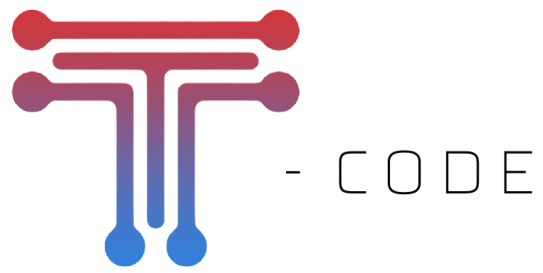
Informe de Revisión Técnica Formal

Vesta Risk Manager

T-Code

Agustín Collareda, Cintia Hernandez, Hugo Frey



Una revisión técnica formal (RTF) es una actividad de garantía de calidad de los sistemas de información. Los objetivos de la RTF son:

Describir errores en la función, la lógica o la implementación de cualquier representación de los sistemas de información.

Verificar que los sistemas bajo revisión alcancen sus requisitos.

Garantizar que los sistemas han sido representados de acuerdo con ciertos estándares predefinidos.

Conseguir un sistema desarrollado en forma uniforme.

Hacer que los proyectos sean más manejables.



Tabla de contenido

[Producto revisado 4](#_Toc188274859)

[Nombre del Producto revisado 4](#_Toc188274860)

[Participantes de la revisión 4](#_Toc188274861)

[Técnica utilizada 4](#_Toc188274862)

[Objetivos de la RTF 4](#_Toc188274863)

[Problemas detectados 5](#_Toc188274864)

[Claridad 5](#_Toc188274865)

[Sugerencia de corrección 6](#_Toc188274866)

[Completitud 6](#_Toc188274867)

[Sugerencia de corrección 6](#_Toc188274868)

[Evaluación 6](#_Toc188274869)

[Estado actual del Producto 6](#_Toc188274870)

[Acciones a tomar 6](#_Toc188274871)

Informe de Revisión Técnica Formal

Producto revisado

Nombre del Producto revisado

El producto que se va a revisar es: Implementación backend. La versión que se va a revisar es la del 15 de noviembre de 2024 y corresponde al área de Análisis y Diseño.

Participantes de la revisión

Los participantes de la RTF son:

* Revisor – Hugo Frey

Técnica utilizada

La revisión se realizó una lista de chequeo con las siguientes preguntas:

|  |  |  |  |
| --- | --- | --- | --- |
| Criterio | Pregunta | Si/No | Comentarios |
| PSR-1 | ¿Los archivos PHP usan solo etiquetas <?php o <?=, sin otras como <?? |  |  |
| ¿Los archivos terminan con una nueva línea vacía? |  |  |
| ¿Los archivos usan codificación UTF-8 sin BOM? |  |  |
| ¿Cada clase está en su propio archivo? |  |  |
| ¿Los nombres de las clases siguen el formato StudlyCaps (por ejemplo, MiClase)? |  |  |
| ¿Las constantes de clase están declaradas en mayúsculas con guiones bajos (por ejemplo, CONSTANTE\_EJEMPLO)? |  |  |
| ¿Los nombres de los métodos siguen el formato camelCase (por ejemplo, miMetodo)? |  |  |
| PER Coding Style 2.0 | ¿Se usa indentación de 4 espacios (no tabs)? |  |  |
| ¿Las líneas de código no superan los 80 caracteres (idealmente) o 120 como máximo? |  |  |
| ¿No se declara más de una variable o propiedad por linea? |  |  |
| ¿No hay espacios al final de las líneas? |  |  |
| ¿Las llaves {} están en la misma línea para estructuras de control (if, for, foreach, etc.)? |  |  |
| ¿Las llaves {} están en la línea siguiente para definiciones de clases y métodos? |  |  |
| ¿Hay un espacio después de palabras clave como if, else, for, foreach, etc.? |  |  |
| ¿No hay espacios antes de los paréntesis de apertura en funciones o métodos? |  |  |
| ¿Hay un espacio después de comas en listas de parámetros? |  |  |
| ¿Los operadores están rodeados de espacios (por ejemplo, $a = $b + $c;)? |  |  |
| ¿La concatenación de cadenas usa espacios alrededor del punto (por ejemplo, $a = 'Hola' . 'Mundo';)? |  |  |
| ¿No hay espacios entre el nombre de la función y el paréntesis de apertura? |  |  |
| ¿No hay espacios dentro de los paréntesis en llamadas a funciones? |  |  |
| ¿Se usa la sintaxis corta [] para arrays en lugar de array()? |  |  |
| ¿No hay espacios después de la apertura [ ni antes del cierre ] en arrays? |  |  |
| ¿La palabra clave extends e implements está en la misma línea que el nombre de la clase? |  |  |
| ¿Los métodos y funciones tienen visibilidad declarada (public, protected, private)? |  |  |
| ¿No hay espacios antes de los dos puntos en la declaración de tipos de retorno? |  |  |
| Alineación con los requerimientos | ¿Se han mapeado correctamente los datos necesarios para cada funcionalidad descrita en los requerimientos? |  |  |
| Validación de tipos de datos | ¿Los tipos de datos asignados a cada columna son los adecuados (por ejemplo, INT, VARCHAR, DATE, etc.)? |  |  |
| Relaciones y dependencias | ¿Se han definido correctamente las relaciones entre tablas? |  |  |

Objetivos de la RTF

En esta revisión, se buscó que la especificación de requerimientos cumpla con estos criterios: Claridad, Completitud, Consistencia, Trazabilidad y Testeabilidad para asegurar la calidad de los requerimientos como la calidad de la documentación.

Problemas detectados

Se anexa la lista de chequeo completa

|  |  |  |  |
| --- | --- | --- | --- |
| Criterio | Pregunta | Si/No | Comentarios |
| PSR-1 | ¿Los archivos PHP usan solo etiquetas <?php o <?=, sin otras como <?? |  |  |
| ¿Los archivos terminan con una nueva línea vacía? |  |  |
| ¿Los archivos usan codificación UTF-8 sin BOM? |  |  |
| ¿Cada clase está en su propio archivo? |  |  |
| ¿Los nombres de las clases siguen el formato StudlyCaps (por ejemplo, MiClase)? |  |  |
| ¿Las constantes de clase están declaradas en mayúsculas con guiones bajos (por ejemplo, CONSTANTE\_EJEMPLO)? |  |  |
| ¿Los nombres de los métodos siguen el formato camelCase (por ejemplo, miMetodo)? |  |  |
| PER Coding Style 2.0 | ¿Se usa indentación de 4 espacios (no tabs)? |  |  |
| ¿Las líneas de código no superan los 80 caracteres (idealmente) o 120 como máximo? |  |  |
| ¿No hay espacios al final de las líneas? |  |  |
| ¿Las llaves {} están en la misma línea para estructuras de control (if, for, foreach, etc.)? |  |  |
| ¿Las llaves {} están en la línea siguiente para definiciones de clases y métodos? |  |  |
| ¿Hay un espacio después de palabras clave como if, else, for, foreach, etc.? |  |  |
| ¿No hay espacios antes de los paréntesis de apertura en funciones o métodos? |  |  |
| ¿Hay un espacio después de comas en listas de parámetros? |  |  |
| ¿Los operadores están rodeados de espacios (por ejemplo, $a = $b + $c;)? |  |  |
| ¿La concatenación de cadenas usa espacios alrededor del punto (por ejemplo, $a = 'Hola' . 'Mundo';)? |  |  |
| ¿No hay espacios entre el nombre de la función y el paréntesis de apertura? |  |  |
| ¿No hay espacios dentro de los paréntesis en llamadas a funciones? |  |  |
| ¿Se usa la sintaxis corta [] para arrays en lugar de array()? |  |  |
| ¿No hay espacios después de la apertura [ ni antes del cierre ] en arrays? |  |  |
| ¿La palabra clave extends e implements está en la misma línea que el nombre de la clase? |  |  |
| ¿Las interfaces siguen el mismo formato que las clases? |  |  |
| ¿Los métodos y funciones tienen visibilidad declarada (public, protected, private)? |  |  |
| ¿No hay espacios antes de los dos puntos en la declaración de tipos de retorno? |  |  |
| ¿Los namespaces están declarados en la primera línea del archivo? |  |  |
| ¿Hay una línea en blanco después de la declaración del namespace? |  |  |
| ¿Las declaraciones use están agrupadas y ordenadas alfabéticamente? |  |  |
| ¿Hay una línea en blanco después de las declaraciones use? |  |  |
| Alineación con los requerimientos | ¿Se han mapeado correctamente los datos necesarios para cada funcionalidad descrita en los requerimientos? |  |  |
| Validación de tipos de datos | ¿Los tipos de datos asignados a cada columna son los adecuados (por ejemplo, INT, VARCHAR, DATE, etc.)? |  |  |
| Relaciones y dependencias | ¿Se han definido correctamente las relaciones entre tablas? |  |  |

Claridad

Sugerencia de corrección

Completitud

Sugerencia de corrección

Evaluación

Estado actual del Producto

Acciones a tomar