



El servicio público
es de todos

Función
Pública



modelo integrado
de planeación
y gestión



MEDICIÓN
DESEMPEÑO INSTITUCIONAL

Plan de Pruebas y Documento Técnico de validación de tablas y generación de fichas temporales de validación

Operación estadística Medición del Desempeño Institucional

Versión 2

Diciembre de 2020



Contenido

Introducción.....	4
I. Plan de Pruebas.....	5
1. Objetivos.....	5
2. Alcance.....	5
Fase I – Pruebas a las preguntas	6
Resultados de las pruebas – fase I.....	8
Fase II – Pruebas al aplicativo	8
Resultado de las pruebas – fase II.....	10
Fase III – Pruebas a las visualizaciones con los resultados de la medición	10
Resultados de las pruebas – fase III	11
II. Documento técnico Validación de tablas y generación de fichas temporales de validación.....	12
Estructura de validación.....	12
Input.....	12
Mcr.....	13
Doc	13
Rmd.....	13
Proceso de validación: scr	13
Archivos y Tablas	14
Archivo MotorValida	15
Tabla qMVal	15
Archivo MotorValidaAAMMDD_HHMM	16
qMVal.....	16
Corrida_Casos_V	17
Corrida_Casos_C	18
V_ID_ACCION	19
C_ID_ACCION	20
Archivo CCVPPP.....	20
Códigos.....	21
V01.NoNulo	21
V02.NoDupl.....	21
V02.NoDuplW.....	22
V03.LargoE	23
V04.ValEsp	23
C01.tConD	24
C01.tConT.....	24
C01.tConN	25
C01.tConF.....	25
01_ValidaGeneral.....	26
02_ValidaFicha.....	26
CCVPPP.Rmd.....	27
Listas o reglas de validación por procesos o códigos	27
Proceso C003.....	27
Proceso C004.....	28
Proceso C005.....	29
Proceso C006.....	31
Proceso C007.....	31
Proceso C008.....	32
Proceso C009.....	32
Proceso C010.....	33

Proceso C011.....	33
Proceso C012.....	34
Proceso C013.....	34
Proceso C014.....	35
Proceso C015.....	35
Bibliografía	36

Índice de tablas

Tabla 1 Listado de Mesas de Trabajo para Validación de Preguntas	7
Tabla 2 Resultado Revisión de Pares	8
Tabla 3 Formato de pruebas al aplicativo FURAG.....	9
Tabla 4 Formato de prueba a las visualizaciones	11

Introducción

Con el fin de identificar y mitigar los riesgos asociados al proceso estadístico específicamente en cuanto a las preguntas y al instrumento de recolección de información y a la difusión de los resultados de la medición, se construye el presente documento teniendo como fin orientar la ejecución del plan de pruebas a los instrumentos anteriormente descritos, empleando los lineamientos difundidos por el Departamento Administrativo Nacional de Estadística y las particularidades de la Medición del Desempeño Institucional en el marco del Modelo Integrado de Planeación y Gestión (Dirección de Gestión y Desempeño Institucional, 2019). En consecuencia, las pruebas al diseño de preguntas y al aplicativo de captura sirven de herramienta para validar y ajustar el proceso estadístico con respecto al año base lo cual pretende dar a los datos recolectados mayor consistencia en el marco de la comparación entre vigencias y los resultados proyectados.

En este orden de ideas es necesario probar los resultados a difundir para garantizar transparencia en el dato suministrado, lo cual también se encuentra alineado con la consistencia de los datos en cuanto a la comparación entre vigencias medidas. Por esta razón se llevan a cabo pruebas tanto al aplicativo FURAG antes de capturar datos, como a las visualizaciones en Power BI con los resultados de la medición, para mitigar el riesgo de presentar datos erróneos o que dificulten la comparación entre vigencias.

De otra parte, el proceso mediante el cual se validan las tablas que hacen parte de la dinámica del proceso, se articula en cada código como punto de referencia sobre el cual recae la necesidad de abordar de manera sistemática un conjunto de reglas de verificación. Por esta razón para garantizar la continuidad de la operación estadística, el presente documento describe la interacción lógica en cada parte del proceso. En este orden de ideas, al interior de la Dirección de Gestión y Desempeño Institucional se han identificado las fases que son suficientes para validar la información producida en el ejercicio del procesamiento del FURAG, lo cual ha permitido estandarizar el proceso de manera transversal para construir un código estándar que se implementa en los 15 códigos (en lenguaje R).

Por esta razón, la segunda parte de este documento, tiene como objetivo explicar de forma detallada la interacción de cada tabla que interviene a lo largo del proceso de verificación y validación, gobernado igualmente bajo la estructura del flujograma “procesamiento estadístico – Medición del Desempeño Institucional”.

I. Plan de Pruebas

1. Objetivos

Objetivo general

Garantizar coherencia, trazabilidad y correspondencia entre el resultado obtenido por la operación estadística y el comportamiento de las unidades de observación (entidades de la Rama Ejecutiva del orden nacional), por medio de un plan de pruebas organizado de manera sistemática que permita mitigar los riesgos en torno a las preguntas empleadas en la medición, la herramienta para la captura de datos y la plataforma en la cual se presentan los resultados de la medición.

Objetivos específicos

- Estructurar un plan de pruebas a las preguntas formuladas para la Medición del Desempeño Institucional vigencia 2019 tomado como insumo principal las empleadas en la medición pasada y el análisis estadístico que brinda mayor consistencia para reformular o construir preguntas nuevas.
- Estructura un plan de pruebas al aplicativo FURAG (Formulario Único de Reporte de Avances de la Gestión) que mitigue los riesgos asociados a la captura de datos.
- Estructurar un plan de pruebas a las visualizaciones en Power BI con los resultados de la Medición del Desempeño Institucional.

2. Alcance

Con base en los lineamientos para la definición y aplicación de pruebas en las operaciones estadísticas definidas por el Departamento Administrativo Nacional de Estadística (DANE, 2014), se establecen tres fases para la ejecución de pruebas al diseño que a la vez corresponden a la secuencia lógica y sistemática de ajuste en primera instancia a las preguntas, posteriormente al aplicativo y finalmente a las visualizaciones en las cuales se presentan los resultados de la medición, minimizando el riesgo asociado a la captura de datos, el procesamiento de los mismos y la presentación de resultados con cada uno de los índices calculados. En este orden de ideas la Dirección de Gestión y Desempeño Institucional junto con los líderes de política establecidos en el Modelo Integrado de Planeación y Gestión MIPG son los encargados de ejecutar las pruebas según lo establecido en las fases propuestas.

Con el objetivo principal de garantizar coherencia, trazabilidad y correspondencia entre el resultado obtenido por la operación estadística y el comportamiento de las unidades de observación (entidades de la Rama Ejecutiva del orden nacional), se planean de manera organizada y sistemática unas pruebas, que permita mitigar los riesgos en torno a las preguntas empleadas en la medición y la herramienta para la captura de datos.

Con base en los lineamientos para la definición y aplicación de pruebas en las operaciones estadísticas definidas por el Departamento Administrativo Nacional de Estadística (DANE, 2014), se establecen tres fases para la ejecución de pruebas al diseño que a la vez corresponden a la secuencia lógica y sistemática de ajuste en primera instancia a las preguntas y posteriormente al aplicativo, minimizando el

riesgo asociado a la captura de datos y el procesamiento de los mismos, para al final validar los resultados que se difunden principalmente en las visualizaciones de Power BI.

Fase I – Pruebas a las preguntas

Esta fase se divide en dos momentos; el primero, bajo un esquema de pruebas de escritorio se validan las preguntas del cuestionario diseñado para la aplicación anterior (vigencia 2018) de todas las políticas a partir de dos análisis: uno cualitativo construido con base en las inquietudes que presentaron las entidades objeto de medición en temas como comprensión de la pregunta, redacción o ámbito de aplicación de las políticas; y un análisis estadístico de confiabilidad de las preguntas a través de un análisis univariado, cuyos resultados se pueden apreciar en la carpeta de archivos 2019-08-26_Analisis_confiabilidad_univariados disponible en la ruta \\Yaksa\11300dgd\2020\DOCUMENTOS_APOYO\MIPG\MEDICION_DESEMPEÑO_INSTITUCIONAL\ DISEÑO-PRUEBAS_CAP_6\PRUEBAS\PRUEBAS_CUESTIONARIO_PREGUNTAS\ANALISIS_UNIVARIADOS

Por su parte, el listado de preguntas objeto de análisis cuantitativo y cualitativo se puede apreciar en el archivo 2019-24-09_Preguntas_furag_objeto_analisis_cuantitativo_cualitativo, disponible en la ruta \\Yaksa\11300dgd\2020\DOCUMENTOS_APOYO\MIPG\MEDICION_DESEMPEÑO_INSTITUCIONAL\ DISEÑO-PRUEBAS_CAP_6\ DISEÑO_CUESTIONARIO.

En cuanto a las preguntas de las políticas de defensa jurídica y gestión presupuestal y eficiencia del gasto público, en virtud de que son las políticas de más reciente incorporación en el Modelo, el trabajo de validación y ajuste se llevará a cabo a través de mesas de trabajo entre la Dirección de Gestión y Desempeño y los líderes de dichas políticas.

Posteriormente en el segundo momento bajo la metodología pruebas por pares, es decir, intercambiar las preguntas entre los líderes de política para que puedan validar la pertinencia, coherencia y trazabilidad de las preguntas, especialmente a las relacionadas con las políticas a cargo de Función Pública; si bien es cierto, los cambios introducidos a las preguntas no pueden ser mayores en virtud a que se debe garantizar la comparabilidad con los resultados de la vigencia anterior, la revisión se lleva a cabo en los siguientes aspectos:

- Comprensión, es decir, si están construidas en lenguaje claro y entendible, según el público al que va dirigido.
- Redacción de las preguntas (fraseos).
- Secuencia lógica de pregunta.

La distribución para la revisión es la siguiente:

- Dirección de Empleo Público revisará las preguntas de la Política de Integridad
- Dirección de Participación, Transparencia y Servicio al Ciudadano revisará las preguntas de Gestión del Conocimiento
- Dirección de Gestión del Conocimiento revisará las preguntas de Fortalecimiento Organizacional
- Dirección de Desarrollo Organizacional revisará las preguntas de Empleo Público

Fase I – Pruebas a las preguntas – Validación Preguntas Líderes de política. Para el desarrollo de esta, se adelantaron las siguientes acciones:

1. Se revisaron las fichas de las estructuras temáticas de cada política de la vigencia anterior y se estableció un cronograma de trabajo con los líderes de política para actualizarlas acorde a los últimos lineamientos como base para el alistamiento del cuestionario de preguntas
2. Se programaron reuniones de trabajo con dos entidades nuevas (DNP/DANE) que fueron autorizadas por el consejo asesor de gestión y desempeño institucional para incluir las preguntas de la política de mejora regulatoria y gestión de información estadística y determinar la línea base de esta dos políticas en la presente vigencia
3. Se realizaron 29 mesas de trabajo con cada uno de los líderes para justar y definir las preguntas finales y el ámbito de aplicación del formulario y establecer la confiabilidad de las preguntas (previas y nuevas) con el fin de validar las preguntas de cada una de las políticas de gestión y desempeño institucional:

Tabla 1 Listado de Mesas de Trabajo para Validación de Preguntas

líder política	No de reuniones
Empleo publico	1
Integridad	1
Planeación	1
Gestion presupuestal	4
Fortalecimiento institucional	1
Gobierno digital	2
Seguridad digital	1
Defensa jurídica	1
Mejora normativa	3
Servicio al ciudadano	1
Tramites	1
Participación ciudadana	1
Evaluación	1
Gestion del conocimiento	2
Gestion documental	1
Transparencia	1
Control interno	1
Información estadística	3
General	2
Total	29

4. Se diseñó la tabla de ámbitos de aplicación de cada política y se actualizó el directorio de responsables de cada política
5. Se revisó y validó las variables de medición de las políticas de gestión y desempeño en coordinación con los líderes de política estableciéndose 108 índices para la comparación de los cinco grupos clúster seleccionados a nivel territorial
6. Se validó las propuestas de preguntas de cada líder (duplicidades, ambigüedades, entre otros) y se consolidó el banco de preguntas y respuestas definitivo

Resultados de las pruebas – fase I

Pruebas a las preguntas – pruebas por pares. Los resultados de esta revisión se presentan a continuación:

Tabla 2 Resultado Revisión de Pares

Política	Par asignado	Conclusiones
Integridad	Dirección de Empleo Público	Luego de revisar las preguntas de integridad de acuerdo con el plan planteado por la DGD, no se encontró ninguna situación que requiera de una revisión de las preguntas, por tanto no formulamos observaciones a las mismas.
Gestión del Conocimiento	Dirección de Participación, Transparencia y Servicio al Ciudadano	Las observaciones a las preguntas se centraron en recomendar la unificación de varias preguntas ya que aparecen temas repetidos y esto ayuda a simplificar. Las observaciones se aprecian en el archivo "RE Preguntas Gestión del Conocimiento FURAG" disponible en la ruta \\yaksa\11300DGD\2019\DOCUMENTOS_APOYO\MIPG\APLICACIÓN_FURAG_2019-2\2. DISEÑO Y PRUEBAS\PRUEBAS
Empleo Público	Dirección de Desarrollo Organizacional	En las preguntas de talento humano están redactadas en lenguaje claro y la redacción y la secuencia lógica son adecuadas. Solo dos observaciones breves: En la pregunta 4) falta un artículo ("la") ante de la palabra "información". 4) La entidad cuenta con la caracterización de todos sus servidores respecto a la información registrada en sus hojas de vida, de acuerdo con el formato único de hoja de vida de Función Pública. En la pregunta 12 no se comprende la alusión final referente a la expresión "por nivel jerárquico". Queda la pregunta de si hay concurso por nivel jerárquico, y otros sin nivel jerárquico. 12) Del total de empleos de carrera administrativa con asignación presupuestal con corte al 31 de diciembre y que se encuentran en vacancia definitiva, indique el número de estos empleos que se sometieron o están sometiendo a concurso de mérito, por nivel jerárquico (p.ej. no debe haber empleos del nivel directivo, entonces, ¿por qué abrir la opción?).

Estas observaciones fueron tenidas en cuenta por los líderes de política en la validación de sus preguntas.

Como producto de esta validación se consolida en los cuestionarios definitivos que se cargan al aplicativo FURAG.

Fase II – Pruebas al aplicativo

El fin es identificar los riesgos asociados al aplicativo de captura de la información estadística, en este sentido las pruebas al aplicativo siguen la misma lógica de prueba de escritorio, pero desde la perspectiva de operatividad, diligenciamiento y claridad en la secuencia de respuesta. En este orden de ideas se maneja una matriz en formato Excel para corroborar que los formularios mostrados por el aplicativo efectivamente correspondan a cada una de las entidades objeto de medición, los condicionales

planteados según opción de respuesta, entre otras variables que permiten estandarizar y ajustar el instrumento de medición. Básicamente, estas pruebas son de carácter técnico (OTIC) y de carácter funcional (Dirección de Gestión y Desempeño de la Función Pública) para verificar aspectos tales como:

- Consistencia de los formularios: que estén completos, que solo le aparezcan las preguntas que le corresponden a cada entidad
- Parametrización de las preguntas abiertas
- Verificar que las funciones de exclusión de opciones de respuesta, obligatoriedad, campos de evidencias estén funcionando correctamente
- Accesibilidad y navegabilidad
- Pruebas de concurrencia

En este sentido es necesario resaltar que las pruebas están diseñadas para realizarse una parte desde la Oficina de Tecnologías de la Información y las Comunicaciones (OTIC), en lo que refiere a aspectos técnicos del aplicativo, y otra parte desde la Dirección de Gestión y Desempeño Institucional (DGDI), en lo que respecta a experiencia de usuario y detalles de redacción u ortografía. Sin embargo, para la medición de la vigencia 2019 se diseñaron formatos de pruebas estándar que se implementaron tanto en la OTIC como en la DGDI, dicho formato pide datos básicos de diligenciamiento como nombre de la persona que lo diligenció, ID del formulario que probó, el usuario y la clave con las cuales accedió al aplicativo FURAG, posteriormente se despliegan las siguientes columnas:

Tabla 3 Formato de pruebas al aplicativo FURAG

Columna	Descripción	Datos que acepta la variable	Tipo de datos
No.	Número de requerimientos	1, ..., n	Numérico
Fecha	Fecha del requerimiento	dd/mm/aaaa	Fecha
Tipo de incidente	Inconsistencia encontrada	Lista desplegable (Falla funcional, redacción/ortografía / aspectos de forma / filtro padre no funciona)	Carácter
Descripción	Especificaciones de la inconsistencia detectada	Abierto texto	Carácter
Imagen	Acercamiento visual de la inconsistencia	NA	Imagen
Propuesta (opcional)	En caso de que se pueda proponer alternativa se usa este espacio para detallar la alternativa	Abierto texto	Carácter
Resultado de la observación	Luego de revisarse la observación se procede a detallar si se hizo o no y las respectivas razones	Abierto texto	Carácter

El formato diseñado mantiene la misma estructura a los que usa la OTIC lo cual facilita la interpretación para implementar los ajustes dentro del aplicativo como resultado de las pruebas llevadas a cabo. Al final, luego de hacer las pruebas, en el formato de aceptación de software se procede a hacer la entrega del aplicativo para poner en funcionamiento, este formato es diligenciado por los directores de cada área. En este orden de ideas, con la entrega finaliza el proceso de pruebas al aplicativo FURAG, toda vez que, al pasar los diseños y las pruebas, estaría listo para poner en funcionamiento. Adjunto al documento se anexan los archivos que permiten evidenciar las pruebas llevadas a cabo al aplicativo FURAG.

Resultado de las pruebas – fase II

Se compilan los formatos de pruebas realizadas y se transmiten a la Oficina de Tecnologías de la Información y las Comunicaciones (OTIC) para que se realicen los respectivos ajustes necesarios dentro del aplicativo FURAG en caso de ser requerimientos funcionales o técnicos, al ser detalles de ortografía o redacción se procede desde la misma Dirección de Gestión y Desempeño Institucional a hacer los cambios. Al final en el formato de entrega del aplicativo y en los correos entre áreas se evidencian los cambios pedidos y ajustados en el aplicativo FURAG.

Fase II – Pruebas al aplicativo

Durante los días 7, 8, 12 y 13 de noviembre se llevaron a cabo las pruebas funcionales al Aplicativo, por parte de los servidores de la Dirección de Gestión y Desempeño, para la puesta en producción del mismo en el mes de noviembre (primera Fase de aplicación, vigencia 2019), de acuerdo con la distribución de los formularios; los reportes de las pruebas se hicieron en el formato presentado en la tabla 3, y las incidencias técnicas y funcionales se fueron escalando a la Oficina de las Tecnologías y las Comunicaciones, las cuales fueron subsanadas en su totalidad. En el Archivo 2019-11-15_Consolidado_incidentes_pruebas_aplicativo_furag se encuentran los reportes de incidentes, el cual está disponible en la ruta \\Yaksa\11300dgd\2020\DOCUMENTOS_APOYO\MIPG\MEDICION_DESEMPENO_INSTITUCIONAL\DISEÑO-PRUEBAS_CAP_6\PRUEBAS\PRUEBAS_APLICATIVO.

En reunión del día 18 de noviembre, se suscribe el acta de entrega del aplicativo con lo cual se formaliza su total disponibilidad para su puesta en producción para la medición de la gestión y el desempeño en las entidades territoriales, vigencia 2019. El Acta de aceptación se encuentra disponible en la ruta \\Yaksa\11300dgd\2020\DOCUMENTOS_APOYO\MIPG\MEDICION_DESEMPENO_INSTITUCIONAL\DISEÑO-PRUEBAS_CAP_6\PRUEBAS\PRUEBAS_APLICATIVO.

De otra parte, y con el fin de garantizar que para la puesta en producción del Aplicativo en el mes de febrero (primera Fase de aplicación, vigencia 2019); de igual manera, los reportes de las pruebas se hicieron en el formato presentado en la tabla 3, y las incidencias técnicas y funcionales se fueron escalando a la Oficina de las Tecnologías y las Comunicaciones, las cuales fueron subsanadas en su totalidad. Los archivos los reportes de incidentes, se encuentran disponibles en la ruta \\Yaksa\11300dgd\2020\DOCUMENTOS_APOYO\MIPG\MEDICION_DESEMPENO_INSTITUCIONAL\DISEÑO-PRUEBAS_CAP_6\PRUEBAS\PRUEBAS_APLICATIVO.

Fase III – Pruebas a las visualizaciones con los resultados de la medición

Los resultados de la Medición del Desempeño Institucional se difunden por medio de la herramienta Power BI, debido entre otras cosas, a la calidad de gráficos y las facilidades que ofrece para presentar grandes cantidades de información de manera simple y práctica. En este orden de ideas en la estrategia de difusión se detalla la forma en que se construye el reporte interactivo, así como en el flujograma transversal de tablas y códigos de la Medición del Desempeño Institucional que ilustra los insumos necesarios para construir el reporte, sin embargo, desde la Dirección de Gestión y Desempeño Institucional se diseñan pruebas al reporte como control y calidad de los resultados de la medición. Es de esta forma como el formato que se emplea para pruebas al aplicativo es similar al usado en las

pruebas a las visualizaciones, toda vez que pide el nombre de la persona que hace las pruebas y posteriormente despliega la siguiente información:

Tabla 4 Formato de prueba a las visualizaciones

Columna	Descripción	Datos que acepta la variable	Tipo de datos
No.	Número de requerimientos	1, ..., n	Numérico
Fecha	Fecha del requerimiento	dd/mm/aaaa	Fecha
Sección	Lugar en el cual se encontró la inconsistencia.	Abierto texto	Carácter
Tipo observación	Identificación de la inconsistencia	Lista desplegable (Funcionalidad del sitio, textos, recomendación política, otras.)	Carácter
Descripción	Especificaciones de la inconsistencia detectada	Abierto texto	Carácter
Imagen	Acercamiento visual de la inconsistencia	NA	Imagen
Propuesta	Este campo permite detallar la alternativa encontrada a la inconsistencia.	Abierto texto	Carácter
Resultado de la observación	Luego de revisarse la observación se procede a detallar si se hizo o no el ajuste y las respectivas razones si hay lugar a ellas.	Abierto texto	Carácter

Resultados de las pruebas – fase III

Esta etapa está diseñada para que posterior a las pruebas se socialice con la Dirección de Gestión y Desempeño Institucional los ajustes, para proceder con la aprobación y posterior publicación en el micrositio MIPG en lo que respecta a los resultados de la medición. En la ruta \\Yaksa\11300dgd\2020\DOCUMENTOS_APOYO\MIPG\MEDICION_DESEMPEÑO_INSTITUCIONAL\DISEÑO-PRUEBAS_CAP_6\PRUEBAS\PRUEBAS_VISUALIZACIONES en formato Excel con las pruebas consolidadas y el correo con las actualizaciones a las visualizaciones posterior a las pruebas realizadas.

II. Documento técnico Validación de tablas y generación de fichas temporales de validación

Estructura de validación

El objetivo de la máquina en R es automatizar el proceso de validación, según la estructura de carpetas descrita a continuación:

1. input
2. mcr
3. doc
4. rmd
5. scr

A continuación, se describen las carpetas que entran en el proceso. Posteriormente se describe el proceso.

Input

Dentro de la carpeta Input, se establece un tablero de control en formato .xlsx, denominado MotorValida (la primera hoja de dicho archivo), que contiene en fila por fila, las validaciones definidas por el equipo. Entre las columnas del archivo de control, se encuentran, entre otras, las siguientes:

NOMBRE_FUNCION	PARAMETROS	ID_ACCION	DESCRIPCION_DEFINICION	Proceso
V01.NoNulo	MatrizXC; CodP	F001V04010	No acepta valores nulos	C06
V02.NoDupl	MatrizXC; CodP	F002V04011	No acepta valores duplicados	C06

La columna NOMBRE_FUNCION contiene los nombres de la función que se debe ejecutar. La columna PARAMETROS contiene los parámetros predefinidos en la función, con los que se debe ejecutar dicha función. ID_ACCION hace referencia a un identificador único de cada una de las validaciones. DESCRIPCION_DEFINICION contiene una pequeña definición de la validación, y finalmente, la columna Proceso asocia cada validación a su lugar en el proceso, relacionándola directamente con los nombres de los códigos. Por ejemplo, la fila 1 del cuadro muestra que se valida la tabla MatrizXC, verificando que la columna CodP no debe contener valores nulos, en el punto del proceso C06.

La fila 2 muestra que se valida la tabla MatrizXC, verificando que la columna CodP no debe contener valores duplicados, también en el punto del proceso C06. En el capítulo siguiente se dará una descripción detallada de esta tabla.

Adicionalmente, dentro de la carpeta input se incluyen los archivos que van a entrar en la validación. Por ejemplo, en el caso del cuadro de referencia, la tabla MatrizXC que sirve de insumo para el proceso C06, debe estar incluida en esta carpeta; de otra forma, la validación no puede ser ejecutada.

Mcr

Dentro de la carpeta en mención, se incluyen todos los códigos con extensión R que entran en la validación. Estos códigos se dividen en dos: los que usan una sola tabla y los que usan dos tablas. Los códigos que usan solamente una tabla son:

1. V01.NoNulo
2. V02.NoDupl
3. V02.NoDuplW
4. V03.LargoE
5. V04.ValEsp

Los códigos que definen las validaciones con dos o más tablas, son:

6. C01.tConD
7. C01.tConT
8. C02.tConN
9. C03.nConF

En el capítulo siguiente se dará una descripción detallada de estos códigos genéricos que responden a las necesidades de validación detectadas por el equipo.

Doc

En la carpeta “doc” se almacenan las versiones de corridas de validación necesarias. Estas se versionan con la hora y la fecha de cada una de las corridas. En el capítulo siguiente se dará una descripción detallada de esta tabla.

Rmd

En esta carpeta se construye un código con extensión .Rmd, que hace referencia a un archivo de control que produce posteriormente un .docx. Este último, se alimenta de la última corrida de validación encontrada en la carpeta “doc”, y reporta de manera genérica el resultado de las validaciones, separadas por tipo: si son de las que usa una tabla o dos. Estos reciben el nombre de CCVC01, por ejemplo, para el caso en que se esté validando el proceso C01. Se describe posteriormente el archivo.

Proceso de validación: scr

El proceso de validación está compuesto por dos códigos, que se encuentra en la carpeta “scr”:

1. 01_ValidaGeneral.R
2. 02_ValidaFicha.R

El primero de estos, corre las validaciones y construye el insumo en la carpeta “doc” así:

1. Guarda la hora y fecha de la corrida.



2. Hace una copia del archivo de control (de nuevo en formato xlsx) en la carpeta “doc” y lo renombra con la fecha y hora.
3. Lee el archivo de control actualizado
4. Lee las tablas que va a validar
5. Ejecuta fila a fila las validaciones de una tabla, asignadas a ese proceso.
6. Exporta las validaciones en una pestaña aparte, adicionando una columna que relaciona las cantidades de registros a revisar. En caso de encontrar inconsistencias, las exporta en una pestaña distinta que renombra individualmente con el código único de la validación.
7. Ejecuta fila a fila las validaciones de dos tablas, asignadas a ese proceso.
8. Exporta las validaciones en una pestaña aparte, adicionando una columna que relaciona las cantidades de registros a revisar. En caso de encontrar inconsistencias, las exporta en una pestaña distinta que renombra individualmente con el código único de la validación.

El segundo código se alimenta de este insumo final, para generar la ficha. Lo hace de la siguiente manera, generalmente:

1. Llama la matriz que sirve de insumo de la carpeta “doc”.
2. Lee solamente los resultados generales de validación.
3. Genera en la carpeta .Rmd un archivo de control, llamado de acuerdo al proceso, por ejemplo C006 si está en dicho proceso.
4. Ejecuta el código, y genera un .docx, que es la ficha reportada al equipo.

El objetivo de la ficha, denominada Ficha Temporal de Validación (CCVC01.docx por ejemplo en el caso de estar validando el proceso C01), es tener consolidado un documento que reporte ciertas alertas en cada proceso, y a su vez, una persona responsable de verificar si se puede seguir con el proceso, o se debe corregir algún insumo. La presencia de una alerta, no implica que el proceso debe parar y que se deban corregir ciertos insumos, sino que pone puntos de control para la revisión de estos. En caso que las alertas sean explicadas y entendidas, el proceso puede continuar.

Archivos y Tablas

Esta parte del documento, describe exhaustivamente las tablas generales que entran en el proceso general de la validación. Se debe entender que hay tres “archivos” (documentos que contienen tablas o fichas de salidas) en todo este proceso:

1. MotorValida.xlsx
2. MotorValidaAAMMDD_HHMM.xlsx
3. CCVPPP.docx

Los valores subrayados, son valores dinámicos, que se fijan de acuerdo a los siguientes parámetros:

- a. AA: Hace referencia a los dos últimos dígitos del año
- b. MM: Hace referencia a los dos dígitos del mes, y puede tomar valores desde 01 hasta 12
- c. DD: Hace referencia a los dos dígitos del día, y puede tomar valores desde 01 hasta 31
- d. PPP: Hace referencia al proceso, y puede tomar valores desde C01 hasta C15.

En principio, hay que entender que el código para versionar el proceso MotorValidaAAMMDD_HHMM.xlsx, es el archivo de trabajo, sobre el cual se depositan todas las corridas de validaciones. Este simplemente es copiado desde la carpeta “input”, y versionado en la carpeta “doc”, y leído a la consola, con el fin de extraer de él las validaciones a realizar y con el fin de depositar en él los resultados de las validaciones. Por esto, cada uno de ellos dos, difiere en las hojas que lo conforman, pero concuerdan en al menos una: qMVal.

Archivo MotorValida

Este archivo no es leído ni modificado por el proceso. Sirve como tablero de control para dirigir y definir las validaciones. En él, solamente se define una hoja de cálculo.

Tabla qMVal

Descripción: Esta tabla relaciona cada una de las validaciones dispuestas para cada proceso. Funciona como tablero de control, y es allí donde se definen las validaciones que posteriormente se ejecutarán. Es la única hoja del archivo “**MotorValida.xlsx**”.

Clasificación dentro del proceso: Input

Formato: .xlsx

Número de filas: 109 (este es el valor total, pero depende de cada proceso)

Número de columnas: 14

Input de: Todos los procesos de validación.

Output de: Ninguno

Lista Variables

Variable	Tipo de datos	Descripción
CONS_VALIDA	Carácter	Es un número consecutivo de largo 3, que identifica la posición de cada validación en el documento.
CODIGO_ARCHIVO	Carácter	Codificación de cada archivo que entra en la operación de la validación. El primer carácter es una letra, seguido de dos números. Identifica cada archivo.
CODIGO_TIPARCH	Carácter	Toma el valor de 1. Consiste en definir el tipo de archivo: si es una tabla (1) o si es otro tipo, por ejemplo, una imagen o un código. En este caso solo aplica tipo 1: tabla de texto.
NOMBRE_ARCHIVO	Carácter	El nombre del archivo o archivos que entran en la validación.
DESCRIPCION_ARCHIVO	Carácter	Una breve descripción del archivo, su contenido, funcionalidad, etc.
NOMBRE_CAMPO	Carácter	Los campos de las tablas que entran a ser validados.
DESCRIPCION_CAMPO	Carácter	Una breve descripción del campo, su funcionalidad, etc.
ID_FUNCION	Carácter	Un identificador interno de la función, referente solamente a este proceso.
NOMBRE_FUNCION	Carácter	El nombre de la función como tal, como fue construida y como debe ser invocada.
PARAMETROS	Carácter	Los parámetros necesarios por parte de la función, para ejecutarla. En general el grupo de parámetros son dos: tabla (o tablas) y campo (o campos)
ID_ACCION	Carácter	Es la concatenación del identificador de la función, el código del archivo y el consecutivo de la validación: ID_FUNCION + CODIGO_ARCHIVO + CONS_VALIDA



Variable	Tipo de datos	Descripción
USO	Carácter	Es una variable indicadora, que toma el valor de 1 si se quiere ejecutar dicha validación, o 0 si no.
DESCRIPCION_DEFINICION	Carácter	Es una definición de la validación como tal. Está en términos genéricos del proceso.
Proceso	Carácter	El código del proceso o código al que pertenece la validación.

Archivo MotorValidaAAMMDD_HHMM

Este archivo es inicialmente, una copia de MotorValida.xlsx., ubicada en la carpeta “doc”. Este sirve para comunicar las validaciones definidas con el código principal, además que sirve como retorno e instrumento de depósito de las validaciones. Se relacionan sus hojas finales.

qMVal

Descripción: Esta tabla relaciona cada una de las validaciones dispuestas para cada proceso. Funciona como tablero de control, y es allí donde se están definidas las validaciones que se ejecutarán. Es la hoja 1 del archivo “MotorValidaAAMMDD_HHMM.xlsx”.

Clasificación dentro del proceso: doc

Formato: .xlsx

Número de filas: 109 (este es el valor total, pero depende de cada proceso)

Número de columnas: 14

Input de: Todos los procesos de validación.

Output de: Ninguno

Lista Variables

Variable	Tipo de datos	Descripción
CONS_VALIDA	Carácter	Es un número consecutivo de largo 3, que identifica la posición de cada validación en el documento.
CODIGO_ARCHIVO	Carácter	Codificación de cada archivo que entra en la operación de la validación. El primer carácter es una letra, seguido de dos números. Identifica cada archivo.
CODIGO_TIPARCH	Carácter	Toma el valor de 1. Consiste en definir el tipo de archivo: si es una tabla (1) o si es otro tipo, por ejemplo, una imagen o un código. En este caso solo aplica tipo 1: tabla de texto.
NOMBRE_ARCHIVO	Carácter	El nombre del archivo o archivos que entran en la validación.
DESCRIPCION_ARCHIVO	Carácter	Una breve descripción del archivo, su contenido, funcionalidad, etc.
NOMBRE_CAMPO	Carácter	Los campos de las tablas que entran a ser validados.
DESCRIPCION_CAMPO	Carácter	Una breve descripción del campo, su funcionalidad, etc.
ID_FUNCION	Carácter	Un identificador interno de la función, referente solamente a este proceso.
NOMBRE_FUNCION	Carácter	El nombre de la función como tal, como fue construida y como debe ser invocada.
PARAMETROS	Carácter	Los parámetros necesarios por parte de la función, para ejecutarla. En general el grupo de parámetros son dos: tabla (o tablas) y campo (o campos)



Variable	Tipo de datos	Descripción
ID_ACCION	Carácter	Es la concatenación del identificador de la función, el código del archivo y el consecutivo de la validación: ID_FUNCION + CODIGO_ARCHIVO + CONS_VALIDA
USO	Carácter	Es una variable indicadora, que toma el valor de 1 si se quiere ejecutar dicha validación, o 0 si no.
DESCRIPCION_DEFINICION	Carácter	Es una definición de la validación como tal. Está en términos genéricos del proceso.
Proceso	Carácter	El código del proceso o código al que pertenece la validación.

Corrida_Casos_V

Descripción: Esta tabla relaciona solamente las validaciones tipo V dispuestas para el proceso en cuestión:

1. V01.NoNulo
2. V02.NoDupl
3. V02.NoDuplW
4. V03.LargoE
5. V04.ValEsp

Allí se adiciona una columna llamada "REGISTROS_INCONSISTENTES". En esta pestaña se puede verificar en qué validación saltaron alertas, sólo en los procesos de validación en que se incluye una tabla.

Clasificación dentro del proceso: doc

Formato: .xlsx

Número de filas: depende de cada proceso.

Número de columnas: 15

Input de: Ninguno

Output de: Corrida de los procesos de validación que involucran una sola tabla.

Lista Variables

Variable	Tipo de datos	Descripción
CONS_VALIDA	Carácter	Es un número consecutivo de largo 3, que identifica la posición de cada validación en el documento.
CODIGO_ARCHIVO	Carácter	Codificación de cada archivo que entra en la operación de la validación. El primer carácter es una letra, seguido de dos números. Identifica cada archivo.
CODIGO_TIPARCH	Carácter	Toma el valor de 1. Consiste en definir el tipo de archivo: si es una tabla (1) o si es otro tipo, por ejemplo, una imagen o un código. En este caso solo aplica tipo 1: tabla de texto.
NOMBRE_ARCHIVO	Carácter	El nombre del archivo o archivos que entran en la validación.
DESCRIPCION_ARCHIVO	Carácter	Una breve descripción del archivo, su contenido, funcionalidad, etc.



Variable	Tipo de datos	Descripción
NOMBRE_CAMPO	Carácter	Los campos de las tablas que entran a ser validados.
DESCRIPCION_CAMPO	Carácter	Una breve descripción del campo, su funcionalidad, etc.
ID_FUNCION	Carácter	Un identificador interno de la función, referente solamente a este proceso.
NOMBRE_FUNCION	Carácter	El nombre de la función como tal, como fue construida y como debe ser invocada.
PARAMETROS	Carácter	Los parámetros necesarios por parte de la función, para ejecutarla. En general el grupo de parámetros son dos: tabla (o tablas) y campo (o campos)
ID_ACCION	Carácter	Es la concatenación del identificador de la función, el código del archivo y el consecutivo de la validación: ID_FUNCION + CODIGO_ARCHIVO + CONS_VALIDA
USO	Carácter	Es una variable indicadora, que toma el valor de 1 si se quiere ejecutar dicha validación, o 0 si no.
DESCRIPCION_DEFINICION	Carácter	Es una definición de la validación como tal. Está en términos genéricos del proceso.
Proceso	Carácter	El código del proceso o código al que pertenece la validación.
REGISTROS_INCONSISTENTES	Numérico	Cuenta el número de registros en la tabla que se está validando, que no cumplen con la especificación de consistencia que se solicita.

Corrida_Casos_C

Descripción: Esta tabla relaciona solamente las validaciones tipo C dispuestas para el proceso en cuestión:

1. C01.tConD
2. C01.tConT
3. C02.tConN
4. C03.nConF

Allí se adiciona una columna llamada "REGISTROS_INCONSISTENTES". En esta pestaña se puede verificar en qué validación saltaron alertas, sólo en los procesos de validación en que se incluyen dos tablas.

Clasificación dentro del proceso: doc

Formato: .xlsx

Número de filas: depende de cada proceso.

Número de columnas: 15

Input de: Ninguno

Output de: Corrida de los procesos de validación que involucran dos tablas.

Lista Variables



Variable	Tipo de datos	Descripción
CONS_VALIDA	Carácter	Es un número consecutivo de largo 3, que identifica la posición de cada validación en el documento.
CODIGO_ARCHIVO	Carácter	Codificación de cada archivo que entra en la operación de la validación. El primer carácter es una letra, seguido de dos números. Identifica cada archivo.
CODIGO_TIPARCH	Carácter	Toma el valor de 1. Consiste en definir el tipo de archivo: si es una tabla (1) o si es otro tipo, por ejemplo, una imagen o un código. En este caso solo aplica tipo 1: tabla de texto.
NOMBRE_ARCHIVO	Carácter	El nombre del archivo o archivos que entran en la validación.
DESCRIPCION_ARCHIVO	Carácter	Una breve descripción del archivo, su contenido, funcionalidad, etc.
NOMBRE_CAMPO	Carácter	Los campos de las tablas que entran a ser validados.
DESCRIPCION_CAMPO	Carácter	Una breve descripción del campo, su funcionalidad, etc.
ID_FUNCION	Carácter	Un identificador interno de la función, referente solamente a este proceso.
NOMBRE_FUNCION	Carácter	El nombre de la función como tal, como fue construida y como debe ser invocada.
PARAMETROS	Carácter	Los parámetros necesarios por parte de la función, para ejecutarla. En general el grupo de parámetros son dos: tabla (o tablas) y campo (o campos)
ID_ACCION	Carácter	Es la concatenación del identificador de la función, el código del archivo y el consecutivo de la validación: ID_FUNCION + CODIGO_ARCHIVO + CONS_VALIDA
USO	Carácter	Es una variable indicadora, que toma el valor de 1 si se quiere ejecutar dicha validación, o 0 si no.
DESCRIPCION_DEFINICION	Carácter	Es una definición de la validación como tal. Está en términos genéricos del proceso.
Proceso	Carácter	El código del proceso o código al que pertenece la validación.
REGISTROS_INCONSISTENTES	Númérico	Cuenta el número de registros en la tabla que se está validando, que no cumplen con la especificación de consistencia que se solicita.

V_ID_ACCION

Descripción: Si alguna validación de la hoja Corrida_Casos_V, refleja que hay uno o más registros que revisar puntualmente, según las validaciones definidas, se generará una nueva hoja con el nombre que identifica la validación: V_ + ID_ACCION. Por ejemplo, si se tiene en Corrida_Casos_V:

NOMBRE_FUNCION	PARAMETROS	ID_ACCION	USO	DESCRIPCION_DEFINICION	Proceso	REGISTROS_INCONSISTENTES
V02.NoDupl	anclaW1; item	F002V14 046	1	No acepta valores duplicados	C12	2512

Entonces, en el mismo archivo o libro, se generará una pesaña con el código V_ F002V14046, que contendrá los 2512 registros que saltan como alerta.

Clasificación dentro del proceso: doc

Formato: .xlsx

Número de filas: Depende de cada validación que alerta el proceso.

Número de columnas: Depende de la tabla validada.

Input de: Ninguno

Output de: Corrida de los procesos de validación que involucra una tabla.

Lista Variables: Depende de la tabla validada sobre la que se alerta el proceso.

C_ID_ACCION

Descripción: Si alguna validación de la hoja Corrida_Casos_C, refleja que hay uno o más registros que revisar puntualmente, según las validaciones definidas, se generará una nueva hoja con el nombre que identifica la validación: C_ + ID_ACCION. Por ejemplo, si se tiene en Corrida_Casos_C:

NOMBRE_FUNCION	PARAMETROS	ID_ACCION	USO	DESCRIPCION_DEFINICION	Proceso	REGISTROS_INCONSISTENTES
C01.tConD	dfw01 ; anclaW1; item	C097V13 042	1	Nombres de Var1 incluidos en columna Var3 de tabla Var2	C12	1112

Entonces, en el mismo archivo o libro, se generará una pesaña con el código C_C097V13042, que contendrá los 1112 registros que saltan como alerta.

Clasificación dentro del proceso: doc

Formato: .xlsx

Número de filas: Depende de cada validación que alerta el proceso.

Número de columnas: Depende de la tabla validada.

Input de: Ninguno

Output de: Corrida de los procesos de validación que involucran dos tablas.

Lista Variables: Depende de la tabla validada sobre la que se alerta el proceso.

Archivo CCVPPP

Descripción: Los parámetros PPP corresponden a valores que identifican el proceso: C01 hasta C15. Este archivo corresponde a un .docx. Contiene la hora y fecha de la validación. Además, contiene la información resumida de las corridas, denominadas tipo C y tipo V. Este documento constituye la pieza fundamental de validación en el proceso, pues es aquí donde se consigna si las alertas arrojadas fueron revisadas, y es donde se pone el nombre del responsable, para proseguir en el procesamiento.

Clasificación dentro del proceso: rmd

Formato: .docx

Número de hojas: Depende de cada validación que alerta el proceso, pero en general, son 4: la descripción del proceso, la corrida tipo V, la corrida tipo C y la página que relaciona la información de quien valida y permite seguir en el proceso.

Input de: Ninguno

Output de: CCVPPP.Rmd.

Códigos

Para el procesamiento de las validaciones en el marco de la medición del desempeño se utilizan dos lenguajes de programación, R y Markdown, este último integrado a R por RStudio. La estructura para documentar los códigos es similar a la empleada en las tablas anteriormente descritas.

V01.NoNulo

Es una función que busca valores nulos en una variable o campo de una tabla.

Lenguaje: R

Número de líneas: 27

Tablas de entrada: Cualquiera que disponga el administrador de validaciones

Tablas de salida: No aplica. Genera una tabla en R de la misma estructura que la entrada, únicamente con los registros que saltan a la validación.

Tiempo estimado de ejecución: Depende de la tabla de entrada. Promedio 5 segundos.

Descripción del algoritmo:

Paso	Descripción
1. Marcar posición de filas en la tabla.	Genera una columna adicional a la tabla a la que está atribuida la operación, para marcar la posición de cada registro: un consecutivo de 1 hasta el número de filas.
2. Genera un objeto, subconjunto de la tabla de entrada.	Genera un subconjunto de la tabla (parámetro <i>Tabla1</i> en el código), que consiste en los registros de la tabla de entrada en donde los valores en la columna de interés (parámetro <i>Var1</i> en el código), toma los valores vacíos o NA.
3. Retornar el objeto	Retorna el objeto; subconjunto de la tabla (parámetro <i>Tabla1</i> en el código), que consiste en los registros de la tabla de entrada en donde los valores en la columna de interés (parámetro <i>Var1</i> en el código), toma los valores vacíos o NA.

V02.NoDupl

Es una función que busca valores duplicados en una variable o campo de una tabla.

Lenguaje: R

Número de líneas: 31

Tablas de entrada: Cualquiera que disponga el administrador de validaciones

Tablas de salida: No aplica. Genera una tabla en R de la misma estructura que la entrada, únicamente con los registros que saltan a la validación.

Tiempo estimado de ejecución: Depende de la tabla de entrada. Promedio 5 segundos.

Descripción del algoritmo:

Paso	Descripción
1. Marcar posición de filas en la tabla.	Genera una columna adicional a la tabla a la que está atribuida la operación, para marcar la posición de cada registro: un consecutivo de 1 hasta el número de filas.
2. Genera un objeto, agregando por la variable de interés el número de registros.	Genera un objeto con base en la tabla (parámetro Tabla1 en el código), que consiste en un conteo de los registros por niveles de la columna de interés (parámetro Var1 en el código) de la tabla de entrada.
3. Genera un subconjunto de aquellos casos en que el número de registros es mayor que 1.	Del objeto anterior, genera un subconjunto que consiste de los casos cuyo conteo sea mayor a 1. Posterior, hace el subconjunto de la tabla principal (Tabla 1), con base en el subconjunto.
4. Retornar el objeto	Retorna el objeto; subconjunto de la tabla (parámetro Tabla1 en el código), que consiste en los registros de la tabla de entrada en donde los valores en la columna de interés (parámetro Var1 en el código), está duplicados.

V02.NoDupIW

Es una función que busca valores duplicados en dos variables o campos de una tabla.

Lenguaje: R

Número de líneas: 32

Tablas de entrada: Cualquiera que disponga el administrador de validaciones

Tablas de salida: No aplica. Genera una tabla en R de la misma estructura que la entrada, únicamente con los registros que saltan a la validación.

Tiempo estimado de ejecución: Depende de la tabla de entrada. Promedio 5 segundos.

Descripción del algoritmo:

Paso	Descripción
1. Marcar posición de filas en la tabla.	Genera una columna adicional a la tabla a la que está atribuida la operación, para marcar la posición de cada registro: un consecutivo de 1 hasta el número de filas.
2. Genera un objeto, agregando por las variables de interés el número de registros.	Genera un objeto con base en la tabla (parámetro Tabla1 en el código), que consiste en un conteo de los registros por niveles de las columnas de interés (parámetros Var1 y Var2 en el código) de la tabla de entrada.
3. Genera un subconjunto de aquellos casos en que el número de registros es mayor que 1.	Del objeto anterior, genera un subconjunto que consiste de los casos cuyo conteo sea mayor a 1. Posterior, hace el subconjunto de la tabla principal (Tabla 1), con base en el subconjunto.

4. Retornar el objeto	Retorna el objeto; subconjunto de la tabla (parámetro Tabla1 en el código), que consiste en los registros de la tabla de entrada en donde los valores en las columnas de interés (parámetro Var1 y Var2 en el código), está duplicados.
-----------------------	---

V03.LargoE

Es una función que cuantifica el largo de una variable o campo de una tabla y extrae los que sean diferentes del parámetro esperado.

Lenguaje: R

Número de líneas: 27

Tablas de entrada: Cualquiera que disponga el administrador de validaciones

Tablas de salida: No aplica. Genera una tabla en R de la misma estructura que la entrada, únicamente con los registros que saltan a la validación.

Tiempo estimado de ejecución: Depende de la tabla de entrada. Promedio 5 segundos.

Descripción del algoritmo:

Paso	Descripción
1. Marcar posición de filas en la tabla.	Genera una columna adicional a la tabla a la que está atribuida la operación, para marcar la posición de cada registro: un consecutivo de 1 hasta el número de filas.
2. Genera un subconjunto de aquellos casos en que el ancho de los valores de la columna no sean los esperados.	Del objeto (Tabla 1), genera un subconjunto que consiste de los casos cuyo ancho de caracteres en la columna deseada (Var1) es diferente al esperado (Param).
4. Retornar el objeto	Retorna el objeto; subconjunto que consiste de los casos cuyo ancho de caracteres en la columna deseada (Var1) es diferente al esperado (Param).

V04.ValEsp

Es una función que extrae los registros que tienen valores observados diferentes de lo esperado según el parámetro.

Lenguaje: R

Número de líneas: 41

Tablas de entrada: Cualquiera que disponga el administrador de validaciones

Tablas de salida: No aplica. Genera una tabla en R de la misma estructura que la entrada, únicamente con los registros que saltan a la validación.

Tiempo estimado de ejecución: Depende de la tabla de entrada. Promedio 5 segundos.

Descripción del algoritmo:

Paso	Descripción
1. Marcar posición de filas en la tabla.	Genera una columna adicional a la tabla a la que está atribuida la operación, para marcar la posición de cada registro: un consecutivo de 1 hasta el número de filas.
2. Genera un subconjunto de aquellos casos en que el valor de entrada en la variable de interés no es el esperado.	Del objeto (Tabla 1), genera un subconjunto que consiste de los casos cuyo valor tomado en la columna deseada (Var1) es diferente al esperado (Param).
3. Retornar el objeto	Retorna el objeto; subconjunto que consiste de los casos cuyo valor tomado en la columna deseada (Var1) es diferente al esperado (Param).

C01.tConD

Es una función que selecciona de los nombres de la tabla de interés, aquellos que no están contenidas en una lista deseada, particularmente la columna de otra tabla de referencia.

Lenguaje: R

Número de líneas: 29

Tablas de entrada: Cualquiera que disponga el administrador de validaciones

Tablas de salida: No aplica. Genera una tabla en R de la misma estructura que la entrada, únicamente con los registros que saltan a la validación.

Tiempo estimado de ejecución: Depende de la tabla de entrada. Promedio 5 segundos.

Descripción del algoritmo:

Paso	Descripción
1. Seleccionar de los nombres de la tabla de interés, los valores que no están en la lista deseada.	Se debe generar un objeto que contenga los nombres de la tabla de interés (Tabla 1). De dicho objeto, se deben seleccionar aquellos que no están incluidos en la columna (Var1) definida en la tabla deseada (Tabla 2).
2. Retornar el objeto	Retorna el objeto anterior.

C01.tConT

Es una función que selecciona de una tabla de interés los casos de la variable deseada que no estén incluidos en la misma variable de otra tabla de referencia.

Lenguaje: R

Número de líneas: 29

Tablas de entrada: Cualquiera que disponga el administrador de validaciones

Tablas de salida: No aplica. Genera una tabla en R de la misma estructura que la entrada, únicamente con los registros que saltan a la validación.

Tiempo estimado de ejecución: Depende de la tabla de entrada. Promedio 5 segundos.

Descripción del algoritmo:

Paso	Descripción
1. Seleccionar de la tabla de interés los casos de la variable deseada que no estén incluidos en la columna que lleva el mismo nombre en la otra tabla de referencia.	Se genera un subconjunto de la tabla de interés (Tabla 1) que consiste en los casos en que los valores la variable deseada (Var1) no estén incluidos en la columna que lleva el mismo nombre (Var1) en la otra tabla de referencia (Tabla 2).
2. Retornar el objeto	Retorna el objeto anterior: subconjunto de la tabla de interés (Tabla 1).

C01.tConN

Lenguaje: R

Número de líneas: 29

Tablas de entrada: Cualquiera que disponga el administrador de validaciones

Tablas de salida: No aplica. Genera una tabla en R de la misma estructura que la entrada, únicamente con los registros que saltan a la validación.

Tiempo estimado de ejecución: Depende de la tabla de entrada. Promedio 5 segundos.

Descripción del algoritmo:

Paso	Descripción
1. Seleccionar de la tabla de interés los casos de la variable deseada que no estén incluidos en la columna de referencia en la tabla de referencia.	Se genera un subconjunto de la tabla de interés (Tabla 1) que consiste en los casos en que los valores la variable deseada (Var1) no estén incluidos en la columna de referencia (Var2) en la tabla de referencia (Tabla 2).
2. Retornar el objeto	Retorna el objeto anterior: subconjunto de la tabla de interés (Tabla 1).

C01.tConF

Seleccionar de la tabla de interés los nombres de las variables que no estén incluidos en los nombres de las variables de una tabla de referencia.

Lenguaje: R

Número de líneas: 29

Tablas de entrada: Cualquiera que disponga el administrador de validaciones

Tablas de salida: No aplica. Genera una tabla en R de la misma estructura que la entrada, únicamente con los registros que saltan a la validación.

Tiempo estimado de ejecución: Depende de la tabla de entrada. Promedio 5 segundos.

Descripción del algoritmo:

Paso	Descripción
1. Seleccionar de los nombres de la tabla de interés, los nombres que no están en la lista de nombres de la tabla de referencia.	Se debe generar un objeto que contenga los nombres de la tabla de interés (Tabla 1). De dicho objeto, se deben seleccionar aquellos que no están incluidos en la lista de nombres de la tabla de referencia (Tabla 2).
2. Retornar el objeto	Retorna el objeto anterior.

01_ValidaGeneral

Es el código más importante, porque traduce las validaciones estipuladas lógicamente en el tablero de control en ejecuciones por medio de todos los códigos explicados anteriormente.

Lenguaje: R

Número de líneas: 438

Tablas de entrada: Cualquiera que disponga el administrador de validaciones, el MotorValida (qMVal).

Tablas de salida: MotorValidaAAMMDD_HHMM.

Tiempo estimado de ejecución: Depende de las tablas de entrada. Promedio 120 segundos.

Descripción del algoritmo:

Paso	Descripción
1. Fecha y Hora	Guarda la hora y fecha de la corrida.
2. Extraer una copia del archivo de control	Hace una copia del archivo de control (de nuevo en formato xlsx) en la carpeta "doc" y lo renombra con la fecha y hora
3. Lee el archivo de control	Lee o carga en la consola de trabajo el archivo de control actualizado
4. Carga las tablas de validación	Lee las tablas que va a validar, dispuestas en el input.
5. Ejecuta procesos Tipo V	Ejecuta fila a fila las validaciones de una tabla, asignadas a ese proceso.
6. Exporta las validaciones tipo V.	Exporta las validaciones en una pestaña aparte, adicionando una columna que relaciona las cantidades de registros a revisar. En caso de encontrar inconsistencias, las exporta en una pestaña distinta que renombra individualmente con el código único de la validación
7. Ejecuta procesos Tipo C	Ejecuta fila a fila las validaciones de dos tablas, asignadas a ese proceso
8. Exporta las validaciones tipo C.	Exporta las validaciones en una pestaña aparte, adicionando una columna que relaciona las cantidades de registros a revisar. En caso de encontrar inconsistencias, las exporta en una pestaña distinta que renombra individualmente con el código único de la validación

02_ValidaFicha

El código produce el CCVPPP.Rmd, y posteriormente lo ejecuta, para crear el CCVPPP.docx., empleando la tabla de salida del proceso anterior.

El segundo código se alimenta de este insumo final, para generar la ficha. Lo hace de la siguiente manera, generalmente:

1. Llama la matriz que sirve de insumo de la carpeta “doc”.
2. Lee solamente los resultados generales de validación.
3. Genera en la carpeta .Rmd un archivo de control, llamado de acuerdo al proceso, por ejemplo C006 si está en dicho proceso.
4. Ejecuta el código, y genera un .docx, que es la ficha reportada al equipo.

Lenguaje: R

Número de líneas: 152

Tablas (archivo) de entrada: MotorValidaAAMMDD_HHMM.xlsx

Tablas (archivo) de salida: CCVPPP.docx.

Tiempo estimado de ejecución: Depende de las tablas de entrada. Promedio 15 segundos.

Descripción del algoritmo:

Paso	Descripción
1. Cargue insumos	Llama la matriz que sirve de insumo de la carpeta “doc”.
2. Discrimine resultados	Lee solamente los resultados generales de validación.
3. Produce el .Rmd.	Genera en la carpeta .Rmd un archivo de control, llamado de acuerdo al proceso, por ejemplo C006 si está en dicho proceso.
4. Ejecuta el .Rmd y genera el .docx.	Ejecuta el código, y genera un .docx, que es la ficha reportada al equipo.

CCVPPP.Rmd

El parámetro PPP hace referencia, como antes, a los posibles procesos: C01 hasta C15, según sea el caso. Este código está programado para producir un .docx, que será la Ficha Temporal de Validación.

Lenguaje: Rmd

Número de líneas: Depende del proceso, en promedio 72

Tablas de entrada: No aplica

Tablas de salida: Es la salida del código 02_ValidaFicha.R.

Tiempo estimado de ejecución: 10 segundos.

Descripción: Es un archivo ejecutable, producido desde R, con el objetivo de producir un .docx.

Listas o reglas de validación por procesos o códigos

Proceso C003

En esta fase del proceso, se construye la tabla T007_T0_Consolidada. La validación consiste en las siguientes reglas:

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
C02.tConN	T0; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de T0 están incluidos en CÓDIGO_SIGEP de pivoteC
C02.tConN	pivoteC; T0; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de pivoteC están incluidos en CÓDIGO_SIGEP de T0
C03.nConF	T0; pivoteC	Las columnas de T0 están contenidas en las de pivoteC
V02.NoDupl	pivoteC; CÓDIGO_SIGEP	La matriz pivoteC no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	pivoteC; CÓDIGO_SIGEP	La matriz pivoteC no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	T0; CÓDIGO_SIGEP	La matriz T0 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V04.ValEsp	T0; CIN01C; 0,1	La columna Var2 de la tabla Var1 solo debe tomar los valores Var3
V01.NoNulo	T0; CÓDIGO_SIGEP	La matriz T0 no debe contener valores nulos en la columna CÓDIGO_SIGEP

Aquí incluye, además, para asistir a la validación de la tabla T0 consolidada (nombrada simplemente como T0), la tabla pivoteC, que hace referencia al pivote consolidado. Asiste en dos vías: verifica que esté definida exactamente para las mismas entidades y ayuda a verificar que las preguntas definidas en T0 estén observadas en el pivoteC.

Adicionalmente, se define la validación de valor esperado: valores de 0 y 1 para en particular una sola columna. Esta validación se hace para todos los campos posibles, pero no se muestran ya que son 2.758 columnas.

Proceso C004

En esta fase del proceso, se construye la tabla T011_T1. La validación consiste en las siguientes reglas:

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
C02.tConN	T0; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de T0 están incluidos en CÓDIGO_SIGEP de pivoteC
C02.tConN	pivoteC; T0; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de pivoteC están incluidos en CÓDIGO_SIGEP de T0
C03.nConF	T0; pivoteC	Las columnas de T0 están contenidas en las de pivoteC
V02.NoDupl	pivoteC; CÓDIGO_SIGEP	La matriz pivoteC no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	pivoteC; CÓDIGO_SIGEP	La matriz pivoteC no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	T0; CÓDIGO_SIGEP	La matriz T0 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V04.ValEsp	T0; CIN01C; 0,1	La columna Var2 de la tabla Var1 solo debe tomar los valores Var3



V01.NoNulo	T0; CÓDIGO_SIGEP	La matriz T0 no debe contener valores nulos en la columna CÓDIGO_SIGEP
------------	------------------	--

Aquí incluye, además, para asistir a la validación de la tabla T1 consolidada (nombrada simplemente como T1), la tabla pivoteC, que hace referencia al pivote consolidado. Asiste en dos vías: verifica que esté definida exactamente para las mismas entidades y ayuda a verificar que las preguntas definidas en T1 estén observadas en el pivoteC.

Adicionalmente, se define la validación de valor esperado: valores de 0 y 1 para en particular una sola columna. Esta validación se hace para todos los campos posibles, pero no se muestran ya que son 2.758 columnas.

Proceso C005

En esta fase del proceso, se construye la tabla T010_Pivote_consolidado. La validación consiste en las siguientes reglas:

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
V02.NoDupl	pivoteC; CÓDIGO_SIGEP	La matriz pivoteC no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	pivoteC; CÓDIGO_SIGEP	La matriz pivoteC no debe contener valores nulos en la columna CÓDIGO_SIGEP
C02.tConN	Dpivote3; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de Dpivote3 están incluidos en CÓDIGO_SIGEP de pivoteC
C02.tConN	Dpivote4; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de Dpivote4 están incluidos en CÓDIGO_SIGEP de pivoteC
C02.tConN	Fpivote1; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de Fpivote1 están incluidos en CÓDIGO_SIGEP de pivoteC
C02.tConN	Fpivote2; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de Fpivote2 están incluidos en CÓDIGO_SIGEP de pivoteC
C02.tConN	Fpivote3; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de Fpivote3 están incluidos en CÓDIGO_SIGEP de pivoteC
C02.tConN	Fpivote4; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de Fpivote4 están incluidos en CÓDIGO_SIGEP de pivoteC
C02.tConN	Fpivote5; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de Fpivote5 están incluidos en CÓDIGO_SIGEP de pivoteC
C02.tConN	Fpivote6; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de Fpivote6 están incluidos en CÓDIGO_SIGEP de pivoteC
C02.tConN	Fpivote7; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de Fpivote7 están incluidos en CÓDIGO_SIGEP de pivoteC
C02.tConN	Fpivote8; pivoteC; CÓDIGO_SIGEP; CÓDIGO_SIGEP	Valores de la columna CÓDIGO_SIGEP de Fpivote8 están incluidos en CÓDIGO_SIGEP de pivoteC



V02.NoDupl	Dpivot3; CÓDIGO_SIGEP	La matriz Dpivot3 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	Dpivot3; CÓDIGO_SIGEP	La matriz Dpivot3 no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	Dpivot4; CÓDIGO_SIGEP	La matriz Dpivot4 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	Dpivot4; CÓDIGO_SIGEP	La matriz Dpivot4 no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	Fpivot1; CÓDIGO_SIGEP	La matriz Fpivot1 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	Fpivot1; CÓDIGO_SIGEP	La matriz Fpivot1 no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	Fpivot2; CÓDIGO_SIGEP	La matriz Fpivot2 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	Fpivot2; CÓDIGO_SIGEP	La matriz Fpivot2 no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	Fpivot3; CÓDIGO_SIGEP	La matriz Fpivot3 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	Fpivot3; CÓDIGO_SIGEP	La matriz Fpivot3 no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	Fpivot4; CÓDIGO_SIGEP	La matriz Fpivot4 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	Fpivot4; CÓDIGO_SIGEP	La matriz Fpivot4 no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	Fpivot5; CÓDIGO_SIGEP	La matriz Fpivot5 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	Fpivot5; CÓDIGO_SIGEP	La matriz Fpivot5 no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	Fpivot6; CÓDIGO_SIGEP	La matriz Fpivot6 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	Fpivot6; CÓDIGO_SIGEP	La matriz Fpivot6 no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	Fpivot7; CÓDIGO_SIGEP	La matriz Fpivot7 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	Fpivot7; CÓDIGO_SIGEP	La matriz Fpivot7 no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	Fpivot8; CÓDIGO_SIGEP	La matriz Fpivot8 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V01.NoNulo	Fpivot8; CÓDIGO_SIGEP	La matriz Fpivot8 no debe contener valores nulos en la columna CÓDIGO_SIGEP

En este caso, pivotC hace referencia a la table de salida, el pivote consolidado. Las tablas MpivotK, hacen referencia a los pivotes descargados por separado:

1. M toma valores de D y F, los meses diciembre y febrero.
2. K toma valores de 1 hasta 8, los pivotes construidos con esa codificación: pivot1, etc.

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
V01.NoNulo	diccionarioC; ID	La matriz diccionarioC no debe contener valores nulos en la columna ID
V01.NoNulo	diccionarioC; PREGUNTA	La matriz diccionarioC no debe contener valores nulos en la columna PREGUNTA



V01.NoNulo	diccionarioC; RESPUESTA	La matriz diccionarioC no debe contener valores nulos en la columna RESPUESTA
V01.NoNulo	matrizX; Cód.	La matriz matrizX no debe contener valores nulos en la columna Cód.
V01.NoNulo	matrizX; Tipo.de.pregunta	La matriz matrizX no debe contener valores nulos en la columna Tipo.de.pregunta.
V01.NoNulo	matrizX; INDICE.GENERAL.DE.DESEMPEÑO.INSTITUCIONAL	La matriz matrizX no debe contener valores nulos en la columna INDICE.GENERAL.DE.DESEMPEÑO.INSTITUCIONAL
V01.NoNulo	pivoteC; CÓDIGO_SIGEP	La matriz pivoteC no debe contener valores nulos en la columna CÓDIGO_SIGEP
C01.tConD	pivoteC; diccionarioC; ID	Nombres de pivoteC deben estar incluidos en columna ID de tabla diccionarioC
V02.NoDupl	diccionarioC; ID	La matriz diccionarioC no debe contener valores duplicados en la columna ID
V01.NoNulo	MatrizXC; CodP	La matriz MatrizX no debe contener valores nulos en la columna CodP
V02.NoDupl	MatrizXC; CodP	La matriz MatrizX no debe contener valores duplicados en la columna CodP
C01.tConD	PivoteF; MatrizXC; CodP	Nombres de PivoteF deben estar incluidos en columna CodP de tabla MatrizXC
V01.NoNulo	PivoteF; CÓDIGO_SIGEP	La matriz PivoteF no debe contener valores nulos en la columna CÓDIGO_SIGEP
V03.LargoE	PivoteF; CÓDIGO_SIGEP;4	CÓDIGO_SIGEP de la tabla PivoteF debe tener ancho 4

Las validaciones evidencian que toda la información de cada pivote, queda consignada en el pivote consolidado.

Proceso C006

En esta fase del proceso, se construye la tabla T015_MatrizX_Validada. La validación consiste en las siguientes reglas:

En este caso, PivoteF hace referencia a la tabla de salida del código, al cruzar la matriz X con el pivote. Se asiste de esta para revisar que una vez validada la Matriz X (denominada MatrizXC), las variables validadas de PivoteF deben estar contenidos en la columna que define las variables de puntuación en la matriz X validada.

Proceso C007

En esta fase del proceso, se construye la tabla T016_Pivote_final. La validación consiste en las siguientes reglas:

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
V01.NoNulo	PivoteFN; CÓDIGO_SIGEP	La matriz PivoteFN no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	PivoteFN; CÓDIGO_SIGEP	No acepta valores duplicados
V01.NoNulo	T014Nv; Nombre_Tabla	La matriz T014Nv no debe contener valores nulos en la columna Nombre_Tabla



V01.NoNulo	T014Nv; Nombre_Campo	La matriz T014Nv no debe contener valores nulos en la columna Nombre_Campo
V01.NoNulo	T014Nv; Variables	La matriz T014Nv no debe contener valores nulos en la columna Variables
V01.NoNulo	T014Nv; Función	La matriz T014Nv no debe contener valores nulos en la columna Función
V02.NoDupl	T014Nv; Nombre_Campo	La matriz T014Nv no debe contener valores duplicados en la columna Nombre_Campo
C01.tConR	T014Nv; PivoteFN; Nombre_Campo	Valores de Nombre_Campo en T014Nv no incluidos en los nombres de PivoteFN

La matriz T014Nv es el tablero de control que ejecuta las nuevas variables, es decir, donde se definen y sus reglas. La tabla PivoteFN es la salida del código.

Además de estas validaciones, se genera una tabla por cada nueva variable, denominada F_NV_ii que contiene las variables que se emplean en el cálculo de la nueva variable, la parte cuantitativa (la razón si es el caso) y la posterior transformación para cada una en categorías ordinales. Además, se hace una tabla de frecuencias para administrar el control de la variabilidad, buscando armonía en la puntuación y sus respectivos procesos.

Proceso C008

En esta fase del proceso, se construye la tabla T017_T3 y T018_T4. La validación consiste en las siguientes reglas:

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
V01.NoNulo	MatrizXC; CodP	La matriz MatrizXC no debe contener valores nulos en la columna CodP
V02.NoDupl	MatrizXC; CodP	La matriz MatrizXC no debe contener valores duplicados en la columna CodP
V01.NoNulo	T1; CÓDIGO_SIGEP	La matriz T1 no debe contener valores nulos en la columna CÓDIGO_SIGEP
V02.NoDupl	T1; CÓDIGO_SIGEP	La matriz T1 no debe contener valores duplicados en la columna CÓDIGO_SIGEP
V03.LargoE	T1; CÓDIGO_SIGEP;4	CÓDIGO_SIGEP de la tabla T1 debe tener ancho 4
C01.tConD	T1; T3; idPreg	Nombres de T1 deben estar incluidos en columna idPreg de tabla T3
C01.tConT	T3; T4; Ind	Valores de Ind en T3 deben estar incluidos en columna Ind de tabla T4
C01.tConT	T3; T4; Esc	Valores de Esc en T3 deben estar incluidos en columna Esc de tabla T4

Se asiste de las matrices MatrizXC y T1 para validar la pertinencia de la ejecución, en calcular T3 y T4.

Proceso C009

En esta fase del proceso, se construye la tabla T019_TD. La validación consiste en las siguientes reglas:

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
C01.tConT	T3; T4; Ind	Valores de Ind en T3 deben estar incluidos en columna Ind de tabla T4



C01.tConT	T3; T4; Esc	Valores de Esc en T3 deben estar incluidos en columna Esc de tabla T4
C01.tConT	PivoteF ; TD; CÓDIGO_SIGEP	Valores de CÓDIGO_SIGEP en TD deben estar incluidos en columna CÓDIGO_SIGEP de tabla PivoteF
C02.tConN	PivoteF ; T4; CÓDIGO_SIGEP; codigoSigep	Valores de la columna CÓDIGO_SIGEP de PivoteF están incluidos en codigoSigep de T4
C02.tConN	TD ; T4; CÓDIGO_SIGEP; codigoSigep	Valores de la columna CÓDIGO_SIGEP de TD están incluidos en codigoSigep de T4
C02.tConN	TD ; T3; Ind; Ind	Valores de la columna Ind de TD están incluidos en Ind de T3
V01.NoNulo	TD; CÓDIGO_SIGEP	La matriz TD no debe contener valores nulos en la columna CÓDIGO_SIGEP
V01.NoNulo	PivoteF; CÓDIGO_SIGEP	La matriz PivoteF no debe contener valores nulos en la columna CÓDIGO_SIGEP

Se asiste de las matrices T3, T4 y PivoteF para revisar contenencias.

Proceso C010

En esta fase del proceso, se construye la tabla T020_Dfw. La validación consiste en las siguientes reglas:

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
C02.tConN	PivoteF ; dfw01; CÓDIGO_SIGEP; codigoSigep	Valores de la columna CÓDIGO_SIGEP de PivoteF están incluidos en codigoSigep de dfw01
C01.tConD	dfw01 ; rdfw; idDfw	Nombres de dfw01 deben estar incluidos en columna idDfw de tabla rdfw
V01.NoNulo	dfw01; codigoSigep	La matriz dfw01 no debe contener valores nulos en la columna codigoSigep
V02.NoDupl	dfw01; codigoSigep	La matriz dfw01 no debe contener valores duplicadas en la columna codigoSigep
V02.NoDupl	rdfw; idDfw	La matriz rdfw no debe contener valores duplicadas en la columna idDfw

Se aclara que la tabla T020 hace referencia a dfw01. La matriz dfw01 se acompaña de una suerte de diccionario, que se denomina para el proceso rdfw, que a su vez sirve como insumo de comparación.

Proceso C011

En esta fase del proceso, se construye la tabla T025_Dfw_corregida. La validación consiste en las siguientes reglas:

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
C02.tConN	dfw01 ; dfw01F; codigoSigep; codigoSigep	Valores de la columna codigoSigep de dfw01 están incluidos en codigoSigep de dfw01F
C02.tConN	dfw01F ; dfw01; codigoSigep; codigoSigep	Valores de la columna codigoSigep de dfw01F están incluidos en codigoSigep de dfw01
V01.NoNulo	dfw01F; codigoSigep	La matriz dfw01F no debe contener valores nulos en la columna codigoSigep
V02.NoDupl	dfw01F; codigoSigep	La matriz dfw01F no debe contener valores duplicados en la columna codigoSigep

C03.nConF	dfw01F ; dfw01	Las columnas de dfw01F están contenidas en las de dfw01
C03.nConF	dfw01 ; dfw01F	Las columnas de dfw01 están contenidas en las de dfw01F

Se aclara que la tabla T025 hace referencia a dfw01F.

Proceso C012

En esta fase del proceso, se construye la tabla T026_Ancla_W. La validación consiste en las siguientes reglas:

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
C01.tConD	dfw01 ; anclaW1; item	Nombres de dfw01 deben estar incluidos en columna item de tabla anclaW1
V01.NoNulo	dfw01; codigoSigep	La matriz dfw01 no debe contener valores nulos en la columna codigoSigep
V02.NoDupl	dfw01; codigoSigep	La matriz dfw01 no debe contener valores duplicados en la columna codigoSigep
V01.NoNulo	anclaW1; item	La matriz anclaW1 no debe contener valores nulos en la columna item
V02.NoDupl	anclaW1; item	La matriz anclaW1 DEBE contener valores duplicados en la columna item
V01.NoNulo	anclaW1; value	La matriz anclaW1 no debe contener valores nulos en la columna value
V01.NoNulo	anclaW1; class	La matriz anclaW1 no debe contener valores nulos en la columna class
V01.NoNulo	anclaW1; name	La matriz anclaW1 no debe contener valores nulos en la columna name
V01.NoNulo	anclaW1; parnum	La matriz anclaW1 no debe contener valores nulos en la columna parnum

Se aclara que la tabla T026 hace referencia a anclaW1. El valor de la columna "ítem" no puede no ser duplicado, ya que corresponde cada fila a un parámetro y no existen ítems con un solo parámetro.

Proceso C013

En esta fase del proceso, se construye la tabla T030_Tbl_score_ent_t1. La validación consiste en las siguientes reglas:

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
C02.tConN	pPubl ; dfw01; codigoSigep ; codigoSigep	Valores de la columna codigoSigep de pPubl están incluidos en codigoSigep de dfw01
V01.NoNulo	dfw01; codigoSigep	La matriz dfw01 no debe contener valores nulos en la columna codigoSigep
V02.NoDupl	dfw01; codigoSigep	La matriz dfw01 no debe contener valores duplicados en la columna codigoSigep
V02.NoDuplW	pPubl; codigoSigep; Ind	La matriz dfw01 no debe contener valores duplicados agregando por columnas codigoSigep e Ind

Se aclara que la tabla T030 hace referencia a pPubl en el proceso.

Proceso C014

En esta fase del proceso, se construye la tabla T028_Dfw_h. La validación consiste en las siguientes reglas:

NOMBRE_FUNCION	PARAMETROS	DESCRIPCION_DEFINICION
C02.tConN	dfw01 ; dfwh; codigoSigep ; codigoSigep	Valores de la columna codigoSigep de dfw01 están incluidos en codigoSigep de dfwh
C02.tConN	dfw00 ; dfwh; codigoSigep ; codigoSigep	Valores de la columna codigoSigep de dfw00 están incluidos en codigoSigep de dfwh
V01.NoNulo	dfwh; codigoSigep	La matriz dfwh no debe contener valores nulos en la columna codigoSigep
V02.NoDupl	dfwh; codigoSigep	La matriz dfwh no debe contener valores duplicados en la columna codigoSigep

Se aclara que la tabla T028 hace referencia a dfwh en el proceso. Además, la tabla dfw00 hace referencia a la tabla T027_Dfw_t0, es decir, la tabla del año anterior.

Proceso C015

En esta fase del proceso, se construye la tabla T031_Tbl_equipracion. La validación consiste en revisar que la salida tenga dos entradas. En efecto, la tabla de salida es:

kSuma	kProd
0,064804928	1,25885460

En general, se espera que el valor de la constante de suma kSuma tenga esperanza matemática de 0, mientras que la constante kProd, que es la constante de producto, tenga esperanza de 1.

Bibliografía

DANE. (2014). *Lineamientos para la definición y aplicación de pruebas en las operaciones estadísticas*. Bogotá D.C.: DANE.

Dirección de Gestión y Desempeño Institucional. (2019). *Metodología de Medición de Desempeño de Entidades Públicas*. Bogotá: Departamento Administrativo de la Función Pública.