UNIVERSIDAD DE LA REPÚBLICA URUGUAY

FACULTAD DE INGENIERÍA

Herramienta de apoyo para la aplicación de la metodología CaDQM, centrada en la definición del contexto de datos

Informe de Proyecto de Grado presentado por

Martina Revello, Fernando Rábago

en cumplimiento parcial de los requerimientos para la graduación de la carrera de Ingeniería en Computación de Facultad de Ingeniería de la Universidad de la República

Supervisores

Flavia Serra
Adriana Marotta

Montevideo, 23 de septiembre de 2025

Agradecimientos

Queremos agradecer a nuestras familias y amigos por el apoyo incondicional, la paciencia y el aliento que nos brindaron durante todo el proyecto de grado y a lo largo de la carrera. Agradecemos a nuestras tutoras, Flavia Serra y Adriana Marotta, por la dedicación y el acompañamiento en cada etapa de este proyecto.

Resumen

La literatura demuestra que la mayoría de las actividades de gestión de calidad de datos (CD), están influenciadas por el contexto, sin embargo, muy pocas metodologías consideran el contexto de los datos evaluados. Este proyecto diseña e implementa una herramienta de *software* que apoya la ejecución de la metodología *Context-aware Data Quality Management (CaDQM)*, definida en una tesis de doctorado, la cual guía este proyecto. *CaDQM* consta de 3 fases: Fase 1 - *DQ Planning*, Fase 2 - *DQ Assessment* y Fase 3 - *DQ Improvement*. En particular, esta herramienta ejecuta la fase 1 que consta de tres etapas: ST1 - *Elicitation*, ST2 - *Data Analysis* y ST3 - *User Requirements Analysis*. Al mismo tiempo, otro proyecto de grado se centró en el desarrollo de una herramienta que ejecuta la Fase 2 - *DQ Assessment*. Ambos proyectos trabajaron en una base de datos común, que considera todos sus requerimientos.

La solución desarrollada utiliza una arquitectura con *Django Framework* como *backend*, integrando herramientas de *data profiling*, y un *frontend* desarrollado en *React* que permite interactuar con la herramienta de manera intuitiva. Adicionalmente, esta herramienta incorpora inteligencia artificial para el análisis de documentos, sugerencias de problemas de CD y componentes de contexto.

La validación de la herramienta fue realizada mediante la ejecución de dos casos de estudio. En el primero se utiliza un *dataset* de libros y reseñas, y en el segundo un *dataset* de medicamentos. El objetivo del primer caso de estudio fue verificar el correcto funcionamiento de la herramienta, mientras que el segundo caso de estudio tiene dos objetivos. El primer objetivo es verificar la interoperabilidad entre las herramientas desarrolladas por los dos proyectos de grado y el segundo, es analizar, para un mismo *dataset*, las diferencias y similitudes entre el modelo de contexto obtenido con la herramienta y el modelo de contexto definido por un grupo de expertos de dominio y de CD.

El análisis realizado demuestra la viabilidad de utilizar la herramienta que apoya la ejecución de la metodología *CaDQM*, reduciendo significativamente el tiempo y esfuerzo manual requerido para registrar los datos recabados mediante la aplicación de la metodología. Los resultados obtenidos muestran la efectividad de la herramienta propuesta. Además, dejan en evidencia que, a la hora de realizar tareas de gestión de CD, es importante trabajar con los expertos de dominios y con los usuarios de los datos. Esto último, totalmente alineado con las necesidades identificadas en la bibliografía de CD.

Palabras clave: Calidad de datos, CaDQM, Contexto, *Data profiling*, Inteligencia artificial, metodología de calidad de datos.

Índice general

Capítulo 1

Introducción

En este capítulo se introducen las principales características del proyecto, definiendo el problema que lo motiva, los objetivos generales y específicos que guiaron el trabajo y los resultados esperados. Por último, se describe la estructura del documento.

1.1. Motivación

Los datos ocupan un papel fundamental en las organizaciones de hoy en día, ya que la gran mayoría de la toma de decisiones están basadas en ellos. Una mala calidad de datos (CD) puede generar impactos económicos significativos, debido a decisiones erróneas basadas en datos incompletos o inconsistentes. La CD impacta en la eficiencia y eficacia de cualquier organización, dado que la gran mayoría de las decisiones se basan en los sistemas de información que dependen de que sus datos sean confiables.

La CD no es una propiedad de los datos en sí mismos, sino que es subjetiva y depende del contexto en que se usan estos datos. A pesar de la relevancia del contexto en la CD, el relevamiento de la bibliografía muestra que pocas metodologías de gestión de CD consideran al contexto de los datos en sus actividades. Sin embargo, la metodología de gestión de CD *Context-aware Data Quality Management (CaDQM)*, definida en la tesis de Doctorado de Flavia Serra [1], se centra en la definición y consideración del contexto de los datos en todas sus etapas.

CaDQM ha sido ejecutada en distintas experiencias (curso de CD [2] y módulos de taller), por usuarios con diferentes niveles de conocimiento sobre CD. En todas estas experiencias se identificó la necesidad de contar con una herramienta que apoye la ejecución de la metodología. Como la ejecución debía realizarse manualmente, esto implicaba mucho tiempo de registro de la información y definición del contexto.

1.2. Descripción del proyecto

A partir de la necesidad identificada durante la ejecución manual de *CaDQM*, se plantea la posibilidad de contar con una herramienta que brinde apoyo a expertos de CD durante la ejecución de la metodología. *CaDQM* propone 3 fases: Fase 1 - *DQ Planning*, Fase 2 - *DQ Assessment* y Fase 3 - *DQ Improvement* y cada una de estas fases propone 3 etapas. Específicamente, este proyecto se enfoca en la creación de una herramienta que ejecute la Fase 1 - *DQ Planning*. En paralelo, otro proyecto de grado desarrolla una herramienta para ejecutar la Fase 2 - *DQ Assessment*, ambas herramientas comparten la base de datos. Esto es así porque la salida de la fase 1 (*dataset* cuya calidad será evaluada, lista de problemas de CD y definición del

modelo de contexto) es la entrada de la fase 2. De esta forma, una base de datos común garantiza que los resultados generados en la Fase 1 - *DQ Planning* puedan ser consumidos directamente por la Fase 2 - *DQ Assessment*.

Por lo tanto, la herramienta propuesta en este proyecto permite gestionar y registrar las tres etapas de la Fase 1 - *DQ Planning*: *Elicitation*, *Data Analysis*, y *User Requirements Analysis*. Además, esta solución integra herramientas de *data profiling* para el análisis de los datos e inteligencia artificial (IA). Esta última es usada para asistir al usuario en la identificación de problemas de CD y de componentes de contexto (que definen al modelo de contexto), así como también en la estimación preliminar de la CD. Los resultados obtenidos con IA surgen del análisis de documentos e información que el usuario registra a lo largo de las diferentes etapas.

1.3. Objetivos

El objetivo general de este proyecto es diseñar y desarrollar una herramienta que dé apoyo a expertos en CD en el registro de los resultados obtenidos durante la ejecución de la Fase 1 - *DQ Planning* de la metodología de gestión de CD, dependiente del contexto, *Context-aware Data Quality Management (CaDQM)*.

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

- Realizar un análisis de herramientas de aplicación de metodologías de calidad de datos, haciendo énfasis en el registro de datos.
- Diseñar una herramienta, en base al análisis realizado, asegurando la interoperabilidad con otra herramienta, desarrollada en paralelo por otro proyecto de grado, encargado del desarrollo de la Fase 2 - *DQ Assessment* de *CaDQM*.
- Implementar un prototipo de la herramienta.
- Validar la herramienta ejecutando dos casos de estudio para verificar:
  - La correcta ejecución de cada etapa.
  - La interoperabilidad de las herramientas propuestas por ambos proyectos.
  - La correctitud del modelo de contexto obtenido con la herramienta.

1.4. Resultados esperados

Como resultado de este proyecto, se espera obtener un prototipo de la herramienta de aplicación de la Fase 1 - *DQ Planning* de la metodología *CaDQM*. Esta herramienta debe permitir el registro de los resultados de cada una de las etapas, y debe poder integrarse con el prototipo de herramienta del otro proyecto de grado, de forma transparente para el usuario. Se entregará la implementación y la documentación completa del proyecto, la cual incluirá un manual de usuario detallado y dos casos de estudio que validarán el funcionamiento y los resultados de la herramienta, utilizando distintas fuentes de datos.

1.5. Estructura del documento

El documento se divide en seis capítulos y Anexos. A continuación se describe cada uno de ellos:

- Capítulo 2: se presenta el marco teórico y antecedentes del proyecto, describiendo los principales conceptos abordados.

- Capítulo 3: se describe el análisis y relevamiento de requerimientos y además el diseño y la arquitectura de la herramienta.
- Capítulo 4: se detalla la implementación donde, entre otros tópicos importantes, se describe la integración de herramientas de inteligencia artificial.
- Capítulo 5: se presentan las pruebas realizadas a partir de dos casos de estudio.
- Capítulo 6: se exponen las conclusiones y el trabajo futuro.
- Anexo A: se presenta el manual de usuario y configuración.
- Anexo B: se presentan los resultados de la validación de los dos casos de estudio.
- Anexo C: se presentan los prompts utilizados en el modelo de IA.

Capítulo 2

Marco teórico y antecedentes

En este capítulo se describen los conceptos fundamentales de este proyecto. En primer lugar se introduce la calidad de los datos, su importancia y cómo se gestiona. Luego, se presenta el concepto de contexto de los datos y su relevancia en la calidad de los datos, se describe la metodología *Context-aware Data Quality Methodology* en la que se basa este proyecto y por último, se detallan las herramientas existentes que fueron evaluadas.

2.1. Calidad de datos

La calidad de los datos (CD) se define comúnmente como *fitness for use*, lo que implica que la calidad no es una propiedad del dato en sí mismo, sino que depende de la situación o propósito para el cual se utiliza [3]. Esto significa que los datos son adecuados para su propósito según la perspectiva del usuario final, quien evalúa si la información cumple con sus necesidades particulares. Por tanto, la calidad de los datos es subjetiva y varía según cómo se usan los datos, quién los usa y para qué. La noción de buena o mala CD no puede separarse del contexto en el que se producen o se utilizan [4].

En general, los usuarios de los datos esperan que estos sean relevantes para su uso, correctos y sin inconsistencias, que estén lo más actualizados posible, que se presenten de manera adecuada para sus aplicaciones y que sean de fácil acceso. Es común que las personas asocien la CD únicamente con su exactitud, por ejemplo, en errores de escritura. Sin embargo, los datos pueden no contener errores de este tipo e igualmente tener una mala calidad, como por ejemplo, estar desactualizados o incompletos [2].

2.1.1. Problemas de calidad

Los datos tienen un papel fundamental en las organizaciones, actuando como materia prima esencial para la toma de decisiones operativas y estratégicas. Su calidad impacta directamente en la eficiencia y eficacia de la organización, ya que la mayoría de las decisiones se basan en sistemas de información que dependen de datos confiables [2].

La CD es crucial para la toma de decisiones en una organización, ya que estos guían procesos de todo tipo, desde operaciones diarias hasta estrategias a largo plazo. Además, la CD es fundamental en la realización de tareas como *Machine Learning*, donde los modelos predictivos se entrenan a partir de los datos y los problemas de calidad pueden generar resultados incorrectos o sesgados. Por otro lado, la mala CD puede tener un gran impacto económico en las organizaciones. Existen estudios que muestran que los problemas de CD pueden tener costos muy significativos para las

empresas debido a decisiones erróneas basadas en información incompleta o inconsistente [5]. Los problemas de calidad de datos pueden manifestarse de distintas formas, como por ejemplo, datos incorrectos o que no reflejan la realidad, información incompleta con campos vacíos, o datos desactualizados que no reflejan el estado actual.

Estos problemas pueden generarse durante distintas etapas del manejo de los datos. En la **producción**, los problemas pueden deberse a la recolección de datos mediante ingreso humano, representaciones distintas del mismo objeto de la realidad (por ejemplo, diferentes formatos de fecha), la falta de actualización de los datos o la ausencia de un responsable de su calidad. Durante el **procesamiento**, los problemas pueden surgir al transformar los datos a otras estructuras o formatos, cálculos, o la unión de datos provenientes de distintas fuentes. En el **almacenamiento**, la utilización de distintos formatos para un mismo objeto o la ausencia de formatos definidos puede causar mala calidad, así como el uso de bases de datos mal diseñadas. Por último, en la **utilización**, los problemas de calidad se pueden dar en los cambios en los requerimientos, uso equivocado de los datos, por mala interpretación o aplicación fuera de contexto, problemas de seguridad y acceso, y mal diseño de los sistemas que procesan los datos para su análisis posterior [2].

2.1.2. Gestión de la calidad de datos en un sistema de información

La gestión de la calidad de los datos (GCD) implica evaluar, analizar y mejorar diferentes aspectos de la CD de un sistema de información. Este proceso involucra un conjunto de tareas, que se presentan en la Figura 2.1 [2]. Según los autores en [6], la mayoría de estas actividades son influenciadas por el contexto de los datos.

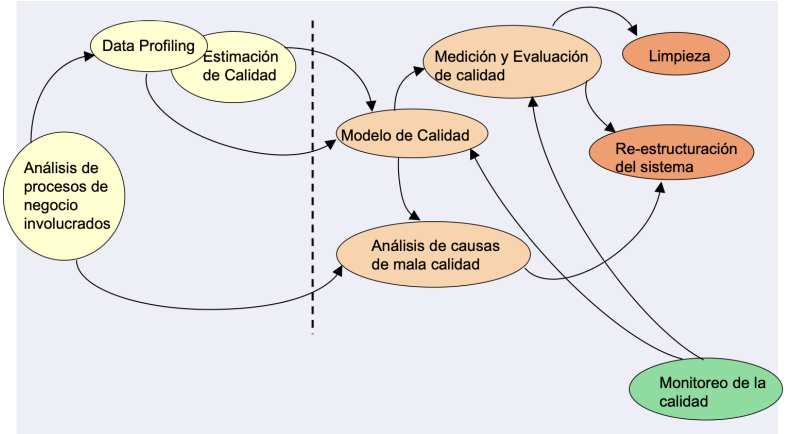

Figura 2.1: Gestión de la calidad de datos en un Sistema de Información. Tomado de [2]

Por una parte se **analizan los procesos de negocio involucrados** para conocer qué procesos generan, procesan y utilizan los datos, y se ejecutan tareas de *data profiling* para obtener un conocimiento inicial sobre las características de los datos y así obtener una **estimación de CD**. Con estos resultados se construye el **modelo de CD**, especificando qué datos se evaluarán, qué características se analizarán y cómo se medirán. Finalmente, se **analizan las causas de la mala CD**, identificando dónde y por qué se originan los problemas.

Otro paso muy importante en el proceso de GCD es la **medición y evaluación** de la calidad para ver el estado actual de los datos. Aquí, se ejecutan las métricas definidas en el **modelo de calidad** y se evalúan los resultados, comparando los valores de calidad obtenidos con otros valores de referencia. A partir de los resul-

tados obtenidos en la evaluación, es posible realizar tareas de **limpieza de datos** y **re-estructuración del sistema** entre otros, para prevenir futuros problemas ya identificados. Luego, se realiza el **monitoreo de la calidad**, volviendo a medir la CD una vez corregidos, ya que pueden solucionarse algunos problemas de calidad pero, podrían surgir otros.

Modelo de calidad de datos

Un modelo de CD define qué características de calidad se van a considerar, sobre qué datos se aplicarán y cómo se medirán. Se define un modelo de calidad para cada conjunto de datos distinto del sistema de información. El modelo es una guía para toda la gestión de calidad y es un componente esencial, ya que genera un marco estructurado para entender, definir y medir la CD en función de las necesidades de los consumidores. Con un modelo bien definido, resulta más fácil evaluar, mejorar y controlar la CD de una organización.

En [2], se define el modelo de calidad a partir de una jerarquía de conceptos, que se presenta en la Figura 2.2. Esta jerarquía se compone de los elementos: **Dimensión**, **Factor**, **Métrica** y **Método**. Los cuales se definen a continuación.

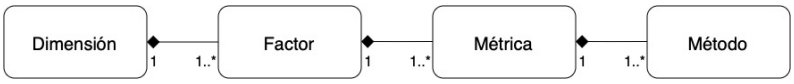

Figura 2.2: Jerarquía de conceptos de calidad. Tomado de [2].

Una **dimensión de calidad** representa una faceta específica de CD a alto nivel. Existen diversas dimensiones de calidad que son relevantes para evaluar los datos. Algunas de las más comunes son:

- Exactitud: Indica qué tan correctos o precisos son los datos.
- Completitud: Indica si los datos contienen toda la información de interés.
- Frescura: Se refiere a si los datos son recientes y actualizados.
- Consistencia: Indica si los datos satisfacen las reglas definidas sobre ellos y sus relaciones.
- Unicidad: Mide el grado de duplicación y no contradicción de los datos.

Cada dimensión se puede dividir en un conjunto de factores. Un **factor de calidad** representa un aspecto particular de la dimensión. Por ejemplo, los factores de la dimensión exactitud son: *exactitud semántica*, *exactitud sintáctica* y *precisión*.

Cada factor puede medirse con diferentes métricas. Las **métricas** definen de qué forma se debe medir un factor de calidad. Al definir una métrica se debe especificar las unidades de medición y la granularidad de la medida. Por ejemplo, una métrica para la exactitud semántica mide si un dato existe en la realidad. Para este caso, las unidades podrían ser 0 (no existe) y 1 (existe), y la granularidad es la celda (si pensamos en una base de datos relacional).

Luego, el **método de medición** es el proceso que implementa una métrica para obtener el valor cuantitativo. Una misma métrica puede ser medida por distintos métodos como se observa en la Figura 2.2. En el ejemplo anterior, para exactitud semántica, para medir si una dirección existe en la realidad es posible crear distintos métodos de medición, por ejemplo, consultando una lista de direcciones válidas.

2.2. Contexto

El término **contexto** tiene múltiples definiciones de distintos autores de diferentes áreas. La etimología de la palabra proviene del latín *contextus*, que significa unión o entrelazado. Se define de forma general, como el conjunto de circunstancias que rodean un suceso o una idea y que son fundamentales para su completa comprensión [7] [8].

Otra definición más precisa que se adapta más a la definición de contexto que buscamos en este proyecto es la de Dey, que define el contexto como cualquier información que puede usarse para caracterizar la situación de una entidad. Una entidad es una persona, un lugar o un objeto que se considera relevante para la interacción entre un usuario y una aplicación, incluidos el propio usuario y la aplicación [9].

En este proyecto, se considera la definición de contexto dado en [1], donde el contexto es la información acerca de los datos cuya calidad es evaluada, y de otros elementos que están fuertemente relacionados a estos datos.

2.2.1. Relevancia del Contexto en calidad de datos

Para llevar a cabo las actividades de gestión de calidad de datos descritas en la Subsección 2.1.2, existen distintas metodologías. Una metodología ofrece un conjunto de pautas y técnicas que, a partir de información de entrada que describe un contexto de aplicación dado, establece un proceso para evaluar y mejorar la calidad de los datos [10].

La calidad de los datos depende en gran medida del contexto, como se explicó en la Sección 2.1, lo que significa que la evaluación de la CD puede variar drásticamente entre diferentes contextos. La mayoría de las actividades de GCD están influenciadas por el contexto de los datos. Sin embargo, muy pocas metodologías de CD consideran explícitamente el contexto de los datos evaluados, y cuando lo hacen, el contexto se aborda solo en sus etapas iniciales [1].

Un ejemplo es la metodología *Comprehensive Data Quality (CDQ)* [11], que aborda el contexto, pero lo hace principalmente en su fase inicial. Esta metodología reconoce la importancia del contexto, pero se limita a la etapa de relevamiento de requerimientos, donde se recopila información del contexto. Luego, el contexto no se actualiza en las siguientes fases del proceso de gestión de calidad de datos [1].

Por otro lado, la tesis de Doctorado de Serra [1] en la que se enmarca este proyecto, propone una metodología llamada *Context-aware Data Quality Methodology (CaDQM)*, que considera el contexto de los datos en todas sus fases.

Esta tesis propone un modelo de contexto, definido como un conjunto de componentes:

- Dominio de aplicación: El área específica donde se utilizan los datos.
- Tipos de usuarios: Los diferentes tipos de usuarios que interactúan con los datos y sus roles.
- Tareas: Las actividades específicas para las que se utilizan los datos.
- Filtrado de datos: Los criterios para seleccionar subconjuntos de datos relevantes.
- Requerimientos de CD y del sistema: Las expectativas específicas sobre la calidad de los datos y los requerimientos técnicos de los sistemas que los manejan.
- Reglas de negocio: Las restricciones y lógicas impuestas por la organización o el dominio.

- Metadatos generales: Información descriptiva sobre los datos.
- Metadatos de CD: Información sobre la calidad de los datos (ejemplos: resultados de mediciones de calidad).
- Otros datos: Datos relacionados con los datos que se están evaluando y que los complementan.

Además, define las relaciones entre estos componentes, por ejemplo, las tareas y las reglas de negocio están vinculadas al dominio de aplicación. Los componentes del contexto y las relaciones entre ellos se presentan en la Figura 2.3.

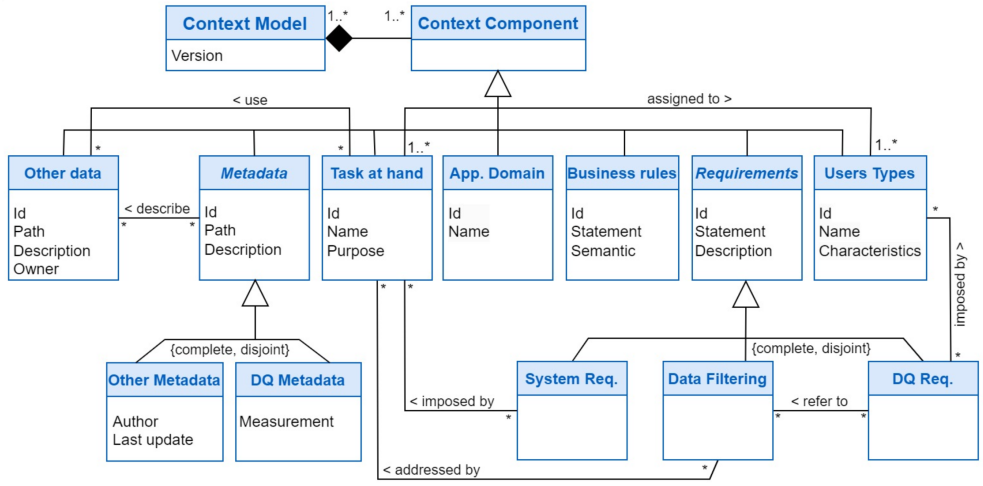

Figura 2.3: Modelo de contexto. Tomado de [1].

Los componentes de contexto influyen en la definición del modelo de calidad de los datos, incluyendo la selección de dimensiones y factores, así como la definición de métricas y métodos de CD. El modelo de contexto sirve para apoyar al experto de calidad a lo largo de las distintas actividades del proceso de GCD.

En la siguiente sección se profundiza en la metodología de gestión de calidad de datos dependiente del contexto *CaDQM*.

2.3. *Context-aware Data Quality Methodology*

La metodología *Context-aware Data Quality Management (CaDQM)* se refiere a la gestión de la CD considerando explícitamente el contexto en el que los datos son utilizados. Este enfoque permite adaptar la evaluación y mejora de la calidad en escenarios específicos, haciendo más eficiente y relevante el proceso de GCD.

A continuación se presentan y describen las 3 fases de la metodología *CaDQM*, una descripción completa de *CaDQM* puede ser consultada en [1]:

- **Planificación de la CD (*DQ Planning*):** En esta fase principalmente, se relevan todos los elementos de la organización. Esta tarea permite realizar una identificación inicial de los componentes del contexto, tales como reglas de negocio, requerimientos específicos de calidad de datos, requerimientos del sistema, condiciones de filtrado de datos y tipos de usuarios involucrados. En segundo lugar, se realiza una estimación de la CD, mediante técnicas de *data profiling* y análisis de requerimientos del sistema. Finalmente, se interactúa con los usuarios expertos en el dominio, para relevar y analizar los requerimientos de usuario, definir nuevos problemas de CD y actualizar el modelo de contexto.

- **Evaluación de la CD (*DQ Assessment*):** En esta fase se define el modelo de calidad, seleccionando las dimensiones y factores, basándose en el contexto identificado. Se ejecutan actividades de medición y evaluación, aplicando métricas específicas de calidad, para luego crear un diagnóstico. Esta fase busca obtener resultados precisos y relevantes de la calidad de los datos teniendo en cuenta el contexto de los datos.
- **Mejora de la CD (*DQ Improvement*):** En esta fase se plantean acciones concretas para mejorar la calidad de los datos, basadas en los resultados obtenidos en las fases anteriores. Estas acciones están directamente influenciadas por los componentes de contexto, asegurando que las estrategias de mejora se alineen efectivamente con las necesidades reales de la organización.

En la Figura 2.4 se presentan las fases de *CaDQM* con sus respectivas etapas. El presente trabajo se basa únicamente en la primera fase: *DQ Planning*.

A continuación se describen en profundad la Fase 1, y se describen en forma resumida las Fase 2 y 3.

2.3.1. Fase 1: *DQ Planning*

Esta fase establece los fundamentos sobre los cuales se realizarán las mediciones y evaluaciones futuras. Está estructurada en 3 etapas, cada una con actividades bien definidas: *ST1 - Elicitation*, *ST2 - Data Analysis*, *ST3 - User Requirements Analysis*.

En esta fase no es obligatorio ejecutar las etapas de forma secuencial. Como se muestra en la Figura 2.4, después de completar *STI - Elicitation*, es posible continuar tanto con la etapa *ST2 - Data Analysis* como con la *ST3 - User Requirements Analysis*, las cuales pueden, a su vez, ejecutarse de forma paralela. Esto permite alternar entre la etapa 2 y la 3, ya que los componentes de contexto que surgen en una pueden enriquecer el análisis de la otra. Además, es posible omitir la ejecución de una de estas dos etapas, si se omite *ST3 - User Requirements Analysis* se usa un enfoque basado únicamente en los datos, o si se omite *ST2 - Data Analysis* se usa un enfoque basado en los requerimientos de los usuarios. Por ejemplo, cuando no hay muchos usuarios de un sector, es probable que se trabaje con enfoque en los datos. Finalmente, se puede avanzar a la Fase 2 *DQ Assessment* de la metodología tanto desde la *ST2 - Data Analysis* como desde la *ST3 - User Requirements Analysis*.

A continuación, se describen las etapas de esta fase.

Etapa 1: *ST1 Elicitation*

En esta etapa inicial se seleccionan y recopilan todos los datos de la organización. Además, se realiza un análisis inicial de los elementos de la organización como procesos, reglas de negocio, etc, y se identifican potenciales problemas de CD. Se establece una definición inicial del modelo de contexto que es actualizado y utilizado en etapas posteriores.

Actividades:

- **A01** - Selección de los datos a ser evaluados: Se identifican todos los conjuntos de datos relevantes para la organización. Finalmente, se selecciona el conjunto de datos cuya calidad es evaluada, llamados *data at hand*.
- **A02** - Análisis de los elementos de la organización: Se identifican y analizan detalladamente todos los elementos de la organización relacionados con los *data at hand*. Esto incluye dominios de aplicación, descripciones de sistemas, servicios, fuentes de datos, metadatos, procesos de negocio, estándares, reglas de negocio,

restricciones, requerimientos de calidad de datos, problemas ya reportados de calidad de datos y características específicas de los usuarios.

- **A03** - Identificación inicial de problemas de calidad de datos: Se documentan problemas de CD previamente reportados o detectados durante el análisis de los elementos de la organización.
- **A04** - Definición inicial del modelo de contexto: Se identifican los componentes del modelo de contexto a partir del análisis realizado en las actividades anteriores. Algunos componentes podrían no estar plenamente identificados en esta etapa inicial, ya que el modelo de contexto se refina en las etapas de las 2 primeras fases.

Los resultados de esta etapa son: *Data at hand*, un reporte inicial sobre los problemas de CD detectados y una primera definición del modelo de contexto.

Etapa 2: *ST2 Data Analysis*

En la segunda etapa, *ST2 Data Analysis*, se abordan técnicas de análisis y *data profiling* para identificar detalladamente nuevos problemas de CD y para realiza una estimación de la CD. En esta etapa, se actualiza el modelo del contexto con información obtenida en el análisis.

Actividades:

- **A05** - *Data profiling*: Se realiza un análisis detallado de los datos para descubrir patrones, inconsistencias y problemas de CD mediante técnicas estadísticas y herramientas automatizadas. Este también se puede complementar con consultas SQL directamente sobre los datos, entre otras técnicas de análisis de datos.
- **A03** - Identificación de problemas de CD: En base a los resultados de *data profiling*, se identifican y documentan los problemas de CD encontrados en los conjuntos de datos analizados.
- **A06** - Estimación inicial del estado actual de la calidad de datos: Se evalúa el grado o nivel general de calidad de los datos analizados, proporcionando una base para la definición del modelo de CD.
- **A07** - Actualización del modelo de contexto: Se actualiza y refina el modelo del contexto previamente definido, incorporando información precisa obtenida del análisis realizado en esta etapa.

Los resultados de esta etapa son: reporte detallado del análisis realizado, una actualización del reporte de problemas de CD y un modelo de contexto actualizado y refinado.

Etapa 3: *ST3 User Requirements Analysis*

El objetivo de esta etapa es la interacción con los usuarios de los datos y expertos del dominio, para identificar claramente sus requerimientos respecto a la calidad de los datos. Esta interacción permite detectar nuevos problemas de CD desde la perspectiva de los usuarios, permitiendo a su vez, actualizar el modelo de contexto.

Actividades:

- **A08** - Interacción con los usuarios: Interacción directa con usuarios de los datos para obtener requerimientos específicos sobre CD.
- **A03** - Identificación de problemas de calidad de datos: Identificación de problemas específicos de CD reportados por los usuarios.

- **A07** - Actualización del modelo de contexto: Nueva actualización del modelo de contexto, incorporando nuevos componentes identificados a partir de la interacción con los usuarios.

Los resultados obtenidos en esta etapa son: reporte de análisis detallado sobre los requerimientos específicos de los usuarios, una actualización del reporte de problemas específicos de CD, y una nueva actualización del modelo de contexto, según los requerimientos identificados.

2.3.2. Fase 2: *DQ Assessment*

El propósito principal de esta fase es, a partir del modelo de contexto definido en la fase anterior, medir y diagnosticar la calidad de los datos. Esta fase se divide en 3 etapas: *ST4 - DQ Model Definition*, *ST5 - DQ measurement*, *ST6 - DQ assessment*, que se describen a continuación.

Etapa 4: *ST4 - DQ Model Definition*

En esta etapa se seleccionan y priorizan los problemas de CD identificados en la Fase 1 - *DQ Planning*. A partir de los problemas identificados y del modelo de contexto ya definido, se define el modelo de CD. Para eso, se seleccionan las dimensiones y factores. Luego, para cada factor, se definen las métricas de CD donde se especifica como se mide, a que granularidad y el dominio del resultado. Por último, se definen los métodos que implementan las métricas.

Los resultados de esta etapa son: problemas de CD clasificados por prioridad y el modelo de CD que considera el contexto específico.

Etapa 5: *ST5 - DQ measurement*

En esta etapa, se diseña la base de datos de metadatos de calidad, donde se van a registrar las mediciones. Luego, se ejecutan los métodos definidos para cada métrica, que tienen como salida valores cuantitativos de la CD, que se van a almacenar en la base de datos antes mencionada. Por último, se actualiza el modelo de contexto con los valores de calidad obtenidos, agregando este nuevo componente de contexto denominado metadatos de calidad.

Los resultados de esta etapa son: la especificación de la base de datos de metadatos de calidad, reporte de medición de CD y el modelo de contexto actualizado

Etapa 6: *ST6 - DQ assessment*

En esta última etapa de la fase 2, se evalúa la CD a partir de los resultados obtenidos en las etapas anteriores. Para eso, se definen umbrales y enfoques de evaluación basados en los requerimientos de calidad definidos por los usuarios, buscando asignar valor cualitativo a la evaluación. Distintos tipos de usuario pueden definir diferentes enfoques de evaluación dependiendo de sus necesidades. Luego, se comparan las mediciones con umbrales antes definidos, para asignar un resultado cualitativo a la calidad de los datos, por ejemplo, “buena”, “regular”, “mala”. Para finalizar, también se almacenan los resultados cualitativos en la base de metadatos de calidad.

Los resultados de esta etapa son: reporte de evaluación de CD.

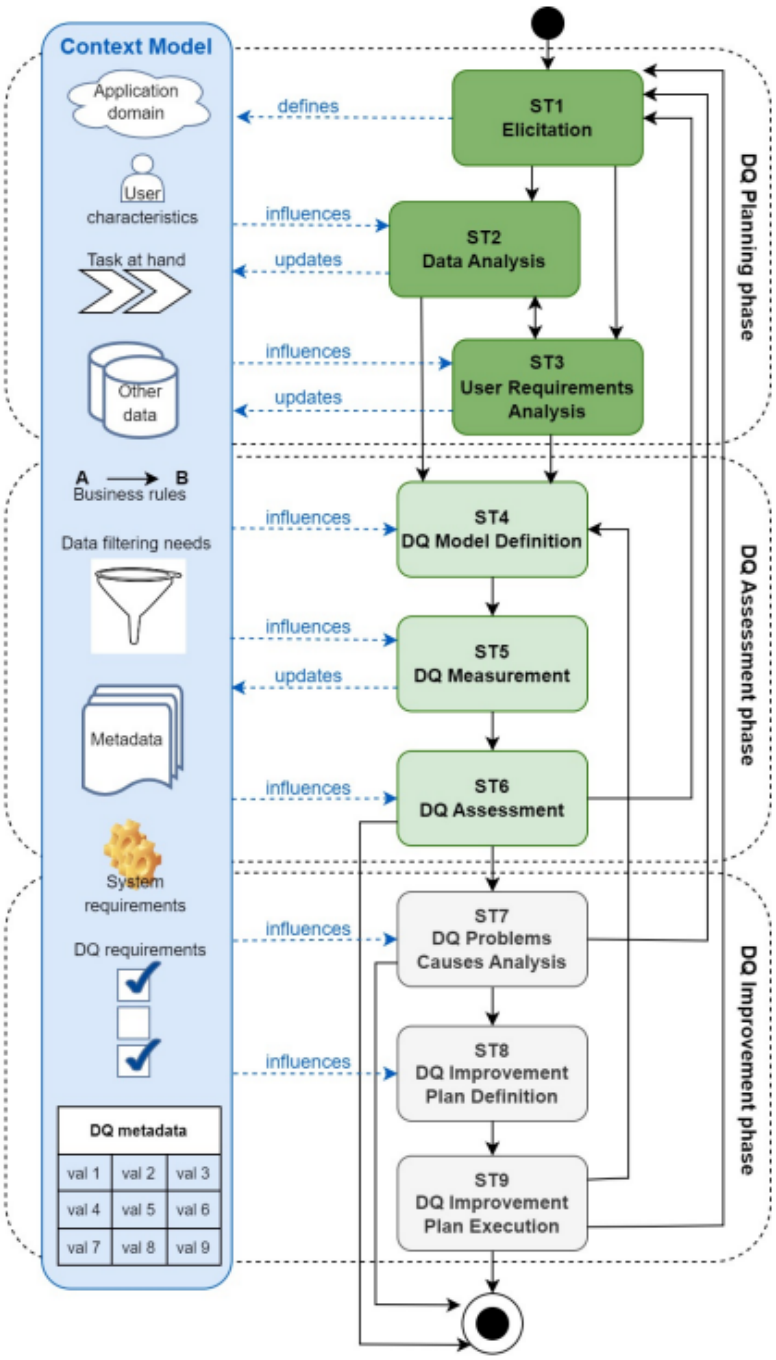

Figura 2.4: Metodología *CaDQM*. Tomada de [1].

2.3.3. Fase 3: *DQ Improvement*

Esta fase se centra en el análisis de las causas de los problemas de CD, en la definición de un plan de mejora y su ejecución. Esta se divide en 3 etapas: *ST7 - DQ Problems Causes Analysis*, *ST8 - DQ Improvement Plan Definition*, *ST9 - DQ Improvement Plan Execution*. Estas etapas se describen a continuación:

Etapa 7: *ST7 - DQ Problems Causes Analysis*

En esta etapa se busca analizar las causas de los problemas de CD, intentando vincular los resultados de las mediciones con los problemas detectados, para encontrar sus orígenes. Además, se priorizan los problemas a resolver, dependiendo de que tan críticos sean y cuales sean sus causas.

Los resultados de esta etapa son: reporte de problemas de CD seleccionados y priorizados con sus causas raíz.

Etapa 8: *ST8 - DQ Improvement Plan Definition*

En esta etapa se define un plan de mejora de la CD. Además se evalúan y seleccionan las técnicas y estrategias de mejora, se analiza el costo de estas técnicas frente al costo de la mala CD. Finalmente, se priorizan y eligen las técnicas más adecuadas que se van a utilizar en el siguiente paso.

Los resultados de esta etapa son: reporte de análisis de costos y un reporte del plan de mejora de CD.

Etapa 9: *ST9 - DQ Improvement Plan Execution*

Una vez definido el plan de mejora, se implementa el plan de acción, poniendo en práctica todas las técnicas y estrategias definidas en la etapa anterior.

Al finalizar la fase 3, es posible finalizar el proceso de GCD. Como se observa en la Figura 2.4, *CaDQM* permite monitorear la CD, ejecutando las fases tantas veces como sea necesario. Esto a su vez, permite identificar cambios en el contexto de los datos, así como también nuevas necesidades que requieran actualizaciones en el modelo de CD.

2.4. Herramientas existentes

En esta sección, se describen herramientas de CD actualmente utilizadas. El análisis buscaba conocer las distintas funcionalidades provistas y las características de las interfaces de usuario.

Entre las herramientas gratuitas y de código abierto, está Great Expectations [12]. Esta herramienta funciona con el concepto de "expectativas", que son reglas que los datos deben cumplir. Por ejemplo, es posible crear una regla que diga "esta columna no puede tener valores vacíos" o "los números en esta columna deben ser entre 0 y 100". La herramienta tiene más de 350 reglas ya definidas que pueden ser usadas, desde lo más básico, hasta análisis estadísticos más complejos.

Entre las herramientas comerciales, está Monte Carlo [13]. Esta plataforma se enfoca en monitorear los datos de manera automática, revisando aspectos tales como, si los datos llegan a tiempo, si tienen el volumen esperado, si la estructura es correcta, así como otras características importantes. Además, esta herramienta utiliza inteligencia artificial para aprender cómo se comportan normalmente los datos y comunicar en caso de situaciones inesperadas.

Por otro lado, se analizó *Talend Data Quality* [14], que es una suite completa para gestionar calidad de datos. Esta herramienta permite hacer un análisis inicial de los datos para entender qué representan, limpiarlos (si tuvieran errores), y monitorearlos. Además, ofrece algoritmos que permiten encontrar patrones en los datos, identificar duplicados, y detectar relaciones entre diferentes campos.

El relevamiento que se realizó permitió identificar un problema común a todas las herramientas analizadas: ninguna considera el contexto de los datos.

Capítulo 3

Análisis y diseño de la herramienta a desarrollar

Este capítulo presenta el análisis y diseño de la herramienta desarrollada para implementar las tres primeras etapas de la Fase 1 - *DQ Planning* de la metodología *CaDQM*. Se detallan los requerimientos funcionales identificados, el diseño de la base de datos común desarrollada en colaboración con el equipo de la Fase 2 - *DQ Assessment*, y la arquitectura de la herramienta enfocada en la Fase 1 - *DQ Planning*.

3.1. Análisis

El proceso de análisis para la herramienta que se propone en este proyecto de grado, se desarrolló en el contexto de un proyecto colaborativo donde se implementan las dos primeras fases de la metodología propuesta en la tesis de Doctorado de Serra [1]. Este enfoque presenta características particulares, ya que dos equipos de desarrollo trabajan de manera coordinada pero independiente: este trabajo se enfoca en la implementación de la Fase 1 - *DQ Planning* de la metodología, y otro proyecto de grado desarrolla la Fase 2 - *DQ Assessment*, estableciendo una comunicación exclusivamente a través de una base de datos común.

3.1.1. Requerimientos

Los requerimientos funcionales de la herramienta propuesta, abarcan las capacidades esenciales necesarias para implementar la Fase 1 - *DQ Planning* de la metodología *CaDQM*. La herramienta debe permitir la creación y administración de múltiples proyectos de manera simultánea, proporcionando a los usuarios la capacidad de trabajar con distintos conjuntos de datos y contextos.

Uno de los requerimientos de la herramienta es la capacidad de generar resultados en la Fase 1 - *DQ Planning*, que puedan ser consumidos directamente por la Fase 2 - *DQ Assessment*, desarrollada por el equipo 2. Esta integración se realiza exclusivamente a través de la base de datos común, donde el equipo 2 consume directamente los componentes de contexto y problemas de calidad de datos identificados mediante la aplicación de la herramienta desarrollada por el equipo 1.

Adicionalmente, la herramienta debe implementar capacidades de análisis de datos automatizado, utilizando herramientas para *data profiling*. Este requerimiento incluye la integración de múltiples herramientas de análisis para proporcionar perspectivas desde diferentes enfoques metodológicos, así como también la generación de reportes visuales que faciliten la interpretación de resultados.

Por otro lado, durante el proceso de desarrollo, se identificó la necesidad de incorporar herramientas de inteligencia artificial, para asistir a los usuarios en la definición y refinamiento de componentes de contexto e identificación de problemas de calidad de datos. Estas funcionalidades utilizan modelos de lenguaje de código abierto, para analizar documentación y datos, sugiriendo componentes de contexto y problemas de calidad de datos. En todo momento, es el usuario quien finalmente decide la persistencia de los componentes de contexto y de los problemas de CD sugeridos. Esta asistencia inteligente representa un valor agregado significativo, ya que facilita la aplicación de la metodología *CaDQM*.

Al finalizar cada etapa de la Fase 1 - *DQ Planning*, la herramienta debe generar reportes con los resultados obtenidos, las decisiones tomadas, y el estado actual del proyecto. Estos reportes proporcionan una visualización de la trazabilidad completa del proceso de GCD.

Un requerimiento específico del proyecto establece que ambos equipos de desarrollo utilicen los mismos conjuntos de datos para las pruebas. Esta consistencia de datos de prueba garantiza que los resultados sean comparables entre fases y que la integración entre equipos pueda ser validada de manera efectiva.

A modo de resumen, se listan los requerimientos antes descriptos:

- Crear múltiples proyectos en simultáneo, permitiendo trabajar con distintos contextos y conjuntos de datos.
- Los resultados generados en la Fase 1 - *DQ Planning* deben poder ser consumidos directamente por la Fase 2 - *DQ Assessment*, mediante una base de datos común compartida entre los equipos de proyectos de grado.
- Integración de herramientas de *data profiling* para análisis automatizado y generación de reportes.
- Sugerencias de componentes de contexto y problemas de calidad de datos, utilizando herramientas de inteligencia artificial.
- Generación de reportes con resultados por etapa.
- Utilización del mismo conjunto de datos de prueba entre ambos equipos.

3.2. Diseño

Esta sección detalla las decisiones de diseño más importantes que guiaron el desarrollo de la herramienta. Se presenta el diseño de la base de datos, utilizada no solo para almacenar los datos, sino también para la comunicación entre las fases 1 y 2 de la metodología *CaDQM*. Adicionalmente, se detalla el diseño de la arquitectura del software, definiendo los diferentes elementos y como interactúan entre ellos.

3.2.1. Diseño de la base de datos

Un aspecto fundamental del proyecto es el desarrollo de una base de datos (BD) común a dos proyectos de grado. Esta BD permite la integración entre nuestra herramienta, enfocada en las primeras tres etapas de la Fase 1 - *DQ Planning* de *CaDQM*, y la herramienta desarrollada por el otro proyecto, responsable de la Fase 2 - *DQ Assessment* de la metodología. Este esfuerzo de coordinación requirió un diseño cuidadoso de la arquitectura de base de datos y un proceso iterativo e incremental de refinamiento para asegurar la compatibilidad entre ambas herramientas.

Para el diseño de la BD común , fue necesario establecer un modelo de datos común que pudiera servir tanto para el registro y gestión de los componentes de contexto, problemas de CD y requerimientos identificados en la Fase 1 - *DQ Planning*, como para las actividades de definición de modelos de calidad, medición y evaluación de la CD, propias de la Fase 2 - *DQ Assessment*. Este proceso implicó sesiones de trabajo conjunto donde ambos proyectos de grado analizaron los requerimientos específicos de sus respectivas implementaciones e identificaron los puntos de convergencia y divergencia en los modelos de datos individuales.

En la Figura 3.1, se muestra el modelo conceptual de la base de datos. En este modelo, se representan las entidades con sus atributos y relaciones. En amarillo (del lado derecho de la imagen), se destacan las entidades utilizadas en la fase 1, en rojo (del lado izquierdo de la imagen) las correspondientes a la fase 2, y en blanco se representan todas las entidades necesarias, para brindar información sobre el proceso de aplicación de la metodología. Estas últimas no son específicas de ninguna de las fases de *CaDQM*.

A continuación, se presentan las entidades generales de la aplicación:

- *user*, se utiliza para la autenticación de la aplicación, permitiendo a los usuarios registrarse y gestionar sus proyectos.
- *project*, representa una ejecución completa de la metodología *CaDQM*. El proyecto actúa como nexo entre los distintos datos que se van recolectando y generando a lo largo del proceso de gestión de calidad de datos. Un usuario puede crear y administrar simultáneamente varios proyectos, cada uno con su propio conjunto de datos, modelo de contexto, problemas de calidad de datos, etc.
- *stage*, representa a las etapas de CaDQM. Cada proyecto se asocia a varias etapas.
- *project_stage*, representa la asociación entre etapas y proyectos. Además, permite registrar cuándo y en qué etapa de *CaDQM* se detectó cada problema de CD.
- *status*, representa el estado de ejecución de cada etapa dentro de un proyecto (por ejemplo: to do, in progress, done).
- *data_at_hand*, representa el conjunto de datos seleccionado, cuya calidad va a ser evaluada.

Por otro lado, se representan las entidades necesarias para ejecutar la Fase 1 - *DQ Planning*. Estas se presentan a continuación:

- *context*, representa el modelo de contexto, que además de un identificador y un nombre, tiene una versión. Cada proyecto debe tener asociado un solo modelo de contexto. Si se vuelve a iterar en la metodología sobre el mismo conjunto de datos, se crea un proyecto nuevo, con un modelo de contexto nuevo, pero que se vincula a la versión anterior mediante el atributo *previous_version*.
- *context_component*, esta entidad tiene una relación N a N con la entidad que representa al contexto, lo que significa que un contexto puede tener varios componentes y un componte puede estar asociado a varias versiones del contexto. Cada tipo de componente de contexto tiene sus propios atributos y relaciones con otras entidades, sin embargo todos comparten la relación con el modelo de contexto.

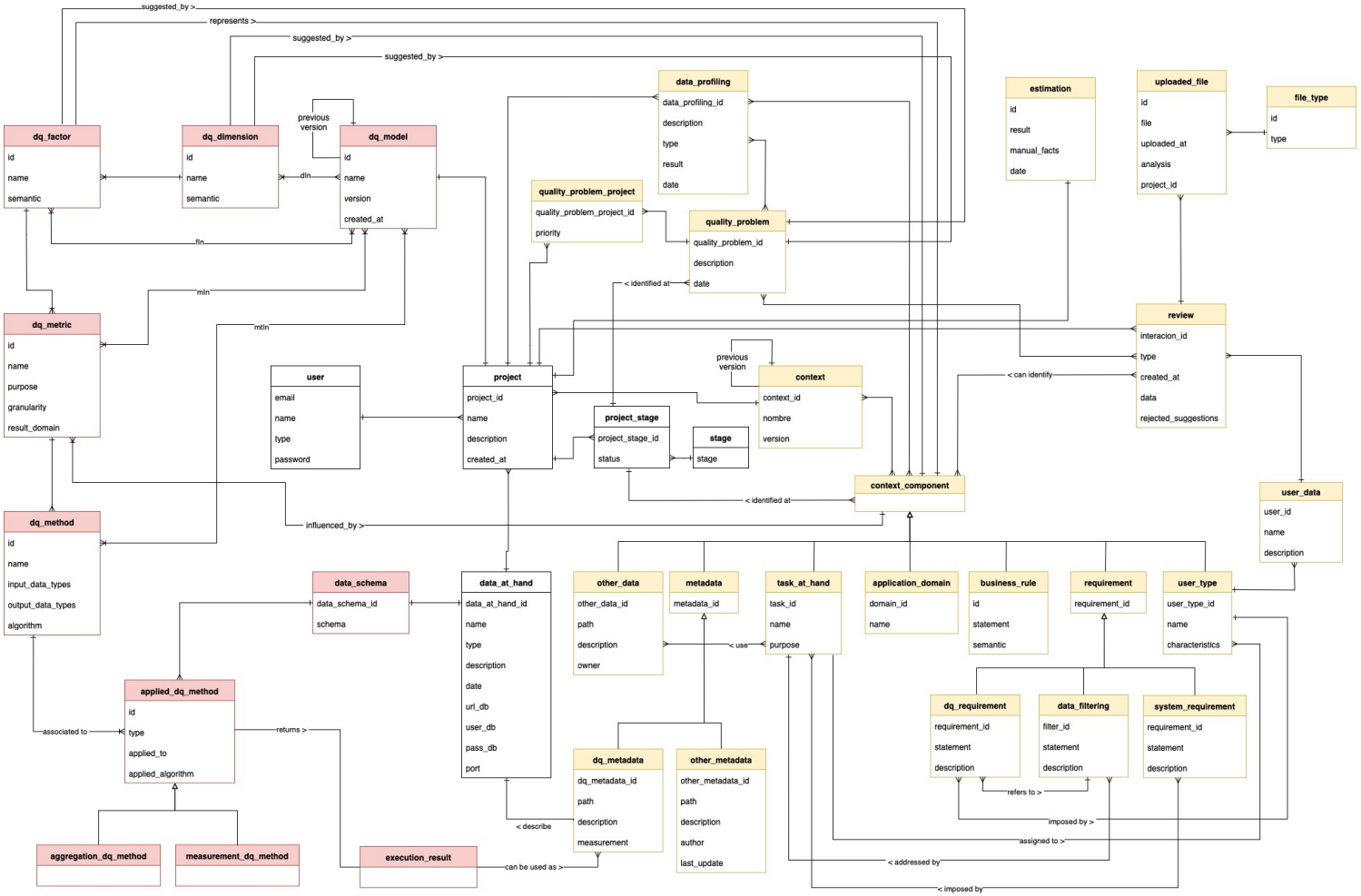

Figura 3.1: Modelo Conceptual de la Base de Datos

Los subtipos correspondientes a las componentes de contexto son disjuntos y completos, lo que implica que un componente de contexto debe ser de uno y solo uno de estos tipos. Los subtipos identificados son los que se presentan a continuación:

- *application_domain*, dominio de aplicación de los *data at hand*.
- *business_rule*, reglas de negocio definidas sobre el esquema de los *data at hand*.
- *metadata*, representa los metadatos en general. Existen dos tipos, que se representan con subtipos de esta entidad: *dq_metadata*, que corresponde a los metadatos de calidad y *other_metadata*, que son otros metadatos.
- *other_data*, otros datos que dan contexto a los *data at hand*.
- *requirement*, representa los requerimientos en general. Estos deben ser de alguno de estos subtipos: requerimientos de calidad, representados con *dq_requirement*, requerimientos de filtrado de datos, representados con *data_filtering* y requerimientos del sistema, representados con *system_requirement*.
- *task_at_hand*, tareas que realizan los usuarios de los *data at hand*. Estas pueden estar asignadas a distintos tipos de usuario.
- *user_type*, distintos tipos de usuarios que trabajan con los *data at hand*.

Cada subtipo de los componentes de contexto añade atributos y relaciones específicas, pero todos ellos heredan la vinculación con el modelo de contexto, por lo tanto, también con el proyecto.

Además, se presentan las siguientes entidades:

- *quality_problem*, son los problemas de CD identificados durante la ejecución de la metodología y tienen una prioridad asignada dentro de un proyecto.
- *review*, esta entidad es necesaria en la Etapa 1 - *Elicitation*, donde se guardan los datos de la organización en forma de texto o archivos adjuntos. Además, en la Etapa 3 - *User Requirement Analysis*, en la que se interactúa con los usuarios de los *data at hand*, también surge la necesidad de guardar datos en el mismo formato. La entidad *review*, permite representar datos de la organización o datos obtenidos de la interacción con los usuarios de los *data at hand*.
- *data profiling*, esta entidad es necesaria para registrar los resultados del análisis realizado en la Etapa 2 - *Data Analysis*. Esta entidad permite guardar múltiples resultados de distintos tipos.
- *estimation*, entidad que permite almacenar las estimaciones realizadas por el usuario a partir del análisis de los datos.

Por otro lado, las entidades en rojo, son las entidades necesarias para la ejecución de la Fase 2 - *DQ Assessment*. Estas fueron definidas teniendo en cuenta los requerimientos del trabajo de Serra [1] y los requerimientos especificados en el trabajo realizado por el proyecto de grado encargado de la Fase 2 de *CaDQM*. Esta entidades no se generan ni modifican en la Fase 1 - *DQ Planning*, sin embargo, pueden ser consumidas por ésta. Por ejemplo, cuando los resultados de ejecución, recopilados en la base de datos *execution_result*, se utilizan como metadatos de calidad en *dq_metadata*.

Finalmente, este diseño permite registrar todos los datos necesarios en la Fase 1 - *DQ Planning*, asegurando la documentación completa del proceso de GCD, el almacenamiento de toda la información recolectada y la trazabilidad de los datos, guardando cada una de las etapas donde fueron obtenidos. Además, asegura la compatibilidad con el trabajo del otro proyecto de grado, cumpliendo el requisito de tener una base de datos común e intercambiable entre las fase 1 y 2 de la metodología.

3.2.2. Arquitectura de la herramienta

Para cumplir con los requerimientos de la herramienta, se optó por una arquitectura distribuida en capas, que se muestra en la Figura 3.2. Este diseño permite separar las responsabilidades de cada capa, para que la herramienta sea escalable y fácil de mantener.

La arquitectura se compone de un *frontend*, que es una aplicación web que actúa como la capa de presentación e interacción de usuario final. Desde esta interfaz, el usuario puede gestionar las distintas etapas de la Fase 1 - *DQ Planning* de la metodología *CaDQM*, definiendo el proyecto, el modelo de contexto, visualizando los resultados de los procesos de *data profiling*, identificando problemas de calidad de datos, etc.

Por otro lado, el *backend* se implementa como una API REST, encargada de gestionar la lógica de negocio. Esta API recibe las solicitudes del cliente y gestiona el acceso a los datos, la ejecución de los procesos de *data profiling*, y las llamadas a la API de las herramientas externas de IA. Además, esta capa gestiona la conexión a la base de datos donde se encuentran los datos cuya calidad se va a evaluar.

Por último, para el almacenamiento de la información de la ejecución de la Fase 1 - *DQ Planning* de la metodología *CaDQM*, se utiliza una base de datos relacional, implementada a partir del diseño definido en la sección anterior. En esta se registran los proyectos, componentes de contexto, problemas de CD, resultados de análisis, etc.

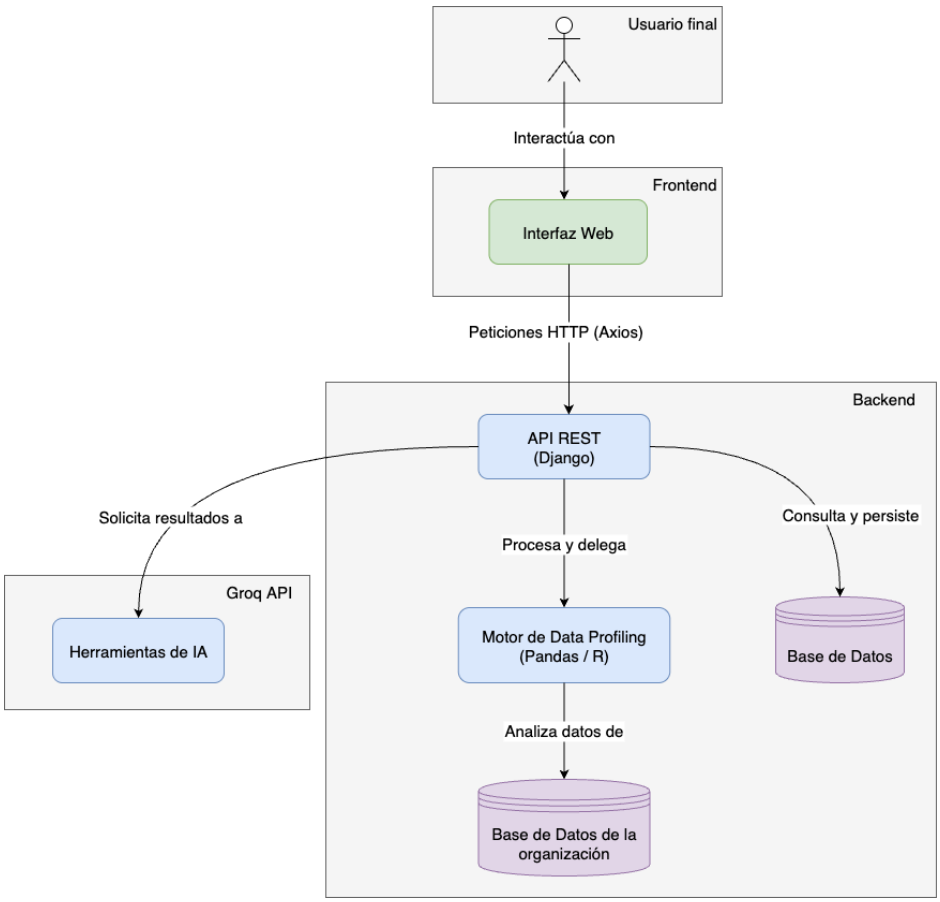

Figura 3.2: Diagrama de la arquitectura general de la herramienta.

Capítulo 4

Implementación

En este capítulo se describen todas las decisiones tomadas con respecto a la implementación de la herramienta. En primer lugar, se detallan todos los aspectos relacionados a la aplicación web desarrollada. Luego, se describe el funcionamiento de la API, su integración con herramientas externas de *data profiling* y APIs externas de inteligencia artificial. Por último, se detalla el alcance y las limitaciones que surgieron al desarrollarla.

4.1. Aplicación web

Para el desarrollo del *frontend* de la herramienta se optó por utilizar la librería *React* [15], utilizando el lenguaje *TypeScript* [16]. La elección de este lenguaje se debe a la experiencia previa en proyectos anteriores con *JavaScript*, lenguaje sobre el que se basa *TypeScript*. *React* permite construir interfaces de usuario dinámicas y reutilizables a través de componentes de software, lo que facilita el desarrollo, mantenimiento y escalabilidad del código. Además, se optó por *TypeScript* porque es un lenguaje tipado, que le brinda seguridad y robustez al código, ya que permite detectar errores de tipo durante la compilación, reduciendo los errores en tiempo de ejecución [17].

La arquitectura del *frontend* se diseñó siguiendo principios de modularidad y separación de responsabilidades para facilitar el mantenimiento y la escalabilidad. La aplicación se basa en una estructura de componentes de software reutilizables que encapsulan funcionalidades específicas.

La navegación entre las distintas vistas se implementó mediante rutas, mientras que la comunicación con el backend se centralizó en una capa de servicio que gestiona las peticiones http, utilizando la librería *Axios* [18]. Esta última permite crear un cliente http que se comunica con el *backend*. Para manejo del estado, se utilizan los *hooks* nativos de *React* [1]. El estilo de la interfaz de usuario de la aplicación se construyó utilizando componentes de software prediseñados, proporcionados por la librería *Material UI* [20], lo que también facilita el desarrollo.

Dado que en este proceso es fundamental que el usuario pueda guardar reportes de resultados obtenidos en cada etapa, se integró la librería *React PDF* [21], que permite crear archivos PDF con toda la información recolectada en el proceso.

El diseño de la interfaz se basó en un diagrama de flujo que modela la experiencia del usuario, presentado en las figuras 4.1 y 4.2: la primera muestra el flujo para crear un proyecto y ejecutar la Etapa 1 - *Elicitation*, mientras que la segunda detalla los

[1] Los *hooks* son funciones especiales que permiten a los componentes funcionales acceder al estado y a otras características de *React*, como el ciclo de vida, sin necesidad de escribir componentes de clase [19].

flujos de las etapas 2 - *Data Analysis* y 3 - *User Requirements Analysis*. Este diagrama sirve para mostrar la secuencia de pasos que un usuario final debe seguir, para realizar una tarea al interactuar con la aplicación. Además, está centrado en las necesidades del usuario y en cuál es la forma más eficiente de abordarlas [22]. Este diagrama fue fundamental para definir la estructura de navegación y la interacción, asegurando un recorrido lógico y coherente.

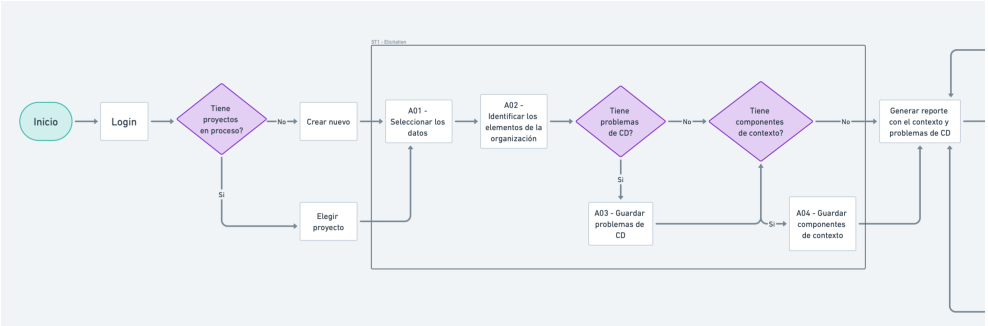

Figura 4.1: Diagrama de flujo de la interfaz de usuario, primera parte.

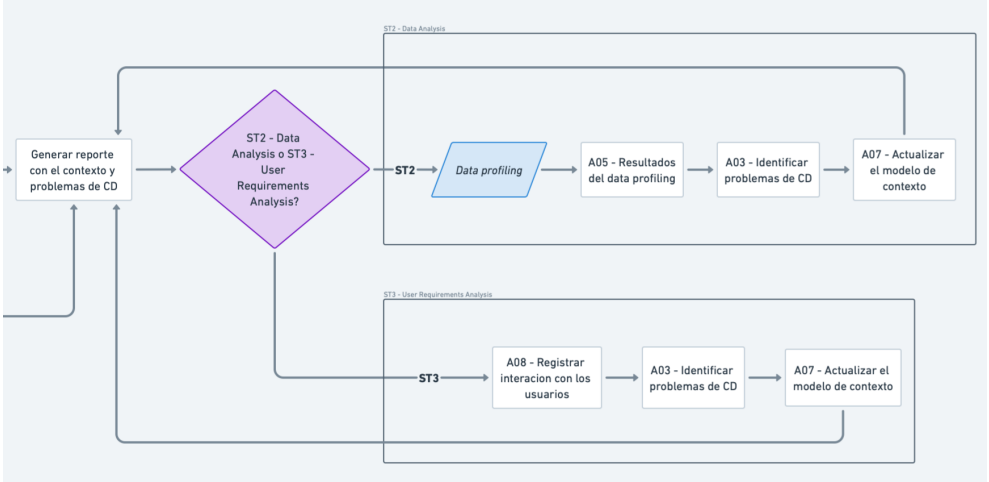

Figura 4.2: Diagrama de flujo de la interfaz de usuario, segunda parte.

Una vez definido el flujo de interacción, se crearon las páginas, los componentes de software genéricos y los *endpoints* de la API necesarios para cada funcionalidad. Como en esta metodología hay varios pasos que se pueden repetir, fue fundamental crear componentes de software que encapsulen las funcionalidades y puedan ser reutilizados en todo el código de manera sencilla. A modo de ejemplo, se mostrarán algunas de las pantallas clave de la aplicación web. El flujo de usuario completo y detallado se encuentra en el manual de usuario en el Anexo A.

El acceso a la herramienta requiere que los usuarios estén autenticados. Para esto, se desarrolló una vista de *login* que se comunica con un *endpoint* de la API para validar las credenciales. Si la autenticación es exitosa, la API retorna un *token*. Este *token* se almacena en el cliente y se envía en la cabecera de todas las solicitudes http posteriores mediante un interceptor de *Axios*, garantizando así el acceso seguro a los recursos del proyecto y a los datos de los usuarios.

Una vez autenticado, el usuario accede a la lista de proyectos, como se muestra en la Figura 4.3. Esta vista consume datos del *endpoint* de proyectos y los presenta en

una lista, indicando el estado de cada etapa (*to do*, *in progress*, *done*). Desde aquí, el usuario puede crear un nuevo proyecto, lo que inicia un nuevo flujo de la metodología desde la Etapa 1 - *Elicitation*.

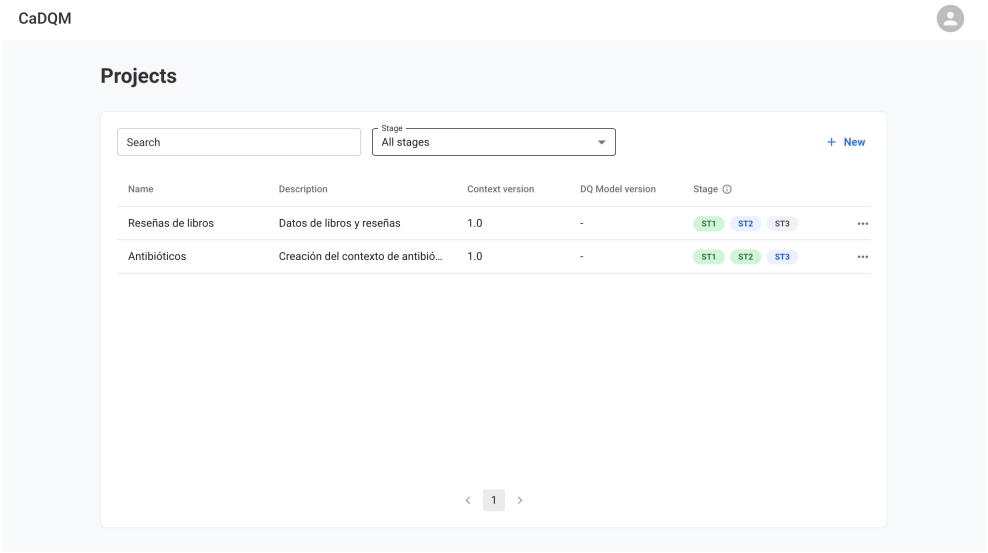

Figura 4.3: Pantalla de lista de proyectos.

Al seleccionar un proyecto, se muestra, entre otros datos, el progreso general en las Fase 1 - *DQ Planning* y Fase 2 - *DQ Assessment* de la metodología *CaDQM*. El usuario puede seleccionar una etapa para comenzar o reanudar la ejecución de la fase 1.

La ejecución las etapas de la Fase 1 - *DQ Planning*, se implementó con una interfaz tipo *wizard* [2], con el objetivo de guiar al usuario de la herramienta a través de las actividades, de forma secuencial. Este diseño se definió junto con el equipo que implementó la Fase 2 - *DQ Assessment*, donde las actividades se ejecutan de la misma forma. Esto es así para mantener la coherencia en la experiencia del usuario, a lo largo de las 2 fases de la metodología. Para crear este *wizard*, se desarrolló un componente de software genérico, que contiene otros componentes de software reutilizables encargados de mostrar el progreso dentro de cada etapa permitiendo, además, la navegación entre actividades. Antes de permitir que el usuario avance a la siguiente actividad dentro de una misma etapa, se ejecutan validaciones para asegurar que la información requerida haya sido completada correctamente. A continuación, se muestra un ejemplo con las actividades de la Etapa 1 - *Elicitation*.

La Figura 4.4 muestra la interfaz para la primera actividad de la etapa 1, que corresponde a la selección de los datos. En este caso, se muestra un formulario que recolecta los datos necesarios para establecer la conexión a la base de datos, donde se encuentra el *data at hand*.

[2]Un *wizard* es un proceso paso a paso que permite a los usuarios ingresar información en un orden prescrito, en el que los pasos posteriores pueden depender de la información introducida en los anteriores [23].

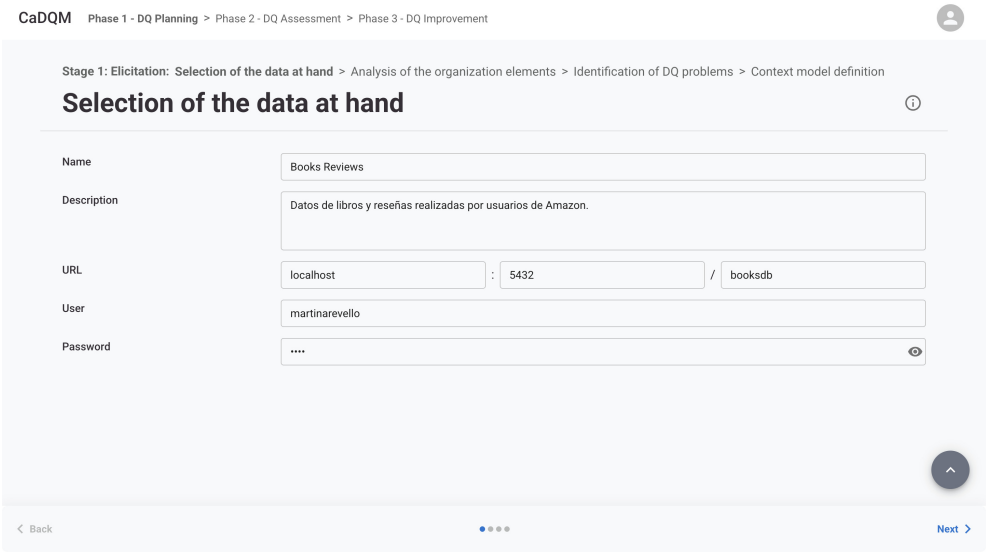

Figura 4.4: Actividad 1 - Selección de los datos.

En la segunda actividad, se analizan los elementos de la organización (ver Figura 4.5). En este caso, la herramienta ofrece dos formas de cargar la información: i) un campo de texto donde se puede escribir información relevante para los *data at hand*, ii) el usuario puede subir archivos que contienen información que le da contexto a los *data at hand*. Para organizar los archivos recopilados, a cada archivo se le puede asignar un tipo, lo que permite que se muestren agrupados por categoría.

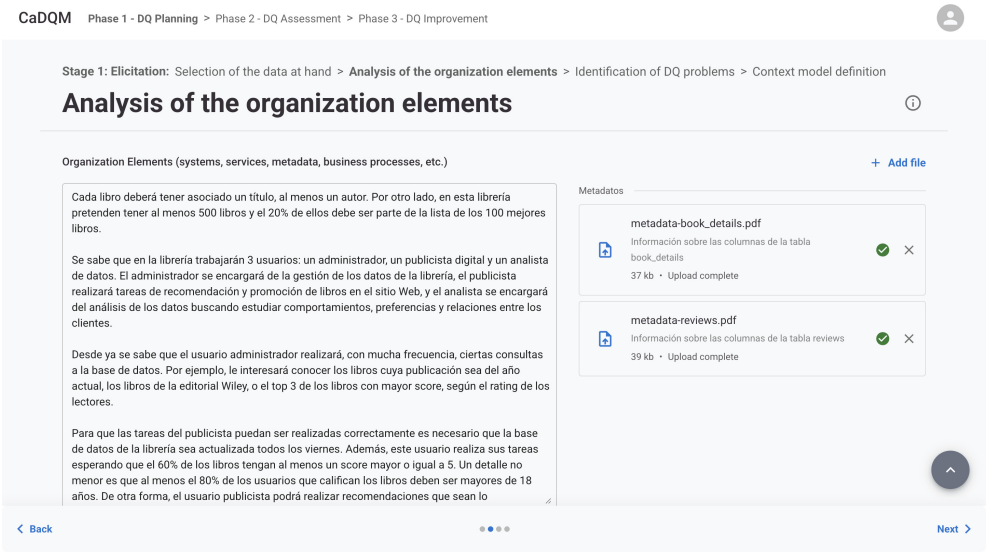

Figura 4.5: Actividad 2 - Análisis de los elementos de la organización.

La tercera actividad es la identificación de problemas de calidad de datos (ver Figura 4.6). En este punto, el usuario puede registrar los problemas de CD manualmente, a través de un formulario que se presenta en un diálogo, o solicitar posibles problemas de CD sugeridos por el asistente de IA. Las sugerencias generadas por la herramienta se muestran en una lista, y surgen de toda la información sobre la organización, recopilada hasta el momento.

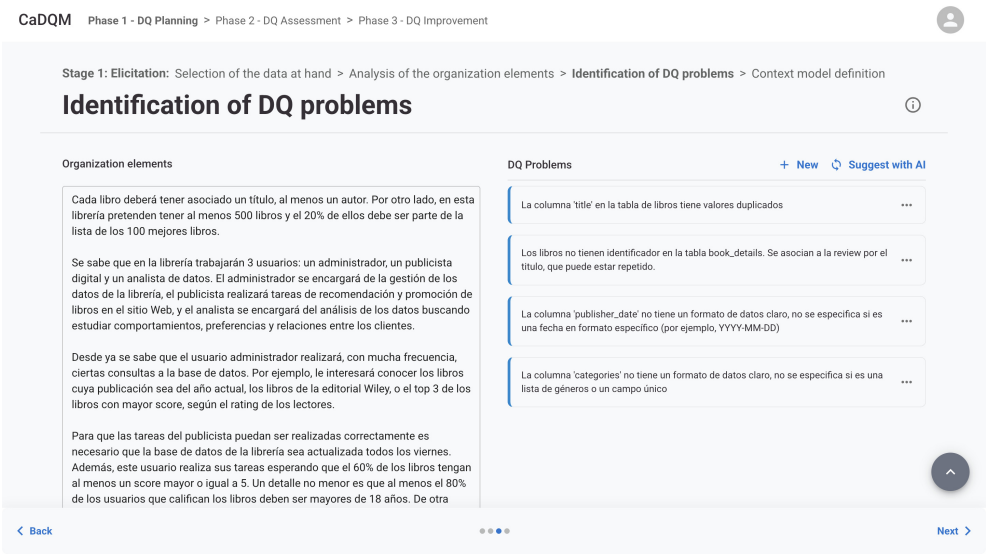

Figura 4.6: Actividad 3 - Identificación de los problemas de calidad.

Finalmente, la última actividad de esta etapa consiste en la definición del modelo de contexto (ver Figura 4.7). En ella, el usuario define distintos tipos de componentes de contexto, tal como fueron introducidos en el marco teórico 2.2.1. De forma similar a la actividad anterior, los componentes de contexto pueden ser creados manualmente o generados a partir de las sugerencias de la herramienta, mediante el uso del asistente de IA. Las sugerencias se elaboran en base a toda la información sobre la organización, previamente registrada.

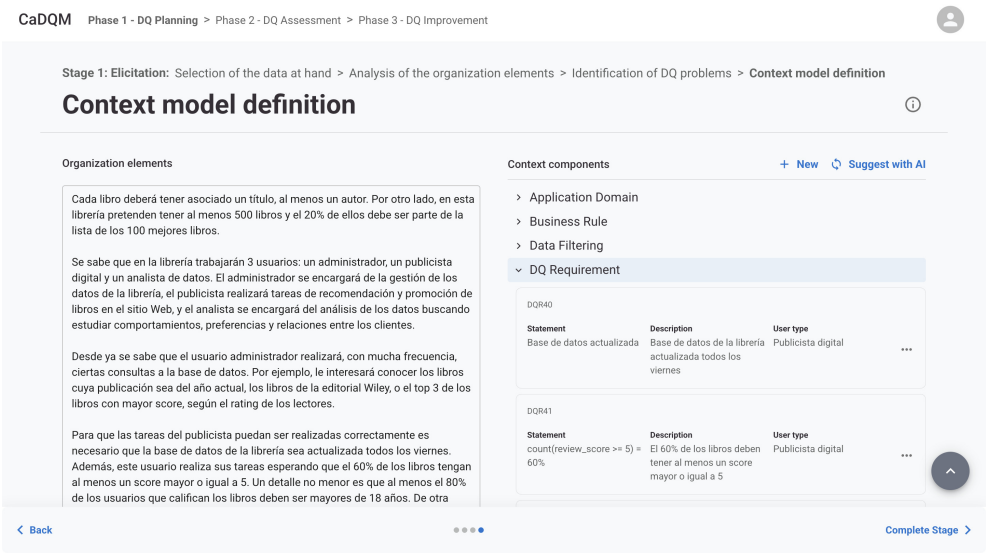

Figura 4.7: Actividad 4 - Definición del modelo de contexto.

Tener en cuenta que todas las sugerencias del asistente de IA, para cualquiera de las actividades, pueden ser aceptadas, modificadas o descartadas. Al finalizar una etapa, la aplicación ofrece la opción de continuar con la siguiente etapa o volver a la lista de proyectos.

Adicionalmente, se incluye la funcionalidad de generar reportes en formato PDF con todos los datos recolectados. El usuario puede seleccionar, entre las etapas finalizadas, qué resultados desea incluir, y a partir de esta selección, la herramienta

construye el reporte agrupando la información según la actividad y la etapa en la que fue registrada.

La generación de reportes se implementó mediante componentes de software reutilizables de *React*. Dado que distintas actividades producen salidas similares, como una lista de problemas de CD o un conjunto de componentes de contexto, se diseñaron componentes de software específicos para representar estos datos. Esto permitió reutilizar código, reducir duplicación y mantener una separación de responsabilidades. En el manual de usuario que se encuentra en el Anexo A, se describe más en detalle la creación de los reportes de las etapas.

4.2. Backend

Para el desarrollo del *backend* se utilizó *Django* [24], un framework de *Python* [25] que ofrece una arquitectura robusta para el desarrollo de aplicaciones web. Este framework permite una rápida implementación de APIs mediante el uso de *Django REST Framework*, lo que facilita la gestión de solicitudes HTTP y el manejo de datos en la base de datos.

Además, se optó por *PostgreSQL* [26] como sistema de gestión de base de datos debido a su estabilidad, escalabilidad y compatibilidad nativa con *Django*. Esta elección permite manejar grandes volúmenes de datos y garantizar la integridad de la información mediante transacciones ACID y restricciones de integridad referencial, aspectos fundamentales para un sistema de gestión de calidad de datos. La conexión con PostgreSQL se estableció mediante psycopg2 [27], una librería que actúa como adaptador entre Django y PostgreSQL, permitiendo la ejecución de consultas y el manejo de conexiones a la base de datos. La gestión de la estructura de la misma se realizó a través del sistema de migraciones de Django, que proporciona un control de versiones automático para el esquema de la base de datos, facilitando la creación, modificación y gestión de las estructuras sin necesidad de alterar manualmente la base de datos.

La arquitectura del *backend* sigue el patrón *Model-View-Template* (MVT) [28] de *Django*, adaptado específicamente para servicios *REST* mediante *Django REST Framework*. Esta arquitectura se estructura en capas bien definidas que facilitan el mantenimiento, la escalabilidad y la extensibilidad de la herramienta. La capa de modelos implementa el modelo conceptual de la metodología *CaDQM*, definiendo entidades y sus relaciones correspondientes. Estos modelos almacenan información y encapsulan la lógica de negocio específica de cada etapa de la metodología, asegurando la coherencia de los datos y las reglas de negocio establecidas por *CaDQM*.

La implementación del *backend* se apoyó en diversas librerías especializadas que facilitan el desarrollo y mejoran la funcionalidad de la herramienta. *Django REST Framework* proporciona herramientas esenciales para la creación de *APIs REST*, permitiendo la serialización de datos, la gestión de permisos y la implementación de sistemas de autenticación. Para la autenticación específicamente, se implementó *Simple JWT* [29], una solución basada en *JSON Web Tokens* [30] [3] que permite una autenticación segura y *stateless*.

La herramienta maneja múltiples fuentes de datos de manera unificada, soportando conexiones a bases de datos relacionales como PostgreSQL [31], MySQL [32] y SQLite [33]. Esto podría representar una limitante inicialmente, y la idea es que pueda escalar

[3] Los JSON Web Tokens (JWT) son un estándar abierto (RFC 7519) que define una forma compacta y autocontenida de transmitir información de manera segura entre partes como un objeto JSON. Esta información puede verificarse y considerarse confiable porque está firmada digitalmente [30].

para soportar otras fuentes de datos como bases de datos no relacionales u otros tipos de archivos.

Para facilitar el despliegue y la configuración del sistema, se implementó una solución basada en *Docker* [34], que simplifica considerablemente el proceso de instalación y puesta en marcha del *backend*. Esta implementación permite levantar el proyecto con comandos simples, evitando la necesidad de instalar manualmente todas las dependencias necesarias, particularmente las relacionadas con R [35] y las librerías asociadas (usado en el *data profiling*, actividad de la Etapa 2 - *Data Analysis* de *CaDQM*). Sin *Docker*, el proceso de configuración requeriría la instalación manual de R, sus paquetes específicos para análisis de datos, y la configuración correcta de *rpy2* [36] para la comunicación entre *Python* y R, proceso que puede resultar tedioso y propenso a errores de compatibilidad entre diferentes sistemas operativos. El uso de contenedores con *Docker*, encapsula todas estas dependencias en un entorno controlado, garantizando que el sistema funcione de manera consistente independientemente del entorno de desarrollo [37].

4.2.1. Etapa 2 - *Data Analysis: Data profiling*

El *Data Profiling*, también conocido como perfilado de datos, es una técnica fundamental en el campo de la gestión de calidad de datos que consiste en el análisis sistemático y automatizado de conjuntos de datos para comprender su estructura, contenido, calidad y características estadísticas [38]. Esta práctica permite identificar patrones, anomalías, inconsistencias y problemas potenciales en los datos antes de su procesamiento o análisis, proporcionando una visión integral del estado de la información disponible.

Durante el proceso de selección de herramientas de *Data Profiling*, se evaluaron múltiples alternativas disponibles en el mercado. Se analizaron herramientas como *Talend Data Quality* [14], que fue descartada por ser una solución comercial cuando se priorizaban alternativas gratuitas. *DataCleaner* [39], aunque es una herramienta de código abierto con capacidades robustas, presentó limitaciones significativas al estar orientada fundamentalmente hacia el uso mediante interfaz gráfica. Se realizaron intentos de ejecutar *DataCleaner* a través de su archivo JAR con parámetros de línea de comandos, pero no fue posible lograr la integración automatizada requerida. Finalmente, se evaluó *Metanome* [40], un framework académico especializado en descubrimiento de metadatos, pero su proceso de compilación complejo y la falta de mantenimiento activo del proyecto lo hicieron inviable para su implementación.

En el marco de este proyecto, se implementaron finalmente dos herramientas complementarias de *Data Profiling*, que ofrecen diferentes enfoques y capacidades de análisis: *ydata-profiling (Python)* y *DataExplorer (R)*. La selección de estas herramientas responde a la necesidad de proporcionar un análisis completo desde diferentes perspectivas metodológicas y técnicas.

La primera herramienta, *ydata-profiling* (antes *pandas-profiling*), es una biblioteca de *Python* para generar automáticamente reportes de análisis exploratorio de datos. Esta herramienta crea análisis detallados y visualizaciones interactivas con mínima configuración por parte del usuario, e integra de forma directa con pandas y NumPy [41–43].

Como complemento a *ydata-profiling*, se integró *DataExplorer*, una biblioteca de R diseñada específicamente para el análisis exploratorio automatizado de datos. Esta herramienta se destaca por su capacidad de generar reportes HTML completos con visualizaciones estadísticas avanzadas. *DataExplorer* se integra directamente con *PostgreSQL* mediante el paquete RPostgres [44] y la interfaz estándar DBI [45], permitiendo establecer conexiones directas con bases de datos y especificar parámetros

como *host*, puerto, nombre de base de datos, usuario y contraseña.

Una característica técnica fundamental de esta implementación es la ejecución de código R dentro del entorno *Python* de *Django*. Esto se realiza con la biblioteca *rpy2*, que actúa como un puente entre *Python* y R, permitiendo ejecutar código R desde aplicaciones *Python*. Rpy2 es una interfaz de bajo nivel que que expone el intérprete de R desde *Python* y permite ejecutar *scripts*, intercambiar datos y acceder a paquetes de R, desde la aplicación *Python* [36].

4.2.2. Base de Datos

La implementación de la BD común presentó desafíos significativos relacionados con la sincronización de cambios en los modelos de datos. Cada modificación en la estructura de la BD debía ser evaluada en términos de su impacto en ambos sistemas, lo que llevó a un proceso de varias iteraciones entre los dos proyectos de grado. Por ejemplo, cuando se identificaba la necesidad de agregar un nuevo campo a una entidad existente, o modificar una relación entre modelos, era necesario coordinar estos cambios para asegurar que no afectaran la funcionalidad ya implementada en la herramienta del otro proyecto de grado.

La funcionalidad de la BD común se validó mediante pruebas de integración donde nuestra herramienta ejecutaba las primeras etapas de *CaDQM*, generando un modelo de contexto y problemas de calidad de datos identificados. Posteriormente, estos datos eran transferidos a la herramienta del otro proyecto de grado, que podía continuar con las actividades de la Fase 2 - *DQ Assessment*, sin pérdida de información ni inconsistencias. Este proceso de validación permitió identificar y resolver incompatibilidades en tiempo real, asegurando que la transición entre fases fuera fluida y que los datos generados en la Fase 1 - *DQ Planning* fueran completamente utilizables en la Fase 2 - *DQ Assessment*.

4.3. Integración de IA

La integración de inteligencia artificial (IA) en la herramienta de GCD que considera el contexto de los datos surgió como un requerimiento adicional durante el desarrollo del proyecto. Esta funcionalidad fue concebida para proporcionar asistencia inteligente y sugerencias automatizadas que mejoren la experiencia del usuario y optimicen los procesos de GCD.

Es fundamental destacar que la integración de IA en esta herramienta es para dar apoyo al usuario, manteniendo siempre el principio de que el usuario tiene la última palabra en todas las decisiones relacionadas con la CD. La IA actúa como un asistente que proporciona recomendaciones basadas en el análisis de los datos y archivos cargados, pero el usuario conserva el control total para aceptar, modificar o descartar estas sugerencias según su criterio experto y conocimiento del dominio específico.

Esta aproximación híbrida humano-máquina busca combinar la capacidad de procesamiento y análisis automatizado de la IA con la experiencia, intuición y conocimiento contextual del usuario, creando un sistema colaborativo que potencia las capacidades de la herramienta propuesta.

4.3.1. Herramientas analizadas

Durante el proceso de selección de herramientas de IA, se evaluaron múltiples alternativas disponibles en el mercado, considerando factores técnicos, económicos y de integración con la arquitectura existente.

Inicialmente se consideraron modelos comerciales como GPT de *OpenAI* [46] y *DeepSeek* [47], los cuales ofrecían capacidades avanzadas de procesamiento de lenguaje natural y análisis de datos. Sin embargo, estas opciones fueron descartadas debido a sus costos asociados, priorizando soluciones gratuitas. Tras esta evaluación inicial, se optó por utilizar la plataforma *Groq* [48] como proveedor de servicios de *IA*, específicamente empleando el modelo *llama-3.3-70b-versatile* [49]. Esta selección se basó en varios factores clave:

- La disponibilidad de un plan gratuito (*free tier*) [4] que permite realizar pruebas y desarrollo sin costos iniciales.
- La alta velocidad de inferencia que proporciona Groq.
- La calidad del modelo Llama3 [49] para tareas de análisis de texto y generación de sugerencias contextuales.

El modelo *llama-3.3-70b-versatile* se caracteriza por su arquitectura optimizada con técnicas de *speculative decoding* que permiten una velocidad de inferencia de hasta 1 600 *tokens* por segundo, lo que resulta fundamental para mantener una experiencia de usuario fluida en el sistema de GCD. Su ventana de contexto de 8 192 *tokens* es adecuada para analizar archivos de datos y documentación técnica de tamaño moderado, mientras que su arquitectura de 70 mil millones de parámetros proporciona respuestas coherentes y de alta calidad [51].

Cabe destacar que durante la mayor parte del desarrollo se empleó el modelo *llama3-8b-8192*. La finalización del soporte de este modelo en agosto de 2024 motivó la necesidad de migrar a una alternativa compatible. Este proceso requirió una evaluación para seleccionar un modelo sustituto que cumpliera con los requerimientos técnicos y funcionales de la herramienta desarrollada.

4.3.2. Evaluaciones

La evaluación del modelo *llama-3.3-70b-versatile* se realizó mediante un proceso iterativo que abarcó múltiples funcionalidades del sistema, cada una con requerimientos específicos de procesamiento de lenguaje natural. Esta evaluación se centró en tres áreas principales:

- Análisis de archivos para identificación de problemas de CD.
- Identificación de componentes de contexto.
- Generación de estimaciones automatizadas a partir de resultados de *data profiling*.

El desarrollo de *prompts* para el análisis automatizado de archivos representó uno de los desafíos más significativos en la integración de IA. Inicialmente, los *prompts* se enfocaban en análisis genérico de texto, pero la necesidad de identificar problemas específicos de CD, llevó a una evolución considerable en su diseño.

La primera versión de los *prompts* se centraba en análisis exploratorio básico, pero rápidamente se identificó la necesidad de orientar el análisis hacia problemas concretos de CD. El *prompt* final implementado instruye al modelo a actuar como "un experto en calidad de datos" y establece claramente que el contenido puede incluir tanto documentación que describe cómo deberían estar estructurados los datos, como datos reales en formatos CSV o Excel.

[4] Groq ofrece un plan gratuito con límites de uso por organización (solicitudes y *tokens* por minuto/día), que varían según el modelo [50].

De forma progresiva se fue mejorando el *prompt*, a medida que la IA iba detectando problemas como valores nulos excesivos, duplicados, inconsistencias en formatos, problemas de integridad referencial, distribuciones anómalas, falta de identificadores únicos y valores fuera de rango. En el Anexo C se pueden ver los diferentes *prompts* utilizados para el análisis de las tres áreas principales anteriormente mencionadas.

4.4. Alcance y Limitaciones

En esta sección se describe, a modo de resumen, el alcance de la herramienta propuesta. Además, se listan las limitaciones actuales.

La herramienta implementa completamente las tres primeras etapas de la Fase 1 - *DQ Planning* de la metodología *CaDQM*. Esto implica la planificación inicial y elicitación de requerimientos, el análisis de problemas de CD, y la definición del contexto. La herramienta puede trabajar con diferentes tipos de bases de datos relacionales como *PostgreSQL* , *MySQL* y *SQLite* .

Una de las capacidades desarrolladas, más innovadoras, es la habilidad para analizar documentos como archivos .docx, .xlsx y .pdf, para extraer automáticamente problemas de CD y componentes de contexto. Esto se logró utilizando inteligencia artificial mediante la plataforma Groq [48] con el modelo llama-3.3-70b-versatile [49], que puede entender el contenido de documentos e identificar elementos relevantes para el análisis de CD.

Para el análisis, más específicamente el *data profiling*, se integraron dos herramientas muy potentes: ydata-profiling [52] para Python [25] y DataExplorer [53] para R [35]. Estas herramientas permiten hacer análisis profundos de los datos, generando reportes detallados sobre distribuciones, correlaciones, valores faltantes y anomalías.

Por otro lado, la herramienta presenta algunas limitaciones importantes que deben ser mencionadas:

- **Alcance de datos:** La herramienta está diseñada principalmente para datos relacionales. Aunque puede procesar algunos documentos mediante IA, pero no está optimizada para trabajar con grandes volúmenes de datos no estructurados como imágenes o videos.
- **Límite para *data profiling*:** Por restricciones de memoria y tiempo de procesamiento, se procesan como máximo 10 000 filas por tabla. Si el *data at hand* supera esa cantidad de filas, se toma una muestra aleatoria de exactamente 10 000 filas, en lugar de procesar todos los datos, lo que puede afectar la precisión del análisis en algunos casos. Esto podría ser una mejora a futuro, optimizando el uso de memoria, e implementando procesamiento por lotes.
- **Autenticación y concurrencia de usuarios:** La aplicación utiliza JWT con *Simple JWT* [29], lo que permite controlar el acceso de manera segura. Durante el desarrollo, el foco estuvo en la funcionalidad más que en el rendimiento masivo, por lo que no se realizaron pruebas exhaustivas con muchos usuarios simultáneos. Debido a esto, la cantidad de usuarios concurrentes que puede soportar es limitada. En despliegues reales de gran escala, sería necesario revisar y optimizar esta capacidad.

Capítulo 5

Validación de la Herramienta

La validación de la herramienta se separó en dos partes. En primer lugar, se realizó una validación funcional para verificar que cada etapa de la metodología *CaDQM* se ejecuta correctamente. Esto implica comprobar que la herramienta registra los datos de cada etapa, permite actualizar la lista de problemas de CD, permite actualizar el modelo de contexto y generar reportes de salida. Para esta validación se utilizó un conjunto de datos sobre libros y reseñas que los usuarios hacen sobre ellos [54]. El *dataset* se utilizó durante todo el proceso de implementación.

En segundo lugar, se realizó una validación para un caso real con un conjunto de datos sobre medicamentos suministrados a pacientes del Hospital de Clínicas, verificando la interoperabilidad con la herramienta desarrollada por el otro proyecto de grado. Además, se comparó el modelo de contexto obtenido utilizando la herramienta propuesta, con el modelo de contexto definido manualmente, para el mismo *dataset*, por expertos de dominio y de CD (definido en el marco de la tesis de Doctorado [1]).

5.1. Caso de Estudio 1: Funcionalidad

El propósito de este caso de estudio es validar las funcionalidades centrales de la herramienta. Específicamente, se busca demostrar que la herramienta ejecuta correctamente cada etapa de la Fase 1 - *DQ Planning*. Además, se valida la identificación de una lista de problemas de CD, disponible durante la ejecución de toda la metodología. Por otro lado, se busca validar el modelo de contexto, que se crea en la Etapa 1 - *Elicitation*, el cual es refinado y actualizado en las fases 1 y 2 de *CaDQM*. Por último, se busca validar los reportes generados en la salida de cada etapa.

5.1.1. Fuente de datos

Para el primer caso de estudio se utilizaron los datos de reseñas de libros publicadas por usuarios en Amazon, disponibles en [54]. Este conjunto de datos fue elegido en conjunto con el otro proyecto de grado, encargado de implementar la Fase 2 - *DQ Assessment*. De esta forma, se validó que la salida de la herramienta propuesta en este proyecto es compatible con la entrada de la herramienta del otro proyecto de grado.

Los datos se presentan como dos archivos con formato csv: `book_details.csv` y `reviews.csv`, cuya estructura se muestra en la Figura 5.1. El primero contiene información general sobre los libros. Incluye las columnas: *title*, *description*, *authors*, entre otras. Por otro lado, el segundo archivo contiene las reseñas de los libros realizadas por usuarios. Contiene las columnas: *id*, *title* (título del libro al cual se le esta haciendo la reseña), *review_score*, *user_id*, entre otras columnas que aportan información sobre la reseña.

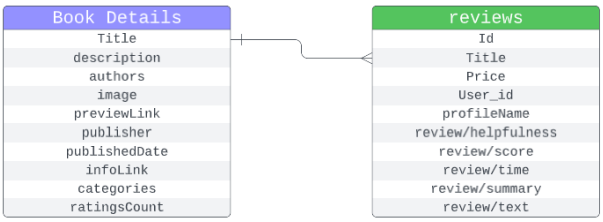

Figura 5.1: Información del dataset de libros y reseñas, tomado de [54].

Los datos de los libros y las reseñas se vinculan a través de la columna *title*, y hay una relación 1 a N entre *book details* y *reviews*, es decir, un libro puede tener muchas reseñas y una reseña debe ser de un solo libro.

5.1.2. Ejecución y Resultados

En este caso de estudio se ejecutaron las etapas 1 - *Elicitation* y 2 - *Data Analysis* de la fase 1 de *CaDQM*. Se optó por no ejecutar la Etapa 3 - *User Requirements Analysis* porque, al ser datos de la web, no se contaba con usuarios de los datos o expertos del dominio necesarios para ejecutar esta etapa. Como se describió en la Subsección 2.3.1, esto es una posibilidad que brinda la metodología, por lo tanto, se realizó un análisis basado únicamente en los datos, en lugar de un análisis centrado en el usuario [1].

Ejecución de la etapa 1: Elicitation

A continuación se describen las actividades de esta etapa.

Actividad 1 - Selección de los datos: Para la primera actividad, como la herramienta solo permite analizar un único conjunto de datos relacional, se creó una base de datos en PostgreSQL a partir de los dos archivos csv y se cargaron los datos de conexión a esa base de datos. Esto valida la capacidad de la herramienta para integrarse con bases de datos relacionales.

Actividad 2 - Análisis de los elementos de la organización: En este caso no son datos de una organización, por lo tanto se hace referencia al dominio de los datos. Entonces, para esta actividad, se cargó un documento con elementos del dominio (tipos de usuario, tareas, requerimientos, etc.), creado a partir de una letra de laboratorio del curso de Calidad de Datos [2], donde se evaluó la calidad del mismo conjunto de datos. Adicionalmente, se utilizaron dos archivos con metadatos, que describen las columnas de las tablas. Los contenidos de estos documentos se encuentran en el Anexo B.1.1.

Actividad 3 - Identificación de problemas de CD: De las las sugerencias generadas por el componente de IA, generadas a partir de la información cargada en la actividad anterior, se seleccionaron los siguientes problemas de CD:

- El campo *title* puede contener títulos duplicados o inconsistentes.
- Duplicados de reseñas si no hay restricción de unicidad en la combinación de *user_id* y id del libro.

- El campo *categories* puede tener valores fuera de un conjunto predefinido de géneros.
- El campo *preview_link* y *info_link* pueden contener enlaces caducados o inválidos.

Actividad 4 - Definición del modelo de contexto: Analizando los elementos del dominio, utilizando la información cargada en la actividad 2 y considerando las sugerencias de la herramienta, surgieron componentes de contexto que fueron agregados al modelo de contexto. Se detalla cuales fueron sugeridas por la IA, cuales modificadas y cuales agregadas manualmente.

Application Domain:

- Libros. - sugerida por IA

Business Rules:

- Tener al menos 500 libros. - sugerida por IA
- Cada libro deberá tener asociado un título, al menos un autor y un editor. - sugerida por IA

Task at Hand:

- Gestión de los datos de la librería. - sugerida por IA
- Recomendación y promoción de libros. - sugerida por IA
- Análisis de los datos buscando estudiar comportamientos, preferencias y relaciones entre los clientes. - sugerida por IA y modificada

User Type:

- Administrador. - sugerida por IA
- Publicista digital. - sugerida por IA
- Analista de datos. - sugerida por IA

Data Filtering:

- Filtrar los libros cuya publicación sea del año actual (asociado a la *Task at hand*, Gestión de los datos). - sugerida por IA y modificada
- Libros de la editorial Wiley (asociado a la *Task at hand*, Gestión de los datos). - sugerida por IA y modificada
- Top 3 de los libros con mayor score (asociado a la *Task at hand*, Gestión de los datos). - sugerida por IA y modificada

DQ Requirements:

- Base de datos de la librería actualizada todos los viernes (asociado a *User type*, Publicista digital). - sugerida por IA como system requirement
- El 60 % de los libros deben tener al menos un score mayor o igual a 5 (asociado a *User type*, Publicista digital). - sugerida por IA y modificada
- Los títulos de los libros deben estar correctamente escritos (asociado a *User type*, Analista de datos). - sugerida por IA y modificada

- Los nombres de los autores debe tener al menos un nombre y un apellido (asociado a *User type*, Analista de datos). - agregada manualmente
- Al menos el 95 % de los libros debe tener título (asociado a *User type*, Analista de datos). - sugerida por IA y modificada

System Requirements:

- Los tiempos de respuesta del sitio Web de la librería no puede superar los 3 segundos. - sugerida por IA

Ejecución de la etapa 2: Data Analysis

A continuación, se presentan las actividades correspondientes a la etapa 2.

Actividad 5 - Data profiling: Para esta actividad se ejecutó el *data profiling* sobre cada una de las tablas de la base de datos, para obtener un resumen de su contenido. Además, utilizando el editor SQL, se ejecutaron consultas directamente sobre el esquema para validar reglas de negocio y requerimientos de CD ya definidos.

Primero se realizó la siguiente consulta SQL para validar el requerimiento de calidad "El 60 % de los libros deben tener al menos un score mayor o igual a 5":

```
SELECT
    ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM book_details),
        2) AS porcentaje
FROM (
    SELECT r.title
    FROM reviews r
    WHERE r.review_score >= 5
    GROUP BY r.title
) as subquery;
```

La ejecución de la consulta dio un resultado de 84.72 %, por lo que se verifica el cumplimiento del requerimiento de calidad.

Luego, a partir del *data profiling* se detectó que la columna `ratings_count` en la tabla `book_details` contiene un 76.6 % de valores nulos. Como esta columna, según los metadatos, refiere al promedio de reseñas de un libro, se utilizó el editor SQL de la herramienta para verificar si estos libros tenían reseñas válidas en la tabla `reviews`, realizando la siguiente consulta:

```
SELECT
  ROUND(count(bd.title) * 100.0 / (SELECT COUNT(*) FROM
      book_details), 2) AS per_faltantes
FROM book_details bd
WHERE
  bd.ratings_count IS NULL
  AND EXISTS (
    SELECT 1
      FROM reviews r
      WHERE r.title = bd.title AND r.review_score <> 0
   );
```

Esta consulta dio como resultado que el 76,58 % de los libros tienen `ratings_count` nulo, pero tienen alguna reseña con puntuación mayor a 0, lo que confirma una inconsistencia en los datos, identificando un nuevo problema de CD.

Por otro lado, a partir de los reportes de *data profiling*, se detectó que: el 19 % de las reseñas no tienen usuario asignado, el 15 % de los libros no tienen autor y el 38 % no tienen editor. Esto lleva a la identificación de nuevos problemas de CD.

Actividad 3 - Identificación de problemas de CD: A partir de la información recolectada en la actividad anterior, se registraron nuevos problemas de CD:

- El 15 % de los libros no tienen autor, violando la regla de negocio BR35.
- El 38 % de los libros no tienen editor (*publisher*), violando la regla de negocio BR35.
- El 76,58 % de los libros tienen review pero no tienen promedio de *score* en la tabla de libros (*reviews_count*)
- El 19 % de las *reviews* no tienen usuario

Actividad 6 - Estimación de la CD: En esta etapa, la herramienta permite registrar una conclusión cualitativa sobre el estado general de los datos. Se registró manualmente la siguiente:

- "El 15 % de los libros no tienen autor y el 38 % no tienen editor, violando reglas de negocio. Además, el 19 % de las reseñas no tienen un usuario asociado. Hay una inconsistencia, ya que el 76.58 % de los libros con reseñas no tienen actualizado su contador de calificaciones (*reviews_count*), lo que indica un problema en la integridad de los datos entre las tablas."

Por otro lado, se evaluó la utilidad de las sugerencias generadas por el componente de IA, que sugiere alertas e información que deduce del *data profiling* realizado. En este caso, se seleccionó el siguiente conjunto de alertas, de las sugeridas por la herramienta:

- La columna *user_id* en la tabla *reviews* tiene 1937 (19.4 %) valores faltantes.
- La columna *authors* en la tabla *book_details* tiene 1481 (14.8 %) valores faltantes.
- La columna *publisher* en la tabla *book_details* tiene 3581 (35.8 %) valores faltantes.
- La columna *ratings_count* en la tabla *book_details* tiene 7659 (76.6 %) valores faltantes.

Actividad 7 - Actualización del modelo de contexto: En esta etapa no se identificaron nuevos componentes de contexto a partir del *data profiling*, por lo que no se realizaron actualizaciones en el modelo de contexto.

Al finalizar la ejecución de la Fase 1 - *DQ Planning*, la herramienta permitió generar un reporte en PDF con toda la información recopilada durante el proceso, que se encuentra disponible en el Anexo B.1.2. Este documento se organiza por etapas e incluye, para la Etapa 1 - *Elicitation*, la información sobre los datos cuya calidad es evaluada, los problemas de CD y los componentes de contexto identificados. Para la Etapa 2 - *Data Analysis*, presenta el resultado del *data profiling*, junto con los problemas de CD, los componentes de contexto correspondientes y la estimación de CD registrada.

Adicionalmente, la herramienta ofrece la posibilidad de generar reportes específicos, resumidos, con todos los problemas de CD detectados durante la ejecución de la fase 1, o el modelo de contexto completo, sin necesidad de incluir el resto de la información, ni separarlo por etapas.

5.1.3. Validación de las funcionalidades

La ejecución del caso de estudio sobre libros permitió comprobar que la herramienta cumple con los objetivos planteados para la Fase 1 - *DQ Planning*. En cada etapa fue posible registrar la información requerida, reutilizarla en pasos posteriores y mantener la coherencia del flujo de datos durante la ejecución de toda la fase.

Además, utilizando la herramienta, se ejecutaron análisis de *data profiling* sobre los datos con las herramientas integradas. Además, junto con la herramienta de consultas SQL, fue posible identificar problemas de CD y generar una estimación inicial de los datos.

Por otro lado, se verificó la generación de reportes al finalizar cada etapa y la posibilidad de generar reportes específicos, como el listado de problemas de CD o el modelo de contexto completo.

La persistencia de la información recopilada en la base de datos común, también permitió que los datos generados fueran utilizados por el otro proyecto de grado para la ejecución de la Fase 2 - *DQ Assessment*. Esto valida, tanto la arquitectura diseñada por ambos proyectos de grado, como la capacidad de la herramienta para integrarse de manera efectiva con otro proyecto que implementa una fase posterior de *CaDQM*, garantizando el cumplimiento de los requerimientos de intercambio de datos de manera transparente para el usuario.

5.2. Caso de Estudio 2: Interoperabilidad y Análisis

En este caso de estudio se plantean dos objetivos:

1. Se verifica la interoperabilidad entre las herramientas que implementan la Fase 1 - *DQ Planning* y la Fase 2 - *DQ Assessment*, de *CaDQM*.
2. Se analiza qué tanto se ajusta a la realidad el modelo de contexto obtenido, mediante la aplicación de la herramienta propuesta en este trabajo.

Pensando en el primer objetivo, se siguieron los siguientes pasos:

- Se definió un modelo de contexto mediante la ejecución de la Fase 1 - *DQ Planning* de *CaDQM*, cuyos resultados fueron almacenados en la BD común: i) problemas de CD y ii) componentes de contexto que definen el modelo de contexto de los datos. Este paso se realizó utilizando nuestra herramienta.
- El otro proyecto de grado definió un modelo de CD mediante la ejecución de la Fase 2 - *DQ Assessment* de *CaDQM*, siendo la entrada de la fase 2, todo lo obtenido en la ejecución de la Fase 1 - *DQ Planning* y almacenado en la BD común (mediante nuestra herramienta).

De esta forma, se busca verificar que todos los elementos de entrada de la fase 2 están contemplados en la BD común.

Pensando en el segundo objetivo, tanto para el modelo de contexto obtenido en la fase 1, como para el modelo de CD obtenido en la fase 2, para el mismo *dataset*, se usaron modelos de referencia definidos manualmente. Ambos modelos de referencia fueron definidos en un trabajo conjunto, realizado por expertos de dominio y de CD, en el marco de la Tesis de Doctorado de Serra [1]. De esta forma, el modelo de contexto obtenido con la herramienta propuesta fue comparado con el modelo de contexto de referencia. Este mismo análisis fue realizado para el modelo de calidad obtenido en la fase 2, respecto al modelo de CD de referencia, el cual fue abordado por los estudiantes encargados de la Fase 2 - *DQ Assessment*. A continuación, presentamos los resultados obtenidos en la Fase 1 - *DQ Planning*, en particular para el modelo de contexto.

5.2.1. Fuente de datos

Para este caso de estudio se utilizaron datos reales proporcionados por el Centro de Evaluación de Biodisponibilidad y Bioequivalencia de Medicamentos de la Universidad de la República (CEBIOBE) [1].

CEBIOBE monitoriza la dosis de los medicamentos que reciben un conjunto de pacientes. La dosificación es la determinación de la concentración de un medicamento en un fluido biológico (plasma o líquido cefalorraquídeo) en un momento dado.

Especialmente este conjunto de datos contiene información sobre tres antibióticos: Amikacina, Vancomicina y Gentamicina. Los pacientes se representan con un número, debido a que los datos fueron anonimizados, para evitar manipular datos personales. Las descripción del *dataset* se encuentra en el Anexo B.2.1.

5.2.2. Ejecución y Resultados

En este caso se ejecutaron las tres etapas de la Fase 1 - *DQ Planning* de *CaDQM* (definida en la Sección 2.3), en el siguiente orden: Etapa 1 - *Elicitation*, Etapa 2 - *Data Analysis* y Etapa 3 - *User Requirements Analysis*. Recordar que las etapas 2 y 3 pueden ser ejecutadas en cualquier orden.

En el Anexo B.2.2 se presenta el resultado de la comparación entre el modelo de contexto obtenido al finalizar la ejecución de la etapa 3 y el modelo de contexto de referencia. Los resultados de la comparación realizada presentan las siguientes características:

- Un conjunto de componentes de contexto coinciden en ambos modelos de contexto.
- Un conjunto de componentes de contexto solo aparecen en el modelo de contexto de referencia.
- Un conjunto de componentes de contexto (muy reducido) coinciden en ambos modelos de contexto, pero presentan algunas diferencias.
- Un conjunto de componentes de contexto solo fueron identificados con la herramienta propuesta.

En la Figura 5.2 se presenta gráficamente la distribución de estos componentes de contexto respecto a los componentes de contexto que definen al modelo de contexto de referencia. Notar que los componentes de contexto que solo fueron identificados con la herramienta no son considerados.

Cabe destacar que, para la ejecución de la Etapa 3 - *User Requirements Analysis*, se contó con documentación sobre varias instancias de intercambio con usuarios expertos en el dominio de los datos. Adicionalmente, se realizaron reuniones de validación con uno de los expertos de calidad que trabajaron en la definición del modelo de contexto de referencia (Flavia Serra). En estas comunicaciones se registraron consultas específicas sobre los campos del *dataset*, incluyendo definiciones, valores esperados, formatos, umbrales máximos y mínimos, y tolerancias para datos faltantes en distintas columnas. Esto permitió entender mejor el dominio e incorporar componentes de contexto al modelo.

[1] Centro de Evaluación de Biodisponibilidad y Bioequivalencia de Medicamentos (CEBIOBE), Universidad de la República, Uruguay. http://www.cebiobe.edu.uy/ [55]

Interpretación General de los Resultados

El conjunto de componentes de contexto que coincide en ambos modelos de contexto representan el 52 % de los componentes de contexto que definen al modelo de referencia.

El análisis comparativo reveló que con la ayuda de la herramienta *CaDQM* se logró identificar correctamente más de la mitad de los componentes del contexto (52 %), identificados en el modelo de contexto de referencia. Los componentes de contexto presentes únicamente en el modelo de contexto de referencia (46 %) no constituyen deficiencias de la herramienta, sino elementos que requieren múltiples iteraciones de refinamiento con expertos en el dominio, las cuales no fueron realizadas en esta prueba. Si bien se realizaron múltiples validaciones con un experto en CD, no se realizaron entrevistas con expertos de dominio. Por último, se encontró un componente de contexto (2 %) incluido en el modelo de contexto de referencia, pero que presenta algunas diferencias.

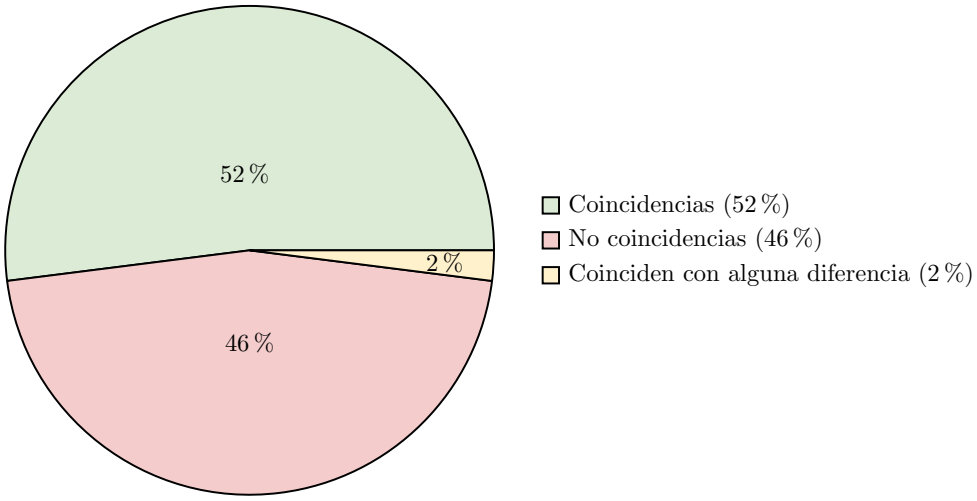

Figura 5.2: Distribución de componentes del contexto con respecto a los componentes de contexto que definen al modelo de contexto de referencia.

Análisis Detallado por Componente de contexto

La Tabla 5.1 muestra el desglose de efectividad por cada tipo de componente de contexto. Tener en cuenta que solo se consideran los tipos de componentes de contexto presentes en el modelo de contexto de referencia:

Categoría	**Coincidencias**	**Efectividad**
Application Domain	1/1	100 %
User Types	3/3	100 %
Tasks At Hand	3/4	75 %
Business Rules	11/22	50 %
DQ Requirements	6/13	46 %
Data Filtering	1/3	33 %
System Requirements	-	-

Tabla 5.1: Efectividad de la herramienta por tipo de componente de contexto.

Los resultados muestran una variación significativa en la capacidad de la herramienta para identificar diferentes tipos de componentes de contexto, desde una efectividad perfecta del 100 % en *Application Domain* y *User Types*, hasta efectividades más moderadas en componentes de contexto que requieren mayor interpretación del dominio de los *data at hand*, como es el caso de *Data Filtering* (33 %), *DQ Requirements* (39 %) y *Business Rules* (50 %).

Componentes de contexto con Efectividad Perfecta (100 %) y Alta (75 %): Los componentes de contexto *Application Domain* y *User Types* alcanzaron una coincidencia perfecta, mostrando que la herramienta es altamente efectiva para identificar los aspectos fundamentales del dominio de aplicación y los perfiles de usuarios involucrados.

El componente de contexto *Tasks At Hand* logró una efectividad del 75 % (3/4 elementos), demostrando que usando la herramienta se puede identificar correctamente la mayoría de las tareas que los usuarios realizan con los datos. La única tarea no identificada (T4: Proveer datos) con la herramienta corresponde a actividades organizacionales específicas que requieren conocimiento del dominio operacional.

Componentes de contexto con Efectividad Moderada (50 %) y Baja (33-46 %): Las *Business Rules* presentaron una efectividad del 50 % (11/22 elementos). Las reglas identificadas corresponden principalmente a validaciones técnicas observables en los datos, mientras que las reglas no detectadas requieren conocimiento específico del dominio clínico o interpretación de políticas organizacionales.

Los componentes de contexto *Data Filtering* (33 %) y *DQ Requirements* (46 %) presentaron las efectividades más bajas, reflejando la complejidad inherente en la identificación automática de necesidades específicas de filtrado y requerimientos de CD. Estos componentes de contexto típicamente emergen en interacciones directas con usuarios y análisis de casos de uso específicos que no son evidentes únicamente a partir del análisis de datos y documentos.

Por otro lado, se identificó un *DQ Requirement* que, si bien se incluye en el modelo de contexto de referencia, presenta algunas diferencias. Este requerimiento hace referencia a qué porcentaje de cada variable se exige para que los análisis sean representativos. En el modelo de contexto de referencia, se establece un umbral del 50 %, mientras que en el contexto generado utilizando la herramienta, el valor es del 80 %. Con el objetivo de validar esta diferencia, se revisó en detalle y manualmente la documentación utilizada durante todas las etapas de la Fase 1 - *DQ Planning*, es decir, la documentación usada por la IA. En este caso se verificó que, en los intercambios con los usuarios, se menciona que el máximo de datos vacíos en cada variable es 20 %. Esto concluye que podría haber existido una confusión al momento de definir el modelo de contexto de referencia, o que ese requerimiento de CD fue modificado posteriormente a partir de nuevas interacciones entre los usuarios, que no fueron registradas.

Por último, utilizando la herramienta se identificaron **5 componentes de contexto adicionales**, que no estaban presentes en el modelo de contexto de referencia, los cuales se detallan en la Tabla B.5 del Anexo B.2. Estos componentes de contexto son: 2 *Business Rules*, 1 *System Requirement* y 2 *Data Filtering Requirements*. Al igual que la diferencia mencionada en el párrafo anterior, se realizó una revisión de la documentación para verificar si estos componentes de contexto debían formar parte del modelo de contexto.

En el caso de las *Business Rules*, se verificó que ambas aparecen mencionadas en los intercambios de correos electrónicos entre los usuarios de los datos. Esta diferencia con el modelo de contexto de referencia podría explicarse por dos motivos: i) es posible que estas reglas hayan sido inicialmente consideradas y luego en otros intercambios

entre los usuarios de los datos, se hayan descartado, o ii) podría deberse a que nunca fueron registradas, a pesar de haberse mencionado en la interacción entre los usuarios.

Con respecto al *System Requirement* y a los *Data Filtering Requirements* identificados utilizando la herramienta, no se encontraron referencias en la documentación revisada. Sin embargo, analizándolos se observó que se trata de componentes de contexto generales, que se relacionan con el uso de un conjunto de datos, y no específicamente con el dominio. Además, estos componentes de contexto adicionales no contradicen ningún componente de contexto definido en el modelo de contexto de referencia. Por lo tanto, se concluye que, aunque no se encuentran en el modelo de contexto de referencia, su inclusión podría ser válida.

Finalmente destacamos que los resultados, respecto a los componentes de contexto de menor efectividad, demuestra la importancia que tiene la Etapa 3 - *User Requirements Analysis* en la fase 1 de *CaDQM*. En esta etapa es donde la interacción con los expertos de dominio es muy relevante para la identificación de las reglas de negocio y las necesidades de los usuarios. Etapa que fue muy poco explotada, ya que si bien existió interacción con un experto de calidad, por el alcance de este caso de estudio, no se interactuó con expertos de dominio.

Conclusiones generales

Respecto al primer objetivo, verificación de interoperabilidad, destacamos que este caso de estudio ayudó a identificar necesidades del proyecto de grado encargado de la Fase 2 - *DQ Assessment*, que aún no habían sido reflejadas en la BD común. Por lo tanto, permitió un diseño de la BD que efectivamente contempla los requerimientos de ambos proyectos de grado.

Respecto al segundo objetivo, comparación del modelo de contexto obtenido utilizando la herramienta con un modelo de contexto real de referencia, se logró una coincidencia del 52 % de los componentes de contexto. Además, se evidenció la relevancia de la participación de los expertos de dominio y los usuarios de los datos en un proceso de GCD.

Capítulo 6

Conclusiones y Trabajo Futuro

En este capítulo se presentan las conclusiones obtenidas a partir del trabajo realizado. Además, se describen trabajos a futuro que se consideran que enriquecerán la herramienta propuesta. Entre esos trabajos se destacan posibles mejoras que fueron detectadas durante la realización de este proyecto.

6.1. Conclusiones

En este proyecto se logró diseñar y desarrollar una herramienta Web que brinda soporte a la Fase 1 - *DQ Planning* de la metodología *Context-aware Data Quality Management (CaDQM)*. La herramienta permite la ejecución de las tres etapas que conforman esta fase: *Elicitation*, *Data Analysis* y *User Requirements Analysis*, permitiendo registrar y gestionar los resultados de cada actividad propuesta por cada etapa, manteniendo una coherencia en el flujo de los datos a lo largo de toda la fase.

La herramienta desarrollada integra herramientas de *data profiling* y un modelo de inteligencia artificial que asiste al usuario en la identificación de problemas de CD, la definición del modelo de contexto y la estimación de la CD, a partir del análisis de documentos y la información registrada por el usuario. Estas funcionalidades permiten reducir el tiempo y esfuerzo requerido para el registro manual de la información, siempre bajo la validación final del usuario que lo registra. Asimismo, la herramienta permite generar reportes con los resultados obtenidos durante la ejecución de la Fase 1 - *DQ Planning*, la cual es un valor agregado al momento de la toma de decisiones.

La validación de la herramienta se llevó a cabo mediante dos casos de estudio. El primero permitió comprobar el correcto funcionamiento de la herramienta, demostrando que es posible ejecutar la Fase 1 - *DQ Planning* completamente, es decir, abordando cada una de las actividades propuestas en ella. El segundo caso de estudio se enfocó en la interoperabilidad de las herramientas desarrolladas por dos proyectos de grado diferentes. La interoperabilidad fue lograda mediante el diseño de una base de datos común, que permitió la integración de la Fase 1 - *DQ Planning* con la Fase 2 -*DQ Assessment*. Esto aseguró que los resultados de la primera fase pueden ser utilizados en fases posteriores de *CaDQM*.

Por otro lado, en el mismo caso de estudio, el análisis del modelo de contexto obtenido con la herramienta propuesta, dio una coincidencia del 52 % al compararlo con un modelo de contexto de referencia, creado por expertos de dominio y CD. La validación permitió identificar que las principales diferencias entre el modelo de contexto obtenido y el de referencia, se debían a la falta de interacción con los expertos de dominio y los usuarios de los datos. Esta interacción se realiza en la etapa tres de

la Fase 1, la cual no fue abordada por no tener acceso a los usuarios de los datos. Este resultado coincide con la necesidad identificada en la bibliografía de CD: el usuario de los datos es muy relevante en las tareas de GCD.

En conclusión, los objetivos planteados en este proyecto fueron cumplidos: se diseñó y desarrolló un prototipo de herramienta Web que brinda soporte a expertos de CD en el registro de la información obtenida durante la ejecución de la Fase 1 - *DQ Planning* de la metodología *CaDQM*, y se diseñó una base de datos común que facilita la integración con otras herramientas para fases posteriores. La herramienta propuesta aporta una contribución original al área de CD, ya que hasta donde fue posible investigar, no existen herramientas de este tipo: implementa una metodología que analiza y define el contexto de los datos para la GCD.

6.2. Trabajo futuro

Durante el desarrollo del proyecto se identificaron distintas propuestas y mejoras para la herramienta implementada. Debido al alcance del proyecto, todas ellas quedaron por fuera de la implementación y se consideran trabajos futuros. Estos son:

- *Deploy*: Uno de los siguientes pasos fundamentales es desplegar la aplicación en distintos ambientes, tanto de desarrollo, donde se puedan probar las funcionalidades, y de producción, donde funcione la herramienta para los usuarios reales y de manera masiva.
- Soportar otras fuentes de datos: Actualmente, la herramienta se restringe solo a bases de datos relacionales. Una mejora sería soportar otros tipos de datos como bases de datos no relacionales, archivos locales, como CSV o JSON y fuentes de datos en la nube. Además, esto implicaría ajustar los análisis de *data profiling* para estos tipos de datos.
- Sumar otras herramienta de *data profiling*: Se propone integrar nuevas herramientas de *data profiling*, a las que el usuario se puede conectar, para realizar y visualizar otros tipos de análisis desde la aplicación. Por el momento, la herramienta brinda la opción de crear reportes de *data profiling* utilizando R [35] e YDataProfiling [52]. Además, podría ser útil que los usuarios integren herramientas de escritorio que ya utilizan, o herramientas de pago a las cuales ya estén suscriptos.
- Crear usuarios con distintos roles y permisos: La metodología *CaDQM* sugiere roles específicos responsables de las actividades en cada etapa, donde un usuario puede desempeñar múltiples roles. Actualmente, la herramienta permite crear usuarios, pero todos tienen los mismos permisos. Como trabajo futuro, se puede implementar un sistema de roles de usuario que refleje lo planteado en la metodología *CaDQM*. Esto permitiría crear perfiles como Administrador, Analista de Datos y Experto de CD, donde cada uno es responsable de una funcionalidad especifica.
- Integrar herramientas de IA: Sería muy útil contar con herramientas de IA más potentes donde se puedan realizar *data profiling* de manera automatizada y con un análisis más amplio de los datos, que no solo sea por tabla como lo es hoy en día, sino también incluyendo relaciones, claves foráneas, etc.
- Adaptar a otras resoluciones de pantalla: Es importante adaptar el *frontend* para que las pantallas sean *responsive*, es decir, que se adapten a distintos tamaños, ya que actualmente es una web de escritorio.
- Unificar con la herramienta propuesta por el otro proyecto de grado: Un paso fundamental es unir la herramienta creada en este proyecto de grado, uniendo

asi, la Fase 1 - *DQ Planning* y la Fase 2 - *DQ Assessment* de *CaDQM*. A pesar de que son 2 códigos fuente distintos e independientes, es importante obtener la unificación de las dos aplicaciones para una única herramienta.

- Investigar e implementar sobre la Fase 3 - *DQ Improvement*: Para completar la ejecución de la metodología *CaDQM*, se propone investigar sobre otras herramientas de CD que aborden actividades de mejora de la CD. Esto último con el objetivo de implementar una herramienta para ejecutar la Fase 3 - *DQ Improvement* y unirla a las dos herramientas ya implementadas e integradas.

Bibliografía

[1] Flavia Serra. *Context-aware Data Quality Management*. Tesis doctoral, PEDECIBA – Universidad de la República, Université de Tours, 2024.

[2] Fing, UdelaR. Calidad de datos e información, 2025. https://eva.fing.edu.uy/course/view.php?id=1073, Último acceso: Agosto 2025.

[3] Richard Y. Wang and Diane M. Strong. Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, 12(4):5–33, 1996. http://www.jstor.org/stable/40398176, Último acceso: Agosto 2025.

[4] Leopoldo Bertossi, Flavio Rizzolo, and Lei Jiang. Data quality is context dependent. In Malu Castellanos, Umeshwar Dayal, and Volker Markl, editors, *Enabling Real-Time Business Intelligence*, pages 52–67, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[5] Carlo Batini and Monica Scannapieco. *Data and Information Quality*. Springer, 2016.

[6] Flavia Serra, Veronika Peralta, Adriana Marotta, and Patrick Marcel. Context-aware data quality management methodology. In *New Trends in Database and Information Systems*, pages 245–255, Cham, 2023. Springer Nature Switzerland.

[7] Alireza Hassani, Alexey Medvedev, Pari Delir Haghighi, Sea Ling, Arkady Zaslavsky, and Prakash Jayaraman. Context definition and query language: Conceptual specification, implementation, and evaluation. *Personal & Ubiquitous Computing*, 5(1), 2019.

[8] Mary Bazire and Patrick Brézillon. Understanding context before using it. In *Proceedings of the International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT 2005)*. Springer, July 2005.

[9] Anind K. Dey. Understanding and using context. *Personal & Ubiquitous Computing*, 5(1), 2001.

[10] Carlo Batini, Cinzia Cappiello, Chiara Francalanci, and Andrea Maurino. Methodologies for data quality assessment and improvement. *ACM Comput. Surv.*, 41:16:1–16:52, 2009.

[11] Carlo Batini, Federico Cabitza, Cinzia Cappiello, and Chiara Francalanci. A comprehensive data quality methodology for web and structured data. *Digital Information Management, 2006 1st International Conference on*, 1:448–456, 01 2006.

[12] Superconductive Health Inc. Great expectations: Always know what to expect from your data, 2024. Framework de validación de datos open-source, https://greatexpectations.io/, Último acceso: Agosto 2025.

[13] Monte Carlo Data Inc. Monte carlo: The data observability platform, 2024. Plataforma comercial de observabilidad de datos, https://www.montecarlodata.com/, Último acceso: Agosto 2025.

[14] Talend Inc. Talend data quality: Enterprise data quality and governance platform, 2024. Suite empresarial de gestión de calidad de datos, https://www.talend.com/products/data-quality/, Último acceso: Agosto 2025.

[15] Meta Platforms, Inc. React: The library for web and native user interfaces. https://react.dev/, Último acceso: Agosto 2025.

[16] Microsoft. Typescript: Javascript with syntax for types. https://www.typescriptlang.org/, Último acceso: Agosto 2025.

[17] Skillions. The importance of typescript in modern frontend development. https://skillions.in/the-importance-of-typescript-in-modern-frontend-development/, Último acceso: Agosto 2025.

[18] JavaScript. Axios: Promise based http client for the browser and node.js. https://axios-http.com/docs/intro, Último acceso: Agosto 2025.

[19] React Team. Hooks — react, 2025. https://react.dev/reference/react/hooks, Último acceso: Agosto 2025.

[20] Material UI, trading as MUI. https://mui.com, Último acceso: Agosto 2025.

[21] React-PDF Contributors. React-pdf: React renderer for creating pdf files on the browser and server. https://react-pdf.org, Último acceso: Agosto 2025.

[22] Interaction Design Foundation. What are user flows?, 2016. https://www.interaction-design.org/literature/topics/user-flows, Último acceso: Agosto 2025.

[23] Therese Fessenden. Wizards: Definition and design recommendations, 2017. https://www.nngroup.com/articles/wizards/, Último acceso: Agosto 2025.

[24] Django. Django: The web framework for perfectionists with deadlines. https://www.djangoproject.com/, Último acceso: Agosto 2025.

[25] Python Software Foundation. Python: A dynamic, open source programming language. https://www.python.org/, Último acceso: Agosto 2025.

[26] PostgreSQL. Postgresql: The world's most advanced open source relational database, 2024. https://www.postgresql.org/, Último acceso: Agosto 2025.

[27] Psycopg. Psycopg: Postgresql database adapter for python, 2024. https://www.psycopg.org/, Último acceso: Agosto 2025.

[28] Django Software Foundation. Django design philosophies: Model-view-template, 2024. https://docs.djangoproject.com/en/stable/faq/general/, Último acceso: Agosto 2025.

[29] Simple JWT. Simple jwt: A json web token authentication plugin for django rest framework, 2024. https://django-rest-framework-simplejwt.readthedocs.io/, Último acceso: Agosto 2025.

[30] Auth0. Json web tokens. https://auth0.com/docs/secure/tokens/json-web-tokens, Último acceso: Agosto 2025.

[31] PostgreSQL Global Development Group. Postgresql: The world's most advanced open source relational database, 2024. https://www.postgresql.org/, Último acceso: Agosto 2025.

[32] Oracle Corporation. Mysql: The world's most popular open source database, 2024. https://www.mysql.com/, Último acceso: Agosto 2025.

[33] SQLite Development Team. Sqlite: Small. fast. reliable. choose any three., 2024. https://www.sqlite.org/, Último acceso: Agosto 2025.

[34] Docker, Inc. Docker: Accelerated container application development. https://www.docker.com/, Último acceso: Agosto 2025.

[35] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. https://www.R-project.org/, Último acceso: Agosto 2025.

[36] Laurent Gautier et al. rpy2: Python interface to the r language, 2023. Python package version 3.5.17, https://rpy2.github.io/, Último acceso: Agosto 2025.

[37] Docker, Inc. What is docker? https://docs.docker.com/get-started/docker-overview/, Último acceso: Agosto 2025.

[38] Carlo Batini, Cinzia Cappiello, Chiara Francalanci, and Andrea Maurino. Methodologies for data quality assessment and improvement. *ACM Computing Surveys*, 41(3):1–52, 2009. https://doi.org/10.1145/1541880.1541883, Último acceso: Agosto 2025.

[39] DataCleaner. Datacleaner, 2025. https://datacleaner.github.io/, Último acceso: Agosto 2025.

[40] Thorsten Papenbrock and Felix Naumann. Metanome: A tool for metadata discovery, 2016. https://github.com/HPI-Information-Systems/Metanome, Último acceso: Agosto 2025.

[41] YData. Ydata profiling - documentation. https://docs.profiling.ydata.ai/, Último acceso: Agosto 2025.

[42] Pandas development team. Pandas documentation. https://pandas.pydata.org/docs/, Último acceso: Agosto 2025.

[43] NumPy Developers. Numpy: The fundamental package for scientific computing with python. https://numpy.org/doc/stable/, Último acceso: Agosto 2025.

[44] Hadley Wickham, Jeroen Ooms, and Kirill Müller. Rpostgres: Rcpp interface to postgresql, 2023. R package version 1.4.5, https://cran.r-project.org/package=RPostgres, Último acceso: Agosto 2025.

[45] R Special Interest Group on Databases. Dbi: R database interface, 2023. R package version 1.1.3, https://cran.r-project.org/package=DBI, Último acceso: Agosto 2025.

[46] OpenAI. Gpt-4 technical report, 2023. https://openai.com/research/gpt-4, Último acceso: Agosto 2025.

[47] DeepSeek. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model, 2024. https://deepseek.com/, Último acceso: Agosto 2025.

[48] Groq. Groq: Fast ai inference, 2024. https://groq.com/, Último acceso: Agosto 2025.

[49] Meta AI. Introducing meta llama 3: The most capable openly available llm to date, 2024. https://ai.meta.com/blog/meta-llama-3/, Último acceso: Agosto 2025.

[50] Groq. Rate limits - groq cloud docs. https://console.groq.com/docs/rate-limits, Último acceso: Agosto 2025.

[51] Groq Inc. Llama-3.3-70b-specdec model documentation, 2024. https://console.groq.com/docs/model/llama-3.3-70b-specdec, Último acceso: Agosto 2025.

[52] YData Team. ydata-profiling: Create html profiling reports from pandas dataframe objects, 2023. Versión 4.0+, https://github.com/ydataai/ydata-profiling, Último acceso: Agosto 2025.

[53] Boxuan Cui. Dataexplorer: Automate data exploration and treatment, 2023. R package version 0.8.2, https://cran.r-project.org/package=DataExplorer, Último acceso: Agosto 2025.

[54] Mohamed Bakhet. Amazon books reviews, 2022. https://www.kaggle.com/datasets/mohamedbakhet/amazon-books-reviews/data, Último acceso: Agosto 2025.

[55] Uruguay Universidad de la República. Centro de evaluación de biodisponibilidad y bioequivalencia de medicamentos. http://www.cebiobe.edu.uy/, Último acceso: Agosto 2025.

[56] GitHub Inc. Github: Where the world builds software, 2025. https://github.com, Último acceso: Agosto 2025.

[57] Fernando Rabago. Tesis backend - cadqm implementation, 2025. https://github.com/frabagocds/Tesis_backend, Último acceso: Agosto 2025.

[58] Martina Revello. Tesis frontend - cadqm web interface, 2025. https://github.com/mrevello/CaDQM1_frontend, Último acceso: Agosto 2025.

Anexo A

Manual de usuario

A.1. Instalación y configuración

El código del proyecto se encuentra en dos repositorios de *GitHub* [56]. Por un lado, el código del *backend* [57] y, por otro, el código del *frontend* [58]. Ambas partes se pueden ejecutar localmente sin necesidad de instalar todas las dependencias, mediante el uso de *Docker* [34], donde se crean dos contenedores para correr la aplicación web y la API simultáneamente.

A.1.1. Requisitos previos

Se debe tener *Docker* [34] instalado y en ejecución.

A.1.2. Instalación

En primer lugar, se deben clonar ambos repositorios. Para eso, se deber ejecutar:

- `git clone https://github.com/mrevello/CaDQM1_frontend.git`
- `git clone https://github.com/frabagocds/Tesis_backend.git`

En el la carpeta `Tesis_backend`, se debe crear un archivo `.env` con las configuraciones de la base de datos, siguiendo el formato que se ejemplifica en el archivo `env-example.txt`.

Luego, dentro de las carpetas `CaDQM1_frontend` y `Tesis_backend`, se deben correr los siguientes comandos:

- Solo la primera vez, compilar las imágenes de Docker, crear los contenedores, instalar todas las dependencias necesarias y ejecutarlos:

  ```
  docker-compose up --build
  ```

- Luego, cada vez que se quiera ejecutar los contenedores:

  ```
  docker-compose up
  ```

Una vez que los contenedores `cadqm_frontend_web` y `cadqm_backend_web` estén en ejecución, la aplicación frontend estará disponible en `http://localhost:3000` y se comunicará con la API del backend en `http://localhost:8000`.

A.2. Funcionalidades

A continuación, se explica el uso de la aplicación mediante un ejemplo de ejecución de toda la Fase 1 - *DQ Planning* de la metodología *CaDQM*.

En primer lugar, se muestra una pantalla de login para autenticarse, que se muestra en la Figura A.1. En caso de que no tener un usuario creado, se puede ir a la pantalla de registro para crear uno, que se muestra en la en la Figura A.2:

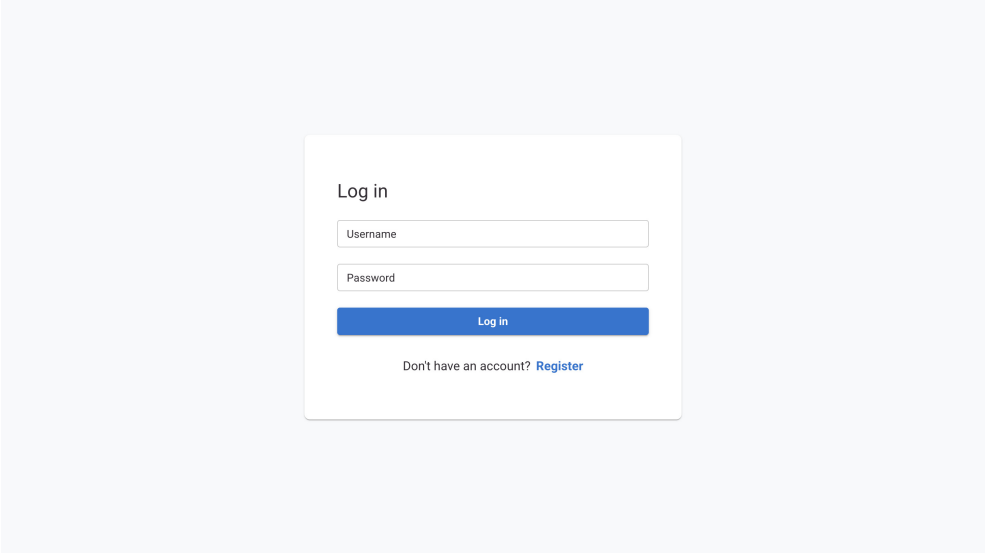

Figura A.1: Autenticación - Pantalla de inicio de sesión.

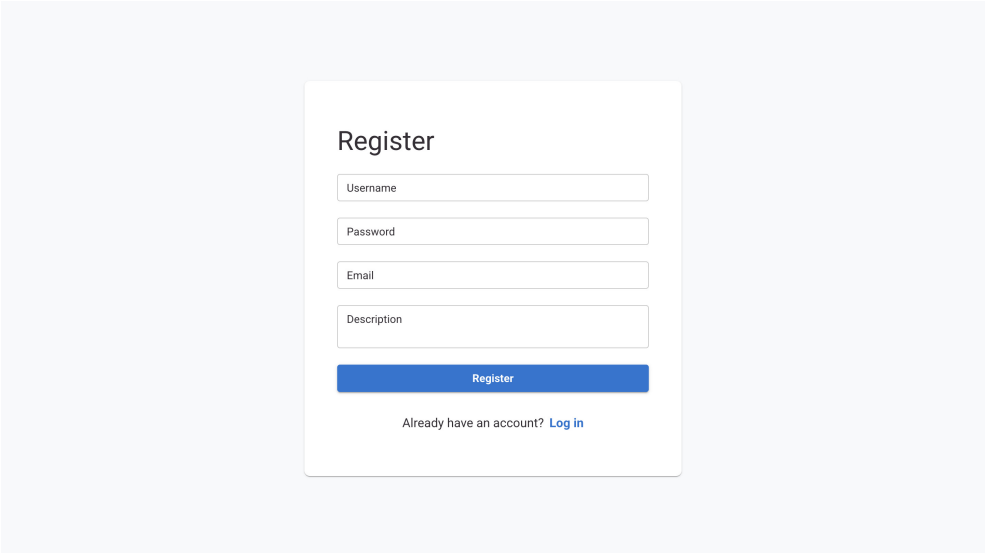

Figura A.2: Autenticación - Pantalla de registro.

Una vez iniciada la sesión, se muestra la pantalla de inicio (ver Figura A.3) donde se puede ver la lista de proyectos del usuario. Estos proyectos pueden ser filtrados, editados (sus datos básicos como su nombre y descripción) o eliminados. Para crear un proyecto nuevo, se debe presionar el botón `[+ New]`, que abrirá un diálogo de creación.

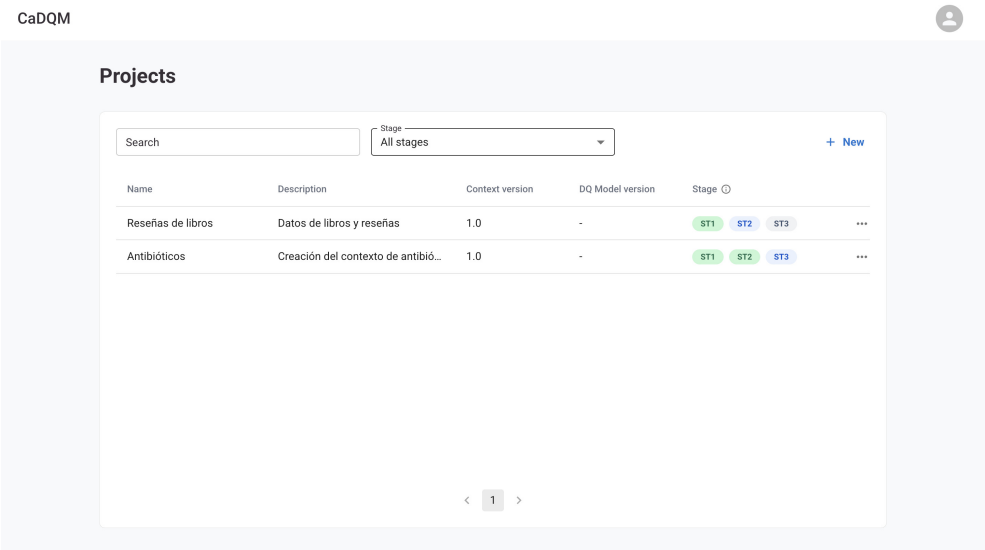

Figura A.3: Pantalla de inicio - lista de proyectos

En la pantalla de la Figura A.3 también se puede reanudar un proyecto en proceso. Para esto, se debe clickear en él, donde aparecerá un *popup* con los detalles del proyecto. Allí se muestra una línea de tiempo con progreso del proyecto, como se muestra en la Figura A.4.

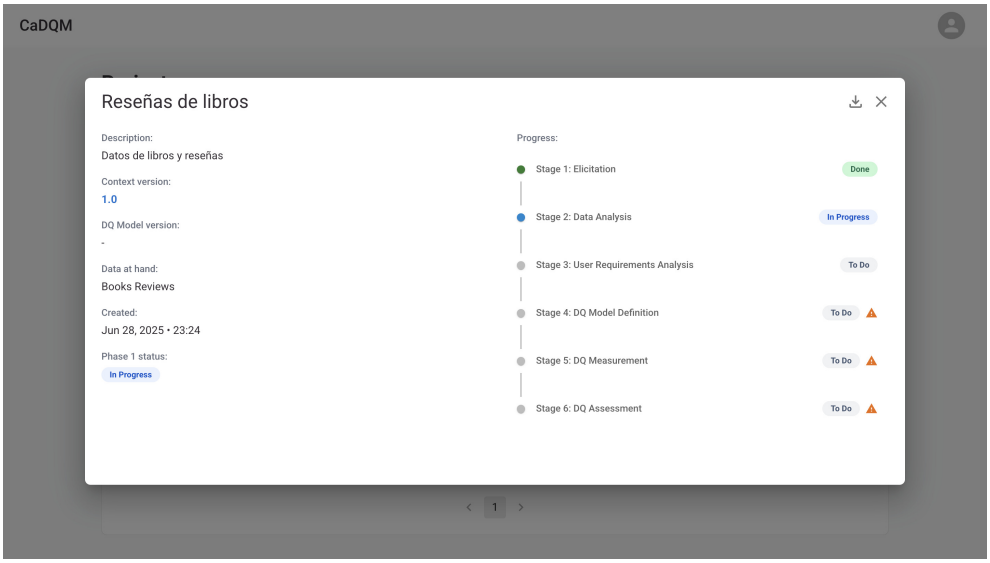

Figura A.4: Detalle del proyecto

En esta línea de tiempo se muestran las etapas de las fases 1 - *DQ Planning* y 2 - *DQ Assessment* con su respectivo estado de ejecución. Al seleccionar una etapa, es posible ejecutarla o reanudar su ejecución, en caso que este habilitada. Una etapa está habilitada para ser seleccionada si la etapa anterior en la metodología ya fue completamente ejecutada. En la Fase 1 - *DQ Planning*, la etapa 1 se puede comenzar o reanudar hasta finalizar su ejecución. Las etapas 2 y 3 están habilitadas una vez que se finaliza la etapa 1, son independientes entre sí y se pueden ejecutar en paralelo. Una vez que una etapa es finalizada, ya no puede ser modificada. Las etapas de la Fase 2 - *DQ Assessment* no están habilitadas para seleccionar en este proyecto.

Al seleccionar una etapa, se inicia un flujo de trabajo guiado, que presenta cada

actividad como un paso secuencial. La navegación entre las actividades se realiza mediante los botones `Atrás` y `Siguiente`. En la actividad final, un botón permite completar y cerrar la etapa. La interfaz muestra en todo momento qué actividades preceden a la actual y cuáles quedan por ser ejecutadas. Además, cada actividad cuenta con un ícono de información (ℹ) que, al pasar el cursor sobre él, despliega una descripción de la tarea a realizar.

A.2.1. Ejecución de la Etapa 1 - *Elicitation*

En esta primera etapa se definen los datos a evaluar y se recopila toda la información de contexto necesaria para las fases posteriores. El proceso consta de cuatro pantallas secuenciales, una por cada actividad.

Actividad 1 - Selección de los datos: En la Figura A.5 se muestra un formulario para ingresar los datos de conexión a la base de datos que se va a evaluar: nombre, descripción (opcional), host, puerto y nombre de la base y, por último, las credenciales. Al avanzar, se chequea que esa conexión sea correcta y se notifica al usuario si tiene algún error.

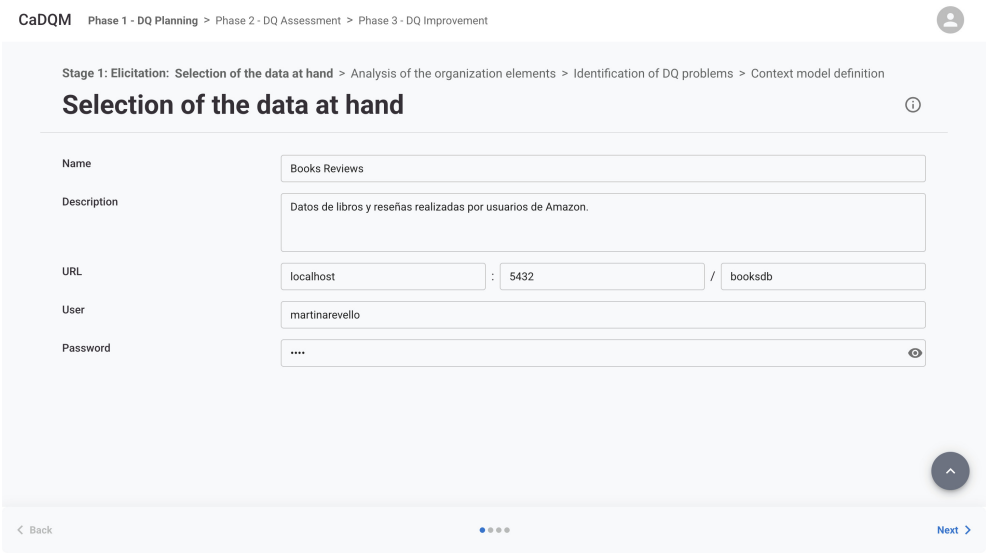

Figura A.5: Actividad 1 - Selección de los datos

Actividad 2 - Análisis de los elementos de la organización: A la derecha de la Figura A.6 se muestra un área de texto donde documentar sistemas, servicios, procesos de negocio, roles de usuario y reglas que impactan el tratamiento de los datos antes seleccionados. A la izquierda de la misma figura, se presenta la posibilidad de agregar documentos (PDF, hojas de cálculo, diagramas, etc), que al subirlos se pueden categorizar por tipo (ej: Metadatos).

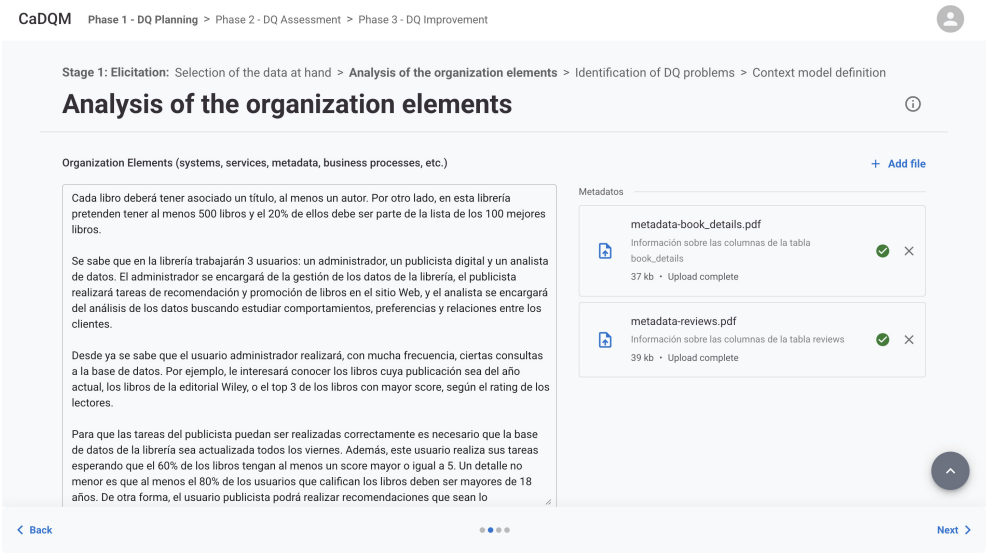

Figura A.6: Actividad 2 - Análisis de los elementos de la organización.

Actividad 3 - Identificación de problemas de CD: Del lado izquierdo de la Figura A.7 se muestran los elementos de la organización cargados en el paso anterior, tanto el texto subido como los archivos que se muestran listados más abajo, para guiar la identificación de problemas de CD. Del lado derecho de la misma figura se puede ver la lista de problemas de CD identificados hasta el momento, donde se pueden editar, eliminar y crear nuevos problemas de CD, utilizando el botón `[+ New]`.

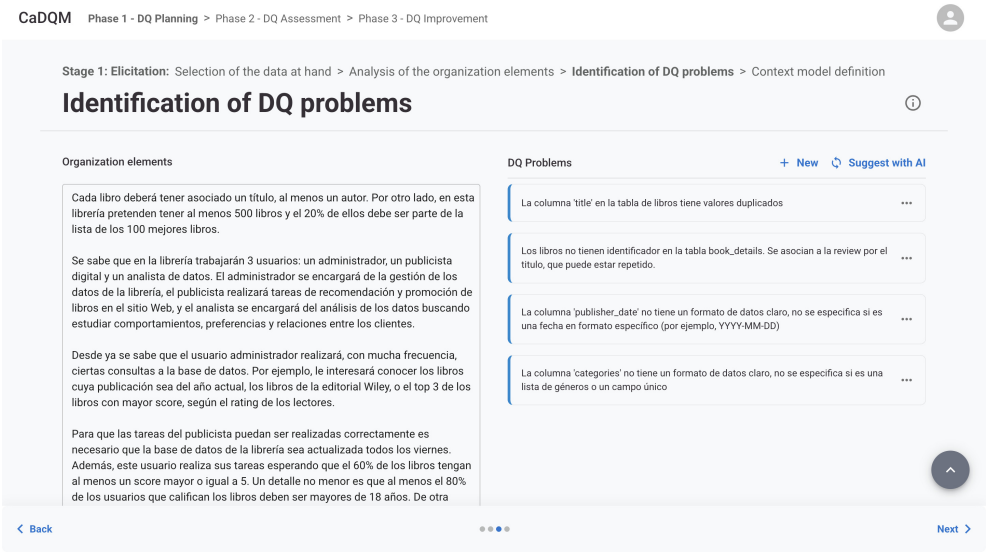

Figura A.7: Actividad 3 - Identificación de problemas de calidad de datos

El botón `[Suggest with AI]` genera, automáticamente, sugerencias basadas en los elementos de la organización, que se muestran como en la Figura A.8. Estas sugerencias pueden ser descartadas, editadas o agregadas al listado de problemas de CD.

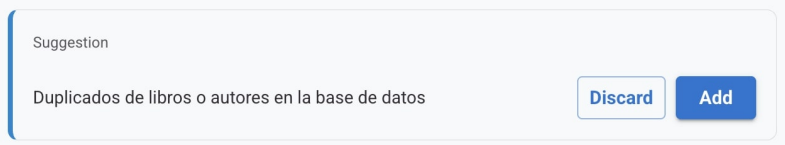

Figura A.8: Sugerencia de problema de CD

Actividad 4 - Definición del modelo de contexto: Del lado izquierdo de la Figura A.9 se muestran los elementos de la organización, como en la actividad anterior. Del lado derecho de la misma figura, se muestran los componentes de contexto separados en las diferentes categorías. Dentro de cada categoría se listan los componentes existentes con su código y sus datos específicos. El botón `[+ New]` abre un diálogo para añadirlos manualmente, mientras que el botón `[Suggest with AI]` propone nuevos componentes de contexto, según los elementos de la organización. Estas sugerencias se pueden cambiar de categoría y editar antes de ser agregadas al listado, también pueden ser descartadas si no fueran relevantes.

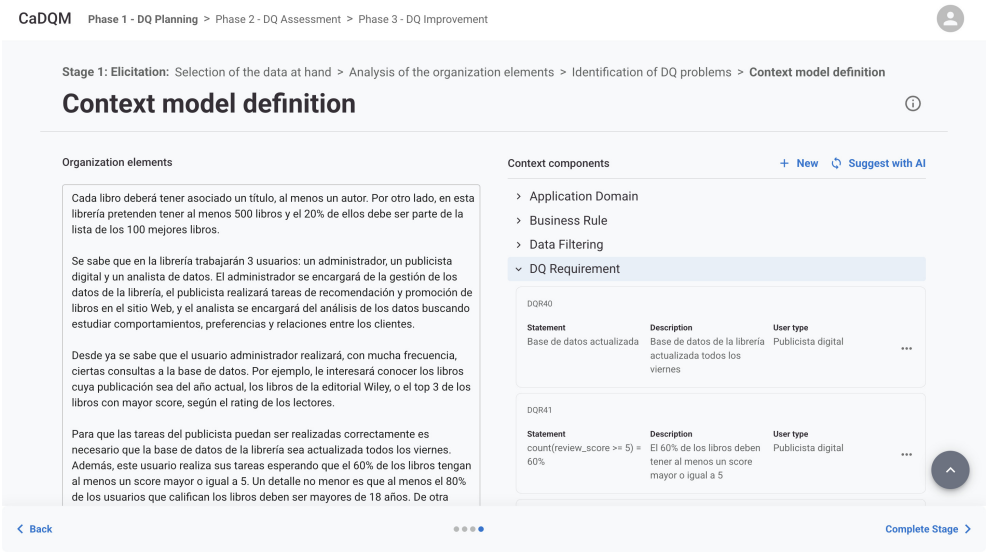

Figura A.9: Actividad 4 - Definición del modelo de contexto

A.2.2. Ejecución de la Etapa 2 - *Data Analysis*

En esta etapa se realiza la actividad de *data profiling*, para entender la estructura y contenido de los datos seleccionados para evaluar su calidad. Además, se identifican nuevos problemas de CD, se estima el nivel de CD general y se actualiza el modelo de contexto. El proceso consta de cuatro actividades.

Actividad 5 - Data profiling: En la Figura A.10 se puede ver, a la izquierda, un diagrama interactivo del esquema de la base de datos, cuya conexión se realizó en la etapa 1. Este diagrama muestra las tablas con sus columnas y tipos, además las relaciones entre ellas. A la derecha de la misma figura, se encuentra un resumen de la base de datos.

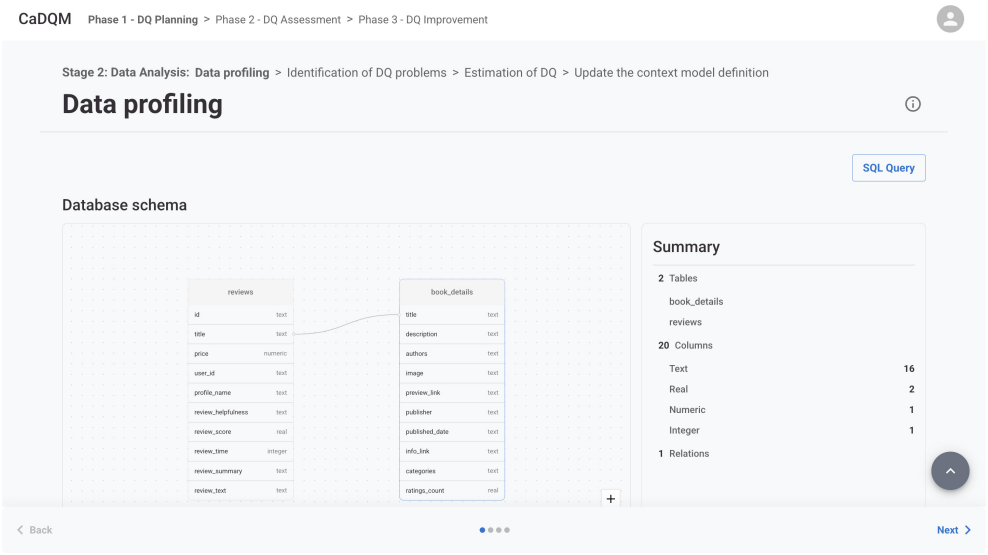

Figura A.10: Actividad 5 - Data profiling. Esquema de la base de datos

Por otro lado, se encuentra el botón `[SQL Query]` que lleva a un diálogo que permite escribir consultas SQL, directamente sobre el esquema, como se ve en la Figura A.11. A su vez, se pueden identificar problemas de CD a partir del resultado de esa consulta, sin tener que avanzar en la actividad, presionando el botón `[Identify DQ Problem]`. Cabe aclarar que en este paso solo se pueden hacer consultas de lectura y no de escritura en la base de datos.

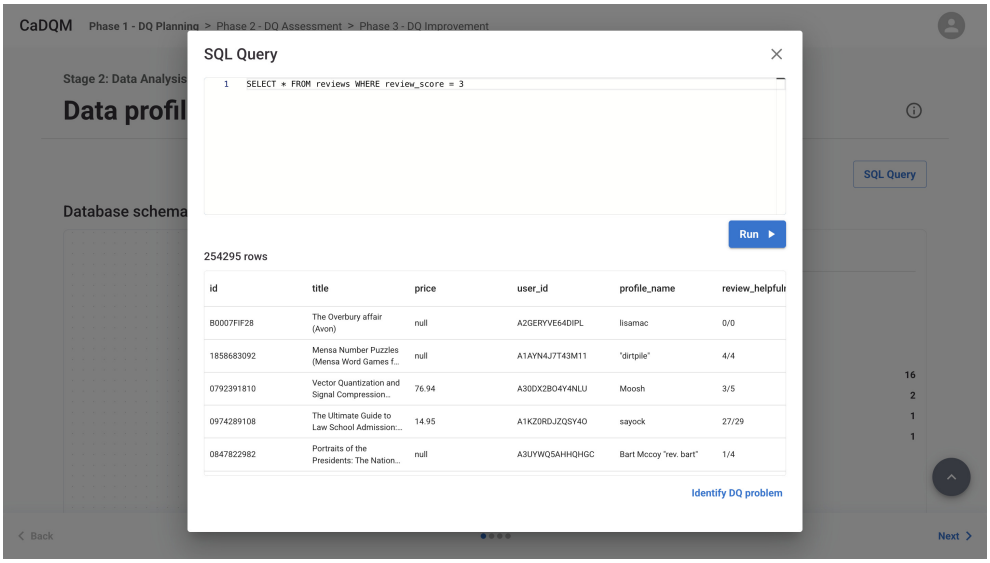

Figura A.11: Actividad 5 - Data profiling. Diálogo de consulta SQL

Al seleccionar una tabla desde el esquema interactivo (ver Figura A.10), en el ejemplo `book_details`, más abajo se cargan los resultados del *data profiling* de esa tabla, que se muestra en dos pestañas:

- **YData profiling** (seleccionada por defecto): muestra el reporte generado con la librería YData.
- **R data profiling**: muestra los resultados del data profiling pero elaborado con herramientas de R.

A su vez, el botón `[Download HTML]`, que se puede ver en la Figura A.12, permite obtener el informe completo en formato HTML para cada una de las herramientas y de las tablas.

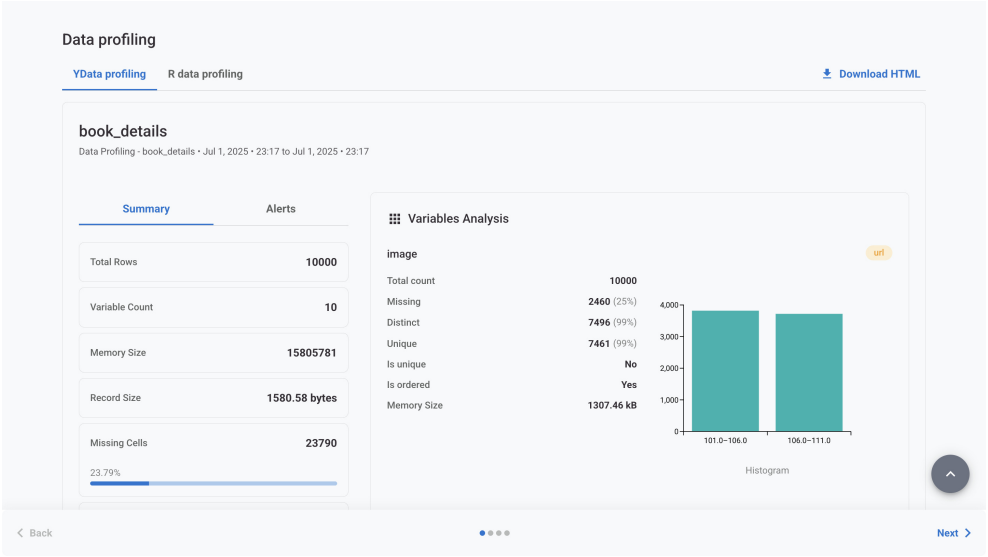

Figura A.12: Actividad 5 - Data profiling

Actividad 3 - Identificación de problemas de CD: La identificación de problemas de CD es similar a la actividad de la etapa anterior: se muestra la lista de problemas como se ve en la Figura A.7, donde se pueden editar, eliminar o agregar nuevos problemas de CD.

Actividad 6 - Estimación de la CD: Para estimar la calidad de los datos, se puede ingresar un texto en el campo de texto, que se muestra en la Figura A.13. Como en otras actividades, el botón `[Suggest with AI]` permite generar sugerencias de manera automática. En este caso, las sugerencias se muestran en forma de "alertas" o "información". Cada una de estas sugerencias puede ser descartada individualmente. El texto ingresado por el usuario se guarda al avanzar en la ejecución.

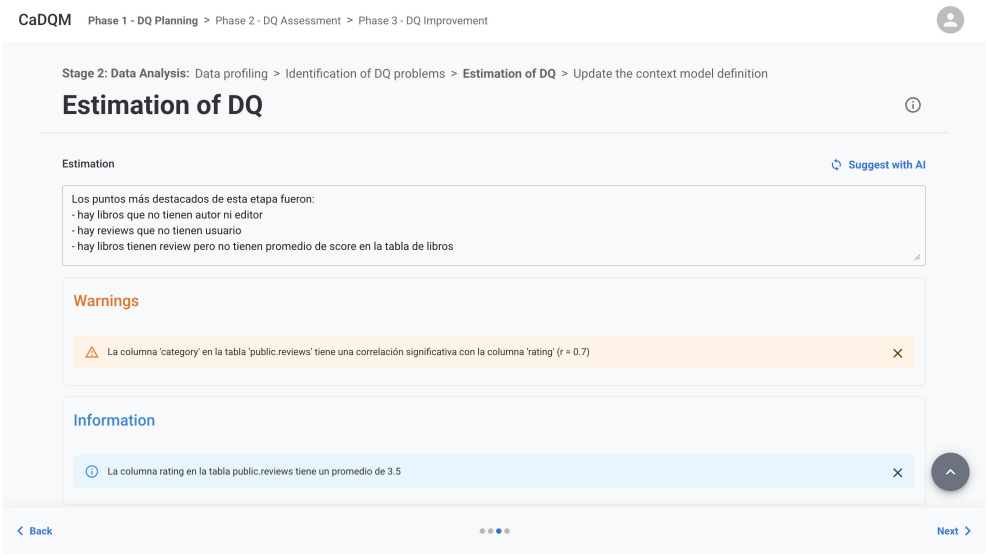

Figura A.13: Actividad 6 - Estimación de la CD

Actividad 7 - Actualización del modelo de contexto: En esta actividad se muestra los componentes de contexto definidos hasta el momento, separado por categorías, de forma similar a la Figura A.9. De la misma forma, es posible editarlos, eliminarlos o agregar nuevos componentes de contexto.

A.2.3. Ejecución de la Etapa 3 - *Interaction with data users*

En la tercera etapa se recopila información directamente de los usuarios de los datos. Se documenta la interacción con ellos, se identifican problemas de CD sugeridos por los usuarios y se actualiza el modelo de contexto. El proceso consta de tres actividades, que se ejecutan de forma similar a las actividades descriptas en la primera etapa.

Actividad 9 - Interacción con los usuarios: Este paso es análogo al análisis de los elementos de la organización de la etapa 1, descriptos en la Sección A.2.1, ya que, de igual forma, se muestra un campo de texto y se le permite al usuario subir archivos y categorizarlos por tipo.

Actividad 3 - Identificación de problemas de CD: En esta actividad, al igual que en la etapa 1, descripta en la Sección A.2.1, se identifican problemas de CD. En este caso, la información que se encuentra en la parte izquierda de la Figura A.7, corresponde a los datos recopilados en la interacción con los usuarios de los datos, realizada en la primera actividad de la etapa 3.

Actividad 7 - Actualización del modelo de contexto: De forma similar a la etapa 1, descripta en la Sección A.2.1, se agregan o actualizan componentes de contexto a partir de la información relevada en la interacción con los usuarios. En este caso, también se pueden considerar sugerencias creadas por la herramienta.

A.2.4. Atajos durante la ejecución de una etapa

Durante toda la ejecución de las actividades de una etapa, está disponible el siguiente botón flotante en la esquina inferior derecha de la pantalla (botón inferior de la Figura A.14).

Figura A.14: Atajos

Al pasar el cursor sobre este botón, se despliegan varias opciones de acceso rápido, como se muestra en la Figura A.14. Estos íconos tienen las siguientes opciones:

- ***View project timeline***: Abre el diálogo con el progreso del proyecto (Figura A.15).

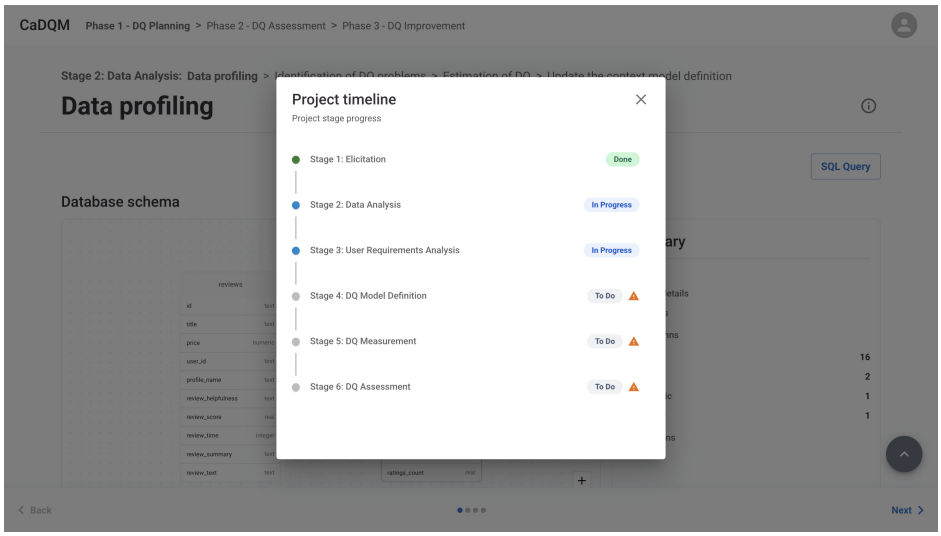

Figura A.15: Diálogo del progreso del proyecto.

- ***View DQ problems***: Abre el diálogo con los problemas de CD (Figura A.16).

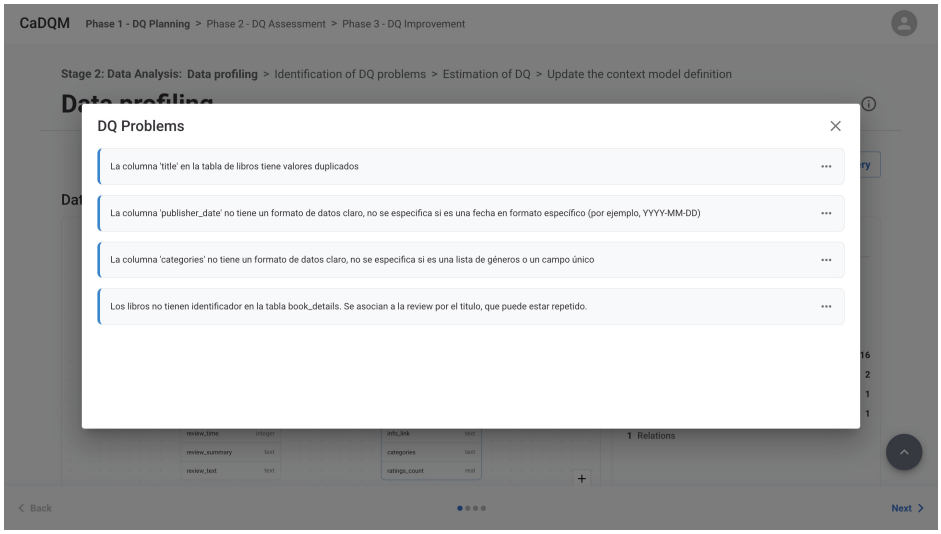

Figura A.16: Diálogo de problemas de CD.

- ***View context model***: Abre el diálogo con los componentes de contexto. Este último se puede abrir en pantalla completa y deja agrupar los componentes de contexto por categoría o por la etapa en fue identificado cada componente de contexto (Figura A.17).

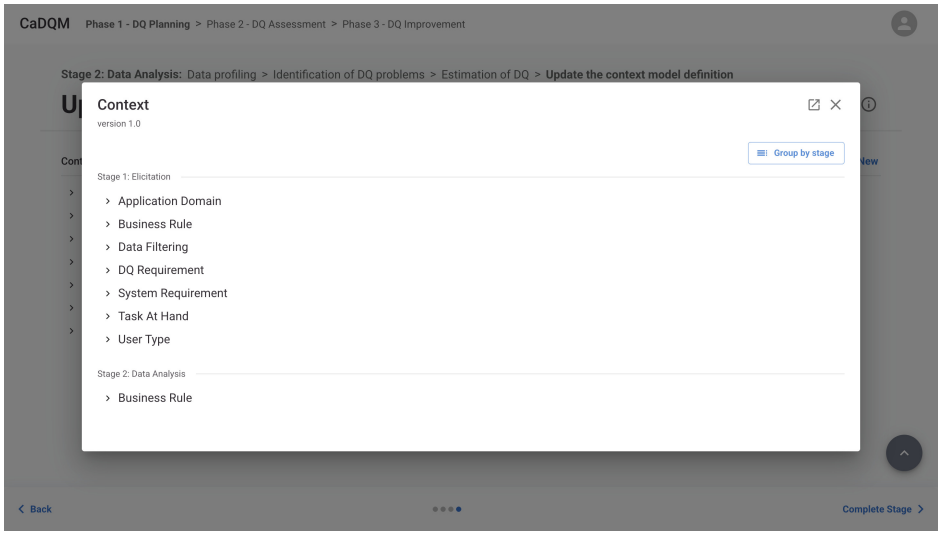

Figura A.17: Diálogo de componentes de contexto.

A.2.5. Completar una etapa

Al completar la última actividad de una etapa, se muestra un diálogo, como se muestra en la Figura A.18, donde se encuentran las siguientes opciones:

- **Generar reporte PDF**: Antes de continuar ejecutando las siguientes etapas, se puede generar un reporte en formato PDF (desde el icono de descargar) que resume la etapa recién completada, así como todas las etapas finalizadas hasta el momento.
- **Continuar a otra etapa**: Permite avanzar a otra etapa habilitada en la metodología. La etapa disponible dependerá de la etapa que acaba de ser completada. Luego de la etapa 1 se puede ir a la etapa 2 o a la etapa 3. Luego de la etapa 2 se puede ir a la etapa 3 y viceversa.
- ***Skip***: Finaliza la etapa actual y redirige al usuario a la pantalla que presenta lista de proyectos, que se encuentra en la Figura A.3.

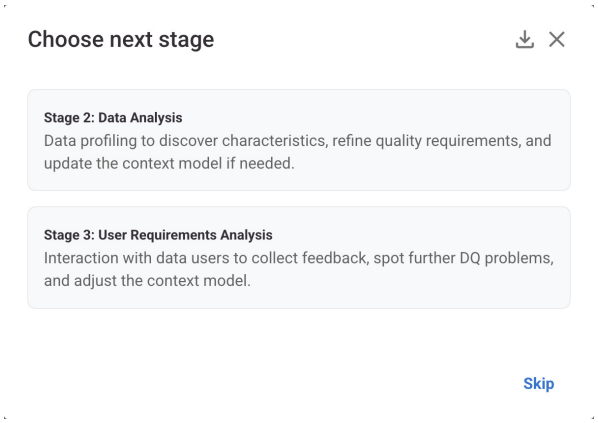

Figura A.18: Diálogo al completar una etapa.

A.2.6. Reporte

Al seleccionar el ícono de descarga para generar un reporte, se abre un diálogo que lista las etapas y actividades completadas hasta el momento, como se muestra en la

Figura A.19. En este diálogo, el usuario puede seleccionar qué elementos desea incluir en el informe y, a partir de la selección, se crea el reporte en PDF con los resultados guardados en cada etapa, categorizado por actividad. Además, tiene la opción de generar un reporte detallado de los problemas de CD y un reporte completo del modelo de contexto. Al presionar `Exportar`, se descarga un archivo PDF con todos los datos recolectados, según las selecciones realizadas por el usuario.

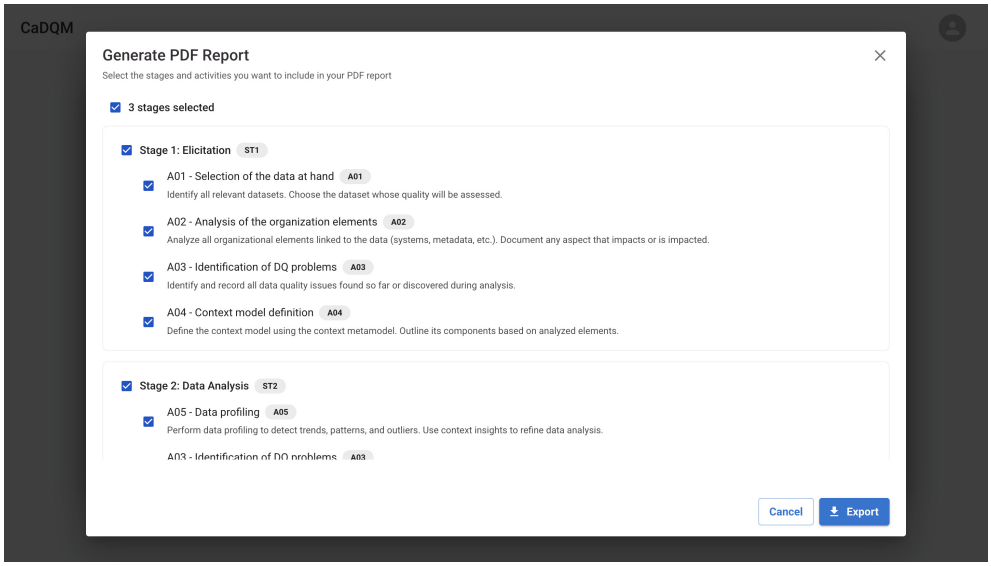

Figura A.19: Diálogo de generación de reportes PDF.

El reporte también puede ser obtenido desde el detalle del proyecto, en todo momento, seleccionando el botón de descarga que se ve en la esquina superior derecha del diálogo de la Figura A.4.

Anexo B

Resultados de la Validación

En este anexo se presentan las fuentes de datos, los documentos utilizados en la ejecución y los resultados obtenidos en la experimentación de los dos casos de estudio realizados.

B.1. Resultados de la Validación del Caso de Estudio 1: Funcionalidad

En esta sección se presenta el contenido de los documentos con elementos del dominio, utilizados en la Actividad 2 - Análisis de los elementos de la organización, de la Etapa 1 - *Elicitation*. Además, se describen los resultados obtenidos en la experimentación del Caso de Estudio 1, presentado en la Sección 5.1.

B.1.1. Descripción de la realidad

Para la Actividad 2 - Análisis de los elementos de la organización, de la Etapa 1 - *Elicitation*, se cargó en la herramienta información sobre el dominio (en formato texto y archivos) el cual se describe a continuación.

En primer lugar, se utilizó el siguiente texto con información sobre el dominio:

> *Cada libro deberá tener asociado un título, al menos un autor y un editor. Por otro lado, en esta librería pretenden tener al menos 500 libros y el 20 % de ellos debe ser parte de la lista de los 100 mejores libros.*
>
> *Se sabe que en la librería trabajarán 3 usuarios: un administrador, un publicista digital y un analista de datos. El administrador se encargará de la gestión de los datos de la librería, el publicista realizará tareas de recomendación y promoción de libros en el sitio Web, y el analista se encargará del análisis de los datos buscando estudiar comportamientos, preferencias y relaciones entre los clientes.*
>
> *Desde ya se sabe que el usuario administrador realizará, con mucha frecuencia, ciertas consultas a la base de datos. Por ejemplo, le interesará conocer los libros cuya publicación sea del año actual, los libros de la editorial Wiley, o el top 3 de los libros con mayor score, según el rating de los lectores.*
>
> *Para que las tareas del publicista puedan ser realizadas correctamente es necesario que la base de datos de la librería sea actualizada todos los viernes. Además, este usuario realiza sus tareas esperando que el 60 % de los libros tengan al menos un score mayor o igual a 5. De otra forma, el usuario publicista podrá realizar recomendaciones que sean lo suficientemente representativas.*

Por otro lado, para la efectividad de las tareas del analista de datos se especifica que al menos el 95 % de los libros debe tener título, que deben estar correctamente escritos y para los nombres de los autores debe aparecer al menos un nombre y un apellido.

Finalmente, se destaca que los tiempos de respuesta del sitio Web de la librería no puede superar los 3 segundos.

En segundo lugar, se cargó en la herramienta dos archivos con metadatos sobre las tablas. La información de estos archivos se encuentra en las Tablas B.1 para *book_details* y B.2 para *reviews*.

Columna	**Descripción**
title	título del libro
description	descripción del libro
authors	nombre de los autores
image	url de la tapa del libro
preview_link	link para acceder a este libro en google books
publisher	nombre del publicador
published_date	fecha de publicación
info_link	link para obtener más información sobre el libro en google books
categories	géneros del libro
raitings_count	promedio del puntaje del libro

Tabla B.1: Metadatos de la tabla *book_details*.

Columna	**Descripción**
id	id del libro
title	título del libro
price	precio del libro
user_id	id del usuario que puntuó el libro
profile_name	nombre del usuario que reseñó el libro
review_helpfulness	calificación de utilidad de la reseña ej: 2/3
review_score	puntuación del libro, rango de 0 a 5
review_time	fecha de la reseña (timestamp)
review_summary	resumen de la reseña
review_text	texto de la reseña

Tabla B.2: Metadatos de la tabla *reviews*.

B.1.2. Resultados

Los reportes generados por la herramienta se encuentran a continuación. En primer lugar se encuentra el reporte con los datos recopilados en cada etapa, en las figuras B.1, B.2, B.3, B.4 y B.5 y en segundo lugar el reporte con todos los problemas de CD identificados durante la ejecución y el modelo de contexto completo, en las figuras B.6, B.7 y B.8.

Project: Libros

Gestión de calidad de datos de libros y reseñas

July 25, 2025

Context version: 1.0

In progress

Stage 1: Elicitation done

A01 - Selection of the data at hand

Data at hand: Books Reviews - Datos de libros y reseñas realizadas por usuarios de Amazon.

Database url: **booksdb**

A02 - Analysis of the organization elements

Cada libro deberá tener asociado un título, al menos un autor y un editor. Por otro lado, en esta librería pretenden tener al menos 500 libros y el 20% de ellos debe ser parte de la lista de los 100 mejores libros.

Se sabe que en la librería trabajarán 3 usuarios: un administrador, un publicista digital y un analista de datos. El administrador se encargará de la gestión de los datos de la librería, el publicista realizará tareas de recomendación y promoción de libros en el sitio Web, y el analista se encargará del análisis de los datos buscando estudiar comportamientos, preferencias y relaciones entre los clientes.

Desde ya se sabe que el usuario administrador realizará, con mucha frecuencia, ciertas consultas a la base de datos. Por ejemplo, le interesará conocer los libros cuya publicación sea del año actual, los libros de la editorial Wiley, o el top 3 de los libros con mayor score, según el rating de los lectores.

Para que las tareas del publicista puedan ser realizadas correctamente es necesario que la base de datos de la librería sea actualizada todos los viernes. Además, este usuario realiza sus tareas esperando que el 60% de los libros tengan al menos un score mayor o igual a 5. De otra forma, el usuario publicista podrá realizar recomendaciones que sean lo suficientemente representativas.

Por otro lado, para la efectividad de las tareas del analista de datos se especifica que al menos el 95% de los libros debe tener título, que deben estar correctamente escritos y para los nombres de los autores debe aparecer al menos un nombre y un apellido.

Finalmente, se destaca que los tiempos de respuesta del sitio Web de la librería no puede superar los 3 segundos.

Files

- metadata-book_details_FIktVOL.pdf - Información sobre las columnas de la tabla book_details
- metadata-reviews_DcqaYMS.pdf - Información sobre las columnas de la tabla reviews

A03 - Identification of DQ problems

- El campo 'categories' puede tener valores fuera de un conjunto predefinido de géneros

Figura B.1: Reporte de libros - Página 1

- El campo 'title' puede contener títulos duplicados o inconsistentes
- El campo 'preview_link' y 'info_link' pueden contener enlaces caducados o inválidos
- Duplicados de reseñas si no hay restricción de unicidad en la combinación de user_id y id del libro

A04 - Context model definition

Application Domain

AD33	
Description	libros

Business Rule

BR34	
Statement	count(libros) >= 500
Semantic	Tener al menos 500 libros.
BR35	
Statement	book_details.title != null && book_details.author != null && book_details.publisher != null
Semantic	Cada libro deberá tener asociado un título, al menos un autor y un editor.

Data Filtering

DF42	
Statement	publisher_date.year == now().year
Description	Filtrar los libros cuya publicación sea del año actual
Task At Hand	Gestión de los datos
DF43	
Statement	publisher == 'Wiley'
Description	Libros de la editorial Wiley
Task At Hand	Gestión de los datos
DF44	
Statement	order by review_score limit 3
Description	Top 3 de los libros con mayor score
Task At Hand	Gestión de los datos

DQ Requirement

DQR45	

Figura B.2: Reporte de libros - Página 2

Statement	Base de datos actualizada
Description	Base de datos de la librería actualizada todos los viernes
User Type	Publicista digital
DQR46	
Statement	count(review_score >= 5) = 60%
Description	El 60% de los libros deben tener al menos un score mayor o igual a 5
User Type	Publicista digital
DQR47	
Statement	Títulos correctamente escritos
Description	Los títulos de los libros deben estar correctamente escritos.
User Type	Analista de datos
DQR48	
Statement	Nombres completos de los autores
Description	Los nombres de los autores debe tener al menos un nombre y un apellido
User Type	Analista de datos
DQR49	
Statement	count(book.title != null) >= 95%
Description	Al menos el 95% de los libros debe tener título
User Type	Analista de datos

System Requirement

SR50	
Statement	Tiempos de respuesta
Description	Los tiempos de respuesta del sitio Web de la librería no puede superar los 3 segundos

Task At Hand

T39	
Name	Gestión de los datos
Purpose	Gestión de los datos de la librería
T40	
Name	Recomendación y promoción de libros
Purpose	Recomendación y promoción de libros en el sitio Web

Figura B.3: Reporte de libros - Página 3

T41	
Name	Análisis de los datos
Purpose	Análisis de los datos buscando estudiar comportamientos, preferencias y relaciones entre los clientes

User Type

UT36	
Name	Administrador
Characteristics	Administrador de la librería
UT37	
Name	Publicista digital
Characteristics	Publicista digital
UT38	
Name	Analista de datos
Characteristics	Analista de los datos de la librería

Stage 2: Data Analysis done

A05 - Data profiling

2 Tables

- book_details - book_details-Y.html
- reviews - reviews-Y.html

20 Columns

- 16 Text
- 2 Real
- 1 Numeric
- 1 Integer

A03 - Identification of DQ problems

- El 15% de los libros no tienen autor, violando la regla de negocio BR35.
- El 38% de los libros no tienen editor (publisher), violando la regla de negocio BR35.
- El 76,58% de los libros tienen review pero no tienen promedio de score en la tabla de libros (reviews_count).
- El 19% de las reviews no tienen usuario.

Figura B.4: Reporte de libros - Página 4

A06 - Estimation of DQ

El 15 % de los libros no tienen autor y el 38 % no tienen editor, violando reglas de negocio. Además, el 19 % de las reseñas no tienen un usuario asociado. Hay una inconsistencia, ya que el 76.58 % de los libros con reseñas no tienen actualizado su contador de calificaciones (reviews count), lo que indica un problema en la integridad de los datos entre las tablas.

Warnings

- La columna user_id en la tabla public.reviews tiene 1937 (19.4%) valores faltantes.
- La columna authors en la tabla public.book_details tiene 1481 (14.8%) valores faltantes.
- La columna publisher en la tabla public.book_details tiene 3581 (35.8%) valores faltantes.
- La columna ratings_count en la tabla public.book_details tiene 7659 (76.6%) valores faltantes.

A07 - Update the context model definition

There is no data for this activity

Figura B.5: Reporte de libros - Página 5

Project: Libros

Gestión de calidad de datos de libros y reseñas

July 25, 2025

Context version: 1.0

In progress

Data Quality Problems

- El 15% de los libros no tienen autor, violando la regla de negocio BR35.
- El 38% de los libros no tienen editor (publisher), violando la regla de negocio BR35.
- El 76,58% de los libros tienen review pero no tienen promedio de score en la tabla de libros (reviews_count).
- El 19% de las reviews no tienen usuario.
- El campo 'categories' puede tener valores fuera de un conjunto predefinido de géneros
- El campo 'title' puede contener títulos duplicados o inconsistentes
- El campo 'preview_link' y 'info_link' pueden contener enlaces caducados o inválidos
- Duplicados de reseñas si no hay restricción de unicidad en la combinación de user_id y id del libro

Context Model

Version 1.0

Application Domain

AD33	
Description	libros

Business Rule

BR34	
Statement	count(libros) >= 500
Semantic	Tener al menos 500 libros.
BR35	
Statement	book_details.title != null && book_details.author != null && book_details.publisher != null
Semantic	Cada libro deberá tener asociado un título, al menos un autor y un editor.

Data Filtering

DF42	
Statement	publisher_date.year == now().year
Description	Filtrar los libros cuya publicación sea del año actual
Task At Hand	Gestión de los datos
DF43	

Figura B.6: Reporte de libros completo - Página 1

Statement	publisher == 'Wiley'
Description	Libros de la editorial Wiley
Task At Hand	Gestión de los datos
DF44	
Statement	order by review_score limit 3
Description	Top 3 de los libros con mayor score
Task At Hand	Gestión de los datos

DQ Requirement

DQR45	
Statement	Base de datos actualizada
Description	Base de datos de la librería actualizada todos los viernes
User Type	Publicista digital
DQR46	
Statement	count(review_score >= 5) = 60%
Description	El 60% de los libros deben tener al menos un score mayor o igual a 5
User Type	Publicista digital
DQR47	
Statement	Títulos correctamente escritos
Description	Los títulos de los libros deben estar correctamente escritos.
User Type	Analista de datos
DQR48	
Statement	Nombres completos de los autores
Description	Los nombres de los autores debe tener al menos un nombre y un apellido
User Type	Analista de datos
DQR49	
Statement	count(book.title != null) >= 95%
Description	Al menos el 95% de los libros debe tener título
User Type	Analista de datos

System Requirement

SR50	

Figura B.7: Reporte de libros completo - Página 2

Statement	Tiempos de respuesta
Description	Los tiempos de respuesta del sitio Web de la librería no puede superar los 3 segundos

Task At Hand

T39	
Name	Gestión de los datos
Purpose	Gestión de los datos de la librería
T40	
Name	Recomendación y promoción de libros
Purpose	Recomendación y promoción de libros en el sitio Web
T41	
Name	Análisis de los datos
Purpose	Análisis de los datos buscando estudiar comportamientos, preferencias y relaciones entre los clientes

User Type

UT36	
Name	Administrador
Characteristics	Administrador de la librería
UT37	
Name	Publicista digital
Characteristics	Publicista digital
UT38	
Name	Analista de datos
Characteristics	Analista de los datos de la librería

Figura B.8: Reporte de libros completo - Página 3

B.2. Resultados de la Validación del Caso de Estudio 2: Interoperabilidad y Análisis

En esta sección se presenta la fuente de datos y los resultados obtenidos en la experimentación del Caso de Estudio 2: Verificación de Interoperabilidad y Análisis del modelo de contexto obtenido, presentado en la Sección 5.2, incluyendo la comparación entre el modelo de contexto generado con la herramienta y el modelo de contexto de referencia, y los componentes de contexto adicionales identificados.

B.2.1. Fuente de datos de medicamentos

El *dataset* se almacena en un archivo CSV que contiene las columnas que se presentan en la Tabla B.3.

Columna	**Descripción**
Fecha	Fecha de dosificación del antibiótico (concentración de un fármaco en un fluido biológico en un momento dado).
Registro	Número de registro que identifica al paciente.
ATB	Antibiótico suministrado: Amikacina (Amika), Vancomicina (Vanco) o Gentamicina (Genta).
Posología	Dosis en la que se administran los medicamentos.
Vía	Vía de administración de la droga.
Día.Últ.Dosis	Día de la última dosis de antibiótico.
RazónTrat	Razón del tratamiento (texto libre).
Estado	Estado clínico del paciente.
IR	Indica si el paciente presenta insuficiencia renal.
Crea	Valor de Creatinina.
Diálisis	Indica si el paciente está en diálisis.
Conc.Valle	Concentración en valle (Cmin) - concentración plasmática mínima alcanzada instantes previos a la siguiente dosis.
Conc.Pico	Concentración en pico (Cmáx) - concentración plasmática máxima alcanzada.
Conc.Cont	Concentración continua.
Conc.PreHD	Concentración antes de hemodiálisis.
Conc.PostHD	Concentración después de hemodiálisis.
Conc.LCR	Concentración registrada solo cuando el paciente recibe Vancomicina y el fluido biológico es LCR.
Conc	Concentración cuando se desconoce el momento de extracción.
Comentarios	Comentarios realizados al momento de la dosificación (texto libre; puede estar vacío).

Tabla B.3: Descripción de las columnas del *dataset* usado en el caso de estudio 2 (ver Sección 5.2).

B.2.2. Resultados

En la Tabla B.4 se presenta la comparación del modelo de contexto obtenido utilizando la herramienta propuesta, con el modelo de contexto de referencia, definido

en [1]. En verde y resaltado en negrita se muestran los componentes de contexto presentes en ambos modelos de contexto, y en rojo, los componentes de contexto que no fueron identificados usando la herramienta. Por otro lado, en amarillo se muestran los componentes de contexto identificados utilizando la herramienta que coinciden con algún componente del modelo de contexto de referencia, pero que presentan alguna diferencia en su enunciado.

Categoría	**Componente de contexto**
Application domain	**AD: Health (Farmacología, Monitoreo terapéutico)**
Business rules	BR1: ATB = "amika" (Amikacina) or "vanco" (Vancomicina) or "genta" (Gentamicina)
	BR2: Fluido biológico = "Plasma" or "Líquido Cefalorraquídeo" (LCR)
	BR3: If notNull(Conc.LCR) → Fluido biológico = "LCR"
	BR4: Registros duplicados representan al mismo paciente
	BR5: If "BIC" in Posología → ATB = "vanco"
	BR6: via = "VO", "IV", "bolsa peritoneal", "intraperitoneal", "intraventricular", "intratecal", "intramuscular"
	BR7: If via = "VO" → ATB = "vanco"
	BR8: If notNull(Conc.LCR) → ATB = "vanco"
	BR9: Día.Últ.Dosis ≤ fecha
	BR10: Distance(Día.Últ.Dosis, fecha) ≤ 1 week
	BR11: If (fecha - Día.Últ.Dosis) > 3 → Posología = "Suspendida"
	BR12: diálisis = "hemodiálisis" ↔ (IR = "Si" or IR = "En HD") or (Crea ≥ 1.2 mg/dL)
	BR13: 0,17 mg/dL ≤ Crea ≤ 20 mg/dL
	BR14: Crea ≥ 1.2 mg/dL ↔ IR = "Si"
	BR15: If notNull(Conc.Cont) → "BIC" in Posología
	BR16: If notNull(Conc.preHD) → Diálisis = "hemodiálisis" and IR = "si" OR IR = "en HD"
	BR17: If notNull(Conc.postHD) → Diálisis = "hemodiálisis" and IR = "si" OR IR = "en HD"
	BR18: if diálisis = "diálisis peritoneal" → Conc.PreHD = NULL and Conc.PosHD = NULL
	BR19: dosis de vanco: múltiplos de 250 mg
	BR20: dosis de amika: múltiplos de 100 mg o 250 mg
	BR21: dosis de genta: múltiplos de 20 mg
	BR22: Todas las concentraciones se miden en mg/L
Users characteristics	**UC1: docentes**
	UC2: estudiantes
	UC3: médicos
Tasks at hand	**T1: registro de datos**
	T2: análisis de datos para investigación

Categoría	Componente de contexto
	T3: análisis de la evolución estadística del servicio (n° de dosificaciones, calidad de la info recibida)
	T4: Proveer datos
Data filtering needs	**DF1: Interesa consultar si un paciente tiene dosificaciones previas**
	DF2: Interesa consultar cuál es la proporción de datos que se encuentran en la columna Conc.
	DF3: Para Via= "VO" and RazónTrat = "clostridium", interesa consultar cuántos pacientes presentan concentración.
DQ requirements	**DQR1: el atributo fecha tiene formato DD.MM.YY.**
	DQR2: el atributo Día.Últ.Dosis tiene formato DD.MM.YY.
	DQR3: 100 % de los registros debe tener el campo fecha.
	DQR4: Para que los análisis realizados sean representativos se exige al menos el 50 % de cada variable.
	DQR5: Si un mismo paciente tiene 2 registros o más que indican "diálisis peritoneal" y "hemodiálisis", entonces en el campo comentarios debe estar aclarado que esto es válido.
	DQR6: If IR = "En HD" → Crea not NULL
	DQR7: If notNull(Conc.preHD) → Crea ≥ 1.2 mg/dL
	DQR8: If notNull(Conc.postHD) → Crea ≥ 1.2 mg/dL
	DQR9: Cuando el texto "error de extracción" aparece en el campo comentarios y el texto "hora de extracción" no aparece en el campo comentarios el registro no es válido.
	DQR10: Registros válidos tienen un valor no nulo en uno, y solo en uno, de los siguientes atributos: Conc.Valle, Conc.Pico, Conc.Cont, Conc.PreHD, Conc.PostHD, Conc.LCR or Conc.
	DQR11: If ATB = "vanco" and "<3.0" in comentarios → (Conc.Valle= "3.0") or (Conc.Pico = "3.0") or (Conc.Cont = "3.0") or (Conc.PreHD = "3.0") or (Conc.PostHD = "3.0") or (Conc.LCR = "3.0") or (Conc = "3.0")
	DQR12: If ATB = "amika" and "<2.3" in comentarios → (Conc.Valle = "2.3") or (Conc.Pico = "2.3") or (Conc.Cont = "2.3") or (Conc.PreHD = "2.3") or (Conc.PostHD = "2.3") or (Conc.LCR = "2.3") or (Conc = "2.3")
	DQR13: If ATB = "genta" and "<0.3" in comentarios → (Conc.Valle = "0.3") or (Conc.Pico = "0.3") or (Conc.Cont = "0.3") or (Conc.PreHD = "0.3") or (Conc.PostHD = "0.3") or (Conc.LCR = "0.3") or (Conc = "0.3")

Tabla B.4: Resultados de la comparación realizada entre el modelo de contexto obtenido con la herramienta y el modelo de contexto de referencia. En verde, componentes de contexto presentes en ambos modelos de contexto, en rojo, componentes de contexto no identificados con la herramienta y en amarillo componentes de contexto presentes en ambos modelos, pero que presentan alguna diferencia.

Finalmente, en la Tabla B.5 se presentan los componentes de contexto identificados utilizando la herramienta, pero que no son parte del modelo de contexto de referencia.

Categoría	**Componente de contexto**
Business rules	BR18: If (IR = Si or IR = En HD) and Crea null → diálisis not null
	BR23: Conc.LCR $\leq$ 10mg/L
System requirements	SR11: Espacio de almacenamiento de datos.
Data filtering needs	DF8: Filtrar por fecha
	DF10: Datos que se encuentran en la columna Conc.

Tabla B.5: Componentes de contexto identificados con la herramienta, que no se encuentran en el modelo de contexto de referencia.

Anexo C

Prompts utilizados en la integración con IA

En esta sección se listaran los diferentes *prompts* utilizados al realizar diferentes peticiones utilizando la herramienta de IA.

C.1. *Prompt* para el análisis de archivos y sugerencias de problemas de calidad de datos

Este *Prompt* se utiliza para el análisis de archivos cargados por el usuario, identificando problemas de calidad de datos mediante comparación con documentación de referencia cuando está disponible.

```
Actúa como un experto en calidad de datos encargado de identificar
posiblesproblemas de calidad en un dataset ubicado en una base de
datos relacional.
La información provista consiste únicamente en archivos de
documentación y texto que describen la estructura esperada, las reglas
de negocio, restricciones y requisitos del dataset.
**No tienes acceso a los datos reales.** Tu análisis debe basarse
únicamente en la documentación para inferir qué problemas de calidad
podrían existir al comparar la estructura y reglas documentadas con
escenarios comunes en la práctica.
Los problemas de calidad de datos pueden incluir, entre otros:
- Datos incorrectos o inconsistentes que no reflejen la realidad
- Información incompleta o campos vacíos
- Datos desactualizados
- Valores fuera de rango o con formatos inválidos
- Duplicados

**Instrucciones de salida:**
- Responde siempre en español.
- Devuelve únicamente una lista JSON de strings, donde cada string
describe un problema de calidad específico, derivado del análisis
de la documentación.
- No incluyas explicaciones, encabezados ni texto adicional fuera del JSON.
```

C.2. *Prompt* para el análisis y sugerencias de los componentes de contexto

Este *prompt* extrae información relevante para poder recomendar diferentes componentes de contexto.

```
Actúa como un experto en calidad de datos encargado en análisis de
contexto de datos de un dataset ubicado en una base de datos relacional.
La información provista consiste únicamente en archivos de
documentación y texto que describen la estructura esperada, las reglas
de negocio, restricciones y requisitos del dataset.
*No tienes acceso a los datos reales. Tu análisis debe basarse únicamente
en la documentación para inferir qué componentes de contexto podrían
definirse.

Las categorías de componentes de contexto a analizar son:
1. Application domain: Dominio de aplicación y contexto general
2. User types: Tipos de usuarios y sus características
3. Task: Tareas y objetivos principales
4. Data filtering requirements: Requisitos de filtrado de datos
5. Data quality requirements: Requisitos de calidad de datos
6. System requirements: Requisitos del sistema
7. Business rules: Reglas de negocio

Instrucciones de salida:
- Responde siempre en español.
- Devuelve un objeto JSON con estas claves, donde cada valor es una
lista de strings con la información relevante encontrada.
- Si no encuentras información para alguna categoría, devuelve una
lista vacía para esa clave.
- No incluyas explicaciones, encabezados ni texto adicional fuera
del JSON.
```

C.3. *Prompt* para el análisis de *data profiling* y sugerencias de la estimación de la calidad de los datos

Este *prompt* analiza los resultados del *data profiling* y genera advertencias y hechos sobre el estado de los datos analizados.

```
Eres un experto en análisis de calidad de datos. A partir del
siguiente resultado de data profiling, genera un análisis detallado
en formato JSON ESTRICTO. IMPORTANTE: Responde SIEMPRE en español.

La respuesta debe tener dos claves principales y dentro las mismas
deben ser en español:

1. 'warnings': una lista de advertencias claras y específicas sobre
problemas potenciales en los datos, como duplicados, valores nulos
o correlaciones sospechosas.

2. 'facts': una lista de oraciones claras en español que describan
de forma objetiva el estado de los datos. No uses estructuras JSON
crudas dentro de facts, sino frases completas como:
 - 'La tabla public.reviews contiene 1000 filas y 3 columnas.'
```

- 'La columna price en la tabla public.reviews tiene 21 valores únicos.'

NO incluyas encabezados, introducciones ni explicaciones adicionales.
La salida debe ser un JSON válido. Quiero que seas lo más extenso
posible en devolver tu respuesta y por favor analiza todos los datos.