**APLICACIÓN DE UN MODELO DE CALIDAD AL MÓDULO DE COMUNICACIÓN Y TRASPASO DE INFORMACIÓN DEL SISTEMA DE GESTIÓN DE DATOS AMBIENTALES DE LA EMPRESA SAN AMBIENTE**

****

**JESSICA DANIELA ROMERO**

**FABIAN ALEXIS MINA POSU**

**INSTITUCION UNIVERSITARIA ANTONIO JOSE CAMACHO**

**FACULTAD DE INGENIERIA**

**INGENIERIA EN SISTEMAS**

**SANTIAGO DE CALI 2019**

**APLICACIÓN DE UN MODELO DE CALIDAD AL MÓDULO DE COMUNICACIÓN Y TRASPASO DE INFORMACIÓN DEL SISTEMA DE GESTIÓN DE DATOS AMBIENTALES DE LA EMPRESA SAN AMBIENTE**

****

**JESSICA DANIELA ROMERO**

**FABIAN ALEXIS MINA POSU**

**ING. BEATRIZ EUGENIA MARIN**

**INSTITUCION UNIVERSITARIA ANTONIO JOSE CAMACHO**

**FACULTAD DE INGENIERIA**

**INGENIERIA EN SISTEMAS**

**SANTIAGO DE CALI 2019**

# **RESUMEN**

## APLICACIÓN DE UN MODELO DE CALIDAD AL MÓDULO DE COMUNICACIÓN Y TRASPASO DE INFORMACIÓN DEL SISTEMA DE GESTIÓN DE DATOS AMBIENTALES DE LA EMPRESA SAN AMBIENTE

Existen diferentes modelos de calidad de software que permiten un proceso de mejora continua con su implementación. En el siguiente trabajo se exponen características de los modelos investigados a través de una revisión de información, se realiza una clasificación de estos de acuerdo con el nivel de procesos; cada modelo cuenta con unas características y una estructura la cual nos permite elegir qué modelo de calidad se puede implementar de acuerdo con el proyecto.

La temática que se encuentra en la implementación de un desarrollo de software se da acorde a la proporción de productos y servicios de calidad estandarizados por las normas que se pueden dar a conocer y se practican en la ingeniería de software que se van implementando acorde al desarrollo estipulado en este caso.

La aplicación del modelo de calidad para evaluación del software desarrollado esta basada en la Norma ISO 2501n y ISO 2502n aplicando fases de metodología descriptas también en base a la IEEE829 que nos comparte la estructura necesaria y pasos aplicables a la implementación de las pruebas a realizar para el desarrollo del proyecto.

# **ABSTRAC**

## APPLICATION OF A QUALITY MODEL TO THE MODULE OF COMMUNICATION AND TRANSFER OF INFORMATION OF THE SYSTEM OF MANAGEMENT OF ENVIRONMENTAL DATA OF THE COMPANY SAN AMBIENTE

There are different software quality models that allow a process of continuous improvement with its implementation. In the following work, characteristics of the investigated models are exposed through a review of the information, a classification of these is made with the level of processes; Each model has characteristics and a structure that allows us to choose a quality model.

The theme that is found in the implementation of a software development is given in a proportion of standardized quality products and services in the standards that can be made known and practiced in the software engineering that will be implemented in the development stipulated in this case.

The application of the quality model for software evaluation is based on ISO 2501n and ISO 2502n phases of the description of the methods in the IEEE829 database are applied. The development of the project.

Contenido

[**RESUMEN** 4](#_Toc30776993)

[APLICACIÓN DE UN MODELO DE CALIDAD AL MÓDULO DE COMUNICACIÓN Y TRASPASO DE INFORMACIÓN DEL SISTEMA DE GESTIÓN DE DATOS AMBIENTALES DE LA EMPRESA SAN AMBIENTE 4](#_Toc30776994)

[**ABSTRAC** 5](#_Toc30776995)

[APPLICATION OF A QUALITY MODEL TO THE MODULE OF COMMUNICATION AND TRANSFER OF INFORMATION OF THE SYSTEM OF MANAGEMENT OF ENVIRONMENTAL DATA OF THE COMPANY SAN AMBIENTE 5](#_Toc30776996)

[**INTRODUCCIÓN** 8](#_Toc30776997)

[**1.) PLANTEAMIENTO DEL PROBLEMA** 9](#_Toc30776998)

[**1.1)** **PREGUNTA PROBLEMA** 13](#_Toc30776999)

[**2. OBJETIVOS** 13](#_Toc30777000)

[2.1) . OBJETIVO GENERAL 13](#_Toc30777001)

[2.2) OBJETIVOS ESPECIFICOS 13](#_Toc30777002)

[**3)** **JUSTIFICACIÓN** 15](#_Toc30777003)

[**3.1) ALCANCE** 17](#_Toc30777004)

[**4)** **MARCO REFERENCIAL** 18](#_Toc30777005)

[**4.1) ANTECEDENTES** 18](#_Toc30777006)

[**4.2) MARCO CONCEPTUAL** 21](#_Toc30777007)

[**4.3) MARCO TEÓRICO** 24](#_Toc30777008)

[**4.3.1) Investigación Metodologías** 25](#_Toc30777009)

[**4.3.2) ISO 25010** 27](#_Toc30777010)

[**4.3.3) NORMA IEE 829** 30](#_Toc30777011)

[**4.4) MARCO LEGAL** 31](#_Toc30777012)

[**5)** **METODOLOGÍA** 31](#_Toc30777013)

[**5.1) Fuentes y técnicas** 31](#_Toc30777014)

[**5.2) Tipos de investigación** 32](#_Toc30777015)

[**5.3) Metodología de desarrollo** 32](#_Toc30777016)

[**FASES PARA SEGUIR** 33](#_Toc30777017)

[**I.** **Programación de pruebas** 33](#_Toc30777018)

[**II.** **EJECUCION DE PRUEBAS** 33](#_Toc30777019)

[**III.** **TERMINO DE LAS PRUEBAS** 33](#_Toc30777020)

[**6)** **TIPOS DE DOCUMENTOS** 34](#_Toc30777021)

[**6.1. PREPARACION DE PRUEBAS** 34](#_Toc30777022)

[**6.1.1 PLAN DE PRUEBAS** 34](#_Toc30777023)

[**7** **CONCLUSIONES** 107](#_Toc30777024)

[**8** **BIBLIOGRAFÍA** 107](#_Toc30777025)

**Lista de imágenes**

**Lista de Tablas**

# **INTRODUCCIÓN**

En este proyecto se aplica un modelo de calidad a el aplicativo que será desarrollado para la empresa san ambiente, el cual va a permitir gestionar todos los datos ambientales que la empresa recoge en determinadas regiones, el modelo de calidad se aplicara antes y durante la etapa de desarrollo del software para así asegurar que todos los aspectos requeridos por el cliente se cumplan y se entrega al final un producto útil. El modelo de calidad incluye desde el levantamiento de requerimientos, metodología de trabajo, diseño de casos de prueba y resultados de casos de prueba aplicados, siguiendo los estándares de las normas ISO 25010 y la norma IEEE 829 para la ejecución de las pruebas a las cuales será sometido el software antes de ser entregado al cliente.

Para ello se refinaron y se elaboraron artefactos como lo son los casos de uso, historias de usuario para tener una documentación más completa para la elaboración de los 8 documentos de la norma IEEE 829 y tener una mejor visión del estado del aplicativo. Mediante pruebas de usabilidad, caja negra, pruebas funcionales, y como resultado se obtiene una documentación más detallada de el aplicativo control currículo y una visión para futuras mejoras a aplicativo

## **1.) PLANTEAMIENTO DEL PROBLEMA**

La empresa San Ambiente ofrece soluciones integrales en productos y servicios que aporten al mejoramiento de las condiciones ambientales y medición de parámetros ambientales, orientados a satisfacer las necesidades y requerimientos de los clientes. Busca el mejoramiento continuo de sus procesos y la productividad de la organización a través del compromiso y la participación de los empleados, socios y proveedores, lo que les ha permitido ser reconocidos en el mercado por la seriedad, profesionalismo en la prestación del servicio.

El problema se centra en que al no implementar un modelo de calidad en el desarrollo del aplicativo se desencadenaría una serie de inconformidades que afectarían el avance de este mismo, en las que se incluyen retrasos en los tiempo de entrega del producto, inconformidad del cliente, no habría rendimiento en las actividades, no se ejecutarían las pruebas del software antes de ser entregado al cliente, esto llevaría a que en caso de un error de código se tendría que hacer un replanteamiento y tardaría mucho más tiempo.

Se desarrollará un software para la empresa San ambiente el cual va a permitir gestionar los datos ambientales que la empresa recoge; durante el desarrollo de este software se llevara a cabo la implementación de un modelo de calidad, para así asegurar que el usuario final reciba un producto con todos los requerimientos especificados.

|  |  |  |  |
| --- | --- | --- | --- |
| SINTOMA | CAUSA | EFECTO | CONSECUENCIA |
| Inconvenientes en la entrega de aplicativo | Demoras en el proceso | Retraso en todo el proceso de desarrollo del software. | Inconformidad por parte del cliente |
| Errores de código | Plan de pruebas no realizado | Replanteamiento del código para los desarrolladores | Inconformidad del cliente |
| No realizar casos de uso y levantamiento de requerimientos | Desvió del objetivo | Desarrolladores tomen otras decisiones. | Inconformidad del cliente al no recibir el producto esperado |
| DIAGNÓSTICO | | PRONÓSTICO | |

**DIAGRAMA CAUSA EFECTO**

Cliente Insatisfecho

Demoras en el proceso

Incumplimiento en tiempo de entrega

No se tiene claro que se desarrollara

Funcionalidad no es correcta

No implementar un debido modelo de calidad en la construcción de un producto de software.

No se tienen en cuenta requisitos obtenidos

No se desarrollo un plan de pruebas

Aplicación con errores

Desvió del objetivo

### **PREGUNTA PROBLEMA**

- ¿Cómo asegurar la calidad de todos los módulos en el proceso de desarrollo del aplicativo de la empresa San Ambiente?

**1.)2 SISTEMATIZACIÓN**

- ¿Cuáles estrategias de calidad se pueden implementar en el ciclo de vida del aplicativo de comunicación y traspaso de la empresa San Ambiente?

- ¿Cómo se pueden validar los elementos de calidad alcanzados en el producto de San Ambiente al igual que la satisfacción del cliente?

# **2. OBJETIVOS**

## 2.1) . OBJETIVO GENERAL

Aplicar un modelo de calidad al módulo de comunicación de información del sistema De gestión de datos ambientales de la empresa San Ambiente de Cali.

## 2.2) OBJETIVOS ESPECIFICOS

- Definir los elementos necesarios para asegurar la calidad en el proceso de desarrollo del producto.

- Desarrollar un plan de evaluación del producto de software.

- Generar el plan de evaluación sobre el producto y presentar sus resultados.

|  |  |  |
| --- | --- | --- |
| OBJETIVOS | ACTIVIDADES | RESULTADOS |
| - Documentar los  elementos necesarios  para asegurar la  calidad en el proceso  de desarrollo del  producto | Levantar requerimientos, Modelado de casos de uso, Modelado de MER. Definir arquitectura y patrones de diseño. | Tener un informe en el que se detalle, sea fácil de comprender y ayude a los desarrolladores en el proceso de creación |
| - Establecer un plan de  evaluación del  producto de software. | Realizar diseño de pruebas (Plan de pruebas) | Tener un informe con todos los planes de prueba a ejecutar. |
| - Ejecutar el plan de  evaluación sobre el  producto y presentar  sus resultados | Realizar pruebas funcionales al aplicativo | Tener un informe con todo el análisis y resultados del plan de pruebas realizado.  Entregar un producto útil al usuario final. De igual manera definir muy bien el producto. |

Tabla 1. Objetivos

Autor: Fabian Mina – Daniela Romero

# **JUSTIFICACIÓN**

El presente trabajo se enfocara en modelar y aplicar un modelo de calidad y su respectivo plan de pruebas para la entrega de la aplicación gestión de datos ambientales para la empresa San ambiente El modelo de calidad aplicado en este proyecto servirá para direccionar al equipo de desarrollo a cumplir con todos los objetivos y requerimientos establecidos, al igual que un plan de pruebas que basa en ocuparse de todos los mecanismos, acciones y herramientas para detectar errores o fallas en el aplicativo que será entregado. Para realizar estos controles se implementan estándares que deben cumplirse para entregar al final un producto de calidad.

Este trabajo servirá como una guía para la realización de proyectos relacionados con la aplicación del modelo de calidad y plan de pruebas a diferentes; se espera que con este proyecto se realice la entrega del aplicativo a la empresa San ambiente cumplimento con todos los requisitos demandados por el cliente y lograr la satisfacción de este mismo.

Según Manuel Prego (Revista digital CW,2004), director general de ESI, considera que, conceptualmente, el auge de los procesos de calidad se debe a que el software está cada vez más “inmerso en los productos y servicios que producimos.

En las empresas de desarrollo de software, la tendencia que tienen es realizar el proceso de las pruebas de software en la última etapa para consolidar la calidad de su producto.

Cada actividad en el desarrollo de software debe ir distribuido a una actividad de pruebas de calidad, para esto se implementan los modelos de calidad los cuales sirven de guía para que el equipo de desarrollo tenga más claro el proceso que se debe seguir durante el desarrollo del software.

# **3.1) ALCANCE**

Verificar el cumplimiento de las normas y estándares de calidad parametrizados por los requerimientos solicitados en el desarrollo, con el fin de que sea un aplicativo que cumpla su funcionalidad, se manejara la validación concreta de los estándares de calidad con planes de pruebas aplicados a este desarrollo obteniendo pruebas exactas de tolerancias a fallos y de mas criterios manejados en las normas de calidad implementadas.

# **MARCO REFERENCIAL**

## **4.1) ANTECEDENTES**

El objetivo de esta sección es mostrar trabajos relacionados con la normativa de gestión de calidad ISO/IEC 2501n que habla sobre la división de modelo de calidad y actualmente se dividen de la siguiente forma la ISO/IEC 25010 system and software quality models, y ISO/IEC 25012 Data Quality model; lo cual en las investigaciones anteriores se tiene presente la calidad de uso y un modelo en general para la calidad, para todos los datos que se almacenan de una forma estructurada.

En (Callejas-Cuervo, Alarcón-Aldana, & Álvarez-Carreño, 2017) Los modelos de calidad del software, un estado del arte para la universidad pedagógica y tecnológica de Colombia, en el cual se pretende explicar los términos propios de calidad de software y los diferentes modelos de calidad aplicados para la revisión de calidad, realizando la clasificación de modelos acorde a (Proceso, producto y uso) existen varios modelos de calidad los cuales realizan procesos de mejora continua con su implementación, debido que para que un sistema informático complete un ciclo de vida debe tener características eficientes para así garantizar como conclusión se evidencio que en base a modelos anteriores han sido de beneficio para los más actuales, las empresas desarrolladoras deben someter su software a certificación que este bajo una norma o estándar de calidad.

En Estévez (2013) la investigación: llamada modelo de calidad para evaluar el software desarrollado en el centro de investigación aplicada y desarrollo en tecnologías de información CIADTI, donde su principal enfoque se hizo en los modelos de calidad del software para los productos desarrollados en la universidad de pamplona ya que un software sometido a un proceso de calidad es más apetecible para el mercado a la hora de realizar ventas, como resultado les fue posible establecer que no había una herramienta o proceso que permitiera evaluar la calidad del software, dado esto utilizaron varios modelos que se diferenciaban sobre los parámetros que iban a ser evaluados y tomaron en cuenta para la realización el diagnostico el diseño y aplicación.

En Scalone (2006) el estudio comparativo de los modelos y estándares de calidad del software, la forma de escoger el modelo o estándar de calidad adecuado que permita hacer determinación de los mejores resultados evitando que haya una mala inversión en recursos, tiempo y costos, se enfocó en las empresas que hacen desarrollo y que están interesadas en someterse a un proceso de certificación de calidad y así aumentar la competencia que ayudara a acceder a nuevos mercados. esto se encuentra en un contexto económico y empresarial que influencia a la hora de determinar un modelo o estándar de calidad adecuado como conclusión se encontró que el software maneja un papel fundamental para el desarrollo de las empresas esto quiere decir que es fundamental aplicar un modelo de calidad para evaluar un software.

En Ramos, Quintana (2014) el tema de modelo de mejora de procesos para la calidad del software basado en CMMI para una entidad financiera, donde se encuentra la inadecuada gestión de los requisitos para el desarrollo de los proyectos generando cuellos de botella, se encuentra una sobrecarga de trabajo que es inadecuado y no permite la entrega de un proceso con eficiencia, se hicieron mejoras donde los requisitos de gestión en un 82.55% lo cual permitió asegurar la alineación entre ellos, se desarrolló un proceso de monitoreo y control de los proyectos, se hizo mejora en 90%, lo cual permitirá ofrecer y dar soporte a la entrega de productos con calidad, se cumplió con los objetivos principales del desarrollo propuesto de CMMI nivel 2 mejorando todos los procesos en un 65.47% en el ciclo de vida del desarrollo del software.

## **4.2) MARCO CONCEPTUAL**

Este proyecto se realizó tomando como referencia el proyecto de grado, “Implementación de pruebas de calidad en la aplicación currículo de la Institución Universitaria Antonio José Camacho” (Aguado,2019), al cual se aplicará un modelo de calidad y un modelo de pruebas de calidad durante la etapa de desarrollo para prevenir errores a largo plazo. Lo anterior se basa en pruebas unitarias, pruebas en la base de datos, análisis de errores en el código y en la documentación de las fases del desarrollo.

#### **Adecuación Funcional**

Representa la capacidad del producto software para proporcionar funciones que satisfacen las necesidades declaradas e implícitas, cuando el producto se usa en las condiciones especificadas. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

* Completitud funcional.
* Corrección funcional.
* Pertinencia funcional.

#### **Eficiencia de desempeño**

Esta característica representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

* Comportamiento temporal.
* Utilización de recursos.
* Capacidad.

**Compatibilidad**

Capacidad de dos o más sistemas o componentes para intercambiar información y/o llevar a cabo sus funciones requeridas cuando comparten el mismo entorno hardware o software. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

* Coexistencia.
* Interoperabilidad.

**Usabilidad**

Capacidad del producto software para ser entendido, aprendido, usado y resultar atractivo para el usuario, cuando se usa bajo determinadas condiciones. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

* Capacidad para reconocer su adecuación.
* Capacidad de aprendizaje.
* Capacidad para ser usado
* Protección contra errores de usuario.
* Estética de la interfaz de usuario.
* Accesibilidad.

**Fiabilidad**

Capacidad de un sistema o componente para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

**Madurez.**

Capacidad del sistema para satisfacer las necesidades de fiabilidad en condiciones normales.

**Disponibilidad**.

Capacidad del sistema o componente de estar operativo y accesible para su uso cuando se requiere.

**Tolerancia a fallos.**

Capacidad del sistema o componente para operar según lo previsto en presencia de fallos hardware o software.

**Capacidad de recuperación.**

Capacidad del producto software para recuperar los datos directamente afectados y reestablecer el estado deseado del sistema en caso de interrupción o fallo.

## **4.3) MARCO TEÓRICO**

El presente trabajo se realiza con el fin de aplicar un modelo de calidad para el modulo de comunicación y traspaso de información del sistema de gestión de datos ambientales de la empresa SAN AMBIENTE teniendo en cuenta que en su etapa de desarrollo es de gran magnitud definir las metodologías que se apicaran, para que sea una herramienta útil que pueda validar los procesos que se han ido desarrollando acorde a los aspectos funcionales;

Se aplicará un modelo de calidad que contribuya con las especificaciones que se hayan solicitado por parte de la empresa san ambiente ya que el manejo de la plataforma es de gran ayuda para llevar un control sobre los documentos de forma ordenada.

El desarrollo de aplicaciones web ha permitido hacer de manera más eficiente la automatización de procesos que antes requerían mucha demanda de tiempo lo cual retrasaba la ejecución de tareas, por lo tanto, se requiere el análisis y evaluación de los aplicativos que se vayan a entregar o a modificar debido a que un software con calidad tiene más rentabilidad y confiabilidad por parte del cliente que desea adquirirlo ya que este debe cumplir con todos los requerimientos funcionales y de rendimiento solicitados.

La aplicación de las pruebas de calidad se trabaja con el fin de mostrar la forma efectiva de medir la calidad del aplicativo en cada etapa de su desarrollo para así brindar a sus clientes servicios y productos de alto nivel demandado.

Para este proyecto se tiene en cuenta que la Norma más Completa para aplicar el control de calidad es la ISO 25000 conocida como SQuaRE (System and Software Quality Requirements and Evaluation), el cual es una familia de normas que tiene por objetivo evaluar la calidad del producto software.

**APLICACIÓN DE UN MODELO DE CALIDAD AL MÓDULO DE COMUNICACIÓN Y TRASPASO DE INFORMACIÓN DEL SISTEMA DE GESTIÓN DE DATOS AMBIENTALES DE LA EMPRESA SAN AMBIENTE**

### **4.3.1) Investigación Metodologías**

|  |  |  |  |
| --- | --- | --- | --- |
| **METODOLOGIA IEEE-829** | **PRODUCTO** | **METODOLOGIA** **ISO / IEC29119** | **PRODUCTO** |
| **Fase1: Preparación de pruebas** | * Plan de pruebas * Especificación del diseño de pruebas * Especificación de casos de prueba * Procedimiento de pruebas * Reporte de transmisión de ítems de pruebas | **Fase1: Definiciones y Vocabulario** | Aplica Fases, pero no relaciona un producto |
| **Fase 2: Proceso de pruebas** |
| **Fase3: Documentación de prueba** |
| **Fase2: Ejecución de las pruebas** | * Log de pruebas * Reporte de incidentes de pruebas |
| **Fase3: Termino de las pruebas** | * Reporte de las pruebas | **Fase4: Técnicas de Prueba** |
|  | | **Fase5:** **Pruebas dirigidas por palabras clave** |

Tabla 2. Fases aplicación de pruebas

Autor: Fabian Mina – Daniela Romero

#### **¿QUÉ FUNCIÓN CUMPLE CADA PRODUCTO EN LA FASE?**

|  |
| --- |
| **WEBQEM SQAE EUROPEO Y AMERICANO ESTANDAR** |
| **Plan de pruebas:** Es un documento eje sobre el cual se desarrollan las pruebas, se define el alcance, el enfoque de recursos y también identifica claramente los ítems y características a probar (herramientas y que se va a utilizar para el plan de pruebas) |
| **Especificación del diseño de pruebas:** Se determina qué necesita ser probado y cómo sería una prueba exitosa. |
| **Especificación de casos de prueba:** Valores exactos de entrada y otros que se requieran.  Valores exactos de salida y cambios del sistema esperados. (resultados que debería de ingresar y dar)  Se especifican pasos para ejecutar las pruebas y identificar las tareas de pruebas a desarrollar. |
| **Procedimientos de pruebas:** Describe cómo el tester ejecutará físicamente la prueba y los pasos necesarios. |
| **Reporte de transmisión de ítems de pruebas:** Describe los ítems para prueba, dónde encontrarlos y da la aprobación para su liberación.  Es importante porque garantiza de que los ítems están listos para ser probados. |
| **Log de pruebas:** Registra los detalles sobre qué casos de pruebas se han ejecutado, en qué orden y sus resultados.  Si hay inconformidades, se levanta o actualiza un reporte de incidentes |
| **Reporte de incidentes de prueba:** Descripción de los detalles encontrados cuando la  prueba no pasó. |
| **Reporte de pruebas:** Resume la información importante sobre las pruebas, incluyendo:  ● evaluación de qué tan bien se realizaron las pruebas,  ● número de incidentes reportados,  ● evaluación sobre la calidad del sistema  El documento es importante porque sirve para decidir si la calidad del  sistema es suficiente para continuar. |

Tabla 3. Definición proceso de pruebas

Autor: Fabian Mina – Daniela Romero

### **4.3.2) ISO 25010**

En este modelo de calidad se determinan las características de calidad que se van a evaluar a un producto de software determinado. Se tienen en cuenta aspectos como la adecuación funcional, eficiencia de desempeño, la compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad y portabilidad, criterios que se evalúan conforme a los requerimientos solicitados por los usuarios los cuales tienen un valor referente a evaluar para así garantizar la calidad de las características y subcaracteristicas del producto, se procederá a aplicar las ISO 2501n que es la división para el modelo de calidad y 2502n se basa en la división para la medición de calidad.

* **ISO/IEC 25000** - *Guide to SQuaRE*: contiene el modelo de la arquitectura de SQuaRE, la terminología de la familia, un resumen de las partes, los usuarios previstos y las partes asociadas, así como los modelos de referencia.
* **ISO/IEC 25001** - *Planning and Management*: establece los requisitos y orientaciones para gestionar la evaluación y especificación de los requisitos del producto software.
* **ISO/IEC 2501n – División de Modelo de Calidad**
* Las normas de este apartado presentan modelos de calidad detallados incluyendo características para calidad interna, externa y en uso del producto software. Actualmente esta división se encuentra formada por:
* **ISO/IEC 25010 - *System and software quality models*:** describe el modelo de calidad para el producto software y para la calidad en uso. Esta Norma presenta las características y subcaracterísticas de calidad frente a las cuales evaluar el producto software.
* **ISO/IEC 25012 - *Data Quality model*:** define un modelo general para la calidad de los datos, aplicable a aquellos datos que se encuentran almacenados de manera estructurada y forman parte de un Sistema de Información.
* **ISO/IEC 2502n – División de Medición de Calidad**
* Estas normas incluyen un modelo de referencia de la medición de la calidad del producto, definiciones de medidas de calidad (interna, externa y en uso) y guías prácticas para su aplicación. Actualmente esta división se encuentra formada por:
* **ISO/IEC 25020 - *Measurement reference model and guide*:** presenta una explicación introductoria y un modelo de referencia común a los elementos de medición de la calidad. También proporciona una guía para que los usuarios seleccionen o desarrollen y apliquen medidas propuestas por normas ISO.
* **ISO/IEC 25021 - *Quality measure elements*:** define y especifica un conjunto recomendado de métricas base y derivadas que puedan ser usadas a lo largo de todo el ciclo de vida del desarrollo software.
* **ISO/IEC 25022 - *Measurement of quality in use*:** define específicamente las métricas para realizar la medición de la calidad en uso del producto.
* **ISO/IEC 25023 - *Measurement of system and software product quality*:** define específicamente las métricas para realizar la medición de la calidad de productos y sistemas software.
* **ISO/IEC 25024 - *Measurement of data quality*:** define específicamente las métricas para realizar la medición de la calidad de datos.

### **4.3.3) NORMA IEE 829**

El desarrollo de este plan de pruebas esta basado por la norma IEEE 829-1983, la cual describe un estándar para la documentación de las pruebas de software que se muestran los tipos de documentos que pueden producirse durante un proceso de prueba.

El objetivo de la norma es proporcionar un conjunto estandarizado de pasos a seguir para la documentación de pruebas de software. Existen 8 tipos de pasos que pueden usarse en 3 etapas distintas de las pruebas de software.

El modelo de calidad del producto definido por la IEEE 829 se encuentra compuesto por las ocho características de calidad que se muestran en la siguiente Imagen:

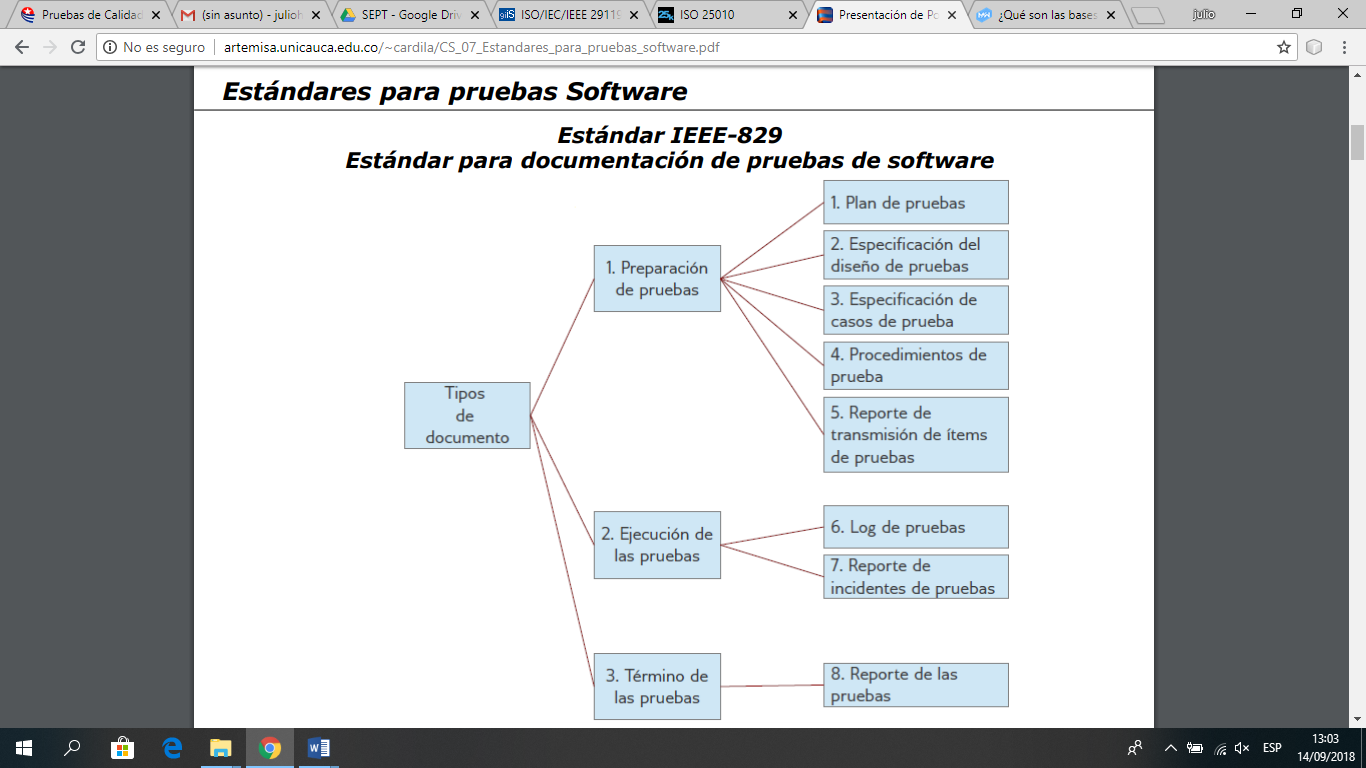


Imagen 1. Fases estándar IEEE 829

## **4.4) MARCO LEGAL**

Todas las actividades propuestas y desarrolladas dentro del proyecto quedan inmersas dentro del acuerdo de cooperación interinstitucional realizado entre la empresa San ambiente y la institución Universitaria Antonio José Camacho, firmado en el mes de abril de 2019.

# **METODOLOGÍA**

En esta parte se observará la metodología a emplear para cumplir con los objetivos postulados anteriormente.

## **5.1) Fuentes y técnicas**

Para la evaluación propuesta del plan de pruebas se tomará en cuenta los acuerdos con el desarrollador y los requerimientos funcionales y no funcionales.

* Conocimiento previo de los requerimientos
* Implementación de las pruebas
* Resultados de evaluación del modulo de comunicación y traspaso de información del sistema de gestión de datos ambientales de la empresa SAN AMBIENTE

## **5.2) Tipos de investigación**

En el presente trabajo se accede a la investigación factible y evaluativa ya que representan la factibilidad de procedimientos requeridos para las actividades de analizar y evaluar los recursos necesarios para desarrollo del proyecto.

* Tipo documental: donde se accederá a fuentes primarias, secundarias las cuales sen encuentran relacionadas con libros, artículos, tesis, documentos oficiales, reportes asociados entre otras fuentes oficiales.
* Tipo aplicativo: será la que nos dará un resultado a los criterios evaluados y propuestos para la aprobación de los atributos de calidad referidos.

## **5.3) Metodología de desarrollo**

En el presente proyecto se irán aplicando los criterios de evaluación de acuerdo con el desarrollo de los módulos y de las solicitudes por los requerimientos de usuarios ya que se irán haciendo las fases de pruebas de acuerdo con la entrega de los módulos desarrollados para esto se tendrán en cuanta las fases de la metodología descriptas en la norma IEEE -829 con los criterios de evaluación de calidad que nos presenta la ISO 25010, dentro de estas metodologías y estándares estipulados.

# **FASES PARA SEGUIR**

## **Programación de pruebas**

* Plan de pruebas
* Especificación del diseño de pruebas
* Especificación de casos de prueba
* Procedimiento de pruebas
* Reporte de transmisión de ítems de pruebas

## **EJECUCION DE PRUEBAS**

* Log de pruebas
* Reporte de incidentes de pruebas

## **TERMINO DE LAS PRUEBAS**

* Reporte de las pruebas
* Cierre

# **TIPOS DE DOCUMENTOS**

## **6.1. PREPARACION DE PRUEBAS**

### **6.1.1 PLAN DE PRUEBAS**

De los siguientes elementos se compone el plan de pruebas:

• Definición del alcance

• Definición de los recursos

• Criterios de calidad

• Definición de ítems a probar

• Tipos de pruebas escogidas para validar los criterios

• Riesgos

Se elabora el documento de desarrollo de pruebas para este proyecto se le dio el nombre de aplicación de un modelo de calidad al módulo de comunicación y traspaso de información del sistema de gestión de datos ambientales de la empresa San Ambiente, en el cual se describen los pasos, el alcance, enfoque y recursos que se tuvieron en cuenta durante la etapa de pruebas de proyecto, identificando las herramientas que se usaran durante la etapa de pruebas, los ítems a probar, que tipo de pruebas se harán basándose en los requerimientos funcionales y cuáles serán los alcances de cada fase de pruebas y sus riesgos.

Las pruebas después de ser ejecutadas deben ser documentada por orden y describiendo cual fue el resultado que se obtuvo (pass/fail) y si se tuvieron inconformidades se describe una observación o se realiza el reporte del incidente causado con la prueba que describe los detalles encontrados cuando la prueba no tuvo el resultado esperado (pass); por ultimo se realiza un reporte de pruebas en el que se describe el rendimiento, conducta, incidentes del aplicativo; la importancia de este último reporte será evaluar la calidad del sistema, con el que se podrán tomar decisiones si la calidad del sistema que será entregado a un usuario final es suficiente para continuar con el proceso de desarrollo del mismo.

#### **DEFINICION DE LOS RECURSOS**

Se nombran algunas herramientas para la realización de pruebas

* Herramientas para pruebas unitarias:

Tabla 1: Tipos de herramientas para la elaboración de pruebas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tabla de tipos de herramientas para elaboración de pruebas | | | | |
| Tipo de prueba | Herramienta | Descripción | Adecuada para este proyecto | Facilidad de uso |
| Pruebas de código | Visual Studio | Identificar e interactuar con objetos gráficos | Si | Media |
| Visual Studio | Interfaz web | Si | Media |
| Jest | Pruebas unitarias | Si | Media |
| JMeter | Prueba de stress | Si | Media |
| Análisis de errores de código | Visual Studio Debugging | Recorrido de código y análisis de errores | Si | Media |

Tabla 4. Herramienta elaboración de pruebas

Autor: Fabian Mina – Daniela Romero

• **PostgreSQL**

Es un gestor de base de datos relacional orientado a objetos y libre, que va dirigido a una comunidad de desarrolladores, no tiene gestor de defectos haciendo muy difícil conocer el estado de sus defectos.

• **Angular Material**

Es un framework para aplicaciones web desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página. Su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC), en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles.

• **Node.js**

Es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación ECMAScript

* **Visual Studio Code**

Es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código.

* **Jest**

Es una plataforma de pruebas universal, la cual se adapta a cualquier marco o biblioteca de JavaScript, tiene como objetivo trabajar sin configuración en la mayoría de proyectos de JavaScript; jest se puede ejecutar en proyectos usando Angular, Babel, Node, Typescript, Read, Vue entre otros.

* **DEFINICION DE ITEMS A PROBAR**

Los ítems para probar se dividen en varios módulos:

• **Interfaz gráfica:** Se tendrá en cuenta cada una de las vistas del aplicativo.

**Pruebas de usabilidad:** Se prueba el ítem mediante encuestas realizadas a diferentes usuarios ajenos al software desarrollado.

**Pruebas Eficiencia:** Se prueba el comportamiento del aplicativo y los tiempos de respuesta a las peticiones realizadas por los usuarios.

• **lógica del aplicativo:**

**Pruebas Unitarias:** Se prueban métodos de clases, por ejemplo: métodos get/set, se realizarán pruebas en las líneas de código validando que el método codificado reciba y entregue valores correctos e incorrectos.

• **base de datos:**

**Pruebas de consultas masivas**: Se realizan pruebas de carga donde se simula la conexión de múltiples usuarios consultando la misma base de datos.

**Pruebas Eficiencia:** Se prueba el comportamiento del aplicativo y los tiempos de respuesta a las peticiones realizadas por los usuarios.

* **Login (Seguridad):** Se probará la validación de los datos, autenticación de los usuarios y encriptación de contraseñas en base de datos, se evitará que los datos personales como contraseñas no se ingresen en texto plano.

#### **TIPOS DE PRUEBAS ESCOGIDAS PARA VALIDAR LOS CRITERIOS DE CALIDAD.**

Con las siguientes pruebas se quiere satisfacer los criterios de calidad que se plantearan

• Pruebas Unitarias

• Pruebas de usabilidad

• Pruebas eficiencia

• Pruebas de seguridad

#### **6.1.1.1 ALCANCE**

Verificar el cumplimiento de las normas y estándares de calidad parametrizados por los requerimientos solicitados en el desarrollo, con el fin de que sea un aplicativo que cumpla su funcionalidad, se manejara la validación concreta de los estándares de calidad con planes de pruebas aplicados a este desarrollo obteniendo pruebas exactas de tolerancias a fallos y de más criterios manejados en las normas de calidad implementadas.

El documento de desarrollo de pruebas se elabora con el nombre de aplicación de modelo calidad para el módulo de comunicación de traspaso de la empresa San Ambiente, y en el se especifica paso a paso las etapas para que lograr hacer una prueba exacta definiendo el alcance, enfoque y recursos utilizados en la etapa de prueba, ítems a probar, herramientas que se usaran, alcance de cada prueba y los riesgos.

#### **CRITERIOS DE CALIDAD**

Para el plan de pruebas de este proyecto se tomarán algunos criterios de calidad según la norma ISO 25010

**Adecuación Funcional:** Representa la capacidad del producto software para proporcionar funciones que satisfacen las necesidades cuando se usa en condiciones especificadas. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

* Completitud funcional. Grado en el cual el conjunto de funcionalidades cubre todas las tareas y los objetivos del usuario especificados.
* Corrección funcional. Capacidad del producto o sistema para proveer resultados correctos con el nivel de precisión requerido.
* Pertinencia funcional. Capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas y objetivos de usuario especificados.

Para la adecuación funcional se realizarán las siguientes pruebas:

* Pruebas Unitarias
* Casos de uso

**Eficiencia:** Asegura que la aplicación logre cumplir los objetivos, pero en un tiempo óptimo y sin retrasos.

Esta característica representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

* Comportamiento temporal. Los tiempos de respuesta y procesamiento.
* Utilización de recursos. Las cantidades y tipos de recursos utilizados cuando el software lleva a cabo su función bajo condiciones determinadas.
* Capacidad. Grado en que los límites máximos de un parámetro de un producto o sistema software cumplen con los requisitos.

Para la eficiencia dentro del software se realizarán las siguientes pruebas:

* Pruebas de carga o consulta masiva
* Tiempos de respuesta del aplicativo

**Usabilidad:** Es la medida de la calidad de la experiencia que tiene un usuario cuando interactúa con un producto o sistema

Capacidad del producto software para ser entendido, aprendido, usado por el usuario bajo determinadas condiciones. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

* Capacidad para reconocer su adecuación. Capacidad del producto que permite al usuario entender si el software es adecuado para sus necesidades.
* Capacidad de aprendizaje. Capacidad del producto que permite al usuario aprender su aplicación.
* Capacidad para ser usado. Capacidad del producto que permite al usuario operarlo y controlarlo con facilidad.
* Protección contra errores de usuario. Capacidad del sistema para proteger a los usuarios de hacer errores.
* Estética de la interfaz de usuario. Capacidad de la interfaz de usuario de agradar y satisfacer la interacción con el usuario.
* Accesibilidad. Capacidad del producto que permite que sea utilizado por usuarios con determinadas características y discapacidades.

Para la elaboración de pruebas de usabilidad se realizaron formatos para usuarios donde se mostrarán los resultados de la revisión y el refinamiento del trabajo realizado

**Seguridad**: Es la medida de la confidencialidad que tienen los archivos del aplicativo

Capacidad de protección de la información y los datos para que personas no autorizados no puedan leerlos o modificarlos. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

* **Confidencialidad.** Protección contra el acceso de datos e información no autorizados.
* **Integridad.** Componente para prevenir accesos o modificaciones no autorizados.
* **No repudio.** Demostrar las acciones o eventos que han tenido lugar, y no puedan ser repudiados posteriormente.
* **Responsabilidad.** Rastrear de forma inequívoca las acciones de una entidad.
* **Autenticidad.** Demostrar la identidad de un sujeto o un recurso.

Para la seguridad dentro del software se realizarán pruebas para validar la autenticación de los usuarios al sistema se realizarán las siguientes pruebas:

* Casos de uso
* Pruebas Unitarias.

#### **RIESGOS**

Los riesgos que se pueden tener en este plan de pruebas son que al ejecutar las pruebas con las herramientas planteadas anteriormente no se esperen los resultados que se tienen planeados, ya que el volumen de la aplicación puede ser un poco grande para las herramientas, y que puede que no sean óptimas para ejecutarse con este tipo de proyectos o que al ser ejecutadas tenga errores en el código y dificulte el proceso de pruebas.

* + 1. **Especificación del diseño de pruebas**

Este plan de pruebas se realiza con la finalidad de elegir los componentes que se van a pasar por un proceso de pruebas, de esta manera aprobar y verificar que los requerimientos funcionales y no funcionales del aplicativo se cumplan de igual forma el documento de pruebas permitirá continuar con el procedo de desarrollo del proyecto.

Al momento de ejecutar el plan de pruebas, es posible recopilar información sobre los errores o fallas que se presenten, buscando asegurar la calidad del proyecto en proceso de desarrollo y su correcto funcionamiento.

Con el plan de pruebas se busca realizar pruebas unitarias con el framework Jest, realizar pruebas de depuración de código y con la aplicación Jmeter se harán pruebas de stress ala base de datos alojada en el gestor PostgreSQL

**6.1.2.1 Lista de requerimientos**

|  |  |
| --- | --- |
| # | Transferencia y comunicación |
| RQ1 | El sistema debe permitir el enlace y transferencia de datos de los dataloger mediante Internet. |
| RQ2 | El sistema debe permitir crear, modificar, borrar, visualizar comandos para ser utilizados por los protocolos de comunicación X y Y. |
| RQ3 | El sistema debe permitir elegir qué tipo de protocolo de comunicación usará una estación, pero en una organización puede haber múltiples estaciones con diferentes protocolos y a su vez, que usen diferentes comandos. |
| RQ4 | El sistema debe permitir establecer los parámetros necesarios para garantizar la comunicación vía FTP como cliente o servidor. |
| RQ5 | El sistema debe permitir crear plantillas para leer los archivos planos y cargar los datos al sistema. |
| RQ6 | El sistema debe permitir usar las plantillas de textos planos de FTP para otras estaciones. |
| RQ7 | El sistema debe permitir acceder al FTP de acuerdo a la base de tiempo de las estaciones, leer los archivos de texto y almacenar la información en la base de datos. |
| RQ8 | El sistema debe permitir establecer los parámetros necesarios para garantizar la comunicación vía Modbus como cliente o servidor. |
| RQ9 | El sistema debe permitir configurar los comandos de módbus para acceder a los datalogers y obtener la información de estos. |
| RQ10 | El sistema debe permitir acceder a las estaciones mediante Modbus de acuerdo a la base de tiempo creado y almacenar la información en la base de datos. |
| RQ11 | El sistema debe permitir usar las plantillas comandos de Modbus para otras estaciones. |
| RQ12 | El sistema debe permitir subir textos planos de manera manual cuando una estación no tenga acceso a Internet |

Tabla 5. Listado Requerimientos

Autor: Fabian Mina – Daniela Romero

* + - 1. **Estrategias**

Para las estrategias de este proyecto se utilizaron y refinaron diagramas de casos de uso, casos de uso.

A partir de los diagramas de casos de uso y casos de uso se presenta a continuación:

**Diagrama de casos de uso**

En el diagrama de caso de uso, se muestra el actor y las actividades que debe realizar en la aplicación desarrollada.

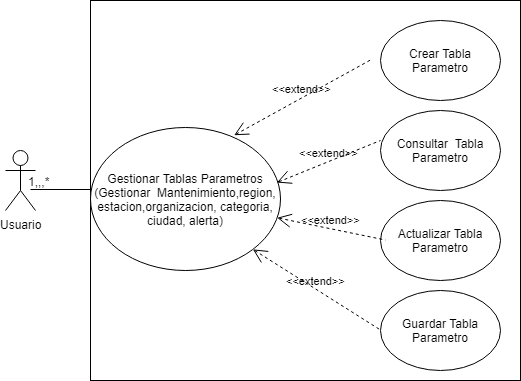


Imagen 2. Diagrama casos de uso

Autor: Fabian Mina -Daniela Romero

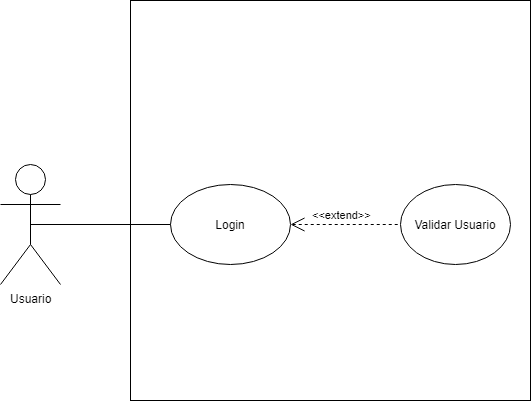


Imagen 3. Diagrama casos de uso validar usuario

Autor: Fabian Mina -Daniela Romero

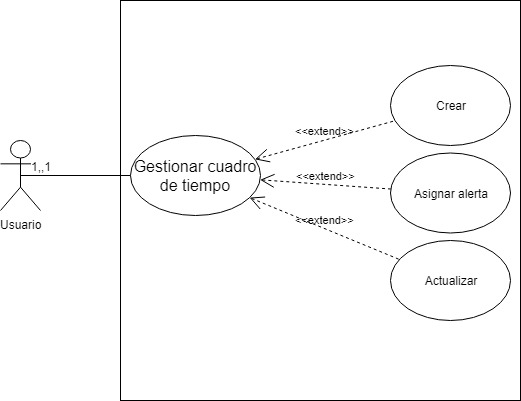


Imagen 4. Diagrama casos de uso cuadro de tiempo

Autor: Fabian Mina -Daniela Romero

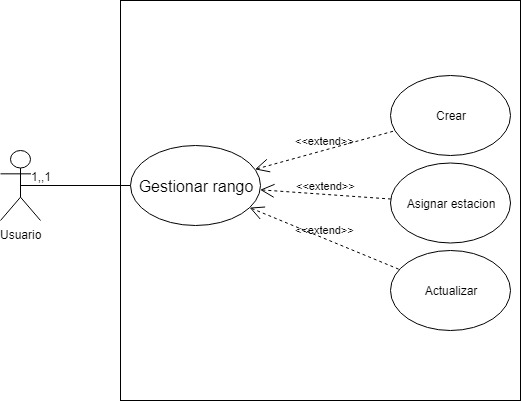


Imagen 5. Diagrama casos de uso gestionar rango

Autor: Fabian Mina -Daniela Romero

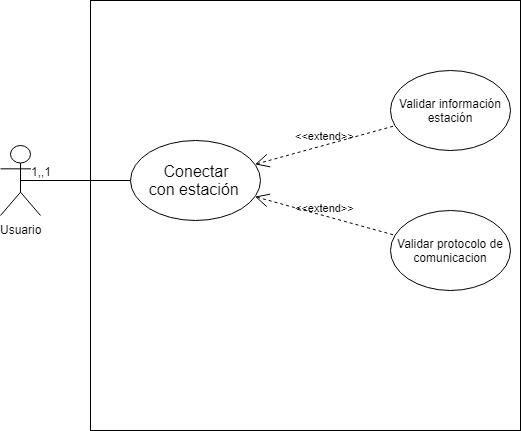


Imagen 6. Diagrama casos de uso gestionar estación

Autor: Fabian Mina -Daniela Romero



Imagen 7. Diagrama General casos de uso

Autor: Fabian Mina -Daniela Romero

**Casos de uso**

En los casos de uso, se muestra detalladamente las actividades o acciones que realiza actor y la respuesta que tiene el sistema para llevar a cabo algún proceso.

**CASOS DE USO 1**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Conectar con estación FTP | | | | | CU1 |
| Actores | Administrador, Operador | | | | | |
| Referencias | Requerimiento RF 1 | | |  | | |
| Precondición | El sistema deberá estar conectado a la red de internet, la estación ya debe estar creada con anterioridad | | | | | |
| Postcondición | El sistema debe mostrar los datos básicos de la estación a la que se ha conectado | | | | | |
| Autor | Fabián Mina | Fecha |  | | Versión |  |
| Aprobación | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Conectarse mediante protocolo FTP a una estación |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal | | | |
| 1 | El usuario selecciona el nombre de la estación. | **2** | El sistema muestra la información almacenada para ese nombre de estación.  -Id  -Serial  -Abreviación nombre corto de la -estación  -organización  -categoría  -región.  -Localización  -ciudad  -latitud y longitud  -elevación  -Base de tiempo |
| 3 | El usuario confirma la información botón conectar | **4** | El sistema se conecta a la estación seleccionada. |
|  |  | **5** | El sistema se conecta a la estación seleccionada y muestra mensaje en pantalla “Conexión Exitosa” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de que los datos no sean válidos o se presente algún error al momento de llenar el formulario, el sistema debe mostrar mensaje en pantalla, especificando el error que se presentó |
| 2 | Si no existe protocolo de comunicación con esta estación, se mostrará mensaje en pantalla “No existe protocolo de comunicación”. |

Tabla 6. Caso de uso 1

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 2**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Conectar con estación Modbus | | | | | CU2 |
| Actores | Administrador, Operador | | | | | |
| Referencias | Requerimiento RF 1 | | |  | | |
| Precondición | El sistema deberá estar conectado a la red de internet, la estación ya debe estar creada con anterioridad | | | | | |
| Postcondición | El sistema debe mostrar los datos básicos de la estación a la que se ha conectado | | | | | |
| Autor | Fabián Mina | Fecha |  | | Versión |  |
| Aprobación | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Conectarse mediante protocolo Modbus a una estación |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal | | | |
| 1 | El usuario selecciona el nombre de la estación. | **2** | El sistema muestra la información almacenada para ese nombre de estación.  -Id  -Serial  -Abreviación nombre corto de la -estación  -organización  -categoría  -región.  -Localización  -ciudad  -latitud y longitud  -elevación  -Base de tiempo |
| 3 | El usuario confirma la información botón conectar | **4** | El sistema se conecta a la estación seleccionada. |
|  |  | **5** | El sistema se conecta a la estación seleccionada y muestra mensaje en pantalla “Conexión Exitosa” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de que los datos no sean válidos o se presente algún error al momento de llenar el formulario, el sistema debe mostrar mensaje en pantalla, especificando el error que se presentó |
| 2 | Si no existe protocolo de comunicación con esta estación, se mostrará mensaje en pantalla “No existe protocolo de comunicación”. |

Tabla 7. Caso de uso 2

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 3**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear Validación | | | | | | CU3 |
| Actores | Administrador, Operador | | | | | | |
| Referencias | Requerimiento RF 1 | | | |  | | |
| Precondición | El sistema deberá tener almacenada información del RF 1 para realizar calibración de datos. | | | | | | |
| Postcondición | En la pantalla de selección quedan los datos con los cuales se realiza el ajuste calibración. | | | | | | |
| Autor | Fabián Mina | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Gestionar estación |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal | | | |
| 1 | El usuario Selecciona la hora, minutos  (GMT) | **2** | El sistema valida el GMT por estación y corrige la hora |
| 3 | El usuario Digita el campo (Factor) X |  |  |
| 4 | El usuario Digita el valor en campo (Factor) +b | **5** | El sistema valida de manera automática los campos con la formula Y= mx+b |
| 6 | El usuario presiona botón calibrar. Para realizar calibración de los datos manualmente | **7** | El sistema realiza validación de los datos digitados |
| 8 | El usuario guardar la información diligenciada | **9** | El sistema guarda la información, Muestra mensaje en pantalla. “Información Guardada con éxito” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | El sistema intenta escribir la hora y los minutos: El campo está inhabilitado para escritura y solo permite la selección. |
| 2 | El usuario digita un valor alfanumérico: El campo solo permite valores numéricos por lo tanto no registra ningún valor diferente |
| 3 | En caso de que el usuario no calibre los datos, el sistema lo hará automáticamente. |
| 4 | En caso de que los datos no sean válidos o se presente algún problema al momento de llenar el formulario, el sistema debe mostrar mensaje en pantalla, especificando el error presentado. |

Tabla 8. Caso de uso 3

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 4**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear región (Tabla parámetro) | | | | | | CU4 |
| Actores | Administrador, Operador | | | | | | |
| Referencias | Requerimiento RF 3 | | | |  | | |
| Precondición | El sistema deberá estar abierto en el menú de parametrización | | | | | | |
| Postcondición | La región queda creada en el sistema. | | | | | | |
| Autor | Fabián Mina | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Gestionar las regiones |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 1  Creación de la región | | | |
| 1 | El usuario oprime el botón nueva región | **2** | El sistema carga el Id automático que corresponde a la nueva región y habilita los campos de la región. |
| 3 | El usuario digita el nombre de la región con una longitud máxima de 50 caracteres. |  |  |
| 4 | El usuario digita una descripción de la región con una longitud máxima de 50 caracteres. |  |  |
| 5 | El usuario oprime el botón guardar | **6** | El sistema valida que todos los campos estén llenos y guarda.  Muestra mensaje de confirmación de transacción: “Región creada correctamente” |
|  |  | **7** | El sistema muestra en el listado la nueva región creada. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |
| 2 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Actualización de la región | | | |
| 1 | El usuario digita el nombre de la región o el Id de la región. | **2** | El sistema carga los datos de la región. |
| 3 | El usuario oprime el botón editar. | **4** | El usuario habilita los campos nombre y descripción para actualización |
| 4 | El usuario actualiza el campo nombre y descripción y oprime botón guardar | **6** | El sistema valida que todos los campos estén llenos, guarda y muestra mensaje de confirmación “Guardado exitosamente”. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje y el cursor se ubica en el campo. |

Tabla 9. Caso de uso 4

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 5**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear estación (Tabla parámetro) | | | | | | CU5 |
| Actores | Administrador, Operador | | | | | | |
| Referencias | Requerimiento RF 1 y 2 | | | |  | | |
| Precondición | El sistema deberá estar abierto en el menú de parametrización, los datos GMT deberán estar cargados en base de datos. | | | | | | |
| Postcondición | La estación queda creada en el sistema. | | | | | | |
| Autor | Fabián Mina | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Gestionar las estaciones |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 1  Creación de la estación | | | |
| 1 | El usuario oprime el botón nueva estación. | **2** | El sistema carga el Id automático que corresponderá a la nueva estación y habilita los campos de la estación. |
| 3 | El usuario digita el nombre de la estación. |  |  |
| 4 | El usuario digita serial de la estación |  |  |
| 5 | El usuario digita un nombre corto para la estación |  |  |
| 6 | El usuario selecciona GMT de la estación (Zona horaria) | **7** | El sistema despliega la lista de GMT |
| 10 | El usuario selecciona categoría | **11** | El sistema carga la lista de categorías almacenadas |
| 12 | El usuario selecciona región | **13** | El sistema carga la lista de regiones almacenadas |
| 14 | El usuario selecciona protocolo de comunicación (TCP o modbus) |  |  |
| 15 | El usuario selecciona el campo localización | **16** | El sistema carga la lista de localizaciones almacenadas |
| 17 | El usuario Selecciona el campo ciudad | **18** | El sistema carga la lista de ciudades almacenadas |
| 19 | El usuario diligencia los campos latitud y longitud.  El usuario digita estos campos en grados decimales |  |  |
| 20 | El usuario diligencia el campo elevación  El usuario digita estos campos en grados decimales |  |  |
| 21 | El usuario ingresa un número entero entre 0 y 120 para la base de tiempo.  (base de tiempo entre 0 y 2 horas) | **22** | El sistema valida que una estación no tenga asignada la misma base de tiempo. |
| 23 | El usuario digita una descripción de la estación. |  |  |
| 24 | El usuario oprime el botón guardar | **25** | El sistema valida que todos los campos estén llenos y guarda.  Muestra mensaje de confirmación.  “Estación creada correctamente” |
|  |  | **26** | El sistema muestra en el listado la nueva estación creada. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |
| 2 | El usuario no selecciona ningún valor de la lista. El sistema por defecto debe colocar el valor “Sin definir” y se debe tomar como un valor vacío al intentar guardar. |
| 3 | El usuario digita valores alfanuméricos Se muestra el mensaje “Los valores deben ser decimales” |
| 4 | La estación ya existe con la misma base de tiempo muestra mensaje “Ya existe esta base de tiempo en la estación” |
| 5 | Hay campos vacíos, el sistema muestra el mensaje “Hay campos vacíos”, selecciona el campo en rojo y el cursor se ubica en el campo. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Actualización de la estación | | | |
| 1 | El usuario digita el nombre de la estación o el Id de la región. | **2** | El sistema carga los datos de la estación. |
| 3 | El usuario oprime el botón editar. | **4** | El sistema habilita los campos nombre, serial, nombre corto, GMT, descripción, región.  Localización  Ciudad, latitud y longitud, elevación para actualización |
| 5 | El usuario actualiza los campos nombre, serial, nombre corto, GMT, descripción, región.  Localización  Ciudad, latitud y longitud y oprime botón guardar | **6** | El sistema valida que todos los campos estén llenos, guarda y muestra mensaje de confirmación.  “Guardado con Satisfactoriamente” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |
| 6 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |

Tabla 10. Caso de uso 5

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 6**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear organización (Tabla parámetro) | | | | | | CU6 |
| Actores | Administrador | | | | | | |
| Referencias | Requerimiento RF 1 y 2 | | | |  | | |
| Precondición | El sistema deberá estar abierto en el menú de parametrización | | | | | | |
| Postcondición | La organización quedara creada en el sistema. | | | | | | |
| Autor | Fabián Mina | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Gestionar las organizaciones del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 1  Creación de la organización. | | | |
| 1 | El usuario oprime el botón nueva organización | **2** | El sistema carga el Id automático que corresponde a la nueva organización y habilita los campos de la organización. |
| 3 | El usuario digita el nombre de la organización. Longitud máxima de 50 caracteres. |  |  |
| 4 | El usuario digita el Email. Longitud máxima de 50 caracteres. |  |  |
| 5 | El usuario digita número de contacto. Longitud máxima de 50 caracteres. |  |  |
| 6 | El usuario digita una descripción para la organización. Longitud máxima de 50 caracteres. |  |  |
| 7 | El usuario oprime el botón guardar | **8** | El sistema valida que todos los campos estén llenos y guarda. Muestra mensaje de confirmación.  “Organización creada correctamente” |
|  |  | **9** | El sistema muestra en el listado la nueva organización creada. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |
| 2 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Actualización de la organización | | | |
| 1 | El usuario digita el nombre de la organización o el Id de la organización. | **2** | El sistema carga los datos de la organización. |
| 3 | El usuario oprime el botón editar. | **4** | El sistema habilita los campos nombre de la organización, email, numero contacto y descripción para actualización. |
| 5 | El usuario actualiza el campo nombre de la organización, email, numero de contacto, descripción y oprime botón guardar | **6** | El sistema valida que todos los campos estén llenos, guarda y muestra mensaje de confirmación: “Organización actualizada correctamente” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |
| 2 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |

Tabla 11. Caso de uso 6

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 7**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear categoría (Tabla parámetro) | | | | | | CU7 |
| Actores | Administrador | | | | | | |
| Referencias | Requerimiento RF 1 y 2 | | | |  | | |
| Precondición | El sistema deberá estar abierto en el menú de parametrización | | | | | | |
| Postcondición | La categoría quedara creada en el sistema. | | | | | | |
| Autor | Fabián Mina | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Gestionar las categorías del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 1  Creación de la categoría. | | | |
| 1 | El usuario oprime el botón nueva categoría | **2** | El sistema carga el Id automático que corresponde a la nueva categoría y habilita los campos de la categoría. |
| 3 | El usuario digita el nombre de la categoría. |  |  |
| 4 | El usuario digita una descripción para la categoría. |  |  |
| 5 | El usuario oprime el botón guardar | **6** | El sistema valida que todos los campos estén llenos y guarda.  Muestra mensaje de confirmación de transacción  “Categoría creada correctamente” |
|  |  | **7** | El sistema muestra en el listado la nueva categoría creada. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |
| 2 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Actualización de la categoría | | | |
| 1 | El usuario digita el nombre de la categoría o el Id de la organización. | **2** | El sistema carga los datos de la categoría. |
| 3 | El usuario oprime el botón editar. | **4** | El sistema habilita los campos nombre de la categoría y descripción para actualización. |
| 5 | El usuario actualiza el campo nombre de la categoría, descripción y oprime botón guardar | **6** | El sistema valida que todos los campos estén llenos, guarda y muestra mensaje de confirmación. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 6 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |
| 5 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |

Tabla 12. Caso de uso 7

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 8**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear Ciudad (Tabla parámetro) | | | | | | CU8 |
| Actores | Administrador | | | | | | |
| Referencias | Requerimiento RF 1 y 2 | | | |  | | |
| Precondición | El sistema deberá estar abierto en el menú de parametrización, por cada región, las ciudades deberán corresponder. Se mostrará combo box para seleccionar región a la que pertenece la ciudad. | | | | | | |
| Postcondición | La ciudad quedara creada en el sistema. | | | | | | |
| Autor | Daniela Romero | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Gestionar las ciudades del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 1  Creación de la Ciudad. | | | |
| 1 | El usuario oprime el botón nueva Ciudad | **2** | El sistema carga el Id automático que corresponde a la nueva Ciudad y habilita los campos de la Ciudad. |
| 3 | El usuario digita el nombre de la Ciudad. |  |  |
| 4 | El usuario selecciona la región a la que pertenece la ciudad | **5** | El sistema despliega la lista de regiones almacenadas en base de datos. |
| 6 | El usuario digita una descripción para la Ciudad. |  |  |
| 7 | El usuario oprime el botón guardar | **8** | El sistema valida que todos los campos estén llenos y guarda.  Muestra mensaje de confirmación. |
|  |  | **9** | El sistema muestra en el listado la nueva ciudad creada. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |
| 2 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Actualización de la Ciudad | | | |
| 1 | El usuario digita el nombre de la ciudad o el Id de la ciudad. | **2** | El sistema carga los datos de la ciudad. |
| 3 | El usuario oprime el botón editar. | **4** | El sistema habilita los campos nombre de la ciudad, región y descripción para actualización. |
| 5 | El usuario actualiza el campo nombre de la Ciudad, región, descripción y oprime botón guardar | **6** | El sistema valida que todos los campos estén llenos, guarda y muestra mensaje de confirmación. “Ciudad actualizada correctamente” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |
| 2 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |

Tabla 13. Caso de uso 8

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 9**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear rango (Tabla parámetro) | | | | | | CU9 |
| Actores | Administrador | | | | | | |
| Referencias | Requerimiento RF 1 y 2 | | | |  | | |
| Precondición | El sistema deberá estar abierto en el menú de parametrización, por cada estación, los rangos deberán corresponder. Se mostrará combo box para seleccionar estación a la que pertenece el rango. | | | | | | |
| Postcondición | Los rangos quedaran creados en el sistema. | | | | | | |
| Autor | Daniela Romero | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Gestionar los rangos del sistema, los cuales definen el porcentaje de variación al gestionar la información. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 1  Creación de rango. | | | |
| 1 | El usuario oprime el botón nuevo Rango | **2** | El sistema carga el Id automático que corresponde al nuevo rango y habilita los campos. Rango mínimo, rango máximo, rango, descripción. |
| 3 | El usuario digita un nombre para el rango. Longitud máxima de 50 caracteres. |  |  |
| 4 | El usuario digita valor mínimo del nuevo rango.  (Valor numérico) valor mínimo 0 |  |  |
| 5 | El usuario digita valor máximo del nuevo rango.  (Valor numérico) |  |  |
| 6 | El usuario selecciona la estación a la que pertenece el rango | **7** | El sistema despliega la lista de estaciones almacenadas en base de datos. |
| 8 | El usuario digita una descripción para el rango.  Longitud máxima de 50 caracteres. |  |  |
| 9 | El usuario oprime el botón guardar | **10** | El sistema valida que todos los campos estén llenos y guarda.  Muestra mensaje de confirmación de transacción “Guardado exitosamente” |
|  |  | **11** | El rango guardado se muestra en la cuadrícula donde están listados todos los rangos |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |
| 2 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Actualización del rango | | | |
| 1 | El usuario digita el nombre del rango o el Id. | **2** | El sistema carga los datos del rango. |
| 3 | El usuario oprime el botón editar. | **4** | El sistema habilita los campos nombre del rango, rango, región y descripción para actualización. |
| 5 | El usuario actualiza el campo nombre del rango, rango, región, descripción y oprime botón guardar  Longitud máxima de 50 caracteres. | **6** | El sistema valida que todos los campos estén llenos, guarda y muestra mensaje de confirmación. “Rango actualizado correctamente” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |
| 2 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |

Tabla 14. Caso de uso 9

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 10**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear alerta (Tabla parámetro) | | | | | | CU10 |
| Actores | Administrador | | | | | | |
| Referencias | Requerimiento RF 1 y 2 | | | |  | | |
| Precondición | El sistema deberá estar abierto en el menú de parametrización, las alertas se crearán igual para todas las regiones. | | | | | | |
| Postcondición | Las alertas quedaran creadas en el sistema, enviaran notificación automáticamente. | | | | | | |
| Autor | Daniela Romero | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | | |

|  |
| --- |
| Propósito |
| Gestionar las alertas del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 1  Creación alerta. | | | |
| 1 | El usuario oprime el botón nuevo alerta. | **2** | El sistema carga el Id automático que corresponde a la nueva alerta y habilita los campos. |
| 3 | El usuario digita un nombre para la alerta. Longitud máxima de 50 caracteres. |  |  |
| 4 | El usuario digita una descripción para la alerta creada. Longitud máxima de 50 caracteres. |  |  |
| 5 | El usuario selecciona tipo de alerta.  (Fax - Email – SMS ) |  |  |
| 6 | El usuario oprime el botón guardar | **7** | El sistema valida que todos los campos estén llenos y guarda.  Muestra mensaje de confirmación de transacción.  “Alerta guardada con éxito” |
|  |  | **8** | Se muestra en la cuadrícula donde están listados todas las alertas creadas. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Actualización de alerta | | | |
| 1 | El usuario digita el nombre o el Id de la alerta | **2** | El sistema carga los datos. |
| 3 | El usuario oprime el botón editar. | **4** | El sistema habilita los campos nombre de alerta, tipo y descripción para actualización. |
| 5 | El usuario actualiza el campo nombre de alerta, estación, tipo y oprime botón guardar | **6** | El sistema valida que todos los campos estén llenos, guarda y muestra mensaje de confirmación. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |

Tabla 15. Caso de uso 10

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 11**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear base de tiempo (Tabla parámetro) | | | | | | CU11 |
| Actores | Administrador | | | | | | |
| Referencias | Requerimiento RF 1 y 2 | | | |  | | |
| Precondición | El sistema deberá estar abierto en el menú de parametrización. | | | | | | |
| Postcondición | Los tiempos quedaran creados en el sistema | | | | | | |
| Autor | Daniela Romero | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Gestionar los tiempos para validar los datos del sistema, en el cuadro de tiempo selecciona el rango de tiempo para el cual se van a consultar los registros gestionados en el sistema. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 1  Creación tiempos. | | | |
| 1 | El usuario oprime el botón nuevo tiempo | **2** | El sistema carga el Id automático que corresponde a los tiempos y habilita los campos. |
| 3 | El usuario digita un nombre para los tiempos.  Longitud máxima de 50 caracteres. |  |  |
| 4 | El usuario digita una escala de tiempos a partir de 1 minuto.  La escala de tiempo ira de 1 y finaliza en 240 (ej.1 2 5 6 10 15 30 60 120 125 130 150 170 190 200 210 215 220 225 230 240). | **5** | El sistema valida que el tiempo se configure en formato minutos. |
|  |  | **6** | El sistema validad que el rango de tiempo no supere los 1440 minutos. |
| 7 | El usuario digita una descripción para los tiempos registrados  Longitud máxima de 50 caracteres. |  |  |
| 8 | El usuario selecciona la alerta deseada, donde recibirá notificación | **9** | El sistema despliega las alertas creadas y almacenadas en base de datos.  Fax  E-mail  Mensaje  Ventana Emergente |
| 10 | El usuario oprime el botón guardar | **11** | El sistema valida la información guardad, muestra mensaje en pantalla guardado con éxito  . |
|  |  | **12** | Se muestra en la cuadrícula donde están listados todos los tiempos creadas. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de presentarse un error el sistema mostrara mensaje en pantalla indicando el error. |
| 2 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |
| 1 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Actualización de tiempos | | | |
| 1 | El usuario digita el nombre o el Id | **2** | El sistema carga los datos. |
| 3 | El usuario oprime el botón editar. | **4** | El sistema habilita los campos nombre y rango de tiempos. |
| 5 | El usuario actualiza el campo nombre, rango de tiempos y oprime botón guardar.  Longitud máxima de 50 caracteres. | **6** | El sistema valida que todos los campos estén llenos, guarda y muestra mensaje de confirmación. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |
| 1 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |

Tabla 16. Caso de uso 11

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 12**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Validar Usuario (login) | | | | | | CU12 |
| Actores | Administrador Operador Visualizador editor final editor preeeliminar | | | | | | |
| Referencias | Requerimiento RF 1 y 2 | | | |  | | |
| Precondición | Los usuarios para realizar login deberán estar almacenados en base de datos. | | | | | | |
| Postcondición | Los tiempos quedaran creados en el sistema | | | | | | |
| Autor | Daniela Romero | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Iniciar sesión |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 1  Iniciar sesión. | | | |
| 1 | El usuario digita usuario o correo electrónico.  Longitud máxima de 50 caracteres. | **2** | El sistema valida formato correo electrónico (ejemplo@ejemplo.com) |
| 3 | El usuario digita contraseña Longitud máxima de 20 caracteres mínimo 7 caracteres, debe contener mínimo 1 mayúscula y 1 carácter numérico. |  |  |
| 4 | El usuario oprime la opción Ingresar | **5** | El sistema valida que la información ingresada sea correcta. |
|  |  | **6** | El sistema ingresa al aplicativo |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |
| 2 | El usuario ingresa contraseña diferente a las reglas (1mayuscula 1numero, min 7 dígitos. |
| 3 | En caso de que el usuario no se encuentre registrado, se deberá mostrar mensaje en pantalla “El usuario no existe” |
| 4 | Se digita contraseña o usuario no valido, el sistema muestra mensaje “Credenciales no validas” |
| 5 | Hay campos en blanco. El sistema no ingresa al aplicativo y muestra mensaje de validación. “Debe llenar los campos obligatorios” |

Tabla 17. Caso de uso 12

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 13**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Modificar datos descargados | | | | | | CU13 |
| Actores | Administrador | | | | | | |
| Referencias | Requerimiento RF 1 y 2 | | | |  | | |
| Precondición | -Los datos se encuentren habilitados para ser modificados.  -El editor preliminar y editor final solo podrán modificar los datos.  - si el editor final bloquea los datos, el editor preliminar no podrá modificar los datos. | | | | | | |
| Postcondición | Los tiempos quedaran creados en el sistema | | | | | | |
| Autor | Daniela Romero | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Modificar datos procesados |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 1  Editor Preliminar. | | | |
| 1 | El editor preliminar ingresa a la plantilla creada. |  |  |
| 2 | El editor preliminar selecciona la estación para modificar datos. | **3** | El sistema carga los datos de la estación seleccionada. |
| 4 | El editor preliminar selecciona los datos de la estación que serán modificados. | **5** |  |
| 5 | El editor preliminar modifica los datos procesados. |  |  |
| 7 | El editor final guarda los datos modificados | **8** | El sistema muestra mensaje en pantalla “Guardado con éxito” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de presentarse un error el sistema mostrara mensaje en pantalla indicando el error. |
| 2 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |
| 3 | El editor final bloquea los datos, no podrán ser modificados |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Editor Final | | | |
| 1 | El editor final ingresa a la plantilla creada. |  | . |
| 2 | El editor final selecciona la estación para modificar datos. | **3** | El sistema carga los datos de la estación seleccionada. |
| 5 | El editor final selecciona los datos de la estación que serán modificados. | **6** | El sistema muestra los datos modificados por el usuario preliminar |
| 6 | El editor final modifica los datos procesados. |  |  |
| 7 | El editor final guarda los datos procesados. | **8** | El sistema muestra mensaje en pantalla, “Guardado con éxito” |
| 9 | El editor final bloquea los datos procesados | **10** | El sistema inhabilita los datos para ser modificados por el usuario preliminar.  Los datos no podrán ser modificados por ningún usuario incluyendo el usuario final. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de error el sistema mostrara mensaje en pantalla “Error al guardar los datos” |
| 2 | El usuario intente modificar los datos, el sistema muestra mensaje en pantalla “los datos han sido bloqueados" |

Tabla 18. Caso de uso 13

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 14**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Dar de baja datos procesados | | | | | | CU14 |
| Actores | Usuario Preliminar | | | | | | |
| Referencias | Requerimiento RF 1 y 2 | | | |  | | |
| Precondición | Los datos deberán estar bloqueados | | | | | | |
| Postcondición | Los datos no serán eliminados, quedarán inhabilitados | | | | | | |
| Autor | Fabian Mina | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Inhabilitar datos |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal | | | |
| 1 | El editor final ingresa a la plantilla creada. |  |  |
| 2 | El editor final selecciona la estación para modificar datos. | **3** | El sistema carga los datos de la estación seleccionada. |
| 4 | El editor preliminar selecciona los datos de la estación que serán bloqueados | **5** |  |
| 5 | El editor final bloquea los datos. | **6** | El sistema muestra mensaje de advertencia. “Desea bloquear los datos, estos no se podrán volver a modificar” |
| 7 | El editor final confirma el bloqueo de datos | **8** | El sistema guarda la fecha y usuario de realización de bloqueo de datos. |
|  |  | **9** | El sistema muestra mensaje en pantalla “Bloqueado con éxito” |
|  |  | **10** | Los datos bloqueados se guardan en base de datos |

Tabla 19. Caso de uso 14

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 15**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Conectar Automáticamente con estación | | | | | CU15 |
| Actores | Administrador, Operador | | | | | |
| Referencias | Requerimiento RF 1 | | |  | | |
| Precondición | -El sistema deberá estar conectado a la red de internet  -la estación ya debe estar creada con anterioridad,  -La estación deberá tener configurado el protocolo de comunicación con el cual se hará la conexión. | | | | | |
| Postcondición | El sistema debe mostrar los datos básicos de la estación a la que se ha conectado | | | | | |
| Autor | Fabián Mina | Fecha |  | | Versión |  |
| Aprobación | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Conectarse Automáticamente a una mediante protocolo FTP o Modbus |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal | | | |
| 1 | El usuario selecciona el nombre de la estación. | **2** | El sistema muestra la información almacenada para ese nombre de estación.  -Id  -Serial  -Abreviación nombre corto de la -estación  -organización  -categoría  -región.  -Localización  -ciudad  -latitud y longitud  -elevación  -Base de tiempo |
|  |  | **3** | El sistema verifica el protocolo de comunicación configurado a la estación |
|  |  | **4** | El sistema se conecta a la estación seleccionada. |
|  |  | **5** | El sistema se conecta a la estación seleccionada y muestra mensaje en pantalla “Conexión Exitosa con al estación ” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 2 | En caso de que los datos no sean válidos o se presente algún error al momento de llenar el formulario, el sistema debe mostrar mensaje en pantalla, especificando el error que se presentó |
| 3 | Si no existe protocolo de comunicación con esta estación, se mostrará mensaje en pantalla “No existe protocolo de comunicación”. |

Tabla 20. Caso de uso 15

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 16**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear Mantenimiento | | | | | | CU16 |
| Actores | Administrador, Operador | | | | | | |
| Referencias | Requerimiento RF 1 y 2 | | | |  | | |
| Precondición | El registro de mantenimiento no podrá ser modificado | | | | | | |
| Postcondición | El registro de mantenimiento creado quedara almacenado en base de datos con la respectiva hora y usuario de creación. | | | | | | |
| Autor | Daniela Romero | | Fecha |  | | Versión |  |
| Aprobación | | Aprobado/No aprobado | | | | | |

|  |
| --- |
| Propósito |
| Gestionar el registro de actividad |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 1  Crear mantenimiento. | | | |
| 1 | El usuario oprime el botón nuevo registro | **2** | El sistema carga el Id automático que corresponde al nuevo registro. |
| 3 | El usuario selecciona el nombre de la estación, en el campo registrar estación. |  |  |
| 4 | El usuario digita el nombre del funcionario, en el campo registrar funcionario. |  |  |
| 5 | El usuario digita una descripción o novedad, en el campo novedad |  |  |
| 6 | El usuario oprime el botón guardar | **7** | El sistema valida que todos los campos estén llenos y guarda.  Muestra mensaje de confirmación “Registro guardado con éxito”. |
|  |  | **8** | El sistema carga la hora y fecha de creación automáticamente |
|  |  | **9** | El nuevo registro se almacena en base de datos, y se muestra listado de registros en grilla. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje, selecciona el campo en rojo y el cursor se ubica en el campo. |
| 2 | El usuario intenta escribir más de 50 caracteres: el campo no permite escribir más de 50 caracteres. |

Tabla 21. Caso de uso 16

Autor: Fabian Mina – Daniela Romero

**Casos de uso Procesamiento**

**CASOS DE USO 17**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Criterios de validación | | | | | CU17 |
| Actores | Administrador | | | | | |
| Referencias | Requerimiento RF 14 | | |  | | |
| Precondición | El usuario deberá estar en el módulo procesamiento – Gestión de plantillas de procesamiento. | | | | | |
| Postcondición | Los criterios deben quedar configurados por cada elemento seleccionado. | | | | | |
| Autor |  | Fecha |  | | Versión |  |

|  |
| --- |
| Propósito |
| Creación de criterios |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal Escenario 1  Procesar datos | | | |
| 1 | El usuario ingresa a la pantalla de transferencias |  |  |
| 2 | El usuario selecciona la variable que desea consultar | **3** | El sistema carga la lista de variables almacenadas |
| 4 | El usuario selecciona la fecha y hora que desea consultar |  |  |
| 5 | El usuario ingresa la cantidad de datos a consultar |  |  |
| 6 | El usuario oprime consultar | **7** | El sistema valida la cantidad de transferencias realizadas. |
|  |  | **8** | El sistema carga la cantidad de transferencias realizadas. |
| 9 | El usuario ingresa el porcentaje con el cual se realizará la operación. |  |  |
| 10 | El usuario oprime la opción procesar | **11** | El sistema realiza la operación interna |
|  |  | **12** | El sistema muestra mensaje en pantalla al realizar este proceso “Procesando información” |
|  |  | **13** | El sistema carga la información en pantalla. |
| 14 | El usuario guarda esta información | **15** | El sistema almacena la información cargada en base de datos. |
|  |  | **16** | El sistema muestra mensaje en pantalla “Guardado con éxito” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de presentarse error al momento de realizar operación, se deberá mostrar mensaje en pantalla, la pantalla se recargará para ingresar los datos nuevamente. |
| 2 | En caso de que los datos no sean válidos o se presente algún problema se deberá mostrar en pantalla un mensaje de validación. “Error al guardar información”. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal Escenario 2  Datos Inválidos transferencias realizadas | | | |
|  |  | **1** | El sistema realiza validación de datos |
|  |  | **2** | El sistema muestra mensaje en pantalla con los datos inválidos |
| 3 | El usuario presiona la opción ver | **4** | El sistema carga y muestra la información con los datos inválidos encontrados al momento de procesar la información.  (Topes altos, Topes bajos, Problemas de comunicación, recepción de datos). |
| 5 | El usuario presiona la opción guardar | **6** | El sistema guarda la información, muestra mensaje en pantalla guardado con éxito. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de presentarse error al momento de realizar operación, se deberá mostrar mensaje de validación en pantalla. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal Escenario 3  Corregir datos Inválidos transferencias realizadas | | | |
| 1 | El usuario presiona la opción corregir | **2** | El sistema habilita los campos con los datos inválidos |
| 3 | El usuario modifica los campos inválidos | **2** | El sistema muestra mensaje en pantalla con los datos inválidos |
| 3 | El usuario presiona la opción guardar | **4** | El sistema guarda la información digitada  . |
|  |  | **5** | El sistema guarda la información, muestra mensaje en pantalla guardado con éxito. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de presentarse error al momento de realizar operación, se deberá mostrar mensaje de validación en pantalla. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal Escenario 4  Invalidar datos Inválidos transferencias realizadas | | | |
| 1 | El usuario presiona la opción invalidar | **2** | El sistema muestra mensaje en pantalla “¿desea invalidar los datos?” |
| 3 | El usuario selecciona la opción si | **4** | El sistema muestra mensaje en pantalla “Los datos fueron inhabilitados” |
|  |  | **5** | El sistema recarga la pagina para realizar procesamiento de datos nuevamente |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de presentarse error al momento de realizar operación, se deberá mostrar mensaje de validación en pantalla. |

Tabla 22. Caso de uso 17

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 18**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear plantillas | | | | | CU18 |
| Actores | Administrador | | | | | |
| Referencias | Requerimiento RF 17 | | |  | | |
| Precondición | El sistema selecciona la estación, para la cual se creara la plantilla. | | | | | |
| Postcondición | La plantilla creada deberá visualizarse en el listado de plantillas de configuración. | | | | | |
| Autor |  | Fecha |  | | Versión |  |

|  |
| --- |
| Propósito |
| Creación de plantilla de procesamiento |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal | | | |
| 1 | El usuario selecciona variable | **2** | El sistema despliega la lista de elementos almacenados en base de datos. |
| 3 | El usuario selecciona rango máximo para elemento. |  |  |
| 4 | El usuario selecciona rango mínimo para elemento. |  |  |
| 5 | El usuario selecciona valor pico |  |  |
| 6 | El usuario selecciona valor constante para el elemento. |  |  |
| 7 | El usuario guarda la plantilla | **8** | El sistema muestra mensaje de validación. “Plantilla creada con éxito” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | SI hay un campo sin diligenciar, el sistema mostrara mensaje de validación y seleccionara el campo en rojo“ |

Tabla 23. Caso de uso 18

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 19**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Ver Plantillas | | | | | CU19 |
| Actores | Administrador | | | | | |
| Referencias | Requerimiento RF 17 | | |  | | |
| Precondición | El usuario deberá estar en el módulo procesamiento – Gestión de plantillas de procesamiento. | | | | | |
| Postcondición | Los datos de reglas y criterios deben quedar configurados por cada elemento seleccionado. | | | | | |
| Autor |  | Fecha |  | | Versión |  |

|  |
| --- |
| Propósito |
| Creación de reglas o criterios |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal | | | |
| 1 | El usuario selecciona el elemento a configurar | **2** |  |
| 3 | El usuario despliega plantillas creadas | **4** | El sistema desplegara todas las plantillas creadas en un campo select |
| 5 | El usuario modifica la plantilla seleccionada y guarda los datos. | **6** | El sistema muestra los datos de la plantilla a modificar |
|  | El usuario inhabilita plantilla |  | El sistema muestra mensaje que la plantilla seleccionada fue inhabilitada. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de que los datos no sean válidos o se presente algún problema al realizar el crud se deberá mostrar en pantalla un mensaje de alerta. |

Tabla 24. Caso de uso 19

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 20**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear usuario | | | | | CU20 |
| Actores | Administrador | | | | | |
| Referencias | Requerimiento RF 17 | | |  | | |
| Precondición | El usuario deberá estar en el módulo procesamiento – Gestión de usuarios de procesamiento,  La organización ya debe estar creada. | | | | | |
| Postcondición | Los usuarios quedaran atados a los roles configurados y asignados | | | | | |
| Autor |  | Fecha |  | | Versión |  |

|  |
| --- |
| Propósito |
| Creación de reglas o criterios |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal | | | |
| 1 | El usuario oprime botón nuevo o agregar | **2** | El sistema carga el Id automático que corresponde al nuevo usuario |
| 3 | El usuario digita documento de identidad |  |  |
| 4 | El usuario digita nombres del usuario |  |  |
| 5 | El usuario Digita apellidos |  |  |
| 6 | El usuario selecciona cargo |  |  |
| 7 | El usuario selecciona la organización |  |  |
| 8 | El usuario selecciona rol para el usuario. | **9** | El sistema despliega la lista de roles almacenados en base de datos. |
| 10 | El usuario guarda la información. | **11** | El sistema muestra mensaje en pantalla “Usuario Guardado con éxito” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de que los datos no sean válidos o se presente algún problema se deberá mostrar mensaje de validación en pantalla. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Actualización de usuario | | | |
| 1 | El usuario digita el documento de usuario o el Id. | **2** | El sistema carga los datos del usuario. |
| 3 | El usuario oprime el botón editar. | **4** | El usuario habilita los campos nombres, apellidos, documento, cargo, rol para actualización |
| 5 | El usuario actualiza el nombre, apellidos, documento, cargo, rol y oprime botón guardar. | **6** | El sistema valida que todos los campos estén llenos, guarda y muestra mensaje de confirmación. |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje y el cursor se ubica en el campo. |

Tabla 25. Caso de uso 20

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 21**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear Roles | | | | | CU21 |
| Actores | Administrador | | | | | |
| Referencias | Requerimiento RF | | |  | | |
| Precondición | El administrador, operador y visualizador no afectaran los datos descargados. | | | | | |
| Postcondición | Los usuarios tendrán asignados diferentes permisos por usuario | | | | | |
| Autor |  | Fecha |  | | Versión |  |

|  |
| --- |
| Propósito |
| Creación roles del sistema |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal | | | |
| 1 | El usuario oprime botón nuevo o agregar | **2** | El sistema carga el Id automático que corresponde al nuevo rol |
| 3 | El usuario digita nombre del rol.  (Operador, Visualizador, editor preliminar, editor final) |  |  |
| 4 | El usuario selecciona permisos a cada rol |  |  |
| 5 | El usuario Digita descripción para cada rol |  |  |
| 6 | El usuario guarda la información | **7** | El sistema muestra mensaje en pantalla “Usuario Guardado con éxito” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de que los datos no sean válidos o se presente algún problema se deberá mostrar mensaje de validación en pantalla. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Actualización de rol | | | |
| 1 | El usuario digita el Id del rol guardado. | **2** | El sistema carga los datos almacenados. |
| 3 | El usuario oprime el botón editar. | **4** | El usuario habilita los campos nombres, permisos y descripción para actualización |
| 5 | El usuario actualiza el nombre, permisos, descripción y oprime botón guardar. | **6** | El sistema valida que todos los campos estén llenos, guarda y muestra mensaje de confirmación. “Guardado con éxito” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje y el cursor se ubica en el campo. |

Tabla 26. Caso de uso 21

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 22**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear permisos | | | | | CU22 |
| Actores | Administrador | | | | | |
| Referencias | Requerimiento RF | | |  | | |
| Precondición | El usuario deberá estar en el módulo procesamiento – Gestión de datos de procesamiento. | | | | | |
| Postcondición | El usuario debe contar con permisos administrador | | | | | |
| Autor |  | Fecha |  | | Versión |  |

|  |
| --- |
| Propósito |
| Creación de permisos para cada rol. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal Escenario 1  Visualizador | | | |
| 1 | El usuario selecciona la opción crear permisos. | **2** | El sistema carga el id automático que corresponde al nuevo permiso. |
| 3 | El usuario selecciona el rol visualizador |  |  |
| 4 | El usuario asigna los permisos para visualizador. | **5** | **Visualizador:** solo podrá ver datos de las estaciones |
| 4 | El usuario digita descripción para el estado. |  |  |
| 5 | El usuario guarda la información | **6** | El sistema muestra mensaje en pantalla “Usuario Guardado con éxito” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de que los datos no sean válidos o se presente algún problema se deberá mostrar mensaje de validación en pantalla. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal Escenario 2  Operador | | | |
| 1 | El usuario selecciona la opción crear permisos. | **2** | El sistema carga el id automático que corresponde al nuevo permiso. |
| 3 | El usuario selecciona el rol operador |  |  |
| 4 | El usuario asigna los permisos para operador. | **5** | **Operador:** Puede modificar datos de las estaciones (nombre, serial, organización, categoría, región,  localización, cuidad, latitud y longitud, elevación, parametros) |
| 4 | El usuario digita descripción para el estado. |  |  |
| 5 | El usuario guarda la información | **6** | El sistema muestra mensaje en pantalla “Usuario Guardado con éxito” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 6 | En caso de que los datos no sean válidos o se presente algún problema se deberá mostrar mensaje de validación en pantalla. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal Escenario 3  Editor Preliminar | | | |
| 1 | El usuario selecciona la opción crear permisos. | **2** | El sistema carga el id automático que corresponde al nuevo permiso. |
| 3 | El usuario selecciona el rol Editor Preliminar |  |  |
| 4 | El usuario asigna los permisos para Editor Preliminar. | **5** | **Editor Preliminar:** Valida y realiza corrección de datos. |
| 4 | El usuario digita descripción para el estado. |  |  |
| 5 | El usuario guarda la información | **6** | El sistema muestra mensaje en pantalla “Usuario Guardado con éxito” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de que los datos no sean válidos o se presente algún problema se deberá mostrar mensaje de validación en pantalla. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal Escenario 4  Editor Final | | | |
| 1 | El usuario selecciona la opción crear permisos. | **2** | El sistema carga el id automático que corresponde al nuevo permiso. |
| 3 | El usuario selecciona el rol Editor Final |  |  |
| 4 | El usuario asigna los permisos para editor Final. | **5** | **Editor Final:** Realiza validación, corrección de datos, edita ediciones realizadas por editor preliminar y cierra datos |
| 4 | El usuario digita descripción para el estado. |  |  |
| 5 | El usuario guarda la información | **6** | El sistema muestra mensaje en pantalla “Usuario Guardado con éxito” |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 5  Actualización de estado | | | |
| 1 | El usuario digita el Id del estado guardado. | **2** | El sistema carga los datos almacenados. |
| 3 | El usuario oprime el botón editar. | **4** | El usuario habilita los campos nombre, descripción para actualización |
| 5 | El usuario actualiza el nombre, descripción y oprime botón guardar. | **6** | El sistema valida que todos los campos estén llenos, guarda la información |
|  |  | **7** | El sistema muestra mensaje de confirmación |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje y el cursor se ubica en el campo. |

Tabla 27. Caso de uso 22

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 23**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Crear estados | | | | | CU23 |
| Actores | Administrador | | | | | |
| Referencias | Requerimiento RF | | |  | | |
| Precondición | El usuario deberá estar en el módulo procesamiento – Gestión de datos de procesamiento. | | | | | |
| Postcondición | El usuario debe contar con permisos administrador | | | | | |
| Autor |  | Fecha |  | | Versión |  |

|  |
| --- |
| Propósito |
| Creación de estados bandera |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal | | | |
| 1 | El usuario oprime botón nuevo o agregar | **2** | El sistema carga el Id automático que corresponde al nuevo estado |
| 3 | El usuario digita nombre del estado.  (Dato valido)  (Dato invalido)  (tope alto)  (tope bajo)  (Problema de comunicación)  (Recepción de datos)  (Valor negativo) |  |  |
| 4 | El usuario digita descripción para el estado. |  |  |
| 5 | El usuario guarda la información | **6** | El sistema muestra mensaje en pantalla “Usuario Guardado con éxito” |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | En caso de que los datos no sean válidos o se presente algún problema se deberá mostrar mensaje de validación en pantalla. |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal escenario 2  Actualización de estado | | | |
| 1 | El usuario digita el Id del estado guardado. | **2** | El sistema carga los datos almacenados. |
| 3 | El usuario oprime el botón editar. | **4** | El usuario habilita los campos nombre, descripción para actualización |
| 5 | El usuario actualiza el nombre, descripción y oprime botón guardar. | **6** | El sistema valida que todos los campos estén llenos, guarda la información |
|  |  | **7** | El sistema muestra mensaje de confirmación |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | Hay campos vacíos, el sistema muestra el mensaje y el cursor se ubica en el campo. |

Tabla 28. Caso de uso 23

Autor: Fabian Mina – Daniela Romero

**CASOS DE USO 24**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caso De Uso | Asignar estados | | | | | CU24 |
| Actores | Administrador | | | | | |
| Referencias | Requerimiento RF | | |  | | |
| Precondición | El usuario deberá estar en el módulo procesamiento – Gestión de datos de procesamiento. | | | | | |
| Postcondición | El usuario debe contar con permisos administrador | | | | | |
| Autor |  | Fecha |  | | Versión |  |

|  |
| --- |
| Propósito |
| Asignación de estados bandera |

|  |  |  |  |
| --- | --- | --- | --- |
| Curso Normal | | | |
|  |  | **1** | El sistema valida la información procesada. |
|  |  | **2** | El sistema valida valores con topes altos. Asigna estado (Tope alto) |
|  |  | **3** | El sistema valida valores con topes bajos. Asigna estado (Tope bajo) |
|  |  | **4** | El sistema valida valores con topes bajos. Asigna estado (Tope bajo) |
|  |  | **5** | El sistema valida valores con problemas de comunicación. Asigna estado (problemas de comunicación) |
|  |  | **6** | El sistema valida valores con Recepción de datos. Asigna estado (Recepción de datos) |
|  |  | **7** | El sistema valida valores negativos. Asigna estado (valores negativos) |

|  |  |
| --- | --- |
| Cursos Alternos | |
| 1 | El sistema pregunta al usuario que desea hacer:” Corregir” “Invalidar” “No hacer nada” |
|  | El usuario selecciona Corregir: Envía a CU 1 Curso Normal Escenario 3  El usuario selecciona Invalidar: Envía CU1 Curso normal Escenario 4 |
| 2 | El sistema pregunta al usuario que desea hacer:” Corregir” “Invalidar” “No hacer nada” |
|  | El usuario selecciona Corregir: Envía a CU 1 Curso Normal Escenario 3  El usuario selecciona Invalidar: Envía CU1 Curso normal Escenario 4 |
| 3 | El sistema pregunta al usuario que desea hacer:” Corregir” “Invalidar” “No hacer nada” |
|  | El usuario selecciona Corregir: Envía a CU 1 Curso Normal Escenario 3  El usuario selecciona Invalidar: Envía CU1 Curso normal Escenario 4 |
| 4 | El sistema pregunta al usuario que desea hacer:” Corregir” “Invalidar” “No hacer nada” |
|  | El usuario selecciona Corregir: Envía a CU 1 Curso Normal Escenario 3  El usuario selecciona Invalidar: Envía CU1 Curso normal Escenario 4 |
| 5 | El sistema pregunta al usuario que desea hacer:” Corregir” “Invalidar” “No hacer nada” |
|  | El usuario selecciona Corregir: Envía a CU 1 Curso Normal Escenario 3  El usuario selecciona Invalidar: Envía CU1 Curso normal Escenario 4 |
| 6 | El sistema pregunta al usuario que desea hacer:” Corregir” “Invalidar” “No hacer nada” |
|  | El usuario selecciona Corregir: Envía a CU 1 Curso Normal Escenario 3  El usuario selecciona Invalidar: Envía CU1 Curso normal Escenario 4 |
| 7 | El sistema pregunta al usuario que desea hacer:” Corregir” “Invalidar” “No hacer nada” |
|  | El usuario selecciona Corregir: Envía a CU 1 Curso Normal Escenario 3  El usuario selecciona Invalidar: Envía CU1 Curso normal Escenario 4 |

Tabla 29. Caso de uso 24

Autor: Fabian Mina – Daniela Romero

* 1. **Especificación casos de prueba**

Componentes para evaluar

**Usabilidad: Para las pruebas de usabilidad realizamos un test de usuario** que se basa en la observación y análisis de cómo un grupo de usuarios reales manipulan el software y cuál es su comportamiento en cada actividad que realice, esta prueba es recomendable usarla para descubrir errores de diseño motivados por el no cumplimiento en el desarrollo de principios generales de usabilidad

Para el test de usuario se evaluaron las siguientes actividades.

* Seleccionar estación para conexión
* Crear Mantenimiento
* Crear región
* Crear estación
* Crear organización
* Crear organización
* Crear Ciudad
* Crear rango
* Crear alerta
* Crear cuadro de tiempo
* Login

Para el test de usuario se entrevistarán 20 personas las cuales consiste en realizar una encuesta y obtener como resultado que tan amigable es el software para las partes interesadas.

**Ejemplo de Test de usuario**

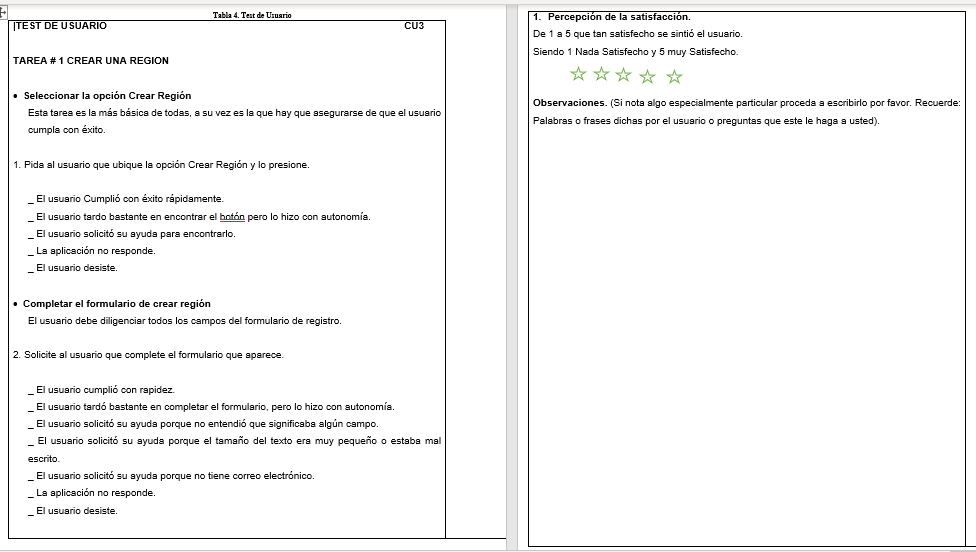


Imagen 8. Test de usuario

Autor: Fabian Mina – Daniela Romero

**Funcional**: Incluyen lo que son las pruebas unitarias y pruebas de integración, con las cuales se evaluara fragmentos de código para validar que se cumpla la funcionalidad.

**Formato registro de pruebas**

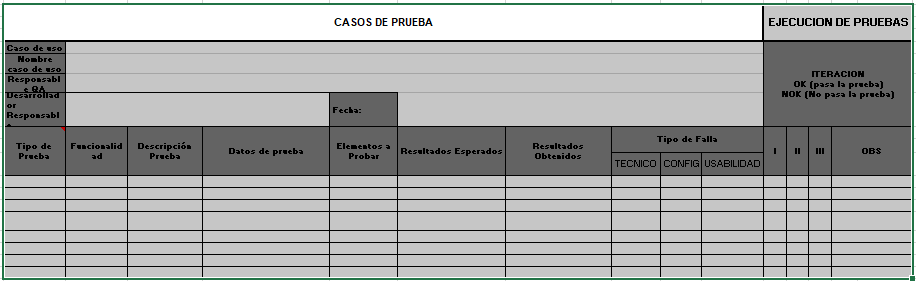


Imagen 9. Formato registro de pruebas

Autor: Fabian Mina – Daniela Romero

**Eficiencia:** Se prueba el comportamiento del aplicativo y los tiempos de respuesta a las peticiones realizadas por los usuarios, para ello se realizaron los casos de prueba (Imagen 6) donde se establece los tiempos y los resultados esperados; para llevar un control a la hora de realizar pruebas se diseña un formato donde se debe ir verificando si los tiempos de espera en el software cumplieron con lo establecido

**Seguridad:** Se probará la, autenticación de los usuarios y encriptación de contraseñas en base de datos, para esta prueba utilizamos el CU11 VALIDACION DE USUARIO

Se realizarán el test basándonos en el caso de prueba de CU 11

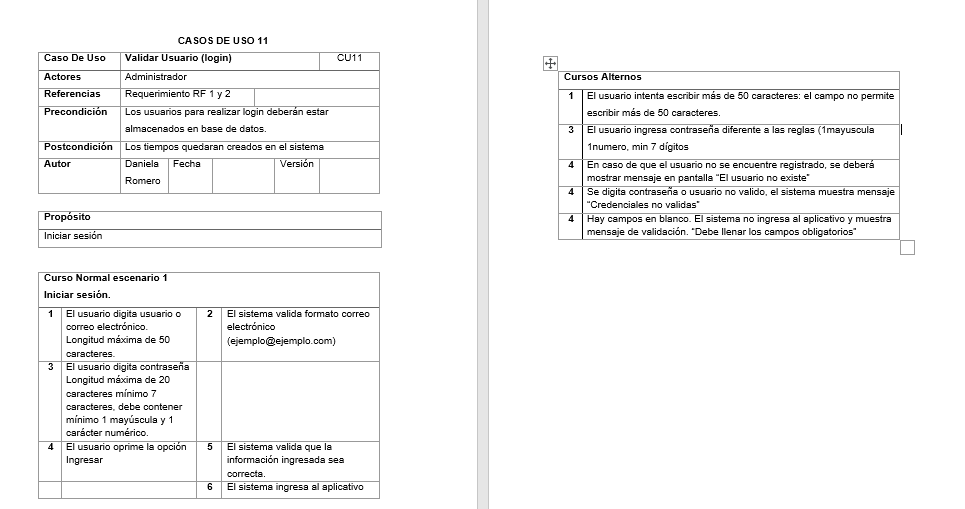


Imagen 10. Ejemplo caso de uso 11

Autor: Fabian Mina – Daniela Romero

* 1. **Procedimiento de prueba**

Herramientas que se usaran para la elaboración de este proyecto.

**Framework Jest:** Se validan fragmentos de código de cada clase, probando los métodos completos, crea, consultar, actualizar, ver; la respuesta deberá ser positiva (pass) para cada test de prueba.

**Carga masiva de datos:** Se realiza con la herramienta Jmeter V.5.2, se verificará la conexión de la herramienta con el gestor de base de datos PostgreSQL, se realizará una carga de 100 usuario en un término de 15 segundos.

* 1. **Reporte de transmisión de prueba**

Describe los ítems para prueba, donde encontrarlos y se aprueba para su liberación. Es muy importante ya que garantiza al encargado de pruebas tester QA de que los ítems están listos para ser probados.

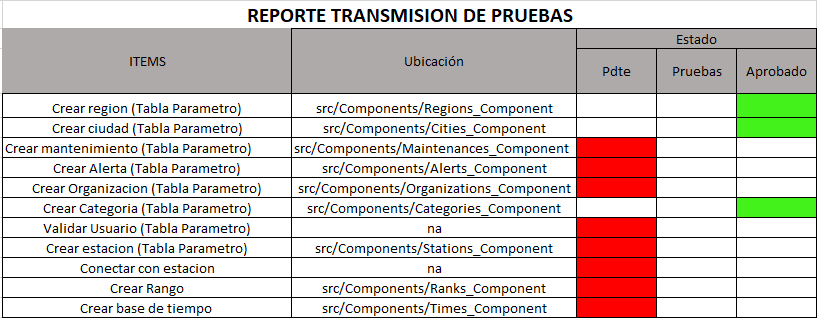


Imagen11. Reporte transmisión de pruebas

Autor: Fabian Mina – Daniela Romero

Las pruebas realizadas en este proyecto se realizaron con las siguientes herramientas integrándolas para una correcta ejecución. Ver imagen 12

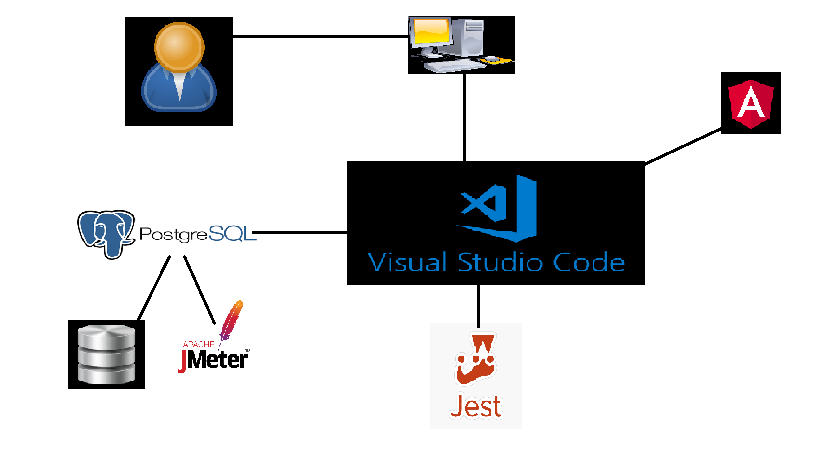


Imagen 12. Diagrama herramienta usadas

Autor: Fabian Mina – Daniela Romero

Con este diagrama se muestra la estructura de herramientas para la ejecución de pruebas de el aplicativo que sus componentes son:

Herramientas usadas para la funcionalidad del proyecto

* Visual Studio Code

Es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java, angular, php. Existe además un número importante de módulos para extenderlo. Visual Studio Code​ es un producto libre y gratuito sin restricciones de uso.



* PostgreSQL 11

Es un gestor de base de datos relacional orientado a objetos y libre, que va dirigido a una comunidad de desarrolladores, no tiene gestor de defectos haciendo muy difícil conocer el estado de sus defectos.



* Angular

Framework para aplicaciones web desarrollado en typescript, angular es de código abierto, se utiliza para crear crear y mantener aplicaciones web de una sola página, su objetivo es aumentar las aplicaciones basadas en navegador con capacidad, modelo vista controlador (MVC)



* 1. **Ejecución de pruebas**
     1. **Log de prueba**

En este documento se registran los detalles sobre que casos de pruebas se han ejecutado, en que orden y sus resultados (Pass/fail). Si hay inconsistencias, se levanta o se actualiza un reporte de incidencias.

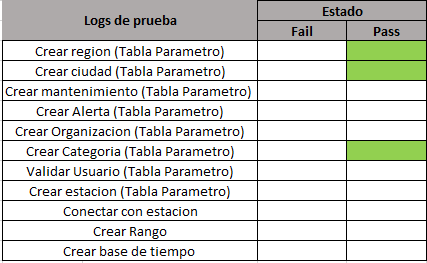


Imagen13. Logs de prueba

Autor: Fabian Mina – Daniela Romero

* **Pruebas de usabilidad**
* **Pruebas Funcionalidad**

Se realizaron estas pruebas funcionales de manera manual, probando la funcionalidad de cada acción del aplicativo, tiempo de inicio a la pantalla principal, mensajes mostrados en pantalla, guardar información, modificar información.

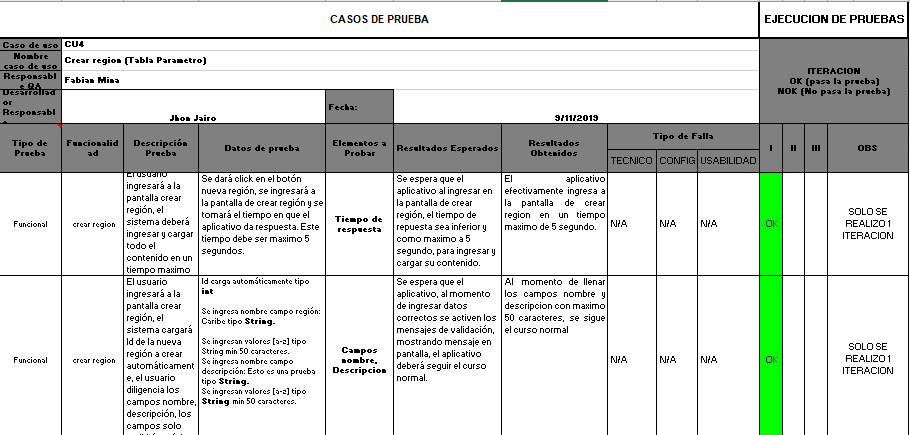


Imagen 14. Ejemplo pruebas funcionales

Autor: Fabian Mina – Daniela Romero

* **Pruebas Unitarias**

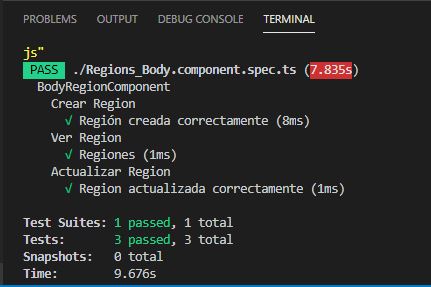
Se escribieron pruebas unitarias con la herramienta Jest, realizando un test completo de cada método. Se probaron los métodos guardar, crear, consultar, modificar. 

Imagen 15. Resultado prueba unitaria componente región

Autor: Fabian Mina – Daniela Romero

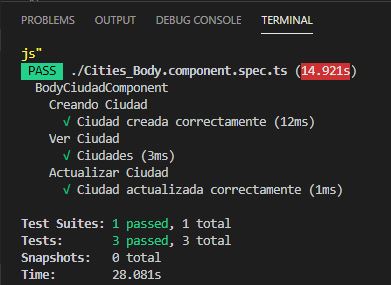


Imagen 16. Resultado prueba unitaria componente ciudad

Autor: Fabian Mina – Daniela Romero

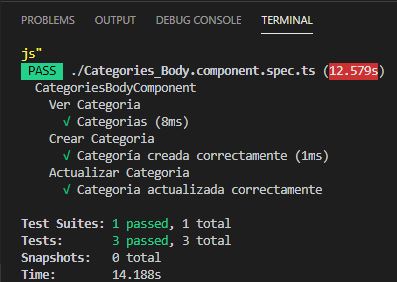


Imagen 17. Resultado prueba unitaria componente categoría

Autor: Fabian Mina – Daniela Romero

* **Pruebas de carga**

Se realizaron las pruebas de carga con la herramienta Jmeter, donde se simularon 500 usuarios ingresando simultáneamente en un ciclo de 15 segundos.

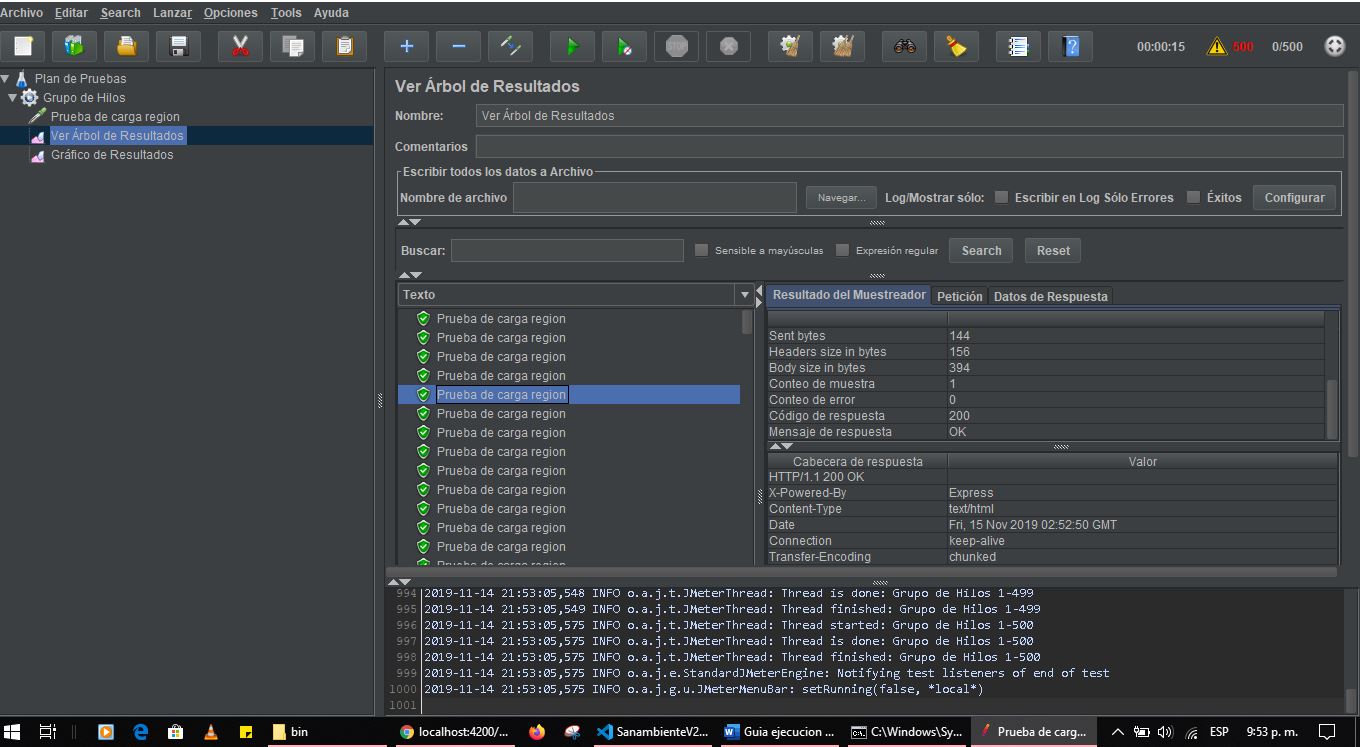


Imagen 18. Resultado prueba de carga componente región

Autor: Fabian Mina – Daniela Romero

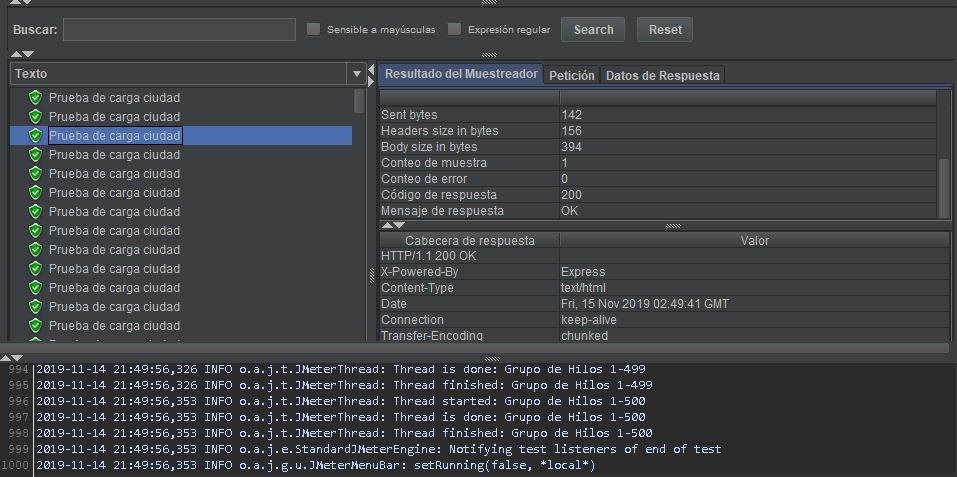


Imagen 19. Resultado prueba de carga componente Ciudad

Autor: Fabian Mina – Daniela Romero

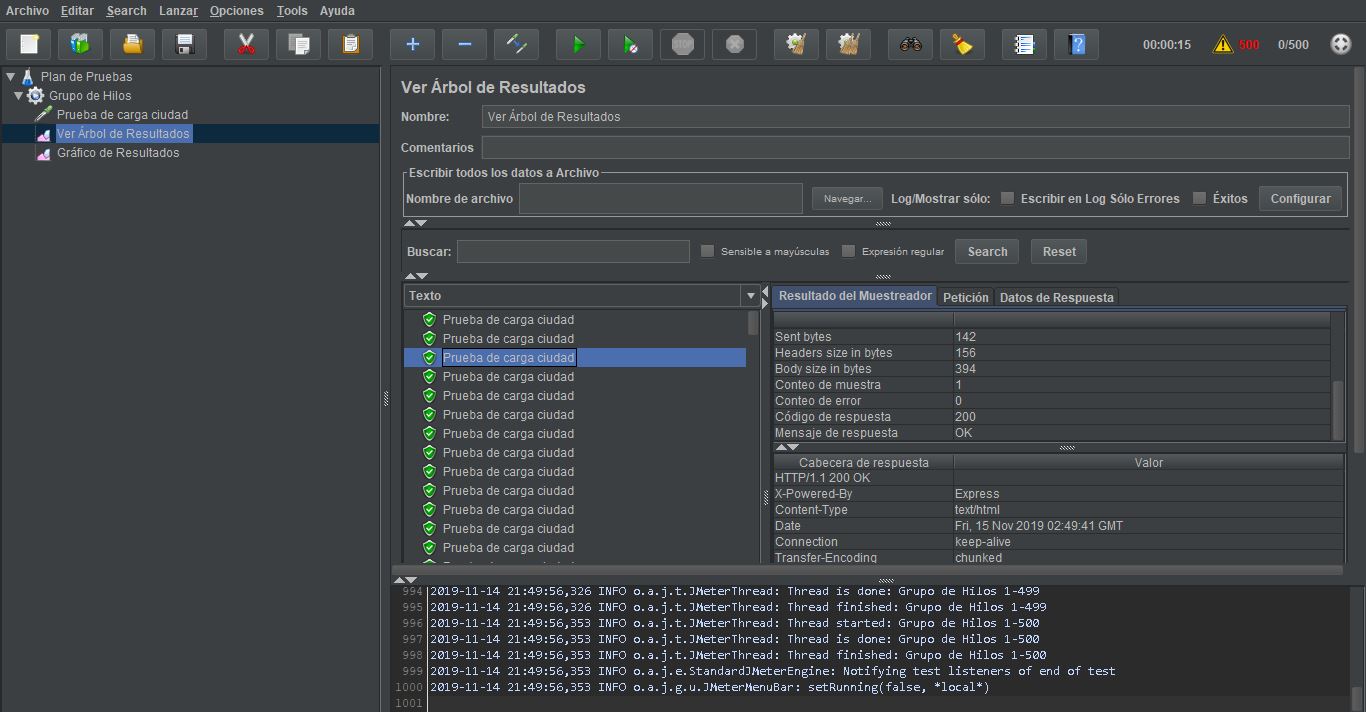


Imagen 20. Resultado prueba de carga componente Categoría

Autor: Fabian Mina – Daniela Romero

* + 1. **REPOTE DE INCIDENTES DE PRUEBA**

Durante la ejecución de pruebas, se registra un reporte de incidencia, donde se evidencian todas las inconformidades o errores presentados al momento de realizarle pruebas a los componentes.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Reporte de incidentes de pruebas** | | | | |
| **Autores:** | **Daniela Romero – Fabian Mina** | | | |
| **CU** | | **ELEMENTO PROBADO** | **ITERACION** | **DETALLE** |
| CU 3  Crear región | | Botón Guardar | I | Cuando se llenan los campos nombre y descripción, y se da en la opción guardar el aplicativo no muestra mensaje de confirmación. |
| CU 7  Crear ciudad | | Botón Guardar | I | Al momento de guardar la información, y presionar el botón guardar, no se muestra mensaje de confirmación |
| Nota: Revisar formato casos de prueba Fase 2 | | | | |

Tabla 30. Reporte de incidencias

Autor: Fabian Mina – Daniela Romero

* 1. **TERMINO DE PRUEBAS**

Una ves terminado el proceso de prueba, se realiza una evaluación de los resultados de las pruebas, que tan bien salieron, el numero de incidentes reportados, evaluación de la calidad del sistema.

* + 1. **REPORTE DE PRUEBAS**

|  |  |
| --- | --- |
| **REPORTE DE PRUEBAS**  **Evaluación** | |
| **Cantidad de componentes probados** | **3** |
| **Cantidad de test aplicados** | **9** |
| **Numero de incidencias reportadas** | **2** |
| **Observación.**  Al realizar las pruebas a los componentes Ciudad, Región y categoría se puede evidenciar que el aplicativo en etapa de desarrollo ha cumplido con los requerimientos establecidos desde un principio, se espera la integración de los componentes faltantes para así asegurar la calidad de los mismos y lograr entregar un producto útil para el usuario final. | |

Tabla 31. Reporte de Pruebas

Autor: Fabian Mina – Daniela Romero

# **CONCLUSIONES**

Para concluir se demuestra que al aplicar un modelo de pruebas es una buena opción para diagnosticar si el producto es confiable y de buena calidad, con ello al aplicar este modelo de pruebas al aplicativo, se demuestra que va en un proceso de desarrollo viable y se evidencia que aun esta presto a mejoras continuas para futuras versiones, ya que el aplicativo está en etapa de desarrollo y proceso de madurez.

# **BIBLIOGRAFÍA**

Callejas-Cuervo, M., Alarcón-Aldana, A. C., & Álvarez-Carreño, A. M. (2017). Modelos de calidad del software, un estado del arte. *ENTRAMADO*, *13*(1), 236-250. https://doi.org/10.18041/entramado.2017v13n1.25125

Chumacero Botet, I. (2010). Sistema de Calidad Total. Metodología y ejemplo de aplicación a Empresas de Diseño. *Revista de Arquitectura e Ingeniería*, *4*(2). Recuperado de http://www.redalyc.org/resumen.oa?id=193915954005

CS\_07\_Estandares\_para\_pruebas\_software.pdf. (s. f.). Recuperado de http://artemisa.unicauca.edu.co/~cardila/CS\_07\_Estandares\_para\_pruebas\_software.pdf

ESTÁNDARES DE CALIDAD DEL SOFTWARE. (s. f.). Recuperado 7 de octubre de 2018, de http://estandarescalidadsoftware.blogspot.com/

Fernando Arciniega. (2017, mayo 23). Normas y Estándares de calidad para el desarrollo de Software. Recuperado 7 de octubre de 2018, de https://fernandoarciniega.com/normas-y-estandares-de-calidad-para-el-desarrollo-de-software/

IEEE 829 Norma para el software y la documentación de prueba. (s. f.). Recuperado 25 de noviembre de 2018, de https://prezi.com/yh-yob98\_ayd/ieee-829-norma-para-el-software-y-la-documentacion-de-prueba/

Metodologias De Un Modelo De Calidad. (s. f.). Recuperado 25 de noviembre de 2018, de http://www.cyta.com.ar/biblioteca/bddoc/bdlibros/tqm/3\_metodologia/3\_metodologia.htm

NORMAS ISO 25000. (s. f.). Recuperado 7 de octubre de 2018, de https://iso25000.com/index.php/normas-iso-25000

Scalone, L. F. (s. f.). MAESTRIA EN INGENIERIA EN CALIDAD, 461.