



Plataforma para la Gestión de Calidad de Datos en Gobierno Digital

Diciembre - 2020

Estudiantes:

Nicolás Fiumarelli

Agustin Fornio

Supervisores:

Dra. Ing. Laura González

MSc. Ing. Bruno Rienzi

Resumen

Durante los últimos años, debido al gran avance de las tecnologías de la información y comunicación, los datos se han vuelto cada vez más importantes en las organizaciones. En particular, esto se da en el área de gobierno digital, en la cual las organizaciones gestionan cantidades de datos cada vez más grandes. En este contexto, resulta fundamental realizar una correcta gestión de la calidad de datos, ya que puede tener un gran impacto en la calidad de los servicios públicos que se brindan a los ciudadanos así como en la definición de estrategias y políticas públicas.

Con esta motivación, la Agencia de Gobierno Electrónico y Sociedad de la Información y del Conocimiento (AGESIC) en conjunto con el Instituto de Computación y el Instituto de Agrimensura de la Facultad de Ingeniería de la Universidad de la República, avanzaron en la definición de un framework que apunta a contribuir a la sistematización de la gestión de la calidad de datos en organizaciones vinculadas al gobierno digital en Uruguay. Entre otras cosas, el framework propone un proceso que guía dichas organizaciones en la estructuración y ejecución de las actividades necesarias para gestionar la calidad de los datos.

Este proyecto propone una plataforma para dar soporte a la gestión de calidad de datos aplicando dicho framework. Esta plataforma, permite que diferentes organizaciones, a través de las etapas definidas en dicho framework, lleven a cabo la gestión de calidad de datos en diversos escenarios relevantes para las mismas. Algunas de las principales etapas a las que brinda soporte la plataforma son caracterizar el escenario sobre el cual aplicar el framework, conocer las características de los datos del escenario y la definición de un modelo de calidad de datos.

La propuesta fue validada mediante la implementación de un prototipo, el desarrollo de un caso de estudio y reuniones con referentes de AGESIC para evaluar la adecuación funcional.

Palabras claves: gestión de calidad de datos, framework, gobierno digital, AGESIC.

Tabla de Contenido

1. Introducción	5
1.1. Objetivos	5
1.2. Aportes	6
1.3. Organización del Documento	7
2. Marco Teórico	8
2.1. Calidad de Datos	8
2.2. Modelos de Calidad de Datos	9
2.3. Gestión de Calidad	10
2.4. Componentes del Framework	11
2.5. Modelo de Calidad de Datos de Referencia	12
2.6. Escenarios de Trabajo	14
2.7. Proceso para la Gestión de Calidad de Datos	15
3. Análisis	19
3.1. Descripción General	19
3.2. Requerimientos	20

3.3. Principales Actores y Casos de uso	21
3.4. Decisiones Tomadas	25
4. Solución Propuesta	27
4.1. Descripción General	27
4.2. Arquitectura	28
4.3. Principales Funcionalidades	32
5. Implementación	43
5.1. Herramientas y Tecnologías	43
5.2. Control de Acceso	47
5.3. Representación Gráfica	48
5.4. Gestión del Modelo de Calidad	51
5.5. Gestión de Versiones del Modelo	53
6. Caso de Estudio y Validación	57
6.1. Descripción del Caso de Estudio	57
6.2. Aplicación del Framework	58
6.3. Validación AGESIC	69
6.4. Resumen	70

7. Conclusiones y Trabajo a Futuro	72
7.1. Conclusiones	72
7.2. Trabajo a Futuro	73
Referencias	76
A. Estructuras JSON del Diagrama	77
B. Estructuras JSON del Modelo de Calidad	81
C. Mockups iniciales	83

1. Introducción

Las organizaciones constantemente manejan mayores cantidades de datos y de diferentes tipos. De acuerdo a [1], las empresas que inviertan para administrar este recurso vital, tendrán más posibilidades de éxito que aquellas que no lo hacen. En el contexto de gobierno digital, esta realidad brinda oportunidades para las operaciones diarias de las organizaciones gubernamentales, la toma de decisiones y planes estratégicos.

Sin embargo, estas oportunidades pueden verse afectadas debido a los problemas de calidad de datos característicos de estos contextos. A modo de ejemplo, dos o más organizaciones que gestionan los datos de un mismo ciudadano pueden manejar datos contradictorios del mismo. Además, los datos pueden estar desactualizados, tener problemas de calidad causados por el ingreso manual en los sistemas y/o diferir en el formato.

Debido a esto, AGESIC en conjunto con la Facultad de Ingeniería, desarrollaron un framework para la gestión de calidad de los datos en las diferentes organizaciones asociadas al gobierno digital.

La plataforma que se propone en este proyecto apunta a brindar soporte para la aplicación del framework, facilitando que las organizaciones cuenten con información de mejor calidad a la hora de tomar decisiones estratégicas para poder ofrecer mejores servicios y productos a los ciudadanos de Uruguay.

1.1. Objetivos

El proyecto tiene como objetivo general desarrollar una plataforma que brinde soporte al proceso para la gestión de calidad de datos descrito en el “Framework para la Gestión de la Calidad de Datos en Gobierno Digital” [2].

Para cumplir con este objetivo, se plantean los siguientes objetivos específicos:

- Estudiar el conocimiento existente referido a calidad de datos y procesos de gestión de calidad de datos.
- Analizar la realidad planteada, identificando las principales funcionalidades, actores y requerimientos que una plataforma debería tener para brindar soporte al framework.
- Proponer una plataforma que permita la gestión de la calidad de datos utilizando el framework definido y que cumpla con los requerimientos identificados.
- Diseñar e implementar un prototipo de la plataforma que permita la evaluación de su factibilidad técnica.
- Evaluar la adecuación funcional de la plataforma en base al prototipo, casos de estudio e instancias de validación con actores interesados.

1.2. Aportes

Los principales aportes de este proyecto son:

- Análisis de la realidad planteada y relevamiento de las principales funcionalidades que debe brindar una plataforma de gestión de calidad de datos basada en el framework [2] y adaptada a la realidad del gobierno digital en Uruguay.
- Propuesta de plataforma que brinda soporte para la gestión de calidad de datos a los diferentes actores identificados aplicando el framework.
- Prototipo de la plataforma implementado en Java, que se encuentra disponible en servidores de “Mi nube”¹, que permite una validación preliminar de su factibilidad técnica.
- Validación de la adecuación funcional de la propuesta por medio de un caso de estudio e instancias de validación con integrantes de AGESIC.

¹Plataforma funcional:<http://179.27.96.169:8180/calidad>.

- Esquema de versionado para gestionar un modelo de calidad de datos de referencia, el cual permite tener diferentes históricos de los elementos del modelo.
- Una arquitectura que se ajusta, tanto a la realidad de un organismo centralizado como lo es AGESIC, como también a aquellas organizaciones que prefieran ejecutar la plataforma en un ambiente local.

1.3. Organización del Documento

El resto del documento se encuentra estructurado de la siguiente manera:

- **Marco Teórico.** En este capítulo se definen los conceptos base para facilitar el entendimiento del documento.
- **Análisis.** Este capítulo presenta el análisis realizado para identificar las principales características de la plataforma, los actores y casos de uso.
- **Solución Propuesta.** Este capítulo presenta una descripción general de la solución junto a las principales funcionalidades de la plataforma propuesta.
- **Implementación.** En este capítulo se presentan aspectos claves del prototipo de la plataforma implementada, así como también las tecnologías utilizadas.
- **Caso de Estudio y Validación.** En este capítulo se realiza una evaluación, a partir de un caso de estudio e instancias de validación de la propuesta con AGESIC.
- **Conclusiones y Trabajo a Futuro.** Este capítulo plantea las conclusiones del proyecto y se presentan los lineamientos generales sobre el trabajo a futuro.

2. Marco Teórico

El objetivo del presente capítulo es proveer un marco conceptual para comprender y manejar los distintos elementos relevantes para el proyecto, que incluyen tanto conceptos generales de calidad de datos así como elementos del framework.

Gran parte de los conceptos definidos a continuación provienen de “Framework para la Gestión de la Calidad de Datos en Gobierno Digital” [2].

2.1. Calidad de Datos

La calidad de datos es un área de investigación muy amplia. Debido a que tiene una fuerte influencia en los sistemas de información en diferentes campos, tiene gran relevancia en la industria. El término, según [3], se utiliza para referirse a un conjunto de características que deben tener los datos, como su correctitud y grado de actualización, entre otros.

Asimismo, se define como la capacidad de cumplir con los requisitos de uso de los datos. Por lo tanto, los datos carecen de calidad en la medida en que no cumplen los requerimientos[4], es decir, la calidad de datos depende tanto del uso que se le vaya a dar a los datos como de los datos en sí.

Los datos representan objetos del mundo real que resultan ser aplicables en diversos contextos y se pueden almacenar, procesar o transformar resultando de gran importancia para el contexto en el cual se estén manejando. En particular, la mala calidad de los datos tiene graves consecuencias para la eficiencia y la eficacia de las organizaciones y empresas, ya que “sin buenos datos, las organizaciones están ciegas” [1] y no pueden tomar una decisión correcta porque no tienen una comprensión precisa de lo que está sucediendo dentro o fuera de la empresa.

2.2. Modelos de Calidad de Datos

Como se describe en [5], la calidad de datos se caracteriza por múltiples dimensiones, que ayudan a clasificar los datos o los procesos que manipulan estos datos. La calidad de datos es entonces un concepto “multifacético”, dado que se representa por un conjunto de dimensiones que abordan diferentes aspectos de los datos. Como se presenta en la Figura 1, siguiendo este enfoque se puede definir una jerarquía de conceptos de calidad.

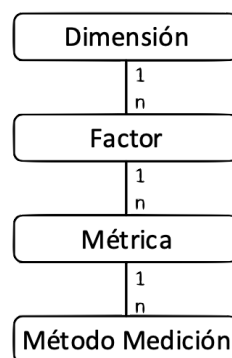


Figura 1: Jerarquía de Conceptos de Calidad de Datos [2]

Una **dimensión** captura una característica de la calidad y puede definirse como un conjunto de factores de calidad con el mismo propósito. Por otro lado, un **factor** representa aspectos específicos de las dimensiones y puede medirse utilizando diferentes métricas. Una **métrica** es un instrumento que define cómo medir cierto factor de calidad y puede ser medida por diferentes métodos. Un **método** es un proceso que implementa una métrica y permite obtener medidas de calidad para la misma.

Para gestionar la calidad de los datos en un sistema, es fundamental definir un modelo de calidad adecuado a las necesidades y prioridades de los usuarios de los datos. El modelo define las dimensiones de calidad a considerar, qué datos se aplican y cómo medirlos. De esta manera, el modelo guía la gestión de la calidad del conjunto de datos específico.

2.3. Gestión de Calidad

La gestión de la calidad de los datos es la tarea de medir, analizar, mejorar y controlar diferentes aspectos de la calidad de los datos que son de interés para un escenario específico. Esta tarea involucra un conjunto de métodos, técnicas y herramientas para administrar la calidad de los datos. La Figura 2 presenta estos elementos.

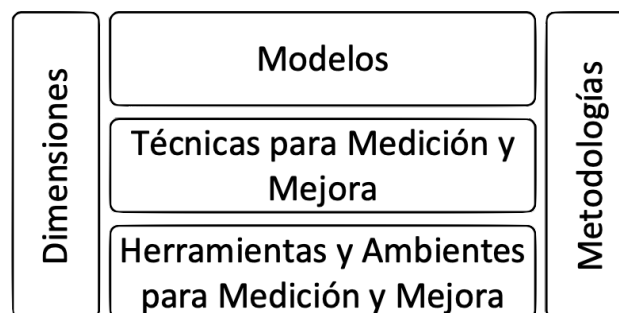


Figura 2: Elementos para la Gestión de la Calidad de Datos [2]

Inicialmente debe existir un modelo de calidad que defina las dimensiones adecuadas para el conjunto de datos de interés. Para construir este modelo, se deben adoptar métodos que apoyen la identificación y clasificación de los problemas de calidad. En función de los tipos de problemas de calidad detectados, se seleccionan las técnicas y herramientas adecuadas para que estas técnicas midan y mejoren la calidad de los datos.

La Figura 3 presenta las distintas etapas en la gestión de la calidad de datos.

La etapa de análisis de procesos de negocio involucrados permite identificar el conjunto de datos relevantes para el escenario. Seguido a esto, en la etapa de *Data Profiling*, se permite conocer los datos para los cuales se construirá el modelo de calidad. A partir de estos resultados y de un análisis preliminar de las causas de la mala calidad, se define un modelo de calidad.

El modelo se utiliza para analizar y evaluar las causas de mala calidad, por lo que tanto la construcción del mismo como el análisis de las causas, serán procesos paralelos e iterativos.

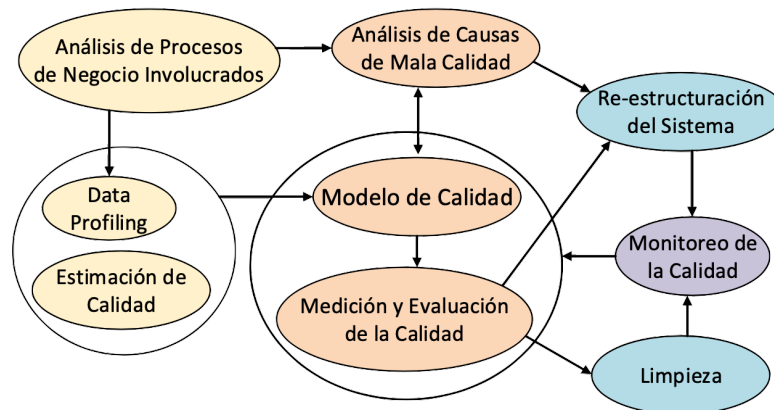


Figura 3: Etapas de la Gestión de Calidad de Datos [2]

Finalmente, para mantener el nivel de calidad de los datos requeridos por las organizaciones, se debe realizar un monitoreo continuo de la calidad. Este seguimiento puede requerir ajustes al modelo de calidad, así como nuevas mediciones y evaluaciones de la calidad de los datos.

2.4. Componentes del Framework

El framework para la gestión de la calidad de datos en el que se basa este proyecto [2] se compone de un conjunto de elementos los cuales se presentan de forma gráfica en la Figura 4.

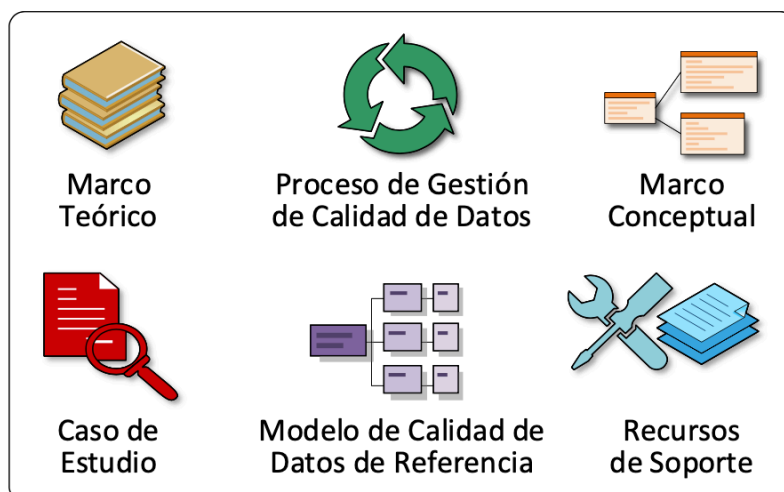


Figura 4: Componentes del Framework para la Gestión de la Calidad de Datos [2]

El **Marco Teórico** proporciona una base teórica para gestionar la calidad de los datos en el gobierno digital y utilizar el framework. Aborda temas de calidad de datos, modelos de calidad de datos, gestión de calidad de datos, etc.

El **Marco Conceptual** define los principales conceptos relacionados con la gestión de la calidad de los datos en el gobierno digital. Los mismos facilitan la comprensión de la gestión de la calidad de los datos en estos contextos.

El **Modelo de Calidad de Datos de Referencia** brinda un conjunto de elementos de calidad de datos extensibles e instanciables (como dimensiones, factores, métricas) para guiar y promover la definición de modelos de calidad de datos para escenarios de trabajo específicos.

El **Proceso para la Gestión de Calidad de Datos** define los roles y etapas involucradas en la gestión de la calidad de los datos en el escenario del gobierno digital.

Recursos de Soporte se define como un conjunto de componentes extensibles que dan soporte o referencia para la gestión de la calidad de los datos y la aplicación del framework.

Por último, el **Caso de Estudio** plantea un escenario de trabajo específico para motivar, ejemplificar y orientar la gestión de la calidad de los datos en estos contextos.

2.5. Modelo de Calidad de Datos de Referencia

El modelo de calidad de datos de referencia con el que se trabajó en el proyecto y que se define en el framework [2], presenta en un primer nivel cinco dimensiones: Exactitud, Consistencia, Completitud, Unicidad y Frescura. El segundo y tercer nivel están formados por los factores de cada dimensión y por las métricas de cada factor, respectivamente. En la Figura 5 se resumen las dimensiones y los factores del modelo.

A continuación, se describen las dimensiones de este modelo [2]:

Exactitud	Correctitud Semántica
	Correctitud Sintáctica
	Precisión
	Exactitud Posicional Absoluta
	Exactitud Posicional Relativa
Consistencia	Integridad Intra-entidad
	Integridad Inter-entidad
	Integridad de Dominio
	Consistencia Topológica
Completitud	Cobertura
	Densidad
	Comisión
Unicidad	Duplicación
	Contradicción
Frescura	Actualidad
	Oportunidad

Figura 5: Dimensiones y Factores del Modelo de Calidad de Datos de Referencia [2]

- **Exactitud** [6] [7]: indica qué tan precisos, válidos y libres de errores están los datos. Se la define como la correcta y precisa asociación entre los estados del sistema de información y los objetos del mundo real.
- **Completitud** [6] [7]: indica si el sistema contiene toda la información de interés. Se define como la capacidad del sistema de información de representar todos los estados significativos de una realidad dada.
- **Frescura** [8]: captura la rapidez con la que los cambios en el mundo real son reflejados en la actualización de los datos. La frescura es un tipo de exactitud no-estructural dependiente de la variable tiempo, lo que implica que un dato que en un determinado momento es correcto, puede no serlo en otro momento .
- **Consistencia** [6] [7]: captura la satisfacción de reglas semánticas definidas sobre los datos, ya sea que cumplan las reglas del dominio o las posibles contradicciones entre los

datos. Pueden ser reglas de integridad de la base de datos o reglas de integridad de los usuarios.

- **Unicidad** [6]: indica el nivel de duplicación entre los datos. La duplicación ocurre cuando una misma entidad está representada dos o más veces en un sistema de información.

2.6. Escenarios de Trabajo

Los escenarios de trabajo aplicables a este framework se llevan a cabo en el marco del gobierno digital, donde diferentes organizaciones ejecutan procesos de negocio y colaboran entre sí para brindar servicios públicos a los ciudadanos. La Figura 6 presenta algunos de los principales elementos que pueden encontrarse en un escenario de trabajo tipo.

Las organizaciones dependen de aplicaciones y sistemas para ejecutar sus procesos de negocio, que pueden implementarse implícita y explícitamente en sistemas especializados. Estas organizaciones utilizan diferentes tipos de colecciones de datos (como bases de datos relacionales), que almacenan información sobre las entidades de negocio relevantes para la organización (por ejemplo certificados, empresas, ciudadanos, beneficios). Estas entidades (por ejemplo, ciudadanos) tienen atributos (por ejemplo, nombre, dirección) que pueden ser de diferentes tipos (por ejemplo, alfanuméricos, geográficos, de imagen).

Las organizaciones ofrecen un portal a través del cual interactúan con los ciudadanos, permitiéndoles formar parte de procesos de negocio (por ejemplo realizar trámites) y acceder a los datos almacenados. Las organizaciones también interactúan entre sí (por ejemplo para consultar datos) y lo hacen a través de servicios de negocio.

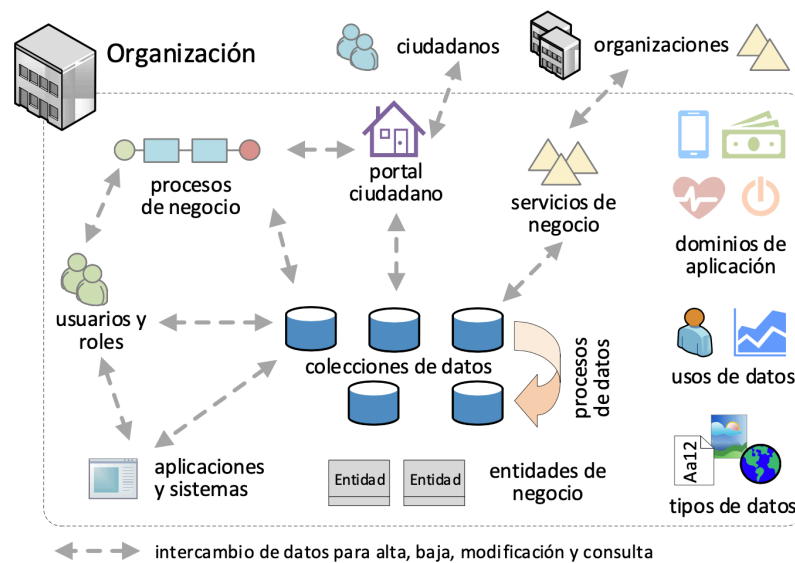


Figura 6: Principales Elementos del Escenario de Trabajo [2]

Los datos en las colecciones pueden ser creados, borrados, modificados y consultados por los distintos tipos de clientes de datos, tanto de forma manual como automática. Los clientes de datos pueden ser aplicaciones, sistemas, procesos de negocio, servicios de negocio, portal ciudadano y procesos de datos.

Las organizaciones pueden operar en distintos dominios de aplicación (por ejemplo salud o energía) y pueden utilizar los datos con distintos fines (por ejemplo operativa de la organización, toma de decisiones).

2.7. Proceso para la Gestión de Calidad de Datos

El proceso de gestión de la calidad propuesto en el framework define un conjunto de actores, roles y etapas involucradas en la gestión de la calidad de los datos en un escenario de gobierno digital. Para aplicar el framework en un determinado escenario, se necesita establecer un Comité de Calidad de Datos, que será el responsable de implementar el proceso de gestión de calidad de datos. En el contexto de una organización se pueden formar distintos comités especializados en diferentes escenarios.

Un comité está compuesto por los siguientes roles:

- **Responsable de Calidad** [9]: responsable de la aplicación del framework en un escenario específico. Es el responsable de todas las etapas del proceso y dirigirá las actividades de los demás miembros del comité.
- **Analista de Calidad de Datos** [9]: analiza los aspectos de calidad de los datos del escenario, como por ejemplo, los requerimientos, los actores relevantes, los problemas de calidad de datos y las métricas de calidad de datos, entre otros.
- **Técnico de Calidad de Datos**: responsable de proporcionar e implementar recursos técnicos para la aplicación del framework. Algunas actividades que debe realizar el rol son: examinar datos objetivo e implementar métodos de medición, entre otros.
- **Experto de Negocio** [9] [10]: se encarga de documentar los requerimientos de negocio y evaluar el impacto de los nuevos requerimientos sobre la calidad de los datos y viceversa. Referente del área de negocio en la cual se enmarca el escenario.
- **Experto Técnico** [10]: se centra en la representación de datos de los diferentes sistemas y aplicaciones. Es el referente de los aspectos técnicos vinculados al escenario.

Además, para avanzar en la gestión de calidad, se espera que exista un Director de Calidad de Datos de la Organización [10], quien se encargará principalmente de designar y liderar un comité para cada escenario.

Finalmente, en lo que respecta al proceso de gestión, está constituido por siete etapas (cada una de las cuales se divide en subetapas). La Figura 7 presenta una visión general de este proceso y sus etapas:

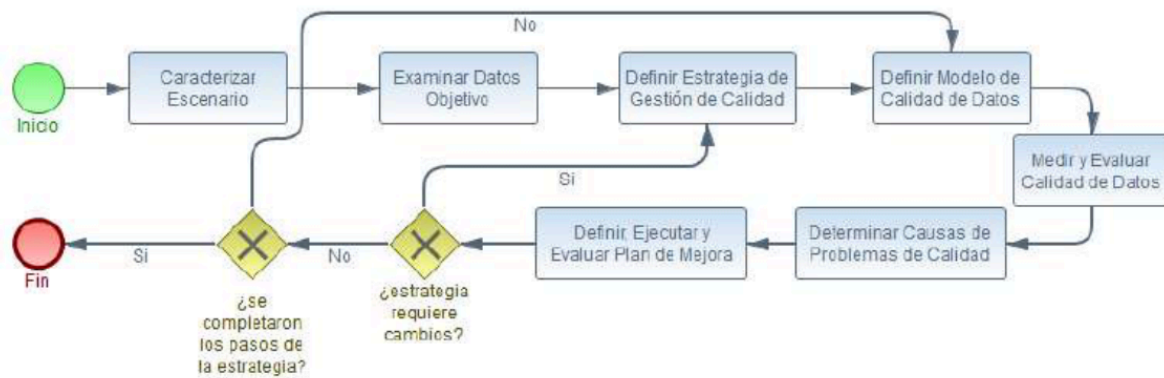


Figura 7: Proceso para la Gestión de Calidad de Datos [2]

A continuación se describe cada una de ellas:

- La etapa **Caracterizar Escenario** tiene como objetivo determinar los elementos relevantes del escenario de trabajo desde un punto de vista técnico (por ejemplo colecciones de datos, aplicaciones, etc) y de la calidad de los datos (por ejemplo problemas de calidad de datos). Estos elementos proporcionan información para las etapas posteriores del proceso.
- El propósito de la etapa de **Examinar Datos Objetivo** es comprender las características de los datos y hacer una estimación preliminar de su calidad, detectar problemas de calidad de los datos y definir nuevos requerimientos de calidad en función de estos problemas.
- La etapa de **Definir Estrategia de Gestión de Calidad** tiene como objetivo definir una estrategia para gestionar la calidad de los datos en un escenario, basada en un conjunto de pasos que abordan un grupo de requerimientos de calidad de datos.
- La etapa **Definir Modelo de Calidad de Datos** tiene como objetivo definir el modelo de calidad de datos para el escenario de trabajo. Este modelo, es construido según los requerimientos de calidad considerados en la estrategia definida.
- El objetivo de la etapa **Medir y Evaluar Calidad de Datos** es realizar mediciones utilizando las métricas y métodos definidos en el modelo de calidad, y evaluar la calidad

de los datos de acuerdo con el perfil de evaluación y las reglas, también definidas en el modelo.

- La etapa **Determinar Causas de Problemas de Calidad** tiene como objetivo determinar las causas de los problemas de calidad ocasionados por las mediciones y evaluaciones realizadas en la etapa anterior. Para esto se pueden tomar como insumo tanto los resultados de las mediciones y evaluaciones, así como los resultados obtenidos de la caracterización del escenario.
- Por último, la etapa **Definir, Ejecutar y Evaluar Plan de Mejora** tiene como objetivo la definición de un plan de mejora para abordar los problemas de calidad detectados, así como la ejecución y posterior evaluación de este plan.

Una vez finalizada la última etapa del proceso, se debe evaluar si la estrategia definida requiere cambios. Si la estrategia no requiere cambios, se debe continuar con el siguiente paso de la misma hasta completar todo lo que define.

Además, el proceso propuesto debe ejecutarse con regularidad para considerar nuevos elementos en el escenario de trabajo y monitorear continuamente la calidad de los datos en el escenario.

3. Análisis

Este capítulo presenta los resultados del análisis realizado en el proyecto, incluyendo una descripción de las características principales de la plataforma, así como también de sus actores, casos de uso y requerimientos.

3.1. Descripción General

Para la etapa de análisis de este proyecto se tomaron como principales insumos la documentación del framework y las reuniones con los tutores. A partir de estos, se identificaron las siguientes características generales de la plataforma, referentes a: Soporte al Proceso de Gestión de Calidad, Soporte para Escenarios de Trabajo en las Organizaciones, Soporte de Comités, Soporte para la Gestión del Modelo de Referencia y Soporte para la Configuración de la Plataforma.

En primer lugar, la plataforma debe dar soporte al proceso de gestión de calidad de datos descrito en el framework, a través de una interfaz web simple y amigable orientada a usuarios con los roles que define el framework (por ejemplo Responsable de Calidad).

También, la plataforma debe dar soporte para que diferentes organizaciones, cada una con sus usuarios, puedan definir sus propios escenarios de trabajo. En cada uno de ellos, deben poder aplicar el proceso de gestión de calidad de datos constituido de siete etapas (cada una formada por un conjunto de subetapas): Caracterizar Escenario, Examinar Datos Objetivo, Definir Estrategia, Definir Modelo de Calidad de Datos, Medir y Evaluar Calidad de Datos, Determinar Causas de Problemas de Calidad y Definir, Ejecutar y Evaluar Plan de Mejoras.

Por otro lado, la plataforma debe dar soporte a los diferentes comités de calidad, que aplican el proceso de gestión para un escenario. Cada usuario integrante de un comité deberá tener diferentes roles sobre el escenario, como por ejemplo: “Responsable de Calidad”, “Analista de Datos”, etc.

Para dar soporte a la gestión del modelo de referencia debe existir la figura del Super Administrador de la plataforma, quien se encargará de gestionar las diferentes versiones del modelo de referencia. A su vez, cada organización debe poder extender este modelo de referencia, añadiendo nuevos elementos de modelo (dimensiones, factores y métricas) de acuerdo a sus necesidades.

Finalmente, en lo que respecta a la configuración de la plataforma, se debe permitir que el Super Administrador registre a las organizaciones que deseen utilizar la plataforma, configure los datos referenciales del framework (por ejemplo, tipos de colecciones de datos) y gestione las diferentes versiones del modelo de calidad.

3.2. Requerimientos

Como resultado del análisis se identificaron los siguientes requerimientos funcionales y no funcionales.

Requerimientos Funcionales:

1. **Soporte a las Distintas Etapas del Framework.** La plataforma debe permitir el ingreso de los distintos datos requeridos por cada etapa del framework, así como también vincular los datos ingresados en las diferentes etapas.
2. **Caracterización Gráfica del Escenario.** En la etapa Caracterizar Escenario, además de permitir el ingreso de los datos de una manera “convencional”, se debe incluir una herramienta de “*Drag & Drop*” para ingresar y relacionar de una manera más interactiva, los diferentes elementos registrados.
3. **Gestión del Modelo de Calidad de Datos de Referencia.** La plataforma debe permitir la gestión de las diferentes versiones del modelo de referencia.
4. **Gestión del Modelo de Calidad de Datos para Organizaciones.** La plataforma debe permitir que cada organización extienda el modelo de referencia, añadiendo sus propios elementos (dimensiones, factores o métricas).

5. **Gestión de los Usuarios en una Organización.** La plataforma debe permitir a una organización, la gestión de sus diferentes usuarios. Además, debe tener la opción de asignar diferentes roles para cada uno de ellos a fin de restringir las diferentes operaciones que los mismos pueden realizar.
6. **Gestión de Información de Configuración.** La plataforma debe permitir realizar configuraciones generales sobre los datos referenciales del framework (técnicas, tipos de relación, tipos de atributos, tipos de colecciones, etc).

Requerimientos No funcionales:

1. La plataforma tiene que estar implementada en Java y utilizar PostgreSQL como manejador de base de datos.
2. La plataforma debe brindar una interfaz de usuario web amigable, orientada para el perfil de usuarios que utilizan el framework.
3. La plataforma debe contar con una arquitectura capaz de ejecutar en un entorno local a una organización o permitir que varias organizaciones vinculadas a gobierno digital, consuman de un servidor central.

3.3. Principales Actores y Casos de uso

De acuerdo al análisis realizado, se identificaron los siguientes actores:

- **Super Administrador.** El administrador general de la plataforma. Su función principal es gestionar el modelo de referencia, así como también configurar los datos referenciales del framework.
- **Administrador.** Es el usuario responsable dentro de una organización y se encarga de la gestión interna de la misma.

- **Comité de Calidad.** El comité de calidad es el encargado de llevar adelante el proceso de gestión de calidad de datos para un escenario en particular.

A continuación se presentan los diagramas de casos de uso del sistema, donde se pueden observar las funcionalidades a las que tienen acceso los distintos tipos de usuario.

La Figura 8 muestra los casos de uso para el Super Administrador, los cuales se agrupan en tres grupos: Usuarios, Modelos y Datos Referenciales.

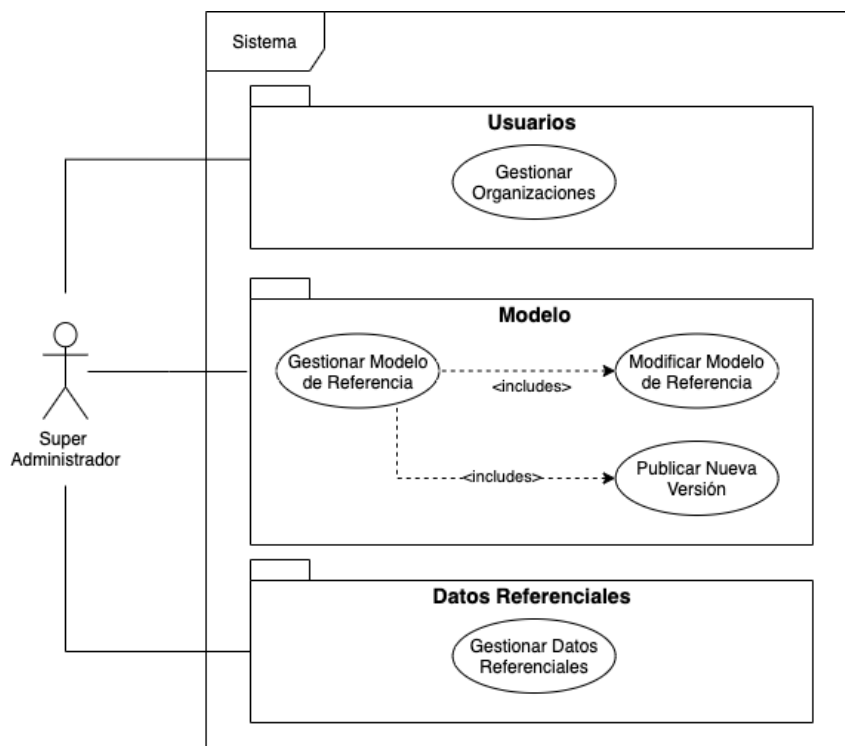


Figura 8: Casos de Uso - Super Administrador

La Figura 9 muestra los casos de uso para el actor Administrador, los cuales se encuentran agrupados en tres conjuntos: Gestión Organización, Gestión Escenarios, y Gestión Modelo.

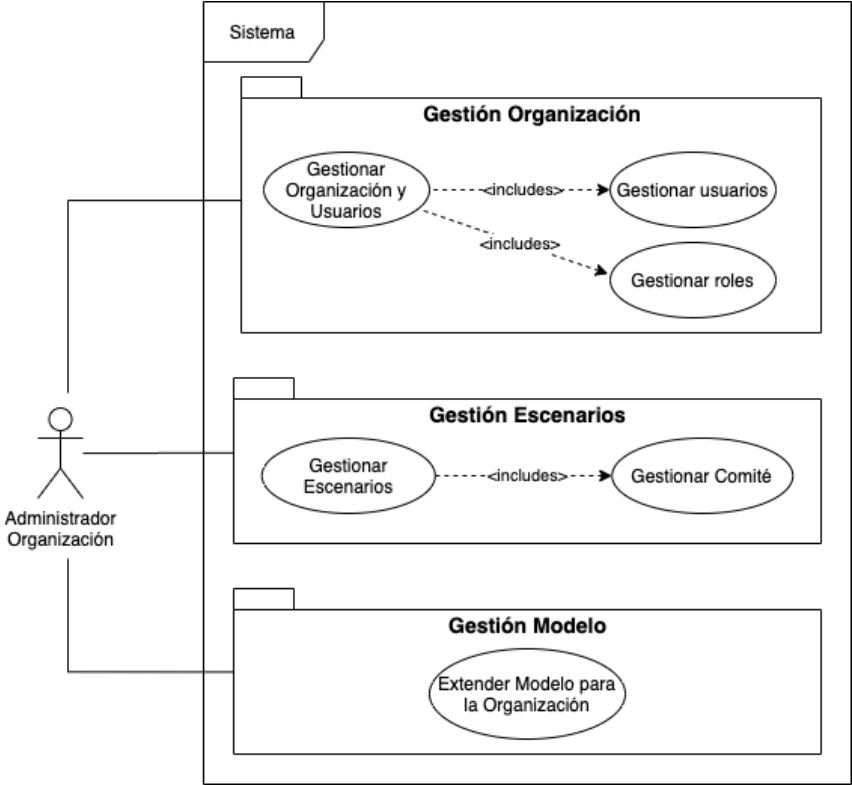


Figura 9: Casos de Uso - Administrador de la Organización

Finalmente, la Figura 10 muestra los diferentes casos de uso para el comité de calidad en cada etapa del proceso.

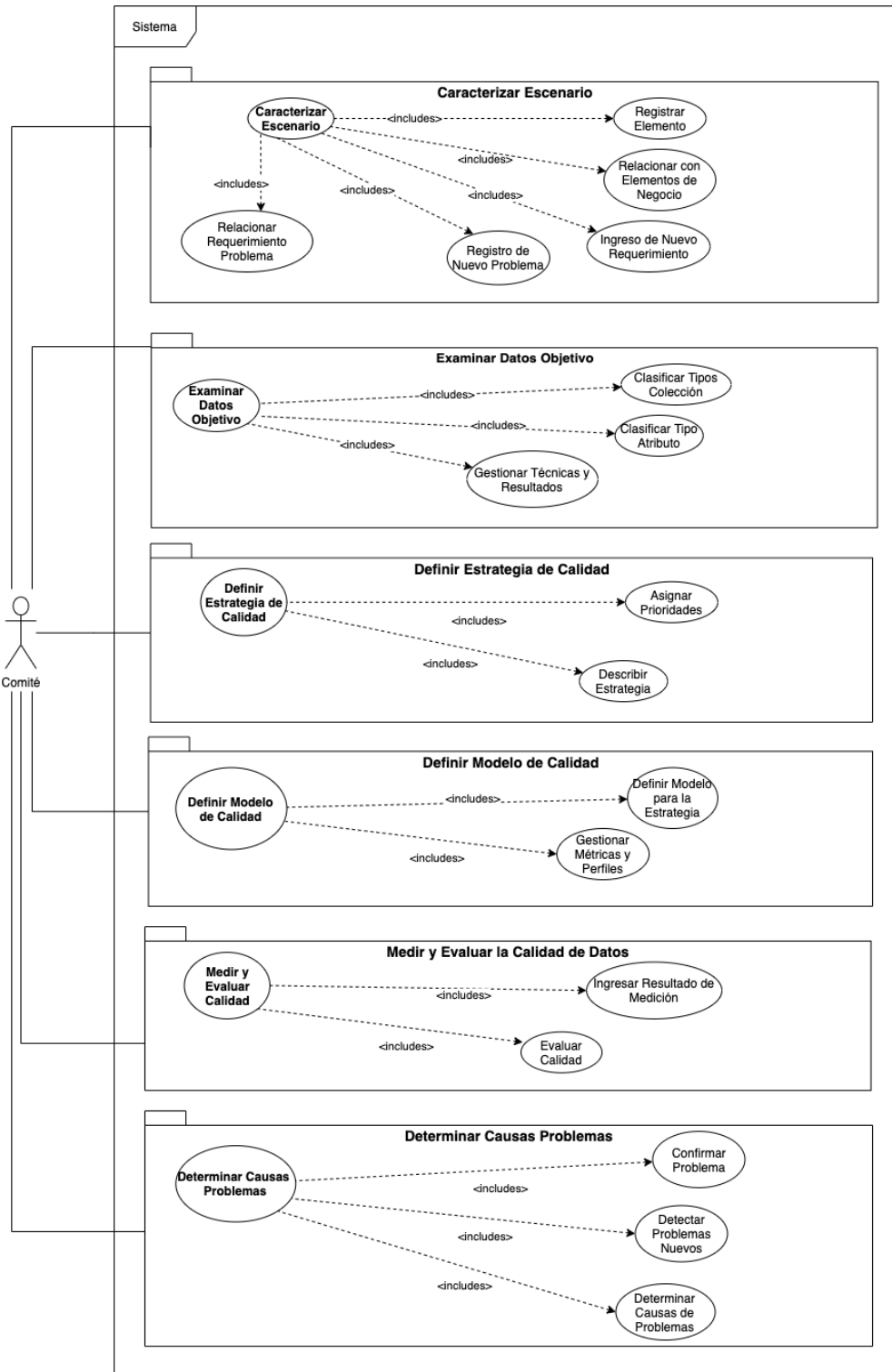


Figura 10: Casos de Uso - Comité

3.4. Decisiones Tomadas

A continuación se presentan las decisiones tomadas durante el proceso de análisis de la solución:

- Considerando el alcance del proyecto, se decidió no abordar la última etapa del proceso (Definir, Ejecutar y Evaluar Plan de Mejora).
- Se analizó cómo abordar las sucesivas modificaciones del modelo de calidad, optando por la inclusión del concepto de “histórico”, el cual refleja los cambios que se realizan sobre un elemento de modelo en una fecha determinada.
- En lo que respecta a los roles se decidió separar lo que son roles de la organización de roles de escenario. Los primeros, tienen permisos a nivel de la organización, mientras que los otros, tienen permisos sobre las diferentes etapas del escenario.
- Para la etapa Caracterizar Escenario, se unificaron las diferentes subetapas que requieren relacionar a los diferentes elementos de negocio. El resultado fue una única subetapa Relacionar Elementos Técnicos. En lo que respecta a la etapa Examinar Datos Objetivo, se separó la subetapa Clasificar Datos Objetivo en: Clasificar Tipos de Colecciones y Clasificar Tipos de Atributo. También, se incorporaron dos nuevas subetapas: Identificar técnicas a Aplicar y Determinar Resultados. Ambas, apuntan a ser un paso previo a Estimar Alcance de Problemas y Estimar Grado de Cumplimiento de Requerimientos. La Tabla 1 presenta las etapas y subetapas resultantes:

Tabla 1: Etapas y subetapas

Etapas	Subetapas
Caracterizar Escenario	Registrar Elementos Relacionar Elementos Técnicos Ingresar Nuevos Requerimientos Relacionar Requerimientos Interés Actores Relacionar Requerimientos con Elementos Técnicos Registrar Nuevos Problemas Relacionar Problemas con Actores Relacionar Requerimiento con Problemas
Examinar Datos	Clasificar Tipos Colecciones Clasificar Tipos Atributos Identificar Técnicas a Aplicar Determinar Resultados Estimar Alcance de Problemas Estimar Cumplimiento Requerimientos Detectar Nuevas Problemas Definir Nuevo Requerimiento
Definir Estrategia de Calidad	Asignar Prioridades Describir Estrategias
Definir Modelo de Calidad	Definir Modelo para la Estrategia Ingresar Métrica Ingresar Perfil
Medir y Evaluar la Calidad de Datos	Ingresar Resultado de Medición Evaluar Calidad
Determinar Causas Problemas	Confirmar Problemas Detectar Problemas Nuevos Determinar Causas de Problemas

4. Solución Propuesta

En este capítulo se presenta una descripción general de la plataforma propuesta y se presentan detalles tanto de su arquitectura general como de sus principales funcionalidades.

4.1. Descripción General

En la Figura 11, se presenta un diagrama general de la plataforma, donde se pueden observar los componentes principales que la conforman y la interacción entre ellos:

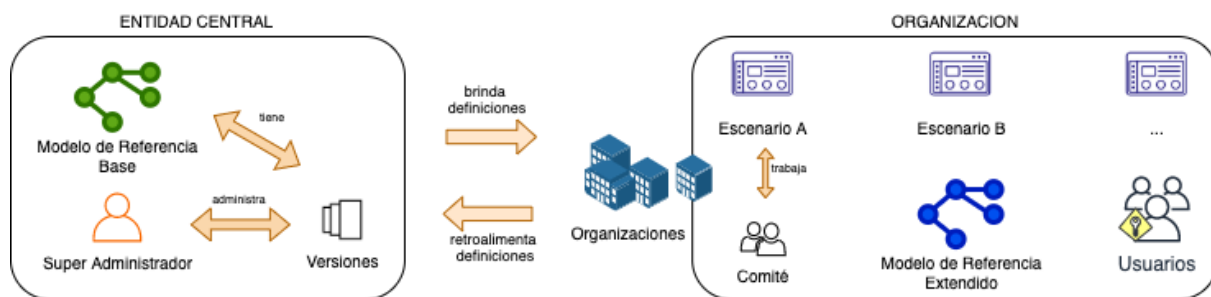


Figura 11: Diagrama General de la Plataforma

Se pueden distinguir dos grandes componentes de la plataforma: la entidad central y las organizaciones. Por un lado, la entidad central es la que provee a las demás organizaciones de versiones del modelo de referencia, técnicas genéricas, tipos de colecciones y más datos referenciales. Las organizaciones por su lado, “retroalimentan” estas “definiciones” que pueden ser de utilidad para otras organizaciones, por ejemplo sugiriendo extensiones al modelo de referencia.

El Super Administrador es quien se encarga, entre otras cosas, de definir las versiones del modelo de referencia que se presenta a las demás organizaciones. Cada versión es un subconjunto de elementos del modelo (por ejemplo, dimensiones, factores).

Las organizaciones están formadas por un conjunto de usuarios, cada uno con roles que determinan a qué funcionalidades tienen acceso los mismos. Además, pueden extender el modelo

de referencia, creando sus propias dimensiones, métricas o factores. Este modelo extendido, es visible para todos los escenarios que existan en la organización.

Las organizaciones tienen varios escenarios donde, en cada uno de ellos, el comité es el encargado de llevar adelante el proceso de gestión de calidad de datos. El comité está integrado por algunos de los usuarios pertenecientes a la organización.

4.2. Arquitectura

La arquitectura planteada permite dos alternativas. Por un lado, que sea una organización central quien se encargue de la administración de la plataforma, brindando un punto de acceso único para el resto de las organizaciones. Por otro lado, las diferentes organizaciones pueden ejecutar la plataforma en su ambiente local, donde el Super Administrador de la plataforma sería alguien interno a la organización.

La Figura 12 presenta la arquitectura lógica de la plataforma.

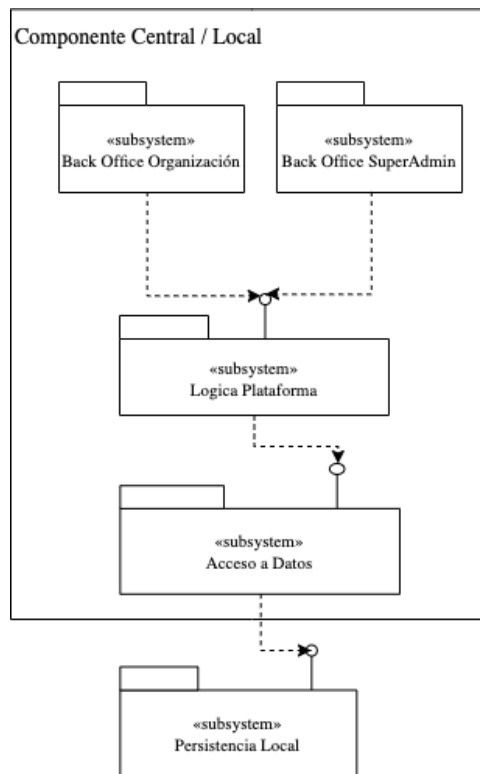


Figura 12: Arquitectura Lógica de la Plataforma

Los componentes representados son:

- **Componente Central / Local.** Incluyen los siguientes subsistemas:
 - **Componente Back Office Organización:** Este componente es responsable de los servicios referentes a las organizaciones (por ejemplo usuarios, escenarios, roles, modelo extendido). Se comunica con la “Lógica Plataforma”.
 - **Componente Back Office SuperAdmin:** Este componente es responsable de los servicios de gestión de los diferentes componentes del sistema (por ejemplo usuarios, modelo de referencia, versionado, datos referenciales). Se comunica con la “Lógica Plataforma”.
 - **Componente Lógica Plataforma:** Este componente es responsable de la lógica de la plataforma (por ejemplo sincronización de la representación gráfica, evaluación

de perfiles, importar/exportar, etc). Se comunica con la lógica de Accesos a Datos.

- **Componente de Acceso a Datos:** Este componente es responsable de la obtención y persistencia de datos (acceso a base de datos).

Por otro lado, la Figura 13 y 14 presentan las dos alternativas de arquitectura técnica de la plataforma, indicando los nodos presentes en cada infraestructura y la localización de los componentes en dichos nodos.

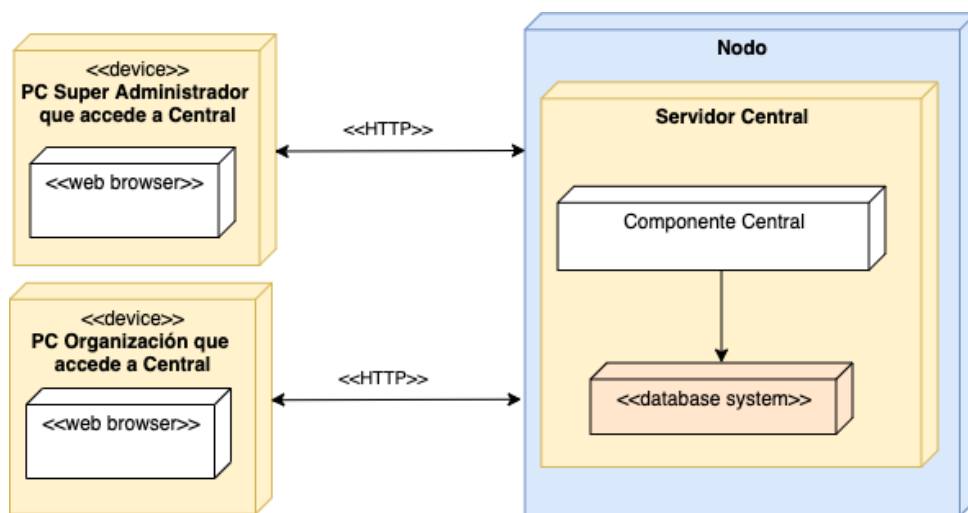


Figura 13: Infraestructura de la Plataforma Centralizada

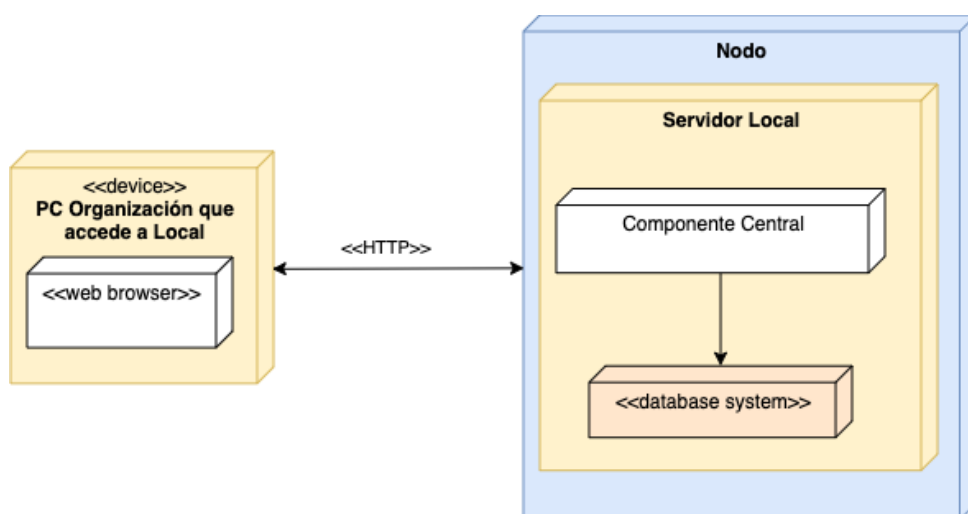


Figura 14: Infraestructura de la Plataforma Local

En el caso de la infraestructura centralizada, se identifican los siguientes nodos:

- **PC Super Administrador que accede a Central (cliente web):** En este nodo se ejecuta un navegador web mediante el cual el Super Administrador de la plataforma, inicialmente AGESIC, accede al sistema que brinda las funcionalidades del Back Office Super Administrador.
- **PC Organización que accede a Central (cliente web):** En este nodo se ejecuta un navegador web mediante el cual los usuarios pertenecientes a una organización acceden al sistema que brinda las funcionalidades del Back Office Organización en el Componente Central.
- **Servidor Central:** En este nodo se mantiene el servidor central y se ejecuta en un contenedor de aplicaciones web. El Componente Central brinda una web para los usuarios, la cual es accedida desde un navegador web. A su vez, el Servidor Central se comunica con el sistema de almacenamiento.

En el caso de la infraestructura local, se identifican los nodos:

- **PC Organización que accede a Local (cliente web):** En este nodo se ejecuta un navegador web mediante el cual los usuarios de una organización acceden al sistema que brinda las funcionalidades tanto del Back Office Organización como del Back Office Super Admin en el Componente Central.
- **Servidor Local:** En este nodo se ejecuta el Componente Central. El componente brinda una web para sus usuarios, la cual es accedida desde un navegador web y permite gestionar la base de datos de manera local.

4.3. Principales Funcionalidades

Esta sección describe los aspectos que se consideran más relevantes de la plataforma en el aspecto funcional, entre ellos el Control de Acceso, Representación Gráfica, Gestión del Modelo de Calidad, Gestión de Versiones del Modelo y Gestión de los Datos Referenciales.

4.3.1. Control de Acceso

La plataforma presenta un esquema para el manejo y control de permisos y roles dentro de una organización (y por lo tanto, dentro de un escenario). Esto permite asignar a un usuario diferentes roles, cada uno de ellos asociados a un conjunto de permisos preexistentes, a fin de restringir las funcionalidades habilitadas para cada uno de ellos en el contexto de la organización.

En principio, se tiene al usuario **Super Administrador**, quien refiere al administrador global de la plataforma. Entre otras cosas, es quien se encarga de la creación de nuevos roles, los cuales están disponibles para todas las organizaciones y, de esta forma, se le pueden asignar diversos roles a los usuarios integrantes de las mismas.

Además, este usuario es responsable de registrar a las organizaciones que desean utilizar la plataforma. Al momento de registrar una nueva organización, el Super Administrador debe especificar el usuario **Administrador** de la misma. Este usuario cuenta con una visión general y con permisos totales para una correcta gestión dentro de la organización. En otras palabras, tiene los permisos relacionados a la gestión de la organización, del modelo y de los diferentes usuarios que forman parte de la organización.

En lo que respecta a los roles y permisos, cada usuario de una organización, a excepción del administrador, puede tener: diversos roles para cada comité para el cual sea miembro y/o roles más “administrativos” dentro de la organización.

El **Comité de Calidad**, es el responsable de aplicar el proceso de gestión de calidad de datos para un determinado escenario. Los integrantes de este comité, son quienes tienen permisos sobre las

diferentes etapas del proceso de gestión, lo cual influye en la forma en la que interactúan con la plataforma en las diversas etapas. Cada integrante puede tener diversos roles asignados dentro de un comité (cada uno con determinados permisos). La Figura 15 representa este esquema.

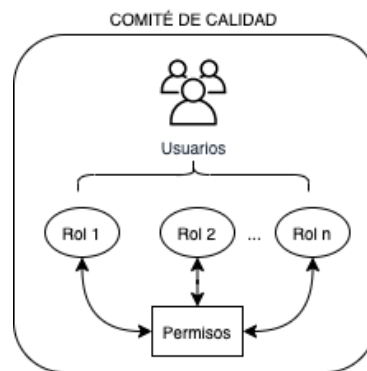


Figura 15: Comité y Roles

Para cumplir con esta funcionalidad, en la plataforma se encuentra definido un permiso para cada una de las etapas existentes en el proceso de gestión de calidad de datos. Cada uno de estos permisos habilita a operar en la etapa correspondiente. A modo de ejemplo, cuando el Super Administrador ingrese en el sistema el rol Responsable de Calidad, sería esperable que le asignase cada uno de los permisos de las etapas (dado que este rol tiene acceso sobre todas ellas).

Dentro de un comité se debe hacer una distinción entre permisos para “lectura” o “lectura y escritura”. Cuando un usuario pasa a formar parte de un comité para un escenario, automáticamente se le otorga permisos para poder visualizar (no editar) toda la información de cada una de las etapas del escenario. Si se quisiera que este usuario también tuviese permisos para manipular la información de alguna de las etapas, se le debe asignar un rol dentro de ese comité. De esta forma, la próxima vez que el usuario ingrese a un escenario para el cual forma parte del comité, puede registrar, editar o eliminar la información en alguna o todas las etapas, según le habilite los permisos asociados a su rol.

En el contexto de una organización, además de los permisos para cada etapa, un usuario puede

contar con roles asociados a **permisos administrativos**. Estos permisos confieren al usuario la posibilidad de realizar tareas vinculadas a la administración de la organización. Existen tres permisos de este tipo:

- **Gestión de escenarios:** Además de otorgar al usuario permisos de edición en todos los escenarios (y en sus etapas), lo habilita a crear, editar o eliminar escenarios dentro de la organización. También le permite manipular los diferentes comités de los escenarios.
- **Gestión del modelo:** permite extender el modelo de referencia mediante la creación, edición o eliminación de nuevas dimensiones, factores, o métricas.
- **Gestión de organización:** relacionado con la estructura de la organización, permite registrar, editar o eliminar usuarios en la organización.

Por lo tanto, la plataforma permite crear y asignar diferentes roles a los integrantes de una organización, con el objetivo de ampliar sus funcionalidades disponibles sobre diferentes partes del sistema.

4.3.2. Representación Gráfica

El objetivo de la etapa Caracterizar Escenario es identificar los elementos relevantes para el escenario, tanto desde un punto de vista técnico y de negocio como desde un punto de vista de la calidad de datos. Se deben identificar por ejemplo, aplicaciones que procesan datos, organizaciones con las que se intercambian datos y colecciones de datos que utilizan estas aplicaciones. La Figura 16 presenta algunos de los conceptos del escenario de trabajo relacionados a organizaciones, colecciones de datos y clientes de datos. Por más detalles del marco conceptual ver [2].

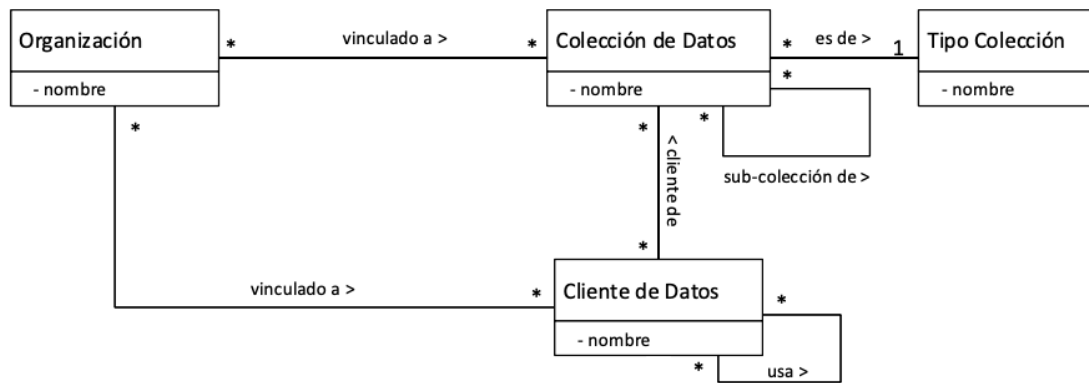


Figura 16: Organizaciones, Colecciones de Datos y Clientes de Datos [2]

El concepto de Organización refiere a aquellas organizaciones involucradas en un escenario de trabajo. Las mismas, se vinculan a colecciones de datos, por ejemplo, porque son responsables de los datos o porque consultan los datos.

Las Colecciones de Datos refieren a conjuntos de datos que pueden estar almacenados de distinta forma o siendo transferidos de un cliente de datos a otro.

Los Tipos de Colección refieren a la forma en que las colecciones de datos se almacenan o acceden.

Los Clientes de Datos refieren a distintos tipos de entidades que realizan operaciones sobre las colecciones de datos.

Tabla 2: Ejemplo

Organizaciones	Ministerio de Salud Pública (MSP) Dirección Nacional de Identificación Civil (DNIC)
Colecciones de Datos:	Certificados de Nacido Vivo (CNV). Tipo: Base de Datos Relacional. Datos Básicos de Ciudadanos (DBC). Tipo: Flujo de Datos.
Clientes de Datos:	Certificado de Nacido Vivo Electrónico (apCNVE) Servicio Básico de Información (servBI)
Relaciones	Organización MSP vinculada a cliente servBI Organización MSP vinculada a cliente apCNVE Organización DNIC vinculada a cliente servBI Cliente apCNVE es cliente de colección CNV Cliente servBI es cliente de colección DBC

La Tabla 2 representa un ejemplo en el que se muestran dos organizaciones que participan en el escenario: Ministerio de Salud Pública (MSP) y la Dirección Nacional de Identificación Civil (DNIC). Dentro de las colecciones de datos involucradas en el escenario hay una base de datos relacional, que almacena información de los certificados de nacido vivo generados, y un flujo de datos que se origina en la invocación de un servicio provisto por la DNIC. Los clientes de datos del escenario son la aplicación del MSP para generar estos certificados (apCNVE) y el Servicio Básico de Información (servBI) de la DNIC.

Para la etapa Caracterizar Escenario, la plataforma propone dos posibilidades para registrar (o editar) los elementos y las relaciones entre los mismos. Una de ellas es un formulario clásico con campos a rellenar. La otra refiere a la utilización de un diagrama, con características de “*Drag & Drop*”. Esto apunta a brindar una interacción más directa e intuitiva con el usuario. La Figura 17 presenta la propuesta de representación gráfica para el ejemplo de la Tabla 2.

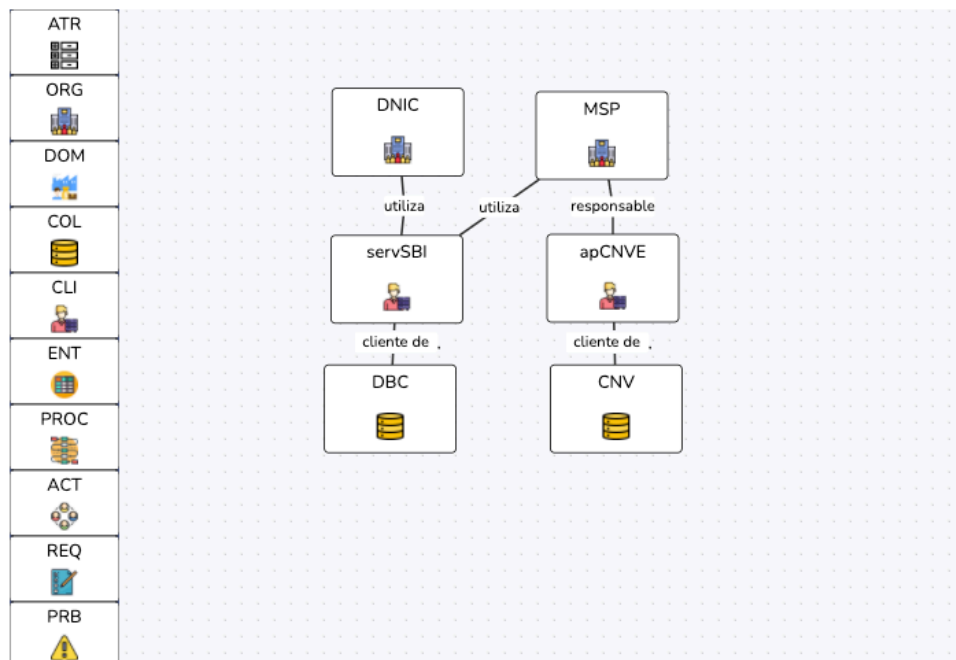


Figura 17: Propuesta de Representación Gráfica para el Ejemplo de Tabla 2

En el diagrama, se propone que los elementos del escenario, sean representados mediante “cajas” con un tamaño predeterminado. Por otro lado, las relaciones son representadas mediante “líneas” que unen las “cajas”. Cada elemento que aparece en el diagrama es interactivo, lo que permite al usuario poder desplazar cualquiera de ellos y posicionarlos donde desee.

Por lo tanto, el fin principal de esta funcionalidad es ofrecer al usuario un panorama general del escenario de trabajo, permitiéndole manipular los elementos de una manera más sencilla y rápida en comparación a un formulario habitual.

Al ser ésta una funcionalidad más “libre” para el usuario, es necesario que la plataforma realice controles en el diagrama a efectos de asegurar su correcto funcionamiento y ajustarse a las restricciones definidas por el marco conceptual en el framework.

4.3.3. Gestión del Modelo de Calidad

Como se presenta en la Figura 18, el modelo de referencia está compuesto por un conjunto de elementos de modelo. Un elemento de modelo refiere a una dimensión, factor o métrica, cada uno con sus atributos correspondientes.

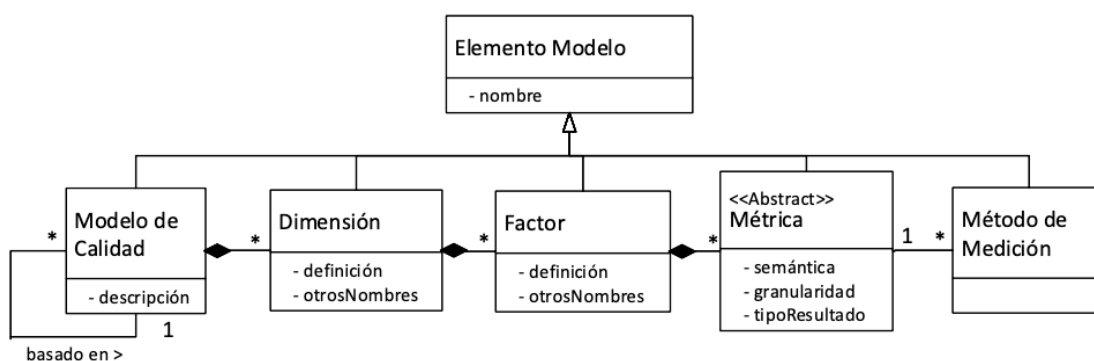


Figura 18: Elementos del Modelo [2]

El Super Administrador es quien se encarga de definir el modelo de referencia, en el cual las diferentes organizaciones se basan al momento de aplicar el proceso en sus diferentes escenarios. Por ejemplo, este usuario puede cargar en la plataforma las dimensiones, factores y métricas del modelo que se presenta en la Sección 2.5.

Por otro lado, solamente aquellos con los permisos suficientes pueden “extender” este modelo para su correspondiente organización. La extensión del modelo por parte de una organización implica añadir sus propios elementos de modelo, que son visibles para los miembros de la organización y por lo tanto, pueden ser utilizados en el proceso de gestión.

Es esperable que los atributos de los elementos de modelo varíen con el tiempo. Por ejemplo, el nombre de una dimensión podría cambiar luego de cierto período. Con el fin de contar con un registro de todos los cambios que se le realizan a cada elemento, surge el concepto de “histórico”.

Un histórico refleja los cambios que se realizaron a un elemento de modelo, en una fecha de-

terminada. De esta manera, la plataforma permite acceder y utilizar los diferentes históricos de una dimensión, factor o métrica, para definir el modelo de referencia.

La Figura 19 muestra la representación de un modelo de calidad compuesto por varios elementos de modelo, donde cada uno presenta un conjunto de históricos.

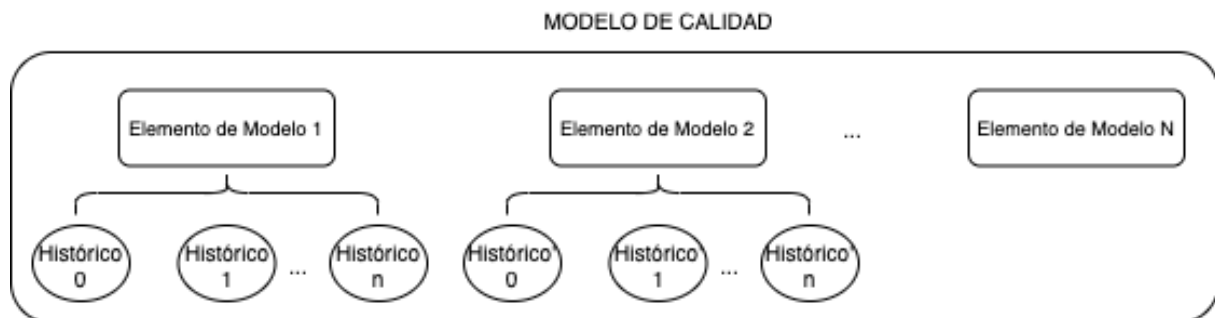


Figura 19: Modelo de Calidad

4.3.4. Gestión de Versiones del Modelo

Como se mencionó previamente, los elementos del modelo pueden cambiar a lo largo del tiempo. Algunos atributos de los elementos que podrían cambiar son: el nombre, la definición, su elemento “padre”. Pero no solamente los elementos de modelo cambian, sino que también lo hace el modelo de referencia que utilizan las organizaciones en sus escenarios.

Así como se tiene el concepto de histórico para elementos de modelo, se tiene el de versiones para el modelo de referencia. Las versiones son gestionadas únicamente por el Super Administrador, y refieren a un conjunto de históricos de elementos de modelo que conforman el modelo de referencia. El modelo de referencia que se le presenta a las organizaciones, corresponde a una versión en particular del modelo de referencia.

En la Figura 20 se muestra la representación de una versión, la cual tiene un conjunto de históricos donde cada uno forma parte de un elemento de modelo específico.

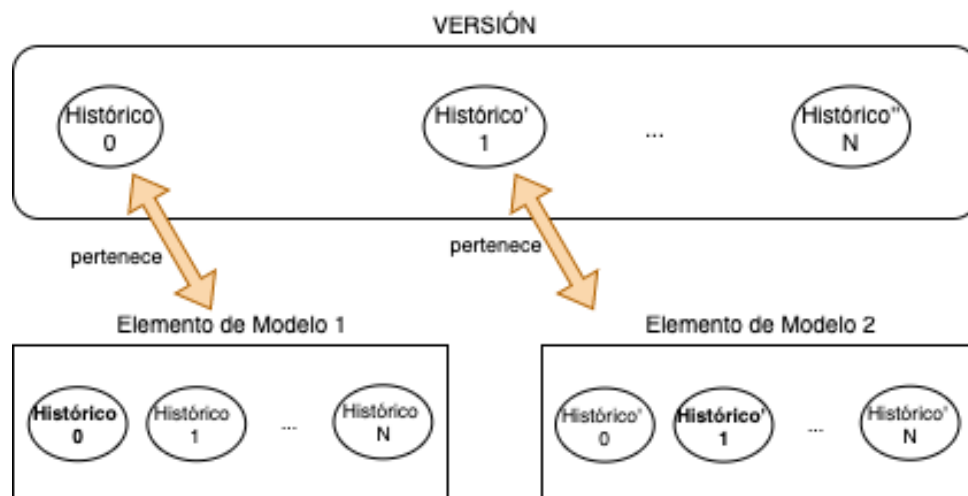


Figura 20: Versiones de Modelo

En lo que respecta a las versiones, el Super Administrador podrá:

- Crear una nueva versión del modelo
- Editar o Eliminar una versión existente
- Publicar una versión ya existente

Crear una nueva versión, implica entre otras cosas, seleccionar qué históricos de elementos de modelo forman parte de la versión. También en el caso de querer añadir un histórico de un elemento de modelo que no exista en el modelo de referencia, se da la opción para registrar un nuevo elemento de modelo directamente sobre la versión (el cual es añadido al modelo de referencia).

Una vez creada la versión, para que se encuentre disponible en las demás organizaciones, se debe publicar. Esto significa, que la versión queda disponible para el resto de las organizaciones integradas a la plataforma.

4.3.5. Gestión de los Datos Referenciales

Para las diferentes etapas del proceso, el framework define un conjunto de datos referenciales. Los datos referenciales definen un conjunto de valores posibles para un dominio. Por ejemplo, en la etapa Examinar Datos Objetivos, se definen los siguientes tipos de colecciones de datos: Base de Datos Relacional, Base de Datos Documental o Flujo de Datos. Estos tipos pueden además variar en el tiempo.

Estos datos referenciales deben ser ingresados a la plataforma por el Super Administrador. Una vez creados, se encuentran habilitados para que todas las organizaciones los puedan utilizar en las diferentes etapas del proceso según corresponda. La Figura 21 presenta un resumen de los diferentes tipos de datos referenciales.

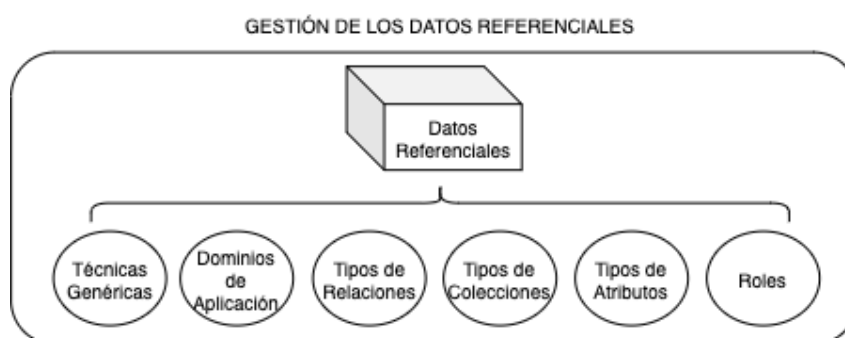


Figura 21: Tipos de Datos Referenciales

El primero de ellos, corresponde a los diferentes **Dominios** (por ejemplo, salud, energía, educación, etc) en los que pueden operar las diferentes organizaciones que se registran en la primera etapa del proceso. Los diferentes dominios que ingrese el Super Administrador, son sugeridos a un comité al momento de registrar un nuevo elemento (de tipo dominio) en la etapa Caracterizar Escenario.

También, durante la primera etapa del proceso (principalmente), es muy común que los diferentes elementos identificados (organizaciones, dominios, clientes de datos, etc) se relacionen entre sí. Los posibles valores que pueden adquirir estas relaciones difieren según el tipo de elemento.

El Super Administrador puede registrar diversos **Tipos de Relaciones** que son las opciones con las que dispone un comité al momento de relacionar dos elementos. Por ejemplo: una organización “opera” en un dominio de aplicación.

Otros tipos de datos preexistentes que pueden extenderse con el tiempo, son los **Tipos de Colecciones** y **Tipos de Atributos**. Ambos, son utilizados en la etapa examinar datos objetivo para especificar los tipos de las colecciones de datos o atributos identificados en la caracterización de escenario.

En la misma etapa, se aplican las **Técnicas** definidas de *Data Profiling* sobre las diferentes colecciones de datos para obtener información relevante. El Super Administrador es el encargado de crear las diferentes Técnicas Genéricas que las demás organizaciones utilizarán en sus escenarios en esa etapa.

Finalmente, los **Roles** son los diferentes cargos que pueden existir en una organización y en un comité. Como se mencionó, cada rol tiene un conjunto de permisos asociados. Estos permisos pueden estar relacionados al comité de calidad o a la organización. El Super Administrador es el encargado de crear los diferentes roles que las demás organizaciones asignan luego a sus usuarios.

5. Implementación

Este capítulo se centra en la implementación del prototipo de la plataforma. En particular, se describen las herramientas y tecnologías utilizadas, y se detallan aspectos clave de las principales funcionalidades.

5.1. Herramientas y Tecnologías

La plataforma se implementó en Java, utilizando el entorno de desarrollo Eclipse Java EE IDE². El kit de desarrollo de Eclipse incluye las herramientas de desarrollo de Java lo que permite técnicas avanzadas como refactorización y análisis de código.

Como sistema de gestión de bases de datos se utilizó PostgreSQL³, principalmente porque se trata de un proyecto de “software libre” y es una herramienta muy flexible que ofrece un potente conjunto de funciones.

Finalmente, para la implementación de la interfaz web se utilizó HTML, CSS y JavaScript. También, librerías como Wicket, Bootstrap y JointJs. Estas tecnologías se incorporaron en pos de facilitar la programación y obtener mejores resultados.

5.1.1. Wicket

Wicket⁴ es un framework de aplicaciones Java, “software libre”, orientado a componentes y “*server-side*”. A continuación se presentan las principales ventajas de utilizar Wicket [11]:

- Las páginas web no son simplemente archivos de texto enviados al cliente, sino que son instancias de objetos, con lo cual se puede aprovechar el paradigma orientado a objetos para diseñar páginas web y sus componentes.

²Eclipse: <https://www.eclipse.org/downloads/>.

³PostgreSQL: <https://www.postgresql.org/>.

⁴Wicket: <https://wicket.apache.org/>.

- 100 % Software libre: Wicket es un proyecto de Apache que no depende de ninguna compañía privada.
- Wicket se centra en Java y en HTML: la gran mayoría de los frameworks web, fuerzan al usuario a adoptar etiquetas especiales o a usar código del lado del servidor dentro del “*HTML markup*”, contrastando con el concepto de la separación entre la presentación y la lógica de diseño. En Wicket, los mundos de HTML y Java son paralelos y se unen por los “identificadores Wicket”, que son los atributos en HTML y propiedades de los componentes en Java.

En el marco de este proyecto, se utilizó Wicket como lenguaje de desarrollo principal, utilizándose para todo lo que respecta a la lógica de los componentes web del lado del servidor y llamadas AJAX, entre otros.

5.1.2. Bootstrap

Bootstrap⁵ es un framework para el desarrollo de sitios web que permite simplificar el proceso de maquetación [12]. A su vez, la existencia de sus componentes web (tablas, menús, botones, etc) preexistentes facilita aún más el desarrollo.

Otra de sus ventajas, es que ofrece las herramientas para que la visualización del sitio web se adapte a cualquier dispositivo.

En el marco de este proyecto, se utilizó Bootstrap para el desarrollo de la interfaz web, lo cual sirvió de guía para aplicar buenas prácticas y diferentes estándares.

5.1.3. JointJs

Una de las funcionalidades más importantes de la plataforma es contar con la opción de “*Drag & Drop*” para facilitar alguna de las etapas del framework. De esta manera, el usuario tiene la

⁵Bootstrap: <https://getbootstrap.com/>.

posibilidad de interactuar con los elementos de una manera más sencilla.

Para este trabajo se investigaron y tuvieron en cuenta varias librerías, sin embargo se encontró que no todas permiten un alto grado de personalización de las funcionalidades. A su vez, estaba el desafío de encontrar una librería que fuese de “software libre”.

JointJs⁶ es la librería que se utilizó para implementar la funcionalidad. Tiene una versión paga (Rappid) que se basa completamente en su versión gratuita (JointJs). Son varias las funcionalidades que pueden desarrollarse con JointJs, debido a que está implementado puramente en JavaScript, y ofrece a los desarrolladores un alto grado de personalización de los elementos y eventos.

Entre sus funciones principales se destacan:

- Es de “software libre” y su gran comunidad otorga una amplia cantidad de ejemplos para diferentes funcionalidades.
- Permite renderizar cientos de elementos y links con interacción instantánea.
- Cuenta con un conjunto de figuras por defecto para utilizar: rectángulo, círculo, elipse, texto, imagen, path, gráficos, etc.
- Los links y las figuras son personalizables, a los cuales se les puede modificar su estructura, color, tamaño, etc.
- Permite ubicar los elementos y links de manera automática sobre el “*canvas*”.
- Reacciona ante un numeroso conjunto de eventos que ocurren sobre el “*canvas*”
- Provee dos funciones con las cuales se puede cargar un diagrama a partir de un archivo JSON y obtener el JSON correspondiente a un diagrama.

Por otro lado, se tuvo en cuenta la librería GoJs⁷, la cual demostró un alto rendimiento y grado

⁶JointJs: <https://www.jointjs.com/opensource>.

⁷GoJs: <https://gojs.net/latest/index.html>.

de personalización. Sin embargo debió descartarse, ya que si bien permite su ejecución en un entorno local, se requiere de una licencia para ejecutarlo en un dominio público.

Otras librerías como MxGraph⁸, Rete⁹, Mermaid.js¹⁰, Cytoscape¹¹ fueron investigadas y probadas, pero se descartaron ya que no contaban con la libertad suficiente en el desarrollo de interacciones o personalización de elementos como los comentados anteriormente.

5.1.4. Otras Herramientas

Otras herramientas que fueron utilizadas en el desarrollo del proyecto son:

- Google Drive¹²: utilizada para especificar los casos de uso. También, para registrar los requerimientos que surgieron de las diferentes reuniones.
- Draw.io¹³: es una herramienta para la creación y edición de diagramas. Se utilizó para desarrollar los diagramas para el modelo de dominio, casos de uso, despliegue, etc.
- Figma¹⁴: es una aplicación para diseñar interfaces que se ejecutan en el navegador, brindando un conjunto extenso de herramientas para la fase de diseño gráfico de un proyecto. En el Apéndice C se brindan detalles de los mockups iniciales.
- Trello¹⁵: permitió la gestión del proyecto y la organización en el equipo.
- Medios de comunicación: los servicios de videoconferencia, como Skype¹⁶ o Zoom¹⁷, fueron utilizados para las reuniones con los tutores. Por otro lado, el mail fue utilizado principalmente para la agenda de reuniones o el intercambio de archivos.

⁸MxGraph: <https://jgraph.github.io/mxgraph/>.

⁹Rete: <https://rete.js.org/>.

¹⁰Mermaid.js: <https://mermaid-js.github.io/mermaid/>.

¹¹Cytoscape: <https://cytoscape.org/>.

¹²Google Drive: <https://www.google.com/intl/es/drive/>.

¹³Draw.io: <https://www.diagrams.net/>.

¹⁴Figma: <https://www.figma.com/>.

¹⁵Trello: <https://trello.com/es>.

¹⁶Skype: <https://www.skype.com/es/>.

¹⁷Zoom: <https://zoom.us/>.

5.2. Control de Acceso

La plataforma presenta un conjunto de vistas a las cuales no todos los usuarios pueden acceder. Por ejemplo, la vista de “usuarios”, donde se listan todos aquellos que pertenecen a la misma pero solamente se encuentra visible para aquellos que tengan permiso de Gestión Organización.

En la misma línea, la vista “modelo” que presenta el modelo de referencia de la organización, se encuentra pública para todos los miembros de un organismo pero solamente aquellos con permiso de Gestión Modelo son capaces de extender al mismo.

A su vez, cualquier vista relacionada a un escenario está disponible únicamente para los integrantes del comité o para los usuarios con permiso Gestión Escenarios, lo cual no implica que todos pueden manipular la información.

De esta forma, para cada una de las vistas de la plataforma que requieran algún tipo de permiso, se necesita realizar una validación para restringir el acceso cuando corresponda. Por lo tanto, en cada una de ellas se invoca una función para determinar si el usuario que está intentando ingresar a la vista puede continuar o no. Uno de los parámetros que recibe esta función, es el tipo de permiso necesario para acceder a la información de la vista correspondiente. De esta manera, si el usuario tiene un rol asociado al permiso que la función recibe, significa que puede continuar.

Con respecto a los permisos de edición sobre las diferentes vistas de las etapas o subetapas, en cada una de ellas se utilizan componentes web con comportamiento de “habilitado/deshabilitado”. De esta forma, todos los usuarios del comité de ese escenario pueden ver la información de las diferentes etapas, pero solamente pueden editar la información si:

- Es el administrador de la organización.
- El usuario tiene algún rol que contiene el permiso específico para esa etapa.

Otra verificación que se debe llevar a cabo, es la que refiere a los parámetros de la URL. En diversas ocasiones, el identificador del escenario o la organización, se encuentra visible en la URL, para obtener la información correspondiente a los mismos. En estos casos, el usuario podría manipular la URL y acceder a un escenario u organización para el cual no tiene permiso.

Con el fin de controlar estas situaciones:

- El identificador de la organización en la URL, debe ser el mismo que el de la organización a la cual pertenece el usuario.
- El identificador del escenario en la URL, debe ser el mismo que alguno de los escenarios de la organización del usuario, para los cuales tiene acceso.

En caso de no cumplirse alguna de estas dos condiciones, se redirige al usuario al inicio.

Finalmente, en lo que respecta a las vistas del Super Administrador, se asegura que solamente él pueda acceder a estas funcionalidades. Si un usuario (no Super Administrador) intenta acceder a alguna de estas vistas, será redirigido al inicio.

5.3. Representación Gráfica

Como se mencionó anteriormente, los usuarios cuentan con un diagrama que permite representar gráficamente a los elementos del escenario. La interfaz del diagrama se encuentra dividida en dos secciones. En la sección de la izquierda, se incluye una paleta donde se listan los diferentes tipos de elementos que se manejan dentro de la primera etapa del proceso de gestión de calidad de datos. Por razones de diseño y simplicidad, cada uno de ellos se muestra con el tipo de elemento de forma abreviada (además de su imagen representativa). Sobre la derecha se encuentra el “*canvas*” donde se muestran todos los elementos que se han ido registrando en el escenario. Los elementos del escenario se representan mediante “cajas” y las relaciones con “líneas”.

La Figura 22 presenta la interfaz del diagrama descripto.

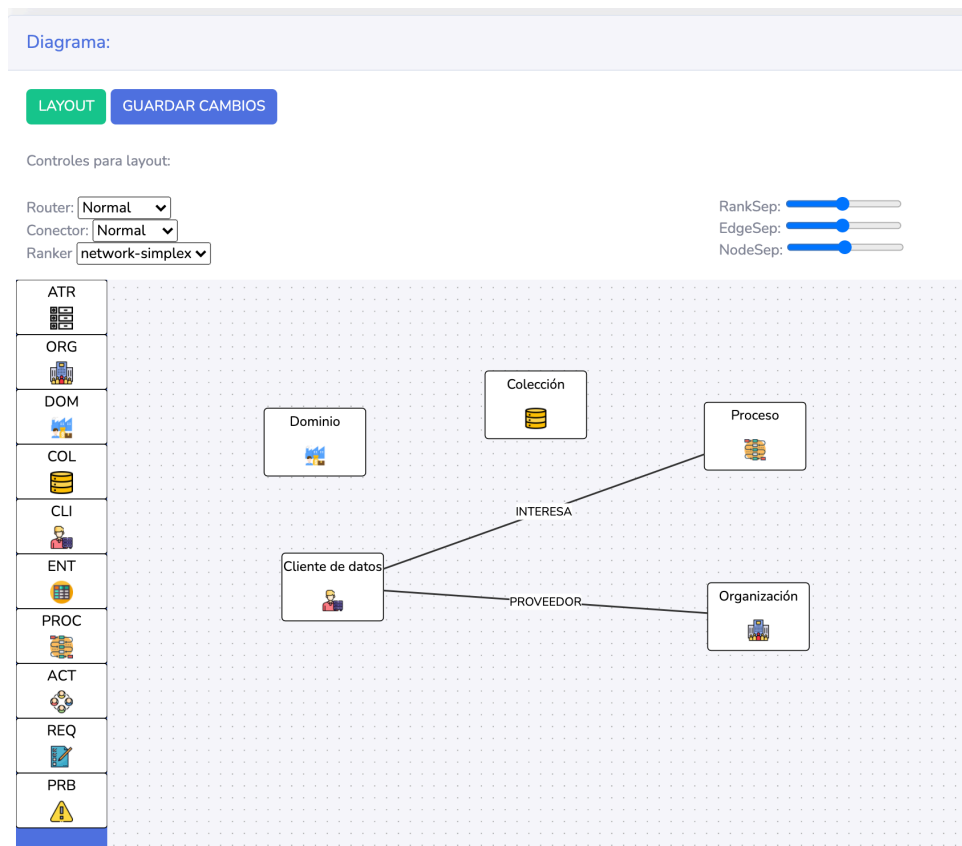


Figura 22: Ejemplo de la Representación Gráfica para un Escenario

Por fuera de estas secciones, existen dos botones: “Guardar Cambios” y “Layout”. Cada vez que el usuario realice algún cambio en el diagrama como por ejemplo la modificación del nombre de un elemento o una relación, deberá presionar el botón “Guardar Cambios”. De esta manera, se persiste la estructura del diagrama en la base de datos; de lo contrario la plataforma no guardará las últimas modificaciones. Por su parte, el botón “Layout” provee una función para “reubicar” todas las “cajas” y “links” en el diagrama de forma automática.

JointJs, la librería utilizada, trata al diagrama como un JSON, almacenando toda la información de sus elementos en un arreglo de objetos llamado “cells”. Cada celda de este arreglo corresponde a una de las “cajas” o “links” previamente mencionadas. En el Apéndice A se brindan detalles de las estructuras JSON correspondientes a los elementos del diagrama.

Además la librería permite responder a una serie de eventos, entre ellos, “arrastrar”, “soltar”, “doble click”, etc. Es a partir de ellos que se logra ofrecer al usuario un conjunto de diferentes acciones sobre los elementos y las relaciones en el diagrama.

Uno de los desafíos al momento de la implementación de esta funcionalidad fue el paralelismo que debe existir entre la lógica del diagrama y el registro/edición de alguno de los elementos del escenario mediante el formulario. Es decir, para cada acción posible que se pudiese ejecutar en la vista formulario, se debe modificar el JSON del diagrama y viceversa.

A continuación se presenta cómo las acciones en el formulario impactan en el JSON del diagrama:

- Registrar nuevo elemento: se debe agregar un nuevo objeto con sus valores correspondientes en el arreglo “cells”.
- Editar atributos de elemento: se debe recorrer el arreglo “cells” hasta encontrar el elemento deseado y actualizar el objeto con los nuevos valores correspondientes.
- Eliminar elemento: se debe recorrer el arreglo “cells” en busca del elemento en cuestión y removerlo. También, se deben remover todas las relaciones que existan para ese elemento.
- Registrar nueva relación: similar a registrar elemento.
- Editar valor de relación: similar a editar elemento.
- Eliminar relación: similar a eliminar elemento.

Recíprocamente, teniendo en cuenta las acciones posibles que se pueden aplicar sobre el diagrama, se detallan los impactos que estos tienen en la base de datos:

- Registrar elemento o relación: para cada elemento o relación creados en el diagrama, se debe agregar en la base de datos (en la tabla correspondiente) un nuevo registro con sus valores.

- Editar elemento o relación: para cada elemento o relación editada en el diagrama, se debe actualizar en la base de datos (en la tabla que corresponda) el registro con los valores nuevos.
- Eliminar elemento: para cada elemento eliminado del diagrama, se debe eliminar de la base de datos en la tabla correspondiente. En el caso de que ese elemento tenga uno o más links asociados, se debe iterar sobre el arreglo “cells” para localizar las relaciones y, posteriormente removerlas de la base.
- Eliminar relación: para cada relación eliminada del diagrama, se debe eliminar de la base de datos.

5.4. Gestión del Modelo de Calidad

Para presentar el modelo de calidad de datos de referencia a los usuarios, se tomó como base la estructura de “árbol” presentada en “Web del Modelo de Calidad de Datos de AGESIC”¹⁸ ya que resulta muy intuitiva. La Figura 23 presenta esto último, mientras que la Figura 24 muestra la estructura del modelo de referencia en la plataforma.

¹⁸Web del Modelo de Calidad de Datos de AGESIC: <https://calidad-datos.agesic.gub.uy/>.

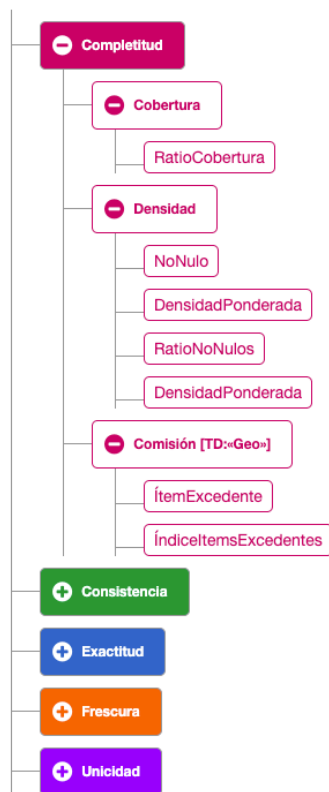


Figura 23: Calidad de Datos - AGESIC

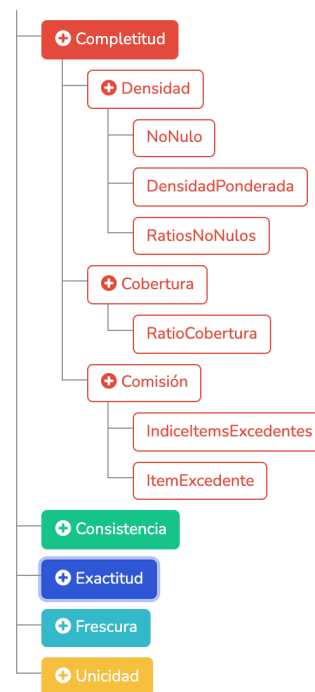


Figura 24: Ejemplo Modelo de Referencia - Plataforma

Cómo se mencionó anteriormente, el modelo de calidad de una organización está compuesto tanto por los elementos de modelo de la última versión del modelo de referencia como por los elementos extendidos. Para poder diferenciarlos y así ofrecer a los usuarios la posibilidad de llevar a cabo un conjunto de acciones (registrar similar, eliminar o editar) sobre los elementos extendidos de su organización, a nivel de la base de datos se añade una relación de un Elemento de Modelo con una Organización lo que resulta suficiente para identificar el Modelo de Referencia Extendido.

Por otro lado, a fin de que las organizaciones puedan compartir sus propios elementos de modelo o utilizar elementos de modelo definidos por otras, cuentan con la funcionalidad de “importar/exportar”. De esta forma, pueden exportar su modelo de calidad con sus elementos extendidos

y compartir el JSON generado con otra organización. Análogamente, pueden importar el JSON del modelo de calidad de otra organización. La estructura del JSON para el modelo de calidad puede verse en el Apéndice B.

Para importar el modelo de calidad de otra organización, se itera sobre cada uno de los elementos de modelo especificados en el JSON de entrada. Aprovechando su estructura jerárquica se recorre en orden: Dimensiones, Factores y Métricas. Para cada uno de ellos, se verifica:

- Si ya existe un elemento extendido para la organización con el mismo nombre que el entrante, se crea un nuevo histórico para ese elemento con los datos obtenidos del JSON.
- En caso contrario, se crea un nuevo elemento extendido para la organización junto a su primer histórico.

5.5. Gestión de Versiones del Modelo

Al momento de registrar una versión del modelo de referencia, se debe especificar:

- Un nombre: refiere al nombre propio de la versión. Si bien por lo general, las versiones son nombradas siguiendo el formato “X.Y.Z”, donde cada letra representa un dígito, se prefirió dejar este campo como texto libre para dar una mayor libertad.
- Notas: permite registrar información adicional sobre la versión.
- Fecha de creación (no editable): corresponde a la fecha en la que se crea la versión.
- Un conjunto de históricos de diferentes elementos de modelo.

Una vez creada la versión, el Super Administrador puede realizar cualquiera de las siguientes acciones: editar la información básica de la versión, registrar nuevos elementos para esa versión, realizar un conjunto de acciones (editar, registrar similar o eliminar) sobre los históricos

pertenecientes a la versión o dirigirse a la vista “modelo maestro”. La Figura 25 proporciona una idea general de esta vista en la plataforma propuesta.

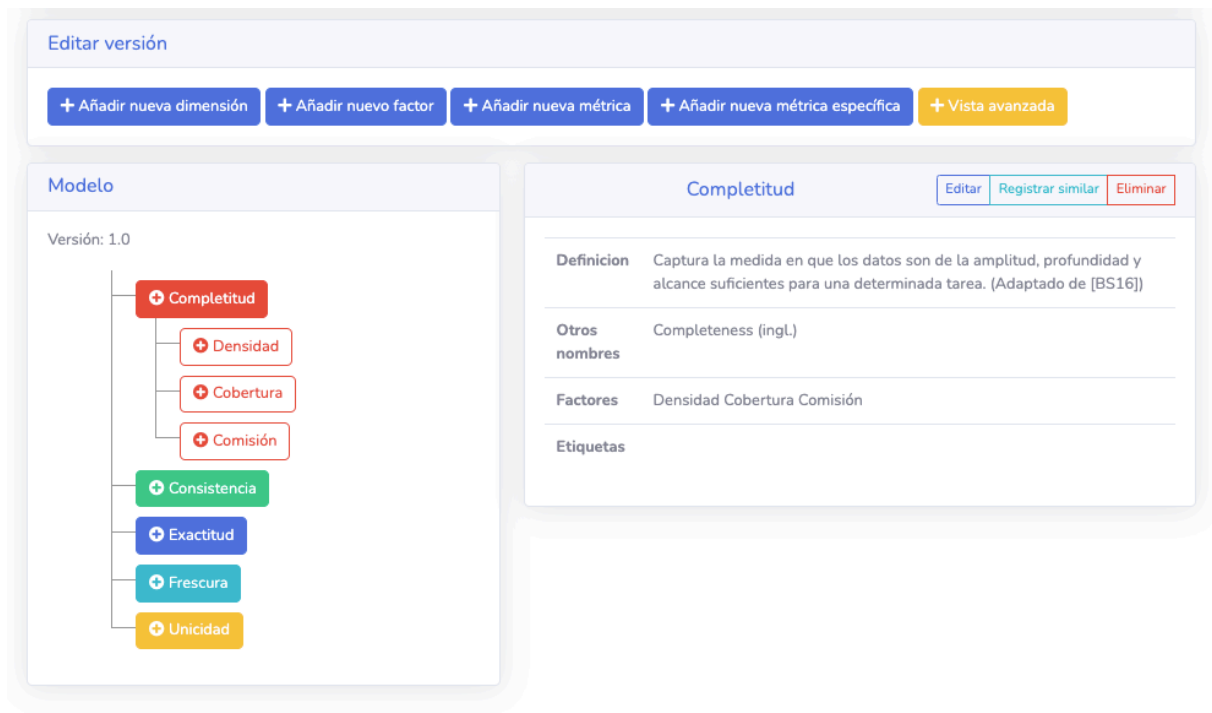


Figura 25: Acciones Posibles del Super Administrador sobre una Versión

La vista “modelo maestro” de una versión, presentada en la Figura 26, tiene **todos** los elementos que conforman el modelo de referencia. Es decir, allí se encuentran todos los elementos de modelo que se fueron registrando a lo largo de la historia. El objetivo de esta vista es poder visualizar al modelo de referencia y permitir al Super Administrador seleccionar históricos de elementos para que formen o no, parte de la versión.

The screenshot displays two main panels. The left panel, titled 'Elementos del Modelo', shows a tree structure of model elements under the current version '1.0'. The elements are: 'Completitud' (red), 'Cobertura' (red), 'RatioCobertura' (red), 'Comisión' (red), 'Densidad' (red), 'Comisión' (red), 'Consistencia' (green), 'Exactitud' (blue), 'Frescura' (teal), and 'Unicidad' (yellow). Each element has a plus sign icon and a checkmark indicating its status. The right panel, titled 'Historicos: Completitud', contains a table with the following data:

Fecha	Actual	Versiones
1/12/20 22:52	Remover de versión	null, 1.0

Figura 26: Vista Maestro

Dentro de esta vista, para cada elemento, se muestra un “tick” que indica si el mismo forma parte de la versión que se está editando o no. A su vez, el usuario puede fácilmente seleccionar qué histórico desea incorporar a la versión que está editando, ya que cuenta con la opción para ello: un botón “añadir a la versión” o “remover de la versión”.

Uno de los controles que se tiene que realizar en este caso, es que no se debe permitir que se registren a la versión, históricos de elementos sin sus correspondientes “padres”. Por ejemplo, si el Super Administrador solamente seleccionase una métrica para registrar en la versión, por defecto, el último histórico del factor “padre” y la dimensión de ese factor, serán marcados automáticamente para añadirse en la versión. Por el contrario, si se desmarca de la versión a una dimensión, también lo harán automáticamente los históricos de los elementos “hijos” que formen parte de la versión.

La funcionalidad de “importar/exportar”, también se encuentra disponible para el Super Administrador. En este caso, al momento de importar un JSON se crea una nueva versión. Ésta, contiene los históricos de los elementos de modelo que se especifican en el archivo. Para cada

uno de los elementos de modelo especificados en el JSON:

- Si ya existe un elemento de referencia con el mismo nombre que el entrante, se crea un nuevo histórico para ese elemento y el mismo es añadido a la versión creada.
- En caso contrario, se crea un nuevo elemento de referencia junto a su histórico y se añade a la versión.

6. Caso de Estudio y Validación

En este capítulo se presenta una validación preliminar de la plataforma propuesta en base al prototipo desarrollado, la cual consistió en la instanciación de un caso de estudio utilizando la plataforma y la interacción con actores interesados.

6.1. Descripción del Caso de Estudio

En esta sección se describe el caso de estudio en el que se aplica el framework, que constituye uno de los componentes del framework [2]. El caso de uso se enfoca en un sistema denominado “Sistema de Reclamos Ciudadanos” (SRC). La Figura 27 presenta una visión conceptual del sistema SRC y los distintos tipos de usuarios involucrados:

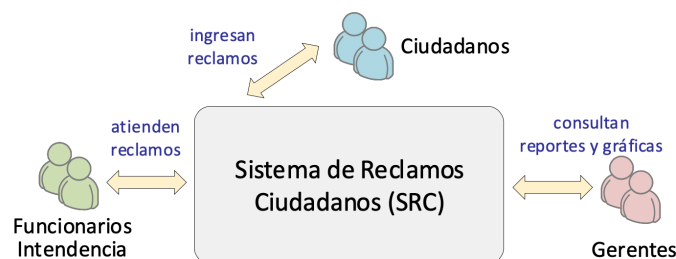


Figura 27: Visión Conceptual - SRC [2]

Los **Ciudadanos**, usuarios finales del sistema, realizan reclamos a través de una aplicación en su dispositivo móvil. Los **Funcionarios** procesan y atienden los reclamos de los ciudadanos. Los reclamos se asignan a diferentes áreas de la intendencia de acuerdo a la categoría del reclamo (por ejemplo: contenedor lleno). Los **Gerentes** de diferentes organismos tienen acceso a funcionalidades de reportes y gráficas que les brindan soporte para la toma de decisiones. La Figura 28, presenta los principales componentes del sistema SRC vinculados a la gestión de reclamos en una intendencia específica: la Intendencia de Montevideo (IM).

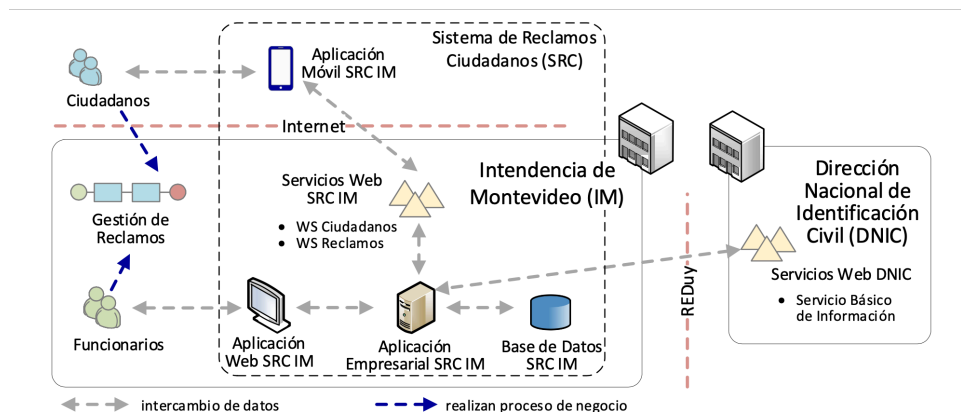


Figura 28: Componentes del Sistema SRC para la Gestión de Reclamos [2]

Al momento de registrar un reclamo, el ciudadano tiene que asignar una categoría, una ubicación geográfica, una descripción y opcionalmente imágenes. Estos reclamos son asignados por los funcionarios a un área específica de la intendencia para su resolución. Eventualmente, los reclamos pueden ser rechazados si cuentan con información insuficiente.

La lógica de negocio del sistema SRC está implementada en una aplicación empresarial, a la que se puede acceder mediante una aplicación web utilizada por los funcionarios o una aplicación móvil utilizada por los ciudadanos. En particular, la aplicación móvil accede a la funcionalidad de la aplicación empresarial a través de servicios Web (WS Ciudadanos, WS Reclamos).

La aplicación utiliza una base de datos relacional para almacenar los datos que gestiona y consume un servicio Web (i.e. Servicio Básico de Información) de la Dirección Nacional de Identificación Civil (DNIC) para validar el número de cédula de identidad ingresado en el registro de un ciudadano.

6.2. Aplicación del Framework

Esta sección presenta la aplicación del framework en el caso de estudio, utilizando el prototipo de la plataforma propuesta.

6.2.1. Configuración

Para poder utilizar la plataforma, es necesario realizar una serie de acciones de configuración inicial.

1. **Registrar al Super Administrador.** Consiste en insertar un registro en la tabla “*Users*”. Se debe especificar el email, la contraseña (encriptada mediante el algoritmo MD5) y no se le debe asignar una organización.
2. **Publicar versión inicial del modelo.** Una vez iniciada la sesión con este usuario, se debe registrar y publicar, la primera versión del modelo de referencia.
3. **Configuración de los datos referenciales iniciales.** Es importante configurar los datos referenciales iniciales de la plataforma como tipos de colecciones, tipos de atributos, etc.
4. **Alta de organización y Administrador.** Por último, se debe dar de alta a una organización junto a su administrador. Así es como se le concede acceso a la organización para comenzar a utilizar la plataforma.

En las próximas secciones, se describe la aplicación del framework con la plataforma en cada una de las etapas planteadas.

6.2.2. Caracterización Escenario

La primera etapa del proceso de gestión de calidad del framework, plantea identificar los elementos relevantes (organizaciones, dominios de aplicaciones, colecciones de datos, etc) para el escenario de trabajo, así como también las relaciones entre ellos. A su vez, se deben ingresar los requerimientos y problemas de calidad de datos relevantes para el escenario y su relación con el resto de los elementos del escenario.

La Figura 29 muestra la forma en la que se pueden registrar algunos de estos elementos en la plataforma.

Figura 29: Registrar un Elemento de Negocio

Por otro lado, en la Figura 30 se puede ver cómo se registra una relación entre un cliente de datos y una organización. Las tablas de relación entre elementos son generadas automáticamente por la plataforma de acuerdo a los elementos registrados.

	Intendencia de Montevideo (IM)	Dirección Nacional de Identificación Civil (DNIC)
WS Ciudadanos IM	responsable	
WS Reclamos IM	responsable, utiliza	
Servicio Básico de Información	utiliza	responsable
Aplicación Móvil SRC IM	responsable	
Aplicación Web SRC IM	responsable, utiliz:	
Aplicación Empresarial SRC IM	responsable, utiliz:	
Ciudadano		
Funcionario IM	responsable	

Figura 30: Registrar una Relación - Cliente de Datos y Organización

En esta etapa se debe especificar sobre cuáles de los elementos del escenario aplican los requerimientos identificados. La Figura 31 presenta a nivel del diagrama, las relaciones para un conjunto de requerimientos (R1 y R2) y actores de datos:

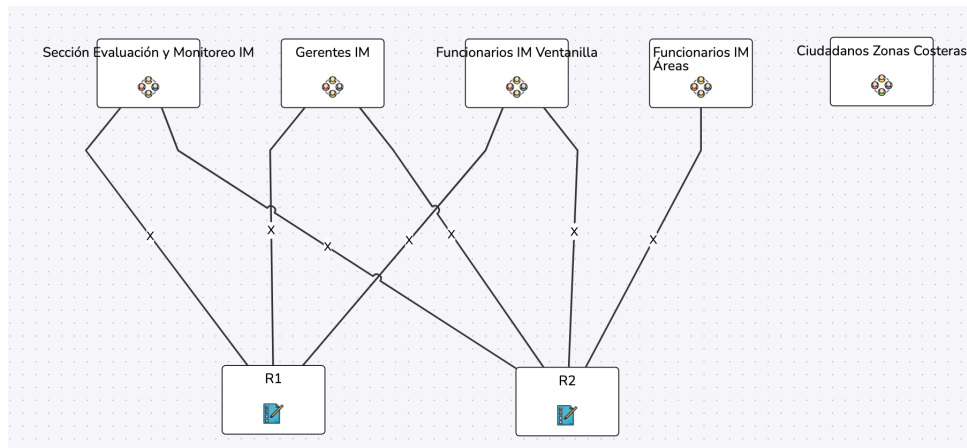


Figura 31: Diagrama - Requerimientos y Actores

De manera similar, se registran problemas de calidad de datos (nombres irreales, correos electrónicos inexistentes, etc) y se determinan qué actores los identifican. A partir de estos problemas, pueden surgir nuevos requerimientos. Para registrar y relacionar estos nuevos elementos, la plataforma ofrece una interfaz similar a las anteriores.

6.2.3. Examinar Datos Objetivo

La segunda etapa consiste en examinar las características de los datos y realizar una primera estimación de su calidad, así como detectar problemas de calidad de datos y definir nuevos requerimientos a partir de estos.

Se comienza por clasificar los tipos de colecciones y de atributos para los elementos que se registraron en la etapa anterior. En la Figura 32, se puede ver cómo la plataforma permite clasificar los tipos de las colecciones de datos. Similar es el caso para clasificar el tipo de los atributos.

Colección de Datos	Tipo de Colección
BD SRC IM	BD RELACIONAL

Figura 32: Tipos de las Colecciones de Datos

Se identifican las técnicas que se van a aplicar, se establecen los resultados obtenidos y se estima el alcance de los problemas. A modo de ejemplo, se realiza una exploración directa de los datos de la colección “Base de Datos SRC IM”. Luego de aplicar la técnica, se registran los resultados obtenidos para cada configuración de la misma, tal como se muestra en la Figura 33. Para cada resultado, se registra un nuevo problema de calidad.

Configuración	Parámetros	Alias	Resultado obtenido	Archivo	Acciones
BD SRC IM: Ciudadano (nombres,apellido)		Nombres extraños	elcarbone, vidatrico	<input type="button" value="Seleccionar archivo"/> No se eligió archivo	<input type="button" value="Editar"/>
BD SRC IM: Ciudadano (emai)		Errores en dominios	@hotmaL.com, @gmaL.com	<input type="button" value="Seleccionar archivo"/> No se eligió archivo	<input type="button" value="Editar"/>
BD SRC IM: Ciudadano (fechaNa)		Fechas extrañas	1903, 1906, 2019	<input type="button" value="Seleccionar archivo"/> No se eligió archivo	<input type="button" value="Editar"/>

Figura 33: Determinar Resultados - T1

Por otro lado, la Figura 34 muestra cómo se puede estimar el alcance para cada uno de los problemas identificados.

Resultados - T1					
Configuracion	Alias resultado	P1	P2	P3	P4
T1 - BD SRC IM, Ciudadano (emai)	Errores en dominios	<input type="text"/>	ALTO <input type="text"/>	<input type="text"/>	<input type="text"/>
T1 - BD SRC IM, Ciudadano (nombres,apellido)	Nombres extraños	ALTO <input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
T1 - BD SRC IM, Ciudadano (fechaNa)	Fechas extrañas	<input type="text"/>	<input type="text"/>	<input type="text"/>	ALTO <input type="text"/>

Figura 34: Aplicación T1 para Problemas de Calidad de Datos

6.2.4. Definir Estrategia de Calidad

La tercer etapa consiste en definir la estrategia a llevar a cabo para la gestión de la calidad de datos en el escenario. La estrategia depende de los resultados obtenidos en etapas anteriores y a prioridades que se establecen en esta etapa.

Se comienza asignando prioridades a todos los requerimientos identificados. En particular la Figura 35 muestra un ejemplo de cómo la plataforma permite asignar prioridad a los diferentes requerimientos.

Prioridades:		
Requerimiento	Prioridad	Comentarios
R1	ALTA <input type="text"/>	De gran interés para un gran número de actores y aplica a varios elementos del escenario que son también de interés para varios actores
R2	MEDIA <input type="text"/>	...
R3	BAJA <input type="text"/>	...
R4	ALTA <input type="text"/>	...
R5	ALTA <input type="text"/>	...
R6	ALTA <input type="text"/>	...

Figura 35: Asignación de Prioridades a Requerimientos

A continuación, se define una estrategia que consta de dos pasos: abordar los requerimientos con prioridad alta y muy alta, y luego abordar los restantes. La Figura 36 muestra cómo la plataforma permite asignar a cada paso, los requerimientos a abordar.

Asignar requerimientos a los pasos

	Paso 1	Paso 2
R1 - ALTA	X	
R2 - MEDIA		X
R3 - BAJA		X
R4 - ALTA	X	
R5 - ALTA	X	
R6 - ALTA	X	
R7 - BAJA		X
R8 - BAJA		X
R9 - ALTA	X	
R10 - ALTA	X	
R11 - MEDIA		X
R12 - BAJA		X
R13		
R14		
R15		

Figura 36: Pasos de la Estrategia

6.2.5. Definir Modelo de Calidad

La cuarta etapa consiste en la definición de un modelo de calidad de datos para el escenario.

El caso de estudio comienza seleccionando qué elementos del modelo de calidad se van a utilizar. En particular, se comienza identificando métricas para los requerimientos de prioridad alta. Por ejemplo, como se puede observar en la Figura 37, para el primer requerimiento se identifica la métrica Formato(numDoc, DNIC), la cual implica medir la correctitud sintáctica en documentos de identidad uruguayos.

Precargar elementos del modelo Añadir nuevo elemento

Requerimiento	Dimensión	Factor	Métrica	Acciones
R1	Exactitud	Formato	Formato(NumeroDocumento, DNIC)	Eliminar
R4	Exactitud	Correctitud Semántica	CorrectitudSemDebil	Eliminar
R5	Exactitud	Correctitud Semán	CorrectitudSemDebil	Añadir Eliminar

Figura 37: Requerimientos de Calidad y Elementos del Modelo de Referencia

Luego, se deben instanciar las métricas para cada requerimiento, detallando sobre qué entidad/s/atributos actúa y sus reglas de correspondencia.

Finalmente se definen perfiles de evaluación en base a reglas de evaluación para las métricas instanciadas. En particular, la Figura 38 presenta un perfil de evaluación donde para cada una de las métricas con resultado de tipo “boolean”, se define una regla de evaluación que tiene como condición que el resultado sea “true”.

Perfil Básico Eliminar

Nombre: Perfil Básico Editar

Descripción: Métricas con resultado de tipo boolean Editar

REGLAS Añadir nueva regla

Regla	Métrica	Condición	Valor	Acciones
RegDoc: Cumple Formato Documento	Métrica M1	IGUAL	true	Eliminar
RegEmail: Existe Email	Métrica M4	IGUAL	true	Eliminar
RegDomic: Existe Dirección Geo	Métrica M5	IGUAL	true	Añadir Eliminar

Figura 38: Perfil de Evaluación - Perfil Básico

6.2.6. Medir y Evaluar la Calidad de Datos

La quinta etapa consiste en realizar mediciones utilizando las métricas y métodos definidos en el modelo de calidad, así como la evaluación de la calidad de los datos de acuerdo a sus perfiles.

En el caso de estudio, se implementan o adaptan métodos de medición para tomar medidas de calidad de datos en el escenario de trabajo. Asumiendo que ya se cuenta con una implementación base de los métodos requeridos, luego de ejecutados, se almacenan las medidas para su posterior evaluación. En la Figura 39 se presenta cómo la plataforma permite ingresar los resultados de medición para la métrica instanciada anteriormente.

Métrica M1 - Formato(NumeroDocumento, DNIC)

Nombre Genérica: Formato(NumeroDocumento, DNIC)

Descripción Genérica: Indica si el valor del atributo numDoc cumple con el formato de cédula de identidad uruguaya establecido por DNIC, que establece que la cédula tiene siete dígitos seguidos de un octavo dígito verificador que está en función de los otros siete (d1 d2 d3 d4 d5 d6 d7 - w, w=f(di)).

Nombre Instanciada: Métrica M1

RESULTADOS Añadir nuevo resultado

ID Medida	ID Medición	Configuración	Fecha	Resultado	ID Objeto	Acciones
1	12	BD SRC IM, Ciudadano (numDoc)	6/12/19	true	25	Eliminar
2	12	BD SRC IM, Ciudadano (numDoc)	6/12/19	true	27	Eliminar
4	12	BD SRC IM, Ciudadano (numDoc)	6/12/19	true	30	Eliminar
3	12	BD SRC IM, Ciudadano (numDoc)	6/12/19	false	28	Eliminar

Figura 39: Medidas Métrica M1: Formato(numDoc, DNIC)

En base a las medidas tomadas y a las reglas y perfiles de evaluación, la plataforma brinda un resumen de la evaluación de la calidad de datos, como se puede ver en la Figuras 40, 41 y 42.

Evaluación Medidas					
Regla	Métrica	idMedida	idMedición	Fecha	Resultado
Métrica M1 (IGUAL true)	Métrica M1	1 (true)	12	6/12/19	true
Métrica M1 (IGUAL true)	Métrica M1	2 (true)	12	6/12/19	true
Métrica M1 (IGUAL true)	Métrica M1	4 (true)	12	6/12/19	true
Métrica M1 (IGUAL true)	Métrica M1	3 (false)	12	6/12/19	false
Métrica M4 (IGUAL true)	Métrica M4	10 (false)	12	6/12/19	false
Métrica M4 (IGUAL true)	Métrica M4	11 (true)	12	6/12/19	true
Métrica M4 (IGUAL true)	Métrica M4	12 (true)	12	6/12/19	true
Métrica M4 (IGUAL true)	Métrica M4	13 (false)	12	6/12/19	false
Métrica M5 (IGUAL true)	Métrica M5	21 (false)	12	6/12/19	false
Métrica M5 (IGUAL true)	Métrica M5	22 (false)	12	6/12/19	false
Métrica M5 (IGUAL true)	Métrica M5	23 (true)	12	6/12/19	true
Métrica M5 (IGUAL true)	Métrica M5	24 (true)	12	6/12/19	true

Figura 40: Evaluación Medidas

Evaluación Reglas		
idMedición	Regla	Resultado
12 (true, true, true, false)	RegDoc: Cumple Formato Documento (IGUAL true)	0,75
12 (false, true, true, false)	RegEmail: Existe Email (IGUAL true)	0,5
12 (false, false, true, true)	RegDomic: Existe Dirección Geo (IGUAL true)	0,5

Figura 41: Evaluación Reglas

Evaluación Perfiles		
idMedición	Perfil	Resultado
12	RegDoc: Cumple Formato Documento	0,583

Figura 42: Evaluación Perfiles

6.2.7. Determinar Causas de Problemas

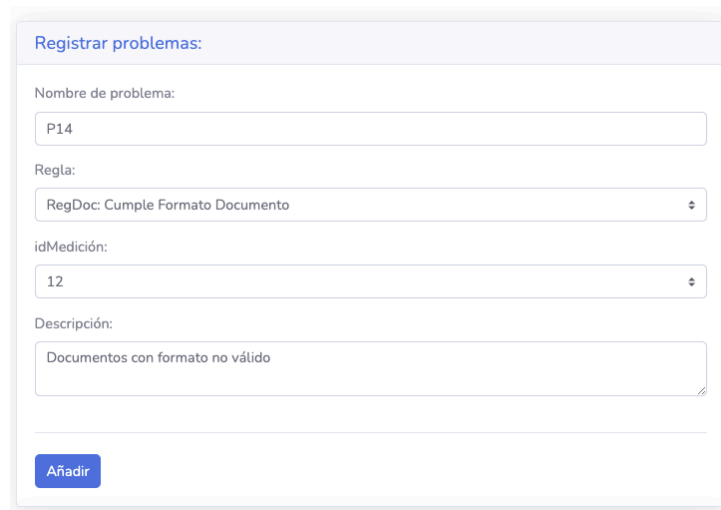
La sexta etapa consiste en determinar las causas de los problemas de calidad de datos que se confirmen o detecten en base a la evaluación de la calidad de datos.

Para confirmar los problemas ya identificados, se analizan las evaluaciones de las reglas vinculadas. En particular, la evaluación de la regla RegDoc es 0,75, lo que permite detectar un nuevo problema. Las Figuras 43 y 44, muestran cómo se confirman y detectan problemas respectivamente en la plataforma.

Problemas:

Problema	Descripción	Regla	idMedición
P1	Nombres irreales. Se han detectado nombres de fantasía, apodos o nicknames en los campos destinados a los nombres y apellidos del ciudadano.	Elija uno	
P2	Correos electrónicos inexistentes. A algunos ciudadanos registrados no les llegan los correos electrónicos que envía el sistema porque ingresaron incorrectamente su dirección de correo.	RegEmail: Existe Email	12
P3	Domicilios no encontrados. Algunos de los domicilios ingresados por los ciudadanos no se corresponden con ninguna dirección oficial de la capa de direcciones que maneja el sistema.	RegDomic: Existe Dirección C	12

Figura 43: Confirmación de Problemas



Registrar problemas:

Nombre de problema:
P14

Regla:
RegDoc: Cumple Formato Documento

idMedición:
12

Descripción:
Documentos con formato no válido

Añadir

Figura 44: Detectar Nuevo Problema

Finalmente, se deben identificar las causas de los problemas de calidad del escenario.

6.3. Validación AGESIC

Luego de haber logrado un prototipo estable, se llevó a cabo una reunión con referentes de AGESIC en el área de Calidad de Datos, quienes estaban familiarizados con el framework. El objetivo del encuentro fue mostrar el funcionamiento completo de la plataforma, mencionando sus principales funcionalidades para así obtener una devolución de su parte.

Como aspecto positivo, destacaron principalmente a la plataforma en sí, ya que hasta el momento los procesos relacionados a la gestión de calidad de datos en base al framework no contaban con una herramienta de soporte especializada. Es así, que encontraron muy útil la existencia de una plataforma que integre los distintos conceptos desarrollados en el framework.

Además, les resultó útil poder visualizar e interactuar con los diferentes elementos de los escenarios de una forma más gráfica.

Como aspectos a mejorar, señalaron los siguientes:

- En la etapa Examinar Datos Objetivo, los resultados registrados luego de aplicar la técnica, deberían quedar asociados a una fecha. Esto permitiría registrar varios resultados en diferentes momentos para una misma técnica.
- Incluir en la plataforma técnicas de *Data Profiling* y automatizar aún más el proceso respecto de la aplicación de estas técnicas.
- A nivel del diagrama, se debería especificar a modo de guía para el usuario, cuáles son las relaciones posibles a registrar entre los distintos tipos de elementos técnicos.
- Al momento de definir las métricas a aplicar en la gestión de calidad de datos de un escenario, el modelo de calidad que se presenta es el común a la organización. Sin embargo, sería preferible que el modelo esté asociado a un escenario, pues como muchas veces sucede en la práctica, hay elementos de modelo que no son relevantes para un escenario en particular.

A modo de conclusión, se obtuvo una devolución positiva sobre la plataforma en general, encontrándola útil para agilizar los procesos de gestión de calidad de datos, de una manera más automatizada.

Por otro lado, sería bueno en un futuro contar con un prototipo avanzado que integre las mejoras mencionadas.

6.4. Resumen

A modo de resumen, se pudo aplicar en la plataforma el caso de estudio en su totalidad, comprobando que da soporte a los diferentes requerimientos planteados, como por ejemplo las diferentes etapas del framework, la representación gráfica, etc.

Además, se llevó a cabo una reunión con los referentes de AGESIC que permitió validar de forma preliminar la plataforma. Si bien se entiende que la plataforma es una propuesta, se adecúa funcionalmente a los requerimientos y tiene potencial para ser utilizada como base en un

prototipo más avanzado.

En base a la aplicación del framework en el caso de estudio utilizando el prototipo, se encontró flexibilidad en la plataforma ya que permite al usuario navegar sobre las diferentes etapas y poder visualizar la información de distintas formas. También, se consideran beneficiosas las diferentes funcionalidades que se le otorgan a una organización respecto a la gestión de los modelos de referencia.

Por otro lado, sería oportuno contar con ciertos componentes visuales que sirvan de guía a los usuarios para comprender qué hacer en cada una de las etapas del proceso.

7. Conclusiones y Trabajo a Futuro

En este capítulo se plantean las conclusiones del proyecto y se presentan los lineamientos generales sobre el trabajo a futuro.

7.1. Conclusiones

El objetivo general del proyecto fue desarrollar una plataforma que brinde soporte al proceso para la gestión de calidad de datos aplicando el framework definido en [2].

Para ello, se realizó un análisis detallado de la realidad planteada, que fue refinado a través de las sucesivas reuniones con los tutores.

En las mismas, se planteó la incorporación de diversos requerimientos, como por ejemplo dar soporte a las distintas etapas del framework, gestión del modelo de referencia, etc. Además se propusieron otros que no se encontraban definidos en el mismo, como por ejemplo, el manejo de versiones, control de acceso, entre otros.

Luego, se implementó un prototipo de la plataforma que permitió confirmar su factibilidad técnica.

Por último, la adecuación funcional de la plataforma fue validada con el desarrollo de un caso de estudio y una reunión con referentes de AGESIC, quienes destacaron su utilidad respecto a la aplicación del framework.

Si bien se encontraron varios aspectos a mejorar en lo que refiere a la plataforma, se considera que la propuesta de plataforma y el prototipo desarrollado sirven de base para futuros trabajos que puedan surgir, respecto a la aplicación del framework en el contexto de las organizaciones vinculadas a gobierno digital.

En este mismo contexto, resulta fundamental contar con herramientas de índole tecnológico que

permitan realizar una adecuada gestión de la calidad de los datos. De esta forma, se apunta a que las organizaciones puedan contar con información de mayor calidad para la toma de decisiones estratégicas, que permitan brindar de una forma más eficiente, mejores servicios y productos a los ciudadanos de Uruguay.

7.2. Trabajo a Futuro

Durante el desarrollo del proyecto surgieron diferentes propuestas que debido al alcance del mismo quedaron por fuera y podrían considerarse como mejoras a realizar:

- **Plataforma multi-lenguaje.** Hasta el momento, los mensajes, campos y otros elementos del sitio, se encuentran en español y son mostrados al usuario en este lenguaje. Wicket, por su lado, permite la utilización de varios lenguajes a través de su etiqueta “`<wicket:message key=“example”><wicket:message>`”. De esta manera, se puede definir un archivo “*.properties*” para cada lenguaje, especificando la traducción deseada para cada “*key*” definido en la plataforma.
- **Sugerir elementos de modelo.** Una función relacionada al modelo de referencia, es que las diferentes organizaciones que utilizan la plataforma puedan sugerir al Super Administrador la incorporación de nuevas dimensiones, factores o métricas al modelo de referencia. De esta manera, el Super Administrador recibiría peticiones por parte de los usuarios y podría tomarlas en cuenta para una próxima versión del modelo de referencia.
- **Acceso a las versiones.** La plataforma podría mostrar a las organizaciones información sobre las nuevas versiones que publica el Super Administrador de la plataforma respecto al modelo de referencia. De esta forma, podrían conocer qué cambios se están llevando a cabo en las nuevas versiones (qué elementos se modificaron, qué nombres cambiaron, etc).
- **Agregado de herramientas, métodos y automatización del Data Profiling.** La plataforma podría ser extendida para incluir herramientas y sugerencias asociadas a cada técnica/-

métrica genéricas. El Super Administrador podría establecer “sub-rutinas” para realizar las mediciones y que la plataforma ejecute los métodos, es decir, que haga la labor de *Data Profiling* sobre atributos y entidades. De ser incluida esta funcionalidad, sin dudas sería relevante ya que permitiría la gestión completa del proceso en detalle.

- **Algoritmos de detección de Causas de problemas.** En esta subetapa la plataforma podría identificar cuáles organizaciones, actores, o clientes de datos están relacionados con el requerimiento que generó la medición que se configuró, y con cuáles colecciones asociadas a qué procesos, y de esta manera sugerir al usuario un estimado de dónde se podría encontrar la causa del problema detectado.
- **Definir, Ejecutar y Evaluar Plan de Mejora.** La plataforma no cuenta con la última etapa del proceso implementada. A grandes rasgos, para agregar esta nueva etapa, se podría dividir la lógica en tres subetapas separadas; por un lado una subetapa “Asignar Prioridades a las Causas Problemas” que tiene como fin la asignación de prioridades a los diferentes problemas detectados. Luego la subetapa “Definir Acciones de Plan de Mejora” donde el usuario definiría las acciones a tomar así como las estrategias a utilizar para cada paso. Finalmente, en la última subetapa “Ejecutar y Evaluar Plan de Mejora”, el usuario luego de realizar una nueva medición de calidad de datos en el escenario a partir de su plan de mejora, podría comparar el nivel de cumplimiento de los perfiles de evaluación considerados.
- **Refactorizar.** La refactorización de código, si bien no agregaría ninguna función adicional a la plataforma, sería de gran utilidad. Existen varios lugares en el código donde se podrían refactorizar funciones o clases, con el fin de simplificar el código y así lograr una mejor mantenibilidad.
- **Responsive.** Si bien la plataforma no está pensada para ser utilizada en “*smartphones*” o dispositivos pequeños, sería oportuno lograr que la plataforma se adapte de la mejor manera posible a los dispositivos medianos. Aunque algunas de las librerías utilizadas para desarrollar esta plataforma permiten que ciertos de sus componentes presenten un

comportamiento “responsive”, no fue una prioridad que se tuvo durante el proceso de implementación.

Referencias

- [1] W. Wayne, “Data quality and the bottom line: Achieving business success through a commitment to high quality data,” *TDWI report*, 2004.
- [2] F. de Ingeniería UDELAR and AGESIC, “Framework para la gestión de la calidad de datos en gobierno digital 1.0,” 2019.
- [3] M. Scannapieco and T. Catarci, “Data quality under a computer science perspective,” *Archivi & Computer*, vol. 2, pp. 1–15, 2002.
- [4] J. E. Olson, *Data quality: the accuracy dimension*. Elsevier, 2003.
- [5] L. Etcheverry, V. Peralta, and M. Bouzeghoub, “Qbox-foundation: a metadata platform for quality measurement,” in *proceeding of the 4th Workshop on Data and Knowledge Quality (QDC’2008)*, 2008.
- [6] F. Clerici and B. Fernández, “Herramienta para la evaluación de calidad de datos,” *Universidad de la República, Uruguay*, 2019.
- [7] B. Bianchi, “Un caso de estudio en calidad de datos para ingeniería de software empírica,” *Universidad de la República, Uruguay*, 2009.
- [8] C. Batini, M. Scannapieco *et al.*, “Data and information quality,” *Cham, Switzerland: Springer International Publishing. Google Scholar*, p. 43, 2016.
- [9] M. Brackett and P. S. Earley, “The dama guide to the data management body of knowledge (dama-dmbok guide),” 2009.
- [10] K. Wende, “A model for data governance—organising accountabilities for data quality management,” 2007.
- [11] I. Vaynberg, *Apache Wicket Cookbook*. Packt Publishing Ltd, 2011.
- [12] M. Shah, “Responsive web development using the twitter bootstrap framework,” 2015.

Apéndice A. Estructuras JSON del Diagrama

A continuación, se muestra un ejemplo del JSON correspondiente para un elemento del diagrama:

```
1 {
2   "type": "basic.DecoratedRect",
3   "size": {
4     "width": 120,
5     "height": 80
6   },
7   "position": {
8     "x": 100,
9     "y": 100
10  },
11  "angle": 0,
12  "id": "82f6bcbb-0306-43fa-82c6-882228c96083",
13  "elementInfo": {
14    "id": 95,
15    "elementType": "TIPO DE ELEMENTO"
16  },
17  "content": {
18    "name": "Ejemplo"
19  },
20  "attrs": {
21    "rect": {
22      "magnet": true
23    },
24    "text": {
25      "text": "Ejemplo"
26    },
27    "image": {
28      "xlink:href": "img/diagram/example.png"
29    }
30  }
}
```

31 }

Entre los campos más importantes:

- **Type**: indica a JointJs que el elemento deberá ser representado mediante una “caja”.
- **Size**: dimensiones de la “caja” dentro del diagrama.
- **Position**: coordenadas del elemento en el diagrama.
- **ElementInfo**: información del elemento que será utilizada en la lógica Java. El campo “*id*”, hace referencia al identificador del elemento en la base de datos y el campo “*elementType*” refiere al tipo del elemento dentro del escenario. Es decir, indica si el elemento es un Cliente de Datos, Requerimiento, etc.
- **Content**: refiere a los campos que un usuario puede editar al presionar dos veces sobre un elemento en el diagrama. En este caso, “*name*” hace referencia al nombre del elemento en sí.

Por otro lado, el JSON correspondiente a un link:

```
1 {
2   "type": "standard.Link",
3   "source": {
4     "id": "9e09a8bf-d1f1-437e-b21e-6d0d95525748"
5   },
6   "target": {
7     "id": "4c601c13-3285-46cf-8dea-b97f40a1d4d5"
8   },
9   "relationInfo": {
10    "relationType": "TIPO_DE_RELACION",
11    "internalSourceId": "9e09a8bf-d1f1-437e-b21e-6d0d95525748",
12    "id": 258,
13    "source": 258,
14    "internalTargetId": "4c601c13-3285-46cf-8dea-b97f40a1d4d5",
```

```
15     "target": 39,  
16     "relation": "RELACION"  
17 },  
18 "labels": [  
19   {  
20     "attrs": {  
21       "text": {  
22         "text": "RELACION"  
23       }  
24     }  
25   }  
26 ],  
27 "id": "b358f345-1929-479e-983c-71885798cc4d",  
28 "attrs": {  
29   "line": {  
30     "targetMarker": {  
31       "d": ""  
32     },  
33     "sourceMarker": {  
34       "d": "M 10 -5 0 0 10 5 20 0 z",  
35       "type": "path"  
36     }  
37   }  
38 }  
39 }
```

Se destacan:

- Source: contiene el identificador (propio de JointJs) correspondiente al elemento “fuente” de la relación.
- Target: similar a lo anterior, para el elemento destino de la relación.
- RelationInfo: información de la relación que será utilizada en la lógica Java. La clave “id”, hace referencia al identificador de la relación dentro de la base de datos; “relationType”

indica qué elementos involucra la relación, *“relation”* refiere al valor de la relación y *“source”* o *“target”*, son los identificadores en la base de datos de los elementos “fuente” y “destino” respectivamente.

Como se puede ver en los ejemplos, JointJs asigna a todos los objetos un identificador formado por números y letras. Este, es generado de forma automática y es utilizado internamente por la librería para “crear” el gráfico.

También, dentro de *“elementInfo”* (en el caso de los elementos) y de *“relationInfo”* (en el caso de los links), es necesario contar con un identificador para el elemento o link en la base de datos. De esta forma se podrá persistir el estado de los elementos cuando el usuario interactúe con alguno de ellos.

Apéndice B. Estructuras JSON del Modelo de Calidad

El siguiente ejemplo presenta el JSON para un modelo de calidad que tiene una dimensión (Complejitud), un factor (Cobertura) y una métrica (RatioCobertura).

```
1 {
2   "dimensions": [
3     {
4       "name": "Complejitud",
5       "alternative_names": "Completeness (ingl.)",
6       "definition": "11111 Captura la medida en que los datos son de la
7         amplitud, profundidad y alcance suficientes para una determinada tarea.
8         (Adaptado de [BS16])",
9       "factors": [
10        {
11          "name": "Cobertura",
12          "definition": "Captura la proporción entre la cantidad de
13            entidades existentes en una determinada colección de datos, y el total
14            de entidades que deberían existir en dicha colección. La cobertura varía
15            si se utiliza la Asunción de Mundo Cerrado, según la cual una colección
16            de datos debería contener todas las entidades de un tipo, o si se
17            utiliza la Asunción de Mundo Abierto, según la cual una colección de
18            datos puede ser una representación parcial de las entidades del mundo
19            real.)",
19          "alternative_names": "Coverage (ingl.)",
20          "metrics": [
21            {
22              "name": "RatioCobertura",
23              "semantics": "Proporción entre la cantidad de instancias de
24                una entidad y el número total de instancias de un referencial de esa
25                entidad.",
26              "granularity": "entidad.",
27              "result_type": "Intervalo real [0.0, 1.0].",
28              "suggested_name": "RatioCobertura(Entidad, Referencial)",
29              "properties": [
```

```
20         "Entidad=Ent",
21         "Referencial(Entidad)"
22     ]
23 }
24 ],
25 }
26 ]
27 },
28 ]
29 }
```

Apéndice C. Mockups iniciales

A continuación se presentan los mockups iniciales que se desarrollaron para realizar una primera aproximación al diseño de la plataforma. Fueron de gran utilidad para relevar requerimientos y lograr plasmar la idea de lo esperado.

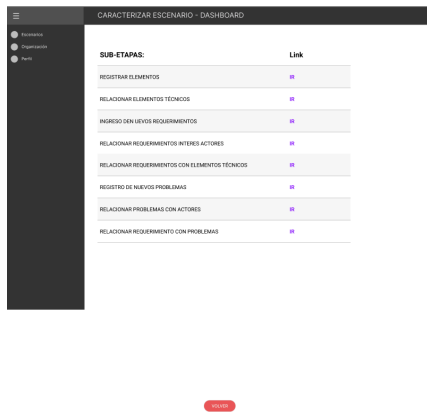


Figura 45: Etapa 1

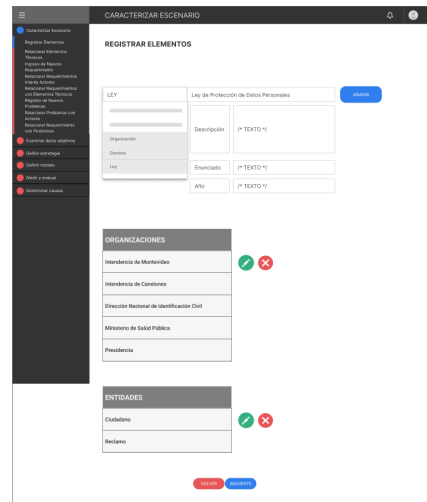


Figura 46: Etapa 1

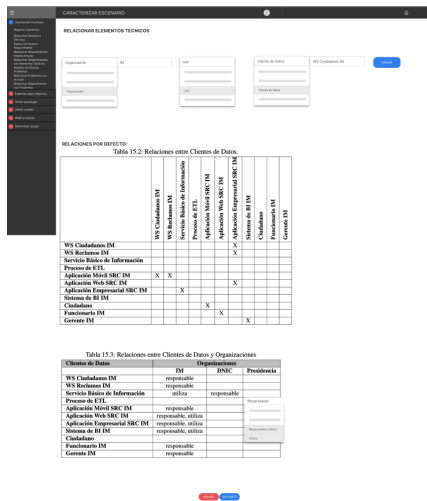


Figura 47: Etapa 1

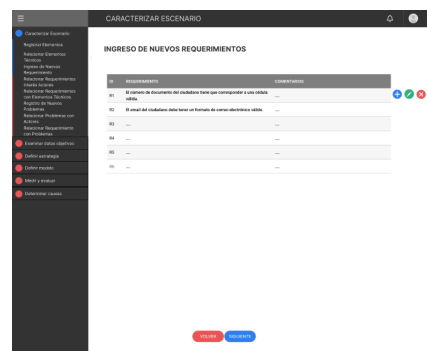


Figura 48: Etapa 1

RELACIONAR REQUERIMIENTOS INTERES ACTORES

Actores	Requerimientos	
	R1	R2
Ciudadanos Zonas Costeras		
Funcionarios IM Ventanilla	X	X
Funcionarios IM Áreas	X	X
Gerentes IM	X	X
Sección Evaluación y Monitoreo IM	X	X

Figura 49: Etapa 1

RELACIONAR REQUERIMIENTOS CON ELEMENTOS TÉCNICOS

Elementos	Requerimientos	
	R1	R2
Entidades de Negocio		X
Ciudadano	X	X
Reclamo		
Atributos		
Ciudadano nombres		
Ciudadano apellido		
Ciudadano numDoc	X	
Ciudadano email		X
Reclamo id		
Reclamo motivo		
Reclamo fecha		
Reclamo correoX		
Reclamo correoY		
Colecciones de Datos		
Base de Datos SRC IM	X	X
Clientes de Datos		
WS Ciudadanos IM	X	X
WS Reclamos IM		
Servicio Básico de Información	X	X
Proceso Gestión de Reclamos	X	X
Aplicación Móvil SRC IM	X	X
Aplicación Web SRC IM	X	X
Aplicación Emprendedor SRC IM	X	X
Ciudadano	X	X
Funcionario IM	X	X
Organizaciones		
IM	X	X
DNIC	X	X
Proceso de Negocio		
Gestión de Reclamos	X	X

Figura 50: Etapa 1

REGISTRO DE NUEVO PROBLEMA

ID	Nombre Problema	Descripción
P1	Nombre problema	El sistema no permite la consulta, según lo establecido en los tiempos de respuesta por los usuarios y según los estándares.
P2	Nombre problema	...
P3	Nombre problema	...
P4	Nombre problema	...
P5	Nombre problema	...

Figura 51: Etapa 1

RELACIONAR PROBLEMAS CON ACTORES

Actores	Problemas Identificados								
	P1	P2	P3	P4	P5	P6	P7	P8	P9
Actores de Datos									
Ciudadanos Zonas Costeras									
Funcionarios IM Ventanilla	X	X	X	X	X				
Funcionarios IM Áreas	X	X	X	X	X				
Gerentes IM		X	X	X	X				
Sección Evaluación y Monitoreo						X	X	X	X

Figura 52: Etapa 1

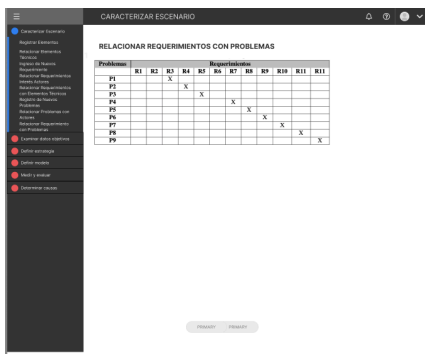


Figura 53: Etapa 1

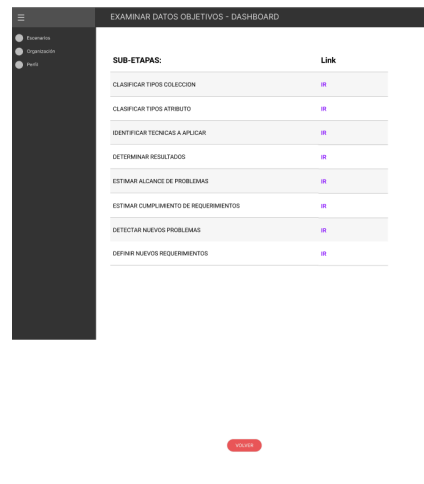


Figura 54: Etapa 2

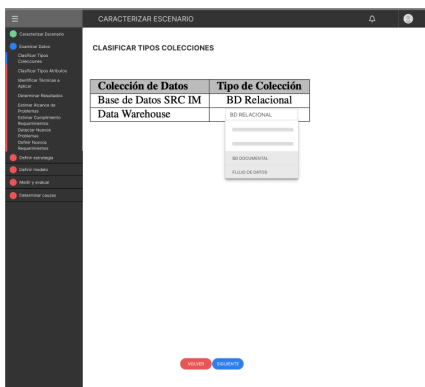


Figura 55: Etapa 2

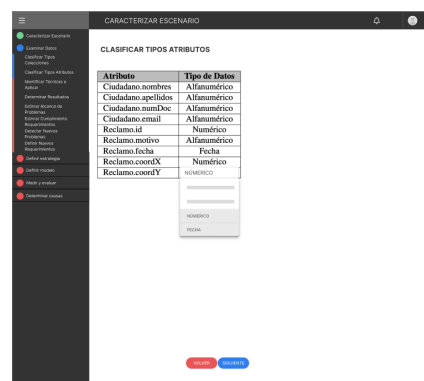


Figura 56: Etapa 2

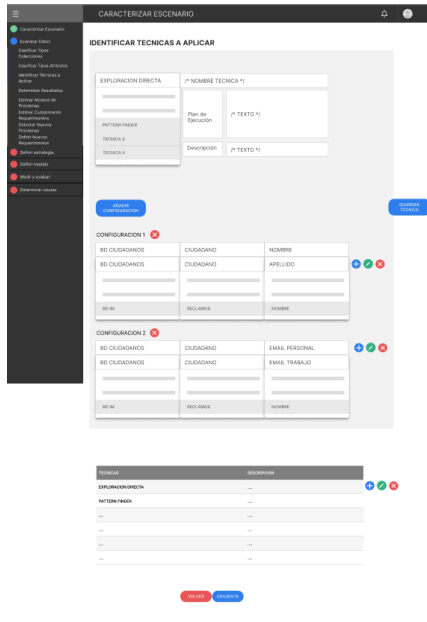


Figura 57: Etapa 2

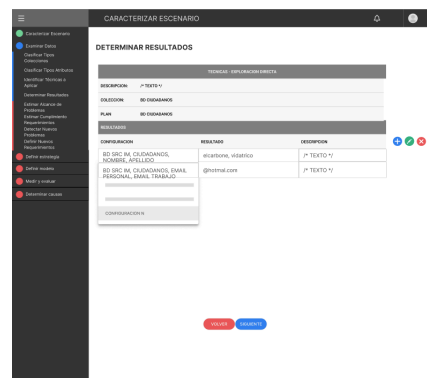


Figura 58: Etapa 2

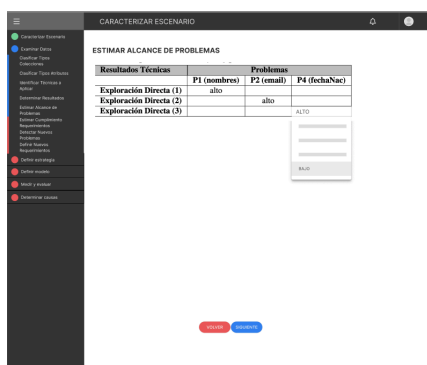


Figura 59: Etapa 2

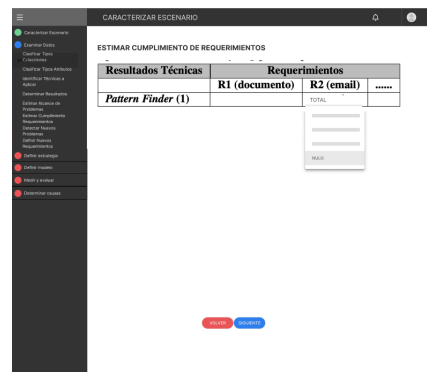


Figura 60: Etapa 2

CARACTERIZAR ESCENARIO

DESCRIBIR ESTRATEGIA

Requerimiento	Paso 1	Paso 2
R1	I	
R2		X
R3		X
R4	X	
R5	X	
R6	X	
R7		X
R8		X
R9	X	
R10	X	
R11		X
R12		X
R13	X	
R14	X	
R15	X	

Figura 65: Etapa 3

EXAMINAR DATOS OBJETIVOS - DASHBOARD

SUB-ETAPAS:

Sub-etapa	Link
DEFINICION DE MODELO PARA LA ESTRATEGIA	Link
INGRESAR METRICAS	Link
INGRESO DE PERFILES	Link

Figura 66: Etapa 4

CARACTERIZAR ESCENARIO

ASIGNAR PRIORIDADES

Exactitud	Consistencia	Complejidad	Unicidad	Frecuencia
Consistid Sinárctica				
Consistid Sinárctica				
Posición				
Exactid Posicional Absoluta				
Exactid Posicional Relativa				
Integridad Inter-entidad				
Integridad Intra-entidad				
Consistencia Topológica				
Cobertura				
Densidad				
Comisión				
Distribución				
Contracción				
Actualidad				
Oportunidad				

R#	Requisito	Factor	Métrica General / Específica
R1	Exactitud	Consistid Sinárctica	Formas (Neuro)Documentos, DMS)
R4	Exactitud	Consistid Sinárctica	ComunicadSinárctid
R5	Exactitud	Consistid Sinárctica	ComunicadSinárctid
R6	Exactitud	Consistid Sinárctica	ComunicadSinárctid
R9	Unicidad	No redundancia	EntidadRelacionad
R10	Complejidad	Densidad	Notas
R11	Consistencia	Integridad Inter-entidad	RegladSinárctid
R14	Consistencia	Integridad Intra-entidad	RegladSinárctid
R15	Exactitud	Consistid Sinárctica	Entidad

Figura 67: Etapa 4

CARACTERIZAR ESCENARIO

INGRESAR METRICA

METRICA GEN.

METRICA ESP. Símbolica

METRICA GEN. Reglas Configurad Símb.

Orígenes Formas/Símbolos

Tipo Res.

Configuración

ES CIUDADANOS	CIUDADANO	NOMBRE
ES CIUDADANOS	CIUDADANO	APPELLIDO
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 68: Etapa 4

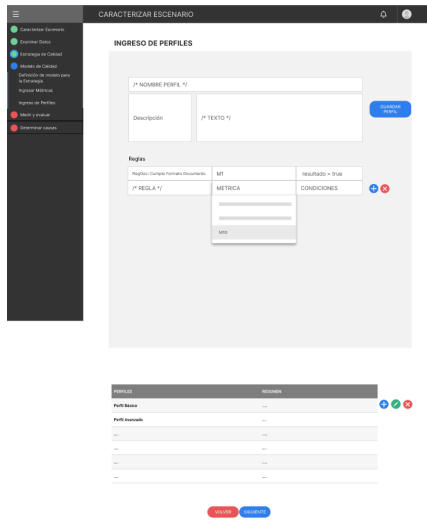


Figura 69: Etapa 4

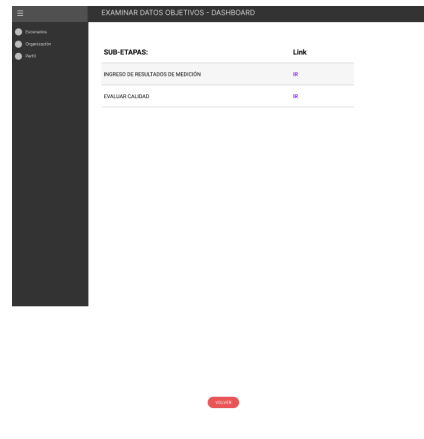


Figura 70: Etapa 5

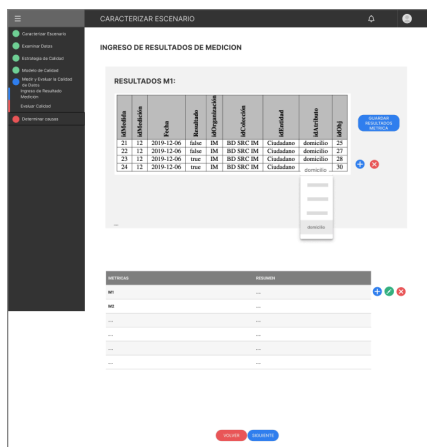


Figura 71: Etapa 5

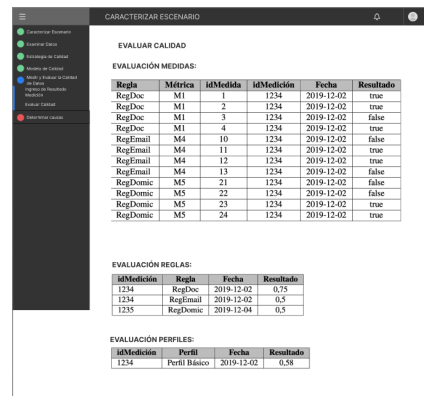


Figura 72: Etapa 5

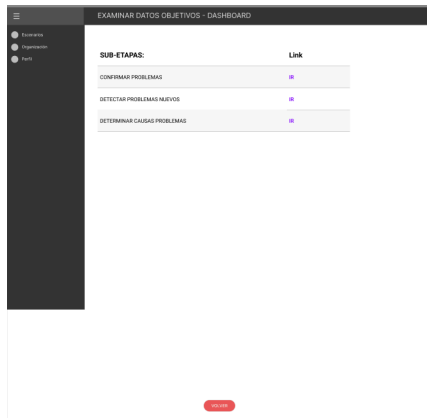


Figura 73: Etapa 6

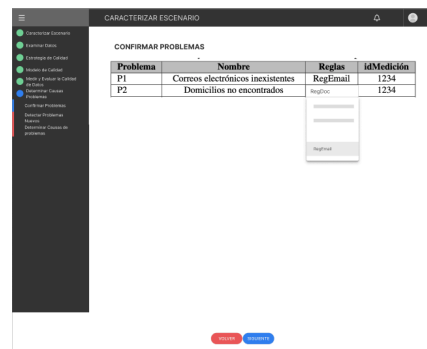


Figura 74: Etapa 6

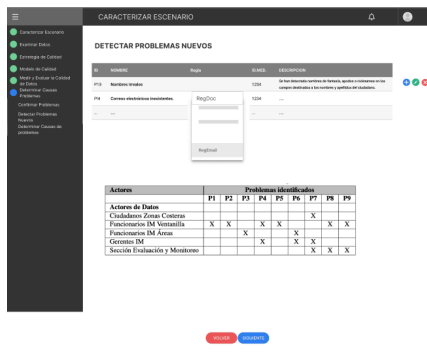


Figura 75: Etapa 6

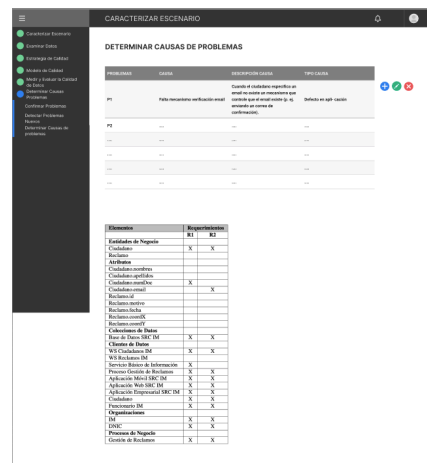


Figura 76: Etapa 6