**Informe técnico sobre análisis de código estático para el software**

**Proveedores**

**Versión 1.0**

**Perito:**

**Nombre: Santiago Andrés Rodríguez Garcés**

**Cedula: 1005650954**

**Email: santiago.rodriguezg@usantoto.edu.co**



**Contenido del informe**

[1. Propósito de la evaluación 2](#_Toc43116759)

[2. Producto a evaluar 2](#_Toc43116760)

[4. Especificación de la Norma de calidad 2](#_Toc43116761)

[5. Requerimientos específicos de calidad funcionales y técnicos. 2](#_Toc43116762)

[6. Atributos de calidad. 5](#_Toc43116763)

[6.1. Asignación de puntajes para atributos de calidad. 5](#_Toc43116764)

[6.2. Atributos internos. 5](#_Toc43116765)

[6.2.1. Tamaño de sistemas y código fuente (20%) 5](#_Toc43116766)

[6.2.2. Complejidad del software (20%) 7](#_Toc43116767)

[6.2.3. Deuda técnica (20%) 8](#_Toc43116768)

[6.2.4. Seguridad (20%) 10](#_Toc43116769)

[6.3. Atributos Externos. 12](#_Toc43116770)

[6.3.1. Usabilidad (20%) 12](#_Toc43116771)

[6.3.2. Fiabilidad (30%) 13](#_Toc43116772)

[6.3.3. Mantenibilidad (50%) 13](#_Toc43116773)

[7. Recomendaciones y conclusiones 13](#_Toc43116774)

[7.1. Seguridad 13](#_Toc43116775)

[7.2. Confiabilidad 13](#_Toc43116776)

[7.3. Usabilidad 14](#_Toc43116777)

[8. Bibliografía 14](#_Toc43116778)

[9. Firma del perito. 14](#_Toc43116779)

# Propósito de la evaluación

El propósito de la evaluación del software llamado Proveedores es identificar problemas de seguridad, mantenibilidad de código, interacción del usuario con el software (usabilidad), hacer mediciones de calidad con algunas métricas como la densidad de comentarios, la complejidad cognitiva, la complejidad ciclomática, entre otros.

# Producto a evaluar

El software a evaluar es un sitio web para la gestión de proveedores, contratos y usuarios. Los usuarios quienes pueden tener roles de administrador.

El proyecto está compuesto por las siguientes carpetas y archivos:

* **Css:** Almacena los archivos CSS o de estilos.
* **Js:** Almacena los archivos JavaScripts.
* Archivos para la conexión a la base de datos, gestión de proveedores, contratos y usuarios, envío de correos electrónicos. Estos archivos son archivos PHP.

Versión 1.0.

1. **Especificación de la(s) herramientas usadas para medición**

**SonarQube**

Es una plataforma la cual permite medir y analizar la calidad de código usando diversas herramientas de análisis estático, algunas de sus características son:

* Está hecha en Java, versión 7.7.
* Detección de código duplicado, código muerto, bugs, entre otros.
* Recomendaciones de estándares de codificación.
* Medición de la complejidad ciclomática.
* Escrito en Java versión 8 y 11
* Permite analizar más de 20 lenguajes de programación, cuenta con la documentación de

reglas correspondientes para cada lenguaje.

Usa herramientas de análisis estático como:

● **Checkstyle:** Cuyo propósito es verificar el cumplimiento de las reglas de codificación,

como las buenas prácticas, complejidad ciclomática entre otras. (Márquez, 2019).

● **PMD:** Es un identificador de problemas potenciales, código duplicado, código muerto y más (Márquez, 2019).

● **Findbugs:** Se encarga de encontrar errores en cuanto a malas prácticas,

Vulnerabilidades de código y más (Márquez, 2019).

# Especificación de la Norma de calidad

La norma ISO 25000 tiene como objetivo la creación de estándares, organizar, enriquecer y unificar los procesos y requisitos del software, “La familia ISO es el resultado de la evolución de normas anteriores, proporciona una guía para los usos de estándares internacionales llamados requisitos donde la norma establece criterios para la especificación de requisitos de calidad de productos software” (excentia, 2018).

Al utilizar esta norma trae muchos beneficios al momento de definir los requerimientos de calidad para las pruebas donde se demuestra la calidad del software y permite asegurarnos de que el producto sea seguro y confiable, que tenga buena calidad.

# Requerimientos específicos de calidad funcionales y técnicos.

La empresa requiere que se evalué los siguientes requerimientos funcionales y técnicos específicos

|  |  |  |
| --- | --- | --- |
| Ítem | Requerimiento de calidad | Prioridad |
| 1 | Que el software pueda funcionar en sistemas operativos Android, MacOS, Windows XP, Windows 7 y Windows 10 (en 32 y 64 bits) | Alta |
| 2 | Que permita trabajar en forma rápida e intuitiva (cuente con ayudas visuales y auditivas interactivas en el software). | Media |
| 3 | Que tenga soporte multiidiomas, especialmente inglés y español | Media |
| 4 | Que permita adecuar su estilo de visualización para adecuarse a personas con limitaciones visuales (Ley 1680 de 20 de noviembre de 2013) | Alta |
| 5 | Implementación de Ley 1581 del 2012 – Protección de datos (HABEAS DATA) | Alta |
| 6 | Permita generar reportes en EXCEL Y PDF. | Baja |
| 7 | Funcionalidad/modulo para reportar errores técnicos o funcionales desde el software. | Media |
| 8 | Permitir acceso a 100 usuarios simultáneos | Alta |
| 9 | Tolerancia a fallos (caída de red, apagones eléctricos frecuentes). | Media |
| 10 | Integración con office | Baja |
| 11 | Cumplimiento del 80% con el estándar OWASP, priorizando en las vulnerabilidades de robo de información, XSS, SQL injection y ransomware. | Alta |
| 12 | Capacidad de respaldo y recuperación de información desde el software. | Media |
| 13 | El software debe demandar mínimos recursos de hardware (cpu Intel celeron, 2 gigas de Ram) | Media |

1. El software es un sitio web por lo que funciona en todas las versiones de los sistemas operativos y al tener un diseño responsivo (uso de materialize como framework CSS) se adapta en dispositivos móviles (Android y IOS).

**Cumplimiento: 100%**

1. El sitio web permite trabajar de forma rápida ya que no tiene funcionalidades complejas. Se deben mejorar las ayudas visuales para los usuarios.

**Cumplimiento: 60%**

**Recomendaciones:**

* Agregar nuevas ayudas visuales para indicar al usuario las opciones del sitio.
* El tiempo estimado para estos cambios es de 1 mes ya que el sitio no es grande.
* Se necesitará un programador frontend y backend con un sueldo de 2.600.000 cada uno, para un total de 5.200.000.

1. La gran mayoría del sitio está disponible en el idioma en español a excepción del pie de pagina que tiene unas partes en inglés.

**Cumplimiento: 20%**

**Recomendaciones:**

* Implementar la traducción completa del sitio web a inglés. Para ello se debería usar un framework como Laravel o Symfony que permiten implementar esta funcionalidad con más rapidez y seguridad.
* Se estima un tiempo aproximado de 3 meses para la implementación del soporte multi idiomas.
* Se necesita un desarrollador frontend y backend para un total de 15.600.000.

1. El sitio web usa las etiquetas correspondientes en los campos de los formularios y al utilizar materialize como framework CSS le da un tamaño adecuado a la letra y facilita que el sitio sea soportado por los lectores de pantalla.

**Cumplimiento: 70%**

**Recomendaciones:**

* Se debe implementar todo el sitio web con soporte para lectores de pantalla.
* Se estima un tiempo aproximado de 1 mes en el cual se necesita un ingeniero con el sueldo de 2.600.000.

1. No cumple con este ítem.

**Cumplimiento: 0%**

**Recomendaciones:**

* Encriptar las contraseñas de los usuarios.
* Validar los formularios.
* Mejorar la seguridad para protegerse de ataques comunes de robo de información.
* Mejorar los permisos de las personas administradoras que pueden acceder a la información.
* Para la implementación de estas mejoras se estima un tiempo de 2 meses.
* No tiene política de privacidad y términos de condiciones.
* Se necesita un desarrollador frontend y backend para un total de 10.400.000.

1. El sitio web no tiene funcionalidades para generar reportes en Excel o PDF.

**Cumplimiento: 0%**

**Recomendaciones:**

* Implementar funcionalidades de generar reportes es útil para una mejor administración del sitio web y experiencia del usuario.
* Se estima un tiempo de 1 mes y un desarrollador backend con un sueldo de 2.600.000.

1. No cumple con este ítem.

**Cumplimiento: 0%**

**Recomendaciones:**

* No tiene paginas para los errores HTTP comunes como lo son 404 y 500.
* No se muestran mensajes de error cuando se ingresa información errónea en los formularios.
* Implementar funcionalidades que permitan generar informes de errores para hacer una mejor gestión del software y evitar problemas de seguridad.
* Se necesita un desarrollador backend con un sueldo de 5.200.000 por dos meses.

1. No cumple debido a que el sitio web está en local y no se puede tener 100 usuarios simultáneamente.

**Cumplimiento: 0%**

**Recomendaciones:**

* Es recomendable comprar un hosting y un servidor.
* Se necesita de los siguientes recursos: Un desarrollador frontend y backend, adquirir un hosting, un dominio.
* Se estima un costo de 5.200.000 para los desarrolladores, 300.000 para el Hosting y 35.000 para el dominio.
* El tiempo para implementar estas funcionalidades es de un 1 mes.

1. Lo mismo del ítem anterior.
2. El sitio web no tiene funcionalidades para integración con office.

**Cumplimiento: 0%**

**Recomendaciones:**

* Implementar funcionalidades que permitan generar informes en formatos PDF Y Excel para hacer una mejor gestión del software.
* Se necesita un desarrollador backend con un sueldo de 5.200.000 por dos meses.

1. El Sitio no cuenta con protección de SQL injection, Cross-site request forgery (CSRF) y ataques comunes de robo de información.

**Cumplimiento: 0%**

**Recomendaciones:**

* Validar formularios.
* Encriptar contraseñas y datos sensibles.
* Implementar buenas prácticas de programación y estándares para mitigar los ataques.
* Se necesita un desarrollador backend y frontend por 2 meses para un total de 5.200.000.

1. El sitio web no cuenta con respaldo de la información.

**Cumplimiento: 0%**

**Recomendaciones:**

* Implementar una base de datos de respaldo de la información. Se necesita un Ingeniero backend con un sueldo de 2.600.000.

1. Si cumple con el consumo de recursos de hardware obteniendo la siguiente información:

* Memoria 231,1 MB
* Disco Duro 0.1 MB\*seg
* CPU 8%

**Cumplimiento: 100%**

# Atributos de calidad.

Los atributos de calidad que se utilizaran para la evaluación del Software Proveedores, de acuerdo a lo especificado en el siguiente cuadrado:

*Tabla 1 Atributos de calidad*

|  |  |
| --- | --- |
| ATRIBUTOS INTERNOS | Características del software que determinan su habilidad para satisfacer las necesidades propias e implícitas. |
| ATRIBUTOS EXTERNOS | Características del software que determinan su habilidad para satisfacer las necesidades explicitas e implícitas. |
| ATRIBUTOS EN USO | Características del software que determinan los requerimientos de los usuarios finales de manera que satisfagan sus necesidades. |

## Asignación de puntajes para atributos de calidad.

Los puntajes establecidos a los atributos de calidad seleccionados de acuerdo a las necesidades, se muestran en la siguiente tabla:

*Tabla 2, Asignación de pesos sobre la medición de atributos.*

|  |  |
| --- | --- |
| Tipo de atributo | Puntaje |
| Atributos internos | 65 |
| Atributos externos | 35 |
| Total | 100 |

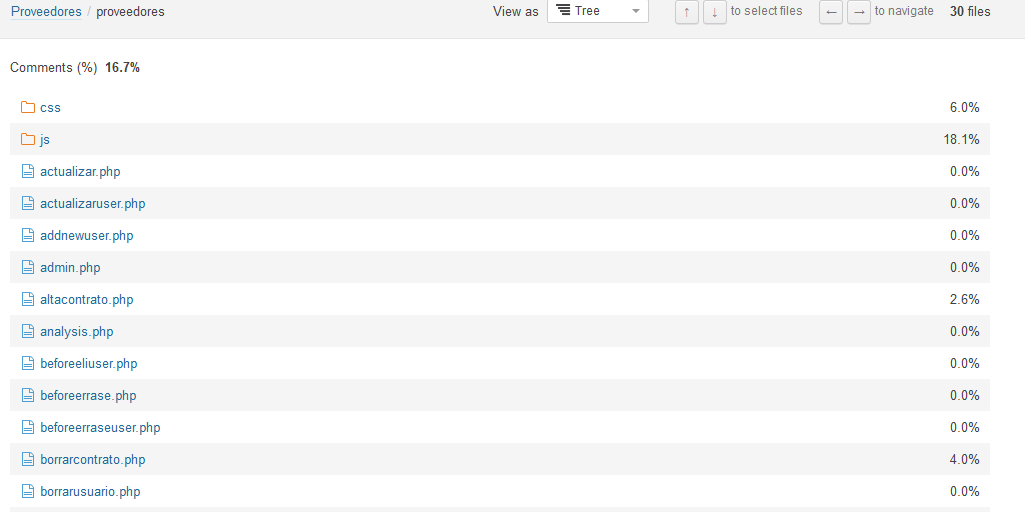
## Atributos internos.

### Tamaño de sistemas y código fuente (20%)

* **Total de líneas de código**: 18.640.

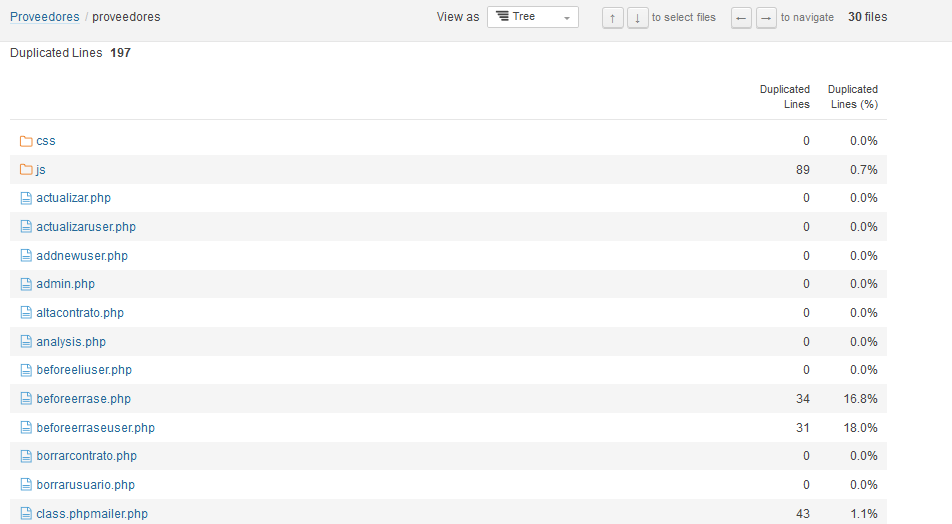


* **Densidad de comentarios.**



El proyecto tiene una densidad de comentarios de 16.7% la cual se concentra en las carpetas de css (6%) y js (18.1%).

* **Duplicidad del código.**

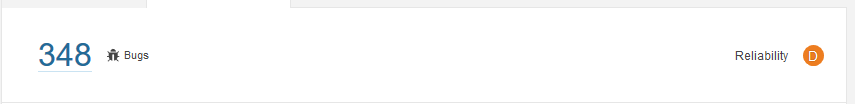


**197** líneas de código duplicadas, que se divide en la carpeta js con 89 líneas, los archivos beforeerrase.php, beforeerraseuser.php y class.phpmailer.php con 34, 31 y 43 líneas respectivamente.



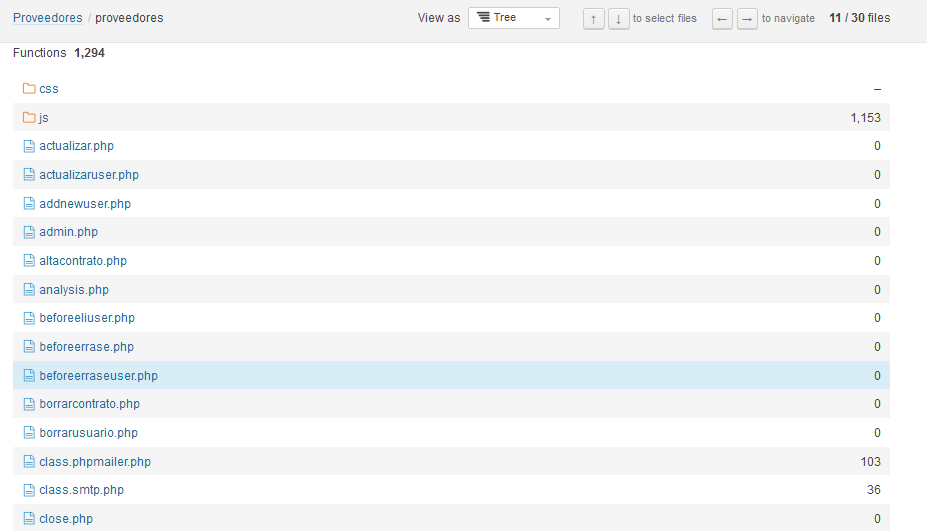
Código duplicado en los archivos beforeerrase.php y beforeerraseuser.php.

* **Bugs**

****

Se encontraron 348 bugs.

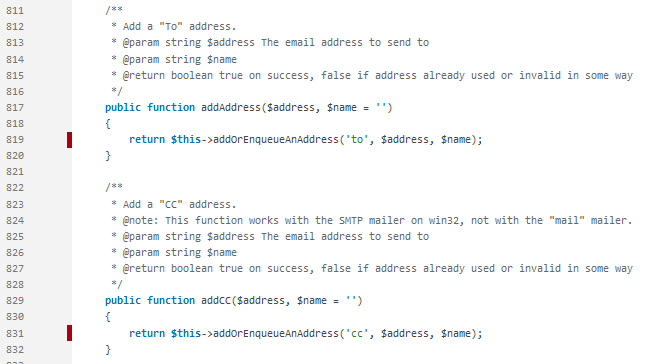
* **Cantidad de métodos**



El proyecto tiene 1.294 funciones o métodos repartidas de la siguiente manera:

* + Js: 1.153
  + class.phpmailer.php: 103
  + class.smtp.php: 36
  + PHPMailerAutoload.php: 2

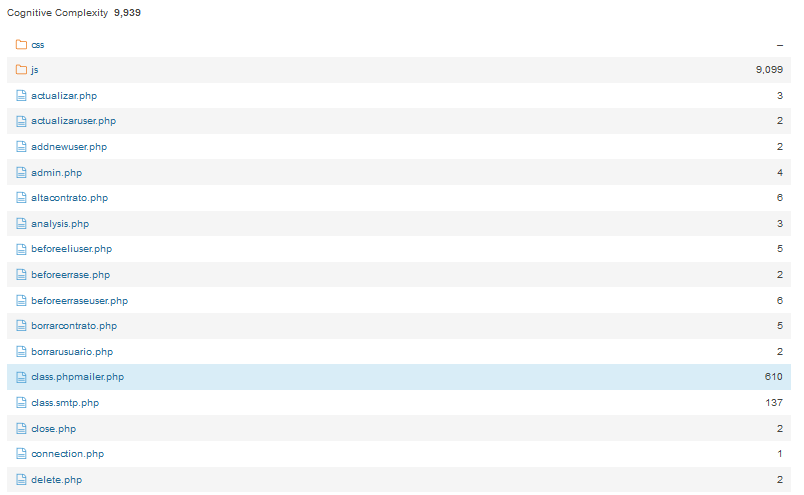
Cabe resaltar que los archivos PHP mencionados anteriormente corresponden a paquetes de terceros.



Funciones encontradas. Se evidencia que tienen una documentación adecuada y cumplen con una tarea específica. Todos los métodos cumplen con un numero de líneas menor a 30.

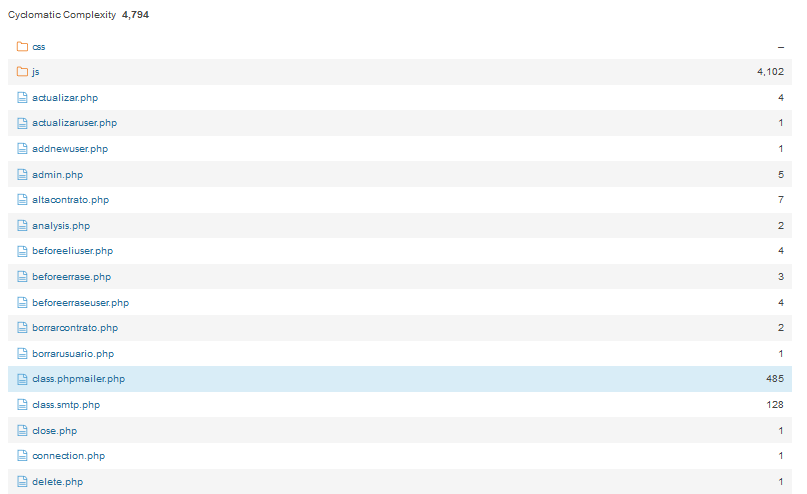
### Complejidad del software (20%)

* **Complejidad cognitiva**



Con un total de 9.939. El modulo con mayor complejidad es js con 9.099.

* **Complejidad ciclomática**



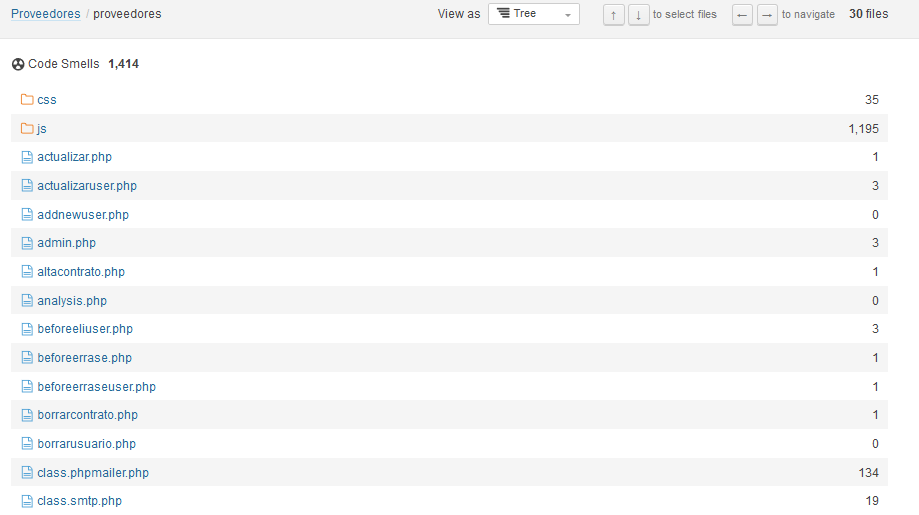
Con un total de 4794 distribuidos de la siguiente manera (archivos con mayor complejidad):

* + Js: 4102
  + class.phpmailer.php: 485
  + class.smtp.php: 128

Los archivos mencionados anteriormente corresponden a paquetes de terceros. La complejidad de los archivos propios es baja.

### Deuda técnica (20%)

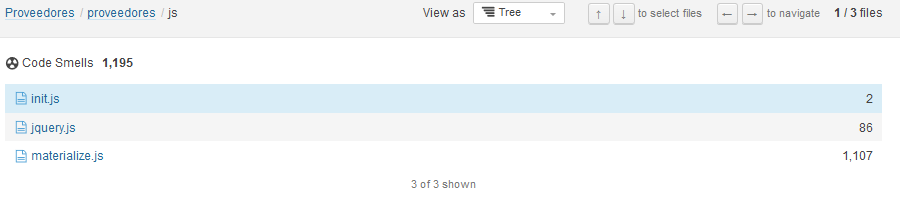
* **Código smell**

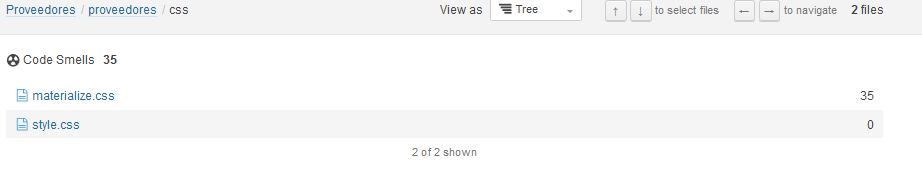


Con un total de 1.414 code smell.



Code smell que indica eliminar la variable ya que no se usa.



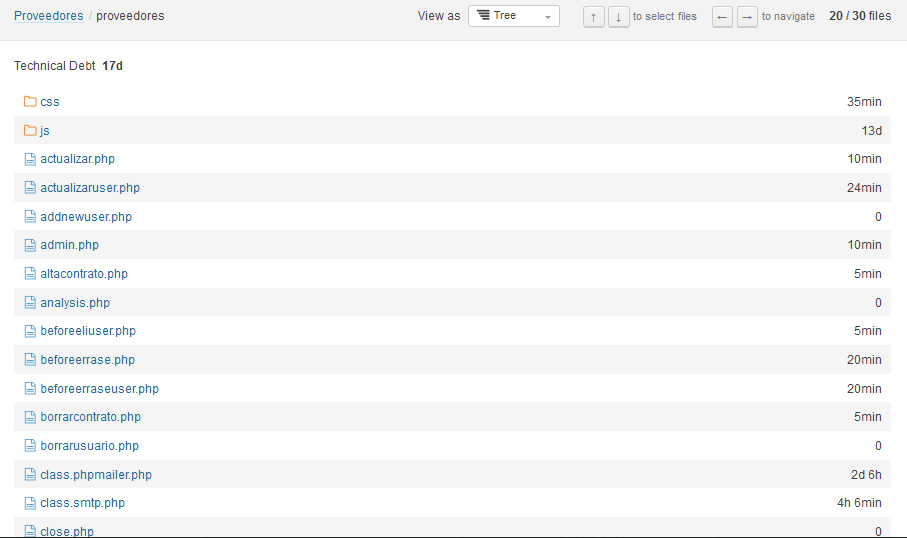


Muchos code smell corresponden a archivos de terceros (materialize.css, materialize.js, jquery.js).

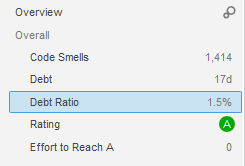


Se encontraron code smells que hacen referencia a código comentado.

* **Deuda**

****

El proyecto tiene una deuda de 17 días para aplicar correcciones.

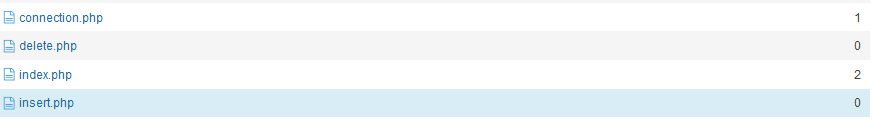


El porcentaje de deuda técnica general en el proyecto es de 1.5%, se encontró un porcentaje bajo que quiere decir que el proyecto está bien.

### Seguridad (20%)

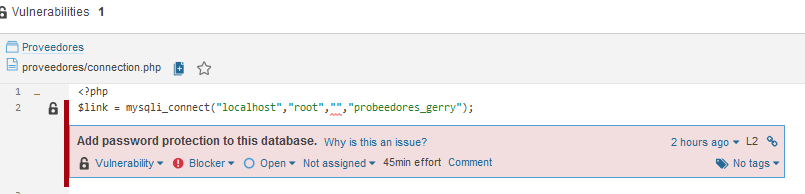
* **Vulnerabilidades**

****

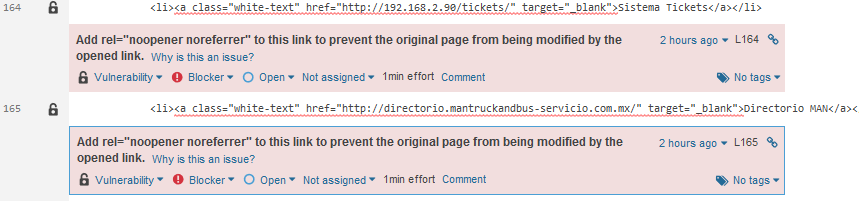


Se presentan tres vulnerabilidades:

* + connection.php: 1
  + index.php: 2



Se encontraron vulnerabilidades en la conexión a la base de datos ya que no cuenta con contraseña, siendo un problema de seguridad.



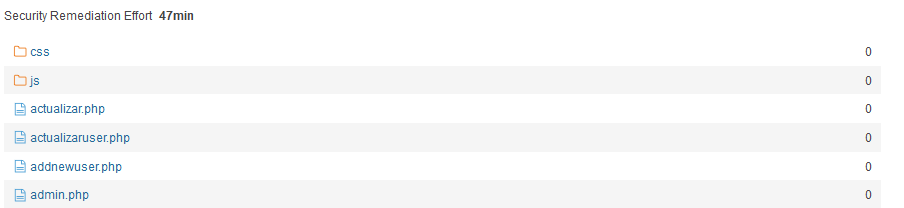
Falta del atributo *rel="noopener noreferrer"* en los enlaces que dirigen a paginas externas.

* **Clasificación**

****

La clasificación de seguridad se encuentra en **E** cuando hay al menos una vulnerabilidad.

* **Esfuerzo de reparación**

****

Se estima 47 minutos para aplicar las correcciones de seguridad.

* **Puntos de acceso de seguridad**



Con un total de 48.

* **Puntos de acceso de seguridad revisados**



0% de puntos de acceso revisados ya que no se han hecho correcciones.

## Atributos Externos.

### Usabilidad (20%)

La usabilidad hace referencia a la buena interacción del usuario con el aplicativo teniendo en cuenta lo siguiente:

* **Efectividad**

Se evidencia que las partes de la interfaz grafica cumplen con los estándares. Según el código analizado el diseño del sitio web es sencillo el cual debería mejorarse para mejor la experiencia del usuario. No se implementaron mensajes de error cuando el usuario ingresa información errónea afectando la seguridad.

**Cumplimiento: 20%**

* **Eficiencia**

La navegación del sitio web es entendible a simple vista. Tiene un diseño sencillo que permite la fácil interacción y navegabilidad. Los títulos de las secciones concuerdan con el contenido y maneja una ortografía adecuada.

**Cumplimiento: 80%**

* **Satisfacción**

A pesar de que tiene una navegabilidad sencilla y fácil de entender es recomendable mejorar el diseño del sitio. Si recomendaría el sitio a más personas para que lo visiten ya que les puede interesar sus funcionalidades.

**Cumplimiento: 70%**

Para un total de **57%** en la característica de usabilidad.

### Fiabilidad (30%)

* El sitio web no cuenta con tolerancia de fallas de seguridad. No encripta las contraseñas de los usuarios, la base de datos no tiene contraseña, los formularos no tienen validaciones.
* No tiene paginas en las que se muestren los mensajes correspondientes a los errores mas comunes de HTTP como lo son 404 y 500.
* No cuenta con protección contra los ataques comunes de robo de información. (CRSF, SQL injection)
* Maneja roles para el acceso a la información. Sin embargo, con los problemas de seguridad es posible que se puedan cambiar dando permisos a usuarios mal intencionados.

**Cumplimiento: 5%**

### Mantenibilidad (50%)

* No se cuenta con una documentación adecuada del código lo que dificulta que el software sea mantenido por otra persona ajena al creador, ya que no se entenderá el código con rapidez.
* No cuenta con un diseño de la base de datos. Se tiene que analizar todo el proyecto para identificar las tablas.
* El código es confuso ya que se mezcla mucho código PHP con HTML lo que dificulta la legibilidad y mantenibilidad.

**Cumplimiento: 30%**

# Recomendaciones y conclusiones

## Seguridad

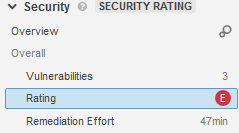
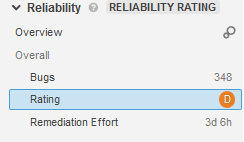
* Se deben validar los formularios.
* Encriptar las contraseñas de los usuarios.
* Colocar contraseñas a la base de datos.
* Implementar protección para los ataques comunes de robo de información.
* Mejorar los roles de los usuarios para el acceso a la información.

Se recomienda contratar un ingeniero frontend y backend por 3 meses, con gastos de 15.600.000.

## Confiabilidad

Debido a los problemas de seguridad el sitio web no es confiable. No brinda una protección adecuada de la información de los usuarios y la información recolectada que hace referencia a sus funcionalidades.

Se recomienda implementar las correcciones de seguridad y aplicar las buenas técnicas de programación para que sea más fácil el mantenimiento y legibilidad del código.



Según SonarQube la clasificación de fiabilidad y seguridad es **D** y **E** respectivamente lo que indica que es necesario tomar medidas en el proyecto para mejorar su confiabilidad debido a que se encuentran errores críticos.

Se recomienda contratar un ingeniero frontend y backend por 3 meses, con gastos de 15.600.000.

## Usabilidad

* Se recomienda usar un Framework web para facilitar la mantenibilidad del código, el desarrollo de nuevas funcionalidades.
* Mejorar el diseño del sitio web facilitando la interacción y experiencia del usuario.
* Crear Ayudas para el usuario en próximas versiones.

Se recomienda contratar un ingeniero frontend y backend por 3 meses, con gastos de 15.600.000.

# Bibliografía

Márquez, R. (01 de 02 de 2019). Evalúa la calidad de tu código con SonarQube. Obtenido de Paradigma:

<https://www.paradigmadigital.com/dev/evalua-la-calidad-de-tu-codigo-con-sonarqube/>

excentia. (18 de abril de 2018). excentia. Obtenido de excentia: <https://www.excentia.es/iso-25000>

# Firma del perito.

|  |  |
| --- | --- |
| **Responsable de la evaluación** | **Firma** |
| Nombre: Santiago Andrés Rodríguez Garcés  GitHub: <https://github.com/ssaannttyy20/quality-software> |  |