





**Contenido del informe**

[1. Propósito de la evaluación 2](#_heading=h.gjdgxs)

[2. Producto a evaluar 2](#_heading=h.30j0zll)

[3. Especificación de la(s) herramientas usadas para medición 2](#_heading=h.1fob9te)

[4. Especificación de la Norma de calidad 2](#_heading=h.3znysh7)

[5. Requerimientos específicos de calidad funcionales y técnicos. 2](#_heading=h.2et92p0)

[6. Atributos de calidad. 3](#_heading=h.3dy6vkm)

[6.1. Asignación de puntajes para atributos de calidad. 3](#_heading=h.1t3h5sf)

[6.2. Atributos internos. 3](#_heading=h.4d34og8)

[6.2.1. Tamaño de sistemas y código fuente 3](#_heading=h.2s8eyo1)

[6.2.2. Complejidad del software 3](#_heading=h.17dp8vu)

[6.2.3. Deuda técnica 3](#_heading=h.3rdcrjn)

[6.2.4. Seguridad 3](#_heading=h.26in1rg)

[6.3. Atributos Externos. 3](#_heading=h.lnxbz9)

[6.3.1. Usabilidad 3](#_heading=h.35nkun2)

[6.3.2. Fiabilidad 4](#_heading=h.1ksv4uv)

[6.3.3. Mantenibilidad 4](#_heading=h.44sinio)

[7. Recomendaciones y conclusiones 4](#_heading=h.2jxsxqh)

[7.1. Seguridad 4](#_heading=h.z337ya)

[7.2. Confiabilidad 4](#_heading=h.3j2qqm3)

[7.3. Usabilidad 4](#_heading=h.1y810tw)

[8. Bibliografía 4](#_heading=h.4i7ojhp)

[9. Firmas de evaluadores. 4](#_heading=h.2xcytpi)

1. **Propósito de la evaluación**

Comprobar el estado del código base del proyecto de facturación y su debido funcionamiento. Encontrar posibles errores o fallos que este pueda llegar a tener , esto para tener un control de errores y para que sean tomados en cuenta para futuras versiones. Corroborar que el proyecto cumpla con los estándares de calidad establecidos en las normativas ISO 25000, en dado caso de que no se cumplan estos estándares se procederá a informar de ello para que los desarrolladores lo tengan en cuenta para futuras versiones.

Examinar y verificar que los requerimientos funcionales y técnicos específicos , llevado a cabo por expertos en el ámbito de calidad de software.

1. **Producto a evaluar**

El software es de tipo empresarial:

Versión: 1.0

* Tiene como propósito la facturación e inventario de productos
* Guardar las ventas realizadas con los datos tanto del producto como del cliente.
* Posee un historial de compras por cliente.
* Cuenta con una base de datos en la cual se guardan todos los registros mencionados en los puntos anteriores.
* Tiene un apartado visual para facilitar el manejo del proyecto.

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

Sonarqube: SonarQube es un programa de código libre , que ayuda a evaluar el código fuente de proyectos informáticos , este programa usa algunos de las herramientas más útiles diseñadas para el análisis estático de código. Alguna de estas herramientas las cuales son utilizadas por SonarQube son PMD , FindBugs, Checkstyle.

La version utilizada de SonarQube es la ”8.3.1.34397”, está corre sobre el puerto 9000; el tamaño de esta imagen”SonarQube” es de 1.7 gb.

Sonar Scanner: Es un plugin de código abierto que se puede utilizar con SonarQube por lo que lo hace un complemento que permite una fácil interacción con el mismo y este es el escáner predefinido cuando no hay un escáner específico para su sistema de compilación.

la version utilizada de SonarScanner es la 4.3.0.2102 y está corre sobre el sistema operativo windows.

Sonar Scanner fue descargado de la página de SonarQube Docs

Eclipse: Es una herramienta de código abierto diseñada para programación , esta fue utilizada para la ejecución del proyecto a evaluar , poder entender y examinar el código fuente de el proyecto.

Docker: Es un proyecto que nos ayuda a virtualizar y automatizar el despliegue de cualquier tipo de aplicaciones por medio de imágenes y de contenedores. Este fue utilizado para virtualizar SonarQube y poder realizar el análisis.

1. **Especificación de la Norma de calidad**

La ISO 25000 son las normas o estándares de calidad que debe tener un software, es básicamente un conjunto de buenas prácticas al momento de desarrollar software como tener una buena documentación del código. Con el cumplimiento de estas normativas se espera que el producto tenga una mayor vigencia.



La norma ISO 25000 tiene varios estándares para el desarrollo de software como lo son los referentes al modelo del software, los requisitos y recomendaciones para la implementación, una definición de módulos y la documentación que estos deben tener.

1. **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) | Baja |
| 5 | Implementación de Ley 1581 del 2012 – Protección de datos (HABEAS DATA) | Alta |
| 6 | Permita generar reportes en EXCEL Y PDF. | Media |
| 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 |

Para cada uno de los anteriores ítems deben analizar con qué porcentaje de cumplimiento está el software y en caso que no sea del 100%, dar recomendaciones para su implementación. Determinando cuál sería el costo económico y en tiempo para su implementación.

|  |  |  |
| --- | --- | --- |
| 1 | Que el software pueda funcionar en sistemas operativos Android, MacOS, Windows XP, Windows 7 y Windows 10 (en 32 y 64 bits) | 100% |
| 2 | Que permita trabajar en forma rápida e intuitiva (cuente con ayudas visuales y auditivas interactivas en el software). | 80 |
| 3 | Que tenga soporte multi idiomas, especialmente inglés y español | 30% |
| 4 | Que permita adecuar su estilo de visualización para adecuarse a personas con limitaciones visuales (Ley 1680 de 20 de noviembre de 2013) | 50 |
| 5 | Implementación de Ley 1581 del 2012 – Protección de datos (HABEAS DATA) | 30% |
| 6 | Permita generar reportes en EXCEL Y PDF. | 50 |
| 7 | Funcionalidad/modulo para reportar errores técnicos o funcionales desde el software. | 0 |
| 8 | Permitir acceso a 100 usuarios simultáneos | ------ |
| 9 | Tolerancia a fallos (caída de red, apagones eléctricos frecuentes). | ------ |
| 10 | Integración con office | 0 |
| 11 | Cumplimiento del 80% con el estándar OWASP, priorizando en las vulnerabilidades de robo de información, XSS, SQL injection y ransomware. | 80% |
| 12 | Capacidad de respaldo y recuperación de información desde el software. | 50 |
| 13 | El software debe demandar mínimos recursos de hardware (cpu Intel celeron, 2 gigas de Ram) | 100 |

1. **Atributos de calidad.**

Los atributos de calidad que se utilizaran para la evaluación del Software Facturación, 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 explícitas e implícitas. |
| ATRIBUTOS EN USO | Características del software que determinan los requerimientos de los usuarios finales de manera que satisfagan sus necesidades. |

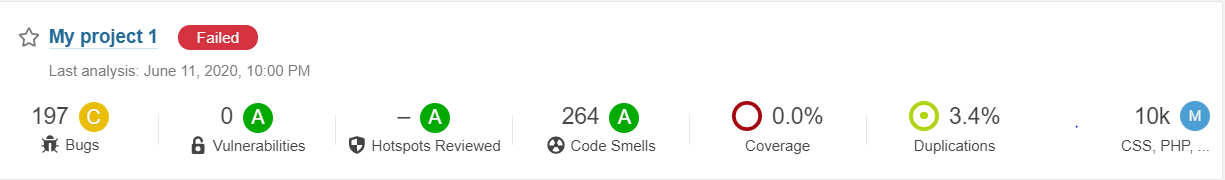
* 1. **Asignación de puntajes para atributos de calidad.**

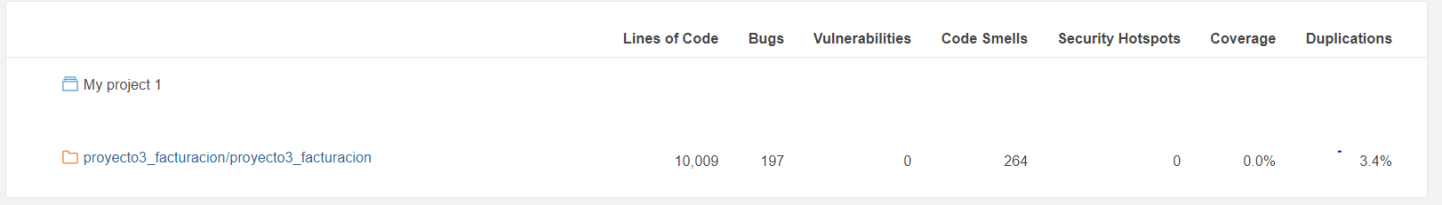
Los puntajes establecidos a los atributos de calidad seleccionados de acuerdo a las necesidades, se muestran en la siguiente tabla:

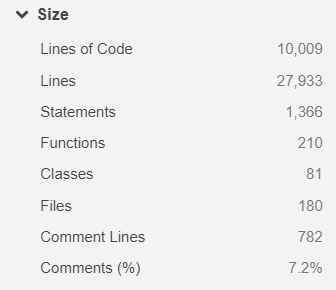
*Tabla 2, Asignación de pesos sobre la medición de atributos.*

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

* 1. **Atributos internos.**
     1. **Tamaño de sistemas y código fuente (20%)**

****

****

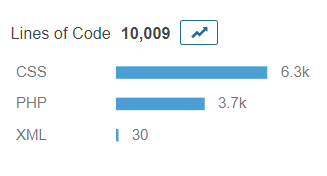
****

Líneas de código: 10009

Duplicidad:3.4%

Código smell: 264

Bugs: 197

****

**En este proyecto se utilizaron los lenguajes css, php y xml**

* + 1. **Complejidad del software(20%)**
* **Complejidad ciclomática**

****

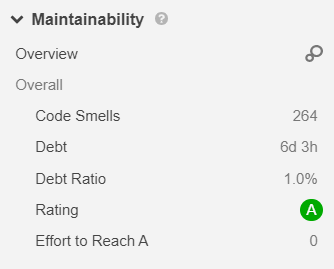
Este proyecto tiene una complejidad ciclomática de 297, lo que indica que pueden realizarse pruebas sin muchos inconvenientes.

* **Complejidad cognitiva**

****

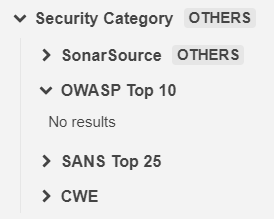
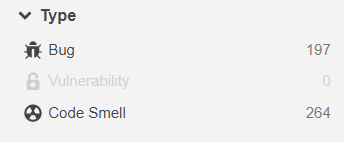
La complejidad cognitiva de este proyecto es de 173, no tiene gran complejidad para el entendimiento del proyecto.

* + 1. **Deuda técnica (20%)**

****

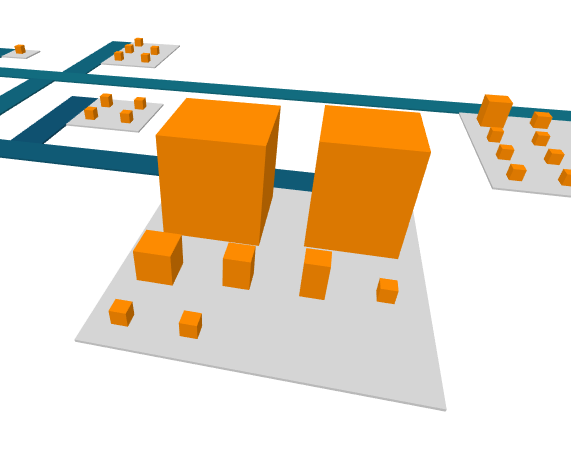
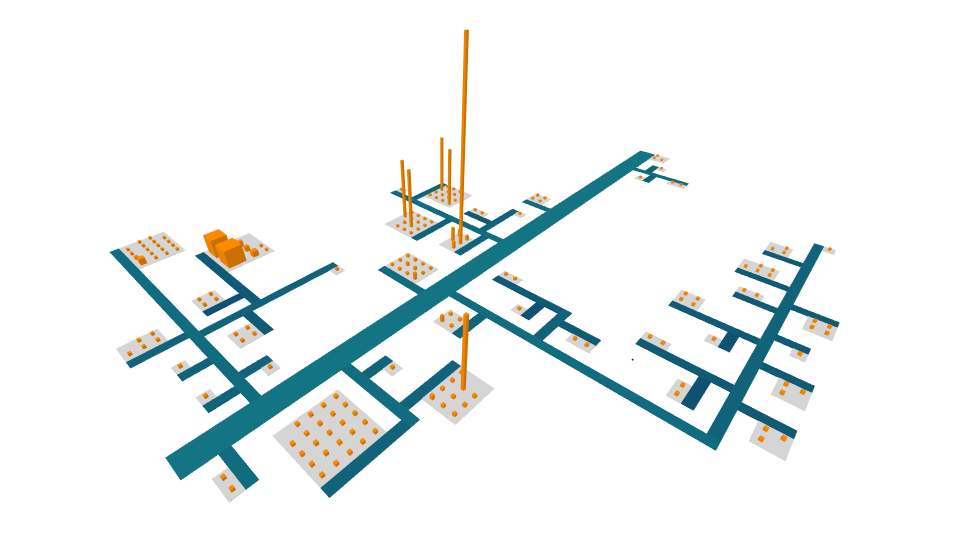
La deuda técnica es de 6 días y 3 horas, esto indica que el proyecto tiene varios errores con el código y corregirlos conlleva un gasto de tiempo un poco extenso, junto con recursos de la compañía de personal y monetarios.

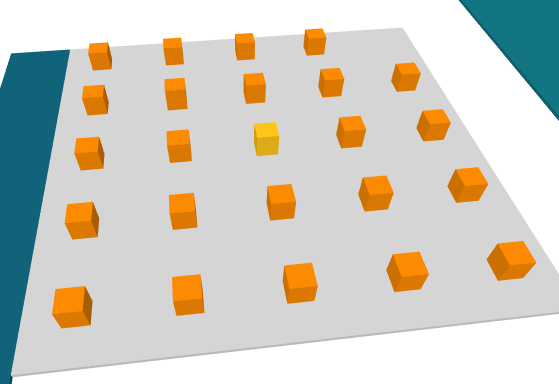
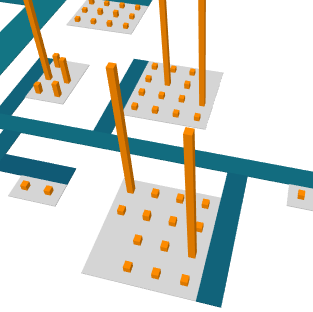
* + 1. **Seguridad(20%)**

****

Al no tener ninguna vulnerabilidad, este proyecto es bastante seguro.

* + 1. **3d code metrics (20%)**

****

****

* 1. **Atributos Externos.**
     1. **Usabilidad (20%)**

A pesar de no tener una gran cantidad de errores y que la mayoría de estos errores son por el uso erróneo de etiquetas el proyecto tiene un problema al momento de ejecutarse, y es la base de datos la cual no se encuentra adjunta al proyecto. Por lo cual solo se tiene acceso a la pantalla login de la aplicación.

* + 1. **Fiabilidad (30%)**

Con base en el código, la cantidad de errores y la gravedad de los mismos se puede decir que el software tiene una tolerancia media en cuestión de errores o fallos que pueda llegar a tener, por su deuda técnica, puede ser un gasto mayor el intentar solucionar estos errores. Aunque en su parte de prevención en la seguridad de la aplicación es bastante alta.

* + 1. **Mantenibilidad (50%)**

Al tener una deuda técnica de 6 días puede ser un problema el intentar arreglar inconvenientes que el software tenga.

1. **Recomendaciones y conclusiones**

Como conclusión general en el proyecto no se encuentran las herramientas o complementos necesarios para poder ser utilizado, en siguientes versiones la base de datos debería estar en el proyecto.

* 1. **Seguridad**

En la parte de seguridad no hay mayor observación, ya que no se encontró vulnerabilidad alguna.

* 1. **Confiabilidad**

En su versión actual no presenta vulnerabilidades como lo muestra el reporte de sonarqube.

* 1. **Usabilidad**

Al no tener la base de datos no se puede realizar una evaluación del software completa.

1. **Bibliografía**

íneas de Código ----- <https://www.sonarqube.org/>

Normativa Iso <https://iso25000.com/index.php/normas-iso-25000>

Líneas de Código-Lenguajes ----- https://www.sonarqube.org/

Líneas de Código-Líneas Comentadas ----- https://www.sonarqube.org/

Líneas Duplicadas ----- https://www.sonarqube.org/

Complejidad ciclomática ----- https://www.sonarqube.org/

Complejidad Cognitiva ----- https://www.sonarqube.org/

Código Smell ----- https://www.sonarqube.org/

Deuda Técnica ----- https://www.sonarqube.org/

Vulnerabilidades ----- https://www.sonarqube.org/

owasp ----- https://www.sonarqube.org/

3d code ----- https://www.sonarqube.org/

Zonas en conflicto ----- https://www.sonarqube.org/

Patrones ----- https://www.sonarqube.org/

1. **Firma del perito.**

|  |  |
| --- | --- |
| **Responsable de la evaluación** | **Firma** |
| Nombre: Daniel Alejandro Ruiz Sanchez  Empresa: STK Studios  Cargo: Gerente - analista  GitHub: danielruizS | Daniel Alejandro Ruiz Sanchez |