

**Gerencia de Soluciones**

**Documentación Funcional**

NECOPER-364

Contenido

[1 Información del Documento 3](#_Toc106725602)

[1.1. Historial del Documento 3](#_Toc106725603)

[2 Descripción General 4](#_Toc106725604)

[2.1 Necesidad y Alcance *(Mandatorio)* 4](#_Toc106725607)

[2.1.1 Necesidad 4](#_Toc106725608)

[2.1.2 Alcance 4](#_Toc106725609)

[2.2 Dependencias *(Mandatorio)* 4](#_Toc106725610)

[2.3 Procesos de Negocio Afectados *(Mandatorio)* 4](#_Toc106725611)

[2.4 Otras Consideraciones o Supuestos *(Opcional)* 4](#_Toc106725612)

[3. Descripción Detallada de la propuesta de solución *(Mandatorio)* 5](#_Toc106725613)

[4 Sistemas e integraciones afectados *(Mandatorio)* 6](#_Toc106725615)

[5 Perfiles *(Mandatorio)* 7](#_Toc106725616)

[6 Escenarios de prueba *(Mandatorio)* 8](#_Toc106725617)

# Información del Documento

## Historial del Documento

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Autor | Versión | Fecha | Resumen de Cambios | Sección | Hoja |
| RSLEIVA | 1 | 21/9/2023 | Creación DF | - | - |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# Descripción General



## Necesidad y Alcance

1. A causa de la cancelación del proceso Poblar Spr Client Interface - Circuito 15 en Nexus-Prod-Diario se requiere analizar el proceso para evitar cancelaciones.

### Necesidad

1. Ante estas cancelaciones causadas por la inserción de teléfonos mayores al limite permitido se buscara la solución del lado de procesos.

### Alcance

1. Se realizara manejadores de excepciones en el package para que el proceso continue sin cancelar.

## Dependencias

1. Sin Dependencias

## Procesos de Negocio Afectados

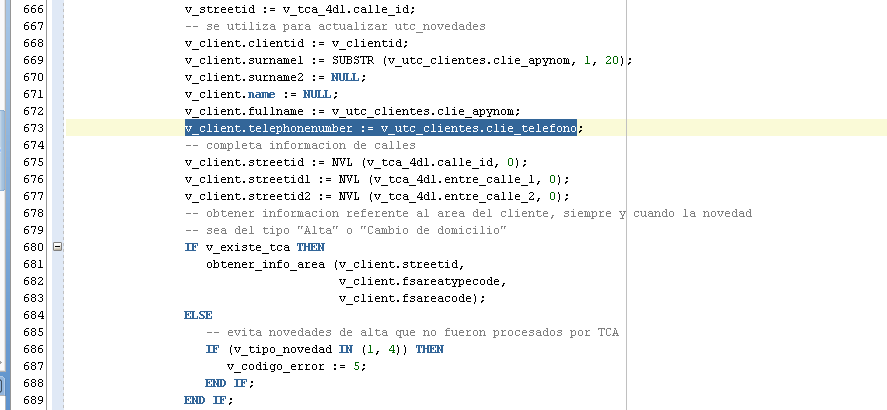
1. Sin procesos de negocios afectados

## Otras Consideraciones o Supuestos

Sin otras consideraciones o supuestos

# Descripción Detallada de la propuesta de solución

Se detecta que en el package NEXUS\_GIS.INTERFAZ\_CLIENTES, en el procedimiento PROCESAR\_NOVEDADES\_UTC al momento de pasar el teléfono, trunca el circuito si el valor pasado supera la cantidad máxima permitida del campo:



Se propone encerrar esa línea dentro de una estructura BEGIN/EXCEPTION para capturar el error y en caso de suceder truncar el teléfono pasado:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

# Sistemas e integraciones afectados

No afecta otros sistemas o integraciones

# Perfiles

No afecta perfiles

# Escenarios de prueba

|  |  |
| --- | --- |
| Requisito Funcional # | 1 |
| Tipo de Falla | Falla de datos |
| Nombre | Pruebas de entrada de datos |
| Prueba realizada por | David Draise |
| Precondiciones | Contar con permisos para GISQA |
| Condiciones de prueba | Modificar un registro de la tabla UTC.CLIENTES, en GISQA01, para que el teléfono tenga más de 20 caracteres. |
| Pasos | P1: Posterior a la modificación del teléfono, correr manualmente el ksh. Esto debería simular el circuito, pasando el teléfono.  P2: Verificar si el teléfono quedo truncado en NEXUS\_GIS.SPRCLIENTINTERFACE. |
| Resultados esperados | Se espera ver el teléfono insertado, pero solo los primeros 20 caracteres de izquierda a derecha. El cambio también contempla posibles espacios de izquierda y derecha por lo que, si hay espacios a la izquierda estos no deben ser considerados. |
| Resultados obtenidos | Se observa que el teléfono ha sido truncado. |
| Resultado del test | Éxito |
| Comentarios | - |

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

**Guía del plan de pruebas**

**Fallas de datos**: ejemplos de fallas relacionadas con la gestión de datos en la app.

* Pruebas de entrada de datos: valida la entrada de datos en los campos.
* Pruebas de integridad de datos: Realizar pruebas para comprobar que los datos se almacenen correctamente
* Pruebas de manejo de errores: Probar ingresos de datos incorrectos y verificar si muestra mensajes de error claros.
* Pruebas de carga de datos: Generar pruebas con cantidad significativa de datos para verificar que la app no presente problemas de rendimiento o errores al procesarlos.
* Pruebas de interoperabilidad: Puedes realizar pruebas de integración con sistemas externos para asegurarte de que los datos se transfieran y se procesen correctamente.

**Fallas de control**: fallas en el control y la validación de los datos ingresados.

* Pruebas de flujo de control: Diseñar flujo de control, y asegurarte que la app se comporte correctamente.
* Pruebas de decisiones: Diseñar casos de prueba para asegurarte de que se tomen las acciones correctas.
* Pruebas de validación de datos: Diseñar escenarios de validación y asegurar que la app rechace o acepte correctamente los datos según corresponda.
* Pruebas de seguridad y permisos: Diseñar casos que intenten acceder a funciones o datos restringidos.
* Pruebas de rendimiento y carga: Simular escenarios de alta demanda para verificar si mantiene el control y responda correctamente en todas las situaciones.

**Fallas entradas/salida**: Analiza situaciones en las que se produzcan errores en los procesos de entrada y salida de datos.

* Pruebas de integración: Se busca identificar problemas de comunicación o incompatibilidades entre los módulos.
* Pruebas de interfaz de usuario: Se verifican aspectos como la usabilidad, la respuesta a las acciones del usuario y la presentación correcta de los resultados.
* Pruebas de compatibilidad: Se verifica que las entradas/salidas se procesen correctamente en cada entorno específico.
* Pruebas de entrada de datos: Se prueban diferentes combinaciones de datos válidos e inválidos.
* Pruebas de salida de datos: Se comparan los resultados obtenidos con los esperados para identificar discrepancias.

**Fallas de interfaz**: Identifica posibles fallas relacionadas con la interfaz de usuario de la aplicación.

* Pruebas de usabilidad: Identificar problemas de diseño, navegación, retroalimentación visual, que afecte la usabilidad del sistema.
* Pruebas de navegación: Se evalúa la estructura de menús, los enlaces entre páginas, los botones de navegación.
* Pruebas de compatibilidad de dispositivos: Se comprueba la visualización en diferentes navegadores web y sistemas operativos.
* Pruebas de respuesta a eventos: evalúan la respuesta de la interfaz ante eventos como tap/clic en botones.
* Pruebas de accesibilidad: evalúan la accesibilidad de la interfaz para personas con discapacidades visuales, auditivas o motoras.

**Fallas de almacenamiento**: Analiza situaciones en las que se produzcan fallas en el almacenamiento y gestión de datos en la aplicación.

* Pruebas de capacidad de almacenamiento: Realizar una carga masiva de datos para evaluar el rendimiento y detectar posibles limitaciones o problemas de almacenamiento.
* Pruebas de integridad de datos: Se verifican precisión, consistencia y validez de los datos.
* Pruebas de recuperación de datos: Se pueden simular escenarios de pérdida de datos y verificar que el sistema pueda recuperarse adecuadamente y restaurar los datos de forma íntegra.
* Pruebas de rendimiento de acceso a datos: Se pueden medir los tiempos de respuesta al realizar consultas o transacciones.
* Pruebas de seguridad de almacenamiento: Se evalúan aspectos como la encriptación de datos, los controles de acceso, las copias de seguridad y la prevención de vulnerabilidades de seguridad.