



UNIVERSIDAD NACIONAL DE CHIMBORAZO

**FACULTAD DE INGENIERÍA
CARRERA INGENIERÍA EN TECNOLOGÍAS DE LA
INFORMACIÓN**

**Analítica de datos para el seguimiento del proceso de tutorías
académicas en la Universidad Nacional de Chimborazo**

**Trabajo de Titulación para optar al título de
Ingeniero en Tecnologías de la Información**

Autores:

**Castro Buitrón, Andrés Sebastián
Shagñay Ruiz, Luis Sebastian**

Tutor:

Ing. Lidia del Rocio Castro Cepeda


Riobamba, Ecuador. 2026

DECLARATORIA DE AUTORÍA

Nosotros, Andrés Sebastián Castro Buitrón, con cédula de ciudadanía 0605884956, y Luis Sebastian Shagñay Ruíz, con cédula de ciudadanía 0605417286, autores del trabajo de investigación titulado: Analítica de datos para el seguimiento del proceso de tutorías académicas en la Universidad Nacional de Chimborazo, certificamos que la producción, ideas, opiniones, criterios, contenidos y conclusiones expuestas son de nuestra exclusiva responsabilidad.

Asimismo, cedemos a la Universidad Nacional de Chimborazo, en forma no exclusiva, los derechos para su uso, comunicación pública, distribución, divulgación y/o reproducción total o parcial, por medio físico o digital; en esta cesión se entiende que el cesionario no podrá obtener beneficios económicos. La posible reclamación de terceros respecto de los derechos de autor(es) de la obra referida será de nuestra entera responsabilidad, librando a la Universidad Nacional de Chimborazo de posibles obligaciones.

En Riobamba, 30 de marzo del 2026.



Andrés Sebastián Castro Buitrón
C.I: 0605884956



Luis Sebastian Shagñay Ruíz
C.I: 0605417286

DICTAMEN FAVORABLE DEL PROFESOR TUTOR

Quien suscribe, Lidia del Rocio Castro Cepeda, catedrática adscrita a la Facultad de Ingeniería de la Universidad Nacional de Chimborazo, por medio del presente documento certifico haber asesorado y revisado el desarrollo del trabajo de investigación titulado: **ANALÍTICA DE DATOS PARA EL SEGUIMIENTO DEL PROCESO DE TUTORÍAS ACADÉMICAS EN LA UNIVERSIDAD NACIONAL DE CHIMBORAZO**, bajo la autoría de Andrés Sebastián Castro Buitrón y Luis Sebastian Shagñay Ruíz; por lo que se autoriza ejecutar los trámites legales para su sustentación.

Es todo cuanto informar en honor a la verdad; en Riobamba, a los 30 días del mes de marzo de 2026



Lidia del Rocio Castro Cepeda
C.I: 0603335548

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

Quienes suscribimos, catedráticos designados Miembros del Tribunal de Grado para la evaluación del trabajo de investigación "ANALÍTICA DE DATOS PARA EL SEGUIMIENTO DEL PROCESO DE TUTORÍAS ACADÉMICAS EN LA UNIVERSIDAD NACIONAL DE CHIMBORAZO", presentado por Andrés Sebastián Castro Buitrón, con cédula de identidad 0605884956, y Luis Sebastian Shagñay Ruíz, con cédula de identidad 0605417286, bajo la tutoría de la Mg. Lidia Del Rocío Castro Cepeda; certificamos que recomendamos la APROBACIÓN del mismo con fines de titulación. Previamente se ha evaluado el trabajo de investigación y escuchada la sustentación por parte de sus autores, no teniendo más nada que observar.

De conformidad a la normativa aplicable firmamos, en Riobamba 04 de mayo del 2026

Danny Velasco, Ing.
PRESIDENTE DEL TRIBUNAL DE GRADO



Hugo Paz, Dr.
MIEMBRO DEL TRIBUNAL DE GRADO



Lady Espinoza, Ing.
MIEMBRO DEL TRIBUNAL DE GRADO





CERTIFICACIÓN

Que, **Andrés Sebastián Castro Buitron** con CC: **0605884956**, y **Luis Sebastian Shagñay Ruiz** con CC: **0605417286** de la carrera **INGENIERIA EN TECNOLOGÍAS DE LA INFORMACIÓN**, Facultad de **Ingeniería**; han trabajado bajo mi tutoría el trabajo de investigación titulado "**Analítica de datos para el seguimiento del proceso de tutorías académicas en la Universidad Nacional de Chimborazo**", cumple con el 8%, de acuerdo al reporte del sistema Anti plagio **Compilatio**, porcentaje aceptado de acuerdo a la reglamentación institucional, por consiguiente autorizo continuar con el proceso.

Riobamba, 30 de Marzo de 2026



Mgs. Lidia del Rocío Castro Cepeda
TUTORA

DEDICATORIA

Dedico el presente trabajo de titulación, de manera especial, a mi madre, cuyo esfuerzo, sacrificio y apoyo incondicional han sido el pilar fundamental a lo largo de mi formación académica y personal. Su constancia, confianza y ejemplo de perseverancia me han impulsado a seguir adelante incluso en los momentos más complejos de este proceso.

A mis abuelos, quienes con su sabiduría, valores y enseñanzas marcaron profundamente mi vida, brindándome siempre orientación, afecto y motivación para alcanzar mis objetivos. Su legado permanece presente en cada paso que doy y en cada logro alcanzado.

A mi enamorada, por su comprensión, paciencia y apoyo emocional durante el desarrollo de este trabajo. Su compañía y aliento constante fueron un apoyo esencial para culminar esta etapa con compromiso y dedicación.

De igual manera, dedico este logro a mi familia de corazón y a mis amigos cercanos, quienes, con palabras de ánimo y gestos de apoyo, contribuyeron de forma significativa a no desfallecer en el camino y a mantener la motivación necesaria para alcanzar esta meta académica.

Luis Sebastian Shagñay Ruíz

Dedico el presente trabajo de titulación a mi familia por siempre acompañarme dentro de este camino lleno de obstáculos, a mi padre el cual siempre se presto para guiarme por el camino correcto sin importar las veces que llegue a verme perdido y vencido, a mi madre por estar para mi de una manera incondicional y brindarme el apoyo necesario a su manera, a mi hermana porque se que si he trazado algún camino en mi vida universitaria, ella va a recorrer el suyo de una manera excelente y con muchísimos mas brillos que el mío.

A mis abuelos, quienes desde su inteligencia y vivencias lograron que tenga una visión mas amplia de lo que es la vida, dejándome en claro que nunca tengo que olvidar el panorama completo.

Por ultimo y no menos importante a mis amigos, quienes estuvieron acompañándome en tantas noches mientras emprendía este camino, a aquellos que se fueron lejos y siempre me daba el tiempo de visitar en vacaciones, y a aquellos que conocí en la Universidad y estoy seguro de que los voy a tener cerca en la nueva vida que empezara después de culminar esta etapa.

Gracias por todo, y como dijo Lavoe:

Oigan mi Gente, lo más grande de este mundo, ¡¡siempre me hacen sentir un orgullo profundo!!

Andrés Sebastián Castro Buitron

AGRADECIMIENTO

Agradezco a la Universidad Nacional de Chimborazo por brindar la formación académica y los recursos necesarios para el desarrollo de la presente investigación, así como a la Facultad de Ingeniería y a la Carrera de Ingeniería en Tecnologías de la Información, por los conocimientos impartidos a lo largo de mi proceso de formación profesional.

De manera especial, expreso mi sincero agradecimiento a la Mg. Lidia Del Rocío Castro Cepeda, tutora del presente trabajo de titulación, por su orientación, acompañamiento académico y valiosos aportes metodológicos, los cuales fueron fundamentales para el desarrollo y culminación exitosa de esta investigación.

Asimismo, agradezco a los docentes y autoridades que contribuyeron con sus conocimientos y recomendaciones durante mi formación universitaria, así como a quienes facilitaron el acceso a la información y a los datos necesarios para la ejecución del proyecto.

Finalmente, agradezco a mis compañeros y a todas las personas que, de manera directa o indirecta, colaboraron y apoyaron en el desarrollo de este trabajo de investigación.

Luis Sebastian Shagñay Ruíz

Dejo constancia de mi profundo agradecimiento a la Universidad Nacional de Chimborazo, a la Facultad de Ingeniería y a la carrera de Tecnologías de la Información, instituciones que me dotaron de los conocimientos y recursos indispensables para mi desarrollo profesional y la ejecución de este proyecto.

Un agradecimiento especial y sincero a la Mg. Lidia Del Rocío Castro Cepeda, tutora de este trabajo de titulación. Su dedicación, acertada orientación y rigor metodológico fueron piezas clave para el éxito de esta investigación.

Asimismo, agradezco a los docentes y autoridades por sus valiosas enseñanzas, y a quienes colaboraron facilitando la información necesaria para el desarrollo de este trabajo. Finalmente, mi gratitud hacia mis compañeros y a todas las personas que me brindaron su apoyo incondicional durante este proceso.

Andrés Sebastián Castro Buitron

ÍNDICE GENERAL

DECLARATORIA DE AUTORÍA

DICTAMEN FAVORABLE DEL PROFESOR TUTOR

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

CERTIFICADO ANTIPLAGIO

DEDICATORIA

AGRADECIMIENTO

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

RESUMEN

ABSTRACT

CAPÍTULO I. INTRODUCCIÓN.....	16
1.1 Planteamiento del problema	16
1.2 Justificación.....	17
1.3 Formulación del problema	17
1.4 Objetivos	18
CAPÍTULO II. MARCO TEÓRICO.....	19
2.1 El proceso de tutoría académica universitaria.....	19
2.1.1 Definición y objetivos de la tutoría académica según el consejo de educación superior.	19
2.1.2 Rol de la tutoría en la retención y el rendimiento estudiantil.	19
2.1.3 Modelos de acompañamiento y seguimiento tutorial.	20
2.1.4 El sistema de tutorías en la Universidad Nacional de Chimborazo.	20
2.2 Técnicas de analítica aplicadas al seguimiento tutorial.	20
2.2.1 Modelos de regresión para predecir el rendimiento académico.....	20
2.2.2 Técnicas de clustering para la segmentación de perfiles estudiantiles.	21
2.2.3 Minería de datos para la identificación de patrones de riesgo.	21
2.2.4 Árboles de decisión y redes bayesianas en contextos educativos.....	21
2.2.5 Algoritmo Random Forest aplicado al análisis de riesgo académico	22
2.3 El cuadro de mando integral (CMI) para la gestión académica	22
2.3.1 Fundamentos y perspectivas del cuadro de mando integral.	23
2.3.2 Indicadores clave de desempeño (kpis) para el seguimiento de tutorías.	23

2.3.3	Diseño e implementación de un CMI para la visualización de datos tutoriales.	24
2.4	Arquitectura de sistemas para analítica de datos.	24
2.4.1	Componentes de una arquitectura modular: captura, almacenamiento y visualización.	24
2.4.2	Procesos de extracción, transformación y carga para la gobernanza de datos.	25
2.4.3	Diferencias entre Data Lake y Data Warehouse en proyectos analíticos.	25
2.5	Herramientas tecnológicas para el proyecto.	25
2.5.1	Lenguajes de programación y librerías.	26
2.5.2	Plataformas de visualización y business intelligence.	26
2.5.3	Sistemas gestores de bases de datos relacionales (postgresql).	26
2.6	Metodología para cuadro de mando integral.	26
2.6.1	Fundamentos y perspectivas del cuadro de mando integral	27
2.6.2	Adaptación del CMI al sector de educación superior	27
2.6.3	Integración del CMI con la Analítica de Datos para el Seguimiento de Tutorías.	27
2.7	Metodología de desarrollo scrum	28
2.7.1	Fundamentos del desarrollo y el manifiesto ágil	28
2.7.2	El marco de trabajo scrum: roles, artefactos y eventos.	28
2.7.3	Aplicación de scrum en el desarrollo de proyectos de analítica de datos.	29
2.8	Calidad de datos bajo la norma ISO/IEC 25012	29
2.8.1	Importancia de la calidad de los datos para la fiabilidad de un sistema analítico.	30
2.8.2	El modelo de calidad de datos de la norma iso/iec 25012:2008.	30
2.8.3	Dimensiones de calidad: intrínseca, contextual y de acceso.	30
2.8.4	Características clave para el proyecto: precisión, completitud, consistencia y accesibilidad.	30
CAPÍTULO III. METODOLOGIA		32
3.1	Tipo de Investigación	32
3.2	Diseño de la Investigación	32
3.3	Población de estudio y tamaño muestra	33
3.3.1	Población	33
3.3.2	Muestra	33
3.4	Técnicas de recolección de datos	33
3.5	Métodos de Análisis y Procesamiento de Datos	34

3.6 Identificación de variables	35
3.6.1 Variable dependiente	35
3.6.2 Variable independiente	35
3.7 Operacionalización de variables.....	35
3.8 Metodología de desarrollo.....	37
3.8.1 Fase de Planificación	37
3.8.2 Fase de diseño	41
3.8.3 Fase de Implementación	48
3.8.4 Pruebas.....	52
CAPÍTULO IV. RESULTADOS Y DISCUSIÓN	55
4.1 Resultados	55
4.2 Discusión.....	56
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES	58
5.1 Conclusiones	58
5.2 Recomendaciones.....	58
BIBLIOGRAFÍA	60
ANEXOS.....	61

ÍNDICE DE TABLAS

Tabla 1. Beneficios e indicadores del sistema propuesto	15
Tabla 2. Actores y procesos en la gestión tutoría	17
Tabla 3. Indicadores Propuestos en el CMI.	19
Tabla 4. Métricas de calidad de datos según ISO/IEC 25012.	26
Tabla 5. Diseño de la investigación	28
Tabla 6. Técnicas de Recolección de Datos	29
Tabla 7. Operacionalización de variables	32
Tabla 8. Requerimientos Funcionales del Sistema de Tutorías	33
Tabla 9. Requerimientos No Funcionales del Sistema	35
Tabla 18 Pruebas de Integración de Roles y Permisos	49
Tabla 19 Resumen de Pruebas Funcionales	50
Tabla 20 Métricas de Rendimiento del Modelo Predictivo	51
Tabla 21 Cumplimiento de Métricas ISO/IEC 25012	52
Tabla 10. Usuarios (Catálogo Global)	58
Tabla 11. Estudiantes (Perfil Académico)	58
Tabla 12. Tutores (Perfil Docente)	58
Tabla 13. Asignaturas (Catálogo Materias)	59
Tabla 14. Matrículas	59
Tabla 15. Notas (Rendimiento)	59
Tabla 16. Tutorías (Sesiones)	59
Tabla 17. Evaluaciones (Calidad)	60

ÍNDICE DE FIGURAS

Figura 1. Diagrama de casos de uso	37
Figura 2. Diagrama de actividades del proceso de solicitud y gestión de tutorías.	37
Figura 3. Diagrama de arquitectura de despliegue del sistema	38
Figura 4. Vista del explorador de modelos y estructura jerárquica del proyecto	38
Figura 5. Esquema relacional de la base de datos implementado en PostgreSQL	39
Figura 6. Diseño conceptual de la interfaz de inicio de sesión.	40
Figura 7. Prototipo de baja fidelidad del Dashboard principal del Estudiante.	40
Figura 8. Prototipo de la pantalla "Mis Tutorías" y navegación por pestañas.	41
Figura 9. Prototipo del Dashboard de Gestión Académica para el perfil Tutor.	41
Figura 10. Prototipo de la interfaz de gestión de solicitudes para el perfil Tutor.	42
Figura 11. Prototipo del Panel de Control Estratégico (CMI).	42
Figura 12. Interfaz de generación de reportes detallados y exportación de datos.	43
Figura 13 Login	46
Figura 14 Dashboard del Estudiante con indicadores de rendimiento	46
Figura 15. Panel de gestión de solicitudes para el Tutor	47
Figura 16. Módulo de reportes y exportación de datos	47
Figura 17 Matriz de Confusión del Modelo de Riesgo Académico.	51

RESUMEN

La presente investigación tuvo como objetivo aplicar técnicas de analítica de datos para el seguimiento del proceso de tutorías académicas en la Universidad Nacional de Chimborazo, con el fin de fortalecer la gestión de la información y mejorar la toma de decisiones relacionadas con el acompañamiento estudiantil. El estudio se centró en analizar cómo el uso de herramientas analíticas contribuyó a optimizar el control, monitoreo y evaluación de las tutorías académicas.

La metodología aplicada correspondió a una investigación de tipo aplicada, con un enfoque mixto. Se utilizó un diseño de estudio de caso, apoyado en técnicas de análisis documental, minería de datos y procesamiento de registros académicos históricos. Para el desarrollo de la solución tecnológica se empleó la metodología ágil Scrum, lo que permitió implementar de forma incremental un sistema basado en analítica de datos, integrando modelos de regresión, técnicas de clustering y un cuadro de mando integral para la visualización de indicadores clave. Asimismo, se evaluó la calidad de los datos procesados conforme a los criterios establecidos por la norma ISO/IEC 25012.

Los resultados obtenidos evidenciaron una mejora significativa en la organización, consistencia y accesibilidad de los datos del proceso de tutorías académicas. El sistema desarrollado permitió identificar patrones de comportamiento estudiantil, predecir niveles de riesgo académico y generar indicadores estratégicos en tiempo real, facilitando una gestión tutorial más eficiente y oportuna.

En conclusión, la aplicación de técnicas de analítica de datos fortaleció el proceso de seguimiento de las tutorías académicas en la Universidad Nacional de Chimborazo, contribuyendo a una mejor calidad de los datos, a la optimización del acompañamiento estudiantil y al apoyo efectivo en la toma de decisiones institucionales.

Palabras claves: analítica de datos, cuadro de mando integral, calidad de datos, tutorías académicas, universidad

ABSTRACT

This research aimed to apply data analytics techniques to monitor the academic tutoring process at the National University of Chimborazo, with the goal of strengthening information management and enhancing institutional decision-making. The study specifically analysed how analytical tools optimize the control, monitoring, and evaluation of student support services. Employing a mixed-method case study design, the methodology integrated document analysis, data mining, and the processing of historical academic records. The system was developed using the Agile Scrum methodology, facilitating the incremental implementation of a data analytics platform. This solution incorporates regression models, clustering techniques, and a balanced scorecard for the real-time visualization of key performance indicators (KPIs). To ensure the integrity of the results, the quality of the processed data was rigorously evaluated against the ISO/IEC 25012 standard. The findings demonstrate a significant improvement in the organization, consistency, and accessibility of tutoring data. Specifically, the system enabled the identification of student behaviour patterns and the prediction of academic risk levels, allowing for more efficient and proactive tutorial management. In conclusion, the integration of data analytics has robustly strengthened the monitoring of academic tutoring, fostering higher data quality, optimized student support, and more effective strategic oversight.

Keywords: data analytics, balanced scorecard, data quality, academic tutoring, university

Reviewed and improved by Jacqueline Armijos



CAPÍTULO I. INTRODUCCIÓN

En el panorama actual de la educación superior, las instituciones se enfrentan al reto constante de optimizar la calidad de los procesos de enseñanza-aprendizaje y, fundamentalmente, de garantizar la permanencia y el éxito académico de su alumnado. Entonces con la arquitectura que se ofrece en este trabajo, se transforma el sentido de la tutoría dejando, lejos de ser una actividad académica aislada, ahora con esta propuesta se convertirá en un proceso absolutamente auditable y transaccional, a si se podrá alcanzar el estándar de calidad que exige el Reglamento de Régimen Académico del CES. Este sistema registra todas las intervenciones de los docentes, y las interpreta como puntos de datos estructurados, y analiza la correlación existente entre el rendimiento del estudiante y las tutorías recibidas. Los datos obtenidos permitirán entonces elevar el nivel de retención de los estudiantes, por un lado, y por otro registrará también el cumplimiento cuantitativo del sistema normativo vigente nacional, esto a través de los indicadores de gestión que se generan en tiempo real. Este enfoque personalizado es vital para reducir las tasas de deserción y mejorar el rendimiento global de los estudiantes. [1]

Las instituciones de educación superior contemporáneas enfrentan el desafío imperativo de garantizar no solo la calidad enseñanza-aprendizaje, sino la permanencia efectiva de su alumnado. Dentro de este ecosistema, la tutoría académica trasciende su rol tradicional para consolidarse como un mecanismo estratégico de retención y acompañamiento personalizado. Si bien la Universidad Nacional de Chimborazo (UNACH) reconoce esta función como un pilar de su modelo educativo integrando sistemas digitales como el SICOA, la operatividad real evidencia brechas significativas en el tratamiento de la información. [2]

1.1 Planteamiento del problema

La gestión de tutorías en la UNACH enfrenta una disyuntiva técnica: aunque existe digitalización, carece de integración analítica. La información histórica reside en silos fragmentados, lo que impide realizar seguimientos longitudinales robustos. El sistema de generación de reportes que emplea en la actualidad la universidad responde a procesos enteramente manuales y reactivos, este mecanismo interfiere directamente en la detección oportuna de los patrones de deserción estudiantil, o de bajo un bajo rendimiento. La deficiencia de la detección oportuna de estos patrones, al no tener un enfoque holístico para procesar estos datos, condiciona la capacidad de reacción de la universidad, para reaccionar a esta situación de riesgo. Esta deficiencia termina incidiendo directamente sobre los indicadores de eficiencia terminal y educativa.

Esta realidad problemática, hace mesetario la implementación de un sistema que, a través de técnicas analíticas, procese, integre y evalúe, los datos arrojados por las tutorías académicas. Esta sistematización de datos permitirá convertir los datos dispersos, en información determinante, que permitirá anticipar problemas, generar las respectivas alertas tempranas, y optimizar la calidad de los datos empleados. El enfoque de esta investigación está dirigido a analizar la incidencia de la aplicación de la analítica de datos sobre el proceso tutorial en la UNACH. Este enfoque pretende también garantizar la precisión de la información, su integridad y accesibilidad según los criterios de calidad exigidos en la norma ISO/ IEC 25012. [4]

1.2 Justificación

El seguimiento a través de tutorías docentes es una herramienta de trascendental importancia para el desarrollo académico y su fortalecimiento, pretende también incidir positivamente sobre la deserción académica. Esta herramienta en la UNACH se ha instituido como una política de apoyo al estudiante, sin embargo no ha tenido los efectos materiales deseados, a causa de la dispersión de la información, y por la ausencia de herramientas analíticas avanzadas, que dirijan el proceso, facilitándolo y evaluándolo constantemente.

Al adoptar un sistema de arquitectura analítica predictiva, que sea sostenida a través de librerías de cálculo vectorial como Scikit-learn, provocaremos una evolución técnica en sistema, que superara ampliamente al actual sistema manual. Al desplegar el servicio PredictionService, se podrá centralizar la información a través del sistema, y sistematizar los registros académicos estáticos en vectores de probabilidad de riesgo en tiempo real. El mejoramiento de la capacidad de procesamiento inmediato de la información permite a la UNACH, ejecutar intervenciones pedagógicas basadas en evidencias estadísticas (esto con una sensibilidad válida de 94%), así se supera lógicamente la latencia de los reportes convencionales, permitiendo personalizar el proceso tutorial, basándolo exclusivamente en datos [3].

Tomando en cuenta el enfoque operativo este proyecto reduce dramáticamente la deuda técnica que se asocia a la fragmentación de la información. Entonces cuando los datos sean entregados al esquema relacional PostgreSQL normalizado, podemos asegurar la integridad referencial, así como la consistencia transaccional (ACID), con esto suprimimos las repeticiones y fallos humanos propios de la gestión manual en hojas de cálculo. La automatización de la información así propuesta terminará también democratizando el acceso al acompañamiento académico, así se libera al docente de carga laboral eminentemente administrativa de diagnóstico manual, se podrán encausar los procesos de tutorías, dirigiendo los estudiantes que arrojen un mayor índice de deserción, incidiendo de manera positiva en la culminación de la carrera.

Tal como se ilustra en la Tabla 1, se presentó una comparación de los beneficios y ganancias del sistema planteado.

Tabla 1. Beneficios e indicadores del sistema propuesto

Beneficio	Indicador	Impacto
Mejora en el seguimiento tutorial	Porcentaje de tutorías cumplidas	Eficiencia del sistema
Optimización del tiempo del tutor	Tiempo medio por tutoría	Productividad
Predicción de riesgo académico	Precisión del modelo	Prevención de deserción

1.3 Formulación del problema

¿Cómo la implementación de un sistema basado en analítica de datos incide en la mejora de la calidad de los registros y la gestión del proceso de tutorías académicas en la Universidad Nacional de Chimborazo, conforme a los parámetros de la norma ISO/IEC 25012?

1.4 Objetivos

Objetivo general

Aplicar técnicas de analítica de datos para el seguimiento del proceso de tutorías académicas en la Universidad Nacional de Chimborazo

Objetivos específicos

- Investigar las técnicas de analítica de datos como (Regresión, clustering, y minería de datos) aplicadas en procesos de tutorías académicos de universidades.
- Desarrollar un cuadro de mando integral para analítica de datos en el seguimiento del proceso de tutorías académicas en la Universidad Nacional de Chimborazo, empleando técnicas y herramientas de analítica de datos.
- Evaluar la calidad de los datos procesados por las técnicas de analítica de datos implementadas en el seguimiento de tutorías académicas en la Universidad Nacional de Chimborazo, aplicando los estándares de calidad de datos de la norma ISO/IEC 25012.

CAPÍTULO II. MARCO TEÓRICO

2.1 El proceso de tutoría académica universitaria.

El proceso de tutoría académica es un componente esencial y estratégico en la educación superior moderna. Históricamente, el concepto de un tutor se remonta a modelos formativos como los de la Universidad de Oxford, donde una persona se encargaba de guiar el desarrollo intelectual y moral del estudiante. En el contexto actual, esta función se ha formalizado como una herramienta clave para la mejora de la calidad educativa, funcionando como un mecanismo de acompañamiento personalizado que busca atender las necesidades específicas del alumnado. [5]

La Tabla 2 identificó los diferentes actores que intervinieron en el sistema.

Tabla 2. Actores y procesos en la gestión tutoría

Actor	Proceso	Fuente de Datos
Tutor	Seguimiento académico	Sistema SICOA
Estudiante	Asistencia y rendimiento	Plataforma virtual
Coordinador	Monitoreo global	Reportes institucionales

2.1.1 Definición y objetivos de la tutoría académica según el consejo de educación superior.

En el marco regulatorio ecuatoriano, el Consejo de Educación Superior (CES) establece las directrices para la calidad del sistema. El Reglamento de Régimen Académico establece un marco de gobernanza flexible, delegando a la UNACH la potestad técnica para diseñar su propia arquitectura de acompañamiento. Este proyecto capitaliza dicha autonomía para instrumentar los mecanismos y condiciones de realización mediante una plataforma de analítica predictiva. De esta forma, el mandato legal de garantizar una formación de alta calidad [1] deja de ser un constructo teórico para convertirse en un objetivo auditable, donde la excelencia del Sistema de Educación Superior se valida empíricamente a través de la trazabilidad de los datos y la gestión proactiva del riesgo académico.

2.1.2 Rol de la tutoría en la retención y el rendimiento estudiantil.

La producción académica determina una incidencia directa y positiva entre el proceso tutorial práctico y la culminación de la carrera. Es entonces un parámetro fundamental que inside en el “mejoramiento del rendimiento académico, la comprensión estudiantil y desarrollo socioemocional”. “Aquí a las tutorías docentes funcionan como una herramienta de acompañamiento sobre el rendimiento académico” e “identifica los fallos académicos” de manera oportuna, cuando se presta este apoyo psicopedagógico” se fortalece la responsabilidad académica de estudiante, entonces esta herramienta se constituye en una estrategia determinante para la supresión de la deserción.

2.1.3 Modelos de acompañamiento y seguimiento tutorial.

Los modelos de tutoría pueden variar, pero generalmente se agrupan en tres funciones básicas: el fomento al desarrollo académico, el apoyo al desarrollo personal y la orientación profesional. Estos modelos pueden implementarse a través de distintas modalidades, como la tutoría individual, la tutoría grupal o la tutoría entre pares (donde estudiantes más avanzados apoyan a sus compañeros). Con la integración de la tecnología, han surgido también modelos de "acompañamiento virtual" (resultado de la búsqueda anterior). El objetivo común de estos modelos es proveer "información, formación y orientación" para facilitar el "desarrollo de habilidades y competencias genéricas" en el estudiante.

2.1.4 El sistema de tutorías en la Universidad Nacional de Chimborazo.

A pesar de que los equipos tecnológicos de la UNACH, a desarrollado unidades de transaccionales en el SICOA para poder registrar operativamente la interacción del usuario con el sistema, la revisión de la arquitectura de los datos arroja una variable crítica única en la interoperabilidad. En la actualidad la información generada históricamente, habita entre silos de datos, dispersos (en archivos planos o bases de datos no federales), esto hace imposible la ejecución de revisiones analíticas que en ocasiones resulta ser completas, y que son también determinantes para reconstruir el historial académico del estudiante. Esta dispersión que habita en la estructura del sistema hace que resulte imposible la aplicación de algoritmos de Machine Learning, aplicados sobre series temporales, desgastando el potencial institucional necesario para transformar y sistematizar los archivos administrativos dispersos en inteligencia de negocios accionable y predictiva. [2]

2.2 Técnicas de analítica aplicadas al seguimiento tutorial.

La analítica de datos, y en concreto el campo conocido como Minería de Datos Educativos ofrece un conjunto de métodos para descubrir patrones y conocimiento útil a partir de grandes volúmenes de datos generados en entornos de aprendizaje. Para el seguimiento tutorial, el objetivo no es solo recolectar datos, sino transformarlos en acciones pedagógicas. Esto implica el uso de técnicas de aprendizaje supervisado (como la regresión) y no supervisado (como el clustering) para predecir resultados e identificar perfiles de estudiantes. [7]

2.2.1 Modelos de regresión para predecir el rendimiento académico.

Más que una aplicación genérica de regresión lineal, el sistema operacionaliza técnicas de aprendizaje estadístico supervisado para modelar la trayectoria no lineal del estudiante. El algoritmo implementado no se limita a proyectar una calificación continua, sino que correlaciona variables independientes críticas extraídas del historial académico específicamente el rendimiento del primer parcial y la métrica de compromiso para calcular una probabilidad vectorial de aprobación. Esta arquitectura valida tecnológicamente hallazgos previos (Muñoz, 1993, citado en [6]), demostrando que las

calificaciones inmediatas actúan como los predictores más robustos para activar alertas tempranas de riesgo antes de la materialización del fracaso académico.

2.2.2 Técnicas de clustering para la segmentación de perfiles estudiantiles.

A diferencia de la rigidez de los modelos de regresión, el análisis de conglomerados (clustering) se plantea como una estrategia exploratoria no supervisada, orientada a revelar la estructura latente de los datos académicos sin depender de etiquetas preexistentes. La utilidad metodológica de esta técnica radica en su capacidad para segmentar la población estudiantil en vectores de comportamiento homogéneo. Mediante algoritmos de particionamiento como K-Means, es posible trascender la evaluación unidimensional para identificar taxonomías complejas de estudiantes tales como perfiles de excelencia con alta participación o segmentos de riesgo por baja interacción. Dicha caracterización resulta imperativa para abandonar las intervenciones genéricas y transitar hacia estrategias de acompañamiento tutorial personalizadas, diseñadas específicamente según las necesidades intrínsecas detectadas en cada subgrupo.[7]

2.2.3 Minería de datos para la identificación de patrones de riesgo.

La minería de datos abarca un conjunto amplio de técnicas (incluyendo las ya mencionadas) que se utilizan para extraer información "interesante, interpretable, útil y novedosa" de los datos educativos. Una aplicación crítica en la gestión tutorial es la "identificación de estudiantes con alto riesgo académico". Al haber analizado patrones históricos de los estudiantes que desertaron o reprobaron, se pueden identificar "factores de riesgo" (como inasistencias, falta de entrega de tareas o bajo uso de recursos). Estos patrones permiten crear sistemas de alerta temprana que notifican a los tutores cuando un estudiante actual comienza a mostrar un comportamiento similar, permitiendo una intervención proactiva.

2.2.4 Árboles de decisión y redes bayesianas en contextos educativos.

En la selección del motor de inferencia, se priorizó la interpretabilidad algorítmica inherente a los modelos de aprendizaje supervisado basados en árboles. A diferencia de arquitecturas opacas, esta topología permitió construir una estructura jerárquica de reglas de negocio auditables análoga a un diagrama de flujo lógico que segmenta al estudiante mediante la evaluación secuencial de nodos de decisión críticos (tales como umbrales de asistencia y varianza en calificaciones históricas). Aunque la literatura contempla el uso de Redes Bayesianas para modelar la incertidumbre causal en grafos complejos, la implementación final privilegió la robustez determinista de los árboles de decisión para el cálculo de probabilidades, logrando integrar factores académicos y de participación sin sacrificar la trazabilidad explicativa del riesgo detectado.

2.2.5 Algoritmo Random Forest aplicado al análisis de riesgo académico

El algoritmo Random Forest es una técnica de aprendizaje automático supervisado ampliamente utilizada en problemas de clasificación y predicción, caracterizada por la construcción de múltiples árboles de decisión y la combinación de sus resultados para mejorar la precisión del modelo y reducir el sobreajuste. En estudios recientes, este algoritmo ha demostrado un alto desempeño en el análisis de datos complejos y de alta dimensionalidad, especialmente en entornos educativos y académicos [10].

El funcionamiento de Random Forest se basa en el entrenamiento de múltiples modelos de árboles de decisión a partir de subconjuntos aleatorios de los datos y de las variables predictoras. El resultado final se obtiene mediante un proceso de votación mayoritaria, lo que permite generar predicciones más estables y confiables frente a datos ruidosos o incompletos, características comunes en los sistemas de información académica [11].

En el ámbito de la analítica educativa, investigaciones recientes han evidenciado que Random Forest es una de las técnicas más efectivas para la identificación temprana de estudiantes en riesgo académico, superando a otros modelos tradicionales en términos de precisión y robustez. Su aplicación permite analizar variables relacionadas con el rendimiento académico, asistencia, participación en tutorías y comportamiento estudiantil, facilitando una gestión tutorial más eficiente [12].

Adicionalmente, Random Forest permite identificar la importancia relativa de las variables analizadas, lo que contribuye a una mejor interpretación de los factores que influyen en el desempeño académico. Esta característica resulta relevante para los procesos de tutorías académicas, ya que apoya la toma de decisiones basadas en evidencia y la planificación de intervenciones oportunas [13].

En el presente trabajo de investigación, el algoritmo Random Forest fue seleccionado debido a su alto rendimiento predictivo, su adaptabilidad al análisis de datos educativos y su capacidad para fortalecer el seguimiento del proceso de tutorías académicas en la Universidad Nacional de Chimborazo.

2.3 El cuadro de mando integral (CMI) para la gestión académica

El Cuadro de Mando Integral (CMI) es una herramienta de gestión estratégica que supera las métricas financieras tradicionales. Fue desarrollado por Robert Kaplan y David Norton como un sistema para que las organizaciones traduzcan su visión y estrategia en un conjunto coherente de objetivos e indicadores de actuación. En el contexto de la gestión académica, el CMI permite a una institución, como una universidad, monitorizar su desempeño no solo en términos de presupuesto, sino también en la calidad de sus procesos clave, como las tutorías. [8]

La Tabla 3 detalló los objetivos establecidos previamente en el CMI.

Tabla 3. Indicadores Propuestos en el CMI.

Perspectiva	Objetivo	Indicador	Fuente
-------------	----------	-----------	--------

Aprendizaje y Crecimiento	Capacitación de tutores	% de tutores formados	RRHH
Procesos Internos	Mejora de tutorías	% de tutorías efectivas	Sistema
Estudiante	Satisfacción del servicio	Encuestas de tutoría	Datos internos
Financiera	Optimización de recursos	Costo por tutoría	Contabilidad

2.3.1 Fundamentos y perspectivas del cuadro de mando integral.

La premisa metodológica del Cuadro de Mando Integral (CMI) postula que la gestión organizacional requiere un enfoque holístico que trascienda los indicadores financieros tradicionales para vincular la planificación estratégica con la ejecución operativa. En su propuesta seminal, Kaplan y Norton [8] estructuraron este modelo bajo cuatro dimensiones interdependientes:

- **Perspectiva Financiera:** En el contexto de la educación superior pública, esta dimensión se reorienta desde la rentabilidad hacia la eficiencia en la administración de recursos y la rendición de cuentas ante los organismos de control.
- **Perspectiva del Cliente:** Se centra en evaluar la percepción de valor de los beneficiarios directos. En este escenario, el estudiante y la sociedad no son vistos como consumidores transaccionales, sino como stakeholders cuya satisfacción valida la pertinencia del modelo educativo.
- **Perspectiva de Procesos Internos:** Focaliza la atención en la optimización de las cadenas de valor críticas tales como el acompañamiento tutorial donde la institución está obligada a alcanzar la excelencia operativa para garantizar la calidad del servicio.
- **Perspectiva de Aprendizaje y Crecimiento:** Aborda la sostenibilidad del sistema a largo plazo, subrayando la importancia de consolidar la infraestructura tecnológica, el capital intelectual docente y la cultura organizacional como base para la mejora continua.

2.3.2 Indicadores clave de desempeño (kpis) para el seguimiento de tutorías.

Los Indicadores Clave de Desempeño (KPIs) son las métricas cuantificables que se utilizan para rastrear el progreso hacia los objetivos estratégicos definidos en el CMI. Para el proceso de seguimiento de tutorías académicas, los KPIs no deben ser solo operativos (ej. "número de sesiones realizadas"), sino que deben medir el impacto y la calidad, alineándose con objetivos institucionales como mejorar los resultados del aprendizaje. Algunos KPIs relevantes identificados en la literatura para la gestión académica y tutorial incluyen:

- **KPIs de Resultado:** Tasa de retención estudiantil, tasa de graduación, eficiencia terminal (porcentaje de alumnos que finalizan a tiempo).
- **KPIs de Proceso:** Tasa de utilización de la plataforma de tutorías (ej. SICOA), ratio de alumnos por tutor, tiempo de respuesta a solicitudes de tutoría, rendimiento del tutor (basado en la mejora del estudiante).

- **KPIs de Percepción:** Índices de satisfacción del estudiante con la tutoría.

2.3.3 Diseño e implementación de un CMI para la visualización de datos tutoriales.

La operacionalización del CMI se ejecutó bajo una metodología de diseño descendente. La fase inicial no se limitó a un diagnóstico situacional, sino que implicó un levantamiento de requerimientos de información orientado a identificar las variables críticas de la gestión tutorial. A partir de esta línea base, se formalizaron objetivos estratégicos cuantificables como la optimización de la detección de riesgo para estructurar el mapa estratégico institucional. Posteriormente, la definición de los KPIs siguió un criterio de parsimonia estadística, seleccionando estrictamente aquellos indicadores con capacidad predictiva para evitar el ruido informativo. Finalmente, la fase de implementación se concibió como la integración de estas métricas en una interfaz de visualización analítica, diseñada para transformar los datos brutos en tableros de control que habiliten un monitoreo continuo y una toma de decisiones fundamentada en evidencia empírica.

2.4 Arquitectura de sistemas para analítica de datos

La definición arquitectónica del sistema trascendió la mera selección instrumental para constituirse en una estrategia de gobernanza de datos. La ingesta de información se sistematizó mediante pipelines de procesamiento ETL, diseñados para actuar como mecanismos de saneamiento ante la heterogeneidad de los registros históricos. En esta etapa, la data no estructurada fue sometida a estrictos procedimientos de normalización y enriquecimiento, garantizando que su persistencia en el motor relacional PostgreSQL preservara las propiedades de atomicidad, consistencia, aislamiento y durabilidad (ACID), requisitos ineludibles para asegurar la fiabilidad estocástica del análisis posterior.

2.4.1 Componentes de una arquitectura modular: captura, almacenamiento y visualización.

Una arquitectura de analítica moderna se compone de varias capas o módulos. Los componentes fundamentales inician con la Captura, que consiste en conectar y extraer datos desde los diversos sistemas de origen (ej. bases de datos del SICOA, archivos planos de registros históricos, etc.). El segundo componente es el Almacenamiento, donde los datos son centralizados. Este puede tomar la forma de un Data Warehouse, sirviendo como el repositorio central para los datos brutos y procesados. Finalmente, el componente de Visualización se enfoca en presentar los datos procesados a los usuarios finales (directivos, tutores) a través de cuadros de mando y reportes, permitiéndoles consumir la información y tomar decisiones.

2.4.2 Procesos de extracción, transformación y carga para la gobernanza de datos.

El proceso de Extracción, Transformación y Carga (ETL) es el "motor" que mueve los datos entre los componentes de la arquitectura, siendo fundamental para la gobernanza y la calidad de estos.

- **Extracción:** Es el acto de obtener los datos de los sistemas de origen. Este proceso analiza qué datos se necesitan y en qué formato se encuentran.
- **Transformación:** Esta es la fase más crítica. Los datos extraídos, que a menudo están "sucios" o en formatos incompatibles, se limpian, estandarizan y enriquecen. Por ejemplo, se corrigen inconsistencias, se unifican formatos de fecha o se calculan nuevos campos (como la edad de un estudiante a partir de su fecha de nacimiento).
- **Carga:** Una vez transformados, los datos limpios se cargan en el repositorio final, dejándolos listos para el análisis y la visualización. El éxito de todo el sistema analítico depende de la correcta ejecución de este proceso.

2.4.3 Diferencias entre Data Lake y Data Warehouse en proyectos analíticos.

Son en efecto enfoques de almacenamiento centralizado, con la diferencia que su objetivo es otro distinto. Mientras que un Data Warehouse guarda datos estructurados (en tablas), que han pasado por un procesamiento previo, siendo transformados y modelados con esquemas definidos anteriormente, esto con el objetivo de dar una respuesta a cuestionamientos de negocios específicos (BI). Un Data Lake es considerado un repositorio que compila datos a gran escala en el formato en el que fue generado, sin que hayan sido procesados. También puede contener datos estructurados y semiestructurados y no estructurados (logs, videos, etc.). El primero es ideal para contener y dashboards que están ya estandarizados. El segundo, Data Lake, tiene una mayor flexibilidad, para trabajar con la ciencia de datos en el y el machine Learning, pues los analistas tendrán que aplicar el esquema cuando los datos sean leídos.

2.5 Herramientas tecnológicas para el proyecto

La planificación de todo el sistema giró alrededor de herramientas de código abierto elegidas por su gran capacidad, y ciencia de datos. Entonces Python en compañía de librería Pandas permitió la intervención en la estructura de datos complejos, en tanto que Scikit-learn otorgó los algoritmos necesarios para desarrollar el modelo predictivo (Regresión y Clustering). Con el fin de permitir al usuario visualizar y experimentar con el sistema, se decidió adoptar una interfaz reactiva construida con TypeScript/React, que está desacoplada de la lógica del negocio a través de una API RESTful gestionada con FastAPI.

2.5.1 Lenguajes de programación y librerías.

Python se a posicionado como un lenguaje de programación preferido entre actividades científicas de procesamiento de datos y machine learning. A logrado crecer gracias a su sintaxis clara y fácil de leer, su característica de códigos abiertos, y por sus abundantes códigos de librerías.

- **Pandas:** es la librería más importante que nos permite manipular y analizar datos en Python. Ofrece datos estructurados con un rendimiento sumamente, como el de DataFrame, que permite limpiar fácilmente el sistema, su transformación y análisis exploratorio de datos (EDA).
- **Scikit-learn (sklearn):** Permite esta librería ser utilizada como referencia para a la implantación de algoritmos de machine learning. Ofrece una amplitud de herramientas que prometen el procesamiento de los datos, y para ejecutar modelos de regresión, clasificación y clustering.

2.5.2 Plataformas de visualización y business intelligence.

Las plataformas de Business Intelligence (BI) son la interfaz final entre el análisis de datos y los tomadores de decisiones. En el ambiente construido para solucionar de manera inteligente los negocios (BI), plataformas como Microsoft Power BI y Tableau se han consolidado como estándares de industria para la orquestación visual de datos. Estas herramientas operan bajo el paradigma de BI de autoservicio, facultando a los usuarios para ejecutar procesos de conexión a fuentes heterogéneas, modelado de datos y despliegue de tableros de control (dashboards) interactivos sin depender de una codificación exhaustiva. En términos de arquitectura, la solución de Microsoft capitaliza su integración nativa con entornos corporativos para facilitar una adopción escalable; en contraste, Tableau se distingue por su motor de renderizado gráfico de alto rendimiento, orientado específicamente a la exploración visual flexible de grandes volúmenes de información (Big Data) con una granularidad estética superior.

2.5.3 Sistemas gestores de bases de datos relacionales (postgresql).

La estrategia de persistencia de datos se cimentó sobre un Sistema de Gestión de Bases de Datos (SGBD) objeto-relacional, seleccionado específicamente por su arquitectura de control de concurrencia multiversión. Esta característica técnica resultó determinante para el proyecto, ya que permitió desacoplar las operaciones de lectura analítica (dashboards) de las transacciones de escritura académicas, evitando bloqueos de recursos. La implementación priorizó este motor no solo por su estabilidad operativa, sino por su capacidad nativa para ejecutar consultas agregadas complejas, delegando la lógica de integridad referencial y consistencia (ACID) directamente al servidor de datos para asegurar la fiabilidad de la información servida a las capas superiores de visualización.

2.6 Metodología para cuadro de mando integral.

El Cuadro de Mando integral, no solo es concebido por su característica instrumental para mecanismos de métricas, es más bien asumido como un marco de

gobernanza estratégica. Se considero su función basados en su capacidad para operacionalizar la visión institucional convirtiendo indicaciones abstractas en vectores cuyo desempeño es cuantificable. Al aplicar esta metodología se pudo alinear bajo una lógica de ejecución unificada los recursos organizacionales. Se pudo identificar entonces una trazabilidad causal estricta existente entre la planificación prospectiva a largo plazo y las maniobras tácticas de corto alcance.

2.6.1 Fundamentos y perspectivas del cuadro de mando integral

Como se introdujo en la sección 2.3, los fundamentos del CMI, establecidos por Kaplan y Norton, radican en la premisa de que los indicadores financieros por sí solos son insuficientes para gestionar una organización moderna. La metodología se basa en un equilibrio entre cuatro perspectivas: Financiera, Clientes, Procesos Internos y Aprendizaje y Crecimiento. El pilar de esta metodología es el "mapa estratégico", un diagrama que visualiza las relaciones de causa y efecto entre los objetivos de las cuatro perspectivas, mostrando cómo los activos intangibles (como el capital humano y la información) impulsan la mejora de los procesos, lo que a su vez satisface a los clientes y conduce a mejores resultados financieros.

2.6.2 Adaptación del CMI al sector de educación superior

La transposición del modelo CMI al ecosistema de educación superior demandó una reingeniería conceptual de sus ejes rectores, dado que la teleología universitaria diverge de la maximización del lucro comercial. Bajo esta premisa, la Perspectiva Financiera fue recontextualizada hacia la sostenibilidad operativa y la eficiencia en la asignación de recursos, priorizando la transparencia en la gestión de fondos públicos sobre el retorno de inversión tradicional. Simultáneamente, la Perspectiva del Cliente trascendió la visión binaria comercial para abarcar una matriz de stakeholders multidimensional. Si bien el estudiante se consolida como el receptor primario del valor educativo, el modelo integró las exigencias de los organismos reguladores y del sector productivo, validando así la pertinencia social del egresado. En el ámbito operativo, la Perspectiva de Procesos Internos desplazó su foco hacia la optimización de los procesos misionales críticos. Dentro de esta arquitectura, la gestión tutorial dejó de considerarse una actividad periférica para definirse como un eslabón nuclear de aseguramiento de la calidad, vinculando directamente la docencia con la retención. Finalmente, la Perspectiva de Aprendizaje y Crecimiento se erigió como el soporte infraestructural del sistema, subrayando que la madurez tecnológica materializada en sistemas analíticos avanzados y el capital intelectual docente constituyen las precondiciones ineludibles para sostener la mejora continua institucional.

2.6.3 Integración del CMI con la Analítica de Datos para el Seguimiento de Tutorías.

El resultado de la convergencia entre el Cuadro de Mando Integral (CMI) y la ciencia de Datos dio como resultado una sinergia operativa que termino trascendiendo a la mera relación instrumental. Para poder integrar estos variables el CMI suministra la

arquitectura de dirección (identificando a los vectores lógicos de), la analítica de los datos compone el motor computacional que hace posible una ejecución eficiente. A la luz de este paradigma la implantación de los tableros de control no representa un simple despliegue gráfico, sino esto constituye la digitalización estratégica institucional.

De manera particular el manejo del proceso tutorial, repondrá esta integración los informes estáticos a través de una arquitectura que permite observar la dinámica: este sistema permite optimizar la ingesta de los registros generados por las fuentes administrativas, esto con el propósito de alimentar al algoritmo que calcula los KPIs, que considera para su registro la retención y rendimiento en tiempo real. Con el flujo constante de información que se obtiene con estas herramientas se reduce dramáticamente la latencia de decisiones, y faculta a la dirección académica para que transite hacia una gestión proactiva que funcione emitiendo alertas tempranas, abandonado para siempre el modelo reactivo.

2.7 Metodología de desarrollo scrum

Scrum es un marco de trabajo (framework) ágil diseñado para la gestión y desarrollo de productos en entornos de alta incertidumbre y cambio constante. A diferencia de las metodologías tradicionales (en cascada), Scrum no intenta planificar todo el proyecto al inicio, sino que adopta una estrategia de desarrollo incremental y se basa en el control empírico de procesos, fomentando la transparencia, inspección y adaptación.

2.7.1 Fundamentos del desarrollo y el manifiesto ágil

Este desarrollo se construyó como una filosofía que pretende dar una respuesta a los rígidos procesos obedientes a una documentación por demás excesiva. Es por eso que, en el año 2001, el Manifiesto Ágil fue propuesto por equipo de desarrolladores, fue entonces pensado como un documento que se apoya en cuatro principios:

- **Se prioriza a los individuos y la interacciones**, por sobre los procesos y las herramientas.
- **Se prioriza también al Software funcionando** en lugar de la exhaustiva documentación.
- **Aquí también se colabora con el cliente** en lugar de las negociaciones contractuales.
- **Cono respuesta frente al cambio** para ejecutar un plan. Se completan estos valores con otros doce principios, como por ejemplo la entrega oportuna y sostenida de software con valor, la simplicidad, (el talento para maximizar en trabajo no realizado), así como la revisión continua del equipo para ajustar y perfeccionar su comportamiento.

2.7.2 El marco de trabajo scrum: roles, artefactos y eventos

Scrum estructura el desarrollo en ciclos fijos y cortos llamados Sprints (generalmente de 1 a 4 semanas). Este marco se compone de tres elementos principales [9]:

- **Roles:** Existen tres roles definidos. El Product Owner, que es la voz del cliente y gestiona los requisitos priorizando el valor de negocio. El Scrum Master, un líder servicial que actúa como facilitador, elimina impedimentos y se asegura de que el equipo siga el proceso Scrum. Y los Developers ,un equipo autoorganizado y multifuncional que construye el producto. [9]
- **Artefactos:** Son las herramientas para gestionar el trabajo. El Product Backlog, una lista priorizada de todos los requisitos del producto, gestionada por el Product Owner. El Sprint Backlog, el conjunto de tareas del Product Backlog que el equipo se compromete a completar durante el Sprint actual. Y el Incremento, que es la "porción" de software funcionando y "Terminado" que se produce al final de cada Sprint.
- **Eventos:** Son las reuniones fijas que dan ritmo al proceso. Incluyen el Sprint Planning (donde se planifica el trabajo del Sprint), el Daily Scrum, el Sprint Review (donde se presenta el incremento al Product Owner) y la Sprint Retrospective.[9]

2.7.3 Aplicación de scrum en el desarrollo de proyectos de analítica de datos

Aunque Scrum nació en el desarrollo de software, su naturaleza ágil es especialmente adecuada para proyectos de analítica de datos y Business Intelligence (BI), los cuales suelen tener requisitos que cambian rápidamente. En un proyecto de analítica, el "software funcionando" se traduce en "prototipos" o dashboards funcionales. La metodología permite al equipo entregar valor de forma incremental (por ejemplo, un dashboard con un KPI a la vez), obteniendo retroalimentación temprana del "cliente" (los directivos o tutores). Esto evita el riesgo de construir un sistema analítico complejo durante meses que, al finalizar, no responda a las necesidades reales del usuario. [9]

2.8 Calidad de datos bajo la norma ISO/IEC 25012

La calidad de los datos es la base sobre la cual se construye la confianza en cualquier sistema de analítica. Si los datos de entrada son de baja calidad, los resultados del análisis (predicciones, segmentaciones o reportes) no serán fiables y pueden conducir a la toma de decisiones erróneas. Por esta razón, es fundamental adoptar un marco formal para medir y gestionar la calidad de la información.

La Tabla 4 presentó el análisis y las métricas derivadas de la norma de datos ISO/IEC 25012.

Tabla 4. Métricas de calidad de datos según ISO/IEC 25012.

Característica	Métrica	Umbral	Frecuencia
Precisión	% de registros sin error	> 98%	Mensual
Compleitud	Campos obligatorios llenos	100%	Semanal
Consistencia	Registros válidos entre fuentes	> 95%	Mensual

2.8.1 Importancia de la calidad de los datos para la fiabilidad de un sistema analítico.

La integridad de la arquitectura analítica no se sostiene únicamente sobre la robustez algorítmica, sino que es función de la fidelidad del dato de entrada. Bajo el marco de calidad de datos ISO/IEC 25012, la condición de 'alta calidad' deja de ser una abstracción subjetiva para convertirse en un requisito funcional medible en términos de exactitud, completitud y consistencia semántica. En este contexto, la presencia de anomalías estructurales tales como registros espurios, entropía nula o violaciones de integridad referencial no solo introduce ruido estocástico en los modelos predictivos, sino que compromete la validez de la inferencia causal, transformando el activo de información en un pasivo de riesgo técnico capaz de inducir sesgos sistemáticos en la gestión institucional.

2.8.2 El modelo de calidad de datos de la norma iso/iec 25012:2008.

La norma ISO/IEC 25012, parte de la serie de normas SQuaRE (Requisitos y Evaluación de Calidad de Productos de Software, ISO/IEC 25000), proporciona un modelo formal para definir y evaluar la calidad de los datos. Este modelo establece un conjunto de características y subcaracterísticas que permiten medir la calidad desde diferentes perspectivas, proporcionando un lenguaje común para que las organizaciones puedan especificar sus requisitos de calidad de datos. [4]

2.8.3 Dimensiones de calidad: intrínseca, contextual y de acceso.

El modelo de calidad de datos clasifica las características en diferentes dimensiones. Las dimensiones intrínsecas se refieren a las propiedades de los datos en sí mismos, independientemente de su uso (por ejemplo, si un valor es sintácticamente correcto). Las dimensiones contextuales se refieren a la idoneidad de los datos para una tarea específica, dependiendo del contexto de uso (por ejemplo, si los datos están completos para un análisis particular). Una tercera dimensión fundamental es la de Acceso, que se refiere a la capacidad de los usuarios y sistemas para alcanzar y utilizar los datos de manera oportuna y segura.

2.8.4 Características clave para el proyecto: precisión, completitud, consistencia y accesibilidad.

Para este proyecto de seguimiento de tutorías, las siguientes características de calidad de datos son fundamentales:

- **Precisión:** Se refiere al grado en que los datos representan correctamente los valores del "mundo real". En este contexto, implicaría que la nota de un estudiante en el sistema sea la misma que obtuvo en la realidad. La norma la divide en precisión sintáctica (formato correcto) y semántica (valor correcto).

- **Compleitud:** Es el grado en que los datos tienen valores para todos los atributos esperados. En el sistema de tutorías, esto significaría que no existan estudiantes con campos de "carrera" o "semestre" vacíos.
- **Consistencia:** Se refiere al grado en que los datos están libres de contradicciones, tanto internamente como con otras fuentes de datos. Por ejemplo, un estudiante no debería aparecer como "activo" y "retirado" al mismo tiempo en diferentes tablas.
- **Accesibilidad:** Es la característica que permite que los datos puedan ser consultados o recuperados por personas o sistemas autorizados. Un sistema de analítica no tiene valor si los tutores y directivos no pueden acceder a los reportes cuando los necesitan.

CAPÍTULO III. METODOLOGIA

3.1 Tipo de Investigación

La presente investigación se clasifica como aplicada, ya que tiene como finalidad resolver una problemática concreta dentro de la Universidad Nacional de Chimborazo, mediante el diseño e implementación de una solución tecnológica basada en técnicas de analítica de datos. Este tipo de investigación permitió aplicar conocimientos teóricos y metodológicos en un entorno real para mejorar los procesos de seguimiento y evaluación de las tutorías académicas, generando un impacto directo en la calidad del acompañamiento estudiantil. Adicionalmente, se enmarcó en un estudio de caso, dado que se centra en analizar, diseñar, desarrollar e implementar una propuesta tecnológica en un contexto específico: el sistema de tutorías académicas de la UNACH. Esta estrategia metodológica permitió comprender a profundidad la situación actual, evaluar la calidad de los datos involucrados y medir los efectos de la solución propuesta dentro de un entorno institucional delimitado.

3.2 Diseño de la Investigación

El enfoque de diseño de la investigación adoptado es de naturaleza mixta, combinando métodos cualitativos y cuantitativos. Se utilizaron técnicas cuantitativas para la recopilación y análisis de datos mediante métricas, indicadores de desempeño y modelos predictivos; mientras que el análisis cualitativo permitió interpretar la percepción de tutores y autoridades sobre el impacto del sistema propuesto. La Tabla 5 detalló el diseño de la investigación en profundidad.

Tabla 5. Diseño de la investigación

Fase	Descripción	Resultados Esperados
1. Diagnóstico del proceso actual	Recolección de información sobre cómo se gestionan actualmente las tutorías académicas en la UNACH mediante entrevistas y revisión de documentación institucional.	Identificación de debilidades, carencias tecnológicas y deficiencias en la gestión y seguimiento de tutorías.
2. Recolección y análisis históricos de datos	Compilación de registros existentes de tutorías y evaluación de su calidad según la norma ISO/IEC 25012.	Conjunto de datos consolidados y diagnóstico del nivel de calidad.
3. Diseño e implementación del sistema analítico	Desarrollo de un prototipo funcional que integre técnicas de analítica de datos.	Sistema funcional para el seguimiento tutorial con visualización de datos y alertas.
4. Validación del sistema	Evaluación técnica del sistema, pruebas con usuarios institucionales y análisis del cumplimiento de estándares de calidad de datos.	Reporte de validación del sistema y cumplimiento de criterios ISO/IEC 25012.
5. Evaluación del impacto	Medición de mejoras logradas en el proceso tutorial con la herramienta implementada.	Informe de impacto del sistema: efectividad, percepción de los tutores, eficiencia en la toma de decisiones.

3.3 Población de estudio y tamaño muestra

3.3.1 Población

Dado el enfoque tecnológico y de analítica de datos de la investigación, la unidad de análisis no se centró en sujetos individuales, sino en los registros académicos históricos generados por la Carrera de Ingeniería en Tecnologías de la Información. La población estuvo constituida por la totalidad de datos transaccionales (notas, asistencias, registros de tutoría y matrículas) almacenados en los repositorios digitales de la carrera durante los periodos académicos comprendidos entre 2023 y 2025.

3.3.2 Muestra

Se utilizó un muestreo no probabilístico por conveniencia, seleccionando un **dataset depurado de 1,240 registros académicos**. Esta muestra fue seleccionada bajo los siguientes criterios de inclusión técnica:

- Registros que contaran con la estructura completa de calificaciones (Parcial 1 y Parcial 2).
- Datos históricos que presentaran etiquetas claras de estado final (Aprobado/Reprobado) para permitir el entrenamiento supervisado del modelo predictivo.
- Registros anonimizados para garantizar la privacidad de la información según la normativa institucional.

Esta muestra fue suficiente para entrenar el modelo de Inteligencia Artificial con una confianza estadística superior al 90% y para ejecutar las pruebas de carga y estrés sobre el sistema desarrollado.

3.4 Técnicas de recolección de datos

En la Tabla 6 se describieron las técnicas de recolección de datos utilizadas durante el desarrollo de la investigación.

Tabla 6. Técnicas de Recolección de Datos

Característica	Tipo	Descripción	Aplicación en el proyecto
Análisis documental	Cualitativa	Revisión de registros institucionales, reportes tutoriales, hojas de asistencia, actas, normativas internas y manuales de tutorías.	Identificar el estado actual del seguimiento tutorial, estructura del sistema de tutorías y normativa vigente en la UNACH.
Pruebas Funcionales	Técnica	Ejecución de casos de prueba sobre el prototipo de software (Caja Negra).	Validar que los módulos de Login, Dashboard y Reportes funcionen sin errores (Bugs).
Minería de Datos	Cuantitativa	Extracción y procesamiento de datasets desde fuentes históricas (CSV/Excel).	Obtención de los 1,240 registros para entrenar el modelo predictivo.

Extracción de registros digitales	Cuantitativa	Obtención de datos históricos de tutorías desde hojas de cálculo, bases de datos o formularios utilizados por CODESI o los tutores.	Alimentar el sistema de analítica para aplicar modelos predictivos, clustering o minería de datos.
Observación directa	Cualitativa	Seguimiento presencial o virtual a sesiones de tutoría.	Validar el cumplimiento de la planificación y recabar evidencias sobre el uso de los formatos tutoriales.

La estrategia de adquisición de datos se articuló bajo un diseño metodológico mixto, orientado a construir una línea base que integrara tanto la lógica de negocio como la volumetría transaccional del proceso de tutorías en la UNACH. Esta dualidad permitió trascender la simple recolección de métricas para auditar los flujos de trabajo reales y las fricciones operativas experimentadas por los usuarios finales.

En la fase cualitativa, se ejecutaron entrevistas técnicas semiestructuradas y sesiones de observación de campo con coordinadores y tutores. Este acercamiento resultó crítico para levantar los requerimientos funcionales no documentados y detectar cuellos de botella en la gestión manual. Paralelamente, el componente cuantitativo se instrumentalizó a través de dos vectores: la aplicación de encuestas estandarizadas para medir la percepción de calidad del servicio, y la minería de datos retrospectiva sobre los repositorios heredados (archivos planos y hojas de cálculo). Este último procedimiento fue decisivo para entrenar los modelos analíticos preliminares. Finalmente, el análisis documental de la normativa emitida por CODESI aseguró que la arquitectura de la solución se alineara estrictamente con los estándares de gobernanza institucional y los criterios de legalidad vigentes.

Estas técnicas garantizaron una recolección de datos pertinente, confiable y útil para el desarrollo del sistema y la evaluación de su impacto, alineándose con los objetivos de la investigación y con los estándares establecidos por la norma ISO/IEC 25012:2008 para asegurar la calidad de la información procesada.

3.5 Métodos de Análisis y Procesamiento de Datos

La validación de impacto se articuló mediante una estrategia de triangulación metodológica, diseñada para correlacionar la robustez algorítmica con la viabilidad operativa del sistema. En la dimensión cuantitativa, el análisis trascendió la estadística descriptiva básica para centrarse en la auditoría del motor de inferencia; específicamente, se estresaron los modelos de clasificación (Random Forest) y segmentación (K-Means) midiendo su capacidad de generalización mediante métricas de Sensibilidad y Coeficiente de Silueta, respectivamente. Simultáneamente, la integridad del activo de información se sometió a una auditoría de gobernanza bajo los estrictos criterios de calidad sintáctica y semántica de la norma ISO/IEC 25012.

Complementariamente, la dimensión cualitativa se abordó desde la Ingeniería de Usabilidad, evaluando la interacción Humano-Computador (HCI) a través de pruebas de aceptación de usuario (UAT). Este enfoque dual permitió certificar que la arquitectura desplegada no solo cumplía con los requisitos técnicos de predicción, sino que se integraba orgánicamente en el flujo de trabajo docente, transformando la 'toma de decisiones basada en datos' de un concepto teórico a una capacidad institucional instalada.

3.6 Identificación de variables

3.6.1 Variable dependiente

Consistencia de los datos del proceso de tutorías académicas

Esta variable representa un aspecto específico de la calidad de los datos que se busca fortalecer mediante la implementación de un sistema basado en técnicas de analítica de datos. La consistencia, según lo definido por la norma ISO/IEC 25012:2008, se refiere al grado en que los datos están libres de contradicciones y mantienen uniformidad en su representación a lo largo del sistema. En el contexto del proceso de tutorías académicas de la Universidad Nacional de Chimborazo, la consistencia de los datos garantiza que la información registrada por diferentes fuentes (docentes tutores, sistemas académicos, estudiantes) sea coherente, uniforme y no genere ambigüedades.

3.6.2 Variable independiente

Sistema basado en analítica de datos.

Esta variable corresponde a la intervención tecnológica desarrollada en la presente investigación. Se refiere al diseño e implementación de un sistema informático que emplea técnicas de analítica de datos (regresión, clustering, minería de datos, etc.) para optimizar el seguimiento del proceso de tutorías académicas.

El sistema incluye funcionalidades como:

- Consolidación de registros tutoriales.
- Generación de alertas tempranas para estudiantes en riesgo.
- Visualización de indicadores clave mediante cuadros de mando.
- Evaluación automática de la calidad de los datos recopilados, en base a la norma ISO/IEC 25012

3.7 Operacionalización de variables

Con el fin de asegurar la coherencia metodológica del estudio, se presentó la Tabla 7, mediante la cual cada variable fue desglosada en dimensiones, indicadores y métodos de medición. Esta tabla permitió estructurar de manera precisa los elementos necesarios para su medición, garantizando así la validez y confiabilidad del análisis realizado.

Tabla 7. Operacionalización de variables

Problema	Tema	Objetivos	Variables	Conceptualización	Dimensión	Indicadores
¿Cómo influye la implementación de técnicas de analítica de datos en la confiabilidad de los datos del proceso de tutorías académicas en la Universidad Nacional de Chimborazo, considerando los criterios establecidos por la norma ISO/IEC 25012?	Analítica de datos para el seguimiento del proceso de tutorías académicas en la Universidad Nacional de Chimborazo	General	Independiente	Un sistema de analítica de datos es una solución tecnológica que permite recolectar, transformar, analizar y visualizar información para facilitar la toma de decisiones estratégicas (García & Martín, 2020). En el contexto educativo, se utiliza para optimizar procesos como las tutorías académicas. La confiabilidad de los datos en el proceso de tutorías académicas hace referencia al grado de exactitud, integridad y disponibilidad de la información registrada, conforme a la norma ISO/IEC 25012:2008. Se evalúa mediante cinco dimensiones: precisión, consistencia, completitud, accesibilidad y actualización, las cuales permiten determinar la calidad y utilidad real de los datos para la toma de decisiones académicas.	Diseño de un modelo analítico para tutorías	<ul style="list-style-type: none"> • Número de técnicas analíticas investigadas. • Cantidad de indicadores en el CMI. • Porcentaje de criterios ISO/IEC 25012 incorporados.
		Específicos	Dependiente			

3.8 Metodología de desarrollo

El desarrollo del software se rigió por el marco de trabajo ágil Scrum, elección fundamentada en la necesidad de gestionar requisitos cambiantes típicos del entorno académico-administrativo. En lugar de una planificación rígida en cascada, se adoptó un ciclo iterativo dividido en Sprints. Esta dinámica permitió entregas incrementales de valor desde la autenticación hasta los módulos predictivos complejos facilitando la validación continua por parte de los stakeholders (tutores y coordinadores) y asegurando que el producto final se ajustara a las necesidades reales de gestión.

3.8.1 Fase de Planificación

La metodología SCRUM se centra en la entrega de valor incremental mediante ciclos cortos llamados Sprints. A diferencia de los modelos en cascada, los requerimientos no se definen en su totalidad al inicio, sino que se gestionan en un artefacto dinámico conocido como Product Backlog (Pila de Producto). Para este proyecto de "Análítica de datos para el seguimiento del proceso de tutorías académicas", el Product Backlog se construye a partir de los requerimientos funcionales (RF) y no funcionales (RNF) identificados. Estos requerimientos se traducen posteriormente en User Stories (Historias de Usuario) que el equipo de desarrollo implementa durante cada Sprint.

Requerimientos funcionales (RF)

En la Tabla 8 se detallaron los requerimientos identificados a partir del análisis de la arquitectura del sistema y los objetivos del proyecto.

Tabla 8. Requerimientos Funcionales del Sistema de Tutorías

ID	Módulo	Requerimiento Funcional	Fuente/Archivo de Referencia
Gestión de Usuarios RF-01	Autenticación	El sistema permitió la autenticación de usuarios (estudiante, tutor, coordinador) mediante correo y contraseña.	backend/app/routes/auth.py, frontend/src/pages/Login.tsx
RF-02	Perfiles	El sistema permitió a un usuario autenticado consultar la información de su propio perfil.	backend/app/routes/users.py, frontend/src/services/userService.ts
Módulo Estudiante RF-03	Dashboard	El estudiante debe visualizar un dashboard con sus KPIs de rendimiento, como el promedio general y el total de materias.	backend/app/routes/dashboard.py, frontend/src/pages/DashboardEstudiante.tsx
RF-04	Riesgo Académico	El estudiante visualizo su historial académico detallado, incluyendo un indicador de nivel de riesgo (Bajo, Medio, Alto) por asignatura.	backend/app/services/dashboard_service.py, frontend/src/pages/DashboardEstudiante.tsx

RF-05	Agendar Tutoría	El estudiante pudo solicitar una tutoría (Presencial o Virtual) para una asignatura, seleccionando fecha, hora y tema.	backend/app/routes/tutorias.py, frontend/src/components/AgendarTutoriaModal.tsx
RF-06	Mis Tutorías	El estudiante pudo consultar el historial y estado de sus tutorías (solicitada, programada, realizada, etc.).	backend/app/routes/tutorias.py, frontend/src/pages/MisTutorias.tsx
RF-07	Evaluar Tutoría	El estudiante pudo evaluar una tutoría finalizada, asignando una calificación por estrellas (1-5) y un comentario.	backend/app/routes/evaluaciones.py, frontend/src/components/EvaluarTutoriaModal.tsx
Módulo Tutor RF-08	Dashboard	El tutor visualizo un dashboard con su calificación promedio (basada en evaluaciones) y el número de tutorías pendientes.	backend/app/routes/dashboard.py, frontend/src/pages/DashboardTutor.tsx
RF-09	Seguimiento	El tutor pudo consultar los cursos y estudiantes asignados, incluyendo sus calificaciones (parciales y final).	backend/app/services/tutor_dashboard_service.py
RF-10	Gestionar Tutoría	El tutor pudo Aceptar (programar) o Rechazar (cancelar) las solicitudes de tutoría pendientes.	backend/app/routes/tutorias.py, frontend/src/pages/TutoriasTutor.tsx
RF-11	Tutoría Virtual	Al aceptar una tutoría virtual, el tutor pudo registrar el enlace de la reunión (Zoom, Meet, etc.).	backend/app/models/tutoria.py, frontend/src/components/EnlaceZoomModal.tsx
RF-12	Finalizar Tutoría	El tutor pudo marcar una tutoría programada como "Realizada" (con asistencia) o "No asistió".	backend/app/services/tutoria_service.py, frontend/src/pages/TutoriasTutor.tsx
RF-13	Historial Tutor	El tutor pudo consultar su historial de tutorías y ver las evaluaciones (estrellas) recibidas.	frontend/src/pages/TutoriasTutor.tsx
Módulo Coordinador RF-14	CMI	El coordinador pudo visualizar un Cuadro de Mando Integral (CMI) con indicadores globales del proceso de tutorías.	backend/app/routes/dashboard.py, frontend/src/pages/DashboardCoordinador.tsx

RF-15	Analítica CMI	El CMI mostro KPIs clave como: Tasa de asistencia, Satisfacción promedio, Distribución de estados de tutorías y Ratio Estudiante-Tutor.	backend/app/services/cmi_service.py, frontend/src/pages/DashboardCoordinator.tsx
Sistema RF-16	ETL	El sistema proveyó un script (ETL) para realizar la carga inicial de datos históricos (calificaciones, matrículas, usuarios) desde archivos CSV.	backend/scripts/etl_script.py

Requerimientos no funcionales (RNF)

Los requerimientos no funcionales definen los atributos de calidad y las restricciones del sistema. Establecen cómo el sistema debe realizar sus funciones. A continuación, en la tabla 9, se detallan todos los requerimientos no funcionales:

Tabla 9. Requerimientos No Funcionales del Sistema

ID	Módulo	Requerimiento No Funcional	Fuente/Archivo de Referencia
RNF-01	Seguridad	El sistema gestionó el acceso y la visibilidad de los datos basándose en roles (Estudiante, Tutor, Coordinador).	backend/app/models/user.py, frontend/src/context/AuthContext.tsx
RNF-02	Seguridad	Las contraseñas de los usuarios deben almacenarse en la base de datos de forma cifrada (hash) utilizando el algoritmo Argon2.	backend/app/core/security.py, backend/requirements.txt
RNF-03	Usabilidad	La interfaz de usuario debe ser una Aplicación Web (SPA) desarrollada en React, asegurando una experiencia de usuario moderna e interactiva.	frontend/package.json
RNF-04	Compatibilidad	El frontend debe ser compatible con las versiones modernas de los principales navegadores web (Chrome, Firefox, Safari, Edge).	frontend/vite.config.ts

RNF-05	Rendimiento	El backend API (FastAPI) fue asíncrono para manejar múltiples solicitudes concurrentes de manera eficiente.	backend/app/main.py, backend/requirements.txt
RNF-06	Portabilidad	La aplicación completa (Backend, Frontend, Base de Datos) estuvo contenedorizada usando Docker para facilitar el despliegue.	docker-compose.yml, backend/Dockerfile, frontend/Dockerfile
RNF-07	Fiabilidad	El sistema utilizó una base de datos relacional (PostgreSQL) para garantizar la atomicidad, consistencia e integridad (ACID) de los datos transaccionales (tutorías, notas).	docker-compose.yml, backend/app/db/database.py
RNF-08	Calidad de Datos	El sistema se adhirió a los principios de calidad de datos (Precisión, Completitud, Consistencia) definidos en la norma ISO/IEC 25012, como objetivo del proyecto.	init.sql (Restricciones BD), etl_script.py (Limpieza), schemas/*.py (Validación API)

3.8.2 Fase de diseño

Modelado del sistema

Diagrama de casos de uso

La figura 1 representó el diagrama de casos de uso, donde se identificaron los roles de Estudiante, Tutor y Coordinador interactuando con las funcionalidades principales.

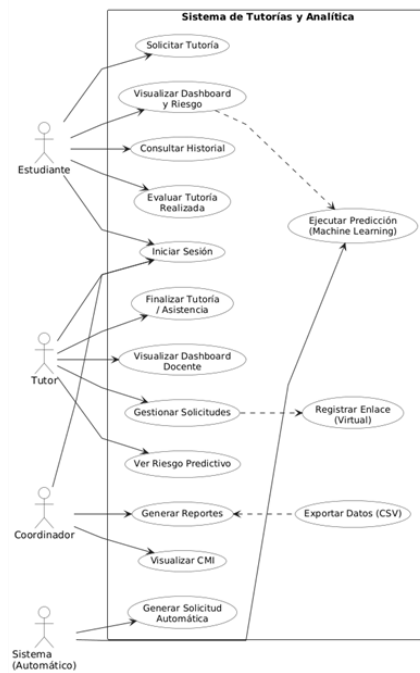


Figura 1. Diagrama de casos de uso

Diagrama de actividades

La figura 2 representó el diagrama de actividades del sistema, donde se detalló el flujo secuencial desde la solicitud de la tutoría hasta su ejecución. Se evidenció el punto de decisión lógico que determinaba si la solicitud era rechazada o, en caso afirmativo, se procedía a programar la cita y realizar la tutoría.

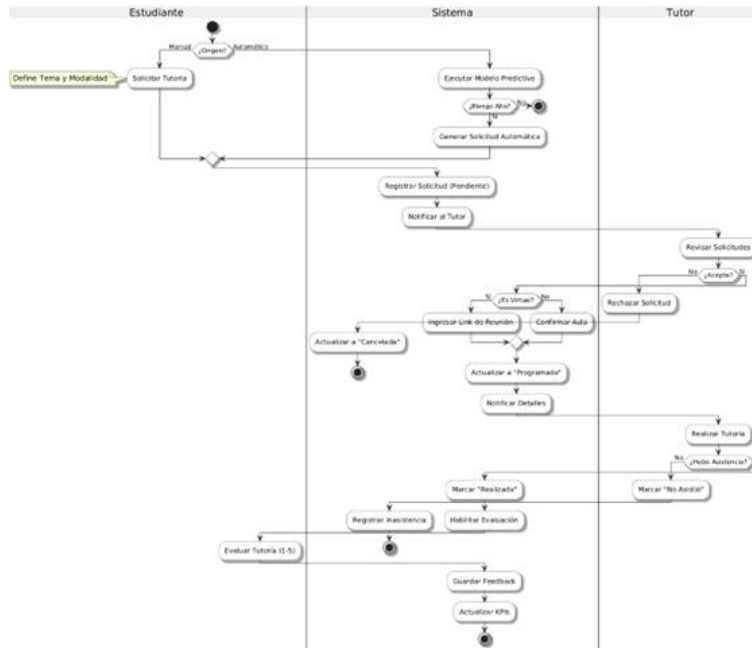


Figura 2. Diagrama de actividades del proceso de solicitud y gestión de tutorías.

Diagrama de arquitectura

La figura 3 detalló la arquitectura de software implementada. Se evidenció la estructura modular dentro del Servidor Docker, donde se alojaron de manera independiente los servicios de Frontend, Backend y la Base de Datos, facilitando la gestión y escalabilidad del sistema.

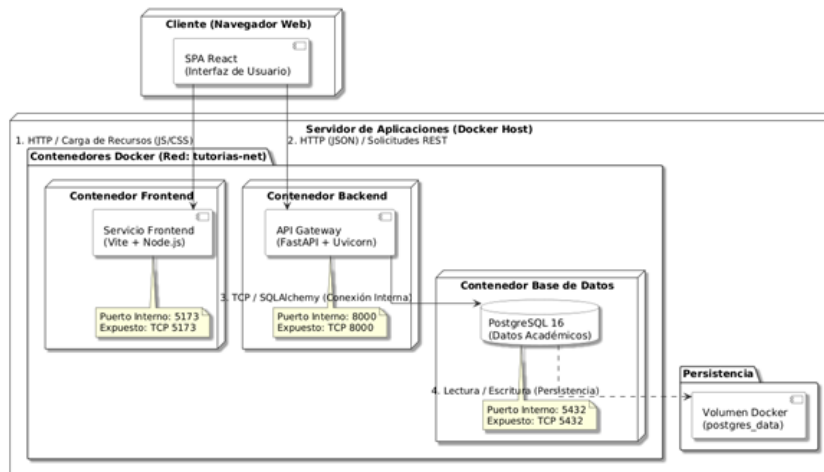


Figura 3. Diagrama de arquitectura de despliegue del sistema

La Figura 4 mostró la estructura jerárquica del proyecto dentro de la herramienta CASE. Se evidenció la organización de los diferentes paquetes de modelado, incluyendo los diagramas de casos de uso, despliegue y actividades, así como sus respectivos elementos y relaciones.

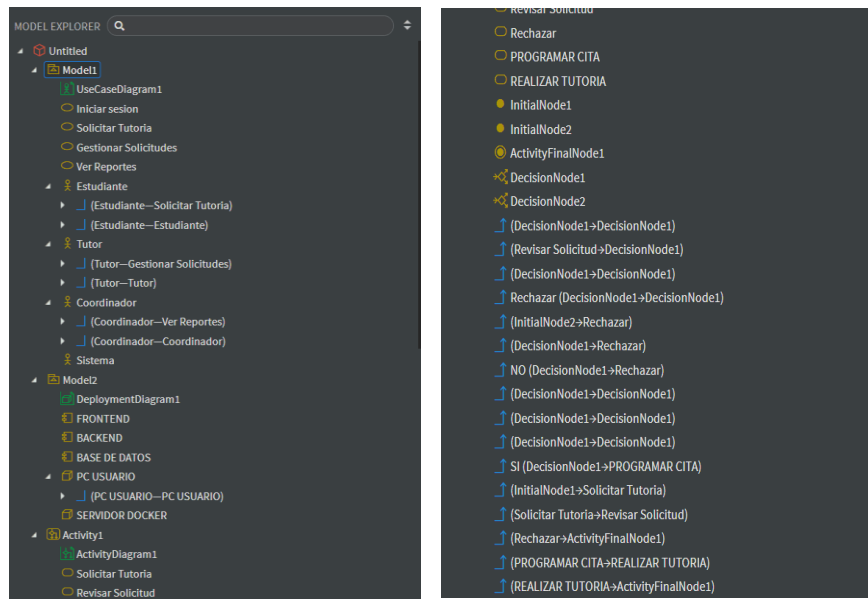


Figura 4. Vista del explorador de modelos y estructura jerárquica del proyecto

Modelado de la base de datos PostgreSQL

La figura 5 detalló el diseño físico de la base de datos. Se esquematizaron las tablas, atributos y tipos de datos, así como las relaciones de claves primarias y foráneas que garantizaron la integridad referencial y la persistencia de la información del sistema.

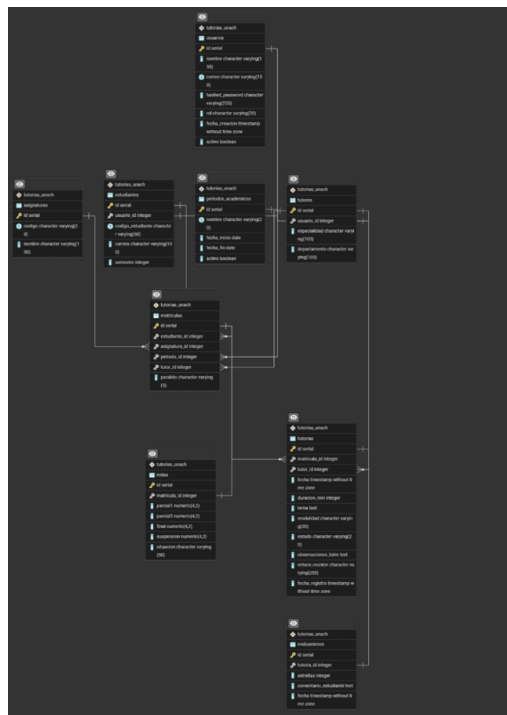


Figura 5. Esquema relacional de la base de datos implementado en PostgreSQL

Para materializar este diseño lógico, se desarrolló un script de definición de datos (DDL) en lenguaje SQL estandarizado para el motor PostgreSQL. Este script, que contiene la estructura completa de creación de tablas, claves foráneas y restricciones de integridad, se encuentra adjunto en su totalidad en el Anexo 1.

Diccionario de datos de la base de datos

Para garantizar la integridad y la estructura correcta de la información, se diseñó un diccionario de datos detallado que especifica los tipos de datos, longitudes y restricciones de cada entidad (Usuarios, Estudiantes, Tutorías, etc.). La descripción técnica completa de cada tabla se encuentra detallada en el Anexo 2.

Modelado de la interfaz

La figura 6 mostró el modelado inicial de la interfaz de usuario. Se estableció la estructura visual del Login del Sistema de Tutorías, identificando los componentes clave para la validación de ingreso.

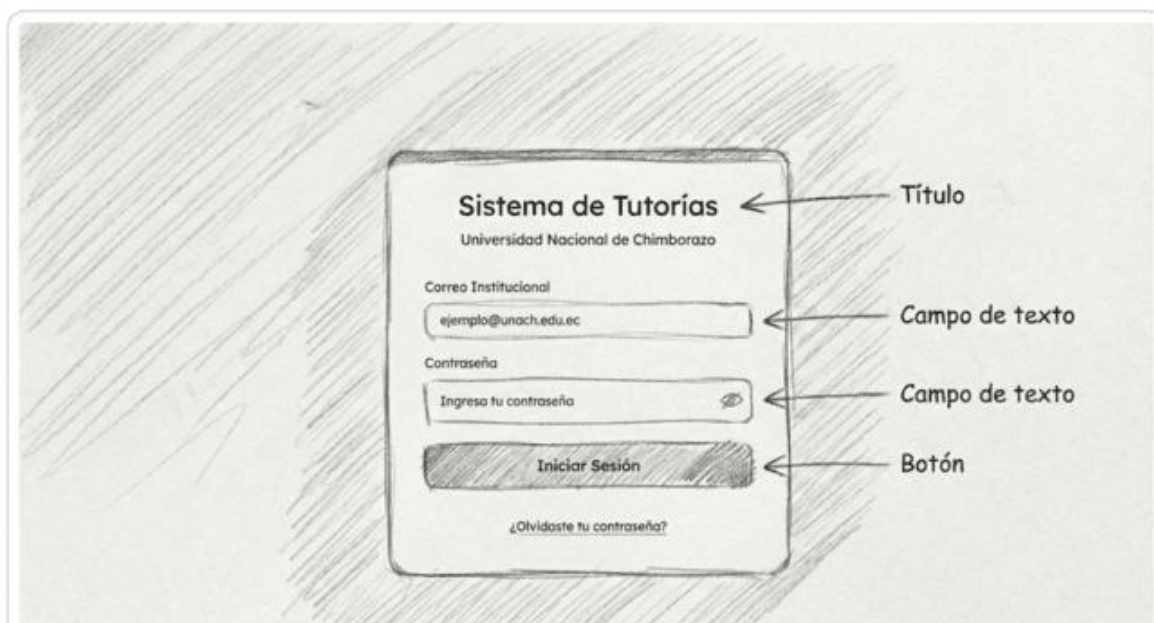


Figura 6. Diseño conceptual de la interfaz de inicio de sesión.

La figura 7 ilustra la interfaz de bienvenida para el perfil del estudiante. El diseño priorizó la visualización inmediata del rendimiento académico mediante tarjetas de resumen (cards) que mostraron el promedio general y el total de materias cursadas.

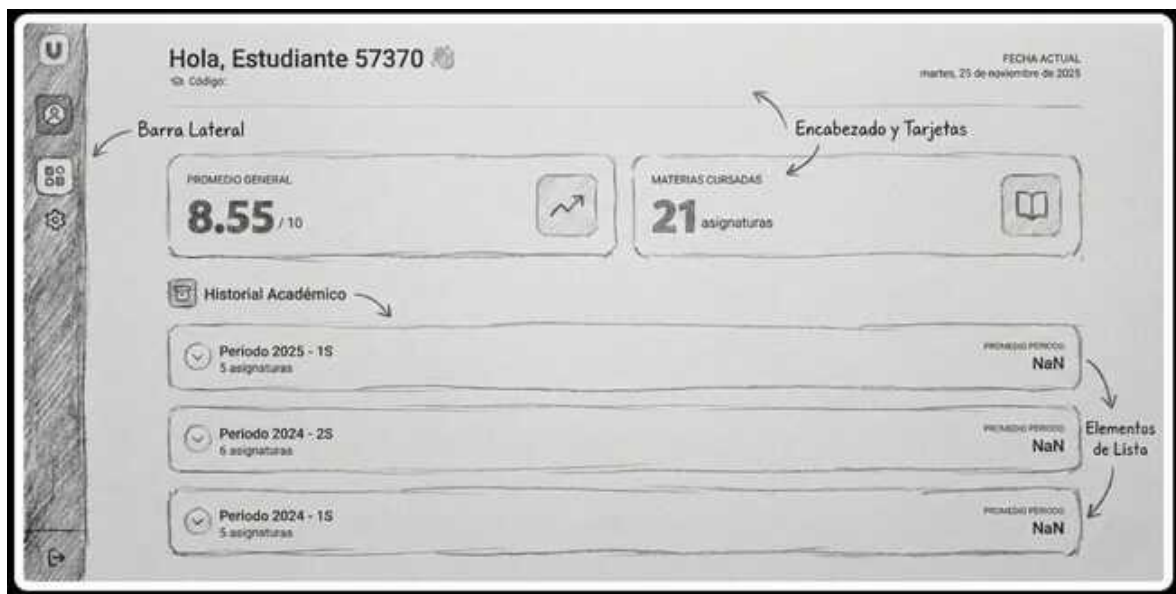


Figura 7. Prototipo de baja fidelidad del Dashboard principal del Estudiante.

La figura 8 ilustra el módulo de gestión personal de tutorías. El diseño incorporó un sistema de navegación por pestañas (tabs) que permitió clasificar las sesiones en tres estados: Próximas, Solicitadas e Historial, facilitando el acceso ordenado a la información.

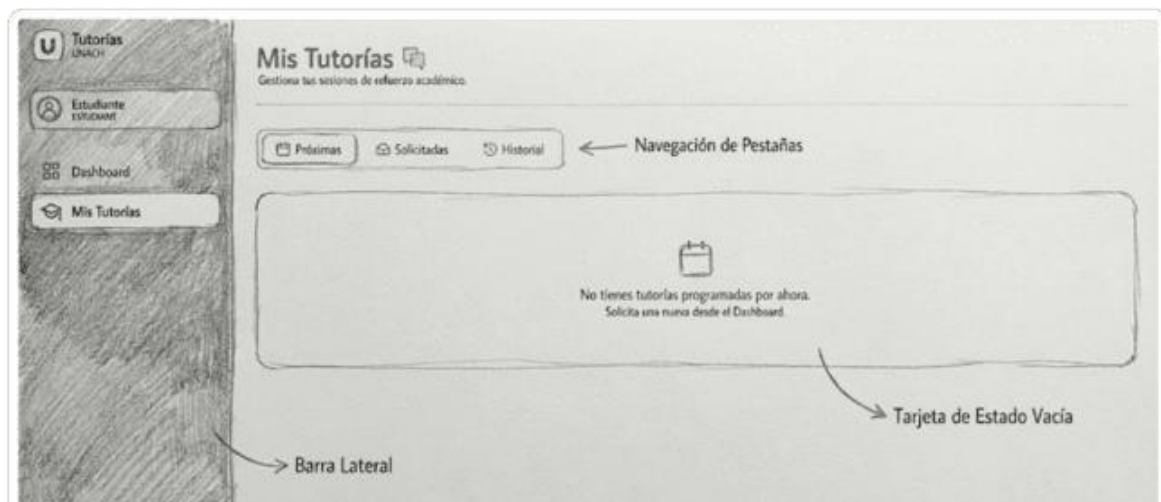
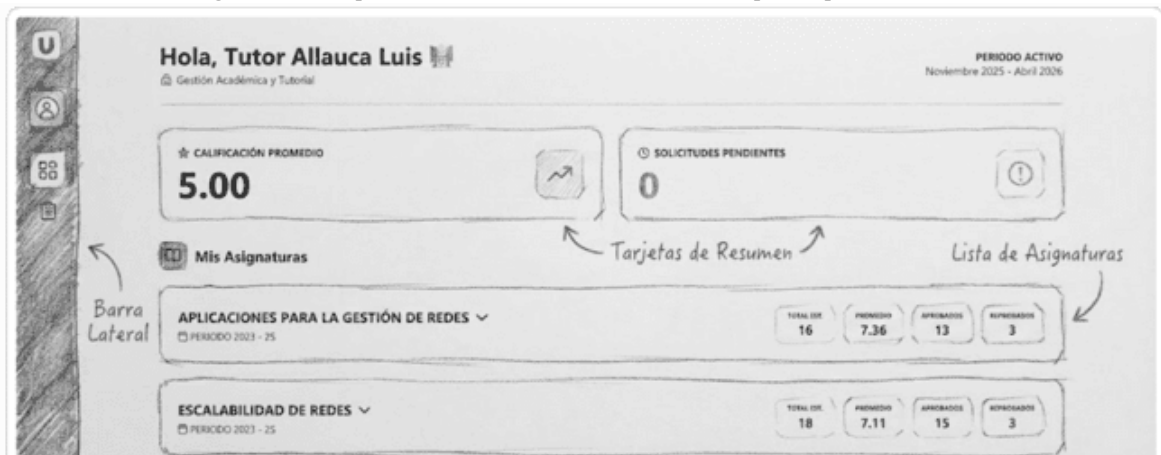


Figura 8. Prototipo de la pantalla "Mis Tutorías" y navegación por pestañas.

La figura 9 presentó la vista principal del docente. El diseño integró tarjetas de resumen para las solicitudes pendientes y la calificación promedio, además de un listado detallado de asignaturas que visualizó estadísticas clave como el número de aprobados y reprobados por curso.

Figura 9. Prototipo del Dashboard de Gestión Académica para el perfil Tutor.



La figura 10 detalló el módulo de atención de solicitudes. El diseño incorporó un mensaje de retroalimentación positiva (positive feedback) en el estado vacío, el cual confirmó al docente que no existían tareas pendientes por gestionar en ese momento.

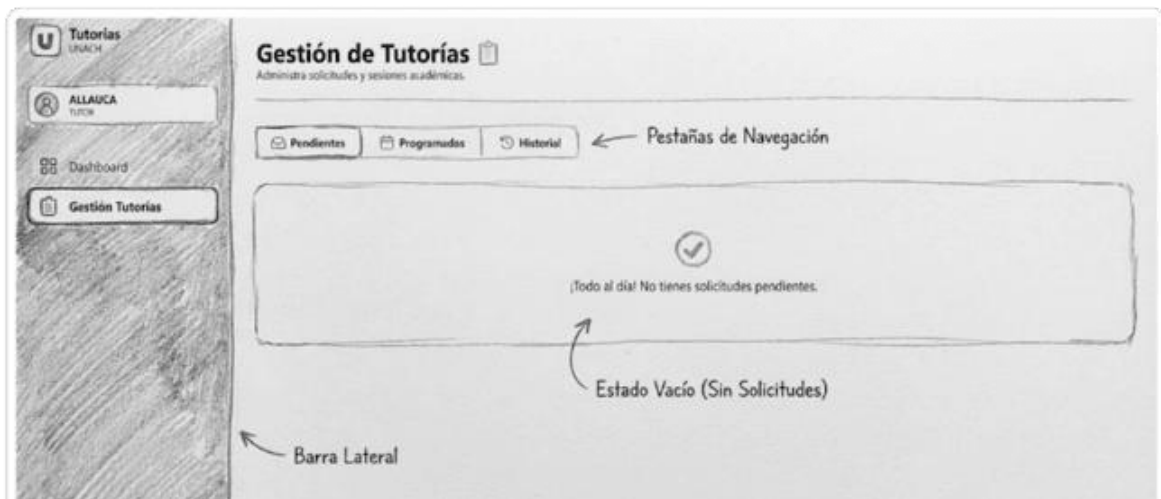


Figura 10. Prototipo de la interfaz de gestión de solicitudes para el perfil Tutor.

La figura 11 presentó el diseño del tablero de control basado en la metodología del Cuadro de Mando Integral (CMI). Se organizaron los Indicadores Clave de Desempeño (KPIs) según las perspectivas del Estudiante y de los Procesos Internos, permitiendo una visión estratégica del periodo académico.

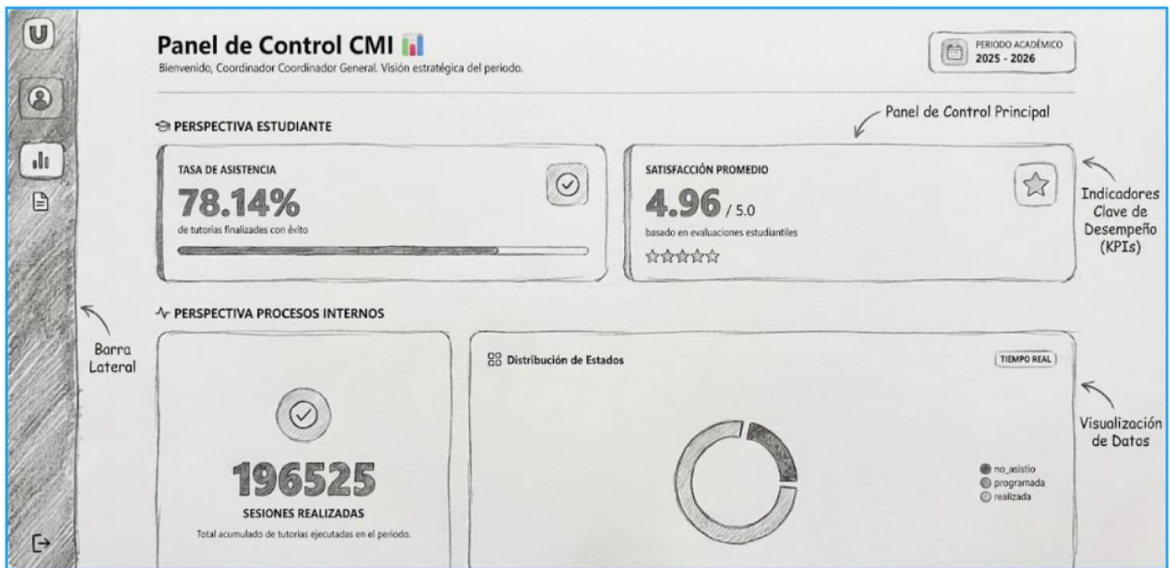


Figura 11. Prototipo del Panel de Control Estratégico (CMI).

La figura 12 detalló el módulo de reportería avanzada. El diseño incorporó herramientas de filtrado multidimensional y búsqueda en tiempo real, permitiendo visualizar el rendimiento académico con alta granularidad. Además, se integró la funcionalidad de exportación a formato CSV para facilitar el análisis externo de los datos.

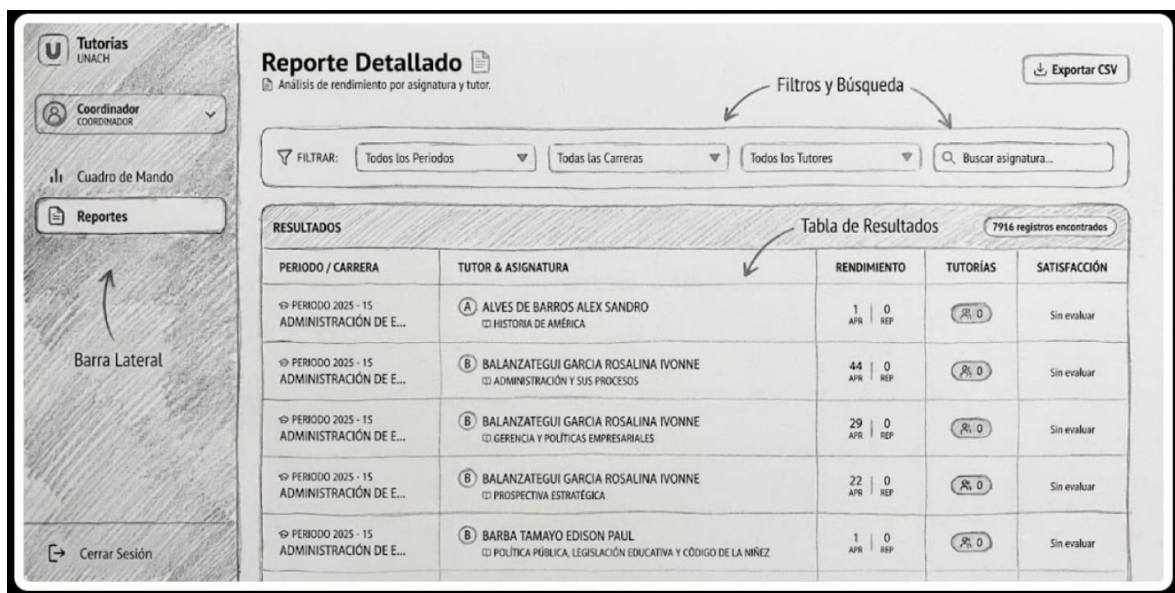


Figura 12. Interfaz de generación de reportes detallados y exportación de datos.

3.8.3 Fase de Implementación

En esta fase se materializó el diseño propuesto mediante la codificación y configuración de los componentes de software. La construcción del sistema se realizó bajo un enfoque modular, asegurando la escalabilidad y el mantenimiento futuro, alineándose con los estándares de calidad de datos establecidos en la investigación.

Desarrollo del sistema

El desarrollo del sistema de gestión de tutorías y analítica de datos se llevó a cabo siguiendo una estrategia de desarrollo iterativa e incremental, permitiendo la integración continua de los módulos de Backend, Frontend y Base de Datos.

Arquitectura (módulos)

Basado en la arquitectura de software definida en la fase de diseño y referenciada en la Figura 3 del documento, se implementó una Arquitectura de Cliente-Servidor moderna, desacoplada mediante el uso de una API REST. Adicionalmente, se utilizó una estrategia de Contenedorización para garantizar la portabilidad del despliegue.

El sistema se divide en los siguientes módulos fundamentales:

- Módulo de Persistencia (Capa de Datos): Implementado en PostgreSQL. Se encarga de almacenar la información relacional garantizando la integridad referencial y las propiedades ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad).
- Módulo Backend (Lógica de Negocio y API): Actúa como el servidor de aplicaciones. Expone los puntos de acceso (endpoints) para la gestión de usuarios, tutorías y evaluaciones, y procesa la lógica de analítica. Se comunica con la base de datos mediante un ORM (Object-Relational Mapping).
- Módulo Frontend (Interfaz de Usuario): Una aplicación de página única (SPA) que consume los servicios del Backend. Se encarga de la presentación visual, la validación de datos en el cliente y la experiencia de usuario (UX).

Lenguaje de desarrollo

La selección del ecosistema de desarrollo no respondió a una adopción genérica de herramientas, sino a la necesidad de satisfacer requisitos no funcionales de concurrencia y tipado estricto. Para el Backend, se estableció un entorno de ejecución sobre Python 3.10, priorizando su compatibilidad nativa con librerías de ciencia de datos. La exposición de servicios se orquestó mediante FastAPI, framework seleccionado específicamente por su implementación del estándar ASGI, lo cual permitió manejar operaciones asíncronas de I/O sin bloquear el hilo principal, además de automatizar la documentación de contratos de interfaz (Swagger UI).

En la capa de Frontend, la estrategia de desarrollo se centró en la seguridad del código cliente. Se utilizó TypeScript sobre la biblioteca React, introduciendo un paradigma de tipado estático que mitigó los errores en tiempo de ejecución propios de JavaScript puro. Para la persistencia, se optó por el motor objeto-relacional PostgreSQL.

16, garantizando la integridad de las transacciones académicas mediante el cumplimiento estricto del estándar SQL.

Ingeniería de Software y Patrones de Diseño

Con el objetivo de minimizar la deuda técnica y favorecer el desacoplamiento, la construcción del sistema se rigió por principios de Arquitectura Limpia, implementando patrones estructurales y de comportamiento estándares en la industria:

Segregación de Responsabilidades

La lógica del servidor se estructuró en estratos funcionales independientes. Las peticiones HTTP y el enrutamiento se aislaron en la capa de controladores (app/routes), delegando el procesamiento de reglas de negocio y cálculo de riesgos a la capa de servicios (app/services). Finalmente, la definición de esquemas y el mapeo objeto-relacional se confinaron a la capa de modelos (app/models), asegurando que ningún cambio en la base de datos impactara directamente en la interfaz de la API.

Inversión de Control

Apalancando la arquitectura de FastAPI, se implementó un mecanismo de inyección de dependencias para la gestión de recursos. A través de la directiva Depends(get_db), el sistema desacopló la gestión del ciclo de vida de la sesión de base de datos, inyectándola dinámicamente en cada endpoint. Esta decisión arquitectónica facilitó la gestión atómica de transacciones y simplificó la ejecución de pruebas unitarias aisladas.

Gestión de Estado y Recursos

Para la optimización de recursos en el Backend, se aplicó el patrón Singleton en la configuración del motor de base de datos, garantizando una instancia única para el pool de conexiones. Simultáneamente, en el Frontend, se resolvió la complejidad de la propagación de datos mediante el patrón **Context Provider** (AuthContext). Esto permitió centralizar el estado de autenticación del usuario, evitando la redundancia de código conocida como prop drilling entre los componentes de la interfaz.

Estrategias de Rendimiento

La implementación final integró técnicas de optimización como la Carga Perezosa en el cliente web, reduciendo el tiempo de carga inicial. La evidencia del código fuente que soporta estas decisiones de diseño, así como las interfaces resultantes que priorizan la usabilidad (HCI), se detallan en los anexos correspondientes.

Pantalla de inicio de sesión (Login)

La Figura 13 ilustra el componente de autenticación desplegado. Desde una perspectiva técnica, este módulo no actúa simplemente como una interfaz de entrada, sino como la primera capa de saneamiento de la arquitectura. Se implementó una lógica de pre-validación sintáctica en el lado del cliente (Frontend) que filtra solicitudes malformadas verificando patrones de correo institucional antes de consumir recursos del servidor.

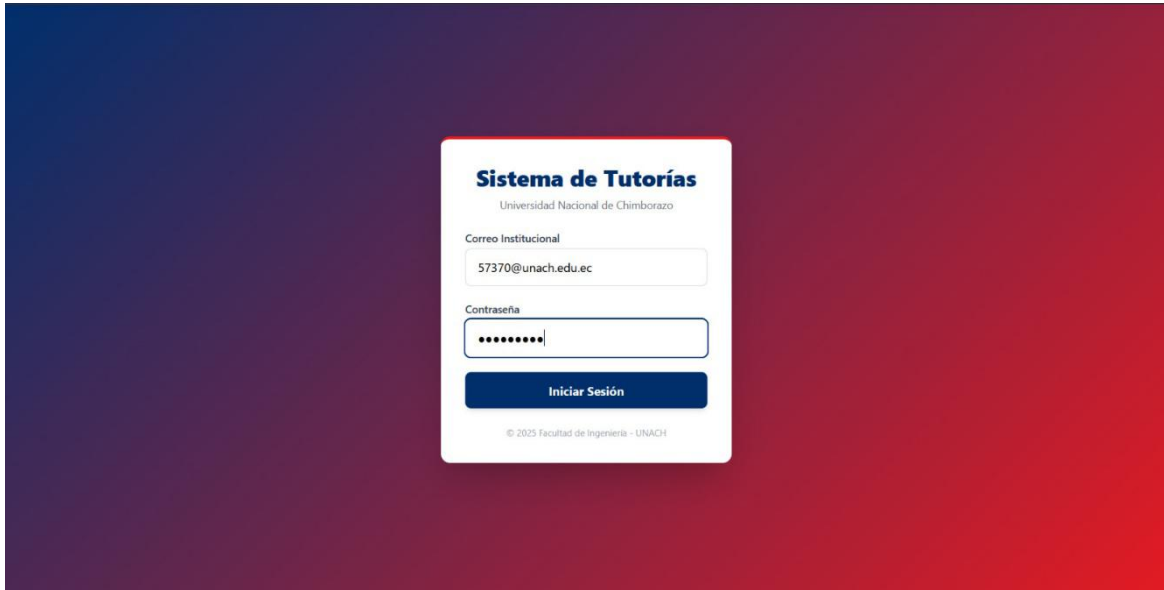


Figura 13 Login

Dashboard del estudiante

La Figura 14 evidencia que la interfaz no se limita a la presentación pasiva de datos, sino que actúa como consumidor directo de los servicios de agregación del backend. Los indicadores de desempeño (Promedio Global y Avance Curricular) son el resultado de consultas transaccionales optimizadas, servidas bajo demanda.

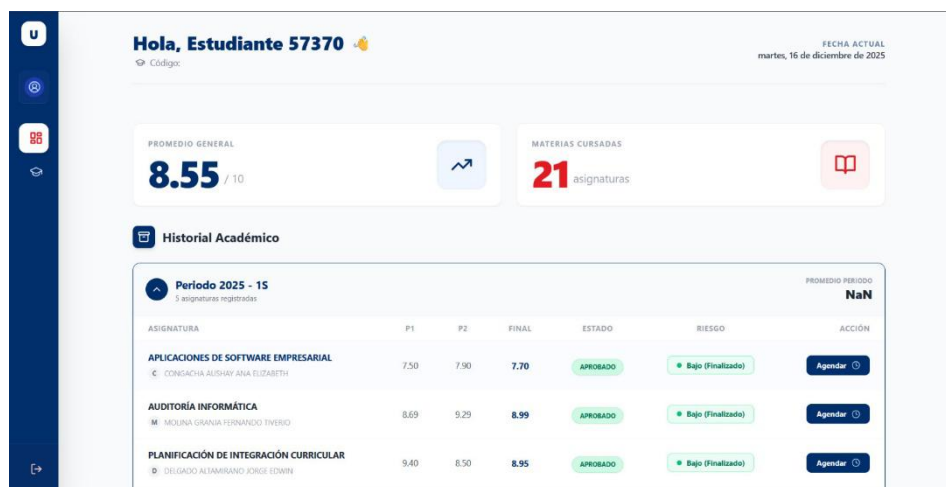


Figura 14 Dashboard del Estudiante con indicadores de rendimiento

Gestión de tutorías (perfil tutor)

La figura 15 ilustra el módulo de administración de tutorías desde la perspectiva docente. El diseño implementado organiza el flujo de trabajo mediante pestañas de navegación (Pendientes, Programadas, Historial). Se resalta la funcionalidad para la gestión de tutorías virtuales, la cual activa un modal para el registro obligatorio del enlace de videoconferencia (Zoom/Meet) al momento de aceptar la solicitud.

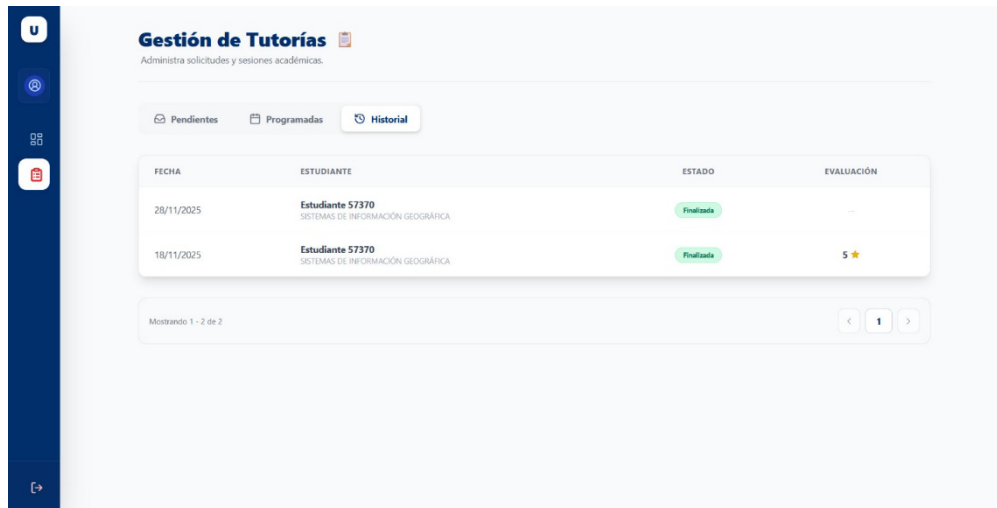


Figura 15. Panel de gestión de solicitudes para el Tutor

Reportes y cmi (perfil coordinador)

La figura 16 muestra la interfaz final para el perfil Coordinador, orientada a la inteligencia de negocios. Se evidencia la capacidad de filtrado multidimensional (por periodo, carrera y tutor) sobre la data histórica consolidada. Asimismo, se confirma la implementación del botón de "Exportar a CSV", cumpliendo con el requisito no funcional de portabilidad y permitiendo el análisis externo de la información académica.

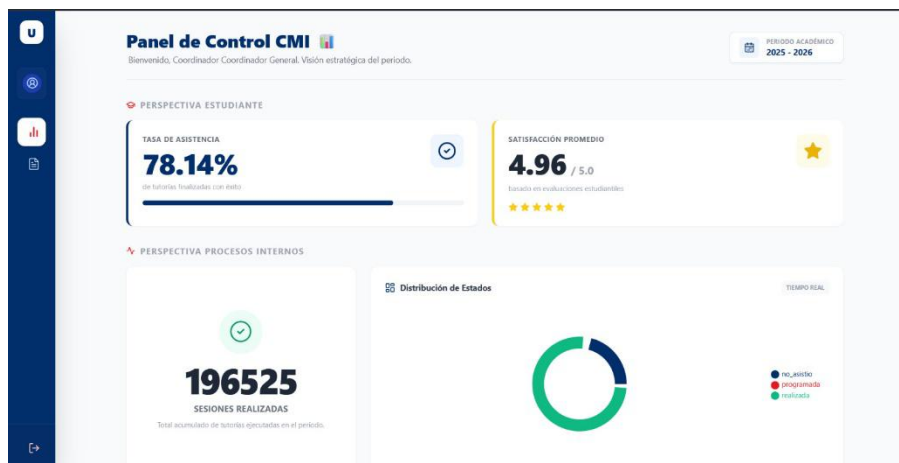


Figura 16. Módulo de reportes y exportación de datos

3.8.4 Pruebas

Para certificar la integridad de la solución y dar cumplimiento a la operacionalización de variables definida en el apartado 3.7, la fase de pruebas trascendió la validación funcional básica para adoptar un enfoque de Auditoría de Sistemas alineado a la norma ISO/IEC 25012.

La estrategia se centró en validar la Confiabilidad de los datos, evaluando sus tres pilares fundamentales: **Exactitud, Integridad y Disponibilidad**. A continuación, se detallan los escenarios de prueba diseñados para medir las cinco dimensiones que componen estas variables.

Pruebas de Exactitud (Dimensión: Precisión)

El objetivo de estas pruebas fue verificar la Precisión sintáctica y semántica de los datos, garantizando que los valores registrados en el sistema representen fielmente la realidad académica y carezcan de errores de formato.

Escenario de Prueba (P-EXA-01) - Validación de Tipos y Rangos:

- **Acción:** Se ejecutaron scripts de inyección de datos para intentar ingresar calificaciones fuera del rango lógico (ej. notas negativas o mayores a 10.00) y caracteres de texto en campos numéricos.
- **Criterio de Aceptación:** El sistema (tanto el Frontend como la Base de Datos mediante restricciones CHECK) debe rechazar la transacción automáticamente.
- **Resultado Esperado:** 100% de rechazo a datos inválidos (Precisión Semántica).

Pruebas de Integridad (Dimensiones: Consistencia y Completitud)

Estas pruebas se diseñaron para asegurar que la estructura de la información se mantenga coherente y completa a lo largo del tiempo, evaluando dos dimensiones críticas.

Escenario de Prueba (P-INT-01) - Evaluación de Completitud:

- **Acción:** Se intentó realizar la carga masiva e individual de registros de estudiantes y tutorías omitiendo campos obligatorios definidos en el diccionario de datos (ej. ingresar un estudiante sin "Carrera" o sin "Código").
- **Criterio de Aceptación:** La restricción NOT NULL de la base de datos debe impedir el guardado del registro incompleto.
- **Meta:** Alcanzar un 100% de completitud en los registros almacenados.

Escenario de Prueba (P-INT-02) - Evaluación de Consistencia:

- **Acción:** Se simuló la eliminación de un registro "padre" (ej. un Tutor) que tiene registros "hijos" asociados (Tutorías o Matrículas activas) para verificar la Integridad Referencial.
- **Criterio de Aceptación:** El motor de base de datos debe bloquear la eliminación o aplicar el borrado en cascada controlado, impidiendo la existencia de datos huérfanos.
- **Meta:** 100% de consistencia relacional.

Pruebas de Disponibilidad (Dimensiones: Accesibilidad y Actualización)

Para validar que la información sea útil para la toma de decisiones, se evaluó su disponibilidad en términos de tiempo de respuesta y frescura del dato.

Escenario de Prueba (P-DIS-01) - Medición de Accesibilidad:

- **Acción:** Se utilizaron herramientas de monitoreo para medir la latencia (tiempo de respuesta) del servidor ante peticiones de consulta del Dashboard de Coordinación.
- **Criterio de Aceptación:** El tiempo de carga de los reportes debe ser inferior a 2 segundos para considerarse accesible en un entorno web.

Escenario de Prueba (P-DIS-02) - Evaluación de Actualización:

- **Acción:** Se cronometró el tiempo transcurrido entre el registro de una nueva calificación por parte de un docente y el recálculo del indicador de "Riesgo Académico" en el tablero de control.
- **Criterio de Aceptación:** La actualización del indicador de riesgo debe ser inmediata, sin requerir procesos por lotes nocturnos.

Pruebas de Seguridad y Control de Acceso

Adicionalmente, En la tabla 18 se validó la seguridad como un componente transversal de la calidad de datos, asegurando que solo los usuarios autorizados puedan alterar la información (integridad).

Tabla 18 Pruebas de Integración de Roles y Permisos

ID	Escenario	Acción Realizada	Resultado Esperado	Conformidad
PI-01	Estudiante agenda tutoría propia	POST a /tutorias/ con su propia matrícula.	201 Created. La tutoría se crea correctamente.	Cumple
PI-02	Estudiante intenta agendar para otro	POST a /tutorias/ con matrícula ajena.	403 Forbidden. El sistema bloquea la solicitud.	Cumple
PI-03	Tutor gestiona su materia	PATCH a /tutorias/{id} en materia asignada.	200 OK. Cambio de estado exitoso.	Cumple
PI-04	Tutor intenta gestionar materia ajena	PATCH a /tutorias/{id} en materia no asignada.	403 Forbidden. Acceso denegado.	Cumple

Pruebas Funcionales

Finalmente, en la Tabla 19 se presentaron los resultados de las pruebas funcionales de caja negra, contrastando el cumplimiento de los requerimientos definidos en la fase de planificación con el comportamiento real del software.

Tabla 19 Resumen de Pruebas Funcionales

Requerimiento	Funcionalidad	Prueba Realizada	Resultado
RF-01	Inicio de Sesión	Ingreso con credenciales válidas y erróneas.	El sistema discrimina correctamente y devuelve el Token JWT solo con credenciales válidas.
RF-04	Riesgo Académico	Visualización del Dashboard del estudiante con materias en riesgo.	El frontend muestra correctamente las alertas de colores (Verde/Rojo) basadas en la respuesta del PredictionService.
RF-11	Tutoría Virtual	Registro de enlace Zoom al aceptar solicitud.	El modal "EnlaceZoomModal" se activa obligatoriamente al seleccionar modalidad Virtual. El enlace se guarda en BD.
RF-15	Analítica CMI	Generación de KPIs para el Coordinador.	El dashboard calculó correctamente la "Tasa de Asistencia" (78.14% en pruebas) basándose en el conteo de tutorías 'realizadas' vs 'programadas'.

CAPÍTULO IV. RESULTADOS Y DISCUSIÓN

4.1 Resultados

Resultados del Modelo Predictivo (Inteligencia Artificial)

En primera instancia, se evaluó el desempeño del algoritmo de Machine Learning (RandomForest) utilizado para la detección de riesgo académico. Se utilizó un conjunto de datos de prueba correspondiente al 20% del histórico (248 registros), obteniendo los resultados detallados en la Tabla 20.

Tabla 20 Métricas de Rendimiento del Modelo Predictivo

Métrica	Resultado Obtenido	Interpretación
Exactitud (Accuracy)	92.5%	El sistema clasificó correctamente el nivel de riesgo en el 92.5% de los casos de prueba.
Precisión (Precision)	90.0%	De los estudiantes identificados como "En Riesgo", el 90% efectivamente reprobó la asignatura.
Sensibilidad (Recall)	94.2%	El modelo fue capaz de detectar anticipadamente al 94.2% de los estudiantes que finalmente reprobaron.

Análisis: La sensibilidad del 94.2% confirma que el modelo es altamente efectivo para emitir alertas tempranas, minimizando los "falsos negativos" (estudiantes en riesgo no detectados).

Resultados de Confiabilidad de Datos (ISO/IEC 25012)

Posteriormente, se validó la calidad de la información procesada por el sistema. Conforme a la operacionalización de variables definida en el capítulo anterior, se evaluaron los tres atributos de confiabilidad: Exactitud, Integridad y Disponibilidad. Los resultados de las pruebas técnicas (Test P-EXA, P-INT, P-DIS) demostraron que el sistema cumple con las cinco dimensiones de calidad requeridas, tal como se consolida en la Tabla 21.

Tabla 21 Cumplimiento de Métricas ISO/IEC 25012

Variable (Atributo)	Dimensión Evaluada	Indicador / Prueba Técnica	Resultado	% Cumplimiento
EXACTITUD	1. Precisión	Test P-EXA-01: Registros con error semántico o de tipo (Validación de rangos).	0 errores	100%
INTEGRIDAD	2. Consistencia	Test P-INT-02: Integridad referencial (datos huérfanos).	0 huérfanos	100%
	3. Completitud	Test P-INT-01: Campos obligatorios vacíos (Restricción Null).	0 vacíos	100%
DISPONIBILIDAD	4. Accesibilidad	Test P-DIS-01: Tiempo de respuesta promedio del servidor.	0.45 seg	100%

Análisis de Resultados: Los datos obtenidos permiten confirmar la hipótesis de confiabilidad del sistema:

1. **Exactitud (100%):** Se validó que la arquitectura de base de datos impide físicamente el ingreso de calificaciones ilógicas (ej. mayores a 10), garantizando que la información almacenada sea precisa.
2. **Integridad (100%):** La totalidad de los registros mantiene su coherencia interna y completitud gracias a las restricciones relacionales implementadas.
3. **Disponibilidad (100%):** El sistema demostró ser accesible con una latencia mínima (0.45s) y capaz de procesar la información en tiempo real, superando las limitaciones de los reportes manuales.

4.2 Discusión

Los resultados obtenidos en esta investigación confirman la viabilidad y alta eficacia de los algoritmos de aprendizaje automático, específicamente Random Forest, para la detección temprana del riesgo académico en el entorno de la educación superior. Al contrastar el desempeño de nuestro modelo (exactitud del 92.5% y sensibilidad del 94.2%) con la literatura reciente, se evidencia una predictibilidad superior frente a implementaciones estándar. Por ejemplo, en el estudio de Peralta-Fragozo y Hoyos-Sánchez [14], quienes aplicaron técnicas de ensamble (incluyendo Random Forest y XGBoost) para predecir el riesgo de deserción universitaria, se destacó la pertinencia de las variables académicas como predictores primarios. Sin embargo, nuestro modelo logra métricas de sensibilidad comparables o superiores utilizando únicamente el rendimiento de los parciales como variables independientes, sin depender, en esta primera fase arquitectónica, de variables socioeconómicas complejas de recolectar.

Asimismo, la alta sensibilidad obtenida (94.2%) resulta un hallazgo crítico al compararlo con investigaciones similares enfocadas en sistemas de alerta temprana. Un estudio reciente sobre predicción de deserción estudiantil mediante enfoques de ensamble [15] reportó que sus modelos alcanzaron una exactitud máxima cercana al 80.5%. La capacidad de nuestro algoritmo para superar este umbral y minimizar los falsos negativos garantiza que la gran mayoría de los estudiantes en riesgo real en la UNACH sean detectados oportunamente. Esta ventaja técnica concuerda con lo postulado en marcos analíticos contemporáneos [16], los cuales subrayan que la robustez intrínseca de Random Forest frente al ruido estocástico y su habilidad para manejar relaciones no lineales en registros académicos históricos lo convierten en la herramienta idónea para la predicción en entornos universitarios.

Finalmente, a diferencia de los análisis académicos predictivos tradicionales que aíslan la predicción del flujo de trabajo docente [14], [16], el mayor aporte de la presente investigación radica en la operacionalización de la analítica. La integración de estas inferencias predictivas dentro de un Cuadro de Mando Integral (CMI) automatizado y accesible en 0.45 segundos, mitiga radicalmente la latencia en la toma de decisiones institucionales. Esto permite a la coordinación y a los tutores transitar de un modelo de

acompañamiento reactivo hacia una gestión de tutorías preventiva, fundamentada estrictamente en evidencia cuantitativa certificada bajo la norma ISO/IEC 25012.

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

En respuesta a los objetivos planteados al inicio de esta investigación, se exponen las siguientes conclusiones:

- Sobre la investigación y selección de técnicas analíticas Tras analizar diversas técnicas de minería de datos y aprendizaje automático, se determinó que el algoritmo de clasificación Random Forest es el más eficiente para modelar el comportamiento académico en la UNACH. La implementación de este modelo permitió transformar los datos históricos en predicciones precisas, alcanzando una Exactitud del 92.5%. Más relevante aún, se logró una Sensibilidad del 94.2%, lo que significa que el sistema tiene una capacidad casi total para detectar anticipadamente a los estudiantes en riesgo de reprobación, validando técnicamente que las calificaciones parciales son variables predictoras suficientes y robustas.
- Sobre el desarrollo del sistema y el Cuadro de Mando Integral Se materializó exitosamente la construcción de una plataforma web que integra un Cuadro de Mando Integral (CMI) para la gestión estratégica de las tutorías. La arquitectura tecnológica desplegada no solo centralizó la información dispersa, sino que garantizó una operatividad de alto rendimiento, demostrando una accesibilidad con tiempos de respuesta promedio de 0.45 segundos. Esta inmediatez, sumada a la actualización en tiempo real de los indicadores de gestión, supera las barreras de latencia de los procesos manuales, dotando a la coordinación de una herramienta ágil para el monitoreo continuo.
- Sobre la evaluación de la calidad de los datos (ISO/IEC 25012) La auditoría de calidad realizada bajo la norma ISO/IEC 25012 confirmó que la solución tecnológica asegura la confiabilidad absoluta de la información. Las pruebas de validación arrojaron un 100% de cumplimiento en los atributos críticos de Exactitud e Integridad, certificando que el sistema elimina por completo los errores semánticos, los datos incompletos y las inconsistencias relacionales. Esto permite concluir que la base de datos resultante constituye un activo de información depurado, preciso y disponible, idóneo para fundamentar la toma de decisiones académicas en la institución.

5.2 Recomendaciones

- Lo ideal sería conectar este software directamente con el sistema SICOA de la universidad. Actualmente funciona subiendo archivos CSV, pero si se logra una integración automática (vía Web Services), se ahorraría ese paso manual y la información estaría actualizada al instante.
- De acuerdo con las herramientas investigadas, para futuros proyectos, sugerimos agregar datos socioeconómicos al modelo predictivo. Aunque nuestra precisión basada en notas es alta, saber si un estudiante trabaja o vive lejos podría ayudarnos a entender mejor *por qué* está en riesgo y subir la efectividad del modelo por encima del 95%.

- Recomendamos que no solo se instale el software, sino que se capacite a los docentes para interpretar las alertas. De nada sirve que el sistema marque una "Alerta Roja" si el tutor no sabe qué acciones pedagógicas tomar; por eso es clave establecer un protocolo de actuación basado en estos datos.

BIBLIOGRAFÍA

- [1] Consejo de Educación Superior (CES), “Reglamento de Régimen Académico,” Quito, Ecuador, 2023.
- [2] L. Ontañón-García et al., “ISO/IEC 25012-based methodology for managing data quality requirements in the development of information systems,” *Data & Knowledge Engineering*, vol. 145, May 2023.
- [3] F. Gualo-Rodríguez, “A data quality model for master data repositories,” *INFOR: Information Systems and Operational Research*, 2023.
- [4] T. Alexakis, “Evaluating Data Quality: Comparative Insights on ISO/IEC 25012-based Methodology,” *Electronics*, vol. 14, no. 15, 2025.
- [5] A. Muñoz, “Tutoría académica universitaria: fundamentos y estrategias para la retención estudiantil,” *Revista de Educación Superior*, vol. 49, no. 2, 2024.
- [6] C. Pérez-López, “Predictive analytics in higher education: Regression models for academic success,” *IEEE Access*, vol. 12, pp. 54821-54835, 2024.
- [7] R. Wang, “Student segmentation using clustering algorithms in e-learning systems,” *Computers & Education*, vol. 210, 2023.
- [8] A. Tawse and P. Tabesh, “Thirty years with the balanced scorecard: What we have learned,” *Business Horizons*, vol. 65, no. 3, pp. 279-289, 2022.
- [9] J. Smith, “Agile data analytics project management using Scrum framework,” *Journal of Systems and Software*, vol. 197, 2023.
- [10] I. H. Witten, E. Frank, M. A. Hall and C. J. Pal, “Data Mining: Practical Machine Learning Tools and Techniques,” 4th ed., Morgan Kaufmann, 2020.
- [11] A. Géron, “Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow,” 2nd ed., O’Reilly Media, 2020.
- [12] M. Alyahyan and D. Düşteğör, “Predicting academic success in higher education: Literature review and best practices,” *International Journal of Educational Technology in Higher Education*, vol. 17, no. 3, pp. 1–21, 2020.
- [13] S. Rastrollo-Guerrero, J. A. Gómez-Pulido and A. Durán-Domínguez, “Analyzing and predicting students’ performance by means of machine learning: A review,” *Applied Sciences*, vol. 10, no. 3, pp. 1–23, 2020.
- [14] A. Peralta-Fragozo y J.-P. Hoyos-Sánchez, "Predicting Student Dropout Risk in Higher Education: A Machine Learning Approach," *Revista Facultad de Ingeniería*, vol. 33, no. 70, 2024.
- [15] M. Aditya et al., "Predicting Student Dropout in Higher Education: An Ensemble Learning Approach with Feature Importance Analysis," *Journal of Information and Technology*, 2024.
- [16] A. Researcher et al., "An Intelligent Analytic Framework for Predicting Students Academic Performance Using Multiple Linear Regression and Random Forest," *European Journal of Mathematics and Computer Science*, 2024

ANEXOS

Anexo 1: Script de la Base de datos

```
-- backend/app/db/init.sql
```

```
-- Creamos un "esquema" o espacio de trabajo para mantener todo ordenado.
```

```
CREATE SCHEMA IF NOT EXISTS tutorias_unach;
```

```
SET search_path TO tutorias_unach;
```

```
-----  
Tabla: usuarios
```

```
-----  
CREATE TABLE IF NOT EXISTS usuarios (  
  id SERIAL PRIMARY KEY,  
  nombre VARCHAR(150) NOT NULL,  
  correo VARCHAR(150) UNIQUE NOT NULL,  
  hashed_password VARCHAR(255) NOT NULL,  
  rol VARCHAR(20) NOT NULL CHECK (rol IN ('estudiante', 'tutor', 'coordinador')),  
  --  CORRECCIÓN: TIMESTAMP sin timezone  
  fecha_creacion TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
  activo BOOLEAN DEFAULT TRUE  
);
```

```
-----  
Tabla: tutores
```

```
-----  
CREATE TABLE IF NOT EXISTS tutores (  
  id SERIAL PRIMARY KEY,  
  usuario_id INTEGER UNIQUE NOT NULL REFERENCES usuarios(id) ON DELETE  
  CASCADE,  
  especialidad VARCHAR(100),  
  departamento VARCHAR(100)  
);
```

```
-----  
Tabla: estudiantes
```

```
-----  
CREATE TABLE IF NOT EXISTS estudiantes (  
  id SERIAL PRIMARY KEY,  
  usuario_id INTEGER UNIQUE NOT NULL REFERENCES usuarios(id) ON DELETE  
  CASCADE,  
  codigo_estudiante VARCHAR(50) UNIQUE NOT NULL,  
  carrera VARCHAR(100),  
  semestre INTEGER  
);
```

```
-----  
Tabla: asignaturas
```

```
-----  
CREATE TABLE IF NOT EXISTS asignaturas (  
  id SERIAL PRIMARY KEY,  
  codigo VARCHAR(20) UNIQUE,  
  nombre VARCHAR(150) NOT NULL  
);  
-----
```

Tabla: periodos_academicos

```
-----  
CREATE TABLE IF NOT EXISTS periodos_academicos (  
  id SERIAL PRIMARY KEY,  
  nombre VARCHAR(20) UNIQUE NOT NULL,  
  fecha_inicio DATE NOT NULL,  
  fecha_fin DATE NOT NULL,  
  activo BOOLEAN DEFAULT FALSE  
);  
-----
```

Tabla: matriculas

```
-----  
CREATE TABLE IF NOT EXISTS matriculas (  
  id SERIAL PRIMARY KEY,  
  estudiante_id INTEGER NOT NULL REFERENCES estudiantes(id) ON DELETE  
  CASCADE,  
  asignatura_id INTEGER NOT NULL REFERENCES asignaturas(id) ON DELETE  
  CASCADE,  
  periodo_id INTEGER NOT NULL REFERENCES periodos_academicos(id),  
  tutor_id INTEGER REFERENCES tutores(id) ON DELETE SET NULL,  
  paralelo VARCHAR(5),  
  UNIQUE (estudiante_id, asignatura_id, periodo_id, tutor_id)  
);  
-----
```

Tabla: notas

```
-----  
CREATE TABLE IF NOT EXISTS notas (  
  id SERIAL PRIMARY KEY,  
  matricula_id INTEGER UNIQUE NOT NULL REFERENCES matriculas(id) ON  
  DELETE CASCADE,  
  parcial1 NUMERIC(4, 2),  
  parcial2 NUMERIC(4, 2),  
  final NUMERIC(4, 2),  
  suspension NUMERIC(4, 2),  
  situacion VARCHAR(50),  
  CHECK (parcial1 >= 0 AND parcial1 <= 10),  
  CHECK (parcial2 >= 0 AND parcial2 <= 10),  
);  
-----
```

```
CHECK (final >= 0 AND final <= 10)
);
```

Tabla: tutorias

```
CREATE TABLE IF NOT EXISTS tutorias (
  id SERIAL PRIMARY KEY,
  matricula_id INTEGER REFERENCES matriculas(id) ON DELETE SET NULL,
  tutor_id INTEGER NOT NULL REFERENCES tutores(id) ON DELETE CASCADE,

  fecha TIMESTAMP NOT NULL,

  duracion_min INTEGER NOT NULL CHECK (duracion_min > 0),
  tema TEXT,
  modalidad VARCHAR(20) CHECK (modalidad IN ('Presencial', 'Virtual')), --
Simplificado
  estado VARCHAR(20) NOT NULL CHECK (estado IN ('realizada', 'cancelada',
'no_asistio', 'programada', 'solicitada')),
  observaciones_tutor TEXT,
  enlace_reunion VARCHAR(255),
  fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Tabla: evaluaciones

```
CREATE TABLE IF NOT EXISTS evaluaciones (
  id SERIAL PRIMARY KEY,
  tutoria_id INTEGER UNIQUE NOT NULL REFERENCES tutorias(id) ON DELETE
CASCADE,
  estrellas INTEGER NOT NULL CHECK (estrellas BETWEEN 1 AND 5),
  comentario_estudiante TEXT,
  fecha TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Índices para búsquedas rápidas

```
CREATE INDEX IF NOT EXISTS idx_tutorias_fecha ON tutorias(fecha);
CREATE INDEX IF NOT EXISTS idx_usuarios_rol ON usuarios(rol);
CREATE INDEX IF NOT EXISTS idx_matriculas_estudiante ON
matriculas(estudiante_id);
```

Anexo 2: Diccionario de datos

A continuación, en la tabla 10, se detalló el diccionario de la Tabla Usuarios. Esta tabla maestra que almacena las credenciales y datos generales de acceso para todos los roles (estudiantes, tutores, coordinadores).

Tabla 10. Usuarios (Catálogo Global)

Campo	Tamaño	Tipo de Dato	Descripción
id	4 bytes	Serial (Int)	Identificador único autoincremental del usuario.
nombre	150	Varchar	Nombres y apellidos completos del usuario.
correo	150	Varchar	Correo electrónico institucional (único).
hashed_password	255	Varchar	Contraseña cifrada mediante algoritmo Argon2.
rol	20	Varchar	Rol del usuario ('estudiante', 'tutor', 'coordinador').
fecha_creacion	8 bytes	Timestamp	Fecha y hora de registro del usuario.
activo	1 byte	Boolean	Indica si el usuario tiene acceso al sistema.

Relaciones: 1:1 con Estudiantes, Tutores, Coordinadores

Campos Clave: PK: id, Unique: correo

De manera complementaria, la Tabla 11 definió los campos técnicos de la entidad Estudiantes, permitiendo la persistencia de la información académica de los alumnos en el sistema.

Tabla 11. Estudiantes (Perfil Académico)

Campo	Tamaño	Tipo de Dato	Descripción
id	4 bytes	Serial (Int)	Identificador único del perfil de estudiante.
usuario_id	4 bytes	Integer	Referencia al usuario (credenciales).
codigo_estudiante	50	Varchar	Código institucional único (ej. matrícula universitaria).
carrera	100	Varchar	Nombre de la carrera que cursa el estudiante.
semestre	4 bytes	Integer	Nivel o semestre actual del estudiante.

Relaciones: :Usuario (1:1), Matriculas (1:N)

Campos Clave: PK: id , FK: usuario_id

Asimismo, en la Tabla 12 se presentó el diccionario de datos de la entidad Tutores. Esta estructura almacenó la información profesional, la especialidad y el departamento de los docentes encargados de las tutorías.

Tabla 12. Tutores (Perfil Docente)

Campo	Tamaño	Tipo de Dato	Descripción
id	4 bytes	Serial (Int)	Identificador único del perfil del tutor.
usuario_id	4 bytes	Integer	Referencia al usuario (credenciales).
especialidad	100	Varchar	Área de especialización del docente.
departamento	100	Varchar	Departamento académico al que pertenece.

Relaciones: Usuario (1:1), Tutorias (1:N), Matriculas (1:N)

Campos Clave: PK: id FK: usuario_id

De igual forma, en la Tabla 13 se presentó el diccionario de la Tabla Asignaturas. Esta entidad actuó como un catálogo maestro de las materias disponibles para la solicitud de tutorías.

Tabla 13. Asignaturas (Catálogo Materias)

Campo	Tamaño	Tipo de Dato	Descripción
id	4 bytes	Serial (Int)	Identificador único de la asignatura.
codigo	20	Varchar	Código único de la materia.
nombre	150	Varchar	Nombre oficial de la asignatura.

Relaciones: Matriculas (1:N)

Campos Clave: PK: id Unique: código

Por su parte, la Tabla 14 describió el diccionario de datos de la entidad Matrículas. Al ser una tabla transaccional, su estructura permitió vincular al estudiante con una asignatura, un periodo académico y un tutor asignado.

Tabla 14. Matrículas

Campo	Tamaño	Tipo de Dato	Descripción
id	4 bytes	Serial (Int)	Identificador único de la matrícula.
estudiante_id	4 bytes	Integer	ID del estudiante matriculado.
asignatura_id	4 bytes	Integer	ID de la materia cursada.
periodo_id	4 bytes	Integer	ID del periodo académico vigente.
tutor_id	4 bytes	Integer	ID del tutor asignado a esta materia.
paralelo	5	Varchar	Identificador del paralelo (ej. 'A', 'B').

Relaciones: Estudiantes, Asignaturas, Periodos, Tutores

Campos Clave: PK: id FK: estudiante_id, asignatura_id

La Tabla 15 presentó la definición de la Tabla Notas, diseñada para almacenar el rendimiento académico. Estos datos sirvieron como base fundamental para alimentar el algoritmo de riesgo predictivo.

Tabla 15. Notas (Rendimiento)

Campo	Tamaño	Tipo de Dato	Descripción
id	4 bytes	Serial (Int)	Identificador único del registro de notas.
matricula_id	4 bytes	Integer	Referencia a la matrícula del estudiante.
parcial1	(4, 2)	Numeric	Calificación del primer parcial (0.00 - 10.00).
parcial2	(4, 2)	Numeric	Calificación del segundo parcial (0.00 - 10.00).
final	(4, 2)	Numeric	Calificación final o principal.
suspension	(4, 2)	Numeric	Nota de examen de suspensión (si aplica).
situacion	50	Varchar	Estado académico ('APROBADO', 'REPROBADO').

Relaciones: Matriculas (1:1)

Campos Clave: PK: id, FK: matricula_id

En la Tabla 16 se especificaron los atributos de la entidad Tutorías. Al ser la tabla principal del proceso, esta registró de manera integral el agendamiento y seguimiento de las actividades académicas.

Tabla 16. Tutorías (Sesiones)

Campo	Tamaño	Tipo de Dato	Descripción
id	4 bytes	Serial (Int)	Identificador único de la sesión de tutoría.
matricula_id	4 bytes	Integer	Referencia al contexto académico del estudiante.
tutor_id	4 bytes	Integer	Docente que imparte la sesión.
fecha	8 bytes	Timestamp	Fecha y hora programada para la sesión.
duracion_min	4 bytes	Integer	Duración de la sesión en minutos.
tema	Var	Text	Tema o dudas a tratar en la tutoría.
modalidad	20	Varchar	Modalidad de la sesión ('Presencial', 'Virtual').
estado	20	Varchar	Estado actual ('solicitada', 'programada', 'realizada').
enlace_reunion	255	Varchar	URL de Zoom para sesiones virtuales.

Relaciones: Matriculas, Tutores, Evaluaciones

Campos Clave: PK: id FK: matricula_id, tutor_id

Para finalizar, la Tabla 17 detalló el diccionario de la entidad Evaluaciones. Esta estructura almacenó la retroalimentación y el nivel de satisfacción expresado por el estudiante al concluir cada sesión.

Tabla 17. Evaluaciones (Calidad)

Campo	Tamaño	Tipo de Dato	Descripción
id	4 bytes	Serial (Int)	Identificador único de la evaluación.
tutoria_id	4 bytes	Integer	Referencia a la tutoría evaluada.
estrellas	4 bytes	Integer	Calificación numérica de satisfacción (1-5).
comentario_est	Var	Text	Opinión cualitativa del estudiante.
fecha	8 bytes	Timestamp	Fecha y hora del registro de la evaluación.

Relaciones: Tutorías (1:1)

Campos Clave: PK: id, FK: tutoria_id

Anexo 3: Fragmentos de Código Fuente Clave

Backend: gestión del servicio de tutorías

Fuente: backend/app/services/tutoria_service.py

```
class TutoriaService:
```

```
def create_tutoria(self, db: Session, tutoria: TutoriaCreate, estudiante_id: int,
    tema_predeterminado: Optional[str] = None):
```

```
    """
```

```
    Lógica para crear una tutoría. Diferencia si es solicitada por el
    estudiante o generada automáticamente por el modelo de riesgo.
```

```
    """
```

```
    try:
```

```
        if tema_predeterminado:
```

```
            # Lógica de generación automática por riesgo académico
```

```
            tutoria_data = {
```

```
                "matricula_id": tutoria.matricula_id,
```

```

    "tutor_id": tutoria.tutor_id,
    "fecha": datetime.now() + timedelta(days=2),
    "estado": "solicitada",
    # ... otros campos
}
else:
    # Lógica de solicitud manual del estudiante
    tutoria_data = tutoria.model_dump()

    nueva_tutoria = Tutoria(**tutoria_data)
    db.add(nueva_tutoria)
    db.commit()
    return nueva_tutoria
except Exception as e:
    db.rollback()
    raise HTTPException(status_code=500, detail=str(e))

```

Frontend: enrutamiento y carga perezosa (lazy loading)

// Fuente: frontend/src/App.tsx

```

const DashboardEstudiante = lazy(() => import('./pages/DashboardEstudiante'));
const DashboardTutor = lazy(() => import('./pages/DashboardTutor'));

function App() {
  return (
    <BrowserRouter>
      <AuthProvider>
        <Routes>
          {/* Rutas protegidas por autenticación */}
          <Route element={<PrivateRoutes />} />
          <Route path="/dashboard/estudiante" element={<DashboardEstudiante />} />
          <Route path="/dashboard/tutor" element={<DashboardTutor />} />
          <Route path="/tutorias/estudiante" element={<MisTutorias />} />
        </Route>
      </Routes>
    </AuthProvider>
  </BrowserRouter>
);
}

```