

213003

**ARQUITECTURA DE REFERENCIA Y PRUEBA DE CONCEPTO PARA REALIZAR
ANALÍTICA DE HISTORIAS CLÍNICAS EN EL CONTEXTO DEL DIAGNÓSTICO
DE APNEA DEL SUEÑO**

Héctor Gerardo Castillo Rodríguez
Eyner Fabian Arias Triana

PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERIA
MAESTRÍA EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
BOGOTÁ, D.C.
2021

213003

ARQUITECTURA DE REFERENCIA Y PRUEBA DE CONCEPTO
PARA REALIZAR ANALÍTICA DE HISTORIAS CLÍNICAS EN EL
CONTEXTO DEL DIAGNÓSTICO DE APNEA DEL SUEÑO

Autores:

Héctor Gerardo Castillo Rodríguez
Eyner Fabian Arias Triana

MEMORIA DEL TRABAJO DE GRADO REALIZADO PARA CUMPLIR UNO
DE LOS REQUISITOS PARA OPTAR AL TÍTULO DE
MAESTRIA EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

Director

Rafael Andrés González Rivera, Ph.D.

Comité de Evaluación del Trabajo de Grado

Alexandra Pomares Quimbaya, Ph.D.

Darío Ernesto Correal Torres, Ph.D.

Página web del Trabajo de Grado

<https://livejaverianaedu.sharepoint.com/sites/Ingsis/TGMISC/213003>

PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERIA
MAESTRÍA EN INGENIERIA DE SISTEMAS Y COMPUTACIÓN
BOGOTÁ, D.C.
Noviembre, 2021

**PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERIA
MAESTRÍA EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN**

Rector Magnífico

Jorge Humberto Peláez, S.J.

Decano Facultad de Ingeniería

Ingeniero Lope Hugo Barrero Solano

Director Maestría en Ingeniería de Sistemas y Computación

Ángela Cristina Carrillo Ramos, Ph.D.

Director Departamento de Ingeniería de Sistemas

Ing. César Julio Bustacara Medina, Ph.D.

Artículo 23 de la Resolución No. 1 de Junio de 1946

“La Universidad no se hace responsable de los conceptos emitidos por sus alumnos en sus proyectos de grado. Sólo velará porque no se publique nada contrario al dogma y la moral católica y porque no contengan ataques o polémicas puramente personales. Antes bien, que se vean en ellos el anhelo de buscar la verdad y la Justicia”

AGRADECIMIENTOS

Eyner Arias

Antes que todo le doy gracias a Dios por habernos dado la posibilidad de vivir la experiencia de cursar esta maestría en la Pontificia Universidad Javeriana.

Le agradezco mucho a mis padres y hermana por brindarme ese apoyo incondicional y por enseñarme a tener el compromiso con la educación constante.

Agradezco también a Hector Castillo por realizar este trabajo de grado conmigo; gracias al director Rafael González por su orientación y consejos los cuales fueron muy útiles para el presente trabajo de grado y también le agradezco a Juan Pablo Pájaro por el apoyo y las ideas brindadas.

Héctor Castillo

Le agradezco a mi esposa Laura por apoyarme en cada paso que doy, por darme aliento en los momentos difíciles y sobre todo por su inmenso amor que llena mi corazón cada día.

Agradezco a mis padres y mis hermanos por su apoyo a la distancia, y por enseñarme que se debe mejorar un poco día a día.

Agradezco a mi compañero de trabajo Eyner Arias, por tomar juntos este reto, a nuestro director Rafael González por guiarnos en cada momento y darnos claridad en los momentos difíciles, también agradezco a Juan Pajaro por todo su apoyo y paciencia.

Contenido

INTRODUCCIÓN.....	11
1. DESCRIPCIÓN GENERAL.....	13
OPORTUNIDAD Y PROBLEMÁTICA.....	13
2. DESCRIPCIÓN DEL PROYECTO.....	15
2.1. OBJETIVO GENERAL	15
2.2. OBJETIVOS ESPECÍFICOS	15
2.3. FASES DE DESARROLLO	15
2.3.1 Exploración de datos actuales.....	16
2.3.2 Arquitectura de referencia	16
2.3.3 Prueba de concepto	16
2.3.4 Validación de la Prueba de concepto.....	16
3. MARCO TEÓRICO / ESTADO DEL ARTE.....	17
3.1. GRANDES VOLÚMENES DE DATOS EN EL ÁREA DE LA SALUD.....	17
3.2. ARQUITECTURAS DE BIG DATA EN EL ÁREA DE LA SALUD	17
3.3. BIG DATA	18
3.3.1. Volumen	18
3.3.2. Variabilidad.....	18
3.3.3. Velocidad	19
3.3.4. Valor	19
3.3.5. Veracidad.....	19
3.4. ARQUITECTURA DE REFERENCIA DE BIG DATA Y ANALÍTICA	19
3.4.1. Por qué una Arquitectura de Referencia	20
3.4.2. Cómo Hacer una Arquitectura de Referencia.....	20
3.5. FLUJOS DE TRABAJO EN ANALÍTICA DE DATOS	23
4. TRABAJOS RELACIONADOS	27
I. DESARROLLO	29
1. EXPLORACIÓN DE DATOS ACTUALES.....	30
1.1. Selección Manual de Variables por Fenotipos.....	30
1.2. Algoritmo de Búsqueda por Diccionario de Palabras	31
1.3. Selección de Tablas por Características	31
1.4. Modelo de Datos de Fuentes.	32

2.	DIRECTRICES ARQUITECTÓNICAS	33
2.1	<i>Atributos de Calidad</i>	33
2.2	<i>Restricciones Arquitectónicas</i>	34
II. ARQUITECTURA DE REFERENCIA PROPUESTA		35
1.	ARQUITECTURA DE REFERENCIA DE GRANDES VOLÚMENES DE DATOS Y ANALÍTICA.....	35
1.1	<i>Notaciones de la Arquitectura de Referencia</i>	35
1.2	<i>Arquitectura de Referencia Propuesta</i>	37
2.	ARQUITECTURA LÓGICA	42
2.1	<i>Modelo de Datos del Almacén</i>	42
2.2	<i>Modelo de Datos de Seguridad</i>	43
2.3	<i>Modelo de Datos de Administración y Organización</i>	44
2.4	<i>Modelo de ETL (Extracción, Transformación y Cargue) de Datos</i>	46
2.5	<i>Modelo de Datos de Resultados</i>	46
3.	ARQUITECTURA DE APLICACIONES	48
3.1	<i>Punto de Vista de Estructura de Aplicaciones</i>	48
3.2	<i>Punto de Vista de Uso de Aplicaciones</i>	50
4.	ARQUITECTURA FÍSICA	51
	<i>Punto de Vista de Tecnología</i>	51
III. INSTANCIAS ARQUITECTÓNICAS		54
1.	PRIMERA ITERACIÓN	54
2.	SEGUNDA ITERACIÓN.....	56
3.	TERCERA ITERACIÓN	57
4.	CUARTA ITERACIÓN.....	59
IV. PRUEBA DE CONCEPTO		60
1.	ALCANCE GENERAL DE LA PRUEBA DE CONCEPTO	60
1.1	<i>Arquitectura de Referencia Propuesta</i>	60
1.2	<i>Modelo de datos del almacén</i>	62
1.3	<i>Modelo de Datos de Administración y Organización</i>	62
1.4	<i>Modelo de ETL</i>	62
1.5	<i>Modelo de Datos de Resultados</i>	62
1.6	<i>Punto de Vista de Estructura de Aplicaciones</i>	63
1.7	<i>Punto de Vista de Uso de Aplicaciones</i>	64
1.8	<i>Punto de Vista de Tecnología</i>	65
2.	DOMINIOS DE LA PRUEBA DE CONCEPTO	66
2.1	<i>Orígenes de Datos</i>	66
2.2	<i>Ingestión de Datos</i>	67
2.3	<i>Almacenamiento de Datos</i>	68

2.4.	<i>Preprocesamiento</i>	69
2.5.	<i>Modelos y Especificaciones de Flujos Trabajo de Analítica</i>	69
2.6.	<i>Interfaz y Visualización</i>	70
3.	USO DE LA AR PROPUESTA EN OTROS PROBLEMAS DE DIAGNÓSTICO	71
V. VALIDACIÓN DE LA ARQUITECTURA.....		73
1.	ATAM.....	73
2.	TAM.....	73
3.	VALIDACIÓN DE LA ARQUITECTURA CON LA GERENCIA DEL HUSI	75
4.	DOCUMENTACIÓN FINAL	76
VI. CONCLUSIONES		77
REFERENCIAS		78

ABSTRACT

Every day we have more clinical data with the potential to transform them into knowledge. This exercise can be complex and time consuming, so a reference architecture is a proposed that allows data scientists to focus on analytical flows, while reducing their design and implementation effort. At the same time, it offers analytical systems architects a framework and guidelines for designing and managing structures that utilize Big Data, processing flows, and variability in data sources, particularly in the healthcare environment. This work offers models of the different architectural views and guidelines, as well as a proof of concept that instantiates this architecture in a sleep apnea diagnostic case from electronic medical records.

RESUMEN

Cada día se tienen más datos clínicos con potencial de transformarlos en conocimiento. Este ejercicio puede ser complejo y demorado, por lo que se propone una arquitectura de referencia que permita a los científicos de datos enfocarse en los flujos analíticos reduciendo su esfuerzo de diseño e implementación. Al mismo tiempo, ofrece a los arquitectos de sistemas analíticos un marco y lineamientos para diseñar y administrar estructuras que utilicen grandes volúmenes de datos, flujos de procesamiento y variabilidad en las fuentes de datos, particularmente en el entorno de la salud. Este trabajo ofrece modelos de las distintas vistas arquitectónicas y directrices, así como una prueba de concepto que instancia esta arquitectura en un caso de diagnóstico de apnea del sueño a partir de historias clínicas electrónicas.

RESUMEN EJECUTIVO

La industria de la salud está en constante innovación y dando grandes avances en la detección y tratamiento de las diferentes enfermedades, pero este crecimiento constante incrementa la necesidad del uso de la tecnología en los distintos procedimientos y es esta relación la que hace que cada día se estén almacenando más datos de pacientes, valoraciones y exámenes médicos, entre otros. Esta generación de datos hace que la ciencia de datos tome un papel fundamental para seguir perfeccionando los diferentes procedimientos médicos.

Para lograr generar conocimiento al ritmo que se generan los datos, los científicos de datos se ven en situaciones complejas, con sistemas heredados donde muy pocos conocen realmente donde se guardan los datos, o se enfrentan a la necesidad de trabajar con datos que no son estructurados. Son estas necesidades las que mueven el presente trabajo de grado.

Este trabajo intenta recopilar la mayoría de las problemáticas presentadas por los científicos de datos (y el personal médico) para abordarlo desde la perspectiva de la ingeniería, la detección de procesos repetitivos, la optimización de recursos para realizar experimentos complejos con los datos, cumplir con las normas de seguridad en la protección de los datos; se toman todas estas características para generar una arquitectura de referencia analítica.

Para llegar a esta arquitectura de referencia se realizaron diferentes análisis, desde el estudio de la composición de la base de datos del Hospital Universitario San Ignacio, una lista de escenarios, requerimientos no funcionales y pruebas de concepto que soportaron los ejercicios analíticos complejos realizados por los científicos de datos.

El proceso que se siguió para llegar a la arquitectura de referencia fue iterativo. En todo momento se tenía en cuenta al científico de datos para que fuera una guía de cuáles eran las funcionalidades mínimas que deberían ser cubiertas por la arquitectura y cuáles eran las cualidades con que debería contar la implementación. Esta distinción es importante porque se buscó que, desde la etapa del diseño, la arquitectura de referencia cumpla con la mayoría de los requerimientos no funcionales de los científicos de datos.

En estos procesos iterativos se generaron distintas instanciaciones de la arquitectura de referencia en las cuales se probaron diferentes escenarios arquitectónicos. Al mismo tiempo se validaba que se cumplieran con los requerimientos no funcionales, hasta desarrollar la arquitectura de referencia y su implementación hasta alcanzar la totalidad de los requerimientos originalmente planteados por los científicos de datos.

De este proceso cabe destacar la generación de la arquitectura de referencia, ya que es un marco que no solo puede servir al equipo interdisciplinario enfocado en la identificación de biomarcadores de apnea del sueño, sino que también podría llegar a ser una solución a cualquier otro equipo que necesite generar flujos analíticos periódicos, así como diferentes fuentes de datos.

Otro de los puntos importantes a destacar es que la prueba de concepto no solo está diseñada para científicos de datos. En comparación con otras arquitecturas, la presentada también

cuenta con módulos de control y configuración, lo cual la convierte en lo suficientemente abierta para añadir fuentes de datos adicionales, esto ayuda a controlar la forma en que se ejecutan los flujos de analítica e indica cómo se pueden guardar los diferentes resultados de las pruebas analíticas.

La arquitectura de referencia se fue evaluando en las diferentes iteraciones, pero se utilizó el modelo TAM (*Technology Acceptance Model*) para poder medir la percepción de utilidad. En esta evaluación se lograron realizar dos encuestas y en ambas el resultado fue muy parecido, indicando que la implementación de la arquitectura de referencia agrega valor a las labores que actualmente realizan los científicos de datos, pero al mismo tiempo no elimina del todo la complejidad de instanciación y configuración.

La arquitectura de referencia propuesta es un marco para que los arquitectos cuenten con directrices para generar arquitecturas de analítica que se puedan llevar a ambientes productivos; sin embargo, junto con la implementación, sigue siendo un producto intermedio que necesita más trabajo para facilitar un apoyo más completo para un ejercicio analítico en el contexto clínico.

INTRODUCCIÓN

Una gran cantidad de datos de la salud de las personas son recopilados día a día en todo el mundo en la atención clínica de cada paciente incluyendo diagnósticos, procedimientos, signos vitales, pruebas de laboratorio, notas médicas, antecedentes, entre otros; y a medida que el paciente sigue asistiendo a las atenciones médicas, se tiene aún más información con la que se puede establecer el desarrollo de una enfermedad y el tratamiento recomendado por el médico tratante.[1] Al mismo tiempo, el alto volumen de registros médicos electrónicos de los pacientes deben ser evaluados y analizados por investigadores de datos quienes realizarán operaciones de integración, limpieza, interpretación, agregación y visualización con la variedad de fuentes de datos extraídos[2] que hacen que emerjan retos interesantes en la recolección, integración, almacenamiento, procesamiento y análisis de datos enmarcados en el volumen, variabilidad y velocidad de los mismos, siendo deseable una arquitectura de Big Data.[3]

De otra parte, aún no existe un entendimiento claro por parte del personal médico de los beneficios que se pueden obtener al combinar las historias clínicas de sus pacientes con las arquitecturas de Big Data, análisis de datos, aprendizaje profundo, estrategias de almacenamiento y procesamiento de datos, combinación que puede dar como resultado el hallazgo de patrones en los pacientes por cada una de sus atenciones médicas, la trazabilidad, el apoyo a la toma de decisiones, el análisis de datos no estructurados y la capacidad predictiva de un diagnóstico.[4]

En el contexto de este trabajo, es de especial interés el síndrome de apnea o hipopnea obstructiva del sueño (SAHOS) que es uno de los diagnósticos que ha sido de mayor importancia en la comunidad médica y que hoy en día se encuentra infradiagnosticado o con diagnóstico incorrecto o simplemente sin diagnosticar [5] [6]. En particular, este trabajo se enmarca como una estructura de apoyo para el Hospital Universitario San Ignacio (HUSI) y las facultades de Medicina y Odontología de la Pontificia Universidad Javeriana, quienes participan de un proyecto sombrilla del presente trabajo enfocado en la identificación de biomarcadores de Fibrilación Auricular asociada a apnea del sueño mediante herramientas de análisis Genome Wide Association, Phenome Wide Association y Big Data, así como de un proyecto de doctorado orientado a generar modelos y métodos analíticos para esta problemática.[7]

El SAHOS o también llamado apnea obstructiva del sueño (AOS) es un trastorno respiratorio del sueño y uno de los principales problemas de salud pública[8] donde la respiración se interrumpe o se vuelve superficial, es decir que existe un colapso repetitivo parcial o completo de las vías respiratorias mientras la persona duerme[9]. Muchos de los síntomas están asociados a ronquidos, somnolencia diurna en exceso, dolores de cabeza al despertarse, poca concentración en las actividades, incluso la forma, posición y función de las estructuras dentomaxilofaciales, entre muchos otros que aún no son concluyentes para determinar un diagnóstico positivo de la AOS, pues son necesarios más estudios vinculados al sueño que ayuden a determinar si la persona tiene el trastorno[10]. Recientemente, se han reconocido y probado las posibilidades de la analítica y el aprendizaje profundo en el apoyo del proceso diagnóstico[11] [12].

El presente trabajo busca apoyar al diagnóstico de la apnea del sueño ayudando al tratamiento enfocado en medicina personalizada o de precisión, obteniendo mejores datos de las comorbilidades de los pacientes debido a las problemáticas encontradas en las historias clínicas electrónicas determinadas por la calidad, diversidad, volumen de los datos y también al gran uso de texto narrativo.

En línea con lo anteriormente mencionado, la contribución en el presente trabajo es proponer el diseño de la arquitectura de referencia de Big Data para analizar los datos de historias clínicas en el contexto del diagnóstico de la apnea del sueño, aunque busca ser extensible para cualquier otro diagnóstico, que permita generar un modelo predictivo, por medio de aprendizaje de máquina, aprendizaje profundo y procesamiento de lenguaje natural. La arquitectura de referencia se valida mediante una prueba de concepto formada por diferentes componentes que permitan almacenar, procesar y analizar grandes volúmenes de datos en *batch*. La arquitectura en gran medida es una solución que facilitará las tareas de los científicos de datos cuando se enfrentan a problemas de calidad de datos con múltiples fuentes, preprocesamientos e integraciones; para que finalmente se pueda llegar a la aplicación de métodos analíticos y así escribir algoritmos de aprendizaje de máquina y aprendizaje profundo.

La arquitectura se centrará en los flujos de datos de las historias clínicas del HUSI incluyendo variables que indiquen fenotipos de los pacientes, los niveles de procesamiento que permitan mostrar resultados que sean útiles para el diagnóstico; adicionalmente, es necesario que la arquitectura esté abierta a la ingesta de otras fuentes de datos, que soporte la carga de trabajo de modelos predictivos de aprendizaje profundo y que sea configurable para los analistas de datos. Entre las limitantes se encuentran: la utilización de infraestructura donde solo se tenga acceso desde la red universitaria y que el software utilizado debe ser libre de cualquier tipo de licenciamiento o en su defecto software a la medida.

En el capítulo 1 se realiza la descripción general del trabajo realizado enfatizando en la oportunidad y problemática que motivaron al desarrollo del proyecto. El capítulo 2 presenta la descripción del proyecto definiendo el objetivo general y los objetivos específicos. En el capítulo 3 se ilustra el estado del arte o marco teórico. El capítulo 4 se realiza una breve descripción de los trabajos relacionados continuando con el capítulo 5 en donde especifican los detalles del desarrollo del proyecto realizado. En el capítulo 6 se mencionan los puntos más importantes de la prueba de concepto realizada y se finaliza con las referencias y anexos del proyecto.

1. DESCRIPCIÓN GENERAL

Oportunidad y problemática

Frente a la necesidad del Hospital Universitario San Ignacio (HUSI) y las facultades de medicina y odontología de la Pontificia Universidad Javeriana y en línea con el trabajo enfocado en la identificación de biomarcadores de Fibrilación Auricular asociada a apnea del sueño mediante herramientas de análisis *Genome Wide Association*, *Phenome Wide Association* y Big Data, se presenta la oportunidad para diseñar y construir una arquitectura que soporte el proceso de analítica de datos. *Genome Wide Association* y *Phenome Wide Association* son los estudios realizados en todo el genoma humano con diferentes variantes genéticas e iteraciones en los genomas de una muestra de personas bastante grande para identificar esa asociación entre genotipo y fenotipo dentro de los registros médicos electrónicos y encontrar múltiples fenotipos y relaciones entre genotipos específicos asociadas a alguna enfermedad en particular, cita que para el presente caso de estudio hacen relación al diagnóstico de la apnea del sueño. [13] [14]

El propósito es que la arquitectura permita la identificación y selección de variables como datos estructurales y no estructurales que describen un fenotipado preciso de cada paciente para llevarlas a través de un procesamiento de lenguaje natural y posteriormente hasta poder diseñar un modelo de analítica que permita predecir cualquier diagnóstico que para este trabajo se presentará como prueba de concepto el de la apnea obstructiva del sueño.

Diariamente el HUSI recolecta una cantidad extrema de datos de cada paciente en su historia clínica haciendo que exista una gran variedad de estos datos en notas médicas, demográficos, enfermedades, procedimientos, antecedentes, signos vitales, diagnósticos, registros de resultados de la polisomnografía cuando se trata de pacientes con AOS incluso los resultados que pueden surgir del tratamiento con CPAP y muchos más registros que contribuyen a un Big Data y que poco a poco se convierte en datos e información inmanejable sino se trata a tiempo con una arquitectura que facilite tareas de gestión, compilación y análisis de datos que involucren enfoques estadísticos y algoritmos de *Machine Learning*, *Deep Learning* e inteligencia artificial.[15] Una arquitectura de referencia ayuda a facilitar el problema mencionado anteriormente porque permite solucionar los problemas de integración y comunicación de componentes tecnológicos, también de almacenamiento y procesamiento de datos, antes de realizar la implementación y desarrollo de la solución. De tal manera se brinda una solución robusta capaz de soportar fallos y prevenir errores y retrasos en la práctica.

En una primera perspectiva global de la base de datos proporcionada por el HUSI se evidencia que no será tarea fácil para el analista de datos seleccionar las variables necesarias para un pipeline de analítica que se quiera desarrollar, es por esta razón que, en adición a lo anteriormente mencionado, es necesario definir estrategias optimizadas para la selección y el tratamiento dinámico de las variables. La selección de variables y de características es una de las etapas más importantes de un pipeline de analítica de datos cuando se trata de mejorar el rendimiento y la precisión de la predicción de los predictores, el entendimiento de cada variable, la identificación de patrones, filtros y agrupamientos.[16]

Con el objetivo de facilitar las tareas de analítica de datos enfocadas al aprendizaje de máquina y al aprendizaje profundo para el HUSI y las facultades de medicina y odontología de la

PUJ, adicionalmente, se debe incluir en la arquitectura de referencia actividades de preprocesamiento de datos, procesamiento de lenguaje natural y la posibilidad de ejecutar algoritmos de aprendizaje de máquina y de aprendizaje profundo que generen modelos de clasificación y todo lo anteriormente mencionado e implementado en un pipeline de analítica de datos los cuales son frecuentemente usados en el campo de la salud de los pacientes para predecir sus diagnósticos usando las historias clínicas y aún más en temporada de pandemia[17] ofreciendo una gran cantidad de beneficios al incluir los métodos de analítica de datos con una arquitectura de referencia de Big Data en la atención médica tales como la atención preventiva que conlleva a la mejora de la calidad de vida de las personas aplicando análisis avanzados de perfiles de pacientes en segmentación y modelos predictivos, también, la detección de enfermedades en etapas más tempranas y facilitar el tratamiento oportuno; y profundizando un poco más, en el análisis de genotipos y fenotipos.[18]

Se descubrió otra problemática con los científicos de datos sobre la creación de un pipeline que les permita realizar el análisis de los datos que les ayude a predecir un diagnóstico con algoritmos de aprendizaje profundo y también selección de variables, generación de vistas minables, normalización de datos y tareas de procesamiento de lenguaje natural para poder disponer de un pipeline de analítica incluyendo su arquitectura de referencia de Big Data que lo soporte.

2. DESCRIPCIÓN DEL PROYECTO

2.1. Objetivo general

Diseñar una arquitectura de referencia que soporte la ejecución de modelos de aprendizaje profundo y procesamiento de lenguaje natural sobre historias clínicas electrónicas en el contexto del apoyo a un proceso de diagnóstico de apnea del sueño.

2.2. Objetivos específicos

- Descubrir y explorar el modelo de datos y fuentes de las HCE del contexto de aplicación.
- Definir los requerimientos no funcionales que deberá soportar la arquitectura.
- Diseñar el modelo de datos fuente y del almacén de datos principal.
- Modelar los procesos de extracción, transformación y cargue de datos con estrategias de integración de sistemas.
- Diseñar y modelar los componentes de software e infraestructura de la arquitectura.
- Diseñar la capa de analítica y de visualización de datos.
- Generar la documentación específica para la administración de datos.
- Implementar prueba de concepto.

2.3. Fases de desarrollo

La metodología que se utilizó fue del tipo iterativa donde se utilizaron pequeños ciclos de vida, de los cuales se obtenía una retroalimentación directa ya sea de los científicos de datos o del comportamiento de los componentes al integrarse en el proyecto. El proyecto se puede dividir en cuatro fases que nos permitieron determinar cuáles eran los componentes más adecuados.

Las fases en las que se dividió el proyecto fueron:

1. Exploración de datos actuales
2. Arquitectura de referencia
3. Prueba de concepto
4. Validación de la prueba de concepto

2.3.1 Exploración de datos actuales

Esta fase corresponde al análisis de la base de datos del Hospital Universitario San Ignacio (HUSI). Esta base de datos es llamada SAHI_PUJ, fue previamente tratada para anonimizar los datos sensibles y se aplicaron diferentes criterios de selección, que en conjunto con los científicos de datos se determinó cuál realmente era la información necesaria para extraer las variables fundamentales.

2.3.2 Arquitectura de referencia

En esta fase se hace un análisis de los requerimientos no funcionales que tienen los científicos de datos, para poder ir modelando diferentes escenarios, estos escenarios nos fueron guiando hasta encontrar la arquitectura que cumpla los escenarios y que también cumpla con los requerimientos de los científicos de datos. En esta etapa se diseñan los diferentes dominios que debe tener la arquitectura de referencia.

2.3.3 Prueba de concepto

Esta fase corresponde al análisis de las diferentes tecnologías actuales, que cumplan con los requerimientos no funcionales, se hace un análisis de componentes por dominio, el cual nos permitió conocer cuáles eran los componentes necesarios para cada dominio de la prueba de concepto; en esta fase también se generan el manual técnico que servirá a los científicos de datos para su posterior utilización.

2.3.4 Validación de la Prueba de concepto

En esta fase se realizan dos evaluaciones para conocer si la prueba de concepto en realidad es útil para los científicos de datos, por un lado, se realiza la validación por medio de ATAM (*Architecture Tradeoff Analysis Method*) la cual nos permite medir la arquitectura con respecto a los atributos de calidad definidos por los científicos de datos.

También se realiza la evaluación TAM (*Technology Acceptance Model*) la cual nos indicara la influencia que tiene la prueba de concepto sobre los científicos de datos, esto nos permite conocer que tan profunda será la interacción de la prueba de concepto con los científicos de datos.

3. MARCO TEÓRICO / ESTADO DEL ARTE

En este capítulo se presentan los conceptos teóricos más importantes que permiten comprender a profundidad el contexto del desarrollo del trabajo realizado en cuanto a la preocupación que tiene el área de la salud frente a grandes volúmenes de datos recolectados de los pacientes en las historias clínicas electrónicas. También se presentan los conceptos sobre Big Data y Arquitectura de Referencia (AR) en el campo de la analítica de datos.

3.1. Grandes Volúmenes de Datos en el Área de la Salud

Todos los sistemas de salud recolectan y mantienen gran cantidad de registros médicos en las historias clínicas electrónicas de sus pacientes: imágenes médicas, notas clínicas, datos genéticos, características fenotípicas, y una gran base de conocimiento clínico de pacientes que hace necesaria una gestión de Big Data, y aplicando métodos de analítica clínica se extrae el valor correspondiente de los datos enmarcados en una Arquitectura de Referencia. Lo anterior hace que existan trabajos para gestionar de forma eficaz los datos clínicos en el seguimiento de pacientes para que los médicos puedan tomar decisiones oportunamente.[19] Y en el campo de la salud, con la AR de Big Data, también se abarcan oportunidades para identificar pacientes de alto riesgo y de alto costo minimizando los precios en la atención médica en cuanto a readmisiones, triage, descompensaciones, eventos adversos y la optimización de tratamientos de diferentes enfermedades.[20]

En el contexto de las atenciones médicas, múltiples fuentes médicas generan grandes volúmenes de datos e incluyen, por ejemplo, imágenes biomédicas, informes de pruebas de laboratorio, notas escritas por médicos y parámetros que permiten el monitoreo de la salud del paciente en tiempo real. Además de su enorme volumen y su diversidad, los datos sanitarios fluyen a gran velocidad. Como resultado, los enfoques de Big Data ofrecen enormes oportunidades con respecto a la eficiencia de los sistemas de salud.[21]

3.2. Arquitecturas de Big Data en el Área de la Salud

Las arquitecturas de Big Data han tenido un gran alcance en el campo de la salud. En los últimos años se evidencia la importancia que tienen este tipo de arquitecturas con trabajos que proporcionan el diseño de plataformas digitales inteligentes para la gestión de pacientes y prevención personalizada. Esto incluye la ingesta de datos estructurados y no estructurados de la salud de los pacientes con IoT (internet de las cosas) y dispositivos portátiles para realizar algo llamado medicina 4P las cuales significan predictiva, preventiva, personalizada y participativa que serán potencializadas por tareas de inteligencia artificial y aprendizaje de máquina las cuales facilitan la atención médica inteligente, haciendo que los pacientes tengan una mejor calidad de vida.[22] Por otra parte, la arquitectura puede estar orientada a microservicios para Big Data que atiende las necesidades de los médicos en cuanto a una visualización más comprensiva, clara y eficiente de los datos de los pacientes en las atenciones médicas, incluso de forma remota.[23]

Entre otras investigaciones orientadas a diseñar arquitecturas de Big Data, existen las que soportan aplicaciones de métodos analíticos de aprendizaje de máquina para predecir un

diagnóstico específico como el del cáncer mediante el modelo de Markov y la agrupación en clústeres con los datos del ADN, considerados como una medida esencial para diagnosticar la enfermedad del cáncer.[24] De igual manera, la arquitectura de Big Data se usa para recopilar los datos clínicos de un paciente y realizar varios modelos de aprendizaje de máquina para llegar al diagnóstico diferencial de síndrome mielodisplásico hipocelular y anemia aplásica.[25] Además, las arquitecturas de IoT soportadas en Big Data son utilizadas para la predicción de diferentes enfermedades crónicas en tiempo real mediante el aprendizaje automático, con el objetivo de mejorar la calidad de vida de los pacientes detectando enfermedades a tiempo. [26]

3.3. Big Data

Big Data, que en otros contextos es traducida al español como macrodatos o datos masivos, es un paradigma que surge cuando existen datos o conjunto de ellos donde el tamaño está fuera del alcance de las herramientas de software comunes como son las aplicaciones de captura, administración y procesamiento de datos dentro de tiempos aceptables. Adicionalmente, el origen de los datos es de diversas fuentes y provienen en diferentes formas como estructurada, semiestructurada y no estructurada sumando que la velocidad en que se transmiten debe ser alta cuando los datos entran y salen de cada elemento de la arquitectura y al exterior de esta, pero a estos conceptos también se han adicionado otros como el valor que tienen estos datos y el grado de confianza que ofrecen para la toma de decisiones. Las cinco características de Big Data se encuentran definidas como Volumen, Variabilidad, Velocidad, Valor y Veracidad.[27]

3.3.1. Volumen

El volumen hace referencia a grandes cantidades de datos. Cuando los sistemas tradicionales comienzan a reflejar fallas por el aumento de los datos, se evidencia que la arquitectura estándar debe cambiar por una de Big Data.[28] Se espera que el volumen de datos de la salud de los pacientes siga creciendo a un ritmo acelerado, incluso más por la pandemia del COVID 19. Desde 2015 los datos se han cuadruplicado hasta llegar a 20 *zettabytes* en el 2021, es decir, 20 billones de *gigabytes*. [29]

3.3.2. Variabilidad

La variabilidad es definida por los diferentes formatos en los que se encuentran los datos.[28] Las variaciones también se presentan en las tasas de flujos de datos. Existe una complejidad alta por los grandes datos que a menudo se producen a través de un conjunto de diversas fuentes de datos, lo que implica que, para realizar muchas operaciones sobre los datos, estas operaciones incluyen la identificación de relaciones, limpieza y transformación de datos que fluyen desde diferentes orígenes. Los datos de los sistemas de salud pueden ser recopilados de diferentes bases de datos de información desde sensores médicos y datos hospitalarios.[30]

3.3.3. Velocidad

La velocidad en este contexto es sinónimo de rapidez.[28] La velocidad no proporciona una descripción coherente de los datos debido a sus picos y valles periódicos. Explicado de otra manera, es la rapidez con la que los datos son recopilados por los sistemas de salud y entre más rápido se encuentren los datos listos para su procesamiento, más rápida será la posibilidad en que los médicos podrán tomar sus decisiones.[31]

3.3.4. Valor

El valor hace referencia a esas características de los datos que hacen que sean muy importantes en el contexto de la salud, pues la idea es encontrar ese valor intrínseco. Se deben poder analizar los beneficios y los costos de recopilar, procesar y analizar los datos para determinar los tipos de recompensas para todos los actores del sistema, en especial, las de cada paciente en cada atención médica.[32]

3.3.5. Veracidad

En la veracidad se habla de fuentes confiables de datos.[28] Los sistemas de salud que proporcionen los datos deben estar acreditados para garantizar la precisión y la certeza de los mismos.[30] La veracidad también está dada en el grado de seguridad en el que se puede ver que los datos sean coherentes, que existan niveles de fiabilidad y credibilidad altos; entonces, los datos no pueden tener errores y menos cuando se trata de la salud de un ser humano; y reuniendo estas características de la veracidad, cuando los datos ingresen a la etapa de aprendizaje de máquina, los resultados serán más confiables y útiles para la toma de mejores decisiones en la salud de los pacientes.[31]

Entre otras características de Big Data se encuentran la de modelos predictivos y herramientas de minería de datos ayudando a la toma de decisiones frente a la selección de variables, datos e información importante en el contexto de la salud de los pacientes en sus historias clínicas.[28]

3.4. Arquitectura de Referencia de Big Data y Analítica

Una arquitectura de referencia puede ser una arquitectura genérica la cual a su vez puede reutilizarse para desarrollar arquitecturas más específicas incluida la arquitectura de software la cual puede considerarse una arquitectura de aplicación. De la arquitectura de referencia se puede extraer instancias para otras arquitecturas más concretas.[33, p. 0] La AR debe ser agnóstica a la tecnología, es decir, que no se tenga preferencia por una tecnología específica.

Adicionalmente, en la AR se especifican configuraciones, extensiones y decisiones clave de arquitectura. Una característica principal de la AR es que debe ser capaz de adaptarse a cualquier necesidad del sistema[34]. La AR como una solución planteada, también se define como diseños predefinidos utilizados para desarrollar alguna aplicación en particular incluyendo diferentes estilos de arquitectura y patrones de diseño y con respecto a su estructuración ge-

neral, todos los elementos deben estar debidamente definidos con sus respectivas responsabilidades [35].

HPE (Hewlett Packard Enterprise) considera que la arquitectura de referencia ayuda a diferentes roles como gerentes de proyectos, desarrolladores de software, arquitectos empresariales y gerentes de TI a la comunicación y colaboración efectiva cuando se tiene un proyecto de implementación obtenido como instancia de la AR que puede anticipar problemas y soluciones que podrían presentarse en el desarrollo del proyecto ayudando a los equipos a disminuir los errores y retrasos que sucedan si no se tiene una referencia inicial de una arquitectura probada y con las mejores prácticas.[36]

3.4.1. Por qué una Arquitectura de Referencia

La AR ofrece una línea base de inicio para dar solución a un problema dado en un contexto definido, y a partir de la AR, se pueden implementar múltiples soluciones funcionales para diferentes usuarios finales según sea el caso.

Al utilizar una AR se está aprovechando una investigación y un conocimiento previo aplicado con anterioridad a otros sistemas similares con nomenclaturas estandarizadas; de esta manera permite obtener aquellos componentes o elementos clave que se deben tener en cuenta para la solución.[37] La AR facilita un mejor diseño e implementación de la arquitectura para que su propósito tenga un mayor éxito disminuyendo complejidades en la planeación, diseño e implementación.[38]

3.4.2. Cómo Hacer una Arquitectura de Referencia

En realidad, no existe una receta específica para crear una AR. La AR es la receta para arquitecturas concretas y debe ser construida obedeciendo los *concerns* o preocupaciones de los *stakeholders* o actores interesados en diferentes dominios de negocio, datos, infraestructura y aplicaciones. La AR se descompone en módulos que, junto con sus integraciones, son las mejores opciones de decisiones de diseño direccionadas por los *concerns*. [39]

Debido a que no existe un paso a paso genérico en el proceso de creación de una AR, con la recopilación de diferentes artículos y textos de arquitectura citados a continuación, se seguirán las siguientes instrucciones para la construcción de la AR:

Pasos	Descripción
1. Identificar referentes	Lo inicial es contextualizar el objetivo de la arquitectura. Se debe realizar una investigación para conocer las especificaciones de la industria o negocio en donde se realizará la AR. Consultar bibliografía, entender el negocio, buscar referencias de arquitecturas existentes y similares que se hayan realizado anteriormente. Determinar si aquellos artefactos arquitectónicos ya se encuentran construidos en donde exista una base para comenzar. También es importante validar la confiabilidad de las fuentes investigadas. También se puede iniciar con la exploración de los datos ampliando el conocimiento para generar de forma organizada las primeras entidades, dominios, módulos y componentes.[40]
2. Definir los atributos de calidad	Los atributos de calidad son los requerimientos no funcionales los cuales desempeñan un papel importante en el proceso de creación de una AR y proporcionan sistemas con arquitecturas de alta calidad. Las diferentes decisiones de arquitectura tomadas a lo largo del proceso son en gran medida influenciadas por los atributos de calidad.[41]
3. Generar la notación de la AR	Obtener o definir una notación que permita simplificar el entendimiento de la arquitectura. La notación puede ser definida alineada por algún estándar como UML (Lenguaje de modelado unificado) o la especificación de <i>Archimate</i> que, aunque está orientada para arquitecturas empresariales, también puede orientarse a arquitecturas de referencia porque permite analizar, describir y comunicar los <i>concerns</i> en arquitecturas que cambian en el tiempo. Las representaciones gráficas como conjunto de entidades y relaciones están enmarcadas en las capas de negocio, aplicaciones y tecnología adicionando elementos de motivación y estrategia.[42] Posteriormente, la notación también facilita la ejemplificación de otros escenarios de arquitectura.[43]

4. Proponer la AR	<p>Definir y proponer la AR. En algunas ocasiones la posición de los bloques puede tener significados de continuidad en el flujo de la arquitectura cuando se encuentran al lado derecho o también pueden determinar algún tipo de alcance cuando se sitúan arriba o como base en la parte de abajo. También para demarcar la interacción entre componentes, se suelen incluir relaciones con líneas que los conectan u se suelen incluir flechas de acuerdo con el flujo de información o según el tipo de asociación. Se recomienda una herramienta de edición de diagramas como Enterprise Architect, visio, AppDiagrams, entre otras muchas más que facilitan esta tarea. No es viable tener el diagrama de la arquitectura sin que exista una explicación clara de cada elemento de la arquitectura y sus interacciones.[40]</p>
5. Generar vistas o puntos de vista de la AR	<p>En ocasiones un único diagrama no es suficiente para explicar la AR en su totalidad. Puede requerirse de otras perspectivas gráficas. Se debe recordar que la arquitectura puede tener múltiples <i>stakeholders</i> con diferentes <i>concerns</i> y esto puede requerir diferentes puntos de vista. En este caso puede ser utilizado UML, Archimate u otra especificación que le permita representar gráficamente los diferentes puntos de vista.[42]</p> <p>Cuando es utilizada la especificación de Archimate, se involucran obligatoriamente las tres capas: negocio (color amarillo), aplicaciones (color azul) y tecnología (color verde) y sus elementos de motivación y estrategia ya sea individualmente o combinando diferentes artefactos entre ellos.</p>

6. Validar la AR	<p>La validación de la arquitectura es una evaluación de los objetivos de negocio con la arquitectura. Debido a la complejidad de la arquitectura es necesario validarla con respecto a los atributos de calidad. El análisis arquitectónico puede ser realizado por el método ATAM (<i>Architecture Tradeoff Analysis Method</i>) encargado de comprender la interacción de los atributos de calidad y sus consecuencias con respecto a las diferentes decisiones de arquitectura tomadas. [44]</p> <p>De manera general los pasos del método ATAM son los siguientes:</p> <ol style="list-style-type: none"> 1. Presentación de ATAM 2. Presentación de los objetivos del negocio 3. Presentación de la arquitectura 4. Identificar los enfoques arquitectónicos 5. Generar árbol de utilidad de atributos de calidad 6. Analizar enfoques arquitectónicos 7. Lluvia de ideas y priorización de escenarios 8. Analizar enfoques arquitectónicos 9. Presentación de resultados
------------------	--

Finalmente, al incluir Big Data y los *pipelines* de analítica, la arquitectura de referencia es la representación del modelo general aplicado a los sistemas en donde interactúa un gran volumen, variabilidad y velocidad de datos involucrando tareas de almacenamiento de datos, gestión de la información, procesamiento de los datos, aplicación de métodos de analítica de datos e interfaces y componentes de visualización.[27]

3.5. Flujos de Trabajo en Analítica de Datos

El concepto los flujos de trabajos en la analítica de datos, pipelines en inglés, está fuertemente enlazado con el flujo de datos que se realiza en las tareas globales de captura y/o extracción, transformación, carga, procesamiento, almacenamiento y análisis de los datos para la toma de decisiones. En las tareas mencionadas anteriormente, existen una serie de eventos como la creación y ejecución de ETL (*Extract, Transform, Load*) con opción de periodicidad, funciones de limpieza de datos, selección de variables y características, reducción de dimensionalidades, implementación de algoritmos de aprendizaje de máquina y aprendizaje profundo, creación de modelos de predicción y la posibilidad de crear y ejecutar procesos en *batch* o en tiempo real.[45]

IBM realiza una arquitectura de referencia para el análisis de alto rendimiento en la salud y en las ciencias de la vida mostrando los flujos de trabajo en un análisis intensivo de entrada y salida de datos que son transformados requiriendo de cargas de trabajo extremadamente intensivas en computación y datos; adicionalmente, se pueden utilizar portales con una interfaz de usuario que permita a los usuarios controlar su acceso, crear, ejecutar y monitorear los diferentes pipelines de analítica.[46]

Para ampliar aún más el concepto de flujos de trabajo de analítica frente a los datos de las historias clínicas de los pacientes, se toma como referencia el enfoque los trabajos realizados por Hersh[47] y Kumar[48] en donde definen un pipeline de analítica con cuatro pasos importantes, el primero es el de la extracción de datos de diferentes fuentes que contengan registros de atenciones médicas de sus respectivos pacientes, incluyendo registros financieros, genómicos y demográficos; el segundo consiste en extraer las características con técnicas empleadas para procesamiento de lenguaje natural o NLP, normalizando, ordenando y extrayendo patrones de los datos; el tercero es el procesamiento estadístico en donde se incluyen técnicas de aprendizaje de máquina y aprendizaje profundo, inferencia estadística y finalmente el cuarto paso es el de efectuar las predicciones, realizar clasificaciones y analizar los resultados obtenidos.

Las operaciones de aprendizaje de máquina o MLOps aumentan ante la necesidad de los ingenieros de datos por implementar pipelines de analítica y existen diferentes herramientas que facilitan estas prácticas y entre las más conocidas están AirFlow, KubeFlow y Google Cloud AutoML que permiten la ejecución de operaciones del ciclo de vida del aprendizaje de máquina.[49] A continuación, se presentan los conceptos de las operaciones destacadas en los flujos de trabajo de analítica:

Operaciones	Definición
ETL	<p><i>Extract, Transform and Load</i> (extraer, transformar y cargar), frecuentemente llamado solo ETL, es el proceso que permite a las organizaciones mover datos desde múltiples fuentes, reformatearlos, limpiarlos y cargarlos en otra base de datos para analizarlos y apoyar un proceso de negocio.[50] En el área de las salud, los datos deben ser extraídos también de múltiples fuentes médicas de datos e integrarlos para posteriormente procesarlos, también se pueden incluir tipos de preprocesamiento como una limpieza previa porque las aplicaciones de ETL son muy eficientes en este contexto.[51] En ocasiones, después del proceso de ETL, los datos pueden quedar listos para el análisis y la predicción. Otra característica de los procesos de ETL es que se pueden incluir en scripts de ejecuciones llamados Jobs los cuales se pueden programar para ejecuciones periódicas.</p>
Procesamiento	<p>El procesamiento de los datos se puede presentar por lotes y por flujos.</p> <p>El procesamiento por lotes se realiza cuando deben ser analizados datos en un periodo de tiempo predeterminado sin limitaciones en cuanto al tiempo de respuesta, y así poder procesar grandes volúmenes de datos con recopilación y almacenamiento por lotes, técnica que ha tenido bastante éxito en el campo de la bioinformática y la salud.[28]</p> <p>El procesamiento por flujos se utiliza cuando los datos se deben procesar con tareas de retroalimentación en tiempo real. Esta técnica es utilizada con mayor frecuencia en el área de las atenciones médicas porque permite la</p>

Operaciones	Definición
	<p>extracción en tiempo real de los datos más relevantes de los pacientes de una gran cantidad de datos que se producen continuamente y que al ser procesados, aumentan los recursos de memoria computacional, pero reduciendo tiempos al realizar cálculos rápidos en los datos dando como resultados alertas tempranas de posibles complicaciones de los pacientes respecto al diagnóstico de enfermedades.[28]</p>
Limpieza de datos	<p>Es una tarea necesaria para impactar la mejor calidad de los datos y consiste en identificar, seleccionar y transformar las variables para determinar, eliminar o reemplazar datos que el científico de datos considera innecesarios o valores atípicos, es decir, un dato en donde su relación es totalmente distante con los demás datos, en inglés es llamado “outlier”; adicionalmente, en la limpieza se pueden eliminar atributos insignificantes o redundantes de los datos que no realizan contribuciones a la precisión del modelo de predicción.[52]</p> <p>Cuando se habla de registros de historias clínicas de pacientes, en algunas ocasiones se puede evidenciar que los médicos cometen errores en la digitación del texto de las notas clínicas lo cual puede generar errores de procesamiento. La limpieza de los datos se utiliza para detectar y/o remover registros inexactos de la historia clínica del paciente.[53]</p>
NLP (Procesamiento de Lenguaje Natural)	<p>El procesamiento de lenguaje natural o NLP (<i>Natural Language Processing</i>) por sus siglas en inglés, es un dominio interdisciplinario encargado de comprender los diferentes lenguajes naturales de los humanos y utilizarlos en la interacción con las computadoras o máquinas.[54] La eficacia de los algoritmos ejecutados en el campo del NLP depende del uso de una gran cantidad de datos para construir modelos simples con una alta calidad.</p> <p>Los métodos utilizados en el NLP son de gran impacto cuando se realizan investigaciones y aplicaciones en el área de la salud, pues en las historias clínicas de los pacientes la mayoría de la información está proporcionada en datos no estructurados, es decir, texto en donde se encuentra un conjunto de palabras, signos de puntuación, números, etc. en donde se representa el estado del paciente, su descripción física y médica, diagnósticos, síntomas, tratamientos, atributos semánticos que implican negación, gravedad, temporalidad y otras especificaciones médicas que agrupan todos sus datos en un historial médico constante.[55]</p> <p>Entre las tareas más destacadas de NLP para las historias clínicas de los pacientes, se encuentran la extracción de atributos y características de los textos a nivel de palabras y oraciones con el propósito de conocer el estado del paciente o el tipo de informe, también está la tokenización o segmenta-</p>

Operaciones	Definición
	<p>ción de palabras, el reconocimiento de entidades y conceptos nombrados para detectar más fácilmente los diagnósticos, síntomas o tratamientos; otra tarea con relacionada con la anteriormente nombrada es el reconocimiento y segmentación por tópicos, técnica empleada para reconocer palabras desde tópicos presentes en el corpus (conjunto de textos orales o escritos), también está la identificación de atributos semánticos como morfemas, palabras y oraciones, y muchas otras tareas que se pueden combinar para que sean de gran influencia en la aplicación de algoritmos de aprendizaje automático.[55]</p>
Aprendizaje de máquina	<p>El concepto de aprendizaje de máquina o “<i>machine learning</i>” ML en inglés, es introducido debido a los juegos de computadora en los años cincuenta en donde se presentó la primera computadora que jugaba ajedrez, luego se incrementó el estudio por los algoritmos que aprendían automáticamente con la clasificación de patrones; ML es una subclase de la Inteligencia Artificial (IA) en donde se enmarcan los algoritmos en el auto-aprendizaje a partir de la experiencia, es decir, que una vez que al algoritmo son ingresados los datos, éste identifica y aprende el patrón y proporciona una salida consecuente de ese aprendizaje adquiriendo más y más inteligencia por cada iteración sin participación humana.[56]</p> <p>Es muy importante la aplicación de ML en las atenciones médicas de los pacientes para evitar sesgos en el diagnóstico y su respectivo tratamiento como apoyo a las decisiones de los médicos, pero los sesgos no son del todo eliminados, siempre existen márgenes de error por ejemplo en su clasificación y medición.[57] Entre los algoritmos más utilizados en la clasificación de texto en historias clínicas se encuentran los clasificadores bayesianos, máquinas de vectores de soporte y árboles de decisión.[58]</p>
Aprendizaje profundo	<p>El aprendizaje profundo o “<i>Deep Learning</i>” DL, surge a partir del ML como una rama adicional con la aparición del concepto de redes neuronales; el DL es más robusto que el ML porque proviene de un enfoque de arquitectura de redes neuronales profundas en donde se tienen varios hiperparámetros y capas entre la entrada y la salida que realizan procesos de extracción y alteración de características de los datos haciendo que esta técnica sea ideal para tratar con datos muy grandes y complejos.[56]</p> <p>El aprendizaje profundo es utilizado para descubrir patrones de conjunto de datos complejos según en el contexto del cuidado de la salud con la predicción de enfermedades hasta servicios personalizados hacia el paciente.[46] Entre los métodos de DL más utilizados en el análisis de texto en historias clínicas de pacientes son las redes neuronales recurrentes y/o convolucionales.[58]</p>

4. TRABAJOS RELACIONADOS

En este capítulo se presentan los trabajos que están relacionados tanto a la generación de una arquitectura de referencia como al procesamiento de historias clínicas. Se generó esta división debido a que no se encontraron trabajos que combinen estas dos temáticas directamente.

Uno de los trabajos que aportó al modelamiento de la arquitectura de referencia es el de Sang, Go & Xu, Lai & De Vrieze, Paul (2016)[43] en el cual muestra una forma actualizada de modelar las arquitecturas de referencia de grandes volúmenes de datos a nivel empresarial en donde se parte de una arquitectura ya definida anteriormente y mejorándola a un nivel alto de abstracción con notaciones definidas de los elementos que intervienen en la nueva arquitectura, lo cual es algo similar a lo que se realiza en el presente trabajo, pues la generación de una notación permite simplificar el entendimiento de los componentes de la arquitectura de referencia. Adicionalmente, los autores realizan el mapeo de la tecnología una vez que ya se tiene la arquitectura de referencia deseada, en este punto, consideraron utilizar solo tecnologías que se presentan en la nube de Amazon Web Services (AWS), lo que lo hace valioso pensando que después la arquitectura de referencia presentada en este trabajo pueda ser implementada en ambientes totalmente de nube o híbridos.

Los autores también expresan en la arquitectura de referencia propuesta los elementos básicos que se deben tener en cuenta los cuales son las diferentes fuentes de datos, recolección, procesamiento y cargue de los datos, analítica de datos, interfaces, visualizaciones y agregan un elemento adicional de *Jobs* y especificación de modelos para la gestión de datos en tiempo real, algo que en la presente arquitectura de referencia de este trabajo no se trabajará, pero si está considerada para realizarse en algún trabajo futuro. Los autores no incluyen la gestión, administración y almacenamiento de resultados y flujos de trabajo de analítica lo cual es parte importante del presente trabajo para cumplir con los requerimientos de los científicos de datos.

El trabajo presentado por John Klein, Ross Buglak, David Blockow, Troy Wuttke, & Brenton Cooper (2016)[39] nos apoya en entender la relación que existe entre los requerimientos de cada dominio y las preocupaciones (*Concerns*) que se presentan en las diferentes capas de los dominios. Los autores presentan una batería de pruebas que se deben realizar al momento de generar una instanciación de la arquitectura de referencia lo cual no es realizado en el presente trabajo debido a que las preocupaciones arquitectónicas son totalmente diferentes.

También se tomaron referencias e ideas de trabajos donde el objetivo sea directamente el análisis de la apnea del sueño como el trabajo de Pépin, J-L, Bailly, S, Tamisier, R. (2020)[15], en el cual muestran la relación tan estrecha entre la recolección de datos, su administración y la detección de los diferentes trastornos ocasionados por la apnea del sueño como lo desarrollado en este trabajo. Otro punto por destacar del trabajo de Pépin, J-L, Bailly, S, Tamisier, R. (2020)[15] es la privacidad de los datos, se debe tener especial cuidado cuando se trabaja con datos sensibles de pacientes, pero también la consideración de tener muchos datos y su tratamiento, por lo cual se deben generar reglas de gobierno de los datos para cumplir con todos los estándares posibles. Esto último se genera en la presente arquitectura de referencia por medio del dominio de seguridad y administración.

Hay algunos artículos que se enfocan en la construcción de software y los patrones, otros prestan más atención a la forma de procesamiento de los datos clínicos, pero no encontramos un artículo que dentro de la arquitectura de referencia ayude a los usuarios a administrar los flujos de datos.

La diferencia que hace la arquitectura de referencia presentada en este trabajo a los trabajos anteriormente citados es la existencia de dominios que con su implementación ayudan a los científicos de datos a administrar las entradas de datos, generar filtros que ayudan a la carga de la información, administrar los flujos de analítica y sus resultados. Adicionalmente, se incluye el detalle de cómo fue construida y validada la arquitectura, así como una representación consistente de los modelos de cada vista.

I. DESARROLLO

En el desarrollo del presente trabajo se siguieron los pasos planteados inicialmente en la propuesta presentada anteriormente que consistían en una exploración inicial con resultados como modelos de datos, en este caso el de las variables seleccionadas de la base del HUSI debido a la necesidad puntual de extraer posibles variables que describieran características de fenotipos de los pacientes que pudieran llevar a aspectos de genotipos para que a partir de estas variables se pudieran llegar a la aplicación de métodos analíticos de aprendizaje profundo y de máquina. Se hace una separación de los modelos de datos de la fuente que en este caso es la base del HUSI y los otros modelos de datos del almacén son los propuestos como en la propuesta de la arquitectura de referencia como vistas lógicas de referencia en la organización de los datos. Adicionalmente, se realiza el diseño de una estrategia de selección de variables del modelo de datos fuentes.

Como paso siguiente se especifican las directrices arquitectónicas tenidas en cuenta con los requerimientos no funcionales y las diferentes preocupaciones de arquitectura, luego se presenta la propuesta de la arquitectura de referencia de grandes volúmenes de datos para analítica con sus respectivos dominios y componentes, además, se resaltan las diferentes vistas de referencia con respecto a la parte lógica, de aplicaciones, y física que se debe tener en cuenta de la arquitectura.

Algo que no estaba en la propuesta inicial del trabajo, pero que se adiciona debido a la gran importancia que tiene el ejercicio antes de realizar una implementación como prueba de concepto de la arquitectura, son las diferentes instancias diseñadas de la arquitectura de referencia para llegar así a la que finalmente se construyó; en donde se describen las ventajas y desventajas y demás detalles de cada iteración realizada.

La prueba de concepto se describe seguidamente de las instancias arquitectónicas con el debido alcance, descripciones de cada una de las implementaciones realizadas con anexos en documentación como manuales de instalación y de configuración y una sección de cómo podría utilizar la arquitectura un científico de datos. En esta documentación se incluye también la descripción específica para el gobierno de datos según los diferentes modelos de datos fuentes, almacén, analítica, seguridad y administrativos. También se incluye una descripción del componente de ETL construido y la descripción de cómo agregar nuevas variables.

Finalmente se presenta la validación de la arquitectura realizada con ATAM (*Architecture tradeoff analysis method*) y TAM (*Technology Acceptance Model*). Con ATAM se realiza el enfoque con los diferentes escenarios arquitectónicos propuestos y con TAM se realiza la validación directamente con los usuarios los cuales fueron los científicos de datos. La arquitectura también fue validada con la gerencia del HUSI de la PUJ en una reunión realizada. Y en la sección final de documentación se explican los diferentes anexos al proyecto.

1. Exploración de Datos Actuales

Se trabajó con la copia de una base de datos proporcionada por el Hospital Universitario San Ignacio (HUSI). Esta base de datos es llamada SAHI_PUJ, fue previamente tratada para anonimizar los datos sensibles y nos permitió aplicar diferentes criterios de selección que en conjunto con los científicos de datos se determinó cuál realmente era la información necesaria para extraer las variables fundamentales.

La base de datos proporcionada por el HUSI es la fuente principal de datos y cuenta con un total de 1329 tablas donde se almacenan los datos de las diferentes citas médicas, así como también el registro de los pacientes, atenciones, diagnóstico, entre otros y desde la perspectiva tecnológica, se realizó la configuración y montaje en un manejador *Microsoft SQL Server 2019*, esto es importante debido a que las consultas se construyeron para esta versión.

También se dio lugar a una reunión con una persona experta en el conocimiento de los datos del HUSI. En la reunión fue mostrada la interfaz gráfica en donde los médicos realizan su consultas y registros de información la cual fue de gran ayuda para identificar las principales tablas importantes según especificaciones proporcionadas por el científico de datos con relación a la identificación de variables que permitan describir fenotipos de los pacientes.

El científico de datos, como usuario principal de la arquitectura, especificó que uno de los objetivos era el de realizar la generación de vistas minables de analítica desde cualquier variable de la base de datos del HUSI. Se observó que la utilización de sentencias SQL era inevitable debido a la naturaleza de la base de datos relacional.

Frente a la falta del diccionario de datos, metadatos y al desconocimiento inicial para seleccionar las variables que realmente necesitan los científicos de datos de las 10.663 columnas totales de la base de datos, fue necesario realizar una exploración exhaustiva la cual estuvo conformada por tres procesos independientes que nos reflejarían posteriormente un modelo de datos origen específico en donde se conforman características fenotípicas en datos numéricos y de texto[7]; a continuación, se describen cada uno de los procedimientos.

1.1. Selección Manual de Variables por Fenotipos

Se realizó una comparación manual de toda la base de datos, tabla a tabla aplicando los siguientes criterios de selección:

1. Se ejecutaba una sentencia SELECT a cada tabla de las primeras 1000 filas (Ver Anexo 1).
2. Se visualizaba rápidamente cada título de las columnas de la tabla y se centraba una atención en particular en donde el título proporcionara indicios de descripción de fenotipos de los pacientes
3. De los datos no estructurados, como el texto en las descripciones, se leen hasta 20 filas aleatoriamente y de cada una sus primeras 10 a 15 palabras para evaluar si se habla de descripciones fenotípicas de cada paciente.

4. Cuando el valor de la columna es un número, se evalúa el significado del título y las primeras 10 filas si se trata de valores de características de fenotipos como peso, talla, presión arterial, entre otras.
5. Cuando el dato es un identificador a otra tabla, se ejecuta la siguiente sentencia SQL para identificar la tabla catálogo a la que pertenece, esto es debido a que en algunas tablas no existe la clave foránea a la tabla correspondiente (Ver Anexo 1).
6. En el caso en que el nombre de la columna sea de texto y todos los 1000 valores consultados son nulos, se realiza una consulta a esa tabla donde encuentre todos los valores no nulos o con tamaño de caracteres mayor a cero para poderla revisar con más detalle.
7. Para detectar si la tabla era transaccional, se evaluaba la cantidad de registros por encima de 1000 los cual consideramos que es la cantidad considerable para evaluar humanamente tablas de más de 10 millones de registros como la de hceConsulta; también se revisa su naturaleza para que fuera valorada y candidata para seleccionar sumando los criterios anteriores, pero si se evidenciaba que era una tabla catálogo con menos de 1000 filas, se listaban por separado para luego revisar si existía la relación con las tablas padre que tuvieran sus respectivas claves foráneas y que también cumplan los criterios anteriores.
8. No se revisaban las tablas de funcionamiento del sistema, ni las que contenían parámetros de configuración.

1.2. Algoritmo de Búsqueda por Diccionario de Palabras

Se utilizó un algoritmo en el lenguaje T-SQL que permitió buscar en toda la base de datos el contenido de las tablas y las columnas las palabras: “apnea” dando como resultado 144 tablas, “polisomnografía” la cual fue encontrada en 77 tablas y la palabra “sueño” que se encontró en 207 tablas. Ver Anexo 6. El algoritmo toma en cuenta la ortografía de la palabra y las minúsculas, por lo cual se convirtió todo a minúsculas para que no quedaran tablas excluidas.

Después de haber acotado el rango de búsqueda de las tablas, se realizó una comparación de sus nombres, lo cual permitió hacer un cálculo si una tabla estaba en más de un grupo de los encontrados. Luego se realizó una validación manual revisando si el nombre de la tabla y sus columnas podrían describir características fenotípicas para proceder a su selección.

1.3. Selección de Tablas por Características

En este proceso se utilizaron consultas en T-SQL (Ver Anexo 1) las cuales nos brindaron datos de la estructura lógica y el tamaño físico de las distintas tablas dentro de la base de datos SAHI_PUJ, aunque estos procesos nos permitieron descubrir cuales tablas eran las que utilizaba el sistema, así como también para indicar qué columnas contenían información, se puede destacar de esta exploración que de las 1329 tablas solo 1094 contenían datos.

Finalmente se combinaron los tres procesos para determinar con exactitud las tablas y columnas (variables) seleccionadas para realizar el modelo de datos actual y así los científicos de datos lo obtengan como referencia para la selección de variables que entrarán a la arquitectura de grandes datos y así poder generar diferentes modelos con diferentes variables en diferentes vistas minables.

En detalle, se realizó la unificación de todas las variables seleccionadas entre las diferentes tablas consultadas. Se tomaron las tablas resultantes de los tres procesos y se cruzaron; y las tablas con mayor frecuencia de aparición fueron seleccionadas inmediatamente, es decir que si había una tabla que aparecía igual en los tres listados de tablas era seleccionada inmediatamente, para las demás se debió de realizar nuevamente una observación manual del nombre de la tabla, sus columnas y si es necesario, el contenido para que fuesen seleccionadas como variables, ésta última revisión explicada se hizo exhaustivamente para las que tenían frecuencia de aparición en uno de los tres listados.

1.4. Modelo de Datos de Fuentes.

El modelo de datos extraído de la base actual se puede observar en el Anexo 2 y se compone de 135 tablas aproximadamente en donde las principales son: admCliente, admAtención, hceConsulta formando así una de las relaciones iniciales que permiten una primera aproximación al entendimiento del modelo; el cliente es el paciente que puede tener muchas atenciones y cada atención puede contener múltiples consultas. Las tablas de consultas conectan casi todas las demás tablas.

En el modelo se realizó la separación por diferentes grupos entre los que se destacan los siguientes:

- Consultas de pacientes en color verde,
- Atenciones con otras tablas en donde sus variables definen características de una atención a un paciente todas en color anaranjado,
- Tablas de diagnósticos en color púrpura
- Tablas de esquemas de atención atadas también a un conjunto más grande del modelo que son las historias clínicas electrónicas en color amarillo neón
- Los antecedentes en color púrpura claro
- Citas a consultas y exámenes en color rojo claro
- Procedimientos y productos en color azul celeste
- Epicrisis en color amarillo claro

Se pueden extraer más grupos, pues el diagrama es muy extenso y complejo de comprender. Adicionalmente, en el diagrama se observa que hay unos pequeños círculos de colores, éstos indican, que las variables que contengan estos círculos son porque en ellas se contienen datos de texto en donde se menciona la palabra que tiene cada color, es decir, la palabra “Polisomnografía” con tilde se encontró contenida en las variables que tengan el círculo de color verde y así con las demás. Se tomaron las siguientes palabras para la búsqueda:

- Polisomnografía – círculo verde
- Polisomnografía – círculo azul
- SAHOS, Apnea y Sueño – círculo amarillo

En la búsqueda se utilizaron las sentencias SQL del Anexo 1 y luego de tener un conjunto más específico, se determinó con el científico de datos las diferentes variables con el símbolo de *check* en el diagrama del modelo de datos fuentes. En este punto ya es clara la selección de

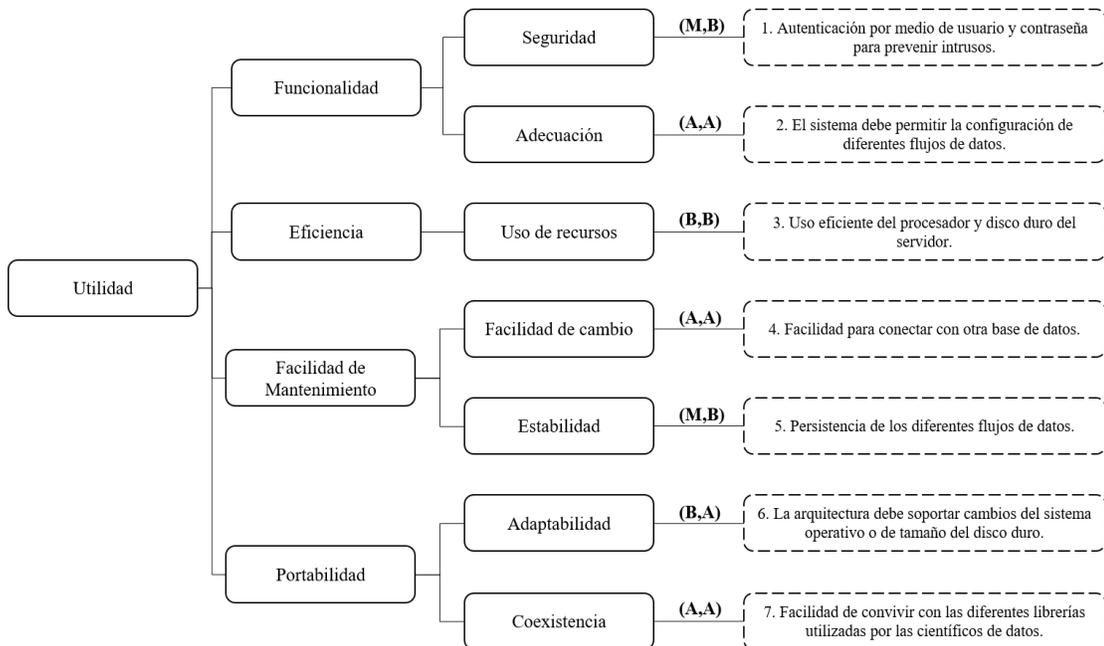
variables de la exploración para comenzar a mapearlas en la prueba de concepto que se tiene planeada realizar.

2. Directrices Arquitectónicas

La principal inquietud de los científicos de datos es contar con una arquitectura que permita la ejecución de las diferentes pruebas de concepto y que al mismo tiempo pueda obtener datos de diferentes fuentes y centralizarlas para su análisis. La arquitectura de referencia fue gradualmente generada como parte de un proceso de diseño el cual permitió ir probando diferentes tecnologías que se adaptaran mejor a la necesidad de los científicos de datos.

2.1 Atributos de Calidad

Existen diferentes modelos donde los atributos de calidad han sido definidos, pero para obtener la arquitectura de referencia consideramos como base la norma ISO/IEC 9126, con ayuda de los científicos de datos realizamos un árbol de utilidad el cual informa de cuales atributos de calidad son los conductores para la arquitectura de referencia.



Donde se prioriza en un modelo de dos dimensiones las cuales son la necesidad por los científicos de datos y el riesgo tecnológico de implementarlo, los valores para priorizar fueron Alto, Medio y Bajo.

Importancia / Riesgo Técnico	Alto	Medio	Bajo
Alto	2, 4, 7	-	-
Medio	-	-	1, 5
Bajo	6	-	3

Si bien todos los puntos fueron importantes, se puso especial atención en los atributos de calidad que se encuentran en la región (Alto, Alto) los cuales son adecuación, facilidad de cambio y coexistencia.

2.2 Restricciones Arquitectónicas

Las restricciones arquitectónicas (*architectural concerns*) se dividieron en dos áreas, las generales que son preocupaciones que son transversales a toda la arquitectura y las específicas estas son muy puntuales, tienden a ser más técnicas y solo afectan a un dominio de la arquitectura.

Específicas

El Hospital San Ignacio (HUSI) y la base de datos SAHI_PUJ fue implementada bajo la tecnología del manejador *Microsoft SQL Server*, esta es una herencia que debe adoptar la arquitectura y al mismo tiempo debe ser capaz de integrarse con otros manejadores de bases de datos relacionales.

Generales

El software utilizado debe ser código abierto, y con licenciamiento del tipo *GNU General Public License (GPL)* o el tipo *Apache License*, aunque también se consideran tecnologías que no cuenten con licenciamiento de código abierto.

El aprovisionamiento de infraestructura, esto es fundamental ya que se debe seleccionar cual es el ambiente donde va a vivir la arquitectura, para esto se cuenta con una restricción por la oferta que puede otorgar la universidad sin incurrir en costos adicionales, este aprovisionamiento se rige bajo las reglas de la facultad de ingeniería de la Universidad Javeriana.

II. ARQUITECTURA DE REFERENCIA PROPUESTA

Esta sección describe la arquitectura de referencia (AR) propuesta diseñada para facilitar múltiples implementaciones concretas de la misma. El diseño se enfoca en resaltar fuentes de datos, procesos de ingesta de datos, almacenamiento, preprocesamiento, flujos de trabajo de analítica, visualización de datos, la seguridad y administración. Se realiza, además, una profundización de la AR en diferentes vistas enmarcadas en la arquitectura lógica, de software y física.

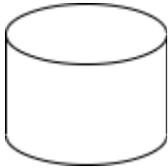
1. Arquitectura de Referencia de Grandes Volúmenes de Datos y Analítica

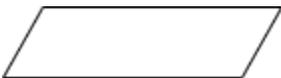
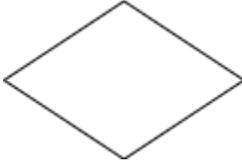
Esta es una arquitectura de referencia de grandes volúmenes de datos debido a la gran cantidad de información gestionada en las historias clínicas de los pacientes atendidos en las diferentes entidades de salud, las múltiples fuentes de datos que pueden existir y el procesamiento que se requiere para la aplicación de diversos métodos analíticos de aprendizaje profundo y de máquina que faciliten la predicción de diagnósticos.

La presente arquitectura de referencia es la guía para realizar la implementación concreta de múltiples instancias de arquitectura como solución a algún problema específico de grandes volúmenes de datos y que permite analizar todo el espectro del problema facilitando la segmentación de éste y una correcta medida de efectividad en el momento de la implementación.

1.1 Notaciones de la Arquitectura de Referencia

Antes de presentar la AR propuesta, se inicia con la explicación de la notación de cada uno de los elementos gráficos creados para describir los diferentes componentes de la arquitectura. Se han definidos nuestros propios elementos de notación de la arquitectura y también teniendo en cuenta la definición realizada por Sang, Xu, y De Vrieze[43] en donde también es definida por los autores teniendo en cuenta el flujo, procesamiento y análisis de los datos.

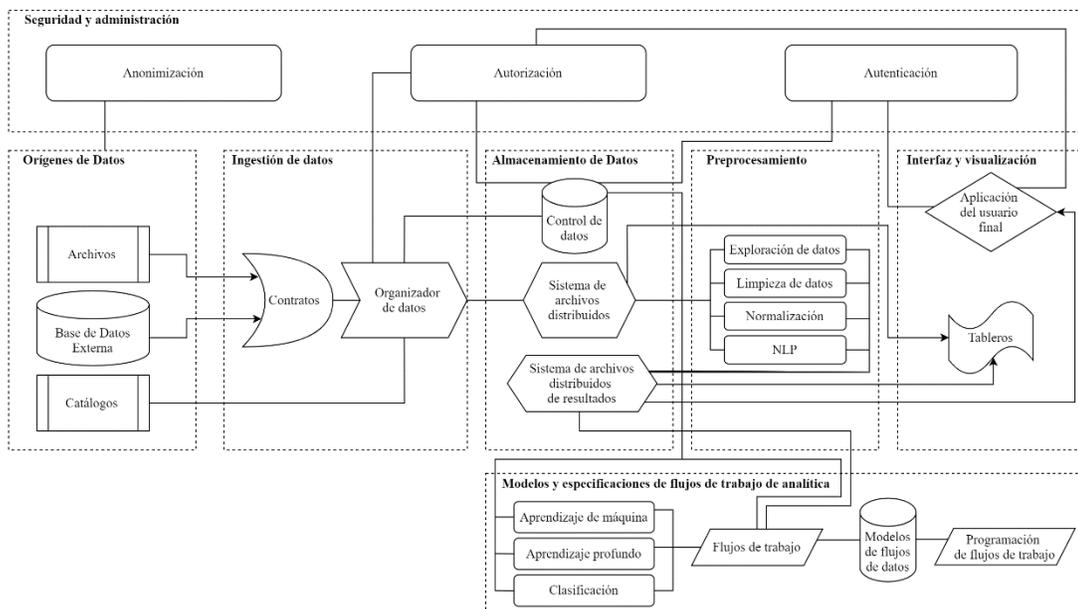
Núm.	Notación	Descripción
1		La representación de un almacén de datos que puede ser relacional o no relacional
2		Esta notación representa los archivos que cuentan con la información que puede ser parte de la ingesta de datos en el sistema.

3		<p>Esta notación representa la forma lógica de los contratos de recepción de datos para que puedan ser parte de la ingesta en el sistema.</p>
4		<p>Esta notación representa el organizador de la información. Aquí se define en dónde serán persistidos los datos y también mecanismos de administración de los contratos. Involucra conceptos de extracción, transformación y cargue de datos.</p>
5		<p>Esta notación representa el almacenamiento de datos en el sistema de archivos distribuidos.</p>
6		<p>Esta notación representa una implementación para el procesamiento lógico de datos.</p>
7		<p>Esta notación representa los flujos de trabajo que contienen diferentes implementaciones que permiten el procesamiento y almacenamiento físico o lógico de los datos.</p>
8		<p>Esta notación representa los reportes o resultados finales que se obtienen de los datos que se encuentran en los diferentes almacenes de datos.</p>
9		<p>Esta notación representa las aplicaciones de interfaz de usuario que pueden observar los científicos de datos para el control y supervisión de las diferentes transformaciones de los datos.</p>

10		Esta notación representa el flujo de la información. Debido a que no tiene dirección, se entiende que es bidireccional.
11		Esta notación representa el flujo de la información; ésta representa también el destino de esta.
12		Representa un dominio específico de un conjunto de componentes.

1.2 Arquitectura de Referencia Propuesta

De acuerdo con las notaciones descritas anteriormente, se presenta el diseño de la arquitectura de referencia para grandes volúmenes de datos y analítica que servirá como referencia para la implementación y la generación de múltiples propuestas arquitectónicas concretas.



La arquitectura se divide en siete dominios diferentes que permiten ver la separación física y lógica de la información las cuales son: Seguridad y Administración, Orígenes de Datos,

Ingestión de datos, Almacenamiento de Datos, Preprocesamiento, Interfaz y Visualización, y Modelos y especificaciones de flujos de trabajo de analítica.

1.2.1. Dominio de Seguridad y administración

Este dominio es transversal a todos los dominios para la proyección de los datos. Posee tres componentes, el primero es para gestionar la anonimización de los datos de entrada a la arquitectura, de ahí su relación con el dominio Orígenes de datos. El objetivo principal de este componente es realizar una conversión de los datos para que no exista la posibilidad de identificar a las personas en la base de datos.

Los siguientes dos componentes son Autorización y Autenticación. La autenticación debe partir desde un usuario y una contraseña que debe ser almacenada en la base de datos totalmente encriptada. Los algoritmos de encriptación deben ser sofisticados como por ejemplo un RSA con llave privada y pública. Al igual se deben contemplar las tareas de gestión de usuarios y cambios de contraseña. Una vez sea realiza la autenticación se debe generar un token encriptado y con tiempo de expiración determinado. La autorización se debe realizar en cada subcomponente que se construya en este elemento, pues cada uno de ellos va a tener contacto directo con los datos, por eso la primera validación que se debe hacer es la del token de entrada lo cual permite la autorización para acceder a esos datos. Para ampliar más el contexto, en este dominio de seguridad y administración se debe seguir un procedimiento de autenticación de inicio de sesión único o también llamado *Single Sign On*.

1.2.2. Dominio de Origen de Datos

Las fuentes de los datos pueden ser diversas, van desde modelos relacionales a no relacionales y desde diferentes motores tecnológicos de gestión de datos. Este dominio busca representar las diferentes bases de datos externas que podrían integrarse, pero si se presentan dificultades en esta conexión, se puede optar por la opción de integración a través de archivos de texto plano definidos por el componente de contratos que permita la ingesta correcta de los datos y su organización.

Las bases de datos externas pueden ser configuradas con diferentes conectores para integrarlas al dominio de ingestión de datos y realizar las ejecuciones de consultas y extracción de los datos correspondientes según su respectivo modelo de datos. La base de datos externa puede estar en la nube o en algún otro nodo que permita la conexión directa desde el dominio de ingestión de datos. También se debe revisar el tipo de base de datos para conocer la naturaleza en la construcción de las consultas para el caso que sean SQL o NoSQL.

También se puede optar por un sistema de archivos o un repositorio en donde las entidades de salud o los usuarios puedan alojar archivos de texto plano con previo acuerdo de estructura en su contenido para que la ingesta de datos pueda ser efectiva.

El componente de catálogos representa aquellos archivos también de texto plano que ya han sido predefinidos y preconfigurados por el científico de datos, los cuales le ayudan a optimi-

zar sus procesamientos, por ejemplo, se podrían encontrar vectores de palabras, diccionarios de datos, ontologías, etc.

1.2.3. Dominio de Ingestión de Datos

Este dominio tiene dos componentes que deben garantizar la extracción, transformación y cargue de datos (ETL). Inicialmente, se debe realizar un perfilamiento de los datos con estructura, tipos y modelos de datos, diccionarios y demás artefactos que ayuden a definir reglas de extracción inmersas en el componente de Contratos.

El organizador de datos debe orquestar las operaciones de ETL. La propuesta es que sea posible en una primera instancia estudiar el origen de los datos para que se desarrollen técnicas de extracción, creación y configuración de conectores a las diferentes fuentes de datos, mapeo y transformaciones necesarias de acuerdo con la naturaleza de los datos de entrada y a los que sean necesarios en la presente arquitectura para lograr su almacenamiento y procesamiento.

Se recomienda implementar estrategias de mapeo diferentes cuando sean fuentes de datos de archivos, es decir, definir si cada archivo será una tabla o una colección o alguna otra entidad y si el contenido de éste serán los datos y atributos organizados por reglase definidas en los contratos. Para el caso de las bases de datos externas se debe establecer la información de la conexión a la base, ejecución de consultas SQL o NoSQL y también tipos de datos. Las reglas de ETL deben ser almacenadas en la base de datos de control, es aquí en donde se debe realizar la implementación de estrategias de mapeo de datos. Para el caso de los archivos catálogos, el organizador podría realizar algún tipo de transformación o simplemente pasar los archivos directamente al sistema de archivos distribuido.

1.2.4. Dominio de Almacenamiento de Datos

Representa el conjunto de componentes que permiten la persistencia de los datos. Se compone de una base de datos de control que puede ser relacional o no relacional y el conjunto de sistema de archivos distribuidos.

El componente que representa la base de datos de control tiene como objetivo almacenar la información que facilite la gestión del organizador de datos, contratos y otros componentes en los dominios de preprocesamiento y flujos de trabajo como reglas de ETL y datos que permitan administrar los resultados obtenidos por los científicos de datos. Además, se deben almacenar datos que ayuden a la interacción entre el componente de aplicación de usuario final y el organizador de datos como reglas y parámetros de autenticación, autorización y anonimización.

Los sistemas de archivos distribuidos son representados en la arquitectura como componentes en donde se deben almacenar los archivos, catálogos y resultados de las implementaciones lógicas de analítica realizadas por los científicos de datos. Este componente debe permitir el almacenamiento, recuperación, descarga y procesamiento de esos archivos.

La diferencia entre el sistema de archivos distribuidos normal con el de resultados es que el primero es en donde se almacenarán los datos provenientes de las operaciones de cargue que realice el organizador de datos. El segundo almacenará los resultados obtenidos por los científicos los cuales son datos estructurados en archivos provenientes de los métodos y aplicaciones de analítica realizados en las etapas de exploración, preparación y modelamiento de datos. Estos resultados pueden ser vistas minables o datos para generar representaciones gráficas, también datos de operaciones de limpieza, normalización de datos y procesamiento de lenguaje natural o resultados obtenidos luego de la aplicación de algoritmos de aprendizaje profundo y de máquina. La ventaja de almacenar los resultados es que pueden ser recuperados en diferentes etapas de los flujos de trabajo de analítica para facilitar las tareas del científico de datos.

1.2.5. Dominio de Preprocesamiento de Datos

En este dominio se deben desarrollar operaciones de exploración, limpieza, normalización, procesamiento de lenguaje natural (NLP) y tratamiento de datos que corresponden a un procesamiento previo a la aplicación de los modelos analíticos.

En el componente de exploración de datos, se pueden implementar consultas al sistema de archivos distribuidos para conocer las vistas minables preestablecidas, tipos de datos, distribuciones, relaciones entre diferentes variables, significado y descripciones de variables, análisis estadísticos y validación de la calidad de los datos. Los resultados obtenidos en este componente podrán ser almacenados en el sistema de archivos distribuido destinado para este fin.

El componente de limpieza de datos está destinado a permitirle al científico de datos realizar tareas de inserción, construcción y/o eliminación de valores predeterminados para mitigar o no los posibles datos faltantes. En este punto ya habría como resultado otra selección de un subconjunto de datos limpios que pueden ser almacenados en la base de datos de control y de resultados para retomar posteriormente.

El componente de normalización permite tareas de conversión, transformación, combinación e integración de datos. Aquí se pueden obtener nuevas variables o nuevos conjuntos de datos que podrían ser almacenados. La normalización también puede contener tareas de alistamiento de datos para ingresar al dominio de modelamiento y flujos de trabajo.

El componente NLP (*Natural Language Processing*) hace referencia a los diferentes métodos para trabajar el texto como por ejemplo la detección y clasificación de lenguaje y sentencias, las técnicas de tokenización, reconocimiento de entidades nombradas, categorización de documentos, *Part of Speech* que es la clasificación de palabras según su subtipo, lematización y la definición de tópicos para detectar ontologías y diferentes temas agrupados en los textos de las historias clínicas de los pacientes según el caso de estudio del presente trabajo.

Para finalizar esta sección, se debe aclarar que el orden de los componentes no es una definición estricta. Es posible que el científico de datos decida establecer diferentes prioridades de

ejecución de componentes según el problema que esté desarrollando, incluso podría omitir alguno de los componentes mencionados en este dominio de arquitectura.

1.2.6. Dominio de Interfaz y Visualización

Este es el dominio de la arquitectura que tiene interacción de una interfaz de usuario gráfica con los científicos de datos para la administración y ejecución de muchas de las tareas que son configuradas en los dominios anteriores.

El componente de aplicación del usuario final representa las diferentes interfaces de usuario que se puedan desarrollar para permitirle al usuario ingresar, actualizar o manipular los datos de configuración que permitan administrar los componentes de los otros dominios de la arquitectura y que tengan datos almacenados en la base de datos de control. Efectivamente se deben contar con tareas de autenticación y autorización para controlar la seguridad y la privacidad de los datos.

Los tableros hacen referencia a aplicaciones para visualizar gráficamente y de forma interactiva los datos trabajados por el científico de datos, también generación de informes y paneles de control.

1.2.7. Dominio de Modelos y Especificaciones de Flujos de Trabajo de Análítica

Este dominio específico es para permitirle al científico de datos realizar las implementaciones para aplicación de métodos analíticos de aprendizaje profundo y de máquina, y dependiendo del tipo de instanciación de la arquitectura, aquí también se podrían ejecutar los componentes del dominio de preprocesamiento.

Los flujos de trabajo o también llamados “*pipelines*” de analítica pueden contener diferentes implementaciones secuenciales y/o en paralelo previamente programadas en donde se puede visualizar el código fuente y los diferentes flujos que encapsulan las piezas de código que cumplen tareas específicas como las tratadas en el dominio de preprocesamiento y también los métodos de aprendizaje profundo, de máquina y de clasificación. Al programar el código de los flujos de trabajo, el mantenimiento es más fácil, pues estos pueden ser controlados y almacenados en una base de datos de modelos de flujos de datos.

Los componentes de aprendizaje de máquina al igual que el de aprendizaje profundo representan todas aquellas implementaciones de algoritmos de modelamiento supervisado y no supervisado. Para el aprendizaje de máquina se podrían implementar algoritmos como por ejemplo regresiones, Naivy Bayes, árboles de decisión, entre otros más, y para el aprendizaje profundo diferentes tipos de algoritmos de redes neuronales. El componente de clasificación debe poder seleccionar el modelo predictivo resultante y realizar la respectiva clasificación de los datos nuevos, por ejemplo, luego de tener un modelo predictivo para determinar si un paciente tiene o no apnea del sueño, se le debe ingresar la historia clínica del paciente y realizar la clasificación de si tiene o no la enfermedad.

Los tres componentes generan resultados que pueden ser almacenados en el sistema de archivos distribuido y en la base de datos de control al igual que los parámetros e hiperparámetros utilizados en el proceso de modelamiento. En este dominio de flujos de trabajo se pueden crear diferentes instancias que a futuro pueden ejecutarse en paralelo, cíclicamente o a demanda; el científico de datos puede tener múltiples flujos de trabajo ejecutándose al mismo tiempo incluyendo análisis de datos en tiempo real.

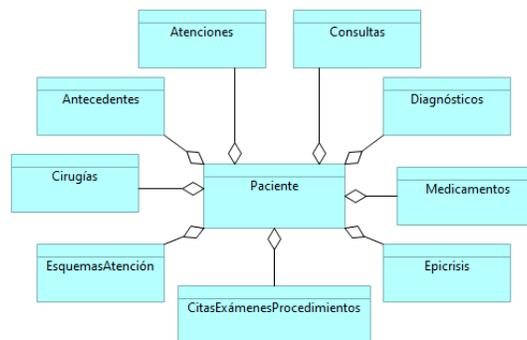
La decisión de mantener por separado los componentes del dominio de preprocesamiento de datos con los de aprendizaje profundo, aprendizaje de máquina y clasificación, es porque es posible que ya existan herramientas tecnológicas de solo integración y no desarrollo para realizar tareas iterativas de limpieza de datos, normalización y NLP. Y si se presenta construcción de componentes a partir del desarrollo de software se puedan realizar de manera independiente y modular haciendo que cada componente sea agnóstico a la tecnología implementada y que solo importen las entradas y salidas de datos.

2. Arquitectura Lógica

La arquitectura lógica del sistema se presenta a continuación con diferentes vistas que representan la estructura de la información para el almacén, la administración, los resultados y la ETL de los datos.

2.1. Modelo de Datos del Almacén

El modelo de datos se realiza definiendo agrupaciones de variables que pueden ser creadas dinámicamente por el científico de datos de acuerdo con las vistas minables que desee generar por medio del organizados de datos y lo podrá realizar tomando como referencia las variables del modelo de datos de fuentes. El modelo de datos del almacén propuesto tiene una estructura de un modelo estrella y se compone de las siguientes agrupaciones orientadas al Paciente:



Cabe resaltar que esta definición es totalmente dinámica, pero debe siempre girar en torno al paciente, es decir, el modelo aquí planteado es solo una propuesta de cómo puede quedar porque el científico de datos puede seleccionar las variables y entidades con las que trabajará.

En esta propuesta se definen las siguientes entidades del modelo:

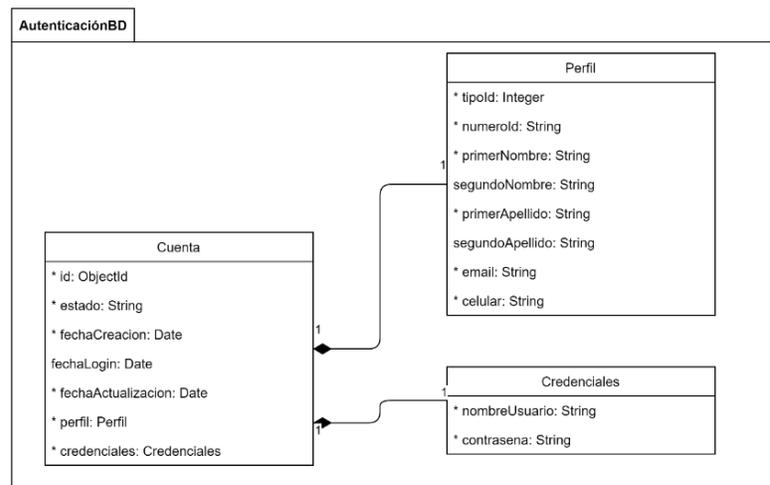
- Paciente (Tabla de hechos): Entidad que contiene las variables que caracterizan a un paciente como fecha y lugar de nacimiento, edad, peso, talla, etc.

Las siguientes son las tablas de dimensiones:

- Atenciones: Son los diferentes servicios de atención adquiridos por un paciente con fecha de ingreso y de salida, tipo de atención y también se puede incluir variables de otras tablas como cantidad de consultas, por ejemplo, para que a la hora de procesarlas no se deban realizar consultas a otras entidades.
- Consultas: Se compone de las variables para los diferentes tipos de consulta como son de anestesia, obstétrica, psiquiatría, incapacidad, circunstancias asociadas a la atención, signos vitales, etc.
- Diagnósticos: En esta entidad se pueden incluir variables con respecto a la identificación de diagnósticos por consultas de cada paciente con el código CIE10.
- Medicamentos: Contiene variables relacionadas con los tipos, nombres, posologías y descripciones de los medicamentos recetados al paciente.
- Epicrisis: Se compone de variables de observaciones y descripciones del resumen médico del paciente, fechas de epicrisis y también se pueden incluir diagnósticos de cada paciente por atención.
- CitasExámenesProcedimientos: Agrupación que posee las variables relacionadas con fechas y observaciones de citas médicas con sus motivos de cancelación si es el caso, tipos de exámenes y procedimientos de los pacientes.
- EsquemasAtención: Son las descripciones de los esquemas por atención de cada paciente en donde se pueden incluir variables como sus medicamentos, dosis, peso, talla, entre otras observaciones.
- Cirugías: Contiene las variables asociadas a los pacientes por atenciones quirúrgicas como el tipo de cirugía y sus respectivas descripciones.
- Antecedentes: Son todos los antecedentes médicos relacionados a un paciente y puede tener variables como la identificación y el tipo de antecedente y su descripción.

2.2. Modelo de Datos de Seguridad

A continuación, se presenta el modelo de datos de administración y configuración que tiene como alcance la seguridad, facilidad de mantenimiento, de cambio y estabilidad.

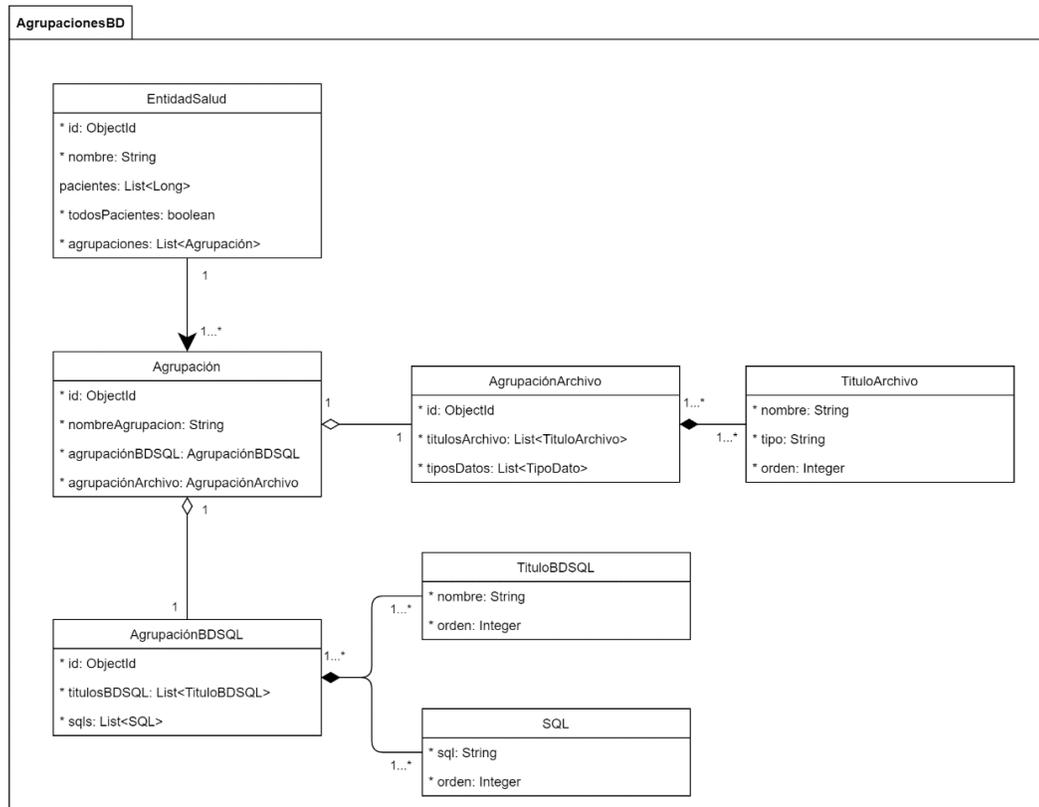


Este modelo a diferencia del anterior es un poco más detallado debido a que se incluyen como referencia mínima de seguridad, los atributos de las entidades propuestas. Cada cuenta de un usuario se compone de su respectivo perfil y sus credenciales para ingresar a la aplicación de la arquitectura por su respectiva interfaz de usuario (UI) que le permitirá gestionar los datos, seleccionar variables y administrar la generación de vistas minables. La Cuenta puede tener varios estados como Activa, Inactiva o Bloqueada; también tiene atributos de auditoría como fecha de creación y actualización y la fecha de su última autenticación (fechaLogin) la cual no es obligatoria inicialmente debido a que cuando se registra el usuario no se especifica una autenticación.

El perfil contiene datos personales básicos del usuario como tipo y número de identificación, sus nombres y apellidos, correo electrónico y celular para tener en cuenta para futuros mecanismos de comunicación. Las credenciales se componen de atributos básicos como el nombre de usuario y la contraseña que será almacenada con el algoritmo de encriptación como por ejemplo SHA512 y validada de la misma manera para conservar el atributo de privacidad de los datos.

2.3. Modelo de Datos de Administración y Organización

El siguiente modelo corresponde a la configuración y selección de variables para la extracción de las vistas minables:



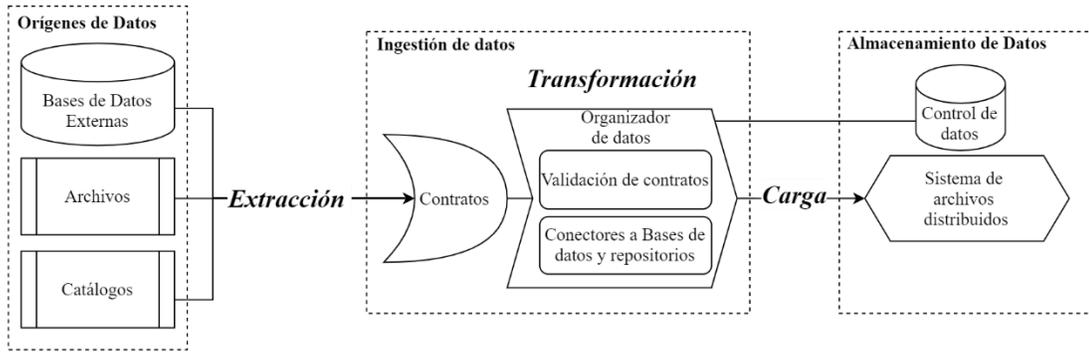
El modelo de agrupaciones tiene como función almacenar la configuración que permita el funcionamiento del modelo de ETL en los componentes organizador de datos y contratos. Se encuentra **EntidadSalud** con su nombre, el listado de pacientes a los que se les desee incluir en la extracción o la propiedad `todosPacientes` para que la ETL sea ejecutada en todos los pacientes de la base. También existe un listado de agrupaciones en donde cada una se define si es por archivo o por sentencias SQL a una base de datos relacional como la del caso de estudio del HUSI. El nombre de la agrupación definirá el nombre del archivo que resultará del proceso de ETL y siguiendo los conceptos mencionados en el modelo de datos del almacén, son también los nombres de la tabla de hechos o dimensiones.

Si es una agrupación de base de datos con sentencias SQL, se define un listado de títulos que son los nombres de las variables seleccionadas en un orden predeterminado. En el listado de sentencias SQL se incluyen múltiples variables y tablas teniendo en cuenta el orden de las variables de salida en la cláusula `SELECT` con el orden de los títulos configurados, esta parte al igual que todo el procedimiento para la utilización de la arquitectura se explica con más detalle en el Anexo 3 que es el documento técnico, administrativo y de configuración.

Las agrupaciones de tipo archivo, se presentan cuando se requiere realizar la integración con otros clientes sin importar la base de datos y se debe definir un contrato que ayudará a dicha integración. **AgrupaciónArchivo** tiene un listado ordenado de títulos o nombres de variables y un listado de tipos de datos por el mismo orden correspondiente.

2.4. Modelo de ETL (Extracción, Transformación y Carga) de Datos

Atendiendo los atributos calidad para la facilidad de cambio y de mantenimiento, se realiza el siguiente diseño de ETL que permite la integración de diferentes fuentes de datos para diferentes instancias de la arquitectura de referencia que será propuesta.



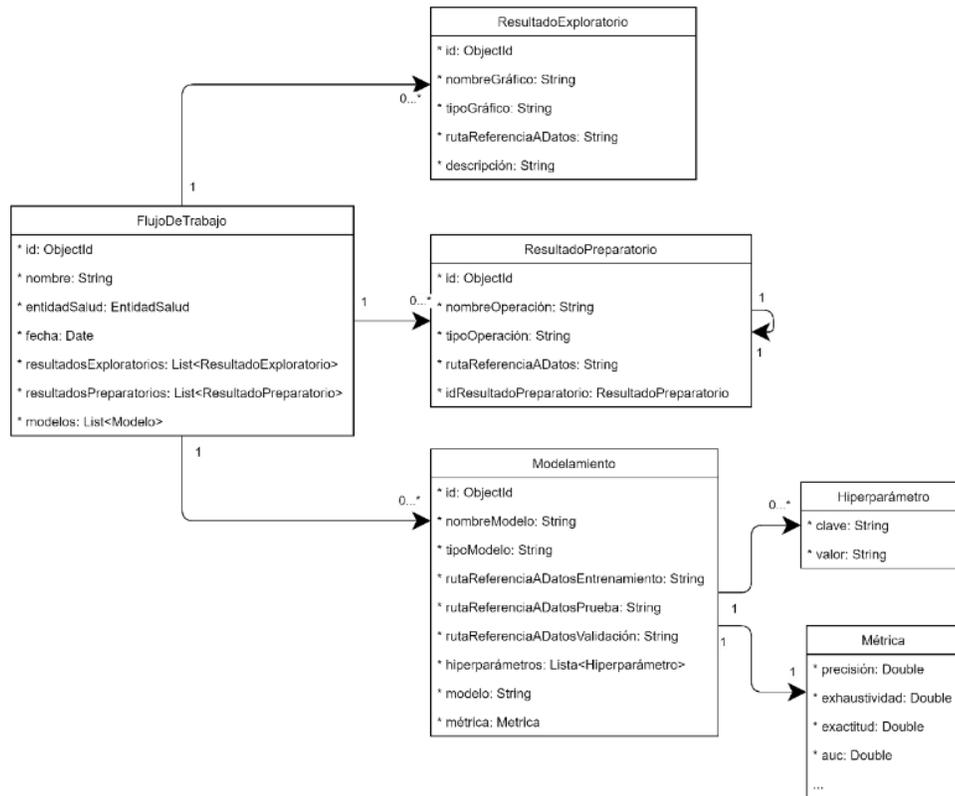
El modelo muestra las tres operaciones principales de un proceso de integración de datos: extracción, transformación y carga.

El proceso de ETL puede tener múltiples orígenes de datos. Cuando se desea cargar información de bases de datos que no se tienen configuradas en los conectores del organizador de los datos, se debe optar por generar los contratos en archivos, enviarlos a la entidad propietaria de los datos que se van a cargar la cual debe construir los Archivos con los datos necesarios de acuerdo con los contratos compartidos. Los archivos pueden estar en formato CSV (*comma-separated values*), el organizador podrá validar los datos ingresados en los archivos y realizar el proceso de extracción, transformación y carga en el sistema de archivos distribuidos. Para que el organizador de datos realice estas tareas de validación y conexión a los orígenes de datos, debe consultar las reglas y parámetros configurados previamente en la base de control de datos.

Para el caso en que sea posible realizar la conexión directa entre el organizador de datos y el motor de base de datos de la entidad propietaria, sea una base de datos estructurados o no, todos los parámetros de conexión deben ser almacenados en la base de control de datos al igual que las sentencias de consulta SQL o NoSQL que le permitirán al organizador de datos ejecutarlas y así realizar la extracción, transformación y carga de los datos.

2.5. Modelo de Datos de Resultados

El siguiente modelo lógico de datos hace referencia a la estructura en la que se pueden soportar los datos relacionados con los resultados obtenidos por el científico de datos. Los objetivos son llevar un histórico de ejecuciones, recuperar los modelos desde un punto anterior y validaciones o evaluaciones las mejores métricas según los modelos analíticos de aprendizaje profundo o de máquina aplicados.



El elemento que representa al flujo de trabajo puede contener como atributos un nombre, una fecha de ejecución, estar asociado a la Entidad de salud a la cual se le está realizando el proceso de analítica sobre los datos proporcionados. También se tienen como atributos tres listados que son los resultados exploratorios, preparatorios y de modelos.

Los resultados exploratorios corresponden a los datos obtenidos en la fase de exploración que realiza el científico de datos en donde se pueden obtener diferentes tipos de gráficos a partir de tablas con datos estructurados que posteriormente pueden ser almacenados en el sistema distribuido de archivos de resultados y referenciar en `rutaReferenciaADatos` la ruta a ese archivo con los datos para el gráfico.

Los resultados preparatorios corresponden a los registros de las operaciones de preprocesamiento de datos realizadas antes de ingresar al modelamiento las cuales corresponden a limpieza de datos, transformaciones, normalizaciones o NLP. De estas operaciones se generan resultados en archivos de datos que serán almacenados en el sistema de archivos distribuido de resultados. Debido a que el resultado de una operación de preparación puede ser el resultado de otra operación de preparación, en el campo `idResultadoPreparatorio` se referencia el resultado preparatorio anterior para mantener esa relación en cadena y retomar los resultados en ese punto.

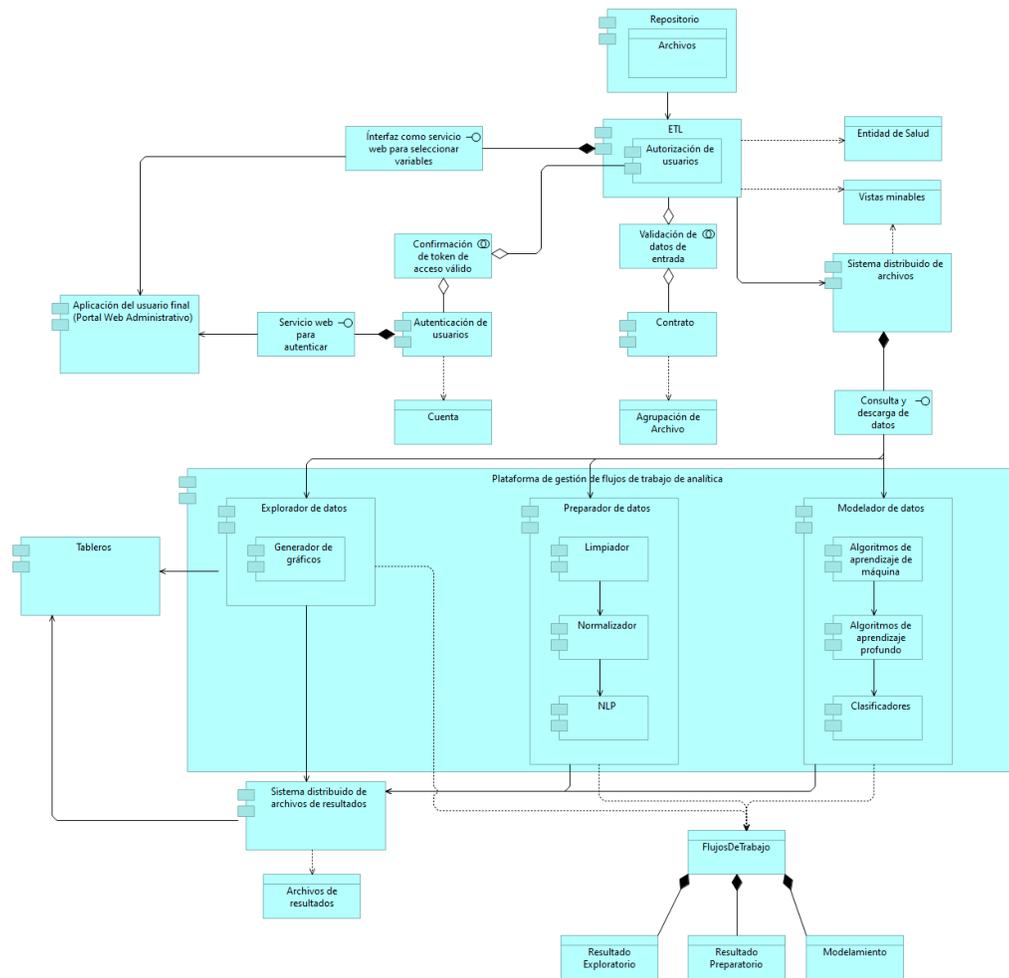
Los modelos hacen referencia a los diferentes métodos de analítica aplicados como aprendizaje profundo y de máquina. Se almacena información como el nombre y tipo de modelo, la ruta del archivo de datos de entrenamiento, prueba y validación almacenados en el sistema de archivos distribuido de resultados; los hiperparámetros utilizados en los diferentes modelos ejecutados, el modelo obtenido y las métricas que resultaron de esos modelos como precisión, exhaustividad (*recall*), exactitud (*accuracy*), AUC (Área bajo la curva ROC), etc. Esto permite que el científico de datos pueda realizar consultas, validaciones y evaluaciones de los diferentes modelos aplicados posteriormente.

3. Arquitectura de Aplicaciones

Siguiendo la especificación de *Archimate*[42], a continuación, se presentan los dos puntos de vista que se consideraron más importantes para representar la estructura tienen las aplicaciones y su uso por el usuario.

3.1. Punto de Vista de Estructura de Aplicaciones

El presente punto de vista representa la forma en que se estructuran las aplicaciones o componentes y es útil principalmente para entender la posible cantidad de aplicaciones que se deben adquirir, construir o configurar, además de la interacción que puede haber entre ellas.



En el diagrama se observa que los elementos son de color azul porque representan la capa de aplicaciones de *Archimate*. El repositorio facilita las operaciones con los archivos al componente ETL el cual realiza tareas de autorización de usuarios, validación de datos de entrada definidos en los contratos. ETL tiene una interfaz para ofrecer el servicio de selección de variables a la Aplicación del usuario final que a su vez consume los servicios ofrecidos desde una interfaz del componente de autenticación.

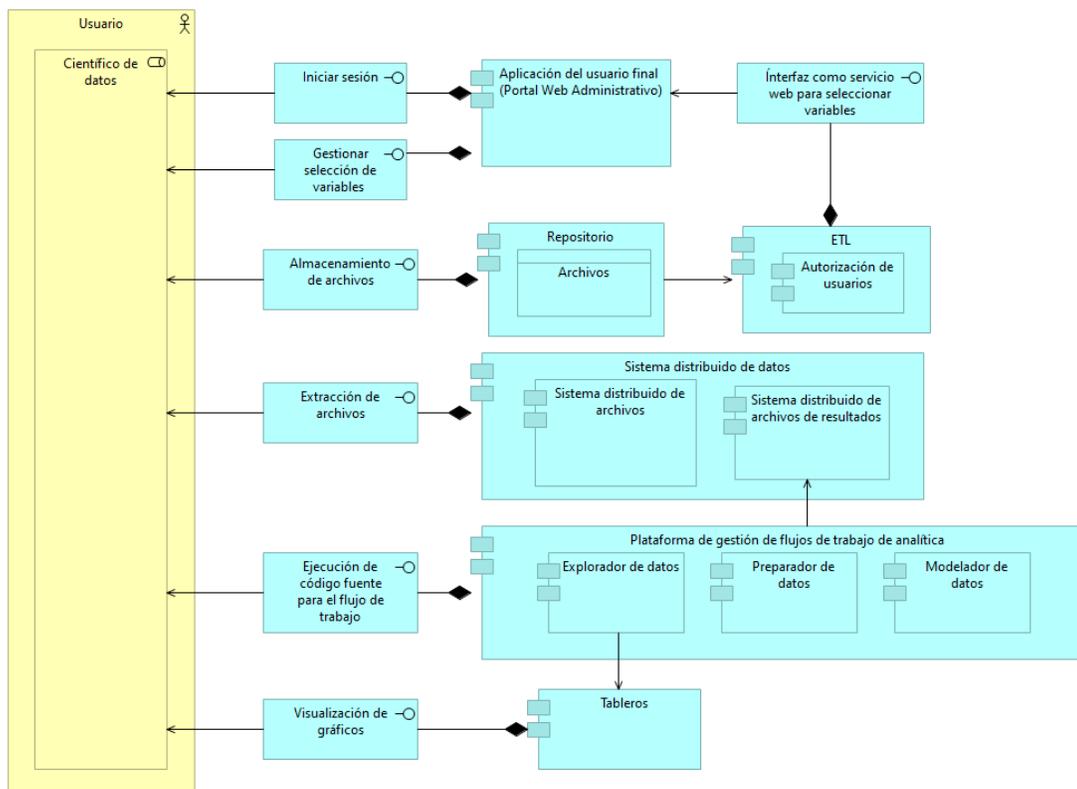
ETL y Autenticación de usuarios realizan una colaboración entre ellos para confirmar y validar el token de acceso que proviene desde la Aplicación de usuario final luego de que éste es autenticado. Autenticación accede al objeto de dato Cuenta que representa los datos de la cuenta de usuario. El componente ETL accede también a los datos de la entidad de salud y las diferentes vistas minables que son alojadas en el componente del sistema distribuido de archivos que ofrece una interfaz de consulta y descarga de datos hacia el componente que representa la plataforma para la gestión de los flujos de trabajo de analítica.

Éste último elemento de los flujos de trabajo se compone de otros tres componentes, el explorador, preparador y modelador de datos. Los tres acceden a los objetos de datos FlujoDeTrabajo para estructurar sus resultados, también al sistema de archivos distribuidos de resultados en donde depositan datos en archivos de resultados.

El explorador de datos se compone del generador de gráficos y se relaciona directamente con los tableros que le permite representar gráficamente esos resultados. El preparador de resultados contiene los componentes que le ayudan a realizar las operaciones de limpieza, normalización y de NLP en los datos. Y el modelador posee componentes para las tareas de aplicación de métodos de aprendizaje de máquina, aprendizaje profundo y clasificación.

3.2. Punto de Vista de Uso de Aplicaciones

Luego de observar la estructura de las aplicaciones, ahora es el turno de ver cómo se usan entre ellas y también con la capa de negocio en color amarillo representando el usuario y su respectivo rol. En este punto de vista se expresan, además, las diferentes interfaces y posibles servicios que a nivel de aplicación son consumidos por el usuario.



En el diagrama se puede observar que el actor Usuario toma el rol de científico de datos y utiliza los servicios ofrecidos por los componentes de aplicación a través de sus interfaces.

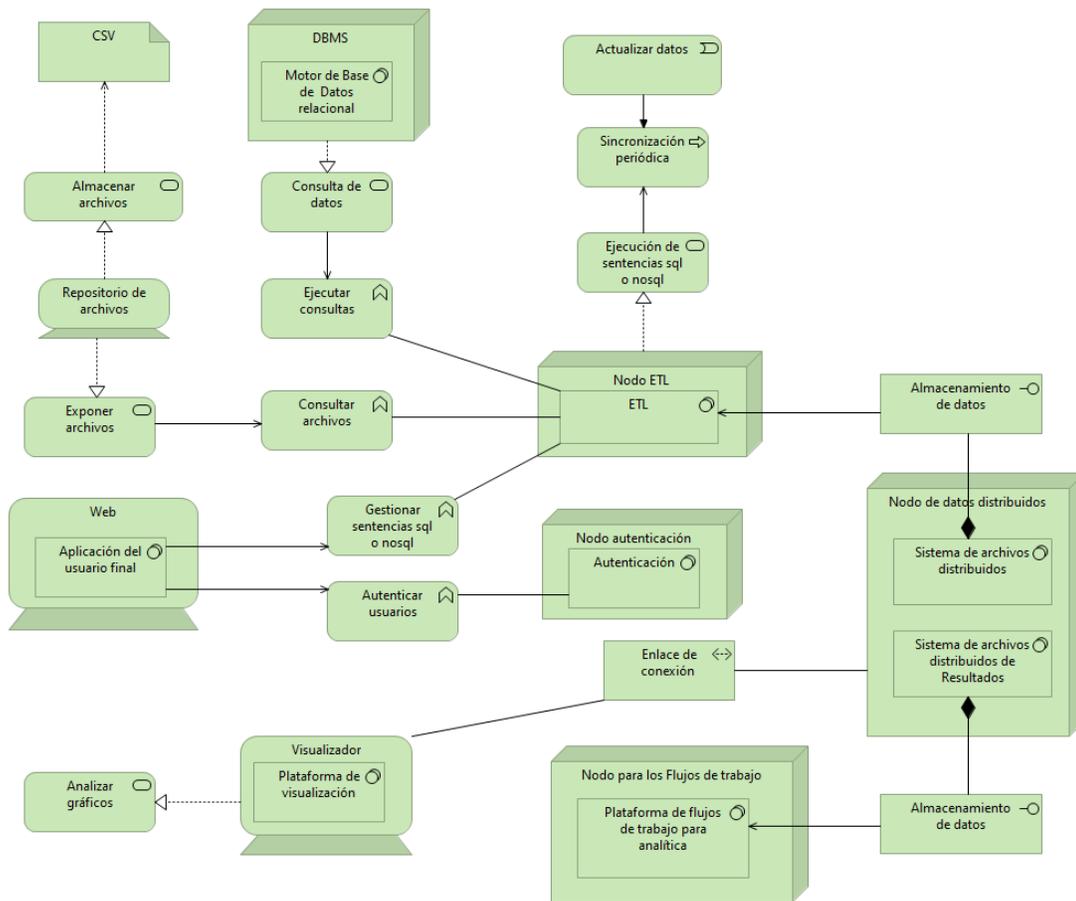
El científico de datos inicia sesión y gestiona la selección de variables con la Aplicación del usuario final. También consume los servicios de almacenamiento que el repositorio de archivos ofrece, y extrae los archivos almacenados en los sistemas distribuidos de datos. El componente de la Plataforma de gestión de flujos de trabajo de analítica proporciona una interfaz para que el científico de datos pueda ejecutar el código fuente que contenga funciones de exploración, preparación y modelamiento de datos. El científico puede utilizar los tableros para visualizar los gráficos necesarios que son operados por el explorador de datos.

4. Arquitectura Física

En la arquitectura física también se utilizará la ayuda de la especificación de *Archimate* para describir los elementos que son utilizados en una capa tecnológica con su estructura y comportamiento que pueden tener a nivel de infraestructura en la arquitectura de referencia propuesta. Para este escenario se presenta el punto de vista de tecnología de la arquitectura.

Punto de Vista de Tecnología

El punto de vista de tecnología es seleccionado para representar el frente físico de la arquitectura de referencia porque contiene elemento de software y hardware que son la base para la arquitectura de aplicaciones planteada anteriormente.



En el diagrama se puede observar que el repositorio de archivos es representado como un dispositivo tecnológico que realiza el servicio de almacenamiento de archivos como los artefactos CSV. También el servicio de exposición de archivos para que éstos puedan ser consultados por el sistema de software ETL contenido en un nodo con el mismo nombre.

El servicio de consulta de datos es realizado por el motor de base de datos dentro de un nodo DBMS, y sirve al ETL para que ejecute funciones de consultas sobre la base de datos. Además, se muestra que puede existir un disparador de eventos para actualizar datos, lo que permite que se ejecute un proceso tecnológico de sincronización periódica con el servicio de tecnología de ejecución de consultas SQL y NoSQL realizado por el sistema de software ETL que también proporciona funciones de tecnología para gestionar las sentencias SQL y NoSQL.

La función para gestionar las sentencias es accedida desde la aplicación de usuario final representado como un sistema de software dentro de un nodo en la web. El nodo de autenticación se compone de un sistema de software que ofrece la función de autenticar usuarios.

El nodo de datos distribuidos se compone de los dos sistemas de archivos distribuidos, el normal y el de resultados que a su vez ambos se componen de dos interfaces que permiten el almacenamiento de los datos, el normal para el ETL y el de resultados para el sistema de software como plataforma de flujos de trabajo de analítica.

El enlace de conexión crea un puente para el intercambio de datos entre el nodo de datos distribuidos y el dispositivo compuesto de una plataforma para visualizar los datos y éste realiza el servicio de tecnología para analizar gráficos.

III. INSTANCIAS ARQUITECTÓNICAS

Con la arquitectura de referencia como guía, se realizaron una serie de sesiones en conjunto con los científicos de datos, donde se obtuvo una lista de atributos con los cuales debería contar la arquitectura y a cada uno de estos atributos se les asignó un peso de valor deseado entre *Alto*, *Medio* y *Bajo*.

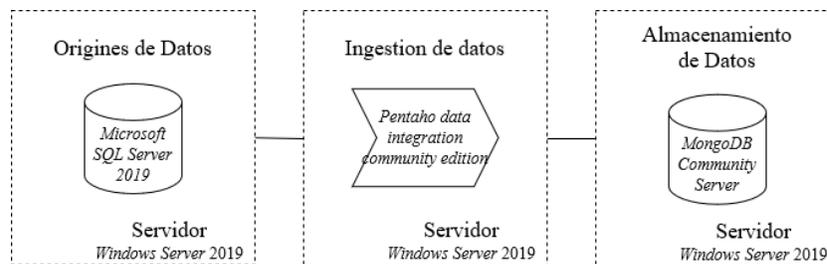
Atributo / Valor Deseado		
Sistema Operativo / <i>Bajo</i>	Hardware Mínimo / <i>Medio</i>	Costo Mensual / <i>Alto</i>
Conocimiento / <i>Alto</i>	Ultima fecha de actualización / <i>Bajo</i>	Licenciamiento / <i>Medio</i>

El siguiente procedimiento de selección de tecnologías proporciona la claridad de cuáles son las posibles instancias de la arquitectura de referencia a implementar.

A continuación, se describen los procesos iterativos que se siguieron para diseñar diferentes instancias de la arquitectura que solvente las necesidades de los científicos de datos. Se presentará el diseño arquitectónico implementado, las ventajas y las desventajas de este, así como las observaciones.

1. Primera Iteración

En esta primera iteración se realizó la instancia de arquitectura, la cual presento problemas al ejecutar los procesos de ETL, por la cual no se logró instanciar los demás dominios.



Ventajas

Esta instancia arquitectónica mostraba la ventaja de ser más flexible en el dominio de ingestión de datos ya que contaba con la implementación de la herramienta *Pentaho Data Integration Community Edition* y *MongoDB Community Server* de la cual cuales se posee un conocimiento alto, esto permitió que la instanciación de estas herramientas dentro de esta instancia fuera mucho más rápida y eficiente.

Desventajas

La desventaja que se descubrió fue que, por el volumen de datos la herramienta de *Pentaho Data Integration Community Edition* no podía generar el modelo de datos necesario para ser almacenado en *MongoDB Community Server*, esta limitante no se encuentra presente en la documentación del fabricante y se desconocía de la misma.

Valoración

Se muestra una tabla con los atributos antes seleccionados y las tecnologías implementadas y las planeadas.

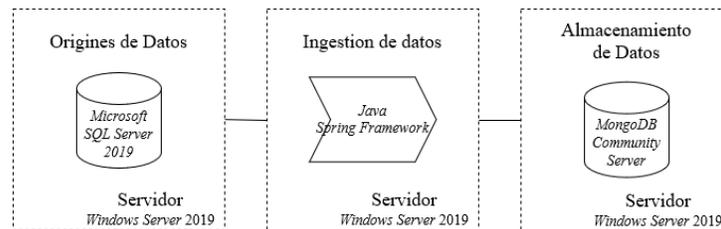
Atributo\Dominio	Fuentes de datos	Ingestión de datos	Almacenamiento de datos	Análisis de datos	Visualización de resultados
Herramienta	Microsoft SQL Server	Pentaho data integration community edition	MongoDB	Python	Python
Versión	SQL Server 2019	Pentaho 9.1	Mongo Community Server 4.4.4	TensorFlow 2 & Python 3.5–3.8	Pandas 1.2.2
Ultima fecha de actualización	Fecha > 2020	Fecha > 2020	Fecha > 2020	Fecha > 2020	Fecha entre 2020 y 2015
Sistema operativo	Windows	Linux & Windows	Linux & Windows	Linux & Windows	Linux & Windows
Hardware mínimo	Procesador: 16 vCPU RAM: 122 GB Espacio del disco: 1 TB	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores
Ambiente	On Premise	On Premise	On Premise	On Premise	On Premise
Costo (M)	0 USD	0 USD	0 USD	0 USD	0 USD
Licenciamiento	software propietario	software libre	software libre	software libre	software propietario
Curva de aprendizaje	Alto	Medio	Alto	Medio	Experto
Puntaje	0.79	0.83	0.90	0.83	0.90
				TOTAL	0.850

Observaciones

La imposibilidad de configurar la herramienta de *Pentaho Data Integration Community Edition* hizo que no se pudiera avanzar en los siguientes dominios, por lo cual esta instancia se descartó al no poder generar un flujo completo de los datos.

2. Segunda Iteración

En esta segunda iteración se realizó la instanciación de arquitectura que se presenta en la siguiente figura.



Ventajas

Esta iteración arquitectónica aporta la ventaja de tener el dominio de la ingestión de datos desarrollado a la medida de los requerimientos expresados por los científicos de datos, lo que permitió generar el modelo de datos deseado para su persistencia.

Desventajas

Esta iteración arquitectónica presenta dos grandes desventajas, la primera se presenta en el dominio de la ingestión de datos ya que se tiene se desarrolló a la medida con la tecnología de *Java & Spring framework*, esto representa para los científicos de datos la necesidad de modificar el código fuente siempre que se necesite un cambio, compilarlo y hacer un despliegue sobre el servidor de aplicaciones.

La segunda desventaja es la cantidad de datos que se persistía en *MongoDB Community Server*, el manejador de la base de datos no soportó la cantidad de documentos que se guardaban, lo cual hacía que se interrumpiera la carga de datos, este es un error conocido por la comunidad y el fabricante no tiene planes para corregirlo.

Valoración

Se muestra una tabla con los atributos seleccionados, así como su puntaje para cada uno de ellos.

Atributo\Dominio	Fuentes de datos	Distribución de datos	Almacenamiento de datos	Análisis de datos	Visualización de resultados
Herramienta	Microsoft SQL Server	Java Spring Framework	MongoDB	Python	Python
Versión	SQL Server	8	Mongo Communi-	TensorFlow	Pandas 1.2.2

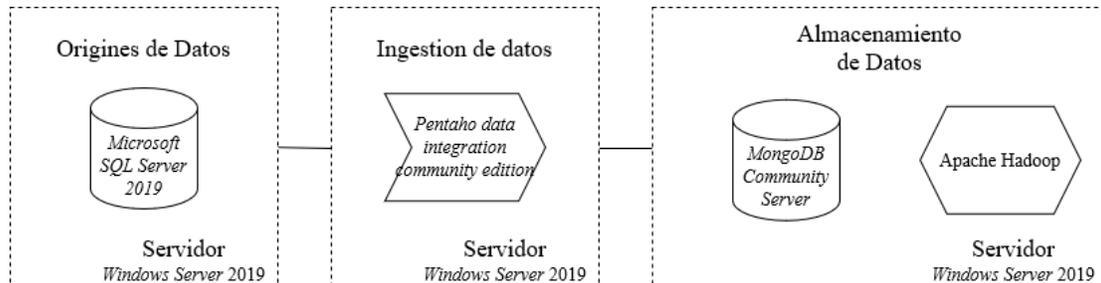
	2019		ty Server 4.4.4	2 & Python 3.5–3.8	
Ultima fecha de actualización	Fecha > 2020	Fecha > 2020	Fecha > 2020	Fecha > 2020	Fecha entre 2020 y 2015
Sistema operativo	Windows	Linux & Windows	Linux & Windows	Linux & Windows	Linux & Windows
Hardware mínimo	Procesador: 16 vCPU RAM: 122 GB Espacio del disco: 1 TB	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores
Ambiente	On Premise	On Premise	On Premise	On Premise	On Premise
Costo (M)	0 USD	0 USD	0 USD	0 USD	0 USD
Licenciamiento	software propietario	software libre	software libre	software libre	software propietario
Curva de aprendizaje	Alto	Alto	Medio	Medio	Experto
Puntaje	0.79	0.90	0.83	0.83	0.90
				TOTAL	0.87

Observaciones

La incapacidad de *MongoDB Community Server* para persistir documentos muy grandes como los necesitan los científicos de datos.

3. Tercera Iteración

En la tercera iteración debido a la facilidad que otorga *MongoDB Community Server*, en el dominio de almacenamiento de datos utilizamos una mezcla mixta que permita el almacenamiento de un gran volumen de datos y que también permita a los científicos de datos realizar varias vistas minables, para lo cual en esta tercera iteración se generó un esquema siguiente.



Ventajas

Esta iteración arquitectónica aporta la ventaja de tener dos componentes en el dominio del almacenamiento de datos, las dos tecnologías con *MongoDB Community Server* y *Apache Hadoop*, esto permite que se desacople la persistencia de los datos.

Desventajas

Al tener dos puntos de control sobre los datos, esto genera que se necesiten validaciones adicionales tanto en software como a los científicos de datos, de lado de software el flujo se hace más complejo. Los científicos de datos deben tener más conocimiento en especial sobre Apache Hadoop ya que al persistir cualquier tipo de información pueden llegar a saturar de datos la implementación de la arquitectura.

Valoración

Se muestra una tabla con los atributos seleccionados, así como su puntaje para cada uno de ellos.

Atributo\Dominio	Fuentes de datos	Distribución de datos	Almacenamiento de datos	Análisis de datos	Visualización de resultados
Herramienta	Microsoft SQL Server	Java Spring Framework	Apache Hadoop	Python	Python
Versión	SQL Server 2019	8	Apache Hadoop 3.0.0	TensorFlow 2 & Python 3.5–3.8	Pandas 1.2.2
Ultima fecha de actualización	Fecha > 2020	Fecha > 2020	Fecha > 2020	Fecha > 2020	Fecha entre 2020 y 2015
Sistema operativo	Windows	Linux & Windows	Linux & Windows	Linux & Windows	Linux & Windows
Hardware mínimo	Procesador: 16 vCPU RAM: 122 GB Espacio del disco: 1 TB	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores	Procesador: 2GHz dual Core RAM: 4 GB Espacio del disco: 25 GB Núcleos de procesador: 2 cores
Ambiente	On Premise	On Premise	On Premise	On Premise	On Premise
Costo (M)	0 USD	0 USD	0 USD	0 USD	0 USD
Licenciamiento	software propietario	software libre	software libre	software libre	software propietario
Curva de aprendizaje	Alto	Alto	Medio	Medio	Experto
Puntaje	0.79	0.90	0.83	0.83	0.90
				TOTAL	0.87

Observaciones

Esta iteración debido a que cumple con las necesidades de los científicos de datos, se considera la final y es la instanciación de la arquitectura de referencia propuesta, en la siguiente sección de la prueba de concepto se detallara cada domino y atributo de esta.

4. Cuarta Iteración

La cuarta iteración es la que corresponde a la iteración final y esta es presentada y detallada en la sección [III. Prueba de Concepto](#).

IV. PRUEBA DE CONCEPTO

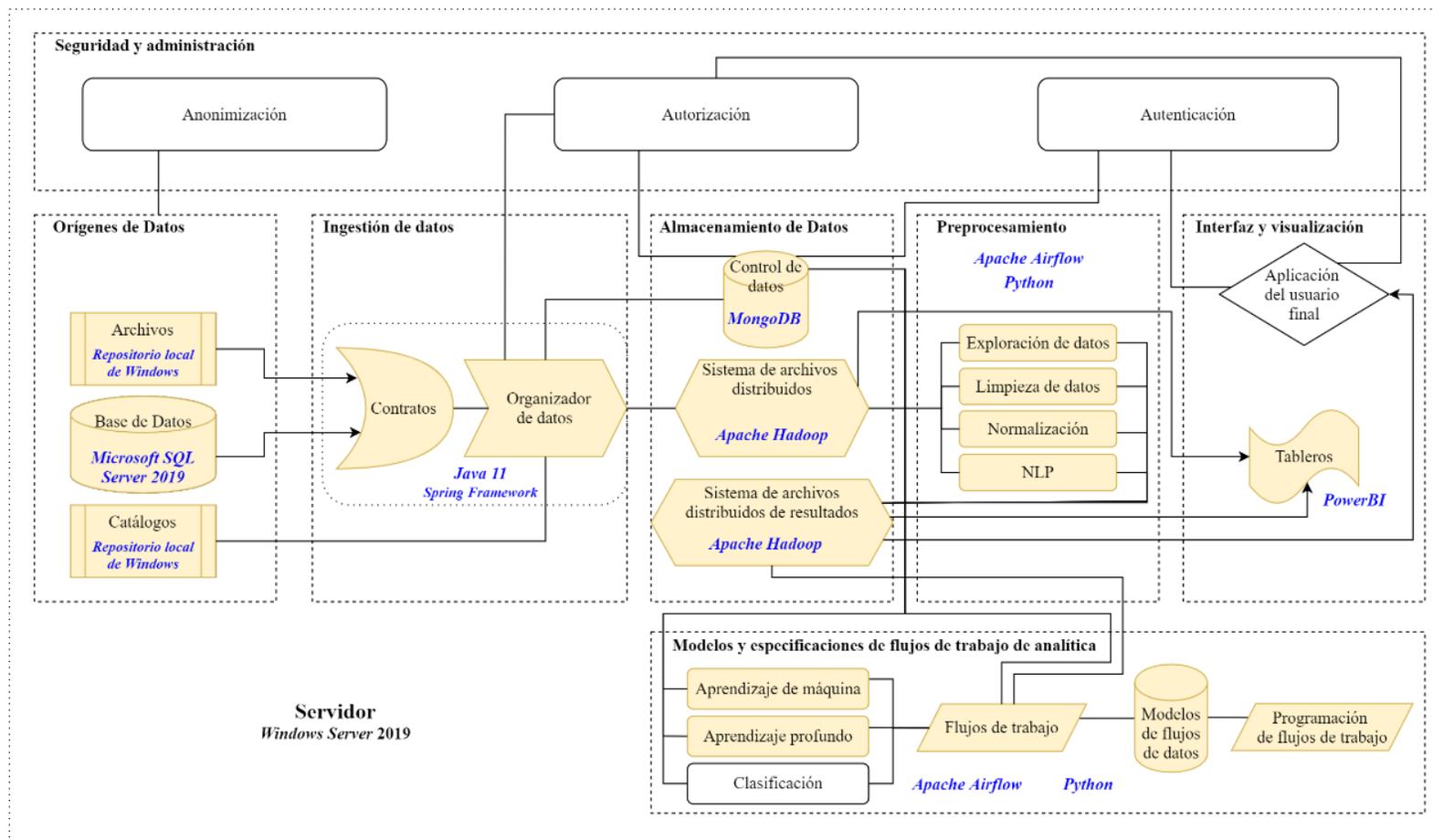
En esta sección se explica el detalle de las tecnologías que se utilizaron en cada dominio y por componentes de la arquitectura de referencia propuesta. A esta prueba de concepto la llamamos una instanciación de la arquitectura de referencia. Se presentan las descripciones de las configuraciones e implementación realizadas partiendo desde la configuración de los datos proporcionados por el HUSI de la PUJ, las operaciones de extracción, transformación y cargue de datos, almacenamiento, administración, configuraciones y desarrollos que soporten el preprocesamiento de lenguaje natural y la ejecución de algoritmos de aprendizaje profundo y de máquina sobre las historias clínicas electrónicas de los pacientes en el contexto de un proceso de diagnóstico de apnea del sueño. El detalle de instalaciones y configuración se encuentra en el Anexo 3.

1. Alcance General de la Prueba de Concepto

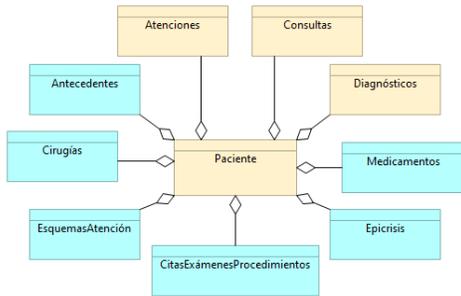
La implementación de la prueba de concepto presenta algunos limitantes los cuales también están reflejados en la sección [2.3 Restricciones arquitectónicas](#), pero ésta al ser una instanciación de la arquitectura de referencia, tiene unas limitantes adicionales que se deben considerar, esta instanciación se rige bajo los acuerdos licenciamiento por volumen del software y plataformas de Dirección de Tecnologías de Información – DTI.[59]

1.1. Arquitectura de Referencia Propuesta

En la siguiente gráfica, los componentes implementados son los que se encuentran en color amarillo y en color azul están los nombres de las herramientas tecnológicas utilizadas.

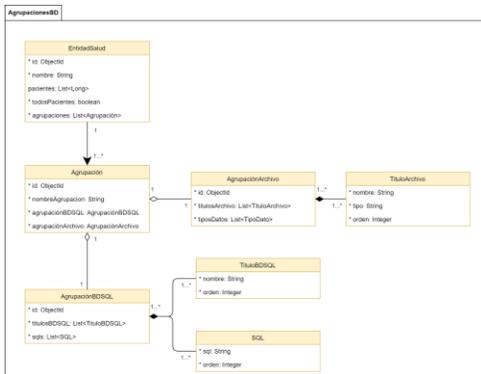


1.2. Modelo de datos del almacén



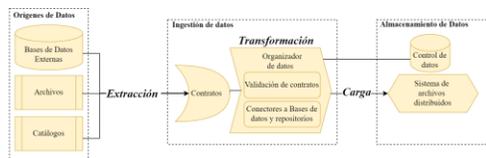
Se realizó la implementación de las cuatro entidades en color amarillo representándolas en un solo conjunto de datos en donde estaban algunos datos del paciente como el id anonimizado, peso, talla, fechas de consultas y atenciones, diagnósticos con el código interno del HUSI y el código CIE-9. Estos datos fueron almacenados en el sistema de archivos distribuidos en Hadoop y forman parte del proceso de ETL entre los dominios de Orígenes, Ingestión y Almacenamiento de datos.

1.3. Modelo de Datos de Administración y Organización



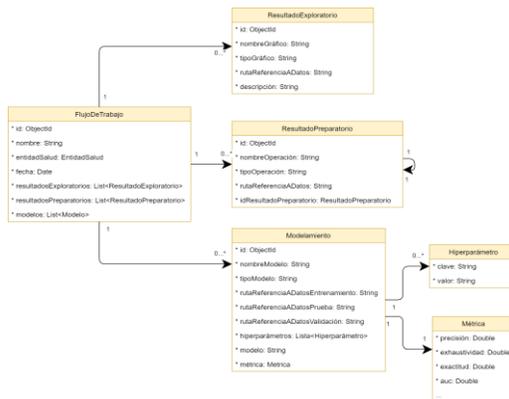
El modelo para la administración y organización de los datos se implementó en su totalidad en esta prueba de concepto. Representa la estructura de los datos para los componentes de Contratos y Organizador de Datos. Es almacenado en la base de control de datos en MongoDB y es fundamental para gestionar la selección de variables requeridas por el Científico de datos que son extraídas del dominio de Orígenes de datos. El *Json* de la estructura de los datos almacenados se puede ver en el Anexo 9.

1.4. Modelo de ETL



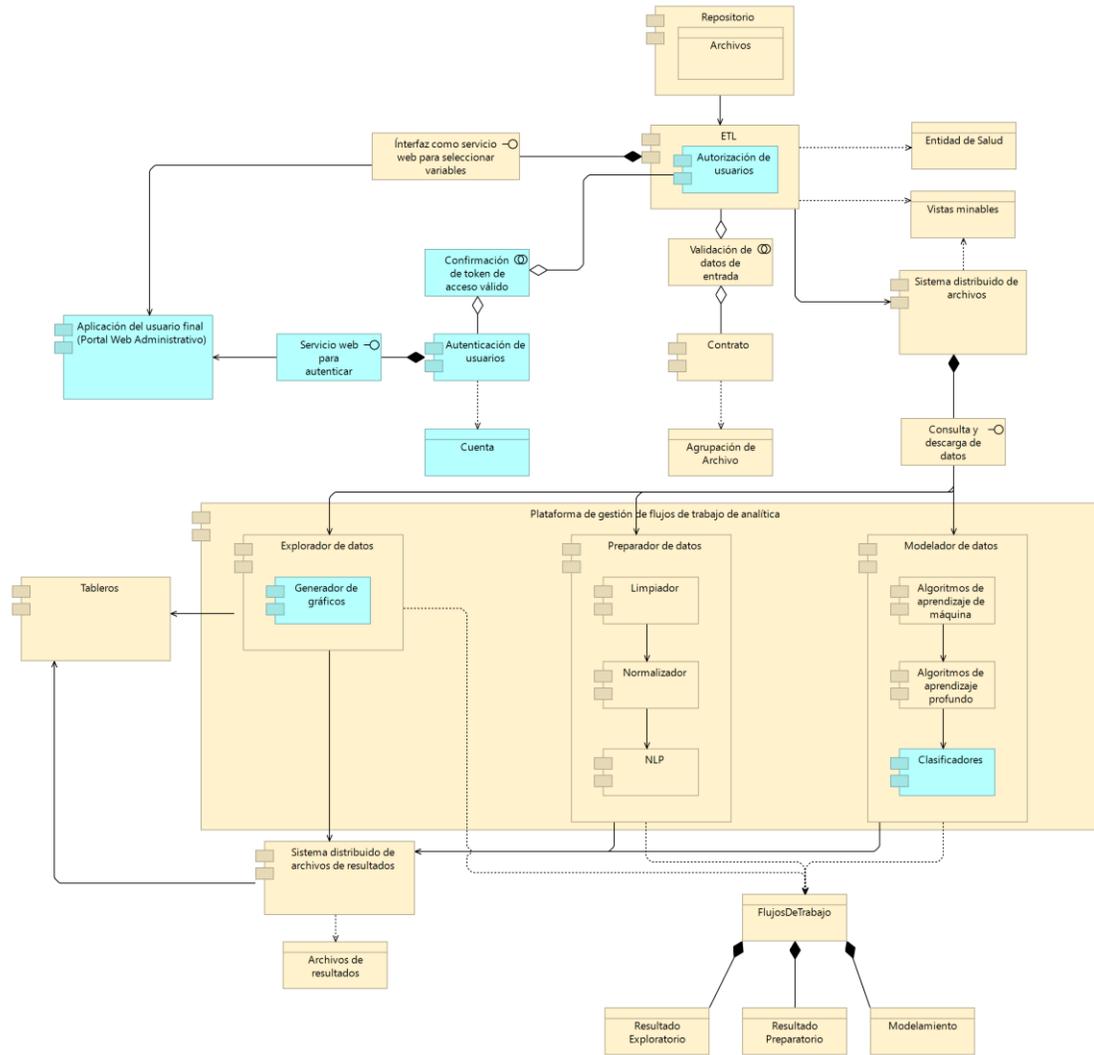
Este modelo fue implementado por completo en la prueba de concepto. Las funcionalidades de extracción, transformación y carga de datos se realizaron por medio de microservicios web desarrollados en lenguaje *Java* mediante *Spring Framework*.

1.5. Modelo de Datos de Resultados



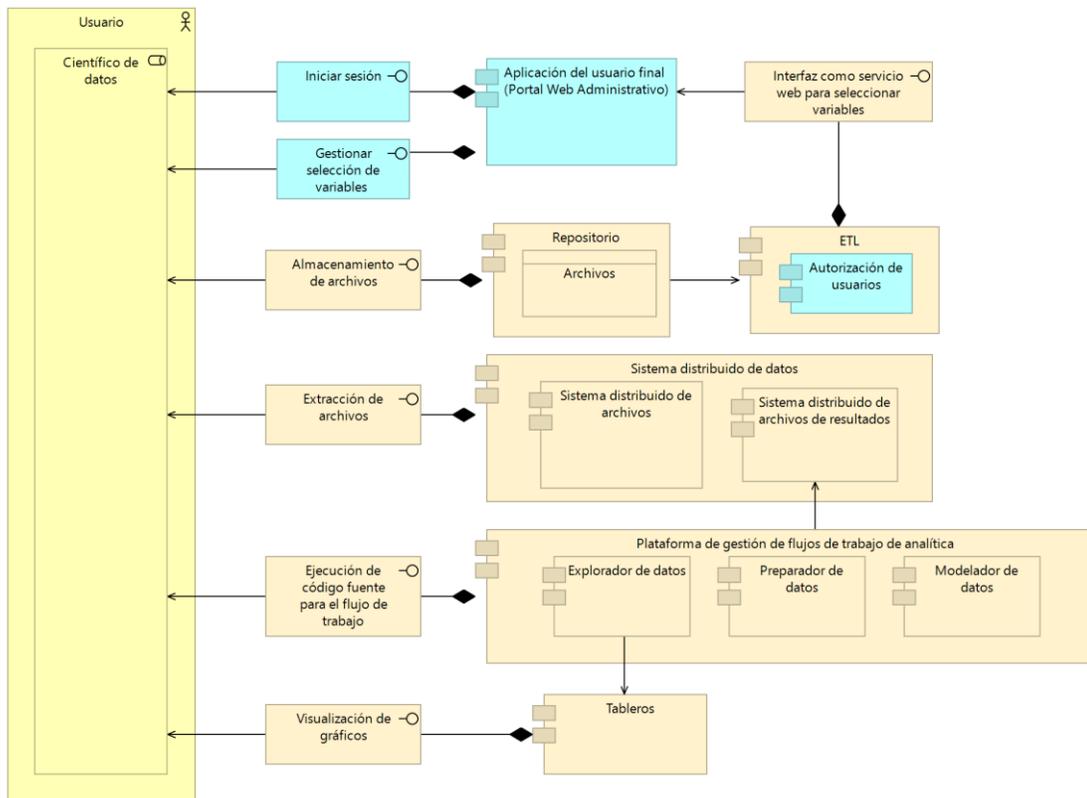
Se realizó la implementación del modelo de datos de resultados por completo. El Organizador de datos ofrece un microservicio web que permite la inserción de estos resultados en MongoDB, Base de Control de datos, respetando la estructura del presente modelo. Este microservicio es consumido desde el código que representa el flujo de trabajo (Anexo 8). El *Json* que se inserta se puede ver en el Anexo 10.

1.6. Punto de Vista de Estructura de Aplicaciones



El color en amarillo representa los elementos que fueron implementados en la prueba de concepto. El repositorio como un sistema de carpetas en Windows para el dominio de Orígenes de datos. El componente de ETL para gestionar la Ingestión de los datos junto con Entidad de Salud y Vistas minables los cuales representan los modelos de administración y del almacén respectivamente. También con la interfaz del servicio para selección de variables y los Contratos con Agrupación de Archivo que es la representación del modelo de administración y organización. El componente de Plataforma de gestión de flujos de trabajo fue implementado con Airflow y cada uno de los componentes de Explorador, Preparador y Modelador de datos se implementaron en bloques de código definidos dentro de Airflow y se adicionó el acceso a los datos de Archivos de resultados y Flujos de trabajo los cuales son representados con el Modelo de datos de Resultados. El componente de Tableros fue instanciado en PowerBI.

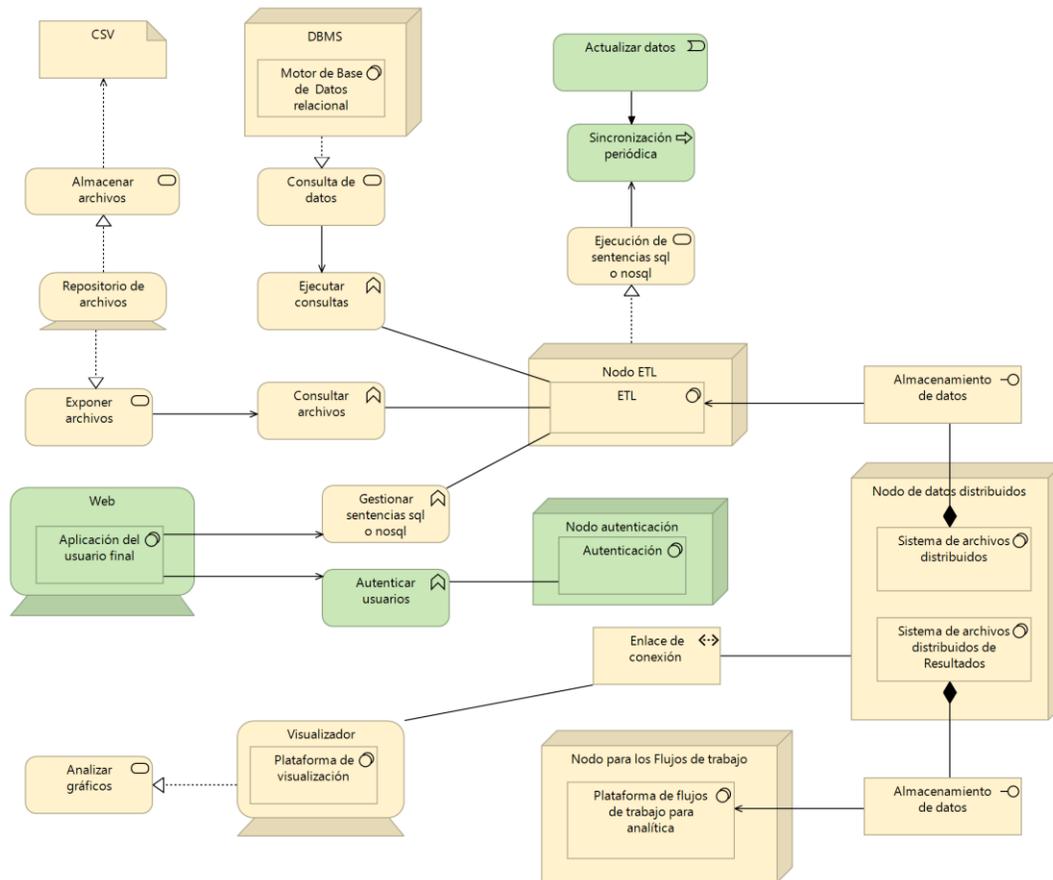
1.7. Punto de Vista de Uso de Aplicaciones



Los elementos en amarillo naranjado fueron los implementados en la prueba de concepto para este punto de vista. Los componentes tienen la misma instanciación de la descripción anterior en el punto de vista de estructura de aplicaciones. Lo que se debe resaltar aquí es que el rol del Científico de datos si existió y utilizó la prueba de concepto por medio de interfaces para selección de variables, almacenamiento de archivos, procesos de ETL, ejecución de los flujos de trabajo en Airflow, generación y visualización de gráficos en PowerBI.

Es necesario resaltar en general que el Modelo de Datos de Seguridad y los demás elementos excluidos en las descripciones anteriores, no se implementaron debido a que no se encontraban entre los objetivos y alcance del presente trabajo; además, están destinados a evaluarse en trabajos futuros para implementar los componentes del dominio de Seguridad que son Anonimización, Autorización y Autenticación; junto con el componente de Aplicación del usuario final y Clasificación incluido en el dominio de Modelos y Flujos de Trabajo de Analítica.

1.8. Punto de Vista de Tecnología



Los nodos de tecnología que están como una caja rectangular como por ejemplo el nodo DBMS, son una representación simbólica de servidores en donde se ejecutan los sistemas de software contenidos en sus nodos correspondientes, en este caso el Motor de Base de Datos Relacional. Son simbólicos porque todo fue implementado dentro de dos máquinas virtuales proporcionadas por la Pontificia Universidad Javeriana. Entonces, el Motor de BD permitía realizar consultas con ejecución de sentencias SQL, El sistema de software ETL fue desplegado en un Nodo ETL el cuál fue un Tomcat predeterminado en *Spring Boot* permitiendo la selección de variables y gestión de consultas SQL. Para el Repositorio de archivos lo ideal es que sea representado como un sistema SFTP (*SSH File Transfer Protocol*), pero debido a los recursos limitados se implementó un sistema de carpetas y archivos en el disco duro del servidor. El sistema de software para gestionar los archivos distribuidos fue implementado con Hadoop. La Plataforma de flujos de trabajo para analítica fue Airflow y la Plataforma de Visualización fue PowerBI.

Los elementos que quedaron en color verde no se implementaron y se resalta la propuesta de crear eventos de actualización de datos que sean disparados para funciones de tecnología de sincronización periódica para la ejecución de sentencias SQL con el propósito de tener los datos siempre actualizados.

2. Dominios de la Prueba de Concepto

Cada componente será analizado por separado desde la vista de datos, la vista de componentes físicos, la vista de riesgos y beneficios arquitectónicos y la configuración realizada, para después unir cada componente que genera la prueba de concepto. A continuación, se enlistan los componentes de la arquitectura de referencia.

- Orígenes de datos
- Ingestión de datos
- Almacenamiento de datos
- Preprocesamiento
- Modelos y especificaciones de flujos trabajo de analítica
- Interfaz y visualización

2.1. Orígenes de Datos

Este dominio corresponde a los orígenes de datos que alimentaran la prueba de concepto y está dividido en tres componentes que son los archivos estructurados de texto (CSV), bases de datos y catálogos de datos.

Esta división hace que este componente sea lo suficientemente abierto para que otras fuentes de datos se puedan integrar fácilmente a la arquitectura y que no se necesite ningún cambio, pero lo suficientemente cerrado para que solo bases de datos con las características deseadas puedan conectarse a la arquitectura.

Riesgos y Beneficios Arquitectónicos

A continuación, se enumeran los riesgos y beneficios que se encontraron al instanciar este componente.

Número	Descripción	Beneficio	Riesgo
1	Persistencia local de la base de datos.	<p>Exploración completa de los datos.</p> <p>Ingesta más rápida de los datos en los componentes conectados.</p> <p>Menor latencia de comunicación entre los componentes.</p>	<p>Exceso de datos que genere un colapso en el servidor.</p> <p>Permite carga de datos innecesarios.</p>
2	Instalación de motor de base de datos <i>Microsoft SQL Server 2019</i>	Permite la carga de la base de datos desde un archivo de respaldo.	Imposibilita cargar otras bases de datos que no sean soportadas por este manejador.

Configuración

En este caso específico de instanciación, se parte de la base de datos proporcionada por el HUSI de la PUJ la cual fue descriptada. Se realizó la extracción de tres archivos: SAHI_PUJ.mdf, SAHI_PUJ_1.ldf y SAHI_PUJ_Indices.mdf (Archivo de base de datos maestro) de aproximadamente 890 GB, para configurar y restaurar la base de datos en Microsoft SQL Server 2019 (RTM) - 15.0.2000.5 (X64) Enterprise Evaluation Edition sobre Windows Server 2019 Standard 10.0. Adicionalmente, para la gestión de la base de datos, fue necesario instalar SQL Server Management Studio v18.7.1. Ver detalle en el Anexo 3, sección 1.1.

Para los componentes de Archivos y Catálogos se simula un repositorio utilizando un directorio en el disco duro del servidor Windows Server 2019 en donde se crea una estructura de carpetas C:\Repositorio\Catalogos, C:\Repositorio\ArchivosCargue y otra adicional para el cargue de los contratos C:\Repositorio\Contratos. Ver detalle en el Anexo 3, sección 1.4.

2.2. Ingestión de Datos

En este dominio corresponde a la selección de los datos que se van a cargar, los componentes de Contratos y Organizador de Datos, sin componentes de administración que permiten controlar y verificar los datos que se van a cargar en los siguientes dominios, así como también controla donde se va a persistir estos datos.

Riesgos y Beneficios Arquitectónicos

Número	Descripción	Beneficio	Riesgo
1	Módulo de contratos es la única forma de ingestar datos.	Se tienen control sobre la ingesta de datos.	Se puede volver un cuello de botella.
2	El módulo Organizador de Datos controla los flujos iniciales de carga de datos.	Se tiene la posibilidad de configurar donde se van a persistir los datos.	Se genera un punto único de falla, haciendo este uno de los módulos más sensibles en la arquitectura.

Configuración

Existen diferentes maneras de implementar la ingestión de los datos y las diferentes operaciones de ETL. Inicialmente se realizó una configuración e implementación en *Pentaho Data Integration* utilizando diferentes conjuntos de ETL en distintos Jobs lo cual es una solución factible, pero debido al conocimiento limitado de Pentaho como herramienta de ETL y también a la necesidad puntual de seleccionar las variables con las que el científico de datos trabajaría, se opta por la construcción de un componente ETL como el Organizador de datos en

lenguaje Java 11 con Spring Framework y dentro de la lógica implementada se integra el componente de Contratos. Los dos permiten leer una serie de reglas y configuraciones realizadas en la base de datos de control que permite generar las condiciones del contrato por entidad de salud. También permite realizar las operaciones de extracción, transformación y cargue de los datos al sistema de archivos distribuido. Además, facilita el cargue de los archivos que contienen catálogos de datos. El detalle se encuentra especificado en el Anexo 3, secciones 1.5. – 2.2. – 3.1.

2.3. Almacenamiento de Datos

En este componente se almacenarán tanto los datos que puedan llegar a utilizar los científicos de datos, los resultados los diferentes procesos analíticos y también la persistencia del control de datos.

Riesgos y beneficios arquitectónicos

A continuación, se enumeran los riesgos y beneficios que se encontraron al instanciar este componente.

Número	Descripción	Beneficio	Riesgo
1	Almacenamiento masivo para cualquier tipo de datos.	Permite que los científicos de datos puedan generar cualquier tipo de reporte.	Libertad de cargar cualquier tipo de dato que pueda generar datos que no son necesarios.

Configuración

Se realizó la instanciación del elemento de control de datos en MongoDB por su facilidad con la gestión de objetos como documentos dentro de colecciones y diferentes bases de datos y su rapidez. La versión instalada fue MongoDB 4.4.4 Community en donde se configuraron dos bases de datos regidos por los modelos de referencia [Modelo de datos de administración y organización](#) y [Modelo de datos de resultados](#) para la administración de agrupaciones y los datos de resultados de la aplicación del preprocesamiento y los flujos de trabajo de analítica. Ver Anexo 3, secciones 1.2. – 2.4. – 3.3.6.

En cuanto a los sistemas de archivos distribuidos se optó por Hadoop versión 3.2.2 que luego de realizar su instalación y configuración en el servidor y al implementar la conexión desde el componente ETL, fue en donde se alojaron las diferentes vistas minables según el modelo de datos estrella presentado en la sección del [2.1. Modelo de datos del almacén](#). El detalle de la instanciación de los datos en Hadoop se puede ver en el Anexo 3, sección 3.3.

El sistema de archivos distribuido de resultados se podría utilizar en el mismo nodo de Hadoop instalado o en uno distinto según sea el caso definido por la persona que realice la instanciación de la presente arquitectura de referencia. En este caso se realizó la configuración

en el mismo Hadoop diferenciando muy bien las rutas tanto para los archivos de vistas minables como para los de resultados. Ver sección 3.2. del Anexo 3.

2.4. Preprocesamiento

En este dominio se realizó la instanciación de los cuatro componentes de análisis exploratorio, limpieza de datos, NLP con la operación de modelamiento de tópicos, y normalización con pca y denoising auto encoder guiados por el científico de datos.

Riesgos y Beneficios Arquitectónicos

A continuación, se enumeran los riesgos y beneficios que se encontraron al instanciar este componente.

Número	Descripción	Beneficio	Riesgo
1	Algoritmos predefinidos.	No se requiere de mayor esfuerzo para aplicar algoritmos de analítica de datos comunes.	Los algoritmos por si generalidad no sean útiles a los científicos de datos.

Configuración

Se realizó la instalación en el servidor de la herramienta llamada Airflow la cual es una plataforma de ejecución de flujos de trabajos de analítica de datos de código abierto en lenguaje Python. Los bloques de código fueron desarrollados por el científico de datos en lenguaje Python y éstos fueron integrados a Airflow la cual permite la gestión y programación de los flujos de trabajo llamados DAG (*Directed Acyclic Graph*). El detalle se puede observar en el Anexo 3, secciones 1.6. en donde se muestran los pasos para la instalación, sección 2.5. en donde está la inicialización y despliegue de Airflow y en la sección 3.3. en donde se muestra la configuración y ejecución de los DAG y su respectivo código fuente.

2.5. Modelos y Especificaciones de Flujos Trabajo de Analítica

Es en este componente donde se definen los diferentes flujos de analítica de datos por parte de los científicos de datos, así como también la programación de estos flujos y su ejecución.

Riesgos y Beneficios Arquitectónicos

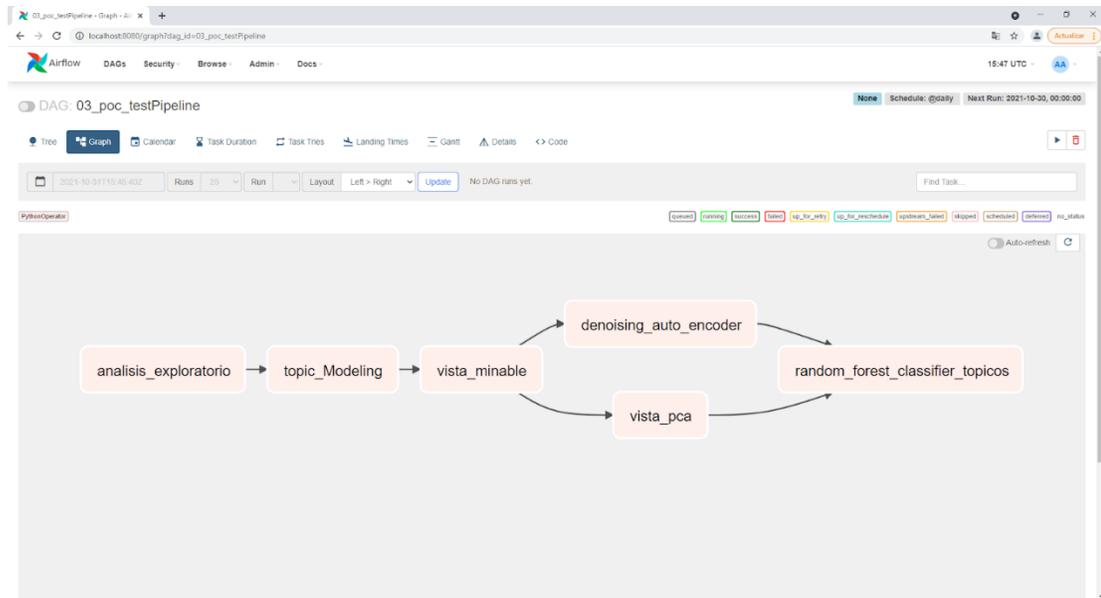
A continuación, se enumeran los riesgos y beneficios que se encontraron al instanciar este componente.

Número	Descripción	Beneficio	Riesgo
--------	-------------	-----------	--------

1	Apertura a utilizar librerías de Python.	Permite que los científicos de datos puedan utilizar diferentes librerías.	Carga de librerías que no cumplan o generen riesgo en la arquitectura.
---	--	--	--

Configuración

Los bloques de código también fueron proporcionados por el científico de datos en donde después de aplicar los métodos de denoising auto encoder y la vista pca, se realiza la aplicación del algoritmo de aprendizaje de máquina llamado árboles de decisión. La ejecución se realizó en Airflow en el mismo DAG y el resultado del grafo del flujo de trabajo es el que se muestra a continuación:



Cada bloque de ejecución almacena los resultados en la carpeta de Resultados de Hadoop por entidad de salud y los parámetros en la base de datos de Mongo según el modelo de datos de resultados. El detalle de la implementación se puede visualizar en el Anexo 3, sección 3.3.

2.6. Interfaz y Visualización

En este dominio se realizó la instanciación de la visualización de los resultados generados por los científicos de datos, en este dominio se presenta el componente de tableros los cuales se pueden configurar desde la herramienta de PowerBI o desde las librerías de Matplotlib que cuenta el lenguaje de programación de Python.

Riesgos y Beneficios Arquitectónicos

Número	Descripción	Beneficio	Riesgo
1	Visualización de tableros con le herramienta de PowerBI	Algunas configuraciones e integraciones son nativas, como la conexión a <i>Hadoop</i> .	Se corre el riesgo de que este domino solo se pueda utilizar bajo el sistema operativo <i>Windows</i> .

Configuración

La instanciación del elemento de tableros para este caso fue realizada por la aplicación de Microsoft llamada PowerBI 2.92.706.0 64-bit el cual permite realizar directamente la conexión con Hadoop, descargar sus respectivas vistas minables o resultados en archivos CSV y utilizar las gráficas predeterminadas por la aplicación. Incluso se puede realizar allí la escritura y ejecución de código en lenguaje Python que facilita el uso de la librería Matplotlib para la visualización de gráficos. En el Anexo 3, sección 3.4. se puede observar la interacción de PowerBI con Hadoop y la generación de gráficos.

3. Uso de la AR Propuesta en Otros Problemas de Diagnóstico

En esta sección se realizará la descripción del paso a paso para utilizar la AR en otros problemas de diagnóstico. El objetivo de este trabajo fue lograr la inclusión de ejecutar algoritmos de aprendizaje profundo y de máquina con tareas previas de procesamiento y todo lo que contempla un flujo de trabajo de analítica de datos en el contexto del diagnóstico de la apnea del sueño, pero la AR propuesta es posible utilizarla para generar y administrar más flujos de trabajo para otro tipo de diagnósticos utilizando como datos de entrada diferentes tipos de datos, pues en este caso se utilizaron datos de texto, pero se pueden incluir imágenes.

Se parte desde un escenario en que se desea implementar la AR propuesta en este trabajo para la ejecución de cualquier tipo de algoritmo de aprendizaje profundo o de máquina con previo procesamiento de datos. Los datos provienen de una base de datos no relacional en donde se encuentran diferentes tipos de datos necesarios para ingresar al sistema entre los cuales se tienen texto e imágenes. Todo es con el fin de generar modelos de analítica para la predicción del diagnóstico de cáncer de pulmón.

Para la instanciación de un sistema con la arquitectura de referencia se necesita los roles entre arquitecto y desarrollador que puedan implementar dominios propuestos en la AR.

El uso de la arquitectura lo realiza un científico de datos, su perfil es definido en la sección siguiente de la validación TAM. Adicionalmente, el campo de implementación de la AR propuesta no es solo en la salud, pues se puede instanciar en otros campos empresariales con la salvedad que se deben generar otros modelos y puntos de vista de arquitectura.

1. Se debe realizar una revisión inicial de la totalidad de la AR propuesta y la prueba de concepto de este trabajo para que el arquitecto y su equipo extraigan lo que les será útil para su implementación y lo que no. En los modelos de datos es posible que se requieran incluir más entidades y atributos para gestionar las imágenes como rutas de la ubicación de cada imagen o la codificación de éstas. También se tendrán ideas iniciales de los componentes básicos que estructurarán el sistema, así como su interacción entre ellos mismos y con el usuario.
2. Seguidamente los arquitectos deben realizar los diseños de arquitectura más concretos, la adquisición de tecnologías y planes de integración. También se debe definir si se realizará construcción de componentes a la medida y plantear las estimaciones de tiempos de entrega y manuales de usuario. Es importante tener presente que las imágenes deben ser cargadas en un componente de sistema de archivos distribuido y que el organizador de datos debe permitir las operaciones de ETL de estos tipos de dato. La selección de variables con respecto al tipo de dato texto también se debe contemplar en este momento.
3. Para la interacción del usuario desde el dominio de Interfaz y Visualización con el de Seguridad se deben diseñar las interfaces gráficas de usuario que podrían construirse y los protocolos de seguridad que involucren los tres componentes de Autenticación, Autorización y Anonimización. De la misma manera se debe abordar el componente de tableros y sus configuraciones para el acceso a los datos cargados en el sistema.
4. Existe la opción de integrar los dominios de preprocesamiento, modelos y flujos de trabajo de analítica o de trabajarlos por separado. Evaluar componentes ya construidos que permiten realizar las tareas de exploración, limpieza, normalización y procesamiento de lenguaje natural o si por el contrario el usuario final como científico de datos las construirá de acuerdo con lo que necesite para el caso de los tipos de datos de texto. Cuando se trata de imágenes, pueden existir tareas de etiquetado y normalización.
5. La implementación del dominio de modelos y flujos de trabajo puede ser un componente de software que permita gestionar los *pipelines* de analítica de datos consumiendo servicios del dominio de preprocesamiento e interactuando con el almacenamiento de datos. En esta implementación se debe permitir la ejecución de algoritmos de aprendizaje profundo y de máquina.
6. El científico de datos debe tener interfaces gráficas de usuario que le permitan seleccionar las variables que entrarán a la arquitectura, las imágenes y los datos para organizarlos en un almacenamiento que siempre se encuentre disponible para que él los pueda obtener desde la implementación del flujo de trabajo codificado para construir los modelos que le permitan determinar el diagnóstico de cáncer de pulmón. Para el caso de las imágenes también el científico de datos las obtendrá para aplicarle los algoritmos de aprendizaje profundo necesarios para determinar el modelo para el diagnóstico. Luego que el científico de datos termina el código del flujo de trabajo, se le debe permitir ingresarlo a la arquitectura por medio de alguna interfaz gráfica para que pueda ser ejecutado. Adicionalmente, desde la interfaz gráfica de usuario se le debe permitir al usuario consultar y visualizar los datos que ha almacenado como resultado de las iteraciones realizadas en el flujo de trabajo.

V. VALIDACIÓN DE LA ARQUITECTURA

1. ATAM

En las tareas de validación de la arquitectura de referencia es importante que se realice el cumplimiento de los objetivos propuestos al inicio del trabajo realizado, los atributos de calidad planteados y las decisiones arquitectónicas tomadas en el diseño y en su respectiva prueba de concepto si hay lugar a ella como en el presente trabajo. En esta ocasión, se validará la arquitectura con la técnica propuesta por la Universidad Carnegie Mellon y el Instituto de Ingeniería de Software denominada ATAM (Architecture Tradeoff Analysis Method), método de análisis del equilibrio de la arquitectura relacionada con los atributos de calidad planteados y dispuestos a ser aceptados en la respectiva evaluación.

Inicialmente se realizó el entendimiento del proyecto, se especificaron algunos atributos de calidad vistos en la sección y se construyó un modelo de arquitectura de referencia de BigData en el contexto de aplicar el análisis de datos a las historias clínicas electrónicas del HUSI para ejecutar algoritmos de aprendizaje profundo y de máquina en el diagnóstico de la apnea del sueño. Estas reuniones se dieron de manera iterativa y en cada iteración se realizaba la validación paso a paso según el proceso de ATAM.

- **Objetivos:** Los objetivos para la validación se encuentran en la sección de [Descripción del proyecto](#).
- **Arquitectura:** Se presenta la arquitectura propuesta en la sección [Arquitectura de referencia propuesta](#).
- **Enfoques Arquitectónicos:** Se realiza la presentación en la sección de [Directrices arquitectónicas](#).
- **Árbol de utilidades de atributos de calidad:** Es presentado en la sección de [Atributos de calidad](#)
- **Lluvia de ideas y priorización de escenarios:** Se realizó una reunión en donde se obtuvieron las ideas que se encuentran en el Anexo 4.
- **Resultados:** Basados en la información recolectada en el proceso de validación ATAM en cuanto a las diferentes propuestas, escenarios, atributos de calidad y priorización se puede determinar que la arquitectura de referencia propuesta cumple con los requerimientos de los científicos de datos a que satisface en un nivel alto la seguridad, eficiencia, funcionalidad, facilidad de mantenimiento y portabilidad, pero en donde se ve que hay más oportunidad de desarrollo es en la adaptabilidad.

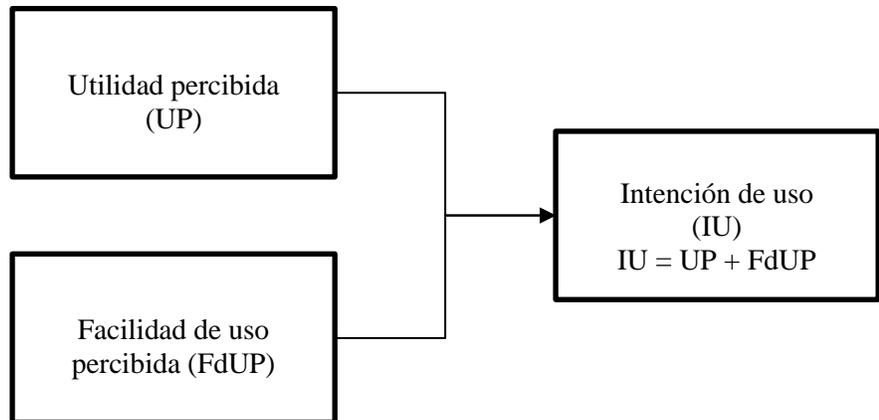
2. TAM

El modelo TAM (*Technology Acceptance Model*), es una teoría que intenta modelar como un usuario va a aceptar la tecnología, este modelo fue inicialmente descrito por Fred Davis en el año de 1989[60].

El modelo indica que existen dos variables previas:

- Utilidad percibida (*UP*)
Describe que el grado en el cual una persona cree que va a utilizar una tecnología afecta como se utilizara esta.
- Facilidad de uso percibida (*FdUP*)
Describe que el grado en el cual una persona cree que va a utilizar una tecnología sin esfuerzo.

Estas dos variables nos dan una pista, sobre cuál es la intención del usuario final; conocer la intención del usuario final es de suprema importancia ya que indica cuales son los puntos de mejora, así como también las fortalezas.



Para conocer estas dos variables, se genera un cuestionario de once preguntas, las primeras seis corresponden a la utilidad percibida y las últimas cinco corresponden a la facilidad de uso percibida, la suma de estas dos variables nos da la intención de uso.

Las dos variables se valoran de uno al siete, donde la calificación de uno indica que esta extremadamente en desacuerdo y la valoración siete indica que se está extremadamente de acuerdo.

Aplicación

Se realiza el cuestionario de once preguntas a un científico de datos, que utilizó la arquitectura para hacer una predicción del diagnóstico de la apnea del sueño, que va a ser el usuario final de la implementación de la arquitectura de referencia.

Perfil del Usuario Final

El usuario final debe tener el cargo de Científico de datos. En algunas industrias es llamado Analista o Ingeniero de datos. Su profesión base puede estar enfocada como estadístico, ingeniero industrial, matemático, ingeniero de sistemas, entre otras profesiones que incluya el análisis de datos y es indispensable que tenga nociones en la aplicación de algoritmos para aprendizaje profundo y de máquina.

A continuación, se indican las preguntas y los resultados.

Resultado y siguientes pasos

El resultado es que la implementación de la prueba de concepto para el único científico de datos que la evalúa nos indica que la Utilidad percibida (UP) es de 4.5 / 7, esto indica que percibe una ligera utilidad en la implementación realizada de la arquitectura de referencia.

Para la variable de Facilidad de uso percibida (FdUP), el científico de datos nos indica un valor de 2.6/7, lo que indica que es bastante improbable que la implementación de la arquitectura de referencia sea fácil de utilizar.

Lo que indica que la intención de uso en genera nos daría el resultado de 7.1/14, lo que nos indica que el científico de datos aún no está seguro si tendrá intención de utilizar la implementación de la arquitectura de referencia, también refleja que el científico de datos les ve utilidad, pero hay que mejorar en la facilidad de uso.

Como siguientes pasos, se considera mejorar la variable de Facilidad de uso percibida (FdUP) debido a que fue la más baja, esto se puede realizar generando una de las siguientes propuestas:

- Generar vistas de usuario final, para los dominios que aún se deben modificar y administrar en código.
- Automatizar algunos flujos de datos por medio de algunos monitores que detecten cambios en los componentes y se configuren automáticamente, esto actualmente se debe hacer manualmente por el científico de datos.
- Desacoplar los dominios en servidores diferentes para que la implementación de la arquitectura sea más robusta a fallos, ya que actualmente toda la implementación se encuentra en un servidor generando un único punto de fallo.

3. Validación de la arquitectura con la gerencia del HUSI

Se realizó una sesión con la gerencia del Hospital Universitario San Ignacio en donde se presentó la arquitectura y sus resultados. La gerencia expone los siguientes comentarios:

- Incluir la utilidad de la arquitectura en función del área de la salud
- Definir la forma estándar de la aplicación de la arquitectura en una institución de la salud.
- Incluir costos aproximados de la implementación de la arquitectura.
- Validar cuál es el aporte a la sociedad de la arquitectura propuesta.
- Medir la utilidad final dentro de un hospital.

4. Documentación final

Para el presente trabajo de grado, se entregan los diferentes manuales y anexos que permiten conocer más sobre la configuración de la arquitectura de referencia:

- ANEXO 1 – Documento de las consultas en SQL
- ANEXO 2 – Modelo de datos fuentes
- ANEXO 3 – Documentación Técnica y Administrativa
- ANEXO 4 – Escenarios por atributos de calidad y priorización
- ANEXO 5 – Tablas de Validación TAM

VI. CONCLUSIONES

El principal logro de la arquitectura propuesta es generar un marco de referencia que se especialice en flujos analíticos aplicados en el campo médico. Otros logros que se pueden destacar son aplicar la administración de los flujos de analítica de datos, tener el control sobre los datos que ingresan a la arquitectura, las diferentes posibilidades de implementar la ingestión de los datos con variables requeridas por el usuario y la facilidad de cumplir con una característica modular en cuanto a los dominios y sus respectivos componentes. El ejercicio de crear una arquitectura y su implementación permite visualizar la instanciación concreta y funcional para encontrar la utilidad efectiva.

Se resalta del presente trabajo que el alcance de los resultados obtenidos con la arquitectura de referencia propuesta y su respectiva implementación en la prueba de concepto es una de las cosas más importantes para tener en cuenta debido a su favorabilidad de caminos ya explorados que permiten mejores resultados por parte de los arquitectos que deseen realizar otro tipo de implementaciones concretas a partir de la AR propuesta, es decir, se disminuye en gran medida la generación de errores en las futuras instanciaciones y permite ver el cómo se representa tangiblemente una arquitectura de referencia.

Una de las contribuciones más significativas al conocimiento de acuerdo con los referentes identificados es que la arquitectura proporciona una forma sencilla de integrar diferentes fuentes de datos y procesarlos masivamente para generar múltiples modelos de clasificación. Por otra parte, el poder incluir la metodología general aplicada a un proceso de analítica para la minería de datos incluyendo el procesamiento de lenguaje natural y la posibilidad de administrar cada etapa del flujo de trabajo, es algo realmente significativo que aporta facilidades de implementación de métodos y aplicaciones de analítica que realizan los científicos de datos. Al mismo tiempo, se describe el proceso de diseño de la arquitectura, su implementación iterativa y los diversos medios de verificación y validación usados para llegar a la propuesta, lo cual contribuye al conocimiento asociado a generación de arquitecturas de referencias que en la literatura todavía presenta ambigüedad y carece de métodos o notaciones estandarizadas.

Al inicio se identificó que la base de datos del HUSI no cuenta con una estructura ideal para la manipulación de los datos debido a que se encontraron obstáculos en la exploración como la no existencia de un diccionario de datos ni modelos de referencia y una mala relación de claves foráneas entre tablas. Lo anterior dificultó la exploración inicial haciendo que se intensificaran los tiempos para conocer el significado de los datos.

El trabajo realizado deja como semilla trabajos futuros como mejorar que los componentes no se direccionen al desarrollo sino a la posibilidad de integrarlos teniendo en cuenta solo las interfaces de conexión. Esto permitirá que usuarios menos técnicos puedan utilizar la prueba de concepto de la arquitectura propuesta. También se puede continuar este trabajo con la implementación como prueba de concepto de los componentes faltantes como son la seguridad y la administración desde una aplicación de interfaz de usuario con la posibilidad de generar la clasificación de si un paciente tiene o no apnea del sueño.

REFERENCIAS

- [1] S. A. Pendergrass y D. C. Crawford, «Using Electronic Health Records To Generate Phenotypes For Research», *Curr Protoc Hum Genet*, vol. 100, n.º 1, pp. e80-e80, ene. 2019, doi: 10.1002/cphg.80.
- [2] B. Ozaydin, F. Zengul, N. Oner, y S. S. Feldman, «Healthcare Research and Analytics Data Infrastructure Solution: A Data Warehouse for Health Services Research», *J Med Internet Res*, vol. 22, n.º 6, p. e18579, jun. 2020, doi: 10.2196/18579.
- [3] M. Y. Santos *et al.*, «A Big Data Analytics Architecture for Industry 4.0», en *Recent Advances in Information Systems and Technologies*, 2017, pp. 175-184.
- [4] Y. Wang, L. Kung, y T. A. Byrd, «Big data analytics: Understanding its capabilities and potential benefits for healthcare organizations», *Technological Forecasting and Social Change*, vol. 126, pp. 3-13, 2018, doi: 10.1016/j.techfore.2015.12.019.
- [5] B. N. Nandish y N. Nandish, «Obstructive Sleep Apnea in Older Adults: Diagnosis and Management», *Advances in Family Practice Nursing*, vol. 3, pp. 41-56, may 2021, doi: 10.1016/j.yfpn.2021.01.002.
- [6] M. Paul, «The Impact of Obstructive Sleep Apnea on the Sleep of Critically Ill Patients», *Critical Care Nursing Clinics of North America*, abr. 2021, doi: 10.1016/j.cnc.2021.01.009.
- [7] J. P. Pajaro, R. A. Gonzalez Rivera, J. C. Castellanos Ramirez, P. Hildalgo Martinez, y L. M. Otero Mendoza, «In search of precision for diagnosis of Obstructive Sleep Apnea», *ERJ Open Research*, vol. 7, n.º suppl 7, 2021, doi: 10.1183/23120541.sleepandbreathing-2021.75.
- [8] A. I. Pack, *Sleep Apnea: Pathogenesis, Diagnosis and Treatment*, Second. CRC Press, 2016.
- [9] Friedman M., *Apnea del sueño y roncopatía. Tratamiento médico y quirúrgico*, vol. 61. Madrid, España: Elsevier - Acta Otorrinolaringológica Española, 2010.
- [10] S.-J. Kim y Kim, Ki Beom, *Orthodontics in Obstructive Sleep Apnea Patients - A Guide to Diagnosis, Treatment Planning, and Interventions*, 1.ª ed., 1 vols. Springer, Cham, 2020.
- [11] F. Jiang *et al.*, «Artificial intelligence in healthcare: past, present and future.», *Stroke Vasc Neurol*, vol. 2, n.º 4, pp. 230-243, dic. 2017, doi: 10.1136/svn-2017-000101.
- [12] D. Yacchirema, D. Sarabia-Jácome, C. E. Palau, y M. Esteve, «System for monitoring and supporting the treatment of sleep apnea using IoT and big data», *Pervasive and Mobile Computing*, vol. 50, pp. 25-40, 2018, doi: <https://doi.org/10.1016/j.pmcj.2018.07.007>.

- [13] V. Tam, N. Patel, M. Turcotte, Y. Bossé, G. Paré, y D. Meyre, «Benefits and limitations of genome-wide association studies», *Nature Reviews Genetics*, vol. 20, n.º 8, pp. 467-484, ago. 2019, doi: 10.1038/s41576-019-0127-1.
- [14] J. C. Denny *et al.*, «Systematic comparison of phenome-wide association study of electronic medical record data and genome-wide association study data», *Nature Biotechnology*, vol. 31, n.º 12, pp. 1102-1111, dic. 2013, doi: 10.1038/nbt.2749.
- [15] J. Pépin, S. Bailly, y R. Tamisier, «Big Data in sleep apnoea: Opportunities and challenges», *Respirology*, vol. 25, 2019, doi: 10.1111/resp.13669.
- [16] I. Guyon y A. Elisseeff, «An Introduction to Variable and Feature Selection», *J. Mach. Learn. Res.*, vol. 3, n.º null, pp. 1157-1182, mar. 2003.
- [17] L. Franz, Y. R. Shrestha, y B. Paudel, *A Deep Learning Pipeline for Patient Diagnosis Prediction Using Electronic Health Records*. 2020.
- [18] W. Raghupathi y V. Raghupathi, «Big data analytics in healthcare: promise and potential», *Health Inf Sci Syst*, vol. 2, pp. 3-3, feb. 2014, doi: 10.1186/2047-2501-2-3.
- [19] G. Manogaran, C. Thota, D. Lopez, V. Vijayakumar, K. Abbas, y R. Sundarsekar, «Big Data Knowledge System in Healthcare», 2017, pp. 133-157. doi: 10.1007/978-3-319-49736-5_7.
- [20] D. W. Bates, S. Saria, L. Ohno-Machado, A. Shah, y G. Escobar, «Big data in health care: using analytics to identify and manage high-risk and high-cost patients.», *Health Aff (Millwood)*, vol. 33, n.º 7, pp. 1123-1131, jul. 2014, doi: 10.1377/hlthaff.2014.0041.
- [21] N. El Aboudi y L. Benhlina, «Big Data Management for Healthcare Systems: Architecture, Requirements, and Implementation», *Advances in Bioinformatics*, vol. 2018, 2018, doi: 10.1155/2018/4059018.
- [22] A. Elgammal y B. Krämer, «A Reference Architecture for Smart Digital Platform for Personalized Prevention and Patient Management», 2021, pp. 88-99. doi: 10.1007/978-3-030-73203-5_7.
- [23] A. Galletta, L. Carnevale, A. Bramanti, y M. Fazio, «An Innovative Methodology for Big Data Visualization for Telemedicine», *IEEE Transactions on Industrial Informatics*, vol. PP, pp. 1-1, 2018, doi: 10.1109/TII.2018.2842234.
- [24] G. Manogaran, V. Vijayakumar, R. Varatharajan, P. M K, R. Sundarasekar, y C.-H. Hsu, «Machine Learning Based Big Data Processing Framework for Cancer Diagnosis Using Hidden Markov Model and GM Clustering», *Wireless Personal Communications*, vol. 102, oct. 2018, doi: 10.1007/s11277-017-5044-z.
- [25] J. Wu, L. Zhang, S. Yin, H. Wang, G. Wang, y J. Yuan, «Differential Diagnosis Model of Hypocellular Myelodysplastic Syndrome and Aplastic Anemia Based on the Medical

- Big Data Platform», *Complexity*, vol. 2018, pp. 1-12, nov. 2018, doi: 10.1155/2018/4824350.
- [26] A. Ed-daoudy y K. Maalmi, «A new Internet of Things architecture for real-time prediction of various diseases using machine learning on big data environment», *Journal of Big Data*, vol. 6, nov. 2019, doi: 10.1186/s40537-019-0271-7.
- [27] I. Mistrik, R. Bahsoon, N. Ali, M. Heisel, y B. Maxim, *Software Architecture for Big Data and the Cloud*, 1st ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2017.
- [28] N. El Aboudi y L. Benhlime, «Big Data Management for Healthcare Systems: Architecture, Requirements, and Implementation», *Advances in Bioinformatics*, vol. 2018, pp. 1-10, jun. 2018, doi: 10.1155/2018/4059018.
- [29] Ingo Borchert y L. Alan Winters, «Addressing Impediments to Digital Trade», *CEPR - University of Sussex*, p. 248, abr. 2021.
- [30] H. Gui, R. Zheng, C. Ma, H. Fan, y L. Xu, *An Architecture for Healthcare Big Data Management and Analysis*, vol. 10038. 2016, p. 160. doi: 10.1007/978-3-319-48335-1_17.
- [31] S. Kumar y M. Singh, «Big data analytics for healthcare industry: impact, applications, and tools», *Big Data Mining and Analytics*, vol. 2, n.º 1, pp. 48-57, mar. 2019, doi: 10.26599/BDMA.2018.9020031.
- [32] M Supriya y AJ Deepa, «Machine learning approach on healthcare big data: a review», *Big Data and Information Analytics*, vol. 5, pp. 58-75, 2020, doi: 10.3934/bdia.2020005.
- [33] M. van Geest, B. Tekinerdogan, y C. Catal, «Design of a reference architecture for developing smart warehouses in industry 4.0», *Computers in Industry*, vol. 124, p. 103343, 2021, doi: <https://doi.org/10.1016/j.compind.2020.103343>.
- [34] D. Garlan *et al.*, *Documenting Software Architectures: Views and Beyond*, 2nd ed. Addison-Wesley Professional, 2010.
- [35] H. Cervantes, L. Castro, y P. Velasco-Elizondo, *Arquitectura de Software: Conceptos y Ciclo de Desarrollo*, 1 st. Cengage Learning, 2016.
- [36] «What is a reference architecture?», *Reference Architecture Definition*, may 17, 2021. [En Línea] <https://www.hpe.com/us/en/what-is/reference-architecture.html> (accedido may 17, 2021).
- [37] J. Klein, «Reference Architectures for Big Data Systems». Published: Carnegie Mellon University's Software Engineering Institute Blog, may 22, 2017. [En línea]. Disponible en: <http://insights.sei.cmu.edu/blog/reference-architectures-for-big-data-systems/>

- [38] M. Galster y P. Avgeriou, «Empirically-grounded reference architectures», 2011, pp. 153-158. doi: 10.1145/2000259.2000285.
- [39] J. Klein, R. Buglak, D. Blockow, T. Wuttke, y B. Cooper, «A Reference Architecture for Big Data Systems in the National Security Domain», en *2016 IEEE/ACM 2nd International Workshop on Big Data Software Engineering (BIGDSE)*, 2016, pp. 51-57. doi: 10.1109/BIGDSE.2016.017.
- [40] J.-A. Mondol, «How to Make a Reference Architecture from Scratch!», jun. 18, 2014. <https://www.linkedin.com/pulse/20140619015204-107175994-how-to-make-a-reference-architecture-from-scratch/> (accedido nov. 08, 2021).
- [41] I. Lytra, C. Carrillo, R. Capilla, y U. Zdun, «Quality attributes use in architecture design decision methods: research and practice», *Computing*, vol. 102, n.º 2, pp. 551-572, feb. 2020, doi: 10.1007/s00607-019-00758-9.
- [42] V. H. Publishing, *Archimate 3.1 Specification*. Van Haren Publishing, 2019. [En línea]. Disponible en: <https://books.google.com.co/books?id=kibNywEACAAJ>
- [43] G. Sang, L. Xu, y P. De Vrieze, *A reference architecture for big data systems*. 2016, p. 375. doi: 10.1109/SKIMA.2016.7916249.
- [44] R. Kazman, M. Klein, y P. Clements, *ATAM: Method for Architecture Evaluation*. Carnegie Mellon University's Software Engineering, 2000.
- [45] A. L'Heureux, K. Grolinger, H. El Yamany, y M. Capretz, «Machine Learning With Big Data: Challenges and Approaches», *IEEE Access*, vol. PP, pp. 1-1, 2017, doi: 10.1109/ACCESS.2017.2696365.
- [46] IBM Systems, «A reference architecture for high performance analytics in healthcare and life science», *Technical White Paper*, p. 15, nov. 2017.
- [47] WILLIAM R. HERSH, «Healthcare Data Analytics. Health informatics: practical guide for healthcare and information technology professionals», vol. 6, p. 13, 2014.
- [48] A. Kumar, F. Niu, y C. Ré, «Hazy: Making It Easier to Build and Maintain Big-Data Analytics: Racing to Unleash the Full Potential of Big Data with the Latest Statistical and Machine-Learning Techniques.», *Queue*, vol. 11, n.º 1, pp. 30-46, ene. 2013, doi: 10.1145/2428616.2431055.
- [49] D. A. Tamburri, «Sustainable MLOps: Trends and Challenges», en *2020 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, sep. 2020, pp. 17-23. doi: 10.1109/SYNASC51798.2020.00015.
- [50] Adnan, A. A. Ilham, y S. Usman, «Performance analysis of extract, transform, load (ETL) in apache Hadoop atop NAS storage using ISCSI», en *2017 4th International*

Conference on Computer Applications and Information Processing Technology (CAIPT), 2017, pp. 1-5. doi: 10.1109/CAIPT.2017.8320716.

- [51] A. Khedr, S. Kholeif, y F. Saad, «An Integrated Business Intelligence Framework for Healthcare Analytics», *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 7, pp. 263-270, 2017, doi: 10.23956/ijarcse/SV715/0163.
- [52] A. K. Sahoo, S. Mallik, C. Pradhan, B. S. P. Mishra, R. K. Barik, y H. Das, «Chapter 9 - Intelligence-Based Health Recommendation System Using Big Data Analytics», en *Big Data Analytics for Intelligent Healthcare Management*, N. Dey, H. Das, B. Naik, y H. S. Behera, Eds. Academic Press, 2019, pp. 227-246. doi: 10.1016/B978-0-12-818146-1.00009-X.
- [53] M. Ambigavathi y D. Sridharan, «Big Data Analytics in Healthcare», en *2018 Tenth International Conference on Advanced Computing (ICoAC)*, dic. 2018, pp. 269-276. doi: 10.1109/ICoAC44903.2018.8939061.
- [54] Dr. I. Ghosh, «Handbook of Statistics: Computational Analysis and Understanding of Natural Languages: Principles, Methods and Applications, volume 38 (Chapter on Bayesian Methods)», 2018, pp. 173-196.
- [55] S. Velupillai *et al.*, «Using clinical Natural Language Processing for health outcomes research: Overview and actionable suggestions for future advances», *Journal of biomedical informatics*, vol. 88, pp. 11-19, 2018.
- [56] N. Sharma, R. Sharma, y N. Jindal, «Machine Learning and Deep Learning Applications-A Vision», *Global Transitions Proceedings*, vol. 2, ene. 2021, doi: 10.1016/j.gltip.2021.01.004.
- [57] M. Gianfrancesco, S. Tamang, J. Yazdany, y G. Schmajuk, «Potential Biases in Machine Learning Algorithms Using Electronic Health Record Data», *JAMA Internal Medicine*, vol. 178, ago. 2018, doi: 10.1001/jamainternmed.2018.3763.
- [58] V. Menger, F. Scheepers, y M. Spruit, «Comparing Deep Learning and Classical Machine Learning Approaches for Predicting Inpatient Violence Incidents from Clinical Text», *Applied Sciences*, vol. 8, p. 981, jun. 2018, doi: 10.3390/app8060981.
- [59] Pontificia Universidad Javeriana, «Acuerdos vigentes de licenciamiento de software y plataformas», mar. 12, 2021. <https://www.javeriana.edu.co/dir-tecnologias-de-informacion/software-puj> (accedido nov. 11, 2021).
- [60] F. D. Davis, «Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology», *MIS Quarterly*, vol. 13, n.º 3, pp. 319-340, 1989, doi: 10.2307/249008.