradas durante la realización de forma personal y lentamente de la gestión de trazas

- Atraso con el cronograma de solución de las trazas.
- Insatisfacción en la realización del proceso.
- Otro (Por favor especifique)

8. ¿Considera necesario el desarrollo de un componente para la configuración de los archivos que gestionan el multilinguaje del Sistema Integral de Gestión

CedruX?

Si  No

9. ¿Considera necesario el desarrollo de un componente para gestión de trazas del Sistema Integral de Gestión CedruX?

Si  No

Anexo 2: Requisitos funcionales del componente configuración de la gestión del multilinguaje.) se muestra una descripción detallada de los requisitos funcionales del componente Gestión de trazas. Los mismos están estructurados por escenarios para su mejor comprensión.

**a. Escenario: Gestionar usuario(s) para el soporte.**

- ✓ RF-1: Crear usuario.
- ✓ RF -2: Modificar usuario.
- ✓ RF-3: Eliminar usuario (s).
- ✓ RF-4: Exportar usuario (s).
- ✓ RF-5: Asignar y notificar traza (s) a usuario.

**b. Escenario: parámetros de configuración.**

- ✓ RF-1: Parámetros de configuración.

**c. Escenario: Gestionar trazas.**

- ✓ RF -1: Importar trazas.
- ✓ RF -2: Exportar trazas.
- ✓ RF -3: Asignar usuario a la (s) trazas (s).

- ✓ RF-4: Mostrar trazas asignadas.

### **Requisitos no funcionales**

#### **1. Usabilidad (USB)**

Los componentes podrán ser usados por cualquier persona que posea conocimientos básicos en el manejo de la computadora y además de la estructura de los XML del Sistema Integral de Gestión de Entidades.

#### **2. Rendimiento (REN)**

Los tiempos de respuesta y velocidad de procesamiento de información serán rápidos, no mayores de 5 segundos para las actualizaciones y 10 para las recuperaciones.

#### **3. Seguridad (SEG)**

No presentan un sistema de autenticación, puesto que solo tendrán acceso a estos, las personas calificadas para interactuar con el manejo y configuración de de los ficheros de configuración de la gestión del multilinguaje así como la gestión de trazas del sistema Cedrux.

#### **4. Portabilidad (POR)**

Los componentes solo trabajaran sobre Windows.

#### **5. Soporte (SOP)**

El componente contará antes de su puesta en marcha con un período de pruebas, se le dará mantenimiento, configuración y se brindará el servicio de instalación.

#### **6. Políticos Culturales (CUL)**

El sistema solo podrá ser utilizado en territorio cubano y por las entidades autorizadas por el Ministerio de las FAR.

El producto no debe contener palabras en otros idiomas.

### 7. Software

- Sistema operativo Windows 98 o superior.
- Componente Framework.Net en su versión 2.0 o superior.
- Servidor de correo.

### 8. Hardware

Para el cliente:

- Requerimientos mínimos: Procesador Pentium III a 133Mhz con 128 Mb de memoria RAM.
- Tarjeta de red.

### Prototipo de interfaz

Los prototipos de interfaz son los que les permite al cliente relacionar los requisitos funcionales establecidos por cada componente con cada escenario, además de proporcionarle una mejor manipulación de los mismos.

- Prototipos de interfaz del componente configuración de la gestión del multilinguaje.

**Escenario: Gestionar proyecto.**

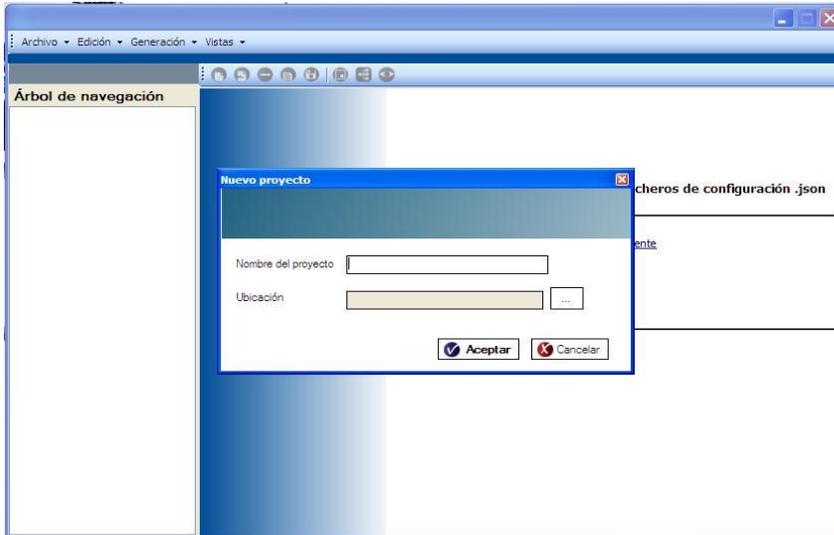


Figura 5: Interfaz correspondiente al RF-1.

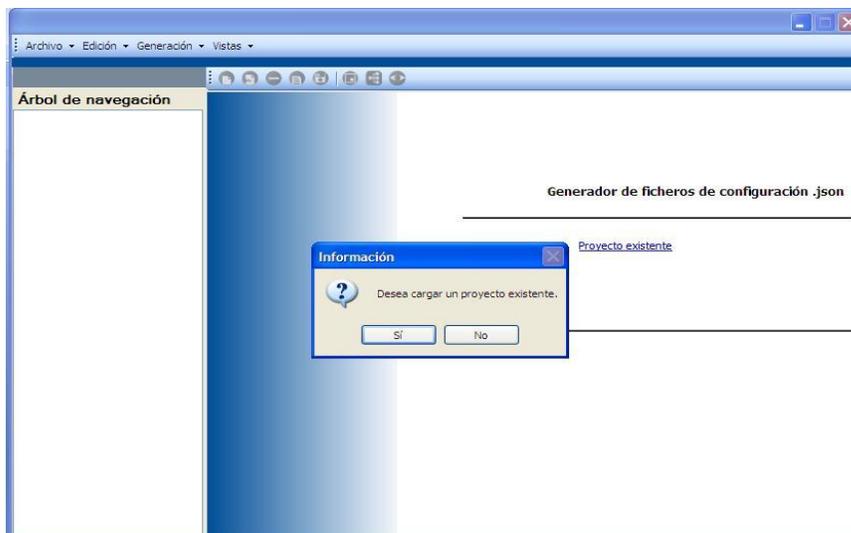
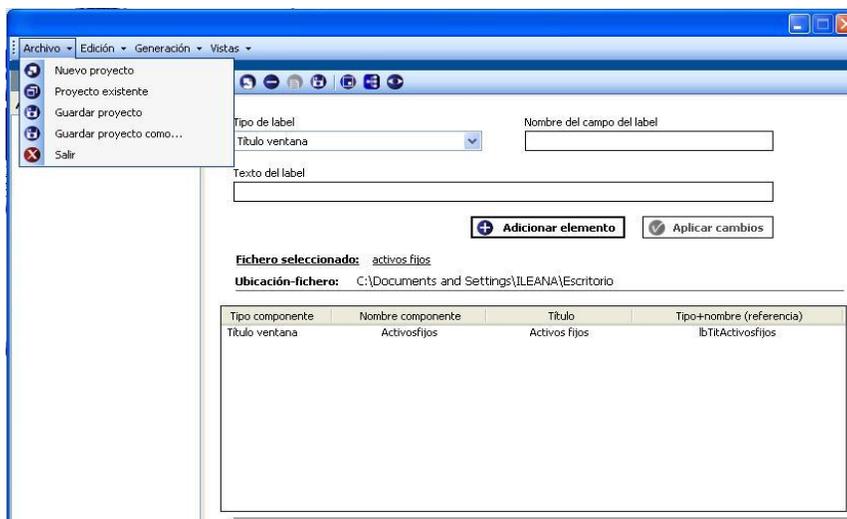
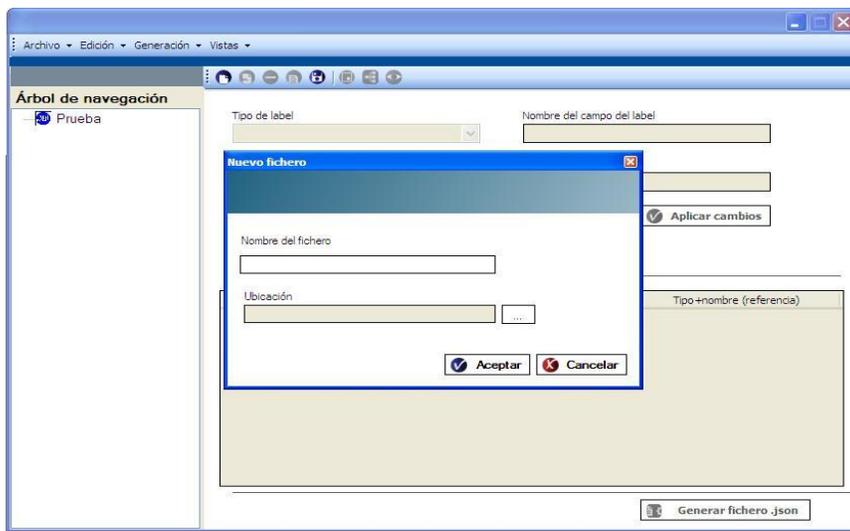


Figura 6: Interfaz correspondiente al RF-2.

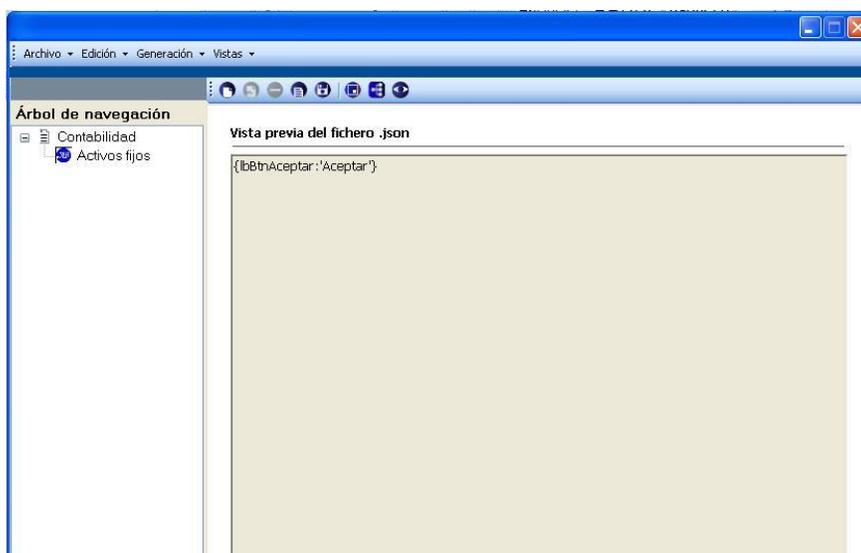


**Figura 7: Interfaz correspondiente a los RF-3 y RF-4.**

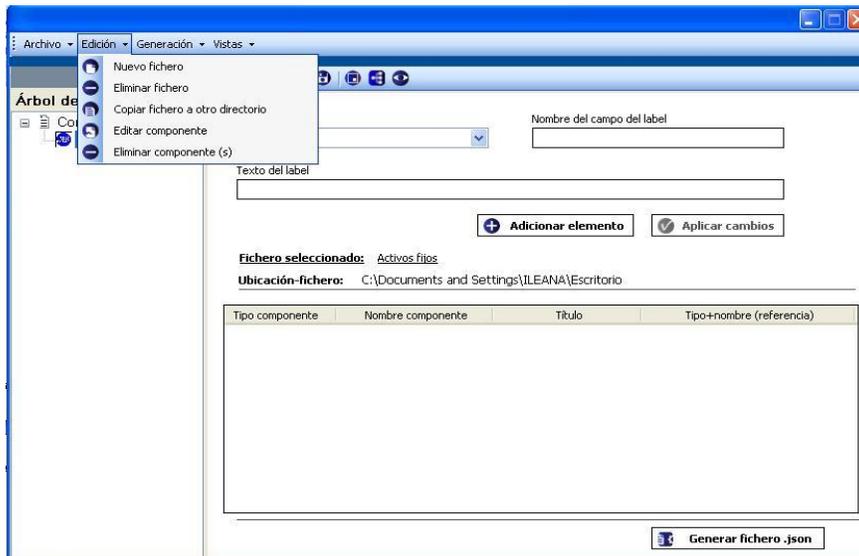
**Escenario: Gestionar fichero.**



**Figura 8: Interfaz correspondiente al RF-1**

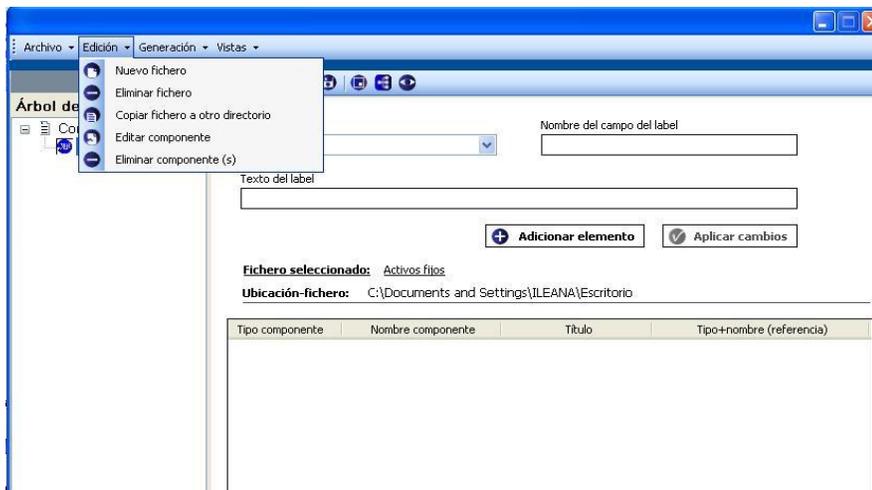


**Figura 9: Interfaz correspondiente al RF-2**

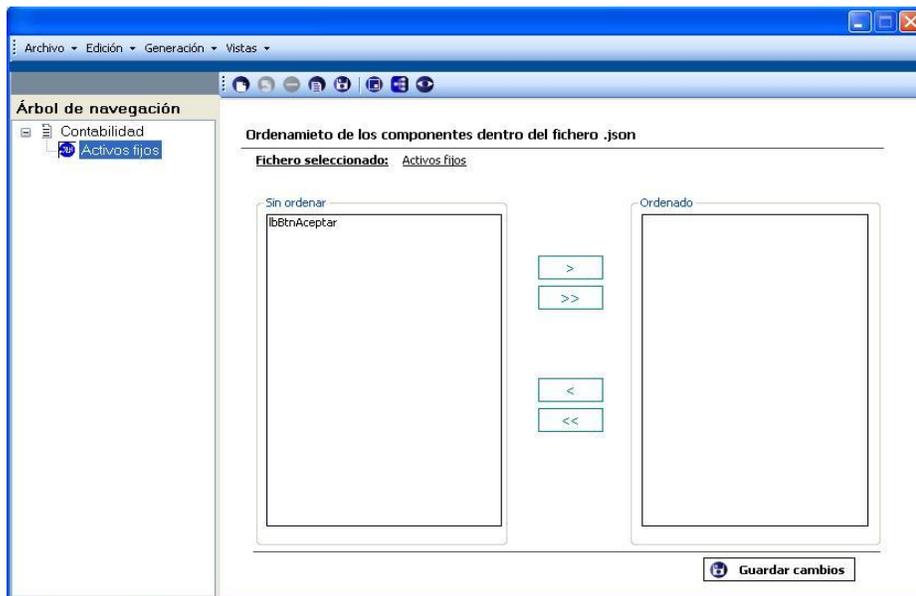


**Figura 10: Interfaz correspondiente a los RF-3, RF-4 y RF-5.**

**Escenario: Gestionar componente.**



**Figura 11: Interfaz correspondiente a los RF-1, RF-2, RF-4 y RF-5.**



**Figura 12: Interfaz correspondiente al RF-3.**

- Prototipos de interfaz del componente Gestión de trazas.

**Escenario: Gestionar usuario (s) para el soporte.**



Figura 13: Interfaz correspondiente al RF-1.

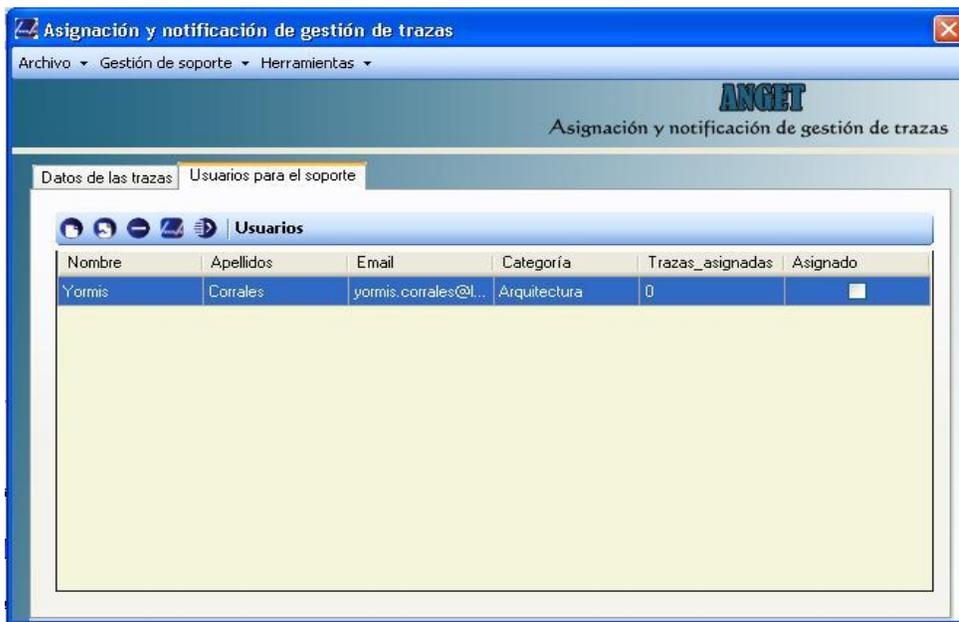


Figura 14: Interfaz correspondiente al RF-2, RF-3, RF-4 y RF-5.

### Escenario: Gestionar trazas.

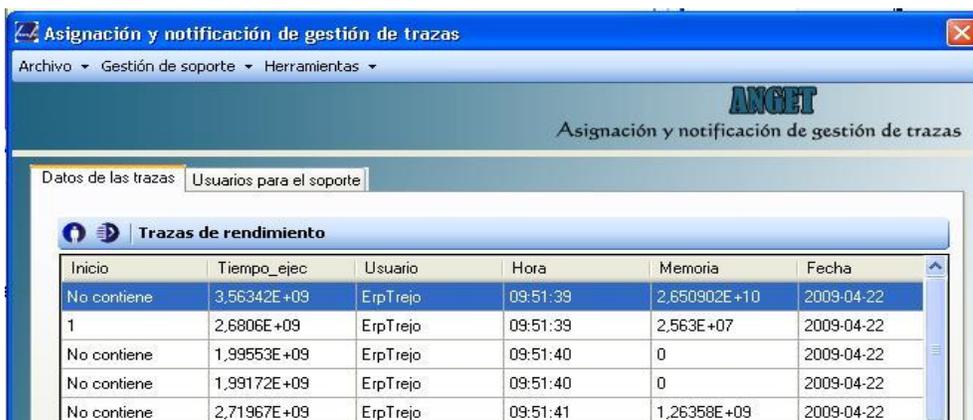


Figura 15: Interfaz correspondiente al RF-1, RF-2 y RF-3.



Figura 16: Interfaz correspondiente al RF-4.

Escenario: Gestionar parámetros de configuración.



*Figura 17: Interfaz correspondiente al RF-1.*

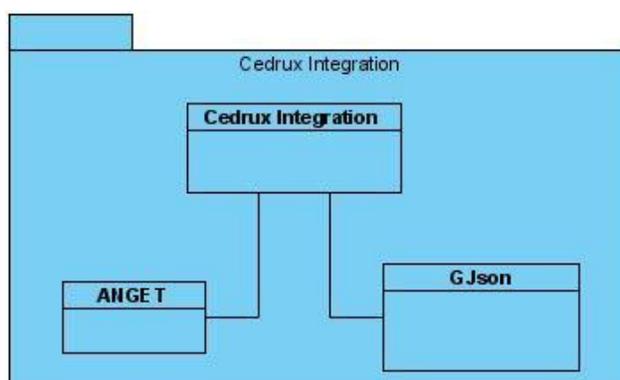
### Diseño de clases

En el (¡Error! No se encuentra el origen de la referencia.) se muestra el diagrama de clases perteneciente al componente configuración de la gestión del multilinguaje.

En el (¡Error! No se encuentra el origen de la referencia.) se muestra el diagrama de clases perteneciente al componente gestión de trazas.

En el siguiente diseño de clases se muestra la integración final de los componentes que tiene comprendido el componente principal (Ver.

Figura 68).



*Figura 68: Diseño de clases de la integración del componente general.*

### Artefactos de implementación

Los artefactos de implementación que no son más que los productos tangibles generados en proceso de desarrollo de los componentes, en este caso los casos de prueba, diagramas de clases, código de fuente, la estructura de los ficheros de configuración .json, así como la estructura de los ficheros XML de donde se realiza la lectura de los datos para trabajar posteriormente con el componente de gestión de trazas y la generación de documentos .pdf de las trazas. También el diagrama de componentes que establece la lógica del diseño de las clases de los componentes integrados dentro del componente desarrollado.

### Esquemas de la configuración de los ficheros .json.

```
{lbBtnAceptar:'Aceptar',
lborden:'Orden',
lbprefijo:'Prefijo',
lbrequisitosConocimiento:'Requisitos de conocimiento',
lbtareasprincipales:'Tareas principales',
lbnomencladores:'Nomencladores',
lbTipoCalif:'Tipo de Calificador',
lbdenominacion:'Denominación',
lborgano:'Órgano',
lbTipocifra:'Tipo cifra',
lbcargocivil:'Cargo civil',
lbCalificador:'Calificador',
lbcargomilitar:'Cargo militar',
lbCategoriaocupacional:'Categoría ocupacional',
lbGrupodecomplejidad:'Grupo de complejidad',
lbTecnica:'Técnica',
lbModulo:'Módulo',
lbAgrupa:'Agrupación',
lbGradomilitar:'Grado militar',
lbtipoDPA:'Tipo DPA',
lbDPA:'DPA'}
```

**Figura 19: Esquema del XML de ficheros .json.**

- Esquema de XML Trazas de rendimiento.

```
<?xml version="1.0" encoding="UTF-8" ?>
<xml>
  <trazas tipo="Rendimiento">
    <traza>
      <falla>No contiene</falla>
      <inicio>No contiene</inicio>
      <accion>entraralsistema</accion>
      <controlador>index</controlador>
      <referencia>portal</referencia>
      <fecha>2009-04-22</fecha>
      <hora>09:51:39</hora>
      <usuario>ErpTrejo</usuario>
      <idestructuracomun>10000004</idestructuracomun>
      <tiempoejecucion>0.356342000</tiempoejecucion>
      <memoria>2.650902000</memoria>
      <categoria>Arquitectura</categoria>
    </traza>
  </trazas>
</xml>
```

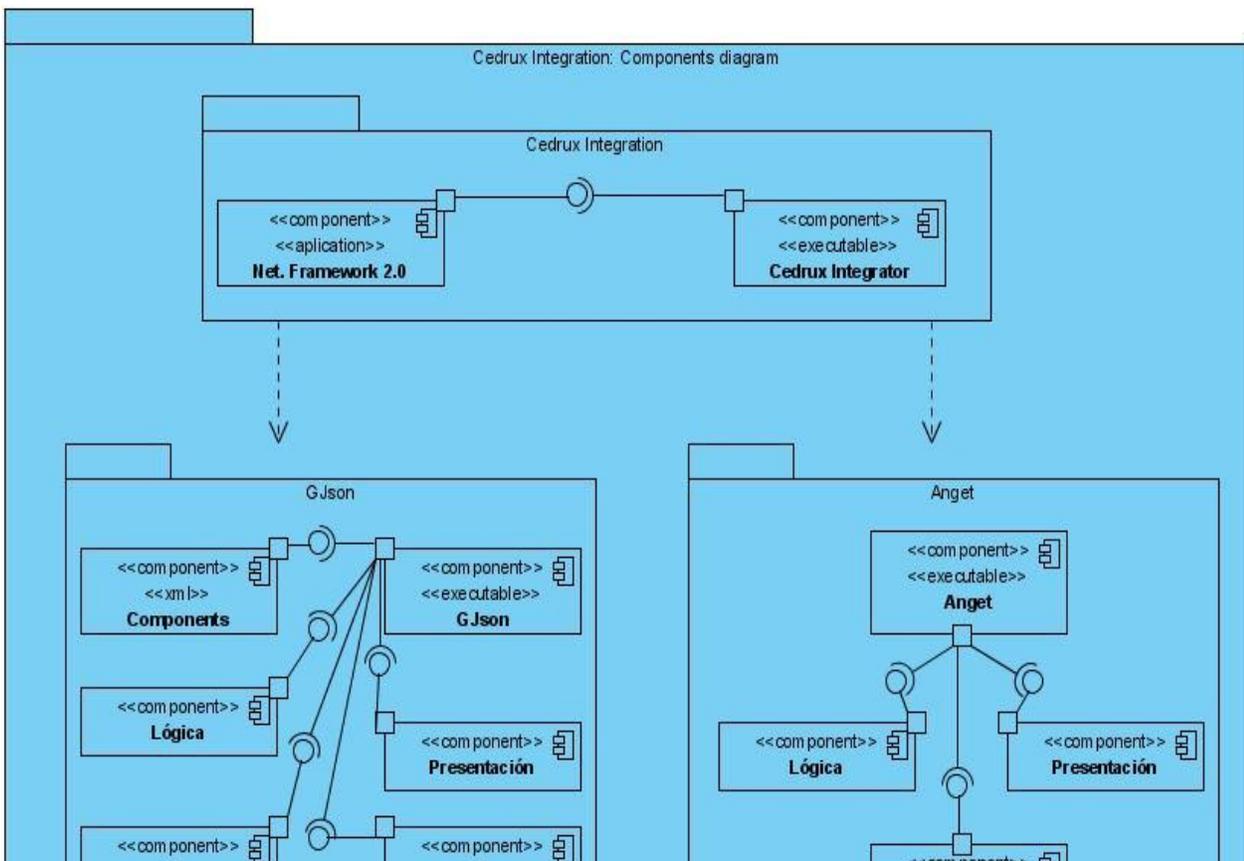
**Figura 20: Esquema del XML de trazas de rendimiento.**

- Esquema de XML Trazas de excepción-integración.

```
<?xml version="1.0" encoding="UTF-8" ?>
<xml>
  <trazas tipo="Excepción de Integración">
    <traza>
      <idestructuracomun>10000003</idestructuracomun>
      <usuario>logistica</usuario>
      <hora>15:32:41</hora>
      <fecha>2009-03-01</fecha>
      <mensaje>Error de integración. Uno de los parámetr
      <descripcion>Ocurrió un error interno del sistema.
      <log>No posee</log>
      <origen>utiles</origen>
      <destino>movimientos</destino>
      <accion>adicionarubicacion</accion>
      <controlador>gestubicproducto</controlador>
      <clase>No posee</clase>
      <metodo>No posee</metodo>
      <codigo>I0C002</codigo>
      <categoria>Traza</categoria>
    </traza>
  </trazas>
</xml>
```

Figura 21: Esquema del XML de trazas de excepción-integración.

### Diagrama de componentes.



*Figura 22: Diagrama de componentes del sistema.*

### Casos de pruebas

Los casos de pruebas o Test Case son condiciones o variables para determinar si los requisitos de una aplicación son parciales o satisfactorios. Se pueden realizar muchos casos de pruebas para determinar si un requisito es satisfactorio. Para esto se puede realizar un caso de pruebas como mínimo por requisitos a no ser que dicho requisito presente requisitos secundarios, en ese caso cada requisito secundario deberá tener como mínimo un caso de prueba para evaluar sus funcionalidades. (23) Algunas metodologías como RUP recomiendan realizar dos casos de pruebas como mínimo por requisitos, donde una realizaría las pruebas positivas y otras las negativas. (24)

Si la aplicación es creada sin requisitos formales, entonces los casos de pruebas se escriben basados en la operación normal de programas de una clase similar. Lo que caracteriza un escrito formal de caso de prueba es que hay una *entrada conocida* y una *salida esperada*, los cuales son formulados antes de que se ejecute la prueba. La *entrada conocida* debe probar una precondición y la *salida esperada* debe probar una post - condición. (23)

### **Características de los casos de pruebas dirigidos a escenarios de casos de uso de un sistema.**

Los requerimientos son la fuente principal para obtener los casos de prueba pero no son el único medio. Muchas veces son insuficientes para proporcionar una base completa que permita desarrollar las pruebas. Por lo que es necesario considerar otros elementos como riesgos, restricciones, tecnologías, cambios, fallos, etc. (25)

Por lo normal los casos de pruebas se derivan de los casos de uso del modelo de casos de uso o del modelo de casos del diseño, con los que se validan los requisitos del sistema. Algunos de los casos de uso más comunes son:

1. **Los Casos de Prueba que especifica cómo probar un caso de uso o un escenario específico de un caso de uso.** Un Caso de Prueba de este tipo incluye la verificación del resultado de la interacción entre los actores y el sistema, que se satisfacen las precondiciones y post - condiciones especificadas por el caso de uso y que se sigue la secuencia de acciones especificadas por el caso de uso. Un Caso de Prueba basado en un caso de uso especifica típicamente una prueba del sistema como “caja negra”, es decir, una prueba del comportamiento observable externamente del sistema.

2. **Los Casos de Prueba que especifica cómo probar una realización de caso de uso-diseño o un escenario específico de la realización.** Un Caso de Prueba de este tipo puede incluir la verificación de la interacción entre los componentes que implementan dicho caso de uso. Los Casos de Prueba basados en una realización de un caso de uso típicamente especifican una prueba del sistema como “caja blanca”, es decir, una prueba de la interacción interna entre los componentes del sistema. (25)

### **Casos de prueba realizados a la solución propuesta.**

A modo de resumen se puede confirmar que la realización de casos de pruebas tiene como objetivo enmarcar todas las funcionalidades que brinda el sistema y comprobar que las mismas cumplen con los requisitos funcionales trazados durante la fase de desarrollo del componente. Además permiten valorar el correcto funcionamiento y operabilidad para

cada escenario previsto con sus particulares características. También sirve como guía al usuario para saber que acción o respuesta corresponde luego de realizar alguna operación dentro de la aplicación.

Se realizaron 24 casos de pruebas a la solución propuesta. La descripción de los mismos se realizó mediante escenarios.

A continuación se muestra una breve descripción del componente configuración de la gestión del multilinguaje enfocándose en el escenario: Gestionar proyecto (Ver. Tabla 1), las demás descripciones detalladas correspondientes al componentes se podrán encontrar en el (¡**Error! No se encuentra el origen de la referencia.**!), al igual que la del componente Gestión de trazas (¡**Error! No se encuentra el origen de la referencia.**

- **Configuración de la gestión del multilinguaje.**

**DCP\_Cargar\_proyecto\_existente.**

**1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft. NET Framework versión 2.0 u otra superior.
- Tiene que existir al menos un proyecto.
- Debe haber guardado anteriormente un proyecto.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
----------------------	---------------------	-----------------------	---------------------

## CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

1: Cargar proyecto existente.	Se carga un proyecto que fue creado anteriormente.	EP 1.1: Cargar proyecto existente.	<ul style="list-style-type: none"><li>– Se pulsa en el vínculo <b>Proyecto existente</b> de la interfaz principal.</li><li>– Se acepta el mensaje de confirmación.</li><li>– Se busca la dirección donde se encuentra.</li><li>– Se pulsa el botón <b>Abrir</b>.</li></ul>
		EP 1.2: Cargar proyecto existente.	<ul style="list-style-type: none"><li>– Se da clic en <b>Archivo/Proyecto existente</b>.</li><li>– Se acepta el mensaje de confirmación</li><li>– Se busca la dirección donde se encuentra.</li><li>– Se presiona el botón <b>Abrir</b>.</li></ul>
		EP 1.3: Cancelar la operación.	<ul style="list-style-type: none"><li>– Se presiona el vínculo <b>Proyecto existente</b>.</li><li>– Se busca o no la dirección donde se encuentra.</li><li>– Se pulsa el botón <b>Cancelar</b>.</li></ul>

## CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

		EP 1.4: Cancelar la operación.	<ul style="list-style-type: none"> <li>– Se da clic en <b>Archivo/Proyecto existente</b>.</li> <li>– Se busca o no la dirección donde se encuentra.</li> <li>– Se presiona el botón <b>Cancelar</b>.</li> </ul>
		EP 1.5: Cancelar la operación.	<ul style="list-style-type: none"> <li>– Se presiona en el vínculo <b>Proyecto existente</b> o dar clic en <b>Archivo/Proyecto existente</b>.</li> <li>– No se acepta el mensaje de confirmación.</li> </ul>

*Tabla 6: Configuración de la gestión del multilinguaje, DCP\_Cargar\_proyecto\_existente.*

### 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	NA	NA	NA	NA

### 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Dirección	Respuesta del sistema	Resultado de la prueba

## CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

EP 1	Cargar proyecto existente	V(Escritorio)	Muestra una ventana de confirmación " Desea cargar proyecto existente", después del usuario aceptar la opción se abre un navegador donde se selecciona la dirección donde se encuentra el proyecto a cargar. Se abre el proyecto seleccionado.	
EP 2	Cargar proyecto existente	V(Documents and Settings)	Muestra una ventana de confirmación " Desea cargar un proyecto existente", después del usuario aceptar la opción se abre un navegador donde se selecciona la dirección donde se encuentra el proyecto a cargar. Se abre el proyecto seleccionado.	
EP 3	Cancelar la operación.	N/A	Se cancela la operación.	
EP 4	Cancelar la operación.	N/A	Se cancela la operación.	
EP 5	Cancelar la operación.	N/A	Se cancela la operación.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

### DCP\_Crear\_nuevo\_proyecto.

#### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- El proyecto no ha sido creado, no existe.

#### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Crear nuevo proyecto.	Se crea un nuevo proyecto.	EP 1.1: Crear nuevo proyecto.	<ul style="list-style-type: none"> <li>- Se da clic en el vínculo <b>Nuevo proyecto</b> en la interfaz principal.</li> <li>- Se introducen los datos del proyecto que va a crear, en este caso el nombre del proyecto.</li> <li>- Se escoge una dirección en el explorador.</li> <li>- Se presiona el botón <b>Aceptar</b>.</li> </ul>
		EP 1.2: Crear nuevo proyecto.	<ul style="list-style-type: none"> <li>- Se da clic en <b>Archivo/Nuevo proyecto</b>.</li> <li>- Se introduce el nombre del nuevo proyecto que se va a crear.</li> <li>- Se escoge una dirección en el explorador.</li> <li>- Se presiona el botón <b>Aceptar</b>.</li> </ul>

## CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

		EP 1.3: Adicionar proyecto dejando campos requeridos en blanco.	<ul style="list-style-type: none"><li>- Se da clic en el vínculo <b>Nuevo proyecto</b>.</li><li>- Se introducen los datos del proyecto que se desea crear, dejando al menos un campo requerido en blanco.</li><li>- Se acepta el mensaje de información que muestra el sistema.</li><li>- Se introducen todos los datos del proyecto que se desea crear.</li><li>- Se presiona el botón <b>Aceptar</b>.</li></ul>
		EP 1.4: Adicionar proyecto dejando campos requeridos en blanco.	<ul style="list-style-type: none"><li>- Se da clic en <b>Archivo/Nuevo proyecto</b>.</li><li>- Se introducen los datos del proyecto que se desea crear, dejando al menos un campo requerido en blanco.</li><li>- Se acepta el mensaje de información que muestra el sistema.</li><li>- Se introducen todos los datos del proyecto que se desea crear.</li><li>- Se presiona el botón <b>Aceptar</b>.</li></ul>
		EP 1.5: Cancelar la operación.	<ul style="list-style-type: none"><li>- Se da clic en el vínculo <b>Nuevo proyecto</b></li><li>- Se introducen o no los datos.</li></ul>

## CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

			<ul style="list-style-type: none"> <li>- Se presiona el botón <b>Cancelar.</b></li> </ul>
		EP 1.6: Cancelar la operación.	<ul style="list-style-type: none"> <li>- Se da clic en <b>Archivo/Nuevo proyecto.</b></li> <li>- Se introducen o no los datos.</li> <li>- Se presiona el botón <b>Cancelar.</b></li> </ul>

*Tabla 7: Configuración de la gestión del multilinguaje, DCP\_Crear\_nuevo\_proyecto.*

### 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	Nombre del proyecto	Campo de texto.	No	Combinación de letras.
2	Ubicación	Explorador de Windows.	No	Una dirección válida.

### 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Nombre	Variable 2 Dirección	Respuesta del sistema	Resultado de la prueba

## CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

EP 1	Crear nuevo proyecto.	V(proyecto)	V(C:\Documents and Settings\Yormis\Escritorio)	Se crea el proyecto, se cierra la ventana para adicionar proyecto y se guarda la información adicionada.	
EP 2	Crear nuevo proyecto.	V(calidad)	C:\Documents and Settings\ Yormis \Escritorio	Se crea el proyecto, se cierra la ventana para adicionar proyecto y se guarda la información adicionada.	
EP 3	Adicionar proyecto dejando campos requeridos en blanco.	I( )	C:\Documents and Settings\ Yormis \Escritorio	Se muestra un mensaje que indica que no se pueden dejar campos en blanco “Debe llenar todos los campos” y se mantiene en la interfaz para adicionar un proyecto.	

## CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

EP 4	Adicionar proyecto dejando campos requeridos en blanco.	V( gal )	I( )	Se muestra un mensaje que indica que no se pueden dejar campos en blanco “Debe llenar todos los campos” y se mantiene en la interfaz para adicionar un proyecto.	
EP 5	Cancelar la operación.	NA	NA	Se cancela la operación.	
EP 6	Cancelar la operación.	NA	NA	Se cancela la operación.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

### DCP\_Guardar\_proyecto\_como...

#### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft. NET Framework versión 2.0 u otra superior.
- Tiene que existir al menos un proyecto.

### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Guardar proyecto como...	Se guarda un proyecto que fue creado anteriormente.	EP 1.1: Guardar proyecto como...	<ul style="list-style-type: none"> <li>– Se marca con doble clic el proyecto que deseas guardar en el árbol de navegación.</li> <li>– Se va a <b>Archivo/Guardar proyecto como...</b> y se da clic.</li> <li>– Se selecciona la dirección donde quiere guardarlo.</li> <li>– Se pulsa el botón <b>Guardar.</b></li> </ul>
		EP 1.2: Cancelar la operación.	<ul style="list-style-type: none"> <li>– Se marca con doble clic el proyecto que deseas guardar en el árbol de navegación.</li> <li>– Se va a <b>Archivo/Guardar proyecto como...</b> y se da clic.</li> <li>– Se pulsa el botón <b>Cancelar.</b></li> </ul>

*Tabla 8: Configuración de la gestión del multilinguaje, DCP\_Guardar\_proyecto\_como.*

#### 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	Ubicación	Explorador de Windows.	No.	Una dirección válida.

### 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Dirección	Respuesta del sistema	Resultado de la prueba
EP 1	Guardar proyecto como.	V(Escritorio)	El sistema muestra un navegador para que el usuario escoja el lugar donde quiere guardar el proyecto. Se guarda el proyecto y se muestra la interfaz anterior.	
EP 2	Cancelar la operación.	N/A	Se cancela la operación.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

#### DCP\_Guardar\_proyecto.

##### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Debe existir al menos un proyecto.

##### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario

## CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

---

1: Guardar proyecto.	Se guarda el proyecto seleccionado.	EP 1.1: Guardar proyecto.	<ul style="list-style-type: none"><li>- Se marca con doble clic el proyecto que deseas guardar en el árbol de navegación.</li><li>- Se da clic <b>Archivo/Guardar proyecto.</b></li><li>- Se pulsa el botón <b>Aceptar.</b></li></ul>
		EP 1.2: Guardar proyecto.	<ul style="list-style-type: none"><li>- Se marca con doble clic el proyecto que deseas guardar en el árbol de navegación.</li><li>- Se da clic en el ícono <b>Guardar proyecto</b> que está en la barra de herramientas.</li><li>- Se pulsa el botón <b>Aceptar.</b></li></ul>

## CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

		<p>EP 1.3:          Guardar proyecto que no tenga fichero(s), y este último no tenga componentes adicionales.</p>	<ul style="list-style-type: none"> <li>- Se marca con doble clic el proyecto que deseas guardar en el árbol de navegación.</li> <li>- Se da clic en el ícono <b>Guardar proyecto</b> que está en la barra de herramientas o en <b>Archivo/Guardar proyecto</b>.</li> <li>- Se muestra un mensaje de información referente a la no existencia de ficheros y componentes.</li> <li>- Se acepta el mensaje anterior.</li> </ul>
--	--	---	--

*Tabla 9: Configuración de la gestión del multilinguaje, DCP\_Guardar\_proyecto.*

### 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	N/A	N/A	N/A	N/A

### 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Guardar proyecto.	Se guarda el proyecto, y se muestra un mensaje de acción realizada correctamente.	
EP 2	Guardar proyecto.	Se guarda el proyecto y se muestra un mensaje de acción realizada correctamente.	
EP 3	Guardar proyecto que no tenga	Se muestra un mensaje de información referente a la no existencia de ficheros y componentes.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/P no procede.

### **Conclusiones**

Durante el transcurso del capítulo se argumentaron los fundamentales aspectos que se llevan a cabo durante el proceso de análisis, comenzando por el levantamiento de los requisitos funcionales que contienen los componentes mencionados y de los requisitos no funcionales para garantizar una ejecución eficaz de la aplicación. También se reflejaron los distintos prototipos de interfaz que se relacionan con los requisitos levantados, se plasmó el diseño de clases para una mejor comprensión de cómo están estructuradas las clases que conforman toda la capa del negocio de los componentes. Se expusieron además los distintos artefactos de implementación generados: Esquemas de configuración de los archivos XML, diagrama de componentes y se describieron los casos de pruebas de los componentes para comprobar si las funcionalidades implementadas en cada uno cumple con los requisitos argumentados en el proceso de desarrollo.

### CAPÍTULO 3: ANÁLISIS DE LA SOLUCIÓN

#### Introducción.

En el presente Capítulo se muestran algunas métricas que se aplican en la actualidad para validar la calidad en el diseño de software y se definen cuales se aplicaron al diseño de cada uno de los componentes que integran la solución propuesta. Estas ayudan a comprender todo el proceso técnico que se utiliza para desarrollar un producto, así como el propio producto. También brinda una Valoración de la solución del componente y la matriz de inferencia de indicadores de calidad.

#### Valoración de la solución.

Confirmación por inspección y provisión de evidencia objetiva de que los requerimientos particulares para un uso específico son alcanzados. En diseño y desarrollo, la validación está relacionada con el proceso de reexaminación de un producto para determinar la conformidad con las necesidades del usuario. La validación es realizada normalmente sobre el producto final bajo condiciones operacionales definidas (26). La valoración de la solución es el paso final en el proceso de evaluación del software. Donde una solución puede ser valorada de diversas maneras, debido a que existen distintos tipos de valoraciones como son:

- Valoración cualitativa.
- Valoración indirecta.
- Valoración directa.

Una valoración se realiza mediante una métrica para asignar uno de los valores de una escala (el mismo puede ser numero o categoría) al atributo de la entidad (sistema, subsistemas o componentes).

**Valoración cualitativa:** Cuando se emplean categorías, como algunos de los atributos más importantes de una entidad, ejemplo: el lenguaje de desarrollo del programa (C, C ++, C #, PHP, JAVA). Es una evaluación sistemática del grado o capacidad de una entidad para satisfacer necesidades o requerimientos específicos.

**Valoración indirecta:** Es la valoración de un atributo derivada del valor de uno o más atributos diferentes. Es la valoración externa de un atributo de un sistema, ejemplo: el tiempo de respuesta a la información alimentada por el usuario, es una valoración indirecta de los atributos del software, debido a que esta medida se verá influenciada por los atributos externos del sistema, así como los propios internos.

**Valoración directa:** Es una valoración del producto, de forma indirecta o directa. Ejemplo: El número de líneas de código, las valoraciones de la complejidad, el número de fallas encontradas durante el proceso y el índice de señales o alertas, son todas las valoraciones internas propias del producto en sí. (26).

### 1.1 Valoraciones aplicadas a la solución propuesta.

Dentro de los distintos tipos de validación del software existentes se escogieron para la solución propuesta las de tipo cualitativa y directa.

Debido a las características que debería representar el sistema se realizó sobre la plataforma de desarrollo .NET de Microsoft, por la gama de facilidades que brinda para el desarrollo de componentes de software, posibilitando para el mismo la interacción con distintos lenguajes y compiladores, así como un rápido desarrollo de aplicaciones (una de las fundamentales necesidades del proyecto). El sistema contiene \_\_\_\_ líneas de código, es totalmente seguro y eficaz debido a que los ficheros no pueden ser configurables, presenta una interfaz amigable para el usuario y fácil de manipular.

### Métricas que evalúan el diseño, análisis de los resultados.

Cuando se planifica un proyecto se tiene que obtener estimaciones del costo y esfuerzo humano requerido por medio de las mediciones de software que se utilizan para recolectar los datos cualitativos acerca del software y sus procesos para aumentar su calidad. Dichas mediciones se realizan mediante un proceso conocido como Métricas, que no es más que un instrumento que cuantifica un criterio.

### 2.1 Métricas de software.

Las métricas del software no mantienen un tamaño estandarizado o estructurado, simplemente el proyecto debe elegir las métricas que cumple con las características o necesidades. Las mismas se basan en cuantificar la complejidad, funcionalidad y eficiencia relacionada con el desarrollo del software, enfatizando sus objetivos en:

- Comprender mejor la calidad del producto.
- Estimar la efectividad del proceso.
- Mejorar la calidad del trabajo realizado en el nivel del proyecto.

Las métricas del software se pueden catalogar como:

**Métricas técnicas:** Que se centran en las características de software por ejemplo: la complejidad lógica y el grado de modularidad.

**Métricas de calidad:** proporcionan una indicación de cómo se ajusta el software a los requisitos implícitos y explícitos del cliente.

**Métricas de productividad.** Se centran en el rendimiento del proceso de la ingeniería del software.

### Métricas orientadas a la función y las Métricas orientadas al tamaño.

### 2.2 Métricas orientadas a objetos.

Las métricas orientadas a objetos se crearon para resaltar lo esencial para el ingeniero de software que es la calidad del producto. Donde el enfoque fundamental de las métricas orientadas a objetos esta en las clases, pieza fundamental para la construcción del software.

El Software Orientado a Objetos (OO) es fundamentalmente distinto del software que se desarrolla utilizando métodos convencionales. Las métricas para sistemas OO deben de ajustarse a las características que distinguen el software OO del software convencional. Estas métricas hacen énfasis en el encapsulamiento, la herencia, complejidad de clases y polimorfismo. Por lo tanto las métricas OO se centran en métricas que se pueden aplicar a las características de encapsulamiento, ocultamiento de información, herencia y técnicas de abstracción de objetos que hagan única a esa clase. Como en todas las métricas los objetivos principales de las métricas OO se derivan del software

convencional: comprender mejor la calidad del producto, estimar la efectividad del proceso y mejorar la calidad del trabajo realizado a nivel del proyecto.

Las medidas y las métricas son componentes clave de cualquier disciplina de la ingeniería; la ingeniería de software orientada a objetos no es una excepción. Lamentablemente, la utilización de métricas para sistemas orientados a objetos ha progresado con mucha más lentitud que la utilización de los demás métodos OO. Sin embargo, a medida que los sistemas OO van siendo más habituales, resulta fundamental que los ingenieros del software dispongan de mecanismos cuantitativos para estimar la calidad de los diseños y la efectividad de los programas OO.

### **Características de las métricas orientadas a objetos.**

El software orientado a objetos es esencialmente distinto del software que se desarrolla utilizando métodos convencionales. Por esta razón, las métricas para sistemas OO deben concordar con las características que distinguen el software OO del software convencional. Donde Berard definió cinco características que cumplen con dicho planteamiento: localización, encapsulamiento, ocultamiento de información, herencia y técnicas de abstracción de objetos.

**Localización:** La localización es una característica del software que indica la forma que se concentra la información dentro de un programa.

**Encapsulamiento:** El empaquetamiento (o enlazado) de una colección de elementos. Entre los ejemplos de encapsulamiento de bajo nivel (software convencional) se cuentan los registros y matrices, y los subprogramas (procedimientos, funciones, subrutinas y párrafos) son mecanismos de nivel medio.

**Ocultamiento de información:** El ocultamiento de información suprime los detalles operativos de un componente de un programa. Tan sólo se proporciona la información necesaria para acceder a ese componente o a aquellos otros componentes que deseen acceder a él.

**Herencia:** La herencia es un mecanismo que hace posible que los compromisos de un objeto se difundan a otros objetos. La herencia se produce a lo largo de todos los niveles de la jerarquía de clases.

**Abstracción:** La abstracción es un mecanismo que permite al diseñador centrarse en los detalles esenciales de algún componente de un programa (tanto si es un dato como si es un proceso) sin preocuparse por los detalles de nivel inferior.

### 2.3 Métricas orientadas a clases.

La clase es la unidad principal de todo sistema OO. Entonces las medidas y métricas para una clase individual, la jerarquía de clases, y las colaboraciones de clases resultarán sumamente valiosas para un ingeniero de software que tenga que estimar la calidad de un diseño.

La clase encapsula a las operaciones (procesamiento) y a los atributos (datos), colabora con otras clases. Siendo todas estas características el factor fundamental de las métricas de CK.

Conjunto de métricas mas aplicadas en la actualidad por los desarrolladores de software, propuestas por: Chidamber y Kemener, para un total de seis métricas y conocidas por: “La serie de métricas CK”:

- Métodos ponderados por clase (MPC).
- Árbol de profundidad de herencia (APH).
- Número de descendiente (NDD).
- Acoplamiento entre clases objeto (ACO).
- Respuesta para una clase (RPC).
- Carencia de cohesión en los métodos (CCM).

Otra de las propuestas de métricas más aplicadas son las métricas propuestas por Lorenz y Kidd, separándolas en cuatro amplias categorías:

- Tamaño
- Herencia
- Valores internos
- Valores externos

### 2.4 Métricas aplicadas a la solución propuesta.

Dentro de las métricas utilizadas para los componentes que integran la solución propuesta se escogieron de la serie de métricas CK:

- **Árbol de profundidad de herencia (APH):** Está dado por la profundidad en la herencia de las clases heredadas de un nodo padre.
- **Número de descendiente (NDD):** Está dado por la cantidad de clases que heredan de un padre.
- **Respuesta para una clase (RPC):** Está dado por el número de relaciones de uso de una clase con otras.

Además se utilizaron dos métricas de la serie definida por Lorenz y Kidd:

- **Tamaño de Clases (TC):** Esta dado por el número de métodos asignados a una clase.
- **Número de Operaciones Redefinidas para una Sub-Clase (NOR):** Está dado por la cantidad de operaciones redefinidas en cada clase hija.

Atributos de calidad que se abarcan:

1. **Responsabilidad:** Consiste en la responsabilidad asignada a una clase en un marco de modelado de un dominio o concepto, de la problemática propuesta.
2. **Complejidad del diseño:** Consiste en la complejidad que posee una estructura de diseño de clases.
3. **Complejidad de implementación:** Consiste en el grado de dificultad que tiene implementar un diseño de clases determinado.

4. **Reutilización:** Consiste en el grado de reutilización que esté presente en una clase o estructura de clase, dentro de un diseño de software.
5. **Acoplamiento:** Consiste en el grado de dependencia o interconexión de una clase o estructura de clase, con otras, está muy ligada a la característica de Reutilización.
6. **Complejidad del mantenimiento:** Consiste en el grado de esfuerzo necesario a realizar para desarrollar un arreglo, una mejora o una rectificación de algún error de un diseño de software. Puede influir indirecta, pero fuertemente en los costes y la planificación del proyecto.
7. **Cantidad de pruebas:** Consiste en el número o el grado de esfuerzo para realizar las pruebas de calidad (Unidad) del producto (Componente, modulo, clase, conjunto de clases, etc.) diseñado.
8. **Nivel de Cohesión:** Consiste en el grado de especialización de las clases concebidas para modelar un dominio o concepto específico.
9. **Abstracción del diseño:** Consiste en la capacidad de modelar lo más cercano posible a la realidad un concepto o dominio determinado. (9)

Las métricas concebidas como instrumento para evaluar la calidad del diseño de los componentes Configuración de la gestión del multilinguaje y gestión de trazas y su relación con los atributos de calidad definidos en este trabajo son las siguientes:

➤ **Tamaño de Clases (TC):**

Atributo que afecta	Modo en que lo afecta
Responsabilidad	Un aumento del TC implica un aumento de la responsabilidad asignada a la clase.
Complejidad de implementación	Un aumento del TC implica un aumento de la complejidad de implementación de la clase.
Reutilización	Un aumento del TC implica una disminución en el grado de reutilización de la clase.

**Tabla 10: Tamaño de Clases (TC).**

➤ **Respuesta para una clase (RPC):**

Atributo que afecta	Modo en que lo afecta
Acoplamiento	Un aumento del RPC implica un aumento del acoplamiento de la clase.
Complejidad de mantenimiento	Un aumento del RPC implica un aumento de la complejidad del mantenimiento de la clase.
Reutilización	Un aumento del RPC implica una disminución en el grado de reutilización de la clase.
Cantidad de pruebas	Un aumento del RC implica un aumento de la cantidad de pruebas de unidad necesarias para probar una clase.

*Tabla 11: Respuesta para una clase (RPC).*

➤ **Árbol de profundidad de herencia (APH):**

Atributo que afecta	Modo en que lo afecta
Complejidad de mantenimiento	Un aumento del APH implica una disminución de la complejidad del mantenimiento de la clase.
Complejidad de diseño	Un aumento del APH implica una disminución de la complejidad del diseño de la clase.
Reutilización	Un aumento del APH implica un aumento en el grado de reutilización de la clase.

*Tabla 12: Árbol de profundidad de herencia (APH).*

➤ **Número de descendiente (NDD):**

Atributo que afecta	Modo en que lo afecta
Abstracción del diseño	Un aumento del NDD puede diluir la abstracción del diseño de clases.
Cantidad de pruebas	Un aumento del NDD implica un aumento de la cantidad de pruebas de unidad necesarias para probar una clase.
Reutilización	Un aumento del NDD implica un aumento en el grado de reutilización de la clase.
Cantidad de pruebas	Un aumento del NDD implica una disminución en el nivel de cohesión de la clase.

**Tabla 13: Número de descendientes (NDD).**

➤ **Número de Operaciones Redefinidas para una Sub-Clase (NOR):**

Atributo que afecta	Modo en que lo afecta
Abstracción del diseño	Un aumento del NOR implica que se ha logrado una buena definición de la abstracción del diseño de clases.
Cantidad de pruebas	Un aumento del NOR implica un aumento de la cantidad de pruebas de unidad necesarias para probar una clase.
Complejidad del mantenimiento	Un aumento del NOR implica un aumento de la complejidad del mantenimiento de la clase.

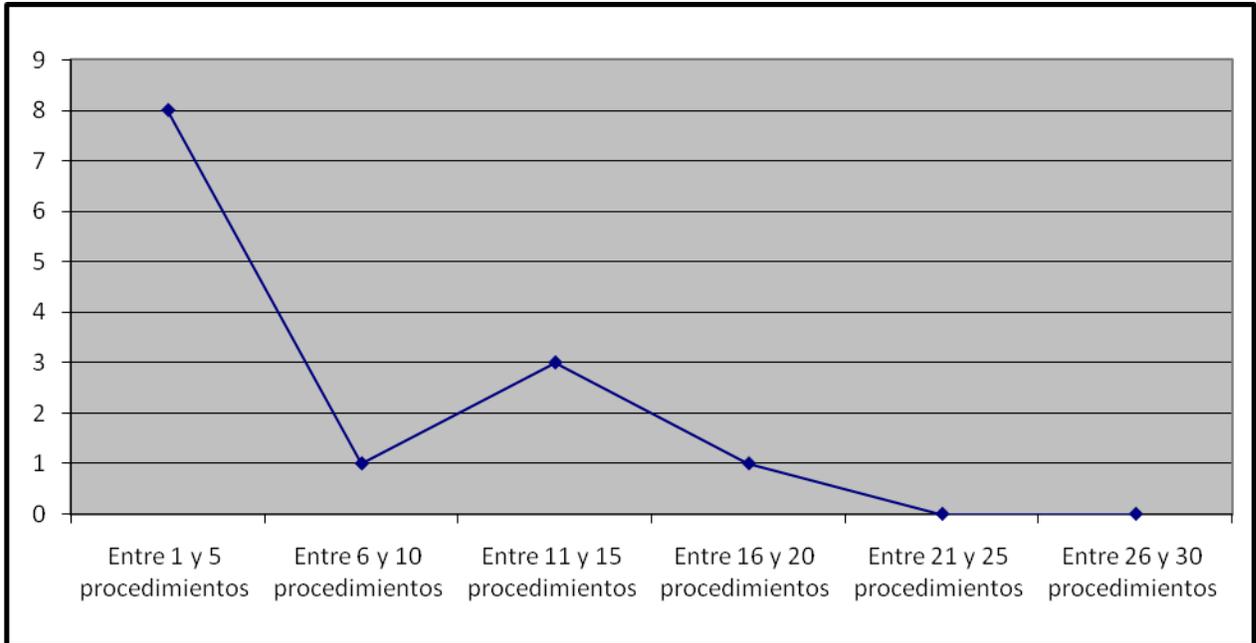
**Tabla 14: Número de Operaciones Redefinidas para una Sub-Clase (NOR).**

(9). A continuación se presentan los resultados obtenidos de la aplicación de los instrumentos de evaluación para medir las métricas anteriormente descritas, y una valoración de los mismos, que fueron realizados a la Capa de negocio de los componentes que conforman la solución propuesta.

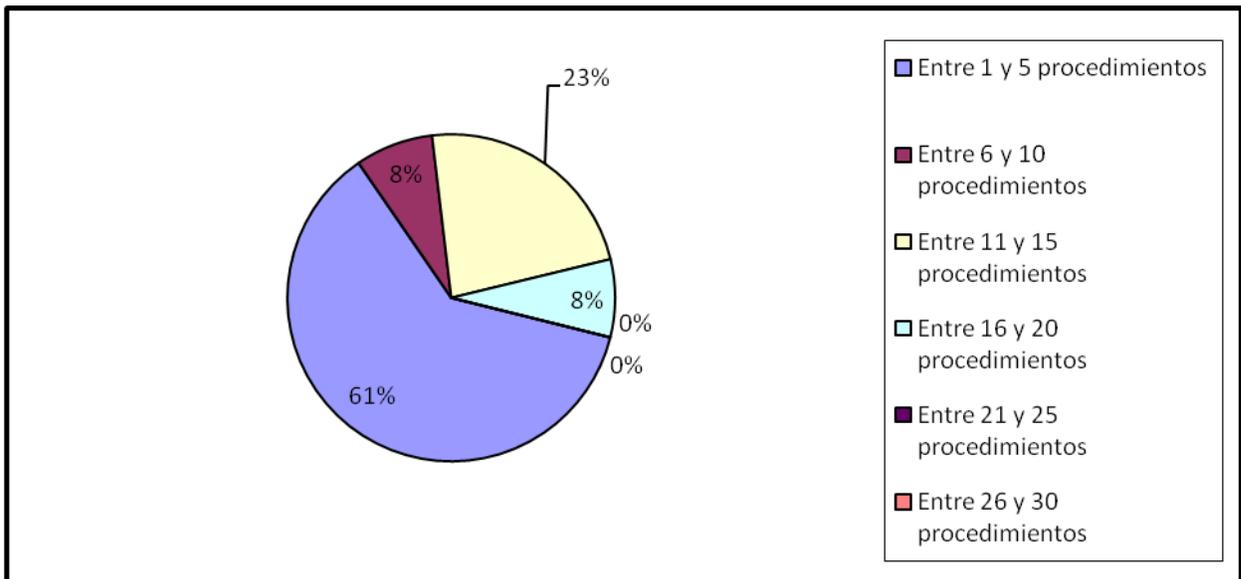
### **2.4.1 Resultados del instrumento de evaluación de la métrica Tamaño de Clases (TC).**

Ver instrumentos y tabla de resultados en (procede

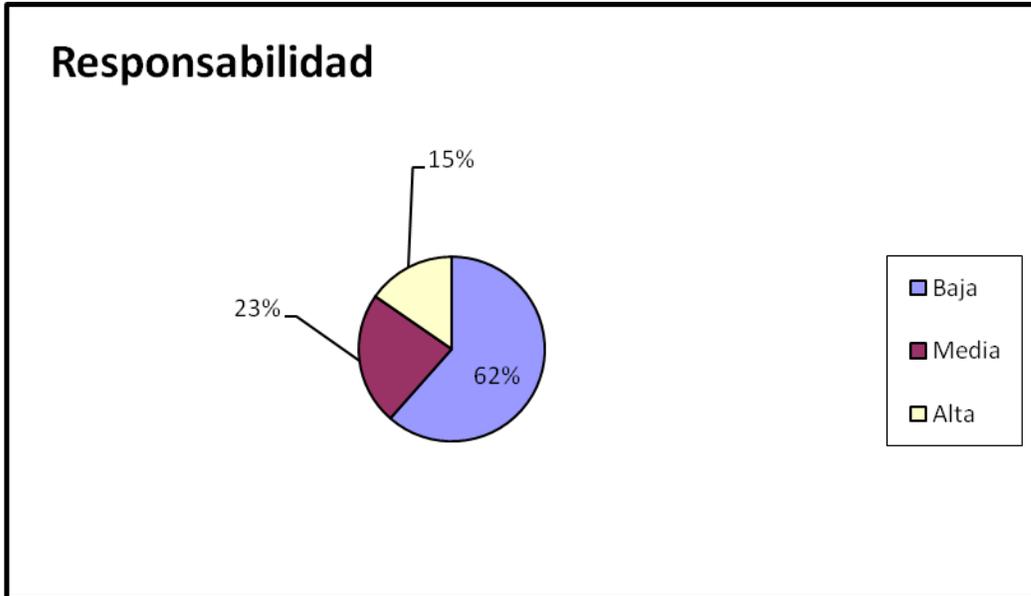
Anexo 8: Instrumento de medición de la métrica de Tamaño de Clases (TC).



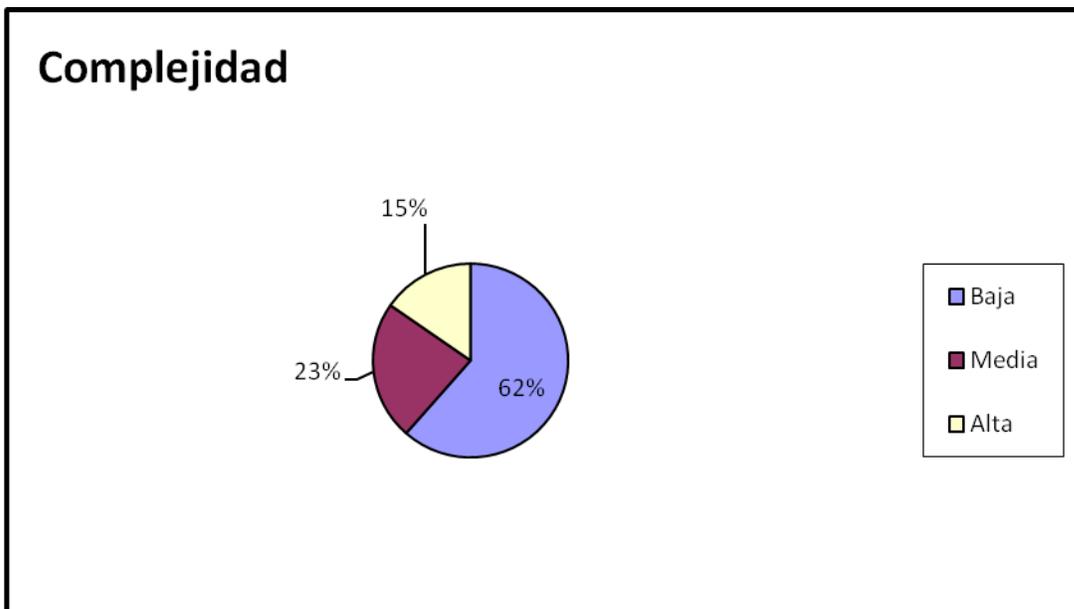
**Figura 23:** Representación de los resultados obtenidos en el instrumento agrupados en los intervalos definidos.



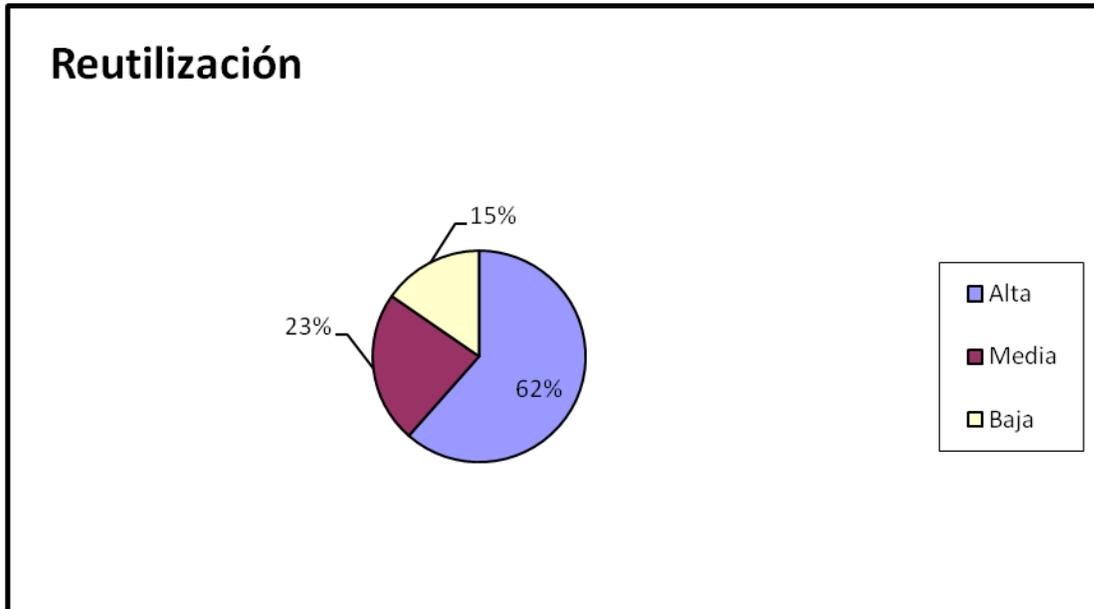
**Figura 24:** Representación en % de los resultados obtenidos en el instrumento agrupados en los intervalos definidos.



**Figura 75:** Representación de la incidencia de los resultados de la evaluación de la métrica TC en el atributo Responsabilidad.



**Figura 86:** Representación de la incidencia de los resultados de la evaluación de la métrica TC en el atributo Complejidad de Implementación.

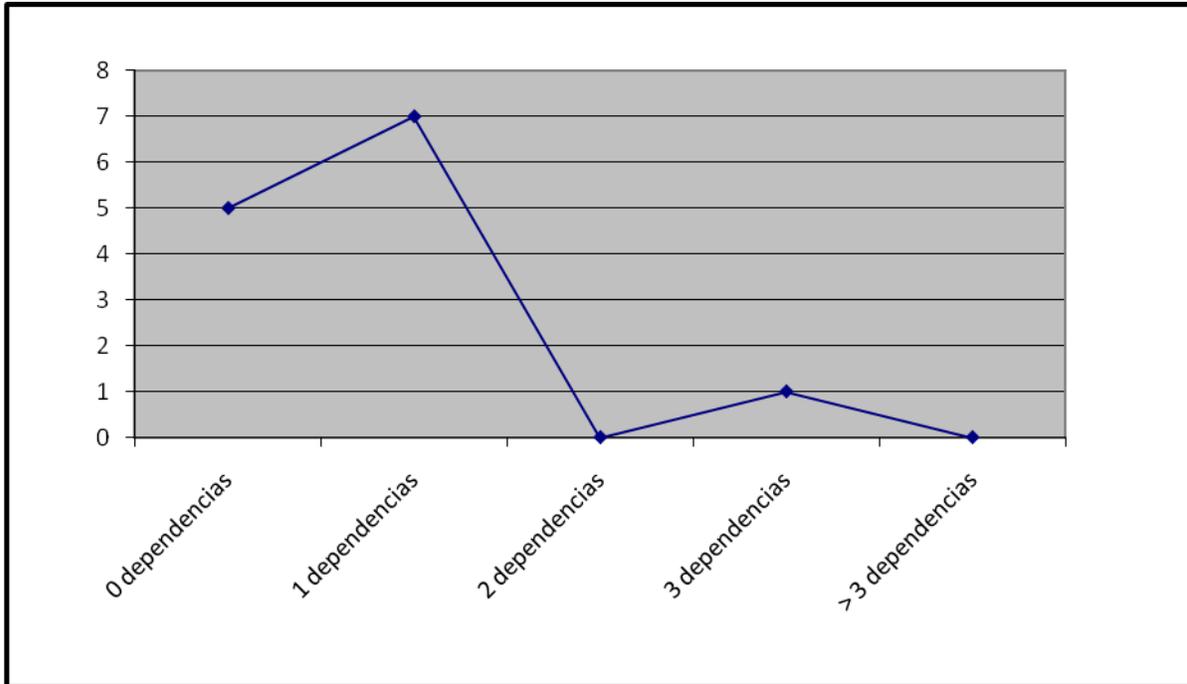


**Figura 27: Representación de la incidencia de los resultados de la evaluación de la métrica TC en el atributo Reutilización.**

Haciendo un análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica TC, se puede concluir que el diseño de los componentes configuración de la gestión del multilinguaje y la gestión de traza tienen una calidad favorable debido a que el 92% de las clases incluidas en estos componentes poseen menos cantidad de operaciones que la mitad del valor máximo registrado en las mediciones. Además el 84% de las clases poseen evaluaciones positivas en los atributos de calidad (Responsabilidad, Complejidad de Implementación y Reutilización).

### **2.4.2 Resultado del instrumento de evaluación de la métrica Respuesta para una clase (RPC).**

Ver instrumentos y tablas en (**Anexo 9: Instrumento de medición de la métrica respuesta para una clase (RPC).**)



**Figura 98:** Gráfica de los resultados de la evaluación de la métrica RPC agrupados por la tendencia de los valores.

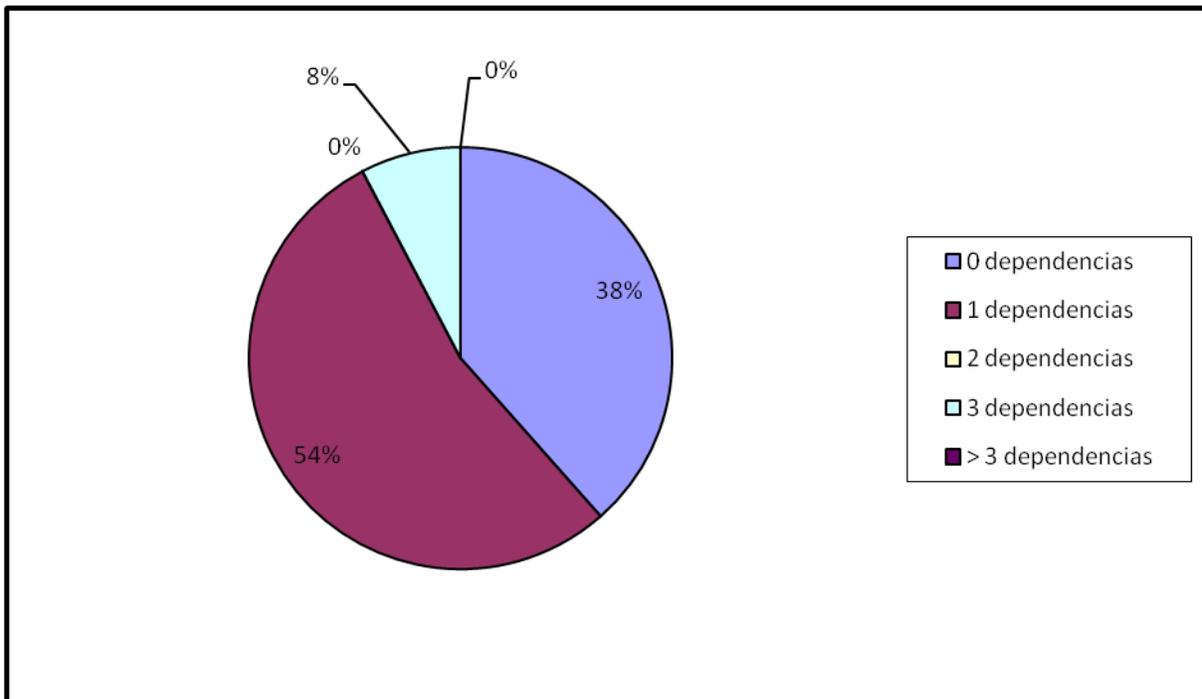


Figura 109: Representación en % de los resultados obtenidos en el instrumento agrupados en los intervalos definidos.

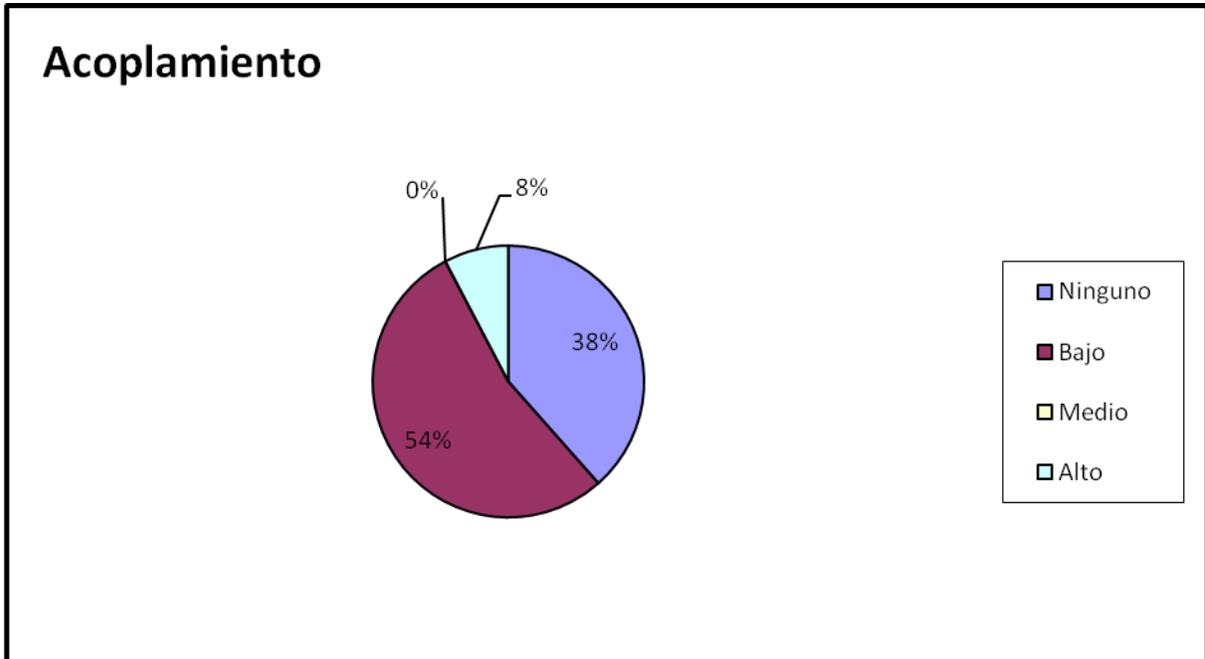


Figura 30: Representación de la incidencia de los resultados de la evaluación de la métrica RPC en el atributo Acoplamiento.

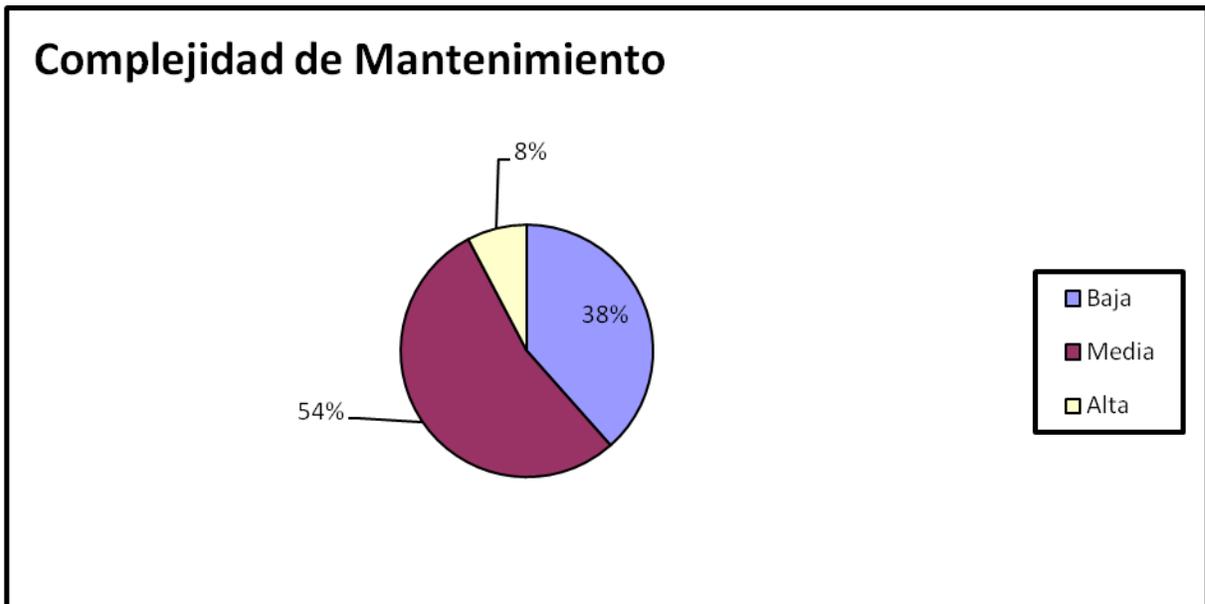
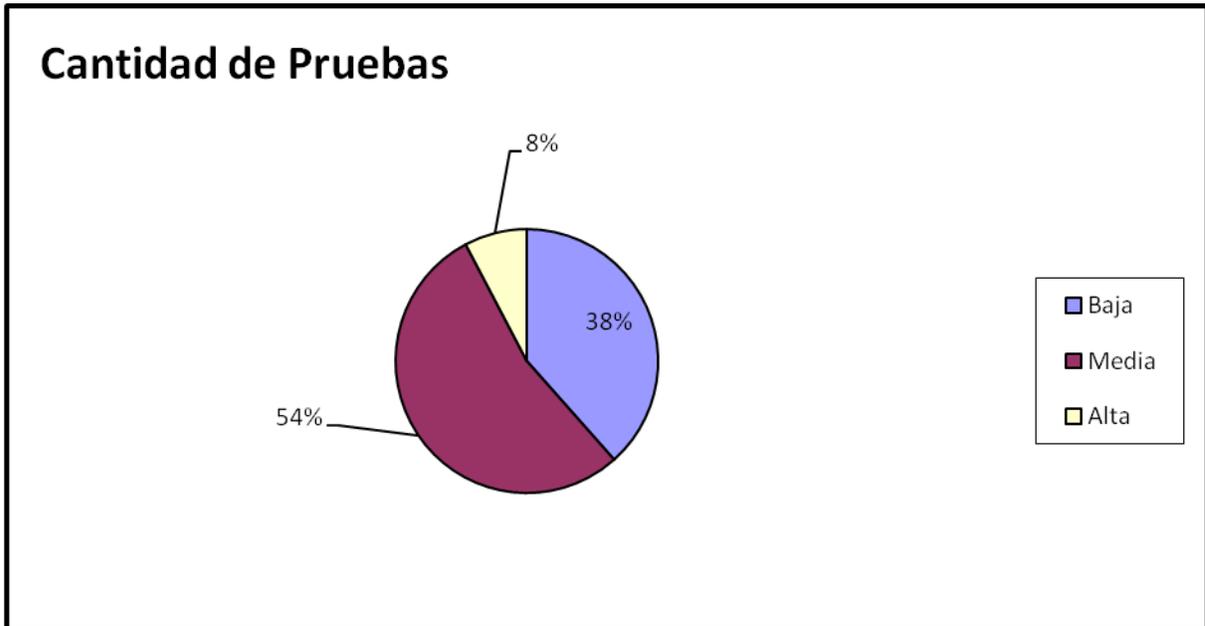
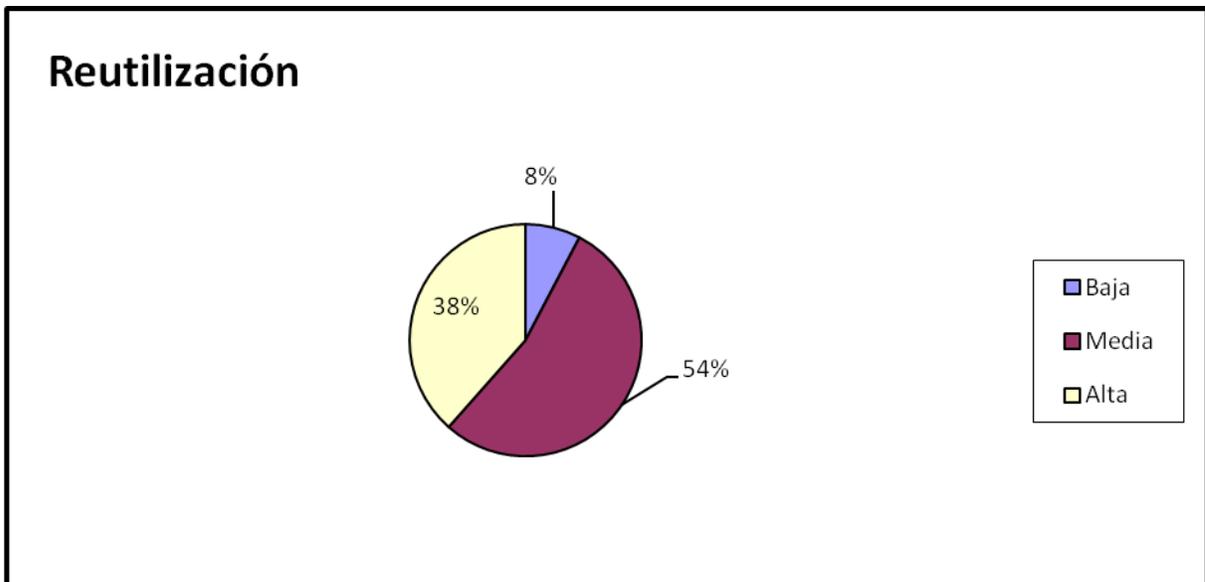


Figura 31: Representación de la incidencia de los resultados de la evaluación de la métrica RPC en el atributo Complejidad de Mantenimiento.



**Figura 32:** Representación de la incidencia de los resultados de la evaluación de la métrica RPC en el atributo Cantidad de Pruebas.



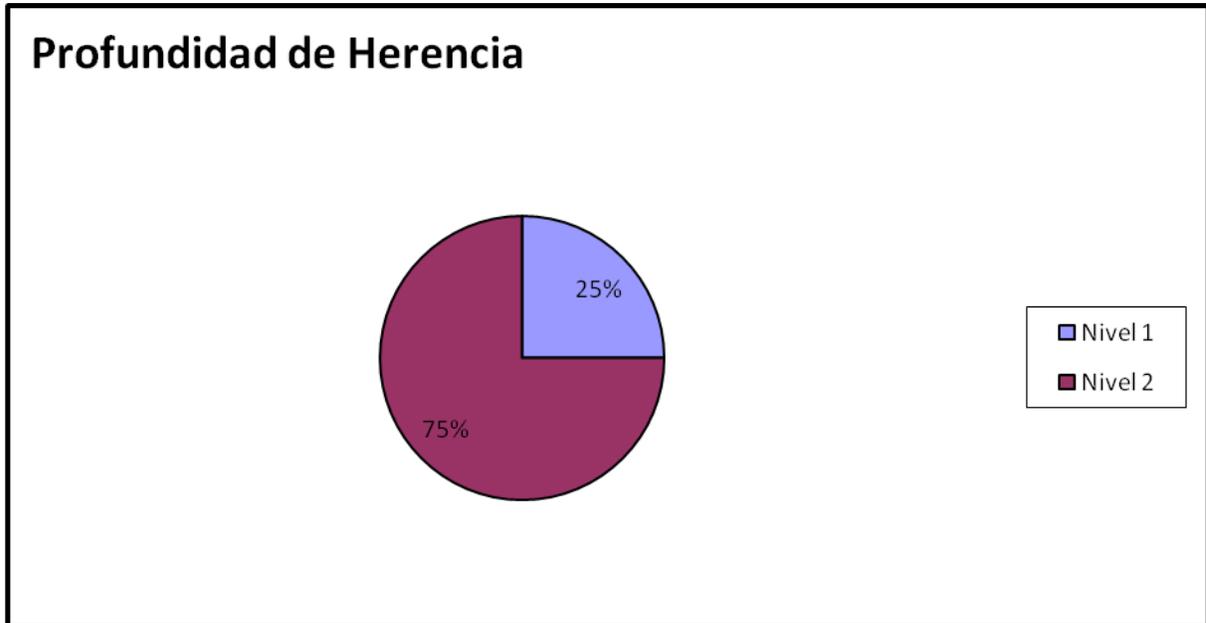
**Figura 33:** Representación de la incidencia de los resultados de la evaluación de la métrica RPC en el atributo Reutilización.

Haciendo un análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica RPC, se puede concluir que el diseño de los componentes configuración de la gestión del multilenguaje y gestión de traza tienen una buena calidad teniendo en cuenta que el 91% de las clases incluidas en estos componentes posee menos de 3 dependencias de otras clases. Además el 38% de las clases no poseen acoplamiento con otras y el 91% posee índices aceptables en cuanto a Acoplamiento. Así mismo los atributos de calidad Complejidad de Mantenimiento, Cantidad de Pruebas y Reutilización se comportan satisfactoriamente en un 91% de las clases.

### **2.4.3 Resultados del instrumento de evaluación de la métrica Árbol de profundidad de herencia (APH).**

Ver instrumentos y tablas de resultados en (

Anexo 10: Instrumento de medición de la métrica Árbol de profundidad de Herencia (APH.)

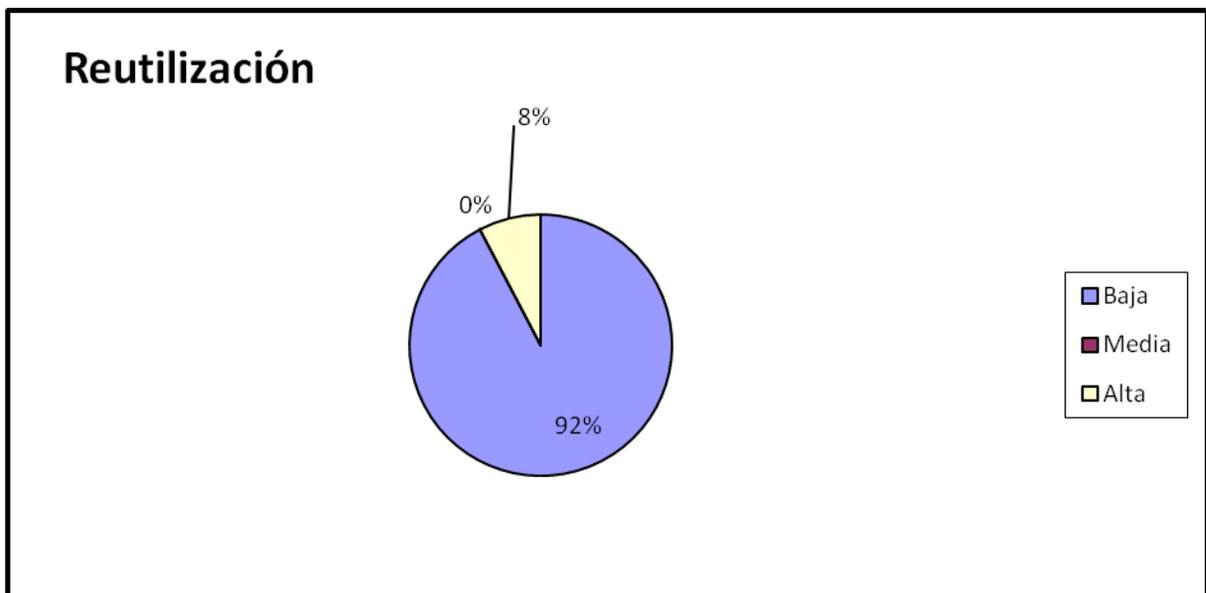


**Figura 34:** Representación en % de los resultados obtenidos en el instrumento agrupados por nivel.

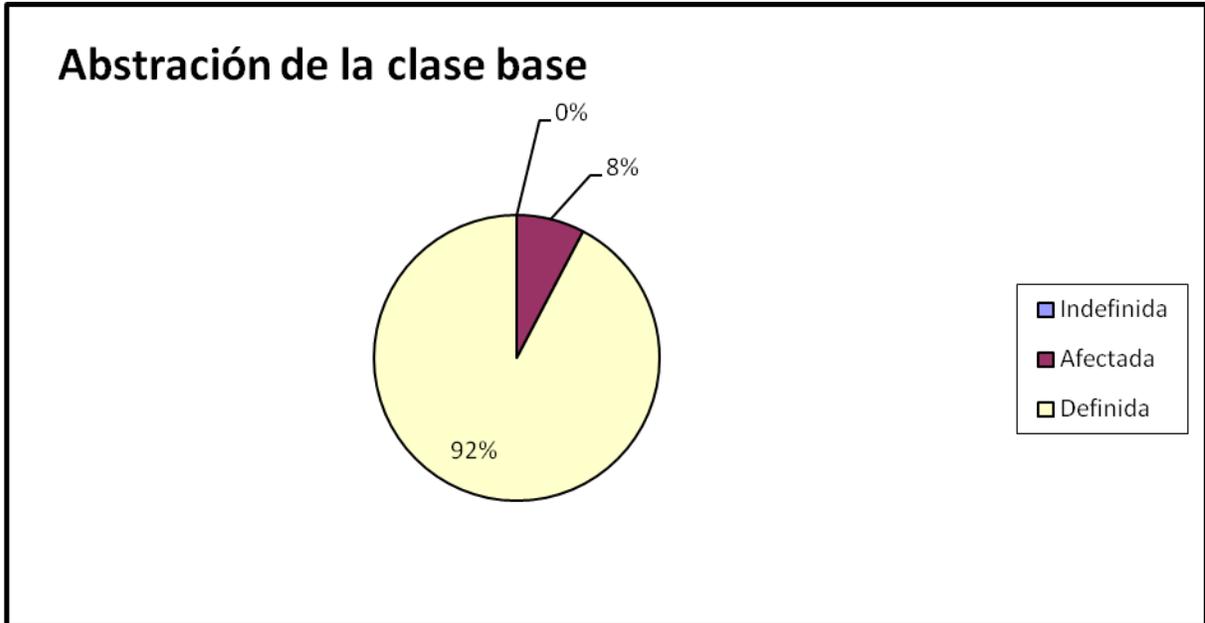
Haciendo un análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica APH, se puede concluir que el diseño de los componentes Configuración de la gestión del multilinguaje y gestión de traza tienen una calidad aceptable teniendo en cuenta que la profundidad de herencia presentes en los subsistemas desarrollados es siempre de 2, el cual es el valor mínimo posible. Analizando los atributos de calidad Complejidad de Mantenimiento, Complejidad de Diseño se puede decir que sin duda se tienen buenos índices debido a que solo se cuenta como máximo con un nivel 2 de profundidad en la herencia. Sin embargo el atributo Reutilización no posee indicadores positivos debido al bajo nivel de profundidad de herencia.

### 2.4.4 Resultados del instrumento de evaluación de la métrica Número de Descendientes (NDD).

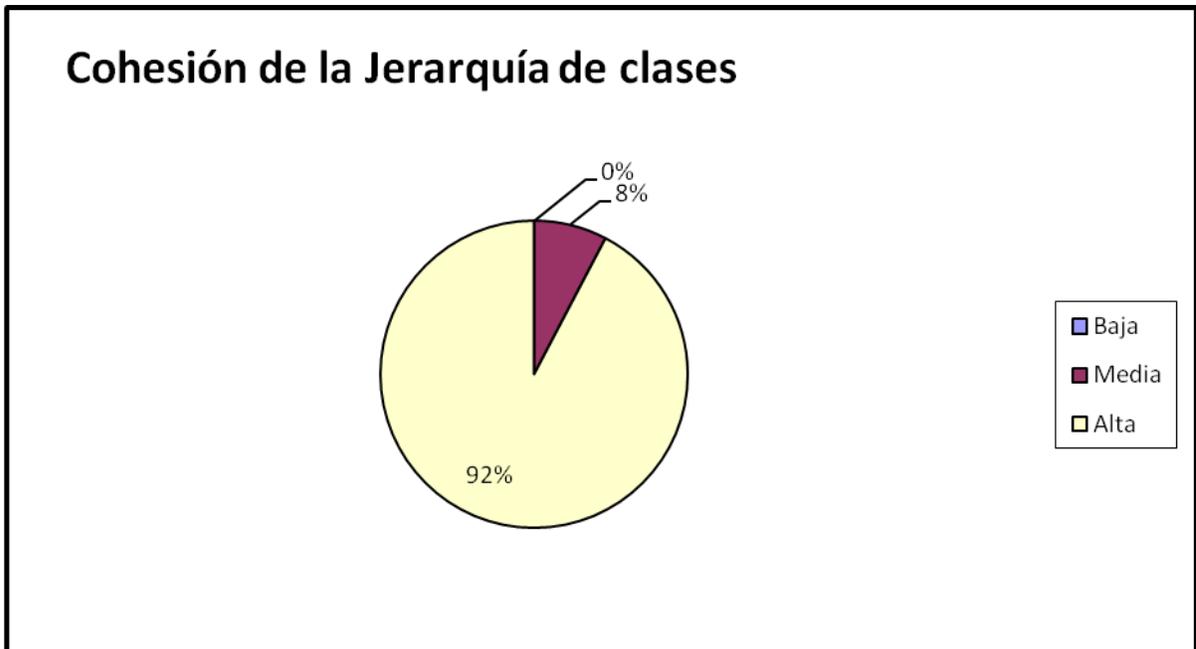
Ver instrumentos y tablas de resultados en (Anexo 11: Instrumento de medición de la métrica Número de Descendientes (NDD).)



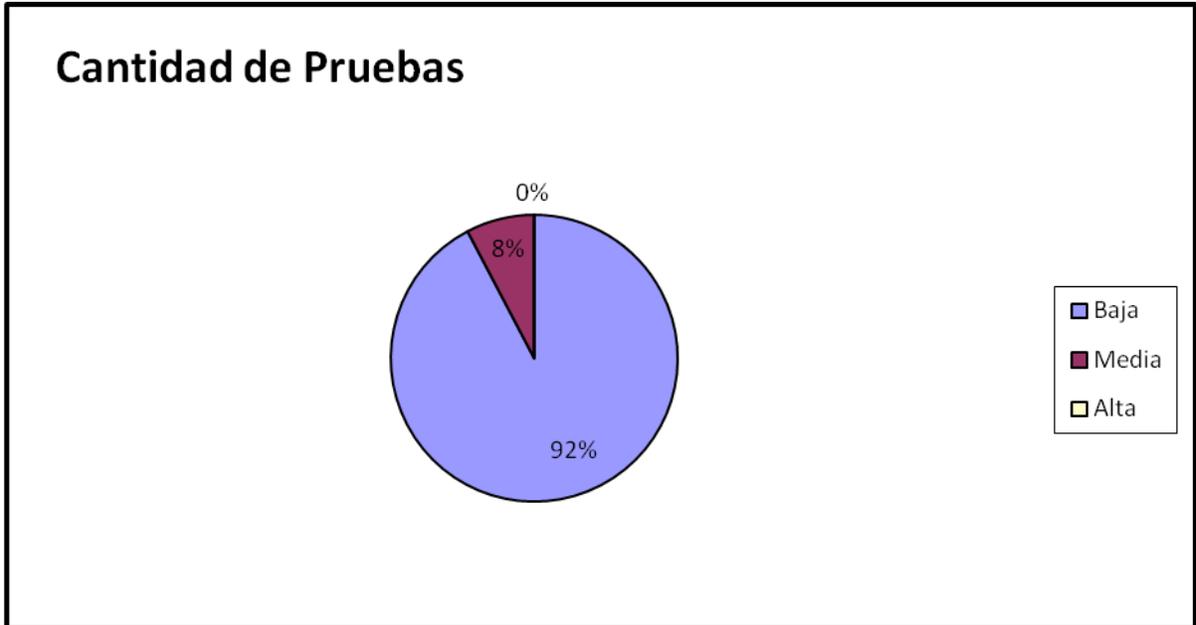
**Figura 35: Representación de la incidencia de los resultados de la evaluación de la métrica NND en el atributo Reutilización.**



**Figura 116:** Representación de la incidencia de los resultados de la evaluación de la métrica NDD en el atributo Abstracción de la clase base.



**Figura 37:** Representación de la incidencia de los resultados de la evaluación de la métrica NDD en el atributo Cohesión de la Jerarquía de clases.

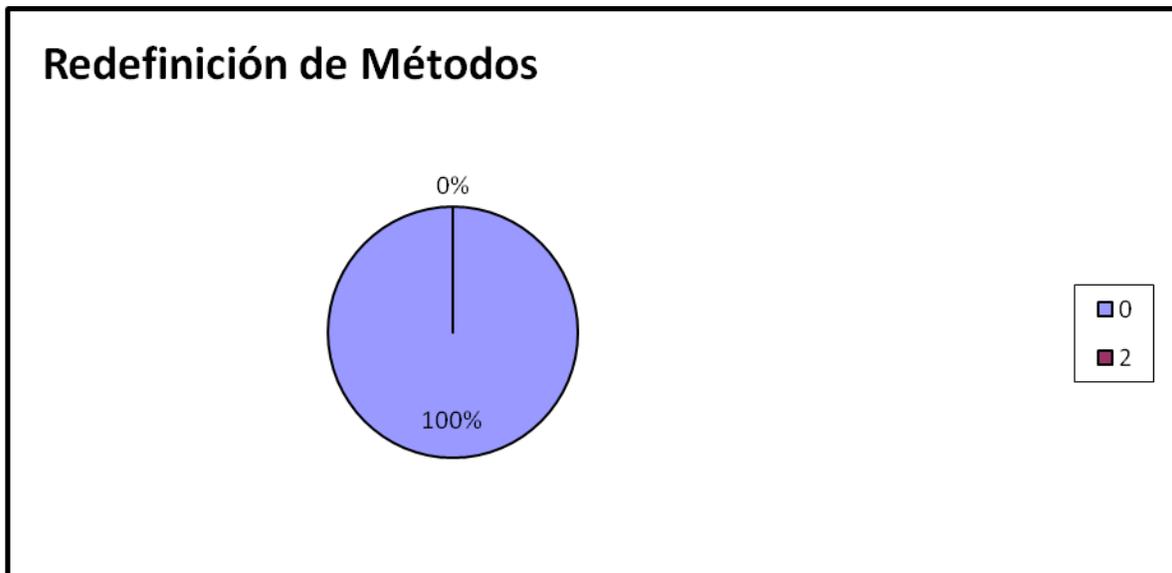


**Figura 38:** Representación de la incidencia de los resultados de la evaluación de la métrica NDD en el atributo Cantidad de Pruebas.

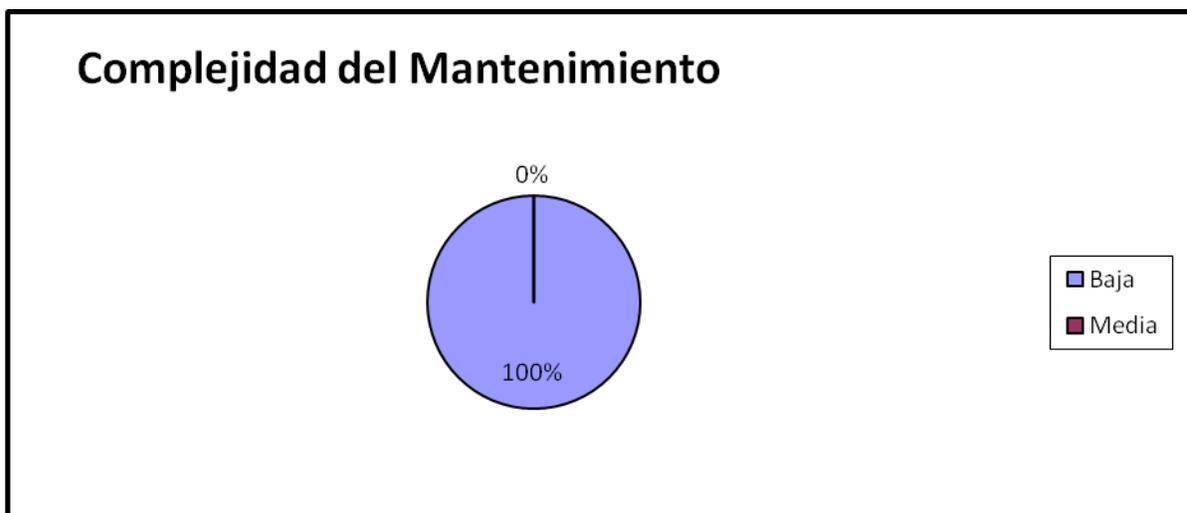
Haciendo un análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica NDD, se puede concluir que el diseño de los componentes Configuración de la gestión del multilinguaje y gestión de traza tienen una calidad aceptable teniendo en cuenta que solo se emplea la herencia en casos bien identificados a partir de las necesidades del negocio o del diseño. Solo una clase posee descendientes y la cantidad de descendientes supera la cantidad de 3. Valorando el atributo de calidad Reutilización se puede decir que por lo antes explicado los índices de Reutilización se mantienen bajos representados por un 92%. En cuanto al atributo Abstracción de la clase base se muestra como existe una tendencia a la conservación de la abstracción. Tanto para la Cohesión de la jerarquía de clases como para el atributo Cantidad de Pruebas los índices son positivos favoreciendo esto al diseño.

### 2.4.5 Resultados del instrumento de evaluación de la métrica Número de Operaciones Redefinidas para una Sub-Clase (NOR).

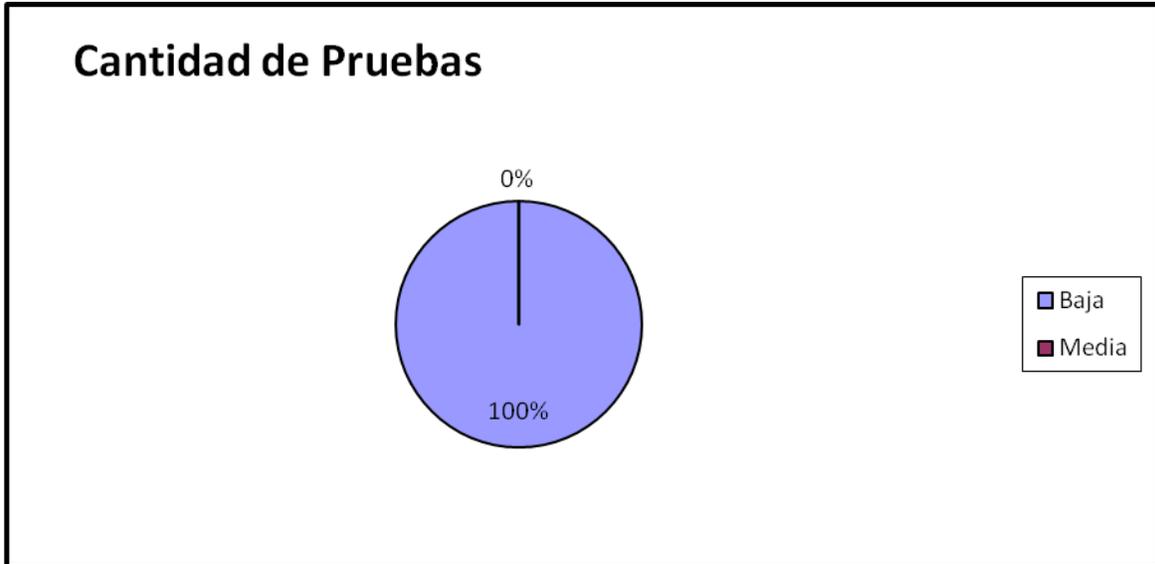
Ver instrumentos y tabla de resultados en (Anexo 11: Instrumento de medición de la métrica Número de Operaciones Redefinidas para una Sub-Clase (NOR).)



*Figura 39: Representación en % de los resultados obtenidos en el instrumento agrupados en los valores existentes.*



*Figura 40: Representación de la incidencia de los resultados de la evaluación de la métrica NOR en el atributo Complejidad del Mantenimiento.*



*Figura 41: Representación de la incidencia de los resultados de la evaluación de la métrica NOR en el atributo Cantidad de Pruebas.*



*Figura 42: Representación de la incidencia de los resultados de la evaluación de la métrica NOR en el atributo Violación de la Abstracción representada por la superclase.*

Haciendo un análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica NOR, se puede concluir que el diseño de los componentes Configuración de la gestión del multilinguaje y gestión de traza tienen una calidad aceptable teniendo en cuenta que ninguna de las clases incluidas en el análisis (solo relacionadas con la herencia) redefinen algún método heredado. Además los indicadores se comportan de forma adecuada para los atributos de calidad Complejidad del Mantenimiento, Cantidad de Pruebas, Violación de la Astrictión representada por la superclase.

### **Matriz de cubrimiento o matriz de inferencia de indicadores de calidad.**

La matriz de cubrimiento o matriz inferencia de indicadores de calidad es una representación estructurada de los atributos de calidad y métricas utilizadas en el capítulo anterior para evaluar la calidad del diseño de los componentes que integran la solución propuesta. La misma permite conocer si el resultado obtenido de la relación atributo/métricas para cada componente es positivo o negativo. Llevando estos resultados a una escalabilidad numérica donde, si los resultados son positivos tendrá un valor de 1, si son negativos de 0 y si no existe relación alguna se tomara como nula (-). Una vez completado los datos de dicha relación se realiza un cálculo donde se promedia la sumatoria de los valores obtenidos de un atributo por cada métrica evaluado, y la división de dicha sumatoria por la cantidad de métricas evaluada (solo se promedian las que arrojan un resultado, las nulas no).

## CAPÍTULO 3: ANÁLISIS DE LA SOLUCIÓN

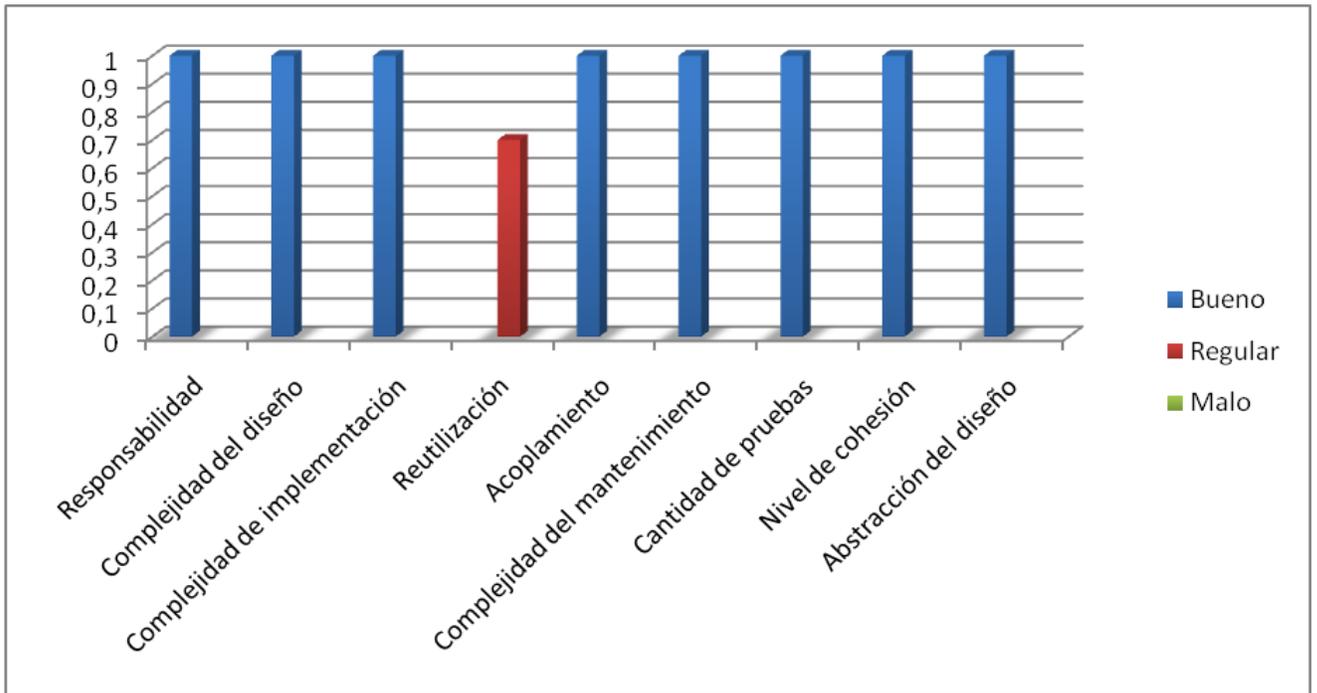
Este valor es el que va a tener el atributo dentro de una tabla que medirá si los atributos fueron buenos, regulares o malos. Se lograron los siguientes resultados:

Atributos/Métricas	TOC	RPC	APH	NDD	NOR	Promedio
Responsabilidad	1	-	-	-	-	1
Complejidad del diseño	-	-	1	-	-	1
Complejidad de implementación	1	-	-	-	-	1
Reutilización	1	1	-	0	-	0,7
Acoplamiento	-	1	-	-	-	1
Complejidad del mantenimiento	-	1	-	-	1	1
Cantidad de pruebas	-	1	-	1	1	1
Nivel de cohesión	-	-	-	1	-	1
Abstracción del diseño	-	-	-	1	1	1

*Tabla 5: Resultados evaluados de la relación Atributos/Métricas por cada componente que integran la solución.*

Calidad	Evaluación
Malo	$\leq 0.4$
Regular	$>0.4$ y $\leq 0.7$
Bueno	$>0.7$

*Tabla 6: Rango de valores para la evaluación técnica de los atributos de calidad evaluados por cada métrica.*



**Figura 43:** Gráfica de los resultados obtenidos de los atributos de calidad evaluados en las métricas.

Luego de los resultados estadísticos plasmados en la matriz de cubrimiento o inferencia de indicadores de calidad se puede valorar que la solución propuesta mantiene una factible funcionalidad, manteniendo valores positivos para cada uno de los atributos de calidad relacionados con las métricas evaluadas en dichos componentes.

### **Conclusiones.**

En el presente capítulo se abarcaron los principales métodos para obtener el nivel de calidad de la solución propuesta. Mediante instrumentos inspirados en métricas para la calidad del diseño y basándose sobre los atributos de calidad trazados para llevar a cabo una medición estricta de cada uno de los componentes, los cuales permitieron afirmar que el diseño realizado se puede valorar de aceptable.

### CONCLUSIONES GENERALES

La investigación realizada en el presente trabajo de diploma permite concluir que la gestión manual de los archivos de configuración de la gestión del multilinguaje y la realización de forma lenta y personalmente de la gestión de trazas trae consigo consecuencias desfavorables para el rendimiento y la productividad en Sistema Integral de Gestión Cedrux.

A raíz de lo antes planteado se arroja la siguiente conclusión: necesidad de desarrollar una herramienta propia que respondiera a los requerimientos del proyecto, pues del estudio referencial se concluyó que no existían herramientas que llevaran a cabo estos procesos.

Se diseñó e implementó dos componentes, para lo cual se utilizaron patrones arquitectónicos y estándares de diseño que propiciarían resultados positivos en materia de cohesión, acoplamiento, reutilización, calidad y facilidad de mantenimiento, lo que permitió obtener los componentes antes mencionados capaz de manejar el multilinguaje del marco de trabajo de Cedrux de una forma exitosa y consistente al igual que la gestión de trazas de sistema en cuestión.

Las soluciones fueron probadas por los métodos validados durante la investigación en el segundo capítulo respondiendo a los escenarios conformados por los requisitos funcionales. De esta forma la solución aportada cumple con las características de calidad exigidas por el proyecto ERP-Cuba.

Para darle cumplimiento al último objetivo específico y base a la afirmación anterior se realiza una validación del diseño mediante la utilización de instrumentos de medición que se inspiraron en métricas para la calidad del diseño. Los resultados arrojados permitieron concluir que el diseño presentaba valores positivos en indicadores de calidad tales como Reutilización, Facilidad de Mantenimiento, Complejidad del Diseño, Complejidad de Implementación, Cohesión, Acoplamiento, Cantidad de pruebas, entre otros. Esto apoya la afirmación de que el diseño desarrollado se puede considerar como satisfactorio.

### RECOMENDACIONES

Se considera importante distinguir las siguientes recomendaciones.

- Extender el uso de la aplicación a plataformas libres, basadas en GNU/Linux, para lo cual se recomienda compilar la aplicación utilizando el proyecto Mono, el cual es una herramienta libre compatible con el framework de .NET.
- Proveer la aplicación de otros componentes para la gestión del marco de trabajo de Cedrux, contribuyendo así al dinamismo en la producción.

Se recomienda finalmente el trazado de una estrategia que responda a la adaptación de los desarrolladores para el uso estos dos componentes que formaran parte del IDE, elevando de esta forma los indicadores de rendimiento y productividad en las líneas de desarrollo y módulos de Cedrux.

### BIBLIOGRAFÍA

1. **Estr@tegia Magazine.** GestioPolis.com. *GestioPolis.com*. [En línea] 04 de 02 de 2008. [Citado el: 19 de Abril de 2009.] <http://www.gestiopolis.com/administracion-estrategia/estrategia/que-es-erp.htm>.
2. BuenMaster.com. *BuenMaster.com*. [En línea] 16 de 08 de 2007. [Citado el: 16 de 04 de 2009.] <http://buenmaster.com/?a=554>.
3. Glosario.net. *Glosario.net*. [En línea] 27 de 10 de 2006. [Citado el: 26 de Abril de 2009.] <http://tecnologia.glosario.net/terminos-tecnicos-internet/ide-860.html>.
4. **Rumbaugh, Jacobson y Booch.** *The unified Modeling Language Reference Manual*. s.l. : Addison-Wesley, 2002.
5. **Rumbaugh, J, Jacobson, I. y Booch, G.** *The Unified Modeling Language. Reference Manual*. Massachusetts : ADDISON-WESLEY. 568, 1998.
6. **Daedalus.** Diseño de sistemas. [En línea] 2006. <http://www.daedalus.es/AreasISDiseno-E.php>.
7. **Microsoft.** *Diseño del software*. 2007.
8. *Medir el proceso de control de configuración, ¿Una utopía para la Industria Nacional de Software?* **Estrada, A.F. y Estévez, I.P.** 2003.
9. **Lazo, Ochoa y Yzquierdo, Raykenler.** *El modelo de diseño del sistema HyperWeb. Módulos de Tratamiento Farmacológico y Configuración*. Habana : s.n., 2007.
10. Symphony. *Symfony*. [En línea] [Citado el: 13 de Abril de 2009.] [http://www.symfony-project.org/jobeeet/1\\_2/Propel/es/19](http://www.symfony-project.org/jobeeet/1_2/Propel/es/19).
11. **JSON.com.** JSON. JSON. [En línea] [Citado el: 21 de Abril de 2009.] <http://www.json.org/json-es.html>.

12. **Sun Microsystems, Inc.** *J2EE Connector Architecture Specification, Version 1.0.* 2007.
13. **Alur, D, Crupi, J y Malks, D.** *Core J2EE Patterns, Best Practices and Design Strategies.* s.l. : Prentice Hall, 2004.
14. **Barrios, Nuñez. Fabricio.** *Investigación de la plataforma J2EE y su aplicación práctica.* Santiago, Chile. : s.n., 2005.
15. **Campos, S.G y Martinez, L.F.** *Programación Extrema: Prácticas, Aceptación y Controversia Volume I.* 2006.
16. **Canos, Jose. H y Penades, Carmen.** *Metodologías Agiles en el Desarrollo de Software.* 2006.
17. **Avila, S.J.V.** *Introducción a Microsoft Solutions Framework.* 2005.
18. **Microsoft Corporation. MSDN Community.** [En línea] 2009. <http://www.msdn.com/netframework>.
19. **Microsoft Corporation.** *Microsoft Solution Framework.* 2006.
20. *Reutilización del Software a partir de Requisitos Funcionales en el Modelo Mecano: Comparación de escenarios.* **López, Oscar, Laguna, Miguel. Ángel y Marqués, José. Manuel.** 2005, Software Privileges Magazine, págs. 41-50.
21. *Reutilización de Requisitos de usuario.* . **García, Peñalvo. Francisco José y López, Oscar.** 2002, Revista Comlombiana de Computación .
22. **Valdés, Ing. Jorge Luis.** Info.DvaT. [En línea] Marzo de 2008. <http://jorgevaldes.overblog.com/>.
23. **Wikipedia Foundation .Inc.** Wikipedia, la enciclopedia libre - Caso de prueba. *Wikipedia, la enciclopedia libre - Caso de prueba.* [En línea] 12 de febrero de 2009. [Citado el: 15 de abril de 2009.] <http://es.wikipedia.org>.
24. **IBM Corporation.** *IBM Rational Unified Process.* Nueva York : United States of America, 2003.

25. **Moreno, Germán Arroyo.** L.S.I. Departamento de lenguajes y sistemas informáticos. Universidad de Granada. *L.S.I. Departamento de lenguajes y sistemas informáticos. Universidad de Granada.* [En línea] 25 de abril de 2006. [Citado el: 15 de abril de 2009.] <http://lsi.ugr.es>.
26. **Oficina Nacional del Gobierno Electrónico e Informática - ONGEI.** Guía técnica sobre evaluación de software en la administración pública. *El Peruano.* 28 de mayo de 2004, pág. 32.
27. (<http://www.taringa.net/posts/apuntes-y-monografias/1508035/Framework.html>).
28. **Deepinder, B. Mooney, T. Garcia, J.** An integrative framework for the assimilation of Enterprise Resource Planning . *An integrative framework for the assimilation of Enterprise Resource Planning* . s.l. : Spring , 2004, Vols. Vol. 44, Iss. 3; pp. 81, 10. .
29. *A Comprehensive Framework for Classifying the Benefits of ERP Systems.* **SHANG, S.** 2000.
30. **Pressman, R.** *Ingengería del software: Un enfoque práctico.* McGraw Hill : s.n., 2000.
31. **Jacobson, I., Booch, G. y Rumbaugh, J.** *The Unified Software Development Process.* s.l. : Addison-Wesley, 2000.
32. **Fowler, Martin.** Inversion of Control. [En línea] 26 de June de 2005. <http://www.martinfowler.com>.
33. **Castle, Proyect.** [www.castleproject.org](http://www.castleproject.org). *Castle Proyect.* [En línea] 2009. <http://www.castleproject.org/container/gettingstarted/index.html>.
34. **Eini, Oren.** Inversion of Control and Dependency Injection: Working with Windsor Container. *MSDN.* [En línea] November de 2006.
35. **Garlan, D. Shaw, M.** *An Introduction to Software Architecture.* 1994.
36. *Aspect-Oriented Programming Enables Better Code Encapsulation and Reuse.* **Dharma, S., Fell, S. y Sells, C.** s.l. : MSDN Magazine, 2002, Vol. March 2002.

37. **Proyect, Castle.** [www.castleproject.org](http://www.castleproject.org). *Castle Proyect*. [En línea] 2009. <http://www.castleproject.org/container/gettingstarted/index.html>.
38. **Equipo Arquitectura del ERP Cedrux.** *Especificación Técnica para el marco de la arquitectura*. 2008.
39. *A Comprehensive Framework for Classifying the Benefits of ERP Systems.* **Shang, S.** 2000.
40. *Conferencias sobre Metodología de la Investigación.* . **Vargas, A.** 2005, Impr. Universitaria. La Habana., págs. 28-44.
41. **Buschmann, F.** *Pattern-Oriented Software Architecture*. s.l. : John Wiley & Sons., 2006.
42. **Marinescu, Floyd.** *EJB Design Patterns*. s.l. : Jhon Wiley & Sons, 2004.
43. **Greca, Pablo La.** Bloque de conocimiento de de Epidata Consulting SRL: Spring IoC Container. *Bloque de conocimiento de de Epidata Consulting SRL: Spring IoC Container*. [En línea] 17 de Octubre de 2007. [Citado el: 5 de abril de 2009.] <http://www.epidataconsulting.com>.
44. **Figueras, Juan Medín Piñeiro y Antonio García.** *Hacia una arquitectura con JavaServer Faces, Spring, Hibernate y otros frameworks*. Sevilla : Ediciones Alfar S.A., 2006.
45. **Booch, Grady.** *Object-Oriented Design with Applications*. The Benjamin : Cummings Publishing Company., 1991.
46. **Adison, S.** *The Art of Software Architecture: Design methods and techniques*. Nueva York : Adison Wiley, 2003.
47. **ARREGUI, J. J. O.** *Revisión Sistemática de Métricas de Diseño Orientado a Objetos*. Madrid, España. : Universidad Politécnica de Madrid, Facultad de Informática., Septiembre 2005.

**ANEXOS****Anexo1: Plantilla de encuesta realizada a los distintos roles que integran el proyecto ERP.**

Encuesta para la valoración de la configuración manual de los archivos de configuración de la gestión del multilinguaje y la realización lenta y personalmente de la gestión de trazas del Sistema Integral de Gestión Cedrux.

Estimado usuario:  
Con el objetivo de ayudarlos a hacer más eficiente su trabajo, estamos realizando un diagnóstico sobre los problemas relacionados con configuración de los archivos de configuración de la gestión del multilinguaje y el proceso gestión de trazas del Sistema Integral de Gestión Cedrux. Le pedimos responda con la mayor veracidad el siguiente cuestionario:

**1. Señale el rol que ocupa dentro de su equipo de trabajo.**

- Jefe de Línea**
- Analista**
- Planificador**
- Arquitecto**
- Desarrollador**
- Otro (Por favor especifique)**

2. Diga si usted ha utilizado los archivos de configuración de la gestión del multilinguaje. (json)

Si  No

3. Diga si usted ha participado en el proceso de gestión de trazas en su línea.

Si  No

4. ¿Se encuentra satisfecho con la configuración manual de la gestión del multilinguaje?

Si  No

5. ¿Se encuentra satisfecho con la realización de forma personal y lentamente de la gestión de trazas?

Si  No

6. Señale las dificultades encontradas durante la configuración manual de de la gestión del multilinguaje.

- Perdida de tiempo.
- Introducción de errores.
- Incremento del tiempo de máquina.
- Baja productividad del equipo de trabajo.
- Disminución de la moral del equipo de trabajo.

Otro (Por favor especifique)

**7. Señale las dificultades encontradas durante la realización de forma personal y lentamente de la gestión de trazas**

Atraso con el cronograma de solución de las trazas.

Insatisfacción en la realización del proceso.

Otro (Por favor especifique)

**8. ¿Considera necesario el desarrollo de un componente para la configuración de los archivos que gestionan el multilinguaje del Sistema Integral de Gestión Cedrux?**

Si  No

**9. ¿Considera necesario el desarrollo de un componente para gestión de trazas del Sistema Integral de Gestión Cedrux?**

Si  No

**Anexo 2: Requisitos funcionales del componente configuración de la gestión del multilinguaje.**

**a) Gestionar proyecto.**

RF -1: Crear nuevo proyecto.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	Proyecto.	Nombre, ubicación.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	No procede.	No procede.
<b>Descripción</b>	Permite crear un proyecto nuevo.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Una vez que este RF se lleve a cabo se pueden crear los ficheros, y el árbol de navegación se actualiza con el nombre del proyecto en cuestión.	
<b>Post-requisito</b>	No procede.	

**Tabla 17: RF-1 Crear nuevo proyecto del escenario Gestionar proyecto.**

RF -2 Cargar un proyecto existente.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Tiene que existir al menos un proyecto previamente creado.	No procede.
<b>Descripción</b>	Permite gestionar los datos implícitos del proyecto, como es el caso de los ficheros con sus respectivos componentes.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Una vez que este RF se lleve a cabo se actualiza el árbol de navegación con la información contenida del proyecto y la tabla que se encuentran en la interfaz principal.	
<b>Post-requisito</b>	No procede.	

**Tabla 18: RF-2 Cargar un proyecto existente del escenario Gestionar proyecto.**

RF -3: Guardar Proyecto como...

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	No procede.	No procede.
<b>Descripción</b>	Permite guardar un proyecto en una dirección diferente.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Una vez que este RF se lleve a cabo se guardan los datos del proyecto en la dirección escogida.	
<b>Post-requisito</b>	No procede.	

**Tabla 19: RF-3 Guardar Proyecto como del escenario Gestionar proyecto.**

RF -4: Guardar Proyecto.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	No procede.	No procede.
<b>Descripción</b>	Permite guardar un proyecto.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Una vez que este RF se lleve a cabo se guardan los datos del proyecto	
<b>Post-requisito</b>	No procede.	

*Tabla 20: RF-4 Guardar Proyecto del escenario Gestionar proyecto.*

**b) Gestionar fichero.**

RF -1: Crear nuevo fichero .json.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	Fichero.	Nombre, ubicación
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe haber creado un proyecto.	Escenario a) RF-1.
<b>Descripción</b>	Permite crear un fichero .json.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Una vez que este RF se lleve a cabo se podrán crear los componentes que estarán incluidos en dicho fichero. El árbol de navegación se actualiza con el nombre del fichero en cuestión.	
<b>Post-requisito</b>	No procede.	

**Tabla 21: RF-1 Crear nuevo fichero .json del escenario Gestionar fichero.**

RF -2: Mostrar vista previa del fichero .json.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe haber creado un fichero.	R1
<b>Descripción</b>	Se muestra una vista previa del fichero una vez construido.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Una vez que este RF se lleve a cabo muestra como quedaría el fichero una vez creado.	
<b>Post-requisito</b>	No procede.	

**Tabla 22: RF-2 Mostrar vista previa del fichero del escenario Gestionar fichero.**

RF -3: Generar fichero .json.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe haber creado un fichero.	RF-1.
<b>Descripción</b>	Permite generar un fichero en el directorio especificado anteriormente.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	El fichero se genera en la ubicación especificada.	
<b>Post-requisito</b>	No procede.	

**Tabla 23: RF-3 Generar fichero .json del escenario Gestionar fichero.**

RF -4: Copiar fichero .json a otro directorio.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	Fichero.	Ubicación.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe haber creado y generado un fichero.	RF-1, RF-3
<b>Descripción</b>	Permite copiar un fichero a otro directorio.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	El fichero se copia en la ubicación especificada.	
<b>Post-requisito</b>	No procede.	

**Tabla 24: RF-4 Copiar fichero .json a otro directorio del escenario Gestionar fichero.**

RF -4: Eliminar fichero .json.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe haber creado un fichero.	RF-1.
<b>Descripción</b>	Permite eliminar un fichero.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se elimina el fichero seleccionado.	
<b>Post-requisito</b>	No procede.	

*Tabla 25: RF-4 Eliminar fichero .json del escenario Gestionar fichero.*

**c) Escenario: Gestionar componente (s).**

RF -1: Adicionar componente (s) al fichero .json.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	Fichero, componente.	Tipo de label, Nombre del campo del label y Texto del label.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe haber creado un fichero.	RF-1.
<b>Descripción</b>	Permite adicionar componentes al fichero previamente creado.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se adicionan los componentes creados al fichero en cuestión.	
<b>Post-requisito</b>	No procede.	

**Tabla 26: RF-1 Adicionar componentes al fichero .json del escenario Gestionar componente.**

RF -2: Editar componente al fichero json.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	Fichero, componente.	Tipo de label, Nombre del campo del label y Texto del label.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe haber creado un fichero con al menos un componente incluido.	RF-1.
<b>Descripción</b>	Permite modificar un componente de un fichero creado.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se modifica el componente seleccionado.	
<b>Post-requisito</b>	No procede.	

*Tabla 27: RF-2 Editar componente al fichero json del escenario Gestionar componente.*

RF- 3: Ordenar o desordenar los componentes del fichero .json.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe existir un fichero con al menos dos componentes implícito.	RF-1.
<b>Descripción</b>	Permite ordenar o desordenar los componentes seleccionados de un fichero en cuestión.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se ordena o desordena los componentes seleccionados.	
<b>Post-requisito</b>	No procede.	

**Tabla 28: RF-3 Ordenar o desordenar los componentes del fichero .json del escenario Gestionar componente.**

RF- 4: Vista de componente (s) del fichero .json.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe existir al menos un fichero creado.	Escenario b) RF-1.
<b>Descripción</b>	Permite mostrar la vista de componentes para realizar las operaciones de adición de los mismos al fichero seleccionado.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se muestra la vista de componentes.	
<b>Post-requisito</b>	No procede.	

*Tabla 29: RF-4 Vista de componente (s) del fichero .json del escenario Gestionar componente.*

RF -5: Eliminar componente (s) al fichero .json.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe existir un fichero con al menos un componente implícito.	Escenario b) RF-1, RF-1.
<b>Descripción</b>	Permite eliminar componente (s) del fichero seleccionado.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se elimina uno o más componentes.	
<b>Post-requisito</b>	No procede.	

*Tabla 30: RF-5 Eliminar componente (s) al fichero .json del escenario Gestionar componente.*

### Anexo 3: Requisitos funcionales del componente Gestión de trazas.

**a) Escenario: Gestionar usuario(s) para el soporte.**

RF-1: Crear usuario.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	Usuario, soporte.	Nombre, apellidos, email, categoría.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	No procede.	No procede.
<b>Descripción</b>	Permite crear usuarios para el soporte. Estos usuarios son los encargados de gestionar las trazas.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se crean los usuarios y se muestran en la tabla de la interfaz principal.	
<b>Post-requisito</b>	No procede.	

**Tabla 31: RF-1 Crear usuario del escenario Gestionar usuario(s) para el soporte.**

RF -2: Modificar usuario.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	Usuario.	Nombre, apellidos, email, categoría.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe existir al menos un usuario creado.	RF-1.
<b>Descripción</b>	Permite modificar un usuario previamente seleccionado.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se modifica el usuario seleccionado.	
<b>Post-requisito</b>	No procede.	

*Tabla 32: RF-2 Modificar usuario del escenario Gestionar usuario(s) para el soporte.*

RF-3: Eliminar usuario (s).

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe existir al menos un usuario creado.	RF-1.
<b>Descripción</b>	Permite eliminar un usuario previamente seleccionado.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se eliminan el o los usuario (s) seleccionado (s).	
<b>Post-requisito</b>	No procede.	

**Tabla 33: RF-3 Eliminar usuario del escenario Gestionar usuario(s) para el soporte.**

RF-4: Exportar usuario (s).

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe existir al menos un usuario creado.	RF-1.
<b>Descripción</b>	Permite exportar los usuarios previamente creados a un documento pdf a un directorio especificado.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se exportan los usuarios a un documento pdf.	
<b>Post-requisito</b>	No procede.	

*Tabla 34: RF-4 Exportar usuario (s) del escenario Gestionar usuario(s) para el soporte.*

RF-5: Asignar y notificar traza (s) a usuario.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Deben existir al menos un usuario y una traza seleccionados previamente. Además los parámetros de configuración deben estar configurados.	RF-1, Escenario b) RF-1, Escenario c) RF-1.
<b>Descripción</b>	Permite asignar y notificar la (s) traza (s) seleccionada (s) al usuario especificado a través de un e-mail.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se asigna y se notifica la (s) traza (s) seleccionada (s) al usuario especificado a través de un e-mail.	
<b>Post-requisito</b>	No procede.	

**Tabla 35: RF-5 Asignar y notificar traza (s) a usuario del escenario Gestionar usuario(s) para el soporte.**

**b) Escenario: Parámetros de configuración.**

RF-1: Parámetros de configuración.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	Parámetros, configuración, correo.	Asunto, cuerpo del mensaje y dirección IP del servidor smtp.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	No procede.	No procede.
<b>Descripción</b>	Permite crear los parámetros de configuración que son necesarios para asignar y notificar las trazas.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se configuran los parámetros.	
<b>Post-requisito</b>	No procede.	

**Tabla 36: RF-1 Parámetros de configuración del escenario Parámetros de configuración.**

c) *Escenario: Gestionar trazas.*

RF -1: Importar trazas.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	No procede.	No procede.
<b>Descripción</b>	Permite cargar las trazas en una tabla de la interfaz principal.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se muestran las trazas en una tabla de la interfaz principal.	
<b>Post-requisito</b>	No procede.	

*Tabla 37: RF-1 Importar trazas del escenario Gestionar trazas.*

RF -2: Exportar trazas.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe existir al menos una traza en la tabla de la interfaz principal.	RF-1.
<b>Descripción</b>	Permite exportar las trazas a un documento pdf en el directorio especificado.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se muestran las trazas en un documento pdf.	
<b>Post-requisito</b>	No procede.	

*Tabla 38: RF-2 Exportar trazas del escenario Gestionar trazas.*

RF-3: Asignar usuario a la (s) traza (s).

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Debe existir al menos una traza y un usuario para realizar este proceso.	RF-1, Escenario a) RF-1.
<b>Descripción</b>	Permite asignar un usuario a la (s) traza (s) seleccionada (s).	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se asigna el usuario a la (s) traza (s) seleccionada (s).	
<b>Post-requisito</b>	No procede.	

**Tabla 39: RF-3 Asignar usuario a la (s) traza (s) del escenario Gestionar trazas.**

RF -4: Mostrar trazas asignadas.

<b>Conceptos tratados</b>	<b>Conceptos</b>	<b>Atributos</b>
	No procede.	No procede.
<b>Precondiciones</b>	<b>Precondiciones</b>	<b>Pre-requisito</b>
	Deben existir trazas asignadas a usuarios previamente creados.	RF-1, Escenario a) RF-5.
<b>Descripción</b>	Permite mostrar las trazas asignadas y notificadas de acuerdo al usuario seleccionado.	
<b>Validaciones</b>	No procede.	
<b>Post-condiciones</b>	Se muestran las trazas asignadas y notificadas.	
<b>Post-requisito</b>	No procede.	

*Tabla 40: RF-4 Mostrar trazas asignadas del escenario Gestionar trazas.*

#### Anexo 4: Diseño de clases del componente configuración de la gestión del multilinguaje.

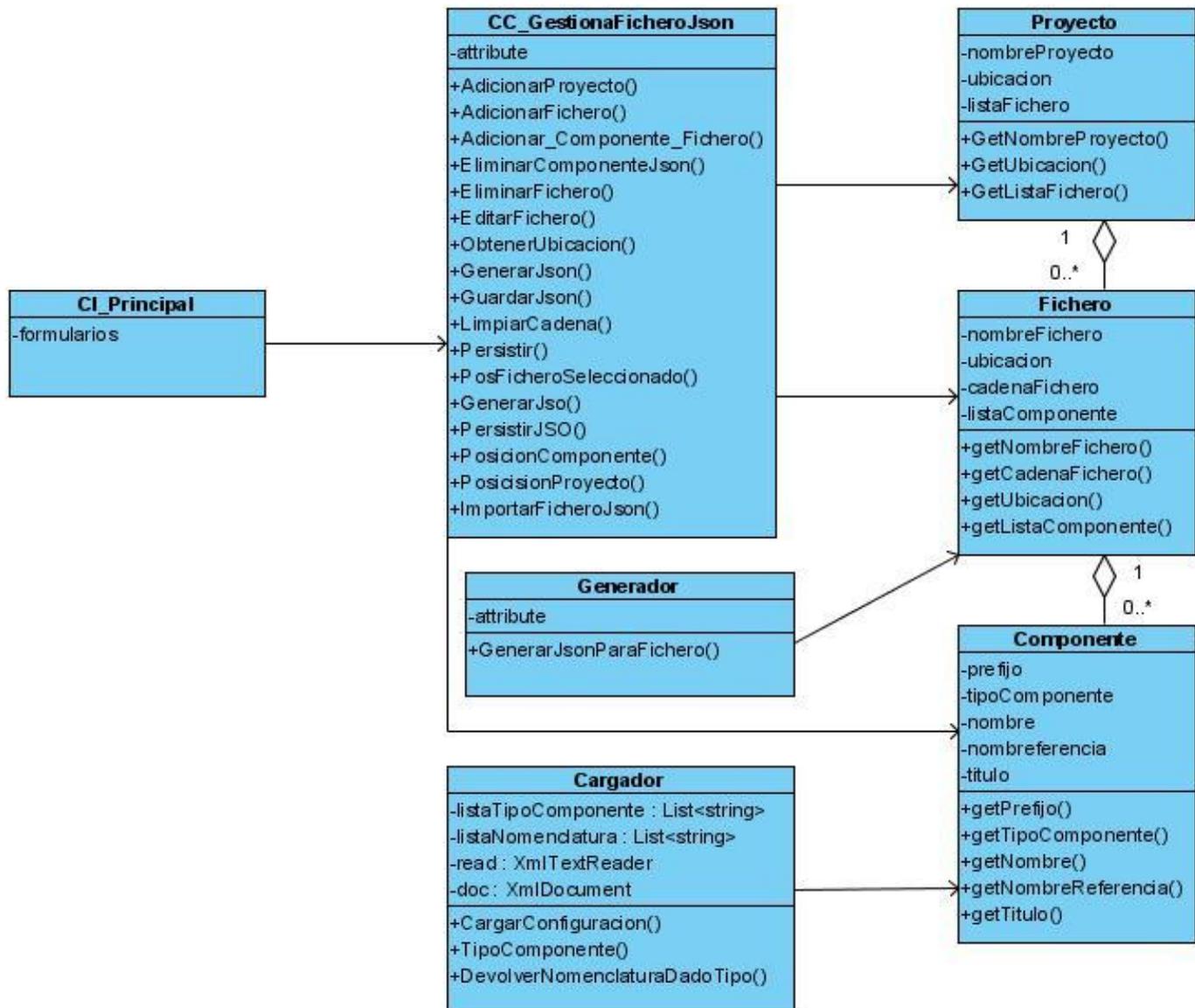


Figura 44: Diseño de clases del componente configuración de la gestión del multilinguaje.

## Anexo 5: Diagrama de clases del componente gestión de trazas.

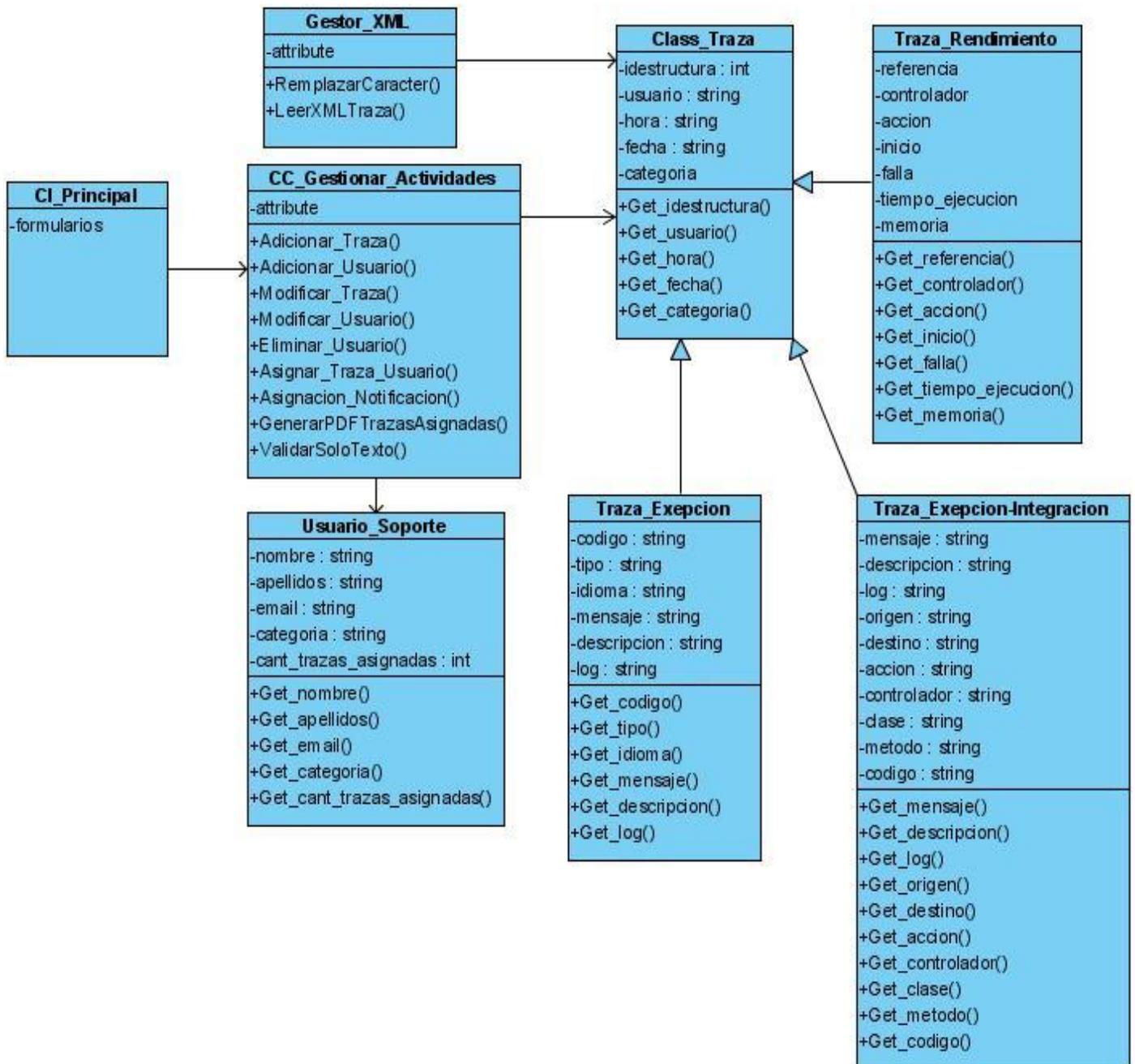


Figura 45: Diseño de clases del componente Gestión de trazas.

## Anexo 6: Diseño de casos de pruebas del componente configuración de la gestión del multilinguaje.

a) Escenario: Gestionar proyecto.

### DCP\_Cargar\_proyecto\_existente.

#### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft .NET Framework versión 2.0 u otra superior.
- Tiene que existir al menos un proyecto.
- Debe haber guardado anteriormente un proyecto.

#### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Cargar proyecto existente.	Se carga un proyecto que fue creado anteriormente.	EP 1.1: Cargar proyecto existente.	<ul style="list-style-type: none"> <li>– Se pulsa en el vínculo <b>Proyecto existente</b> de la interfaz principal.</li> <li>– Se acepta el mensaje de confirmación.</li> <li>– Se busca la dirección donde se encuentra.</li> <li>– Se pulsa el botón <b>Abrir</b>.</li> </ul>

		EP 1.2: Cargar proyecto existente.	<ul style="list-style-type: none"><li>– Se da clic en <b>Archivo/Proyecto existente</b>.</li><li>– Se acepta el mensaje de confirmación</li><li>– Se busca la dirección donde se encuentra.</li><li>– Se presiona el botón <b>Abrir</b>.</li></ul>
		EP 1.3: Cancelar la operación.	<ul style="list-style-type: none"><li>– Se presiona el vínculo <b>Proyecto existente</b>.</li><li>– Se busca o no la dirección donde se encuentra.</li><li>– Se pulsa el botón <b>Cancelar</b>.</li></ul>
		EP 1.4: Cancelar la operación.	<ul style="list-style-type: none"><li>– Se da clic en <b>Archivo/Proyecto existente</b>.</li><li>– Se busca o no la dirección donde se encuentra.</li><li>– Se presiona el botón <b>Cancelar</b>.</li></ul>

		EP 1.5: Cancelar la operación.	<ul style="list-style-type: none"> <li>- Se presiona en el vínculo <b>Proyecto existente</b> o dar clic en <b>Archivo/Proyecto existente</b>.</li> <li>- No se acepta el mensaje de confirmación.</li> </ul>
--	--	--------------------------------	--

1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	NA	NA	NA	NA

## 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Dirección	Respuesta del sistema	Resultado de la prueba
EP 1	Cargar proyecto existente	V(Escritorio)	Muestra una ventana de confirmación  " Desea cargar proyecto existente", después del usuario aceptar la opción se abre un navegador donde se selecciona la dirección donde se encuentra el proyecto a cargar. Se abre el proyecto seleccionado.	

EP 2	Cargar proyecto existente	V(Documents and Settings)	Muestra una ventana de confirmación  " Desea cargar un proyecto existente", después del usuario aceptar la opción se abre un navegador donde se selecciona la dirección donde se encuentra el proyecto a cargar. Se abre el proyecto seleccionado.	
EP 3	Cancelar la operación.	N/A	Se cancela la operación.	
EP 4	Cancelar la operación.	N/A	Se cancela la operación.	

EP 5	Cancelar la operación.	N/A	Se cancela la operación.	
------	------------------------	-----	--------------------------	--

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

## DCP\_Guardar\_proyecto\_como.

### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft. NET Framework versión 2.0 u otra superior.
- Tiene que existir al menos un proyecto.
- 

#### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Guardar proyecto como...	Se guarda un proyecto que fue creado anteriormente.	EP 1.1: Guardar proyecto como...	<ul style="list-style-type: none"> <li>– Se marca con doble clic el proyecto que desea guardar en el árbol de navegación.</li> <li>– Se va a <b>Archivo/Guardar proyecto como...</b> y se da clic.</li> <li>– Se selecciona la dirección donde desea guardarlo.</li> <li>– Se pulsa el botón <b>Guardar</b>.</li> </ul>

		EP 1.2: Cancelar la operación.	<ul style="list-style-type: none"> <li>- Se marca con doble clic el proyecto que desea guardar en el árbol de navegación.</li> <li>- Se va a <b>Archivo/Guardar proyecto como...</b> y se da clic.</li> <li>- Se pulsa el botón <b>Cancelar</b>.</li> </ul>
--	--	--------------------------------	---

1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	Ubicación	Explorador de Windows.	No	Una dirección válida.

## 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Dirección	Respuesta del sistema	Resultado de la prueba
EP 1	Guardar proyecto como.	V(Escritorio)	El sistema muestra un navegador para que el usuario escoja el lugar donde quiere guardar el proyecto. Se guarda el proyecto y se muestra la interfaz anterior.	
EP 2	Cancelar la operación.	N/A	Se cancela la operación.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_Guardar\_proyecto.****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft. NET Framework versión 2.0 u otra superior.
- Tiene que existir al menos un proyecto.

**1.1 Requisitos a probar:**

<b>Nombre del requisito</b>	<b>Descripción general</b>	<b>Escenarios de pruebas</b>	<b>Flujo del escenario</b>
1: Guardar proyecto.	Se guarda el proyecto seleccionado.	EP 1.1: Guardar proyecto.	<ul style="list-style-type: none"> <li>- Se marca con doble clic el proyecto que deseas guardar en el árbol de navegación.</li> <li>- Se da clic <b>Archivo/Guardar proyecto.</b></li> <li>- Se pulsa el botón <b>Aceptar.</b></li> </ul>

		<p>EP 1.2: Guardar proyecto.</p>	<ul style="list-style-type: none"> <li>- Se marca con doble clic el proyecto que deseas guardar en el árbol de navegación.</li> <li>- Se da clic en el ícono <b>Guardar proyecto</b> que está en la barra de herramientas.</li> <li>- Se pulsa el botón <b>Aceptar</b>.</li> </ul>
		<p>EP 1.3: Guardar proyecto que no tenga fichero(s), y este último no tenga componentes adicionales.</p>	<ul style="list-style-type: none"> <li>- Se marca con doble clic el proyecto que deseas guardar en el árbol de navegación.</li> <li>- Se da clic en el ícono <b>Guardar proyecto</b> que está en la barra de herramientas o en <b>Archivo/Guardar proyecto</b>.</li> <li>- Se muestra un mensaje de información referente a la no existencia de ficheros y</li> </ul>

			componentes. - Se acepta el mensaje anterior.
--	--	--	--

### 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	N/A	N/A	N/A	N/A

### 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Guardar proyecto.	Se guarda el proyecto, y se muestra un mensaje de acción realizada correctamente.	
EP 2	Guardar proyecto.	Se guarda el proyecto y se muestra un mensaje de acción realizada correctamente.	
EP 3	Guardar proyecto que no tenga fichero(s), y este último no tenga componentes	Se muestra un mensaje de información referente a la no existencia de ficheros y componentes.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/P no procede.

b) Escenario: Gestionar fichero.

### DCP\_Crear\_fichero.

#### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft. NET Framework versión 2.0 u otra superior.
- No puede existir otro fichero con el mismo nombre.
- Se debe haber creado un proyecto.
- 

#### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Crear nuevo fichero .json.		EP 1.1: Crear nuevo fichero .json.	<ul style="list-style-type: none"> <li>– Se da doble clic al proyecto que se encuentra en el árbol de navegación al cual se desea agregarle un fichero.</li> <li>– Se da clic en <b>Edición/Nuevo fichero.</b></li> </ul>

		<p>EP 1.2: Crear nuevo fichero .json.</p>	<ul style="list-style-type: none"><li>– Se presiona el ícono correspondiente a <b>Nuevo fichero</b> que se encuentra en la barra de herramienta.</li><li>– Se introduce el nombre del fichero.</li><li>– Se selecciona la dirección en el navegador y se acepta.</li></ul>
		<p>EP 1.3: Crear un nuevo fichero dejando campos requeridos en blanco.</p>	<ul style="list-style-type: none"><li>– Se presiona en <b>Edición/Nuevo fichero</b>.</li><li>– Se introducen los datos del fichero que se desea adicionar, dejando al menos un campo requerido en blanco.</li><li>– Se selecciona la dirección en el navegador y se acepta.</li><li>– Se acepta el mensaje de información, da clic en <b>Aceptar</b>.</li></ul>

			<ul style="list-style-type: none"> <li>- Se introducen todos los datos, sin dejar ningún campo vacío.</li> <li>- Se acepta la operación en el botón <b>Aceptar.</b></li> </ul>
		<p>EP 1.4: Crear un nuevo fichero dejando campos requeridos en blanco.</p>	<ul style="list-style-type: none"> <li>- Se presiona el ícono <b>Nuevo fichero</b> que se encuentra en la barra de herramienta.</li> <li>- Se introducen los datos del fichero que se desea adicionar, dejando al menos un campo requerido en blanco.</li> <li>- Se selecciona la dirección en el navegador y se acepta.</li> <li>- Se acepta el mensaje de información, da clic en el botón <b>Aceptar.</b></li> <li>- Se introducen todos los datos, sin dejar ningún campo vacío.</li> <li>- Se acepta la operación en el botón</li> </ul>

			<b>Aceptar.</b>
		EP 1.5: Cancelar la operación.	<ul style="list-style-type: none"> <li>– Se da clic en el ícono <b>Nuevo fichero</b>, en la barra de herramienta.</li> <li>– Se introducen o no los datos del fichero.</li> <li>– Se presiona el botón <b>Cancelar</b>.</li> </ul>

#### 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	Nombre del fichero	<b>Campo de texto.</b>	<b>No</b>	Combinación de letras.
2	Ubicación	<b>Explorador de Windows</b>	<b>No</b>	Una dirección válida.

## 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Nombre	Variable 2 Dirección	Respuesta del sistema	Resultado de la prueba
EP 1	Crear nuevo fichero .json.	V(fichero)	V(Escritorio)	Se crea el nuevo fichero y se muestra en el árbol de navegación.	
EP 2	Crear nuevo fichero .json.	V(nuevo)	V(Mis documentos)	Se crea el nuevo fichero y se muestra en el árbol de navegación.	
EP 3	Crear fichero dejando campos requeridos en blanco.	I()	V(Mis documentos)	Se muestra un mensaje que indica que no se pueden dejar campos en blanco “Debe llenar todos los campos” y se mantiene en la interfaz para adicionar el fichero.	

		V(nuevo)	I()	Se muestra un mensaje que indica que no se pueden dejar campos en blanco "Debe llenar todos los campos" y se mantiene en la interfaz para adicionar el fichero.	
EP 4	Crear fichero dejando campos requeridos en blanco.	I( )	V(Escritorio)	Se muestra un mensaje que indica que no se pueden dejar campos en blanco "Debe llenar todos los campos" y se mantiene en la Interfaz para adicionar el fichero.	

		V( fichero )	I()	Se muestra un mensaje que indica que no se pueden dejar campos en blanco “Debe llenar todos los campos” y se mantiene en la interfaz para adicionar el fichero.	
EP 5	Cancelar la operación.	NA	NA	Se cancela la operación.	
EP 6	Cancelar la operación.	NA	NA	Se cancela la operación.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_Eliminar\_fichero.****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft. NET Framework versión 2.0 u otra superior.
- Tiene que existir al menos un fichero .json.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Eliminar fichero .json.	Se elimina el fichero seleccionado. Y se muestra un mensaje de acción realizada correctamente.	EP 1.1: Eliminar fichero .json.	<ul style="list-style-type: none"> <li>- Se marca con doble clic el fichero que deseas eliminar en el árbol de navegación.</li> <li>- Se da clic en <b>Edición/Eliminar fichero.</b></li> <li>- Se acepta el mensaje de confirmación.</li> </ul>
		EP 1.2: Eliminar fichero .json cuando no existen ficheros creados.	<ul style="list-style-type: none"> <li>- Se da clic en <b>Edición/Eliminar fichero.</b></li> <li>- Se acepta el mensaje de información referente a la no existencia de ficheros.</li> </ul>

		EP 1.3: Cancelar la operación.	<ul style="list-style-type: none"> <li>- Se marca el fichero que deseas eliminar en el árbol de navegación.</li> <li>- Se da clic en <b>Edición/Eliminar fichero.</b></li> <li>- No se acepta el mensaje de confirmación.</li> </ul>
--	--	--------------------------------	--

1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	NA	NA	NA	NA

1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Eliminar fichero .json.	Se elimina el fichero seleccionado y se actualiza la información del mismo.	
EP 2	Eliminar fichero .json cuando no existen ficheros creados.	Muestra un mensaje de información referente a la no existencia de ficheros creados.	

EP 3	Cancelar la operación.	Se cancela la operación.	
------	------------------------	--------------------------	--

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

### DCP\_Copiar\_fichero\_otro\_directorio.

#### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Se debe haber creado un proyecto.
- Debe existir al menos un fichero.

#### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Copiar fichero .json a otro directorio.	Se guarda el fichero a otro directorio que el usuario selecciona.	EP 1.1: Guardar fichero .json a otro directorio.	<ul style="list-style-type: none"> <li>- Se presiona el vínculo <b>Copiar fichero a otro directorio</b>.</li> <li>- Se selecciona la nueva dirección en el navegador.</li> <li>- Se acepta la dirección seleccionada.</li> <li>- Se presiona el botón <b>Aceptar</b> del mensaje de información que se muestra.</li> </ul>

		EP 1.2: Cancelar la operación.	<ul style="list-style-type: none"> <li>- Se presiona el vínculo <b>Copiar fichero a otro directorio.</b></li> <li>- Se da clic en el botón <b>Cancelar.</b></li> </ul>
--	--	--------------------------------	--

**1.1.1 Descripción de variable.**

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	Nombre de carpeta.	Explorador de Windows.	No.	Una dirección válida.

**1.1.2 Juegos de datos a probar**

Id del escenario	Escenario	Variable 1 Dirección	Respuesta del sistema	Resultado de la prueba
EP 1	Copiar fichero .json a otro directorio.	V(Escritorio)	Se guarda el fichero en la dirección seleccionada y se muestra un mensaje de acción realizada correctamente.	
EP 2	Cancelar operación	NA	Se cancela la operación.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_Vista\_previa\_fichero.****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Se debe haber creado un proyecto.
- Debe existir al menos un fichero.
- Debe existir componente(s) en el fichero.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Mostrar vista previa del fichero .json.	Se muestra una vista previa del fichero .json seleccionado.	EP 1.1: Mostrar vista previa del fichero .json.	<ul style="list-style-type: none"> <li>– Se marca con doble clic el fichero que desea visualizar.</li> <li>– Se da clic en la barra de herramienta en el ícono <b>Vista previa del fichero.</b></li> <li>– Se da clic en la <b>Vista de componentes</b>, para regresar a la interfaz anterior.</li> </ul>

		EP 1.2: Mostrar vista previa del fichero .json.	<ul style="list-style-type: none"> <li>- Se marca con doble clic el fichero que desea visualizar.</li> <li>- Se da clic en Vistas/Vista previa del fichero.</li> <li>- Se da clic en la Vista de componentes, para regresar a la interfaz anterior.</li> </ul>
--	--	---	--

**1.1.1 Descripción de variable.**

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1		NA	NA	NA

**1.1.2 Juegos de datos a probar**

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Mostrar vista previa del fichero	Muestra la vista previa del fichero .json seleccionado.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_Generar\_fichero\_json.****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Se debe haber creado un proyecto.
- Debe existir al menos un fichero.
- Debe existir componente(s) en el fichero.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Generar fichero json.	Se genera el fichero json seleccionado.	EP 1.1: Generar fichero json.	<ul style="list-style-type: none"> <li>– Se marca con doble clic el fichero que desea generar.</li> <li>– Se da clic en botón <b>Generar fichero .json</b> que encuentra en la Interfaz de la <b>Vista de componentes</b>.</li> <li>– Se acepta el mensaje de información referente a la generación del fichero.</li> </ul>

		EP 1.2: Generar fichero json.	<ul style="list-style-type: none"> <li>- Se marca con doble clic el fichero que desea generar.</li> <li>- Se da clic en Generación/Generar fichero .json.</li> <li>- Se acepta el mensaje de información referente a la generación del fichero.</li> </ul>
--	--	-------------------------------	--

**1.1.1 Descripción de variable.**

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1		NA	NA	NA

**1.1.2 Juegos de datos a probar**

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Generar fichero json.	Genera el fichero .json seleccionado y se muestra un mensaje de acción realizada correctamente.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

c) Escenario: Gestionar componente.

### DCP\_Adicionar\_componente\_fichero\_json.

#### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft .NET Framework versión 2.0 u otra superior.
- Tiene que existir al menos un proyecto.
- Tiene que existir un fichero .json.

#### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Adicionar componente al fichero .json.	Se le adicionan los componentes al fichero .json seleccionado.	EP 1.1: Adicionar componente al fichero .json.	<ul style="list-style-type: none"> <li>– Se marca con doble clic el fichero al cual se añadirá el componente, este fichero se encuentra en el árbol de navegación.</li> <li>– Se introducen los datos del componente que se añadirá (Tipo de label, Nombre del campo del label, Texto del label).</li> <li>– Se pulsa el botón <b>Adicionar elemento</b>.</li> </ul>

		<p>EP 1.2: Adicionar componente al fichero .json dejando al menos un campo vacío.</p>	<ul style="list-style-type: none"> <li>- Se marca con doble clic el fichero en el árbol de navegación.</li> <li>- Se introducen los datos del componente dejando al menos un campo requerido en blanco.</li> <li>- Se pulsa el botón <b>Adicionar elemento</b>.</li> <li>- Se acepta el mensaje de Aviso "Debe llenar todos los campos".</li> <li>- Se llenan todos los campos.</li> </ul>
		<p>EP 1.3 Adicionar componente al fichero .json repitiendo datos de otro componente creado antes.</p>	<ul style="list-style-type: none"> <li>- Se marca con doble clic el fichero en el árbol de navegación.</li> <li>- Se introducen los datos del componente repitiendo datos de otro componente.</li> <li>- Se presiona el botón <b>Adicionar</b></li> </ul>

			<p><b>elemento.</b></p> <ul style="list-style-type: none"> <li>- Se acepta el mensaje de información "Este componente ya existe."</li> <li>- Se llenan todos los campos con datos diferentes a los componentes creados anteriormente.</li> <li>- Se presiona el botón <b>Adicionar elemento.</b></li> </ul>
--	--	--	---

1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	Tipo de label	Combo Box	No	Una opción de las que te brinda el sistema.
2	Nombre del campo del label	Campo de texto	No	Una combinación de letras
3	Texto del label	Campo de texto	No	Una combinación de letras

## 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Tipo de label	Variable 2 Nombre del campo del label	Variable N Texto del label	Respuesta del sistema	Resultado de la prueba
EP 1	Adicionar componente al fichero .json.	V(Botón)	V(Aceptar)	V(Aceptar)	Se adiciona un componente al fichero y se muestra en la Interfaz.	
EP 2	Adicionar componente al fichero .json.	V(Mensaje)	V(Cancelar )	V(Cancelar)	Se adiciona un componente al fichero y se muestra en la Interfaz.	
EP 3	Adicionar componente al fichero .json dejando al menos un campo vacio.	V(Botón)	V(Aceptar)	I( )	Muestra un mensaje de campo obligatorio "Debe llenar todos los campos", y mantiene la interfaz abierta para adicionar el componente.	

		V(Botón)	I( )	V(Aceptar)	Muestra un mensaje de campo obligatorio "Debe llenar todos los campos", y mantiene la interfaz abierta para adicionar el componente.
EP 4	Adicionar componente al fichero .json repitiendo datos de otro componente creado antes.	V(Botón)	V(Aceptar)	V(Aceptar)	Muestra un mensaje de información "Este componente ya existe", y deja la interfaz abierta para adicionar el componente.

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_Editar\_componente\_fichero\_json.****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft .NET Framework versión 2.0 u otra superior.
- Tiene que existir al menos un proyecto.
- Tiene que existir al menos un fichero .json.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Editar componente al fichero .json.	Se modifican los componentes del fichero seleccionado, según las necesidades del usuario.	EP 1.1: Editar componente(s) al fichero .json.	<ul style="list-style-type: none"> <li>- Se marca el fichero que deseas editar con doble clic en el árbol de navegación y el componente que se modificará con un clic simple.</li> <li>- Se pulsa el menú <b>Edición/ Editar componente.</b></li> <li>- Se introducen los nuevos datos del componente del fichero .json.</li> <li>- Se pulsa el botón <b>Aplicar cambios.</b></li> </ul>

		EP 1.2: Editar componente(s) al fichero .json	<ul style="list-style-type: none"><li>- Se marca con doble clic el fichero que deseas editar en el árbol de navegación.</li><li>- Con doble clic se selecciona el componente que se modificará, este componente se mostrará en el campo correspondiente.</li><li>- Se modifican los datos del componente.</li><li>- Se pulsa el botón <b>Aplicar cambios.</b></li></ul>
		EP 1.3: Editar componente(s) al fichero .json	<ul style="list-style-type: none"><li>- Se marca con doble clic el fichero que deseas editar en el árbol de navegación.</li><li>- Se pulsa el ícono <b>Editar componente</b> de la barra de herramienta.</li><li>- Se introducen los nuevos datos del componente del fichero .json.</li></ul>

			- Se pulsa el botón <b>Aplicar cambios</b> .
--	--	--	--

### 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	Tipo de label.	Combo Box.	No.	Una opción de las que te brinda el sistema
2	Nombre del campo del label.	Campo de texto.	No.	Una combinación de letras.
3	Texto del label.	Campo de texto.	No.	Una combinación de letras.

### 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Tipo de label	Variable 2 Nombre del campo del label	Variable N Texto del label	Respuesta del sistema	Resultado de la prueba
EP 1	Editar componente	V(Botón)	V(Aceptar)	V(Aceptar)	Se modifica el componente seleccionado y	

	al fichero .json.	V(Botón)	V(Aceptar)	I( )	se muestra un mensaje de información "Debe llenar todos los campos"	
		V(Botón)	I( )	V(Aceptar)	Se muestra un mensaje de información "Debe llenar todos los campos"	
		I( )	V(Aceptar)	V(Aceptar)	Se muestra un mensaje de información "Debe llenar todos los campos"	
EP 2	Editar componente(s) al fichero .json.	V(Mensaje)	V(Cancelar )	V(Cancelar)	Se modifica el componente seleccionado y se actualiza la información en el fichero.	
		V(Botón)	V(Aceptar)	I( )	Se muestra un mensaje de información "Debe llenar todos los campos"	

		V(Botón)	I( )	V(Aceptar)	Se muestra un mensaje de información "Debe llenar todos los campos"	
		I( )	V(Aceptar)	V(Aceptar)	Se muestra un mensaje de información "Debe llenar todos los campos"	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_Eliminar\_componente\_fichero\_json.****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft .NET Framework versión 2.0 u otra superior.
- Tiene que existir al menos un proyecto.
- Tiene que existir al menos un fichero .json.
- El fichero debe tener al menos un componente.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Eliminar componente(s).	Se elimina el componente seleccionado al fichero .json.	EP 1.1: Eliminar componente(s).	<ul style="list-style-type: none"> <li>- Se marca el componente que deseas eliminar.</li> <li>- Se pulsa en el menú <b>Edición/ Eliminar componente(s)</b>.</li> <li>- Se acepta el mensaje de confirmación.</li> </ul>

		<p>EP 1.2: Eliminar componente(s).</p>	<ul style="list-style-type: none"> <li>- Se marca el componente que deseas eliminar.</li> <li>- Se presiona el ícono <b>Eliminar componente(s)</b>.</li> <li>- Se acepta el mensaje de confirmación.</li> </ul>
		<p>EP 1.3: Eliminar más de un componente.</p>	<ul style="list-style-type: none"> <li>- Se marcan los componentes que deseas eliminar.</li> <li>- Se presiona en el menú <b>Edición/Eliminar componente(s)</b>.</li> <li>- Se acepta el mensaje de confirmación.</li> </ul>
		<p>EP 1.4: Eliminar más de un componente.</p>	<ul style="list-style-type: none"> <li>- Se marcan los componentes que deseas eliminar.</li> <li>- Se presiona el ícono <b>Eliminar componente(s)</b>.</li> <li>- Se acepta el mensaje de confirmación.</li> </ul>

		<p>EP 1.5: Eliminar uno o más componentes sin ser seleccionado(s).</p>	<ul style="list-style-type: none"> <li>- No se marca él o los componentes que se desean eliminar.</li> <li>- Se acepta el mensaje de información, se da clic en el botón <b>Aceptar</b> de este mensaje.</li> <li>- Se marca el o los componentes que se desea eliminar.</li> <li>- Se presiona el ícono <b>Eliminar componente(s)</b> que se encuentra en la barra de herramienta o en el menú <b>Edición/Eliminar componente(s)</b>.</li> <li>- Se acepta el mensaje de confirmación.</li> </ul>
		<p>EP 1.6: Cancelar Operación.</p>	<ul style="list-style-type: none"> <li>- Se marca el componente que deseas eliminar.</li> </ul>

			<ul style="list-style-type: none"> <li>- Se presiona el menú <b>Edición/Eliminar componente(s)</b> o el ícono <b>Eliminar componente(s)</b>.</li> <li>- No se acepta el mensaje de confirmación.</li> </ul>
--	--	--	---

### 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	N/A	N/A	N/A	N/A

### 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Eliminar componente.	Se elimina el componente del fichero seleccionado, muestra un mensaje de confirmación y se actualiza la información del mismo.	
EP 2	Eliminar componente.	Se elimina el componente del fichero seleccionado, muestra un mensaje e confirmación y se actualiza la información del mismo.	

EP 3	Eliminar más de un componente.	Se eliminan los componentes del fichero seleccionado, muestra un mensaje de confirmación y se actualiza la información del mismo.	
EP 4	Eliminar más de un componente.	Se eliminan los componentes del fichero seleccionado, muestra un mensaje de confirmación y se actualiza la información del mismo.	
EP 5	Eliminar uno o más sin ser seleccionado(s).	Se muestra un mensaje de información: "Debe seleccionar al menos un componente", luego de marcado el componente se elimina el mismo y se actualiza la información del mismo.	
EP 6	Cancelar operación.	Se cancela la operación.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

### **DCP\_Ordenar\_desordenar\_componentes\_fichero\_json.**

#### **1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Tiene que existir al menos un proyecto.
- Tiene que existir un fichero .json.
- Los ficheros tienen que contener componentes.

## 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Ordenar los componentes del fichero .json.	Se ordenan los componentes del fichero seleccionado.	EP 1.1: Ordenar los componentes del fichero .json.	<ul style="list-style-type: none"> <li>– Se marca la opción <b>Vista de ordenamiento</b> en la barra de herramienta.</li> <li>– Se selecciona el fichero con doble clic, que se encuentra en el árbol de navegación.</li> <li>– Se marca en el orden decidido los componentes.</li> <li>– Se presiona el botón <b>Ordenar un elemento</b> u <b>Ordenar todos los elementos</b>.</li> <li>– Se presiona el botón <b>Guardar cambios</b>.</li> </ul>

		EP 1.2: Ordenar los componentes del fichero .json	<ul style="list-style-type: none"><li>- Se marca <b>Vistas/ Ordenamientos de ficheros</b> del menú.</li><li>- Se selecciona el fichero con doble clic,</li><li>- encontrándose en el árbol de navegación.</li><li>- Se marca en el orden decidido los componentes</li><li>- Se presiona el botón <b>Ordenar un elemento</b> u <b>Ordenar todos los elementos.</b></li><li>- Se pulsa el botón <b>Guardar cambios.</b></li></ul>
--	--	---	---

		<p>EP1.3: Desordenar los componentes del fichero .json.</p>	<ul style="list-style-type: none"><li>- Se da clic la opción <b>Vista de ordenamiento</b> en la barra de herramienta.</li><li>- Se selecciona en el árbol de navegación el fichero al cual se le desordenarán los componentes.</li><li>- Se marca en el orden decidido los componentes.</li><li>- Se presiona el botón <b>Desordenar un elemento o Desordenar todos los elementos.</b></li><li>- Se presiona el botón <b>Guardar cambios.</b></li></ul>
--	--	---	---

		EP1.4 Desordenar los componentes del fichero .json.	<ul style="list-style-type: none"><li>- Se da clic <b>Vistas/ Ordenamientos de ficheros</b> del menú.</li><li>- Se selecciona en el árbol de navegación con doble clic el fichero al cual se le desordenarán los componentes.</li><li>- Se marca en el orden decidido los componentes.</li><li>- Se presiona el botón <b>Desordenar un elemento o Desordenar todos los elementos.</b></li><li>- Se pulsa el botón <b>Guardar cambios.</b></li></ul>
--	--	---	---

		<p>EP1.5 Ordenar o Desordenar elemento(s) cuando no existen.</p>	<ul style="list-style-type: none"> <li>- Se da clic en <b>Vistas/Ordenamiento de ficheros</b> del menú o en <b>Vista de ordenamiento</b> en la barra de herramienta.</li> <li>- Se selecciona el fichero al cual desea ordenar o desordenar sus componente(s) en el árbol de navegación con doble clic.</li> <li>- Se acepta el mensaje de información referente a la no existencia de elementos para ordenar o desordenar.</li> </ul>
--	--	--	--

## 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	NA	NA	NA	NA.

## 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Sin ordenar	Variable 2 ordenado	Respuesta del sistema	Resultado de la prueba
EP 1	Ordenar los componentes del fichero .json.	V(Botón1) V(Botón)	V(Botón) V(Botón1)	Se ordenan los componentes del fichero y se muestra en la interfaz "ordenado"	
EP 2	Ordenar los componentes del fichero .json.	V(Mensaje1) V(Mensaje) V(Mensaje 2)	V(Mensaje) V(Mensaje1) V(Mensaje 2)	Se ordenan los componentes del fichero y se muestra en la interfaz "ordenado"	
EP 3	Desordenar los componentes del fichero	V(Botón1) V(Botón4) V(Botón)	V(Botón) V(Botón1) V(Botón4)	Se desordenan los componentes del fichero y se muestra en la interfaz "Sin ordenar".	
EP 4	Desordenar los componentes del fichero .json.	V(Mensaje1) V(Mensaje) V(Mensaje3)	V(Mensaje) V(Mensaje1) V(Mensaje3)	Se desordenan los componentes del fichero y se muestra en la interfaz "Sin ordenar".	

EP 5	Ordenar o Desordenar elemento(s) cuando no existen.	NA	NA	Muestra un mensaje de información referente a la no existencia de elementos para ordenar o desordenar.	
------	---	----	----	--	--

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

### DCP\_Vista\_componente (s).

#### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Tiene que existir al menos un proyecto.
- Tiene que existir un fichero .json.

#### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Vista de componente (s).	Se muestra la vista de componentes del fichero seleccionado.	EP 1.1: Vista de componente (s).	<ul style="list-style-type: none"> <li>– Se selecciona el fichero con doble clic, que se encuentra en el árbol de navegación.</li> <li>– Se marca la opción <b>Vista de componente (s)</b> en la barra de herramienta.</li> </ul>

## 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	NA	NA	NA	NA.

## 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Vista de componente (s).	Se muestra la interfaz vista de componente (s).	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

## Anexo 7: Diseño de casos de pruebas del componente gestión de trazas.

a) Escenario: Gestionar usuario(s) para el soporte.

### DCP\_ Crear\_usuario.

#### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.

#### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Crear usuario.	Se adicionan usuarios para el soporte y la gestión de la (s) traza (s).	EP 1.1: Crear usuario.	<ul style="list-style-type: none"> <li>– Se selecciona en el menú principal la opción <b>Gestión de soporte/Crear usuario</b>.</li> <li>– Se introducen los datos del usuario (Nombre, Apellidos, Email, Categoría)</li> <li>– Se pulsa el botón <b>Aceptar</b>.</li> </ul>

		<p>EP 1.2: Crear usuario dejando al menos un campo vacío.</p>	<ul style="list-style-type: none"><li>- Se selecciona en el menú principal la opción <b>Gestión de soporte/Crear usuario</b>.</li><li>- Se introducen los datos del usuario (Nombre, Apellidos, Email, Categoría)</li><li>- Se pulsa el botón <b>Aceptar</b>.</li><li>- Se acepta el mensaje de Aviso "Debe llenar todos los campos".</li><li>- Se llenan todos los campos.</li></ul>
--	--	---	---

		<p>EP 1.3: Crear usuario introduciendo mal el formato del email.</p>	<ul style="list-style-type: none"><li>- Se selecciona en el menú principal la opción <b>Gestión de soporte/Crear usuario</b>.</li><li>- Se introducen los datos del usuario (Nombre, Apellidos, Email, Categoría)</li><li>- Se pulsa el botón <b>Aceptar</b>.</li><li>- Se acepta el mensaje de Aviso "Verifique que email tenga el formato correcto."</li><li>- Se introduce el email nuevamente con su formato correcto.</li></ul>
--	--	--	--

		<p>EP 1.4: Crear usuario repitiendo datos de otro usuario creado antes.</p>	<ul style="list-style-type: none"><li>- Se selecciona en el menú principal la opción <b>Gestión de soporte/Crear usuario</b>.</li><li>- Se introducen los datos del usuario (Nombre, Apellidos, Email, Categoría)</li><li>- Se pulsa el botón <b>Aceptar</b>.</li><li>- Se acepta el mensaje de información " El usuario que intenta adicionar ya existe."</li><li>- Se llenan todos los campos con datos diferentes a los usuarios creados anteriormente.</li><li>- Se presiona el botón <b>Aceptar</b>.</li></ul>
--	--	---	---

## 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	Nombre	Campo de texto	No	Una combinación de letras
2	Apellidos	Campo de texto	No	Una combinación de letras
3	Categoría	Combo Box	No	Una opción de las que te brinda el sistema.
4	Email	Campo de texto	No	Una combinación de letras

## 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Nombre	Variable 2 Apellidos	Variable 3 Categoría	Variable N Email	Respuesta del sistema	Resultado de la prueba
EP 1	Adicionar usuario	V(Yormis)	V(Corrales)	V(Arquitectura)	V(yormiscm@gmail.com)	Se adiciona un usuario y se muestra en la Interfaz.	

EP 2	Adicionar usuario dejando al menos un campo vacio.	V(Yormis)	V(Corrale s)	V(Arquitectura)	I( )	Muestra un mensaje de campo obligatorio "Debe llenar todos los campos", y mantiene la interfaz abierta para adicionar el usuario.
EP 3	Crear usuario introduciendo mal el formato del email.	V(Yormis)	V(Corrale s)	V(Arquitectura)	V(yormiscmgmail.com)	Muestra un mensaje de campo obligatorio "Verifique que email tenga el formato correcto.", y mantiene la interfaz abierta para adicionar el usuario y corregir el formato.
EP 4	Adicionar usuario repitiendo datos de otro usuario antes creado.	V(Yormis)	V(Corrale s)	V(Arquitectura)	V(yormiscm@gmail.com)	Muestra un mensaje de información "El usuario que intenta adicionar ya existe", y deja la interfaz abierta para adicionar otro usuario.

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_ Modificar usuario.****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Debe existir al menos un usuario creado.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Modificar usuario.	Se modifica el usuario seleccionado o según las necesidades del usuario.	EP 1.1: Modificar usuario.	<ul style="list-style-type: none"> <li>– Se marca el usuario que se desea modificar con un solo clic.</li> <li>– Se selecciona en la barra de herramientas el ícono <b>Modificar usuario.</b></li> <li>– Se introducen los nuevos datos del usuario en la interfaz correspondiente.</li> <li>– Se pulsa el botón <b>Modificar.</b></li> </ul>

		EP 1.2: Modificar usuario.	<ul style="list-style-type: none"><li>- Se marca el usuario que deseas editar con doble clic, este usuario se mostrará en la interfaz correspondiente.</li><li>- Se modifican los datos del usuario.</li><li>- Se pulsa en el botón <b>Modificar</b>.</li></ul>
--	--	-------------------------------	---

		<p>EP 1.3:          Modificar usuario sin ser seleccionado.</p>	<ul style="list-style-type: none"> <li>- Se marca el usuario que se desea modificar con un solo clic.</li> <li>- Se selecciona en la barra de herramientas el ícono <b>Modificar usuario.</b></li> <li>- Se introducen los nuevos datos del usuario en la interfaz correspondiente.</li> <li>- Se pulsa el botón <b>Modificar.</b></li> <li>- Se muestra un mensaje de información: “Debe seleccionar el usuario que desea modificar.”</li> </ul>
--	--	---	---

### 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	Nombre	Campo de texto	No	Una combinación de letras
2	Apellidos	Campo de texto	No	Una combinación de letras

3	Categoría	Combo Box	No	Una opción de las que te brinda el sistema.
4	Email	Campo de texto	No	Una combinación de letras

### 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Nombre	Variable 2 Apellidos	Variable 3 Categoría	Variable N Email	Respuesta del sistema	Resultado de la prueba
EP 1	Modificar usuario	V(Yormis)	V(Corrales)	V(Arquitectura)	V(yormiscm@gmail.com)	Se modifica el usuario correspondiente y se actualizan los datos.	
		V(Yormis)	V(Corrales)	V(Arquitectura)	I( )	Se muestra un mensaje de información "Debe llenar todos los campos"	
		V(Yormis)	V(Corrales)	I( )	V(yormiscm@gmail.com)	Se muestra un mensaje de información "Debe llenar todos los campos"	

		V(Yormis)	I( )	V(Arquitectura)	V(yormiscm@gmail.com)	Se muestra un mensaje de información "Debe llenar todos los campos"	
		I( )	V(Corrales)	V(Arquitectura)	V(yormiscm@gmail.com)	Se muestra un mensaje de información "Debe llenar todos los campos"	
EP 2	Modificar usuario	V(Yormis)	V(Corrales)	V(Arquitectura)	V(yormiscm@gmail.com)	Se modifica el usuario correspondiente y se actualizan los	
		V(Yormis)	V(Corrales)	V(Arquitectura)	I( )	Se muestra un mensaje de información "Debe llenar todos los campos".	

		V(Yormis)	V(Corrales)	I( )	V(yormiscm@gmail.com)	Se muestra un mensaje de información "Debe llenar todos los campos".	
		V(Yormis)	I( )	V(Arquitectura)	V(yormiscm@gmail.com)	Se muestra un mensaje de información "Debe llenar todos los campos".	
		I( )	V(Corrales)	V(Arquitectura)	V(yormiscm@gmail.com)	Se muestra un mensaje de información "Debe llenar todos los campos".	
EP 3	Modificar usuario sin ser seleccionado.	N/A	N/A	N/A	N/A	Se muestra un mensaje de información: "Debe seleccionar el usuario que desea"	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_ Eliminar\_usuario (s).****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Debe existir al menos un usuario creado.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Eliminar usuario (s).	Se elimina el usuario seleccionado.	EP 1.1: Eliminar usuario (s).	<ul style="list-style-type: none"> <li>– Se marca el usuario que se desea eliminar con un solo clic.</li> <li>– Se selecciona en la barra de herramientas el ícono <b>Eliminar usuario (s)</b>.</li> </ul>
	Se eliminan los usuarios seleccionados.	EP 1.2: Eliminar mas de un usuario.	<ul style="list-style-type: none"> <li>– Se marcan los usuarios que se desean eliminar con un solo clic.</li> <li>– Se selecciona en la barra de herramientas el ícono <b>Eliminar usuario (s)</b>.</li> </ul>

		EP 1.3: Eliminar uno o mas usuarios sin ser seleccionados.	<ul style="list-style-type: none"> <li>- Se marcan los usuarios que se desean eliminar con un solo clic.</li> <li>- Se selecciona en la barra de herramientas el ícono <b>Eliminar usuario (s)</b>.</li> <li>- Se muestra un mensaje de información: "Debe seleccionar el usuario que desea eliminar."</li> </ul>
--	--	--	---

1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	N/A	N/A	N/A	N/A

## 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Eliminar usuario (s)	Se elimina el usuario correspondiente y se actualizan los datos del mismo.	
EP 2	Eliminar usuarios (s)	Se eliminan los usuarios correspondientes y se actualizan los datos de los mismos.	
EP 3	Eliminar uno o más usuarios sin ser seleccionados.	Se muestra un mensaje de información: “Debe seleccionar el usuario que desea eliminar.”, luego de marcado el usuario se elimina y se actualiza la información del mismo.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_ Exportar usuario (s).****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Debe existir al menos un usuario creado.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Exportar usuario (s).	Se exporta el o los usuarios creados para un documento pdf.	EP 1.1: Exportar usuario (s).	<ul style="list-style-type: none"> <li>– Se selecciona en la barra de herramientas el ícono <b>Exportar usuario (s)</b>.</li> <li>– Se selecciona el directorio donde se va a guardar el documento pdf.</li> <li>– Se acepta el mensaje de acción realizado satisfactoriamente.</li> </ul>

**1.1.1 Descripción de variable.**

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	N/A	N/A	N/A	N/A

**1.1.2 Juegos de datos a probar**

<b>Id del escenario</b>	<b>Escenario</b>	<b>Respuesta del sistema</b>	<b>Resultado de la prueba</b>
EP 1	Exportar usuario (s)	Se exporta el o los usuarios creados a un documento pdf.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_ Asignar\_ notificar\_ traza (s) \_usuario (s).****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Debe existir al menos un usuario creado.
- Debe existir al menos una traza previamente seleccionada.
- Deben estar establecidos los parámetros de configuración.

## 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Asignar y notificar traza (s) a usuario.	Se asigna y se notifica la (s) traza (s) seleccionada (s) al usuario elegido.	EP 1.1: Asignar y notificar traza (s) a usuario.	<ul style="list-style-type: none"> <li>– Se selecciona el usuario que se encargará de gestionar la traza con un solo clic.</li> <li>– Se selecciona en la barra de herramientas el ícono <b>Asignar y notificar traza (s)</b>.</li> <li>– Se acepta el mensaje de acción realizado satisfactoriamente.</li> </ul>

		<p>EP 1.2: Asignar y notificar traza (s) a usuario sin haber establecido los parámetros de configuración.</p>	<ul style="list-style-type: none"> <li>- Se selecciona el usuario que se encargará de gestionar la traza con un solo clic.</li> <li>- Se selecciona en la barra de herramientas el ícono <b>Asignar y notificar traza (s)</b>.</li> <li>- Se muestra un mensaje "Debe configurar los parámetros".</li> </ul>
--	--	---	--

1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	N/A	N/A	N/A	N/A

**1.1.2 Juegos de datos a probar**

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Asignar y notificar traza (s) a usuario.	Se asigna y se notifica la traza seleccionada al usuario elegido.	
EP 2	Asignar y notificar traza (s) a usuario sin haber establecido los parámetros de configuración.	Se asigna y se notifica la traza seleccionada al usuario elegido.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

b) Escenario: Parámetros de configuración.

### DCP\_ Parámetros\_ configuración

#### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.

#### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Parámetros de configuración.	Se establecen los parámetros de configuración necesarios para la gestión de las trazas.	EP 1.1: Parámetros de configuración.	<p>– Se selecciona en el menú principal la opción <b>Gestión de soporte/Parámetros de configuración</b></p> <p>– Se introducen los datos (Asunto del correo, Dirección smtp del servidor, Cuerpo del mensaje.)</p> <p>– Se pulsa el botón <b>Aceptar.</b></p>

		<p>EP 1.2: Parámetros de configuración dejando al menos un campo vacío.</p>	<ul style="list-style-type: none"> <li>- Se selecciona en el menú principal la opción <b>Gestión de soporte/Parámetros de configuración</b></li> <li>- Se introducen los datos (Asunto del correo, Dirección smtp del servidor, Cuerpo del mensaje.)</li> <li>- Se pulsa el botón <b>Aceptar.</b></li> <li>- Se acepta el mensaje de Aviso "Debe llenar todos los campos".</li> <li>- Se llenan todos los campos.</li> </ul>
--	--	---	--

**1.1.1 Descripción de variable.**

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	Asunto del correo	Campo de texto	No	Una combinación de letras.
2	Dirección smtp del servidor	Campo de texto	No	Una combinación números.
3	Cuerpo del mensaje	Campo de texto	No	Una combinación de letras.

## 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Variable 1 Nombre	Variable 2 Apellidos	Variable N Categoría	Respuesta del sistema	Resultado de la prueba
EP 1	Parámetros de configuración	V(Trazas)	V(10.12.62.208)	V(Trazas para gestionar)	Se establecen los parámetros de configuración y se actualizan los datos.	
EP 2	Adicionar usuario dejando al menos un campo vacío.	V(Trazas)	V(10.12.62.208)	I( )	Muestra un mensaje de campo obligatorio "Debe llenar todos los campos", y mantiene la interfaz abierta para establecer los parámetros de configuración.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

c) Escenario: Gestionar trazas.

### DCP\_ Importar\_trazas.

#### 1. Condiciones de ejecución:

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.

#### 1.1 Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Importar trazas.	Se importan las trazas y se muestran en la interfaz correspondiente.	EP 1.1: Importar trazas.	<ul style="list-style-type: none"> <li>– Se selecciona en el menú principal <b>Herramientas/Importar trazas.</b></li> <li>– Se selecciona el directorio donde se encuentra el xml de trazas.</li> </ul>

#### 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	N/A	N/A	N/A	N/A

**1.1.2 Juegos de datos a probar**

<b>Id del escenario</b>	<b>Escenario</b>	<b>Respuesta del sistema</b>	<b>Resultado de la prueba</b>
EP 1	Importar trazas.	Se importan las trazas y se muestran en una tabla de la interfaz correspondiente.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_ Exportar\_trazas.****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Debe existir al menos una traza importada.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Exportar trazas	Se exportan las trazas importadas en un documento pdf para el directorio especificado.	EP 1.1: Exportar trazas.	<ul style="list-style-type: none"> <li>– Se selecciona en la barra de herramientas el ícono <b>Exportar trazas</b>.</li> <li>– Se selecciona el directorio donde se va a guardar el documento pdf.</li> <li>– Se acepta el mensaje de acción realizado satisfactoriamente.</li> </ul>

## 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	N/A	N/A	N/A	N/A

## 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Exportar trazas	Se exportan las trazas importadas a un documento	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_ Asignar\_ usuario\_ traza (s).****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Debe existir al menos una traza importada.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Asignar usuario a la (s) traza (s).	Se asigna la(s) traza (s) seleccionada (s) a un usuario específico.	EP 1.1: Asignar usuario a la (s) traza (s).	<ul style="list-style-type: none"> <li>– Se selecciona la traza que se desea gestionar.</li> <li>– Se selecciona en la barra de herramientas el ícono <b>Asignar usuario</b>.</li> <li>– Se muestra la interfaz correspondiente para asignar el usuario.</li> </ul>

		EP 1.2: Asignar más de una traza a un usuario.	<ul style="list-style-type: none"> <li>- Se seleccionan las trazas que se desean gestionar.</li> <li>- Se selecciona en la barra de herramientas el ícono <b>Asignar usuario</b>.</li> <li>- Se muestra la interfaz correspondiente para asignar el usuario.</li> </ul>
--	--	--	---

**1.1.1 Descripción de variable.**

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	N/A	N/A	N/A	N/A

**1.1.2 Juegos de datos a probar**

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Asignar usuario a la (s) traza (s)	Se asigna la traza seleccionada a un usuario específico.	
EP 2	Asignar más de una traza a un usuario.	Se asignan las trazas seleccionadas a un usuario específico.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede.

**DCP\_ Mostrar\_ trazas asignadas.****1. Condiciones de ejecución:**

- Debe estar instalado el Microsoft.NET Framework versión 2.0 u otra superior.
- Debe existir al menos una traza importada.
- Debe existir al menos un usuario creado.
- Deben existir al menos un usuario gestionando como mínimo una traza.

**1.1 Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Mostrar trazas asignadas.	Se muestran las trazas asignadas según el usuario seleccionado en la interfaz correspondiente.	EP 1.1: Mostrar trazas asignadas.	<ul style="list-style-type: none"> <li>– Se selecciona en el menú principal la opción <b>Herramientas/Mostrar trazas asignadas.</b></li> <li>– Se selecciona con solo clic en la interfaz correspondiente el usuario, al cual se desea saber las trazas asignadas.</li> </ul>

## 1.1.1 Descripción de variable.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción
1	N/A	N/A	N/A	N/A

## 1.1.2 Juegos de datos a probar

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1	Mostrar trazas asignadas.	Se muestran las trazas asignadas en la interfaz correspondiente.	

**Leyenda:** V, I, o N/A. V indica válido, I indica inválido, y N/A no procede

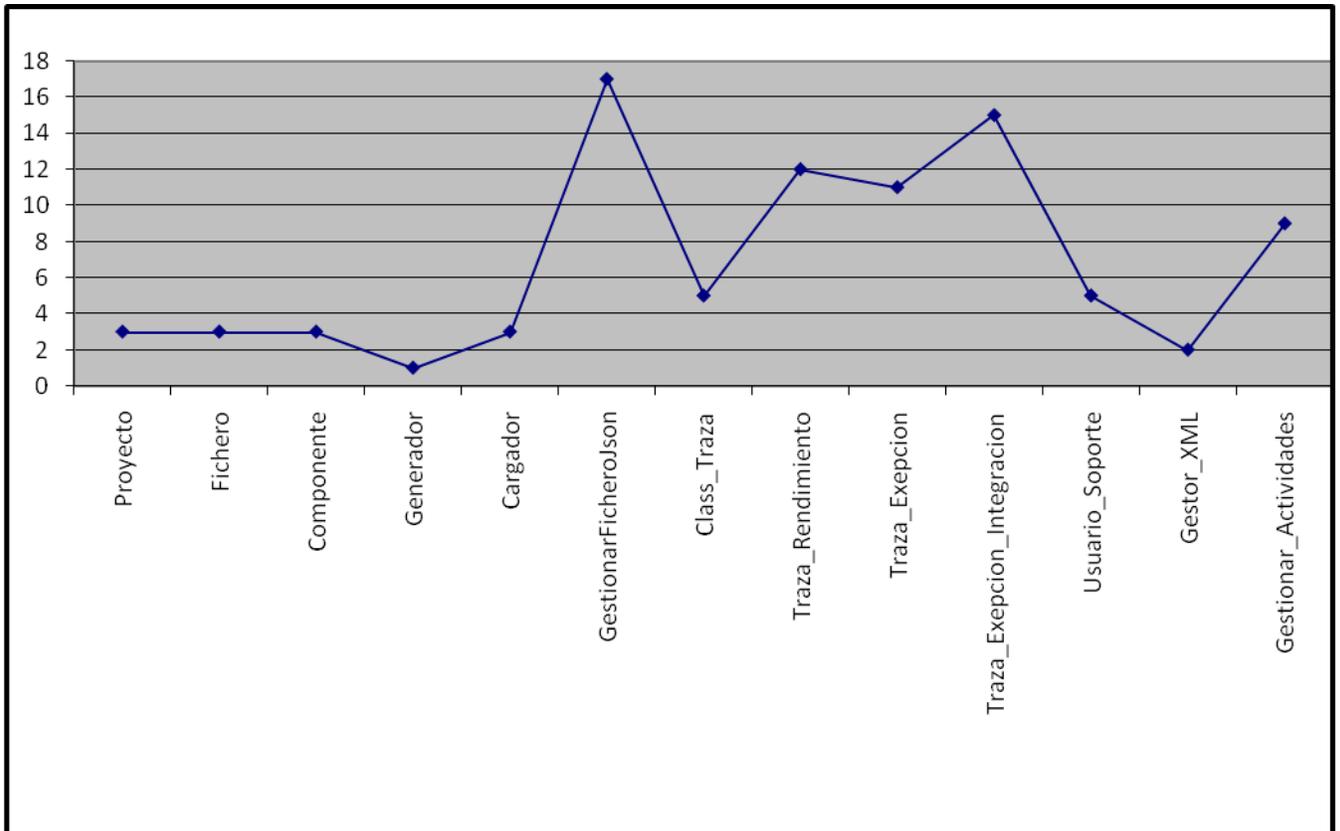
### Anexo 8: Instrumento de medición de la métrica de Tamaño de Clases (TC).

	Categoría	Criterio
Responsabilidad	Baja	$\leq$ Prom.
	Media	Entre Prom. y $2^*$ Pom.
	Alta	$> 2^*$ Prom.
Complejidad implementación	Baja	$\leq$ Prom.
	Media	Entre Prom. y $2^*$ Pom.
	Alta	$> 2^*$ Prom.
Reutilización	Baja	$> 2^*$ Prom.
	Media	Entre Prom. y $2^*$ Pom.
	Alta	$\leq$ Prom.

**Figura 46:** Rango de valores para la evaluación técnica de los atributos de calidad (Responsabilidad, Complejidad de implementación y Reutilización) relacionados con la métrica TC.

Clase	Cantidad de Procedimientos	Responsabilidad	Complejidad	Reutilización
Proyecto	3	Baja	Baja	Alta
Fichero	3	Baja	Baja	Alta
Componente	3	Baja	Baja	Alta
Generador	1	Baja	Baja	Alta
Cargador	3	Baja	Baja	Alta
GestionarFicheroJson	17	Alta	Alta	Baja
Class_Traza	5	Baja	Baja	Alta
Traza_Rendimiento	12	Media	Media	Media
Traza_Exepcion	11	Media	Media	Media
Traza_Exepcion_Integracion	15	Alta	Alta	Baja
Usuario_Soporte	5	Baja	Baja	Alta
Gestor_XML	2	Baja	Baja	Alta
Gestionar_Actividades	9	Media	Media	Media

**Tabla 41:** Resultados de la evaluación de la métrica TC y su influencia en los atributos de calidad (Responsabilidad, Complejidad de Implementación y Reutilización).



**Figura 47: Gráfica de los resultados de la evaluación de la métrica TC y su influencia en los atributos de calidad (Responsabilidad, Complejidad de Implementación y Reutilización)**

## Anexo 9: Instrumento de medición de la métrica respuesta para una clase

	Categoría	Criterio
Acoplamiento	Ninguno	0
	Bajo	1
	Medio	2
	Alto	>2
	Categoría	Criterio
Complejidad Mant.	Baja	$\leq$ Prom.
	Media	Entre Prom. y $2 \times$ Prom.
	Alta	$> 2 \times$ Prom.
	Categoría	Criterio
Reutilización	Baja	$> 2 \times$ Prom.
	Media	Entre Prom. y $2 \times$ Prom.
	Alta	$\leq$ Prom.
	Categoría	Criterio
Cantidad de Pruebas	Baja	$\leq$ Prom.
	Media	Entre Prom. y $2 \times$ Prom.
	Alta	$> 2 \times$ Prom.

(RPC).

**Figura 48: Rango de valores de para la evaluación técnica de los atributos de calidad (Acoplamiento, Complejidad de Mantenimiento, Reutilización y Cantidad de Pruebas) relacionados con la métrica RPC.**

Clase	Cantidad de Relaciones de Uso	Acoplamiento	Complejidad Mant.	Reutilización	Cantidad de Pruebas
Proyecto	0	Ninguno	Baja	Alta	Baja
Fichero	0	Ninguno	Baja	Alta	Baja
Componente	0	Ninguno	Baja	Alta	Baja
Generador	1	Bajo	Media	Media	Media
Cargador	1	Bajo	Media	Media	Media

<b>GestionarFicheroJson</b>	3	Alto	Alta	Baja	Alta
<b>Class_Traza</b>	0	Ninguno	Baja	Alta	Baja
<b>Traza_Rendimiento</b>	1	Bajo	Media	Media	Media
<b>Traza_Exepcion</b>	1	Bajo	Media	Media	Media
<b>Traza_Exepcion_Integracion</b>	1	Bajo	Media	Media	Media
<b>Usuario_Soporte</b>	0	Ninguno	Baja	Alta	Baja
<b>Gestor_XML</b>	1	Bajo	Media	Media	Media
<b>Gestionar_Actividades</b>	1	Bajo	Media	Media	Media

**Tabla 42: Resultados de la evaluación de la métrica RPC y su influencia en los atributos de calidad (Acoplamiento, Complejidad de Mantenimiento, Reutilización y Cantidad de Pruebas).**

**Anexo 10: Instrumento de medición de la métrica Árbol de profundidad de Herencia (APH).**

Clase	Clase padre	Niveles del árbol de herencia
Proyecto		0
Fichero		0
Componente		0
Generador		0
Cargador		0
GestionarFicheroJson		0
Class_Traza		1
Traza_Rendimiento	Class_Traza	2
Traza_Exepcion	Class_Traza	2
Traza_Exepcion_Integracion	Class_Traza	2
Usuario_Soporte		0
Gestor_XML		0
Gestionar_Actividades		0

**Tabla 43: Resultados de la evaluación de la métrica APH.**

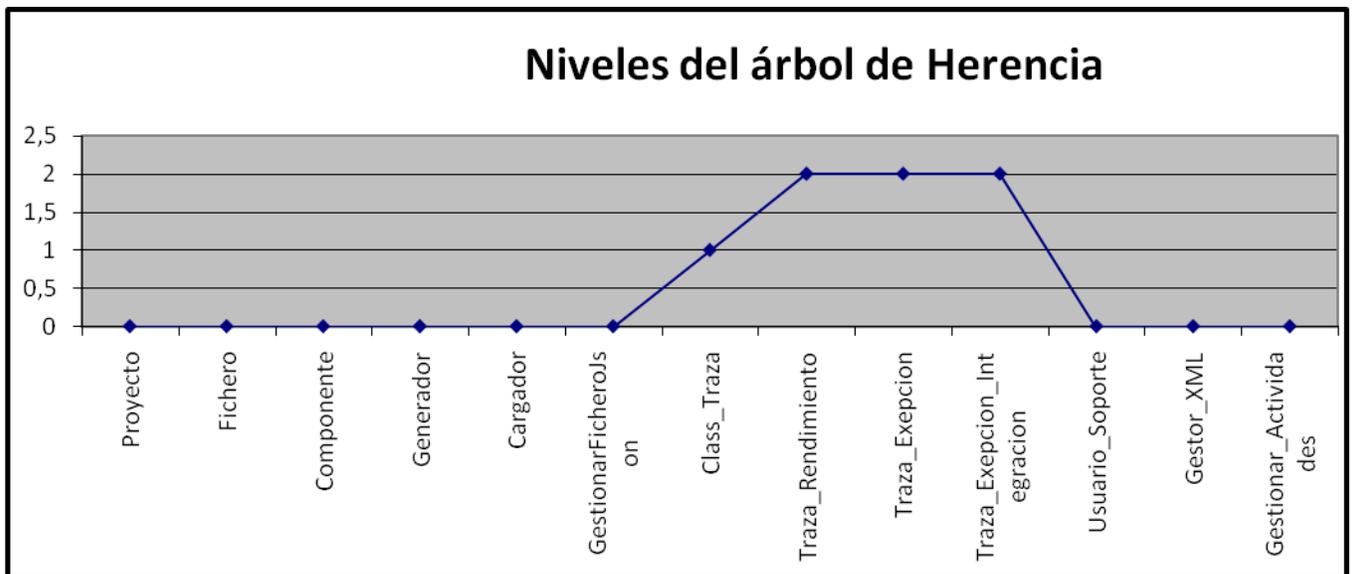


Figura 49: Gráfica de los resultados de la evaluación de la métrica APH.

### Anexo 11: Instrumento de medición de la métrica Número de Descendientes (NDD).

	Categoría	Criterio
Reutilización	Baja	$\leq$ Prom.
	Media	Entre Prom. y $2 \times$ Prom.
	Alta	$> 2 \times$ Prom.

	Categoría	Criterio
Abstracción	Indefinida	$> 5$
	Afectada	Entre 2 y 5
	Definida	$\leq 2$

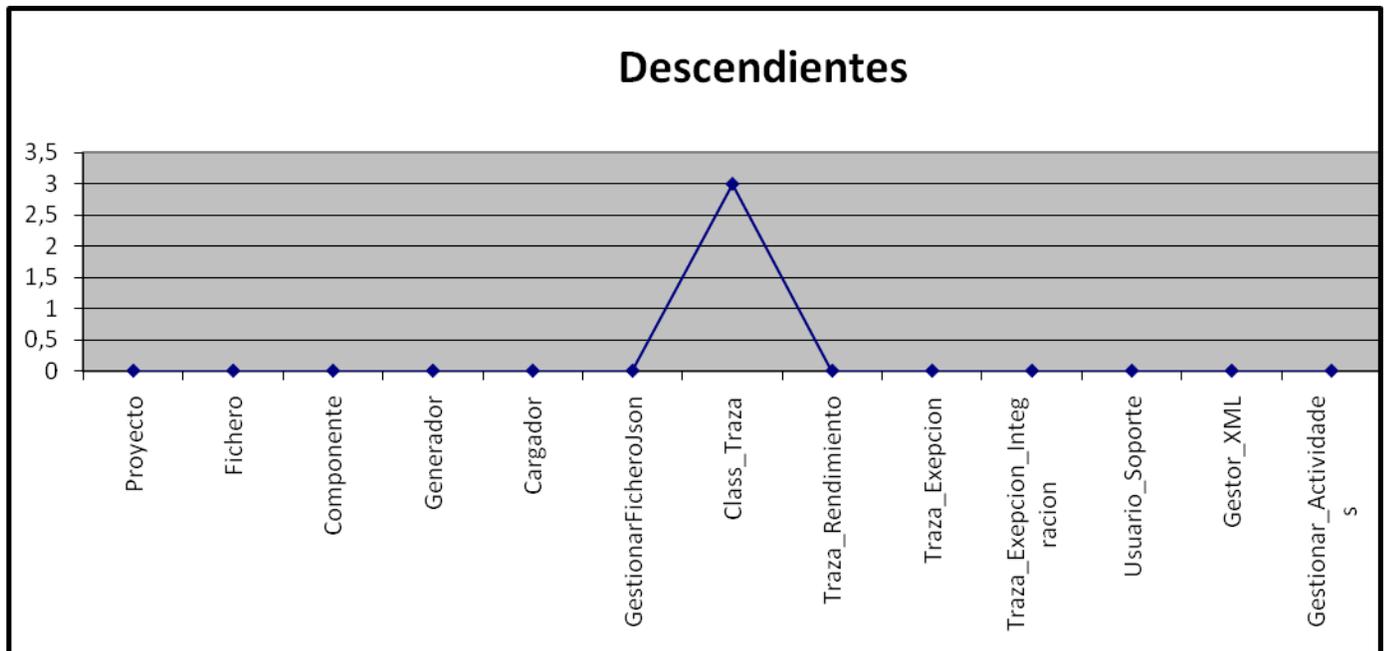
	Categoría	Criterio
Cohesión	Baja	$> 5$
	Media	Entre 2 y 5
	Alta	$\leq 2$

	Categoría	Criterio
Cantidad de Pruebas	Baja	$\leq 2$
	Media	Entre 2 y 5
	Alta	$> 5$

**Figura 50: Rango de valores para la evaluación técnica de los atributos de calidad (Reutilización, Abstracción del diseño, Nivel de Cohesión y Cantidad de Pruebas) relacionados con la métrica NDD.**

Clase	Clase padre	Número de descendientes	Reutilización	Abstracción de la clase base	Cohesión de la JC	Cantidad de Pruebas
Proyecto		0	Baja	Definida	Alta	Baja
Fichero		0	Baja	Definida	Alta	Baja
Componente		0	Baja	Definida	Alta	Baja
Generador		0	Baja	Definida	Alta	Baja
Cargador		0	Baja	Definida	Alta	Baja
GestionarFicheroJson		0	Baja	Definida	Alta	Baja
Class_Traza		3	Alta	Afectada	Media	Media
Traza_Rendimiento	Class_Traza	0	Baja	Definida	Alta	Baja
Traza_Exepcion	Class_Traza	0	Baja	Definida	Alta	Baja
Traza_Exepcion_Integracion	Class_Traza	0	Baja	Definida	Alta	Baja
Usuario_Soporte		0	Baja	Definida	Alta	Baja
Gestor_XML		0	Baja	Definida	Alta	Baja
Gestionar_Actividades		0	Baja	Definida	Alta	Baja

**Tabla 44: Resultados de la evaluación de la métrica NDD y su influencia en los atributos de calidad (Reutilización, Abstracción del diseño, Nivel de Cohesión y Cantidad de Pruebas).**



**Figura 51:** Representación de los resultados obtenidos al aplicar los instrumentos que evalúan la métrica NDD.

**Anexo 11:** Instrumento de medición de la métrica Número de Operaciones Redefinidas para una Sub-Clase (NOR).

	Categoría	Criterio
Complejidad de Mantenimiento	Baja	0
	Media	Entre 0 y 2

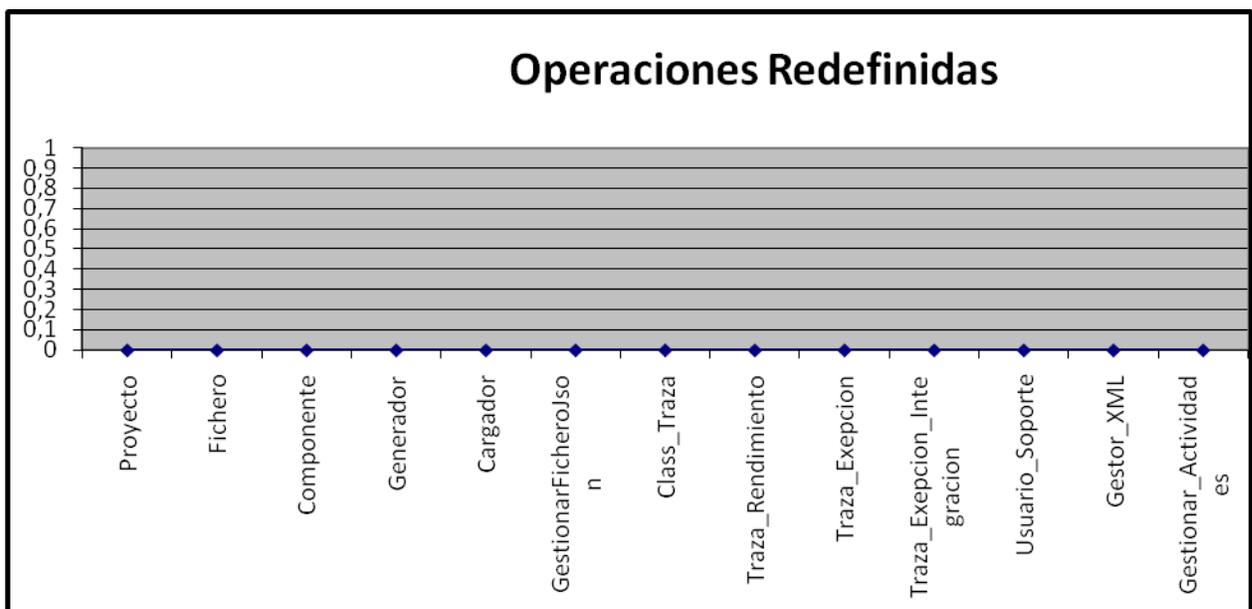
	Categoría	Criterio
Cantidad de Pruebas	Baja	0
	Media	Entre 0 y 2

	Categoría	Criterio
Violación de la Abstracción representada por la superclase	Baja	0
	Media	Entre 0 y 2

**Figura 52:** Rango de valores para la evaluación técnica de los atributos de calidad (Abstracción del diseño, Cantidad de pruebas, Complejidad de Mantenimiento) relacionados con la métrica NOR.

Clase	Clase padre	No de Operaciones redefinidas	Calidad del Diseño	Complejidad de Mant.	Cantidad de Pruebas	Violación de ARS
Proyecto		0	Buena	Baja	Baja	Baja
Fichero		0	Buena	Baja	Baja	Baja
Componente		0	Buena	Baja	Baja	Baja
Generador		0	Buena	Baja	Baja	Baja
Cargador		0	Buena	Baja	Baja	Baja
GestionarFicheroJson		0	Buena	Baja	Baja	Baja
Class_Traza		0	Buena	Baja	Baja	Baja
Traza_Rendimiento	Class_Traza	0	Buena	Baja	Baja	Baja
Traza_Exepcion	Class_Traza	0	Buena	Baja	Baja	Baja
Traza_Exepcion_Integracion	Class_Traza	0	Buena	Baja	Baja	Baja
Usuario_Soporte		0	Buena	Baja	Baja	Baja
Gestor_XML		0	Buena	Baja	Baja	Baja
Gestionar_Actividades		0	Buena	Baja	Baja	Baja

**Tabla 45: Resultados de la evaluación de la métrica NOR y su influencia en los atributos de calidad (Abstracción del diseño, Cantidad de pruebas, Complejidad de Mantenimiento).**



**Figura 53: Representación de los resultados obtenidos al aplicar los instrumentos que evalúan la métrica NOR.**