Docente: Lenis Wong

**Sistema web para la gestión de perfiles de proyectos Inmobiliarios**

PLAN SQA

Curso de Calidad de Software

**Sistema web para la gestión de perfiles de proyectos Inmobiliarios**

**Plan de SQA**

**Versión [4.0]**

**Historia de revisiones**

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión | Descripción | Autor |
| 18/05/2021 | 1.0 | Creación del Plan | Lenis Wong |
| 02/06/2021 | 2.0 | Actualización del Documento | Daniel López |
| 27/07/2021 | 4.0 | Actualización del documento | Daniel López |
|  |  |  |  |

**Lista de contenido**

Contenido

[1. Propósito 5](#_Toc78422907)

[1.1. Objetivo estratégico 5](#_Toc78422908)

[1.2. Objetivos del proyecto 5](#_Toc78422909)

[2. Gestión 6](#_Toc78422910)

[2.1. Organización 6](#_Toc78422911)

[2.2. Responsables 6](#_Toc78422912)

[2.3. Actividades del aseguramiento de la calidad 7](#_Toc78422913)

[3. Documentación 7](#_Toc78422914)

[3.1. Documentación mínima requerida 7](#_Toc78422915)

[3.1.1. Plan de Gestión de configuración 8](#_Toc78422916)

[4. Estándares de Documentación 8](#_Toc78422917)

[5. Estándares, prácticas, convenciones y métricas 9](#_Toc78422918)

[5.1. Estándar de Calidad del producto 9](#_Toc78422919)

[5.1.1. Modelo de Calidad del Producto - General ISO 9126 9](#_Toc78422920)

[5.1.2. Modelo de Calidad del Producto – detallado 12](#_Toc78422921)

[5.1.3. Estándar para Métricas de calidad 17](#_Toc78422922)

[5.1.3.1. Lista de Métricas priorizadas 17](#_Toc78422923)

[5.1.3.2. Definición de Métricas de Calidad 18](#_Toc78422924)

[5.2. Estándar de Calidad del proceso 22](#_Toc78422925)

[5.2.1. Proceso de Desarrollo de Software 22](#_Toc78422933)

[5.2.2. Análisis de requisitos de Software 23](#_Toc78422934)

[5.2.3. Proceso de prueba de calificación del Software 24](#_Toc78422935)

[5.2.4. Proceso de construcción de software. 25](#_Toc78422936)

[5.2.5. Proceso de Gestión de la configuración. 25](#_Toc78422937)

[5.2.6. Proceso de Aseguramiento de la calidad. 26](#_Toc78422938)

[6. Revisiones, Verificación y auditorias. 27](#_Toc78422939)

[6.1. Casos de prueba. 27](#_Toc78422940)

[6.1.1. Casos de prueba UC10-01 Creación de clientes. 27](#_Toc78422941)

[6.1.2. Casos de prueba UC10-02 Creación de clientes. 28](#_Toc78422942)

[6.1.3. Casos de prueba UC10-03 Validación de datos de clientes. 29](#_Toc78422943)

[6.1.4. Casos de prueba UC10-04 Eliminar clientes. 30](#_Toc78422944)

[6.1.5. Casos de prueba UC10-05 Eliminar clientes. 30](#_Toc78422945)

[6.2. Métrica para la simulación. 31](#_Toc78422946)

[6.3. Simulación. 32](#_Toc78422947)

[7. Reporte de problemas y acciones correctivas 33](#_Toc78422948)

[7.1. Procedimiento de registro de problemas. 33](#_Toc78422949)

[7.2. Procedimiento de registro de problemas. 33](#_Toc78422950)

[8. Herramientas, técnicas y metodologías 34](#_Toc78422951)

[8.1. SonarQube. 34](#_Toc78422952)

[8.2. Katalon. 34](#_Toc78422953)

Lista de Tablas y Figuras

Tablas

[Tabla 1 - Actividades del aseguramiento. 7](#_Toc78422954)

[Tabla 2 - Tabla de pesos para el cálculo de métricas 9](#_Toc78422955)

[Tabla 3 - Justificación de las caracteristicas 10](#_Toc78422956)

[Tabla 4 – Justificación de la característica de Funcionalidad 12](#_Toc78422957)

[Tabla 5 – Justificación de la característica de Fiabilidad 13](#_Toc78422958)

[Tabla 6 - Justificación de la característica Usabilidad 14](#_Toc78422959)

[Tabla 7 - Justificación de la característica de eficiencia 15](#_Toc78422960)

[Tabla 8 - Justificación de la característica facilidad de mantenimiento. 15](#_Toc78422961)

[Tabla 9 - Justificación de la característica de Portabilidad 16](#_Toc78422962)

[Tabla 10 - Lista de métricas priorizadas 17](#_Toc78422963)

[Tabla 11 - Métrica Tasa de funcionalidades Aprendidas 18](#_Toc78422964)

[Tabla 12 – Métrica Calificación en la protección de datos de registros de perfiles inmobiliarios. 19](#_Toc78422965)

[Tabla 13 - Indicador de capacidad de recuperación 19](#_Toc78422966)

[Tabla 14 - Métrica Tasa de recuperación de errores. 20](#_Toc78422967)

[Tabla 15 - Tiempo de operación de registro. 20](#_Toc78422968)

[Tabla 16 - Tasa de funcionalidades correctamente implementadas. 21](#_Toc78422969)

[Tabla 17 - Formato de caso de prueba 27](#_Toc78422970)

[Tabla 18 - Caso de prueba UC10-01 27](#_Toc78422971)

[Tabla 19 - Casos de prueba UC10-02 Creación de clientes. 28](#_Toc78422972)

[Tabla 20 - Caso de prueba UC10-03 Validación de datos de clientes 29](#_Toc78422973)

[Tabla 21 - Caso de prueba UC10-04 Eliminar cliente 30](#_Toc78422974)

[Tabla 22 - Caso de prueba UC10-05 Eliminar cliente 30](#_Toc78422975)

[Tabla 23 - Métrica para la simulación 31](#_Toc78422976)

[Tabla 24 - Resumen de la ejecucion de los casos de prueba 32](#_Toc78422977)

[Tabla 25 - Aplicacion de la métrica 32](#_Toc78422978)

[Tabla 26 - Formato de registro de incidentes 33](#_Toc78422979)

Figuras

[Figura 1- Organización del proyecto 6](#_Toc78422980)

[Figura 2 – Modelo de la calidad 10](file:///C:\Users\danie\OneDrive\Escritorio\Maestrias\MII\CICLO%2003\302%20Calidad%20de%20Software\PLAN_SQA-empresa%20inmobiliariaV4.docx#_Toc78422981)

[Figura 3 - Proceso General -Ciclo de Vida del desarrollo de sw. 22](#_Toc78422982)

[Figura 4 - Proceso de análisis de requisitos 23](file:///C:\Users\danie\OneDrive\Escritorio\Maestrias\MII\CICLO%2003\302%20Calidad%20de%20Software\PLAN_SQA-empresa%20inmobiliariaV4.docx#_Toc78422983)

[Figura 5 - Proceso de prueba de calificación del software 24](file:///C:\Users\danie\OneDrive\Escritorio\Maestrias\MII\CICLO%2003\302%20Calidad%20de%20Software\PLAN_SQA-empresa%20inmobiliariaV4.docx#_Toc78422984)

[Figura 6 - Proceso de Evaluación de artefactos 24](file:///C:\Users\danie\OneDrive\Escritorio\Maestrias\MII\CICLO%2003\302%20Calidad%20de%20Software\PLAN_SQA-empresa%20inmobiliariaV4.docx#_Toc78422985)

[Figura 7 - Proceso de construcción de software 25](file:///C:\Users\danie\OneDrive\Escritorio\Maestrias\MII\CICLO%2003\302%20Calidad%20de%20Software\PLAN_SQA-empresa%20inmobiliariaV4.docx#_Toc78422986)

[Figura 8 - proceso de gestión de la configuración 25](file:///C:\Users\danie\OneDrive\Escritorio\Maestrias\MII\CICLO%2003\302%20Calidad%20de%20Software\PLAN_SQA-empresa%20inmobiliariaV4.docx#_Toc78422987)

[Figura 9 - Proceso de aseguramiento de la calidad 26](file:///C:\Users\danie\OneDrive\Escritorio\Maestrias\MII\CICLO%2003\302%20Calidad%20de%20Software\PLAN_SQA-empresa%20inmobiliariaV4.docx#_Toc78422988)

[Figura 10 - Sonarqube 34](file:///C:\Users\danie\OneDrive\Escritorio\Maestrias\MII\CICLO%2003\302%20Calidad%20de%20Software\PLAN_SQA-empresa%20inmobiliariaV4.docx#_Toc78422989)

# Propósito

La consultora inmobiliaria TUTERRENO SAC realiza perfiles de proyectos inmobiliarios para las más grandes empresas del sector inmobiliario. Estos perfiles permiten a las empresas constructoras establecer una idea clara de la rentabilidad que ofrece el terreno incluso previo a la adquisición.

La elaboración de estos perfiles requiere de un uso intensivo del Excel, esto inicialmente fue muy útil, pero se generaron varios inconvenientes debido a no podían controlar los cambios y evitar que los clientes tengan acceso a las reglas de negocio que son propiedad intelectual de la consultora, adicionalmente durante evolutivo se han creado nuevas reglas que son difíciles de actualizar la diversidad de Excel creados en los diferentes proyectos, adicionalmente se requiere que los consultores puedan aumentar su capacidad de elaboración de perfiles. Por ello se solicita a nuestra consultora de software el diseño y construcción de una herramienta que permita la elaboración de estos perfiles de forma tal de satisfacer las necesidades de la consultora inmobiliaria TUTERRENO SAC.

## Objetivo estratégico

Optimizar los procesos de elaboración de perfiles y reducir el tiempo de consultoría en un 40%.

## Objetivos del proyecto

* Crear una base de datos que permita gestionar las consultorías a los clientes y reducir el tiempo en la ubicación de información en un 20%.
* Crear una plataforma de acceso a los clientes que le permita ver el estado de los perfiles en tiempo real, actualmente demora de 2 a 4 horas (10%).
* Mejorar la curva de aprendizaje de los nuevos consultores en un 10%.

El objetivo de este plan es garantizar y verificar que se hayan cumplido los objetivos estratégicos y los objetivos del proyecto.

El aseguramiento se realizará durante la implementación de los Sprint.

# Gestión

## Organización

El equipo scrum al ser un equipo autogestionado no tiene un líder específico cada miembro apoya al proyecto en función a sus habilidades, pero cualquier miembro del equipo puede asumir un rol determinado de necesitar.

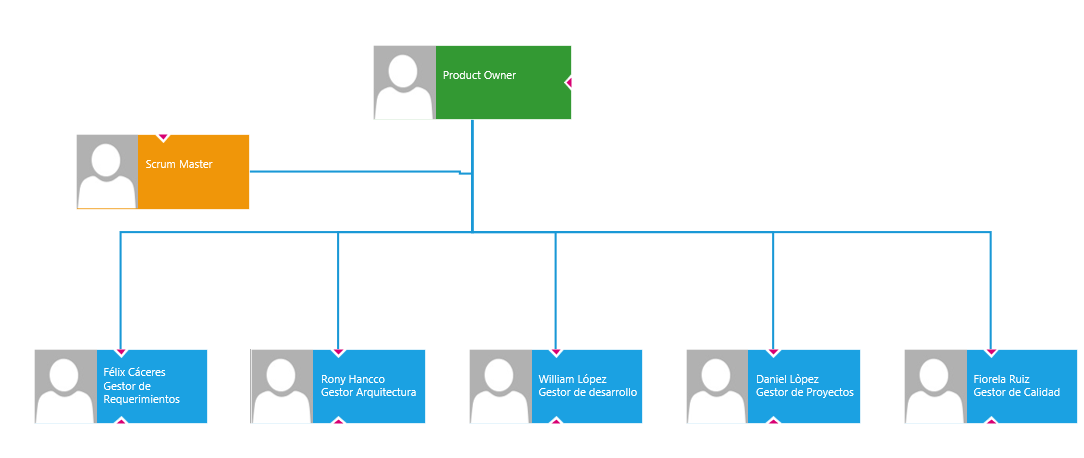


Figura 1- Organización del proyecto

En la figura 1 podemos ver la organización del equipo de proyecto sobre los que recae la responsabilidad de validar la calidad de los entregables en cada una de las etapas del ciclo de vida del desarrollo de software.

## Responsables

|  |
| --- |
| **Líder QA (1)**   * Planificar la Calidad. * Revisar las Entregas. * Evaluar la Calidad de los Productos. * Realizar el Informe Final de Calidad. |
| **Analista de QA (1)**   * Define y Diseña los casos de pruebas. |
| **Tester (2)**   * Llevar a cabo pruebas de estrés * pruebas de rendimiento * pruebas funcionales * pruebas de escalabilidad. |
| Especialista de automatización de pruebas (1)   * Desarrolla los scripts para la ejecución de las pruebas automatizadas. |

## Actividades del aseguramiento de la calidad

La tabla 1 muestra las actividades que se realizarán por cada sprint.

Tabla 1 - Actividades del aseguramiento.

|  |  |  |
| --- | --- | --- |
| **Actividad** | **Semanas** | **Rol** |
| Elaborar el plan SQA (actualización) | 0.5 | Líder QA |
| Diseñar los casos de prueba | 1.5 | Analista QA |
| Elaborar los scripts de automatización | 2.0 | Automatizador |
| Preparar los datos de prueba | 0.5 | Tester 1 y 2 |
| Ejecutar los casos de prueba | 2.0 | Tester 1 y 2 |
| Reportar los bugs (fallas) | 0.2 | Tester 1 y 2 |
| Elaborar el informe de pruebas | 0.2 | Analista QA |
| Evaluar los criterios de salida | 0.1 | Analista QA |
| Evaluar la calidad de los productos | 0.5 | Líder QA |
| Elaborar el informe final de calidad | 0.2 | Líder QA |

EL proyecto contara con 2 iteraciones o Sprint.

# Documentación

## Documentación mínima requerida

Los documentos mininos requeridos en este plan permitirán establecer el enfoque de pruebas, alinear los procesos a los entregables del desarrollo y analizar los entregables para estimar el esfuerzo de pruebas.

* Acta de constitución del proyecto (Project Chapter).
* Cronograma del proyecto.
* Historias de usuarios.
* Producto Backlog
* Sprint Backlog (por sprint).

## Plan de Gestión de configuración

El Plan de gestión de configuración debe contener métodos para identificar componentes de software, control e implementación de cambios, y registro y reporte del estado de los cambios implementados.

# Estándares de Documentación

* Estándar de Programación php
* Estándar para la elaboración de Estilos CSS
* Estándar de Nomenclatura de objetos Base de Datos Mysql.
* Estándar de codificación Java Script.
* Estándar de diccionario de datos
* Estándar de diseño de pantallas del Negocio.

# Estándares, prácticas, convenciones y métricas

## Estándar de Calidad del producto

Este plan se desarrolla basado en el estándar ISO 9126, tomando las características y sub caracteristicas definidas en este modelo, para ello se establecen los pesos y se definen métricas.

## Modelo de Calidad del Producto - General ISO 9126

Para estimar los pesos y las prioridades de las características del modelo de calidad propuesto hemos establecidos la siguiente clasificación que la podemos ver en la Tabla 2 - Tabla de pesos para el cálculo de métricas:

Tabla 2 - Tabla de pesos para el cálculo de métricas

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Clasificación** | **Valor Min** | **Valor Max** |
| N | NO APLICA | 0 | 0 |
| B | BAJO | 1 | 3 |
| M | MEDIO | 4 | 6 |
| A | ALTO | 7 | 9 |

**No aplica:**

Se usa cuando la característica o sub-caracteritica no será evaluada o no es relevante para determinar la calidad de producto.

**Bajo:**

Se usa cuando la característica o sub-caracteritica tiene es poco relevante para determinar la calidad de producto.

**Medio:**

Se usa cuando la característica o sub-caracteritica tiene es medianamente relevante para determinar la calidad de producto.

**Alto:**

Se usa cuando la característica o sub característica tiene es importante relevancia para determinar la calidad de producto.

El la Figura 2 podemos observar como el modelo de calidad destruye los pesos entre cada característica, como resultado de la importancia que cada característica tiene para el proyecto.

Figura 2 – Modelo de la calidad

En la tabla 3 justificamos cada una de las características según el modelo de calidad en función de los requerimientos del producto de software.

Tabla 3 - Justificación de las caracteristicas

|  |  |  |
| --- | --- | --- |
| **Característica** | **Peso %** | **Justificación** |
| **Funcionalidad** | **27.27%** | Por ser una aplicación de uso intenso de cálculos, debe ser lo más exacta posible, segura por trabajar con datos sensibles y altamente operable por el nivel de tecnificación de los usuarios. |
| **Fiabilidad** | **24.24%** | Es importante que la aplicación controle y evita los errores alcanzando los tiempos de recuperación ante fallas establecidos para la creación de perfiles inmobiliarios en tiempo real. |
| **Usabilidad** | **21.21%** | Debido a que el software tiene un alto grado de especialización es importante que la aplicación sea fácil de usar, aprender y operar. |
| **Eficiencia** | **12.12%** | La aplicación requiere un desarrollo en menor tiempo lo cual implica que se use un lenguaje de programación y diseño que incremente la productividad en desmedro de la eficiencia por no ser altamente concurrente. |
| **Mantenibilidad** | **09.09%** | La aplicación cuenta con mecanismos que permiten la estabilidad de sus procesos críticos, además de que la mayoría de sus componentes están a la medida para cumplir con las funcionalidades requeridas por el cliente. |
| **Portabilidad** | **06.07%** | La aplicación tiene la capacidad de poder ser utilizada para otros clientes ya que puede reemplazar fácilmente a cualquier otro sistema que se encargue de generar perfiles inmobiliarios. |
|  | **100.00%** |  |

## Modelo de Calidad del Producto – detallado

**A continuación, pasaremos a describir cada una de las características y justificaremos**

1. **Funcionalidad:**

Esta sub característica mide la capacidad que tiene el producto de software para cumplir con las funcionalidades requeridas. En la tabla 4, Justificamos los pesos de cada una de las sub características. Vemos que la sub característica más relevantes es la exactitud con un 27.07%.

Tabla 4 – Justificación de la característica de Funcionalidad

|  |  |  |
| --- | --- | --- |
| **Sub Característica** | **Peso %** | **Justificación** |
| **Adecuación** | **18.18%** | Es importante que la funcionalidad sea correctamente implementada debido a la sensibilidad de los cálculos. |
| **Exactitud** | **27.27%** | La aplicación realiza múltiples operaciones que requieren exactitud y precisión a 4 decimales. |
| **Inter-operatividad** | **09.09%** | La aplicación interactuara con pocos componentes, básicamente el servidor web y la base de datos, por ello se considera de bajo impacto a esta sub característica. |
| **Seguridad** | **21.21%** | La aplicación maneja datos sensibles de los clientes de la inmobiliaria, por ello debe contar con métricas de seguridad. |
| **Conformidad de la funcionalidad.** | **24.24%** | La aplicación debe cumplir con todas las reglas de calculo establecidas para la confección de perfiles inmobiliarios. |
|  | **100.00%** |  |

1. **Fiabilidad**

La fiabilidad mide la capacidad de producto de software de asegurar un nivel de funcionamiento adecuado.

En la tabla 5 podemos observar la justificación por cada una de las sub características, siendo la más importante la tolerancia a fallos con un 31.03% de importancia.

Tabla 5 – Justificación de la característica de Fiabilidad

|  |  |  |
| --- | --- | --- |
| **Sub Característica** | **Peso %** | **Justificación** |
| **Capacidad de recuperación** | **27.59%** | Se necesita que la aplicación pueda recuperarse ante errores generados sin perjudicar su funcionamiento. |
| **Madurez** | **20.69%** | Se busca que la aplicación controle y evite los errores generados en la creación de los perfiles. |
| **Tolerancia a Fallos** | **31.03%** | La aplicación tiene que seguir operando en situaciones en las que se presentan errores de los servidores, con un mínimo tiempo de recuperación. |
| **Cumplimiento de la fiabilidad** | **20.69%** | La aplicación cumple con las normas de fiabilidad estipuladas por la empresa. |
|  | **100.00%** |  |

1. **Usabilidad**

La característica de usabilidad mide la capacidad el producto de software de ser entendido, aprendido y usado de forma fácil y atractiva.

En la tabla 6 podemos ver que la característica mas levante esta empatada entre el entendimiento y la operatividad, por ello este producto debe ser igualmente entendible y operable.

Tabla 6 - Justificación de la característica Usabilidad

|  |  |  |
| --- | --- | --- |
| **Sub Característica** | **Peso %** | **Justificación** |
| **Entendimiento** | **23.68%** | La aplicación está correctamente separada por funcionalidades en diferentes módulos para mejorar el entendimiento de las tareas en que puede ser usada. |
| **Operatividad** | **23.68%** | La aplicación puede ser controlada y operada con facilidad debido a su interfaz amigable y cuenta con iconos y descripciones en sus diferentes pantallas que dan un indicio a lo que cada opción hace. |
| **Aprendizaje** | **21.05%** | La aplicación facilita el aprendizaje y cuenta con un instructivo que ayuda a su manejo. |
| **Conformidad de la usabilidad** | **15.79%** | La aplicación se adhiere al estándar de diseño de pantallas del Negocio ya establecida. |
| **Atracción** | **15.79%** | La aplicación es medianamente atractiva a la vista del usuario, pero tiene una interfaz amigable. |
|  | **100.00%** |  |

1. **Eficiencia**

La característica de eficiencia mide la forma en que un producto de software se desempeña de forma adecuada.

En la Tabla 7 vemos que es importante la sub característica de utilización de recursos lo que indica que es importante que el producto de software utilice adecuadamente los recursos que utiliza.

Tabla 7 - Justificación de la característica de eficiencia

|  |  |  |
| --- | --- | --- |
| **Sub Característica** | **Peso %** | **Justificación** |
| **Comportamiento de tiempos** | **18.75%** | La aplicación cubre las expectativas de los usuarios relacionado a los tiempos de respuesta y procesamiento que requieren. |
| **Utilización de recursos** | **50.00%** | La aplicación tendrá un uso óptimo de recursos dado que se implementará en los servidores actuales en la nube que dispone la empresa. |
| **Conformidad de eficiencia** | **31.25%** | La aplicación se suscribe a las normas establecidas de eficiencia de la empresa relacionada al tiempo para la creación de perfiles inmobiliarios. |
|  | **100%** |  |

1. **Facilidad de Mantenimiento**

Esta característica mide la capacidad que tiene el producto de software de ser medido. En la Tabla 8 podemos observar que la principal sub característica es estabilidad con un 29.63% indicando que es importante que es importante que sistema muestre estabilidad.

Tabla 8 - Justificación de la característica facilidad de mantenimiento.

|  |  |  |
| --- | --- | --- |
| **Sub Característica** | **Peso %** | **Justificación** |
| **Capacidad de ser analizado** | **18.52%** | La aplicación solo cuenta con mecanismos de cobertura de código, por lo que es el único medio por el cual puede ser analizado. |
| **Capacidad de ser cambiado** | **11.11%** | La aplicación nació como una necesidad de un cliente en particular por lo que sus componentes son cerrados y cambiarlo puede inestabilizar alguna funcionalidad. |
| **Estabilidad** | **29.63%** | Para que los procesos de cálculos no generen potenciales problemas, se tiene en consideración múltiples pruebas unitarias en el código para evitar escenarios de inconsistencia. |
| **Capacidad de ser probado** | **25.93%** | La aplicación cuenta con artefactos que puede ser usado para el uso de las pruebas. |
| **Cumplimiento de la mantenibilidad** | **14.81%** | La aplicación cuenta con una serie de instrucciones para sus pruebas funcionales que serán usadas por el personal de pruebas tal como lo define el manual de mantenibilidad de la empresa. |
|  | **100.00%** |  |

1. **Portabilidad**

Esta característica mide la capacidad que tiene el producto de software para ser trasladado a otros entornos. En la Tabla 9 vemos que la capacidad de reemplazar tiene un 40% de importancia, esta característica es vital ya que este producto esta reemplazando al uso de Excel.

Tabla 9 - Justificación de la característica de Portabilidad

|  |  |  |
| --- | --- | --- |
| **Sub Característica** | **Peso %** | **Justificación** |
| **Adaptabilidad** | **20.00%** | La aplicación solo será visualizada desde cualquier PC por medio de navegadores web (Edge, Google Chrome y Firefox). |
| **Instalabilidad** | **10.00%** | Al ser una aplicación web, las consideraciones de instalación o desinstalación son mínimas. |
| **Coexistencia** | **20.00%** | La finalidad de la aplicación no depende de otros sistemas para su operatividad. |
| **Capacidad de Reemplazar** | **40.00%** | La aplicación reemplaza el Excel que actualmente se utiliza para la creación de perfiles inmobiliarios, por ello es importante su capacidad de reemplazar. |
| **Cumplimiento de Portabilidad** | **10.00%** | La aplicación debe poder ser instalada tanto en la nube de la empresa como en el servidor web local tal como lo define el manual de portabilidad de la empresa. |
|  | **100.00%** |  |

## Estándar para Métricas de calidad

Para la medición de las caracteristicas y sub caracteristicas se han definido una relación de 16 métricas que representan el 80.10% del total de las características.

## Lista de Métricas priorizadas

La Tabla 10 nos muestra la lista de métricas prioridad, para facilitar el análisis podemos observar el peso que tiene cada métrica en el total, también vemos la columna prioridad que tiene relación donde el mayor peso tiene mas prioridad siendo el 1 la mayor prioridad, finalmente la columna peso acumulado nos permite ver como las primeras 16 métricas suman un peso mayor al 80%.

Tabla 10 - Lista de métricas priorizadas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Código** | **Nombre** | **Peso** | **Prioridad** | **peso acumulado** |
| MFI3 | Tasa de recuperación de errores | 0.075235 | 1 | 0.075235 |
| MFI1 | Tiempo mínimo de recuperación g cuando el sistema tenga una caída servicio de BD | 0.066876 | 2 | 0.142111 |
| MF6 | Indicador de cumplimiento de normas de empresas inmobiliarias. | 0.066116 | 3 | 0.208226 |
| ME2 | Nivel de utilización de CPU del servidor. | 0.060606 | 4 | 0.268833 |
| MF5 | Calificación en la protección de datos de registros de perfiles inmobiliarios. | 0.057851 | 5 | 0.326684 |
| MU1 | Capacidad de la aplicación de poder ser entendida. | 0.050239 | 6 | 0.376923 |
| MU2 | Tiempo de operación de registro. | 0.050239 | 7 | 0.427162 |
| MFI2 | Cantidad de errores controlados y evitados en el proceso de registro de perfiles inmobiliarios | 0.050157 | 8 | 0.477319 |
| MFI4 | Conformidad de las normas de fiabilidad estipuladas por la empresa. | 0.050157 | 9 | 0.527476 |
| MF1 | Tasa de funcionalidades correctamente implementadas. | 0.049587 | 10 | 0.577062 |
| MU3 | Tasa de funcionalidades aprendidas | 0.044657 | 11 | 0.621720 |
| MF2 | Calificación en la protección de datos de registros de perfiles inmobiliarios | 0.039378 | 12 | 0.661097 |
| ME3 | Nivel de aceptación de eficiencia estándar de respuesta. | 0.037879 | 13 | 0.698976 |
| MF3 |  | 0.035002 | 14 | 0.733979 |
| MU4 | Capacidad de la aplicación para dar conformidad de la usabilidad | 0.033493 | 15 | 0.767471 |
| MU5 | Capacidad de la aplicación para ser atractiva al usuario. | 0.033493 | 16 | 0.800964 |

## Definición de Métricas de Calidad

A continuación, detallamos las principales métricas, las siguientes tablas nos permiten ver el detalle, elemento importante para obtener la medición de las características y sub caracteristicas del producto de software.

Adicionalmente cada métrica tiene un autor que nos indica como el trabajo de determinar la calidad es un trabajo en equipo.

Tabla 11 - Métrica Tasa de funcionalidades Aprendidas

|  |  |
| --- | --- |
| **NOMBRE DE LA METRICA:** | **Tasa de funcionalidades Aprendidas** |
| **Autor:** | Daniel López |
| **CODIGO:** | MU3 |
| **CARACTERISTICA:** | USABILIDAD |
| **SUBCARACTERISTICA:** | CAPACIDAD DE SER APRENDIDO |
| **PROPOSITO:** | ¿Cuántas funciones que el usuario puede operar y controlar del módulo de gestión de perfiles? |
| **MEDICIÓN, FORMULA Y ELEMENTOS MEDIBLES:** | T: Total de funcionalidades del módulo de gestión de perfiles. Q: Cantidad de funcionalidades que el usuario control de módulo de gestión de perfiles. X: Q/T. |
| **INTERPRETACION DEL VALOR** | El módulo de gestión de perfiles tiene X de capacidad de ser usa operado. 0<=x<=1, donde 1 es el valor óptimo. |

Tabla 12 – Métrica Calificación en la protección de datos de registros de perfiles inmobiliarios.

|  |  |
| --- | --- |
| **NOMBRE DE LA METRICA:** | **Calificación en la protección de datos de registros de perfiles inmobiliarios.** |
| **Autor:** | Félix Cáceres |
| **CODIGO:** | MF5 |
| **CARACTERISTICA:** | FUNCIONALIDAD |
| **SUBCARACTERISTICA:** | SEGURIDAD |
| **PROPOSITO:** | ¿Cuán seguro se encuentran protegidos los datos registrados por el usuario? |
| **MEDICIÓN, FORMULA Y ELEMENTOS MEDIBLES:** | Checklist para determinar cumplimiento de estándar de seguridad de datos. Sumatoria de resultados en base al checklist T= total de preguntas A= Cantidad de preguntas satisfactorias X= A/T |
| **INTERPRETACION DEL VALOR** | 0 < X < 1, Donde 1 sería una calificación perfecta. |
| Entrada para la medición | Checklist |

Tabla 13 - Indicador de capacidad de recuperación

|  |  |
| --- | --- |
| **NOMBRE DE LA METRICA:** | **Indicador de capacidad de recuperación** |
| **Autor:** | Rony Hancco |
| **CODIGO:** | MFI1 |
| **CARACTERISTICA:** | FIABILIDAD |
| **SUBCARACTERISTICA:** | CAPACIDAD DE RECUPERACION |
| **PROPOSITO:** | Medir la capacidad de recuperación cuando el sistema tenga una caída servicio de BD |
| **MEDICIÓN, FORMULA Y ELEMENTOS MEDIBLES:** | RTO: Tiempo en minutos en que el sistema está fuera de servicio. C: Cantidad de eventos Escala T (1 a 5): Escala de tiempo de recuperación por evento 5: RTO <= 2 min 4: 2 min < RTO <= 3 min 3: 3 min <= RTO < 5 min 2: 5 min <= RTO <= 10 min 1: RTO > 10 min N = Suma(T/5)/C |
| **INTERPRETACION DEL VALOR** | El sistema tiene la capacidad de recuperación N frente a una caída de servicio de BD,  0<N<1 donde 1 es el valor óptimo. |

Tabla 14 - Métrica Tasa de recuperación de errores.

|  |  |
| --- | --- |
| **NOMBRE DE LA METRICA:** | **Tasa de recuperación de errores.** |
| **Autor:** | William López |
| **CODIGO:** | FI3 |
| **CARACTERISTICA:** | Fiabilidad |
| **SUBCARACTERISTICA:** | Tolerancia a Fallos |
| **PROPOSITO:** | ¿Cuántos errores que se pueden recuperar cuando el Sistema de generación perfiles inmobiliaria genera errores? |
| **MEDICIÓN, FORMULA Y ELEMENTOS MEDIBLES:** | A = Cantidad de errores recuperados T = Cantidad de errores generados X = A/T |
| **INTERPRETACION DEL VALOR** | 0 <= X <=1 El mayor valor es mejor X = 1, es el mejor de los casos, se recuperaron todos los errores generados. |

Tabla 15 - Tiempo de operación de registro.

|  |  |
| --- | --- |
| **NOMBRE DE LA METRICA:** | **Tiempo de operación de registro.** |
| **Autor:** | Daniel López |
| **CODIGO:** | MU2 |
| **CARACTERISTICA:** | USABILIDAD |
| **SUBCARACTERISTICA:** | CAPACIDAD DE SER OPERADO |
| **PROPOSITO:** | ¿Qué tiempo el usuario demora en el registro de un perfil inmobiliario? |
| **MEDICIÓN, FORMULA Y ELEMENTOS MEDIBLES:** | T: Tiempo óptimo de registro de un perfil inmobiliario. Q: Tiempo que demora un usuario en registrar un perfil inmobiliario. X: Q/T. |
| **INTERPRETACION DEL VALOR** | El registro de perfiles tiene una capacidad de ser operado de X.  0<=x<=1, donde 1 es el valor óptimo. |

Tabla 16 - Tasa de funcionalidades correctamente implementadas.

|  |  |
| --- | --- |
| **NOMBRE DE LA METRICA:** | Tasa de funcionalidades correctamente implementadas. |
| **Autor:** | Fiorela Ruiz |
| **CODIGO:** | MF1 |
| **CARACTERISTICA:** | FUNCIONALIDAD |
| **SUBCARACTERISTICA:** | ADECUACIÓN |
| **PROPOSITO:** | ¿Cuán correcta es la implementación de funciones de la aplicación de perfiles inmobiliarios? |
| **MEDICIÓN, FORMULA Y ELEMENTOS MEDIBLES:** | A: Número de funcionalidades incorrectamente implementadas. Q: Número de funcionalidades descritas en la especificación de requisitos. X: 1 - A/Q |
| **INTERPRETACION DEL VALOR** | El registro de perfiles tiene una capacidad de ser operado de X.  0<=x<=1, donde 1 es el valor óptimo. |

## Estándar de Calidad del proceso

Debido a que decidió llevar el proyecto bajo el marco de trabajo scrum se decidió adoptar el proceso de desarrollo iterativo incremental característico de este tipo de proyecto.

Sin embargo, siempre hay un proceso logico del desarrollo de software, el que debemos seguir para ya sea que estemos en ciclo iterativo o de cascada.



## Proceso de Desarrollo de Software

A continuación, en la Figura 3 definimos los procesos principales que deben considerarse en el desarrollo de software y que el aseguramiento debe validar para producir un producto de calidad.

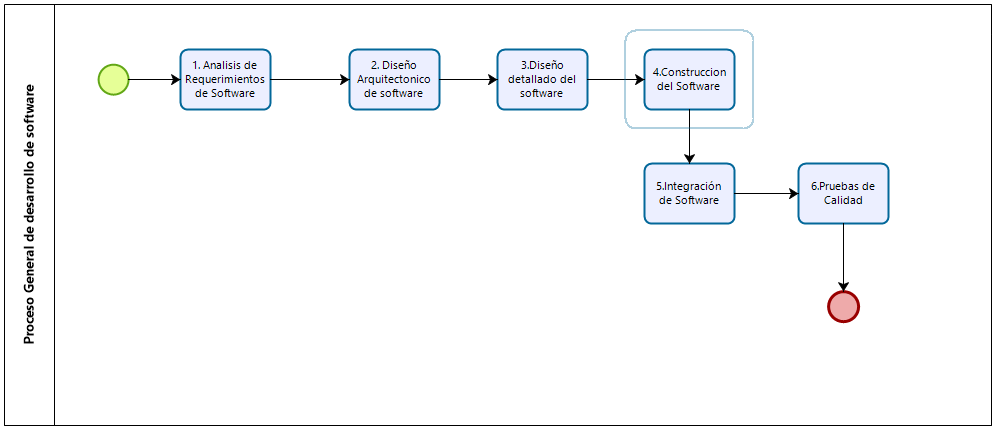
****

Figura 3 - Proceso General -Ciclo de Vida del desarrollo de sw.

## Análisis de requisitos de Software

El proceso de análisis de requisitos permite establecer adecuadamente el alcance asi como los artefactos que recogen todas las especificaciones ya sea funcionales y no funcionales.

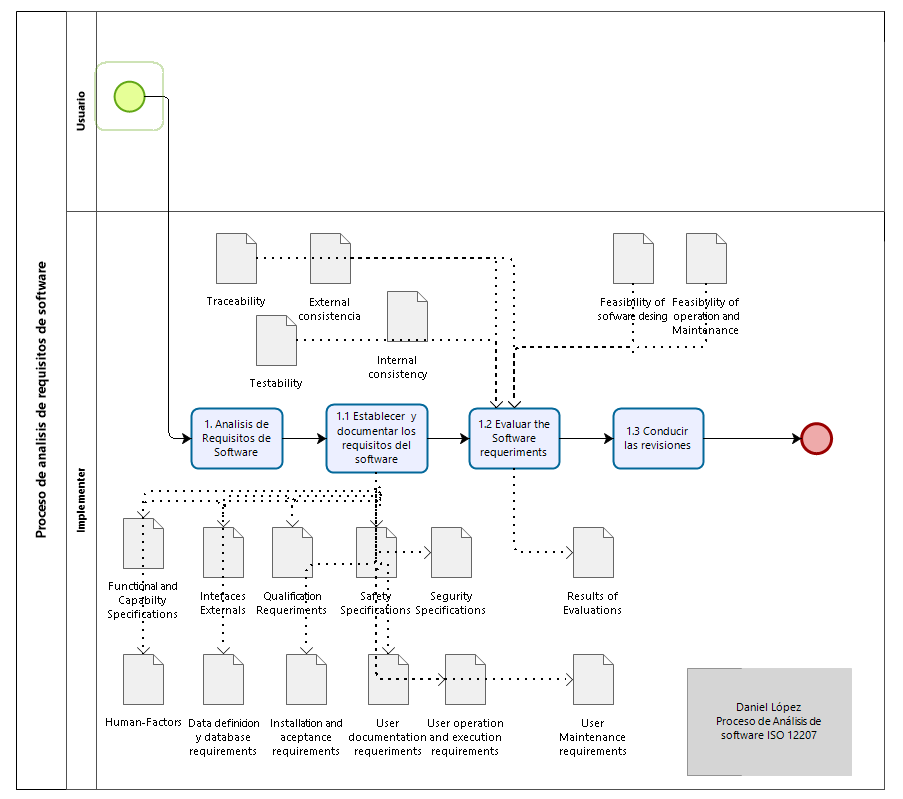


Figura 4 - Proceso de análisis de requisitos

## Proceso de prueba de calificación del Software

En la Figura 5 vemos el detalle del proceso de calificación de prueba, El propósito del proceso de prueba de calificación de software es confirmar que el producto de software integrado cumple con sus requisitos definidos.

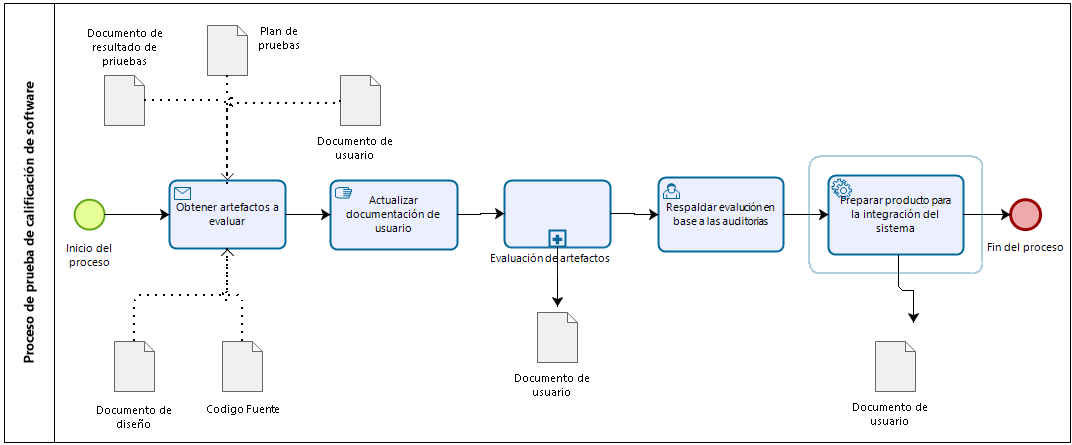
****

Figura 5 - Proceso de prueba de calificación del software

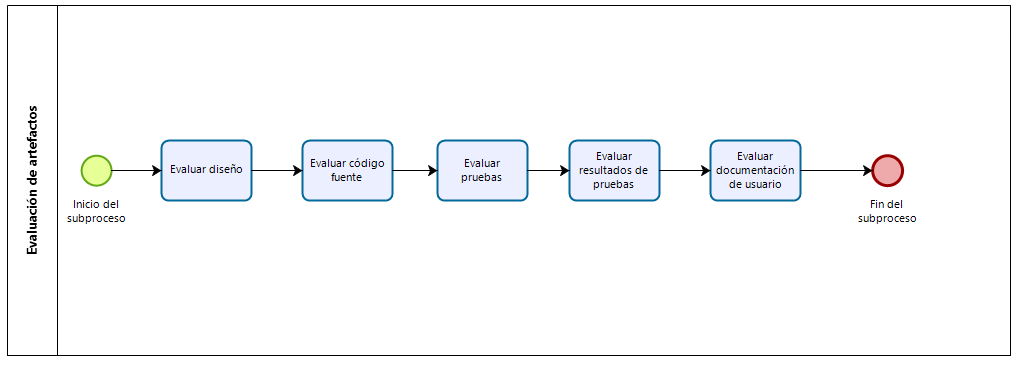
en la figura 6 podemos ver del detalle del proceso de evaluación de artefactos.

Figura 6 - Proceso de Evaluación de artefactos

## Proceso de construcción de software.

El propósito del proceso de construcción de software es producir unidades de software ejecutables que reflejen adecuadamente el diseño del software, en la figura 7 podemos ver el detalle de este proceso.

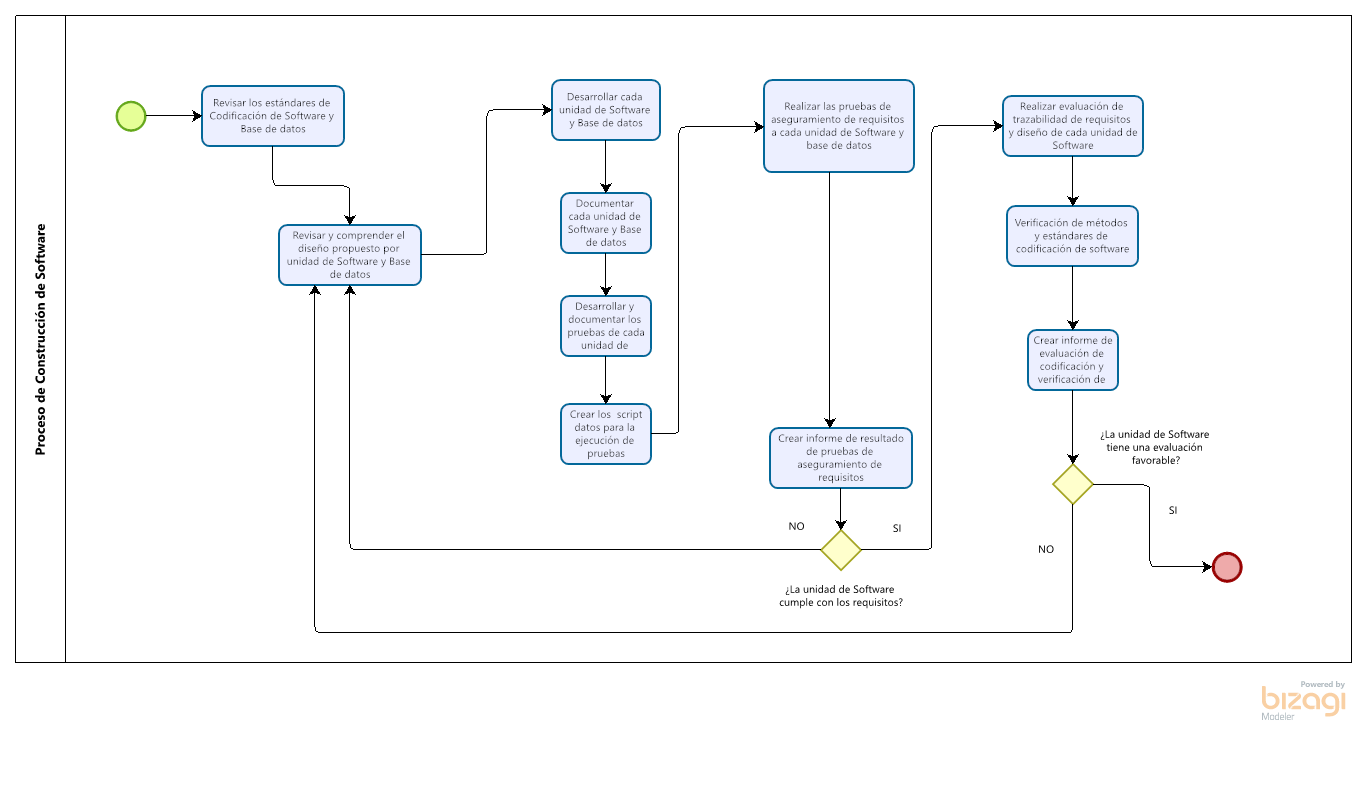
****

Figura 7 - Proceso de construcción de software

## Proceso de Gestión de la configuración.

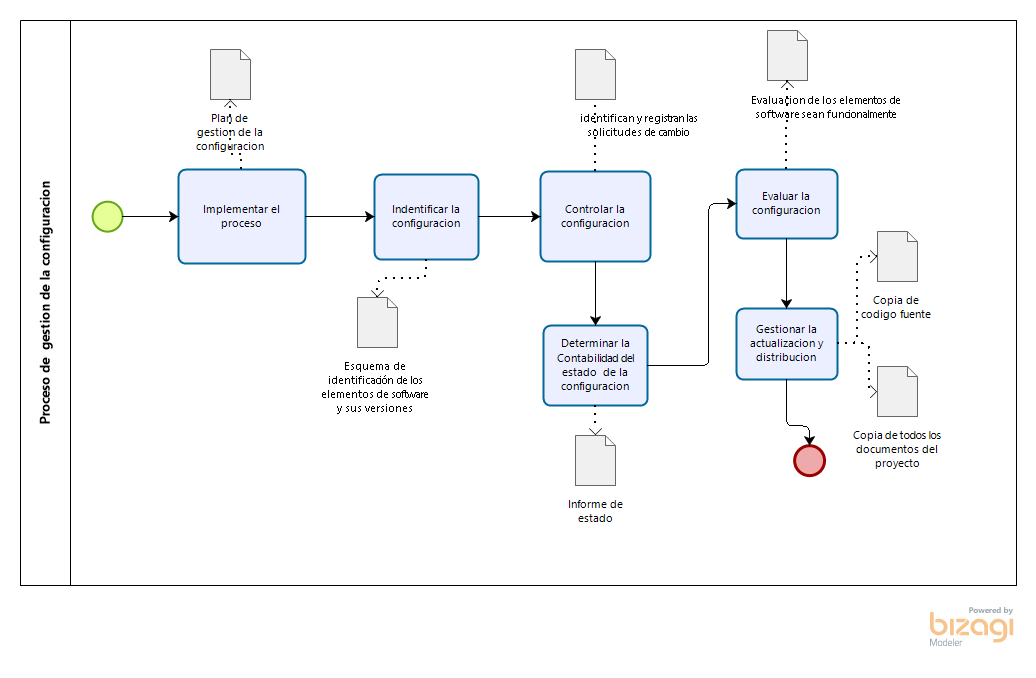
El propósito del proceso de gestión de la configuración de software es establecer y mantener la integridad de los elementos de software de un proceso o proyecto y ponerlos a disposición de las partes interesadas, en la figura 8 podemos ver el detalle de este proceso.

Figura 8 - proceso de gestión de la configuración

## Proceso de Aseguramiento de la calidad.

El propósito del proceso de aseguramiento de la calidad del software es garantizar que los productos de trabajo y los procesos cumplen con disposiciones y planes predefinidos, en la figura 9 vemos el detalle de este proceso.

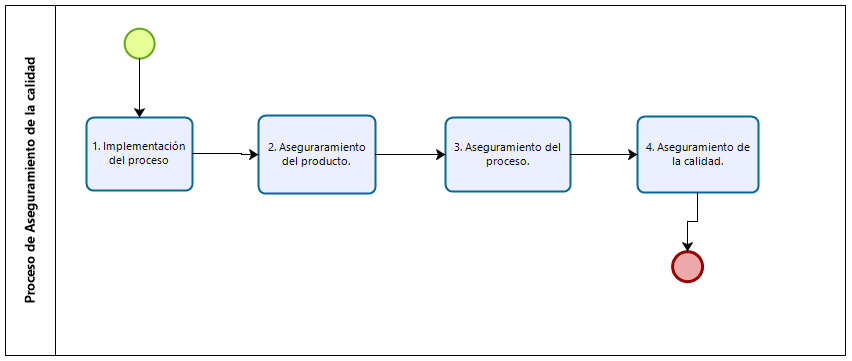
****

Figura 9 - Proceso de aseguramiento de la calidad

# Revisiones, Verificación y auditorias.

## Casos de prueba.

Una de las formas más tecnicas de realizar el testing o las pruebas, es a través de la confección y diseño de casos de pruebas.

A continuación, en la tabla 17 mostramos el formato utilizado para la elaboración de los casos de prueba, describiendo brevemente el uso de cada campo para su correcto llenado.

Tabla 17 - Formato de caso de prueba

|  |  |
| --- | --- |
| id único | “codigo único que identifica a un caso de prueba y permite su trazabilidad” |
| Tipo de Prueba | “Según la caracteristicas de calidad que se va a probar” |
| Funcionalidad | “Nombre de la funcionalidad que se va a probar” |
| Escenario | “Flujo de la funcionalidad que se va a probar” |
| Pre condición | “factores que deben cumplirse para que se pueda ejecutar el caso de prueba” |
| Flujo de Prueba | “Pasos a seguir para realizar la prueba” |
| Resultado Esperado | “Comportamiento esperado de la funcionalidad para este escenario” |
| Post condición | “Estado del sistema despues de ejecutarse la funcionalidad” |
| Listo? | “Indica si el caso está completo”. |
| Estado Ejecución | “indica en que estado esta el caso de prueba” |
| Observación QA | “Comentario del Analista QA sobre la ejecucion del caso” |
| Fecha | “fecha de elaboración del caso de prueba” |

## Casos de prueba UC10-01 Creación de clientes.

El objetivo de este caso descrito en la tabla 18, es verificar el correcto funcionamiento del botón grabar del mantenimiento de clientes del sistema de gestión de perfiles inmobiliarios.

Tabla 18 - Caso de prueba UC10-01

|  |  |
| --- | --- |
| Autor | Daniel López |
| id único | **UC10-01** |
| Tipo de Prueba | **FUNCIONAL** |
| Funcionalidad | Creación de clientes |
| Escenario | Ingreso de un nuevo cliente haciendo click en el botón Grabar |
| Pre condición | 1. ingresar al link http://tienco.dgd-consulting.com/ usuario: @dgd password: dgd 2. hacer clic en configuración 3.hacer clic en clientes 4.El ruc del cliente no debe existir |
| Flujo de Prueba | 1. El usuario hace clic en el botón "nuevo cliente" 2. Completar los siguientes campos: nombre empresa: cliente de prueba1 Nombre Comercial: Nombre comercial 1 RUC:10203040501 Dirección: mi casita 1 master:1 logo: no llenar. 3. hacer clic en el botón grabar. |
| Resultado Esperado | 1. El sistema regresa a la pantalla principal: 2. debe verse el cliente de prueba 1 3. El sistema muestra un mensaje de confirmación del cliente creado. |
| Post condición | Se verifica que el cliente este creado |
| Listo? | 1 |
| Estado Ejecución | Listo para probar |
| Observación QA |  |
| Fecha | **24/07/2021** |

## Casos de prueba UC10-02 Creación de clientes.

El objetivo de este caso descrito en la tabla 19, es verificar el correcto funcionamiento del botón cancelar de la funcionalidad nuevo cliente del sistema de gestión de perfiles inmobiliarios.

Tabla 19 - Casos de prueba UC10-02 Creación de clientes.

|  |  |
| --- | --- |
| Autor | Fiorela Ruiz |
| id único | **UC10-02** |
| Tipo de Prueba | FUNCIONAL |
| Funcionalidad | Creación de clientes |
| Escenario | Ingreso de un nuevo cliente haciendo click en el botón Regresar |
| Pre condición | 1. ingresar al link http://tienco.dgd-consulting.com/  usuario: @dgd  password: dgd 2. hacer clic en configuración 3.hacer clic en clientes |
| Flujo de Prueba | 1. Hacer clic en el botón "nuevo cliente" 2. Completar los siguientes campos nombre empresa: cliente de prueba2 Nombre Comercia: Nombre comercial 2 RUC:10203040502 Dirección: mi casita 2 master:1 logo: no llenar. 3. Hacer clic en el botón regresar. |
| Resultado Esperado | 1. El sistema regresa a la pantalla principal: no debe verse en el sistema el cliente de prueba 2 |
| Post condición | Se verifica que el cliente no esté creado |
| Listo? | 1 |
| Estado Ejecución | Listo para probar |
| Observación QA |  |
| Fecha | **24/07/2021** |

## Casos de prueba UC10-03 Validación de datos de clientes.

El objetivo de este caso descrito en la tabla 20, es verificar el correcto funcionamiento de las validaciones de los campos de la funcionalidad nuevo cliente del sistema de gestión de perfiles inmobiliarios.

|  |  |
| --- | --- |
| Autor | William López |
| id único | UC10-03 |
| Tipo de Prueba | FUNCIONAL |
| Funcionalidad | Creación de clientes |
| Escenario | Ingreso de un nuevo cliente validando los campos obligatorios |
| Pre condición | El usuario ingresa al sistema con los siguientes accesos: 1. Tener acceso al link: http://tienco.dgd-consulting.com/ 2. ingresar el usuario @dgd 3. Ingresar el password dgd Una vez ingresado realizar los siguientes pasos: 4. hacer clic en configuración 5.hacer clic en clientes Es importante que el el ruc del cliente no debe existir. |
| Flujo de Prueba | El usuario debe de seguir los siguientes pasos en el sistema: 1. Hacer clic en el botón "nuevo cliente" 2. Completar los siguientes campos nombre empresa: cliente de prueba1 Nombre Comercial: Nombre comercial 1 RUC:10203040501 Dirección: mi casita 1 master:1 logo: no llenar. 8. hacer clic en el botón grabar. |
| Resultado Esperado | 1. el sistema muestra un mensaje en el campo del RUC: Prolonga a este texto a 11 caracteres o más(En este momento tiene 9 caracteres). |
| Post condición | Se verifica que el cliente no esté creado |
| Listo? | 1 |
| Estado Ejecución | Listo para probar |
| Observación QA |  |
| Fecha |  |

Tabla 20 - Caso de prueba UC10-03 Validación de datos de clientes

## Casos de prueba UC10-04 Eliminar clientes.

El objetivo de este caso descrito en la tabla 21, es verificar el correcto funcionamiento de la funcionalidad eliminar cliente del sistema de gestión de perfiles inmobiliarios.

Tabla 21 - Caso de prueba UC10-04 Eliminar cliente

|  |  |
| --- | --- |
| Autor | Félix Cáceres |
| id único | UC10-04 |
| Tipo de Prueba |  |
| Funcionalidad | Creación de clientes |
| Escenario | Eliminación de un cliente presionando el botón aceptar |
| Pre condición | 1. ingresar al link http://tienco.dgd-consulting.com/  usuario: @dgd  password: dgd 2. Hacer clic en configuración 3. Hacer clic en clientes 4. El ruc del cliente debe existir |
| Flujo de Prueba | 1. El usuario hacer clic en el botón "eliminar". 2. El sistema muestra un mensaje solicitando la eliminación del cliente. 2. El usuario hacer clic en el botón aceptar. |
| Resultado Esperado | 1. El sistema eliminará el registro del cliente seleccionado. |
| Post condición | El cliente ya no se muestra en la pantalla. |
| Listo? | 1 |
| Estado Ejecución | Listo para probar |
| Observación QA |  |
| Fecha |  |

## Casos de prueba UC10-05 Eliminar clientes.

El objetivo de este caso descrito en la tabla 22, es verificar el correcto funcionamiento del botón cancelar de la funcionalidad eliminar cliente del sistema de gestión de perfiles inmobiliarios.

Tabla 22 - Caso de prueba UC10-05 Eliminar cliente

|  |  |
| --- | --- |
| Autor | Rony |
| id único | UC10-05 |
| Tipo de Prueba |  |
| Funcionalidad | Creación de clientes |
| Escenario | Cancelar la eliminación de un cliente |
| Pre condición | 1. ingresar al link http://tienco.dgd-consulting.com/  usuario: @dgd  password: dgd 2. hacer clic en configuración 3.hacer clic en clientes |
| Flujo de Prueba | 1. Elegir el usuario a eliminar. 2. Hacer clic en el botón "eliminar" 3. Hacer clic en el botón cancelar |
| Resultado Esperado | 1. El sistema muestra un mensaje solicitando confirmación de la eliminación |
| Post condición | El cliente no se elimina y permanece en la pantalla |
| Listo? | 1 |
| Estado Ejecución | Listo para probar |
| Observación QA |  |
| Fecha |  |

## Métrica para la simulación.

Para medir el resultado de la ejecución, se platea la métrica detallada en la tabla 23 para establecer el resultado, esta métrica debe estar relacionada con el tipo de caso de prueba que se van a ejecutar.

Tabla 23 - Métrica para la simulación

|  |  |
| --- | --- |
| **NOMBRE DE LA METRICA:** | Medición de cumplimiento de funcionalidades del sistema |
| **CODIGO:** | M2 |
| **CARACTERISTICA:** | FUNCIONALIDAD |
| **SUBCARACTERISTICA:** | CONFORMIDAD DE LA FUNCIONALIDAD |
| **PROPOSITO:** | ¿Cuántas funcionalidades cumple el sistema correctamente? |
| **MEDICIÓN, FORMULA Y ELEMENTOS MEDIBLES:** | T: Total de casos de prueba definido en módulo de configuración del sistema de gestión de perfiles. Q: Cantidad de casos de prueba que fueron ejecutados exitosamente. X: Q/T. |
| **INTERPRETACION DEL VALOR** | El módulo de configuración del sistema de gestión de perfiles, tiene X conformidad de la funcionalidad. 0<=x<=1, donde 1 es el valor óptimo. |
|  |  |

## Simulación.

Para realizar la simulación, cada miembro del equipo ejecuta todos los casos de prueba y estable si fue exitoso o no el caso de prueba para simplificar 1 significa exitoso y 0 no exitoso. En la tabla 24 se resumen estas ejecuciones y su respectivo resultado.

Tabla 24 - Resumen de la ejecucion de los casos de prueba

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **UC10-01** | **UC10-02** | **UC10-03** | **UC10-04** | **UC10-05** |
| Daniel | 1 | 1 | 0 | 1 | 1 |
| Fiorela | 1 | 1 | 0 | 1 | 1 |
| William | 1 | 1 | 0 | 1 | 1 |
| Félix | 1 | 1 | 1 | 1 | 0 |
| Rony | 1 | 1 | 1 | 1 | 1 |

Una vez realizada la ejecución de los casos de prueba, aplicamos la métrica definida, en la tabla 25 vemos un resumen de esta aplicación.

Tabla 25 - Aplicacion de la métrica

|  |  |  |  |
| --- | --- | --- | --- |
| **NOMBRE DE LA METRICA:** | Medición de cumplimiento de funcionalidades del sistema | | |
|  | **T** | **Q** | **Q/T** |
| Daniel | 5 | 4 | 0.8 |
| Fiorela | 5 | 4 | 0.8 |
| William | 5 | 4 | 0.8 |
| Félix | 5 | 4 | 0.8 |
| Rony | 5 | 5 | 1 |

Para efectos académicos hemos sacado un promedio obteniendo lo siguiente

Promedio 0.84 84.00%

El valor más cercano a 100%.

# Reporte de problemas y acciones correctivas

El reporte de problemas y sus respectivas acciones correctivas es un aspecto muy importante ya que no hay sistema perfecto y durante la ejecucion de las pruebas debe realizarse la gestión de estos problemas y la verificación de la aplicación de las acciones de correctivas.

## Procedimiento de registro de problemas.

Durante la ejecucion de los casos de prueba el probador o tester necesita llevar un registro que le permita llevar un control de todos los incidentes, este incidente debe ser tipificado estableciendo el tipo de incidente, establecer si es un bug o una vulnerabilidad, establecer una criticidad que nos permita saber la gravedad del incidente y un estado que nos permita llevar el control de como se resuelven ya que estos incidentes deben ser reportados al equipo de desarrollo para se corrijan y se prueben nuevamente para establecer si se levanto correctamente.

## Procedimiento de registro de problemas.

Para llevar el control de los incidentes proponemos el siguiente formato de la tabla 26

Tabla 26 - Formato de registro de incidentes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Caso de Prueba | Incidente | Estado | Criticidad | Ambiente | Observación |
| UC10-01 |  |  |  |  |  |
| UC10-02 |  |  |  |  |  |
| UC10-03 |  |  |  |  |  |
| UC10-04 |  |  |  |  |  |
| UC10-05 |  |  |  |  |  |
| UC10-06 |  |  |  |  |  |
| UC10-07 |  |  |  |  |  |
| UC10-08 |  |  |  |  |  |
| UC10-09 |  |  |  |  |  |

# Herramientas, técnicas y metodologías

Debido a la complejidad de este proceso es necesaria la utilización de herramientas que soporten la gestión de este proceso. Podemos encontrar una serie de herramientas que nos ayuden en la ejecucion de este plan tanto para la gestión de proyectos de pruebas, como para la ejecucion, incluidas las de automatización.

## SonarQube.

Esta herramienta nos permite hacer pruebas estáticas de codigo validando la calidad de codigo enmarcado en lo que se denomina calidad interna del producto de software.

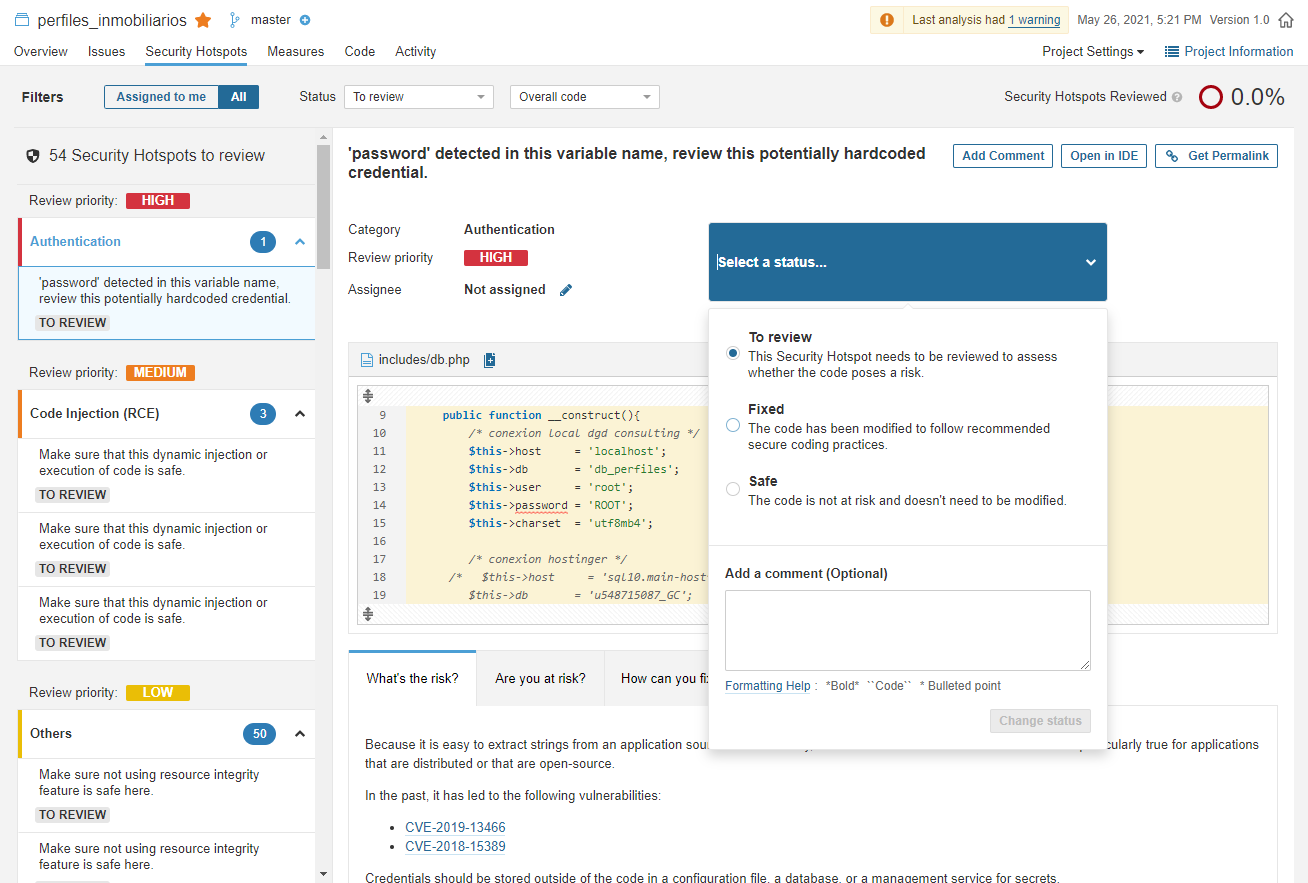


Figura 10 - Sonarqube

## Katalon.

Otra herramienta que encontramos de mucha utilidad es el katalon, esta herramienta nos permite la automatización de pruebas generando script que nos permiten repetir las pruebas de manera automática, muy útil en las denominadas pruebas de regresión.