



UNIVERSIDAD LAICA “ELOY ALFARO” DE MANABÍ  
EXTENSIÓN EN EL CARMEN  
CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN  
Creada Ley No. 10 – Registro Oficial 313 de noviembre 13 de 1985

**PROYECTO INTEGRADOR**

**PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO(A)  
EN TECNOLOGÍAS DE LA INFORMACIÓN**

**SISTEMA INFORMÁTICO CON ÁRBOL DE DECISIÓN PARA  
RENDIMIENTO ACADÉMICO DE ESTUDIANTES DE LA CARRERA  
DE INGENIERÍA EN SOFTWARE DE ULEAM EL CARMEN**

MONTALVAN LOOR KAREN NATHALY

**AUTORA:**

ING. REASCOS PINCHAO RAÚL SAED, MG.

**TUTOR**


EL CARMEN, FEBRERO 2026



Uleam



# CERTIFICACIÓN DEL TUTOR

 Uleam ELOY ALFARO DE MANABÍ	NOMBRE DEL DOCUMENTO: CERTIFICADO DE TUTOR	CÓDIGO: PAT-04-F-004
	PROCEDIMIENTO: TITULACIÓN DE ESTUDIANTES DE GRADO BAJO LA UNIDAD DE INTEGRACIÓN CURRICULAR	REVISIÓN: 1 Página 1 de 1

## CERTIFICACIÓN

En calidad de docente tutor de la Extensión El Carmen de la Universidad Laica "Eloy Alfaro" de Manabí, CERTIFICO:

Haber dirigido, revisado y aprobado preliminarmente el Trabajo de Integración Curricular bajo la autoría de la estudiante MONTALVAN LOOR KAREN NATHALY, legalmente matriculada en la carrera de Ingeniería de Tecnologías de la Información, período académico 2025(1)-2025(2), cumpliendo el total de 384 horas, cuyo tema del proyecto es "SISTEMA INFORMÁTICO CON ÁRBOL DE DECISIÓN PARA RENDIMIENTO ACADÉMICO DE ESTUDIANTES DE LA CARRERA DE INGENIERÍA EN SOFTWARE DE ULEAM EL CARMEN".

La presente investigación ha sido desarrollada en apego al cumplimiento de los requisitos académicos exigidos por el Reglamento de Régimen Académico y en concordancia con los lineamientos internos de la opción de titulación en mención, reuniendo y cumpliendo con los méritos académicos, científicos y formales, y la originalidad del mismo, requisitos suficientes para ser sometida a la evaluación del tribunal de titulación que designe la autoridad competente.

Particular que certifico para los fines consiguientes, salvo disposición de Ley en contrario.

El Carmen, 02 de febrero de 2026.

Lo certifico,



Ing. Saed Reascos Pinchao, Mg.  
Docente Tutor  
Área: Tecnologías de la Información

# TRIBUNAL DE SUSTENTACIÓN



**Universidad Laica Eloy Alfaro de Manabí**  
**Extensión El Carmen**  
**Carrera de Ingeniería en Tecnologías de la Información**

## TRIBUNAL DE SUSTENTACIÓN

**Título del Trabajo de Titulación:**  
Sistema Informático con Árbol de decisión para Rendimiento Académico de estudiantes de la Carrera de Ingeniería en Software de Uleam El Carmen

**Modalidad:**  
Proyector Integrador

**Autor:**  
Montalvan Loor Karen Nathaly

**Tutor:**  
Ing. Reascos Pinchao Raúl Saed, Mg.

### Tribunal de Sustentación:

- **Presidente:** A.S. Minaya Macías Renelmo Wladimir, Mg



---

- **Miembro:** Ing. Pozo Hernández Clara Guadalupe, Mg



---

- **Miembro:** Ing. Mendoza Villamar Rocio Alexandra, Mg



---

**Fecha de Sustentación:**  
20 de febrero 2026

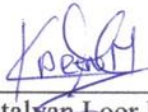
# DECLARACIÓN EXPRESA DE AUTORÍA

UNIVERSIDAD LAICA “ELOY ALFARO” DE MANABÍ  
EXTENSIÓN EN EL CARMEN



## DECLARACIÓN DE AUTORÍA

La responsabilidad del contenido de este Trabajo de titulación, cuyo tema es: SISTEMA INFORMÁTICO CON ÁRBOL DE DECISIÓN PARA RENDIMIENTO ACADÉMICO DE ESTUDIANTES DE LA CARRERA DE INGENIERÍA EN SOFTWARE DE ULEAM EXTENSIÓN EL CARMEN, corresponde exclusivamente a: MONTALVAN LOOR KAREN NATHALY con CI. 1311056970, y los derechos patrimoniales de la misma corresponden a la Universidad Laica Eloy Alfaro de Manabí.



Montalvan Loor Karen Nathaly

C.I. 1311056970

Dedicatoria

*A Dios, quien con su amor y promesas me fortaleció en cada momento. "Esfuézate y sé valiente; no temas ni desmayes, porque el Señor tu Dios estará contigo donde quiera que vayas." (Josué 1:9)*

*A mí mismo, por la perseverancia, la fuerza y la fe que me impulsaron a seguir adelante, aun cuando el camino parecía difícil.*

*A mi madre, por su amor incondicional, su apoyo incansable y por ser siempre una fuente de inspiración en mi vida.*

*A mi padre, por enseñarme que soy capaz de salir adelante por mí misma, y a mis hermanos, por su amor, su apoyo constante y la motivación que siempre me brindaron para no rendirme.*

*Karen*

## **AGRADECIMIENTO**

A la Universidad, por ser el espacio donde pude crecer académica y personalmente, y por darme la oportunidad de alcanzar uno de los mayores logros de mi vida: obtener un título universitario.

A la carrera de Tecnologías de la Información, gracias por cada reto, cada aprendizaje y cada lección que hoy llevo conmigo, no solo como profesional, sino también como persona.

A todos mis maestros, gracias por su paciencia, por compartir su conocimiento y por dejar una huella en mi camino. A cada uno de ustedes, les debo parte de este logro.

En especial al Ing. Saed, por su dedicación, guía y constante apoyo, que marcaron una diferencia significativa en mi formación.

La Autora

# ÍNDICE DE CONTENIDOS

Portada .....	I
Certificación del TUTOR.....	III
Tribunal de sustentación .....	III
Declaración expresa de autoría .....	V
Dedicatoria.....	VI
Agradecimiento.....	VII
Índice de contenidos.....	VIII
Índice tablas .....	XII
Índice gráficos e ilustraciones.....	XIII
Índice de anexos.....	XIV
Resumen.....	XV
Abstract.....	XVI
Capítulo I .....	1
1    Introducción .....	1
1.1    Introducción .....	1
1.2    Presentación del tema.....	2
1.3    Ubicación y contextualización de la problemática .....	3
1.4    Planteamiento del problema.....	3
1.4.1    Problematización.....	3
1.4.2    Génesis del problema.....	4
1.4.3    Estado actual del problema .....	4
1.5    Diagrama causa – efecto del problema .....	5
1.6    Objetivos.....	6
1.6.1    Objetivo general.....	6
1.6.2    Objetivos específicos .....	6
1.7    Justificación .....	6
1.8    Impactos esperados .....	7
1.8.1    Impacto tecnológico.....	7
1.8.2    Impacto social .....	8
1.8.3    Impacto ecológico.....	8
Capítulo II.....	9
2    Marco teórico de la investigación.....	9

2.1	Antecedentes históricos.....	9
2.2	Antecedentes de investigaciones relacionadas al tema presentado.....	9
2.3	Categoría conceptual.....	13
2.3.1	Sistema Informático con Árbol de Decisión.....	13
2.3.1.1	Sistemas Informáticos.....	13
2.3.1.2	Árbol de decisión.....	13
2.3.1.3	Componentes críticos.....	14
2.3.2	Arquitectura de Software para DSS.....	16
2.3.2.1	Modelos cliente-servidor vs. cloud.....	17
2.3.2.2	Microservicios para análisis académico.....	18
2.3.3	Teoría de Árboles de Decisión.....	19
2.3.3.1	Entropía y Ganancia de Información.....	19
2.3.3.2	Criterios de División Univariados.....	19
2.3.4	Algoritmos de Clasificación.....	20
2.3.5	Optimización con Random Forest.....	21
2.3.6	Rendimiento Académico.....	22
2.3.6.1	Teorías del Rendimiento.....	22
2.3.6.2	Indicadores Cuantitativos.....	24
2.3.6.3	Factores Cognitivos.....	26
2.3.6.4	Tecnología y Aprendizaje.....	29
2.3.6.5	Predictores Tempranos.....	30
2.3.6.6	Modelos de Riesgo Académico.....	31
2.4	Conclusiones relacionadas al marco teórico en referencia al tema planteado. ....	33
CAPÍTULO III.....		35
3	Marco investigativo (Diseño metodológico).....	35
3.1	Introducción.....	35
3.2	Tipo de investigación.....	36
3.2.1	Bibliográfica.....	36
3.2.2	De campo.....	36
3.2.3	Descriptivo.....	37
3.3	Métodos de investigación.....	37
3.3.1	Analítico – Sintético.....	37
3.3.2	Inductivo – Deductivo.....	38
3.4	Fuentes de información de datos.....	40

3.4.1	Fuentes primarias.....	40
3.4.2	Fuentes secundarias (profesores).....	40
3.5	Estrategia operacional para la recolección de datos.....	40
3.5.1	Población.....	40
3.5.1.1	Segmentación.....	40
3.5.1.2	Técnica de muestreo.....	41
3.5.2	Análisis de las herramientas de recolección de datos a utilizar.....	42
3.5.2.1	Encuesta dirigida a estudiantes.....	119
3.5.2.2	Encuesta dirigida a docentes.....	121
3.5.2.3	Entrevista.....	118
3.5.2.4	Estructura de los instrumentos de recolección de datos aplicados.....	43
3.5.3	Plan de recolección de datos.....	44
3.6	Análisis y presentación de resultados.....	44
3.6.1	Tabulación y análisis de los datos.....	44
3.6.2	Presentación y descripción de los resultados obtenidos.....	52
3.6.3	Informe final del análisis de los datos.....	52
Capítulo IV.....		55
4	Marco propositivo.....	55
4.1	Introducción.....	55
4.2	Descripción de la propuesta.....	55
4.3	Determinación de recursos.....	55
4.3.1	Recursos Humanos.....	55
4.3.2	Recursos Tecnológicos.....	56
4.3.3	Recursos Económicos.....	57
4.4	Desarrollo.....	57
4.4.1	Fase 1: Análisis de requisitos.....	58
4.4.1.1	Requerimientos funcionales.....	58
4.4.1.2	Requerimientos no funcionales.....	60
4.4.1.3	Requerimientos de hardware y software.....	60
4.4.1.4	Tipos de roles de usuario.....	62
4.4.1.5	Diagramas de caso de uso.....	62
	Diagrama de secuencia.....	65
4.4.1.6	Diagrama de estado.....	69
4.4.2	Fase 2: Diseño del sistema.....	70

4.4.2.1	Diseño de base de Datos .....	71
4.4.2.2	Diseño de interfaz .....	71
4.4.2.3	Pantallas .....	74
4.4.3	Fase 3: Implementación (Codificación).....	81
4.4.3.1	Herramientas de programación.....	81
4.4.3.2	Arquitectura .....	83
4.4.3.3	Seguridad .....	84
4.4.3.4	Clases y Métodos .....	84
4.4.3.5	Codificación.....	87
4.4.4	Fase 4: Pruebas.....	93
4.4.4.1	Pruebas de Datos en frio .....	93
4.4.4.2	Pruebas de Datos reales.....	97
4.4.5	Fase 5: Mantenimiento.....	99
4.4.5.1	manual de usuario .....	99
Capítulo V	.....	104
5	Evaluación de resultados.....	104
5.1	Introducción .....	104
5.2	Presentación y monitoreo de resultados.....	104
5.3	Interpretación objetiva .....	106
Capítulo VI	.....	107
6	Conclusiones y recomendaciones .....	107
6.1	Conclusiones.....	107
6.2	Recomendaciones.....	108
7	Bibliografía .....	109
Anexos	.....	117
Glosario	.....	125

## ÍNDICE TABLAS

Tabla 1 Algoritmos de clasificación.....	20
Tabla 2 Herramientas de recolección de datos.....	42
Tabla 3 Plan de recolección de datos .....	44
Tabla 4 Encuestas realizadas a estudiantes .....	44
Tabla 5 Encuestas realizadas a estudiantes .....	47
Tabla 6 Encuestas realizadas a estudiantes .....	50
Tabla 7 Recursos humanos.....	56
Tabla 8 Recursos tecnológicos.....	56
Tabla 9 Recursos económicos.....	57
Tabla 10 Tipos de roles de usuario.....	62
Tabla 11 Colores .....	71
Tabla 12 Lista de iconos.....	73
Tabla 13 Lenguajes para el desarrollo.....	81
Tabla 14 Librerías usadas.....	81
Tabla 15 Herramientas extras.....	83
Tabla 16 Métodos.....	85
Tabla 17 Pruebas de Datos en frio login .....	93
Tabla 18 Pruebas de Datos en frio Registro Docente.....	94
Tabla 19 Pruebas de Datos en frio Registro materia.....	94
Tabla 20 Pruebas de Datos en frio Registro Asignación.....	95
Tabla 21 Pruebas de Datos en frio Registro Masivo.....	96
Tabla 22 Pruebas de Datos en frio Predicción .....	96
Tabla 23 Pruebas de Datos Reales Login.....	97
Tabla 24 Pruebas de Datos Reales Registro Docente .....	97
Tabla 25 Pruebas de Datos Reales Registro Materia .....	98
Tabla 26 Pruebas de Datos Reales Asignación .....	98
Tabla 27 Pruebas de Datos Reales Carga Masiva .....	98
Tabla 28 Pruebas de Datos Reales Predicción .....	99
Tabla 29 Resultados generales de las predicciones académicas .....	104
Tabla 30 Monitoreo de predicciones por algoritmo.....	105
Tabla 31 Comparación del comportamiento de los algoritmos según las variables .....	105
Tabla 32 Resultados de la carga masiva de datos para predicción.....	105

Tabla 33 Resumen de validación del sistema de predicción.....	106
---	-----

## ÍNDICE DE FIGURAS

Figura 1 Diagrama de causa y efecto del problema.....	5
Figura 2 Modelo para el desarrollo.....	57
Figura 3 Caso de uso asignación.....	63
Figura 4 Caso de uso Predicción.....	63
Figura 5 Caso de uso carga masiva.....	64
Figura 6 Caso de uso Vista Notas .....	65
Figura 7 Secuencia Asignación .....	66
Figura 8 Secuencia predicción.....	67
Figura 9 Secuencia Carga Masiva.....	68
Figura 10 Estados de Asignación.....	69
Figura 11 Secuencia Predicción.....	69
Figura 12 Secuencia Carga Masiva.....	69
Figura 13 Base de Datos .....	70
Figura 14 Página de iconos.....	73
Figura 16 Pantallas Administrador.....	74
Figura 17 Pantallas profesor .....	75
Figura 18 Pantallas Coordinador.....	75
Figura 19 Pantallas estudiantes .....	76
Figura 20 Login.....	76
Figura 21 Menú.....	77
Figura 22 Asignación .....	77
Figura 23 Predicción.....	78
Figura 24 Incumplimientos.....	78
Figura 25 Seguimiento.....	79
Figura 26 Carga Masiva.....	79
Figura 27 Promedios.....	80
Figura 28 Predicción Estudiante .....	80
Figura 29 Patrón MVC. Extraída de: (Hernández U. , 2015) .....	83
Figura 30 Código Login 1.....	87

Figura 31 Código Login 2.....	88
Figura 32 Código Python Predicción 1 .....	89
Figura 33 Código Python Predicción 2.....	89
Figura 34 Código Python Predicción 3.....	90
Figura 35 Código PHP predicción 1 .....	90
Figura 36 Código PHP predicción 2 .....	91
Figura 37 Código PHP predicción 3 .....	91
Figura 38 Código Carga Masiva 1 .....	92
Figura 39 Código Carga Masiva 2 .....	92
Figura 40 Código Carga Masiva 3 .....	93
Figura 41 Flash y Pythonanyware.....	99
Figura 42 Crear cuenta Pythonanyware.....	100
Figura 43 Dashboard Pythonanyware.....	100
Figura 44 Menú Pythonanyware.....	101
Figura 45 Mensaje de Dominios.....	101
Figura 46 Crear Proyecto Web Flask .....	101
Figura 47 Gestión de archivos Pythonanyware .....	102
Figura 48 Code->WSGI.....	102
Figura 49 Configuración WSGI.....	102
Figura 50 Web Creada Pythonanyware y Flask .....	103

## ÍNDICE DE ANEXOS

Anexo A. Aprobación de tema .....	117
Anexo B. Instrumento entrevista.....	118
Anexo C. Instrumentos de encuesta .....	119
Anexo D. Fotografías .....	123
Anexo E. Certificado de coincidencia académica.....	124

## **RESUMEN**

La presente investigación tuvo como objetivo desarrollar un sistema informático basado en técnicas de minería de datos utilizando algoritmos de árboles de decisión para evaluar y mejorar el rendimiento académico de los estudiantes de la carrera de Ingeniería en Software de la Universidad Laica Eloy Alfaro de Manabí (ULEAM) extensión El Carmen. El estudio se fundamentó en un enfoque metodológico cuantitativo apoyado en la recolección y análisis de datos académicos mediante encuestas estructuradas y registros institucionales los cuales permitieron identificar patrones relevantes relacionados con el desempeño estudiantil. Para el desarrollo del sistema, se emplearon técnicas de análisis y diseño de software incorporando un modelo en cascada y la implementación de algoritmos C4.5 y Random Forest para la generación de predicciones de aprobación académica considerando variables como el promedio de calificaciones y el porcentaje de asistencia. Asimismo, el sistema integró procesos de carga masiva de datos y módulos de predicción desarrollados en Python y PHP, garantizando su funcionalidad y escalabilidad. Los resultados obtenidos evidenciaron que el sistema informático permite identificar de manera temprana a estudiantes con riesgo de bajo rendimiento académico, proporcionando información útil para el seguimiento y la toma de decisiones por parte de docentes y autoridades académicas. Las pruebas funcionales y la validación con usuarios demostraron un adecuado nivel de precisión y aceptación del sistema destacándose el algoritmo Random Forest por su mayor estabilidad y desempeño predictivo.

## **ABSTRACT**

This research aimed to develop a computer system based on data mining techniques using decision tree algorithms to evaluate and improve the academic performance of Software Engineering students at the Eloy Alfaro Lay University of Manabí (ULEAM), El Carmen campus. The study was based on a quantitative methodological approach supported by the collection and analysis of academic data through structured surveys and institutional records, which allowed for the identification of relevant patterns related to student performance. For the system's development, software analysis and design techniques were employed, incorporating a waterfall model and the implementation of C4.5 and Random Forest algorithms to generate academic success predictions considering variables such as grade point average and attendance percentage. The system also integrated bulk data loading processes and prediction modules developed in Python and PHP, ensuring its functionality and scalability. The results obtained demonstrated that the computer system allows for the early identification of students at risk of low academic performance, providing useful information for monitoring and decision-making by faculty and academic authorities. Functional testing and user validation demonstrated an adequate level of accuracy and system acceptance, with the Random Forest algorithm standing out for its greater stability and predictive performance.

# CAPÍTULO I

## 1 INTRODUCCIÓN

### 1.1 Introducción

En la actualidad, muchos estudiantes universitarios enfrentan dificultades que afectan su rendimiento académico. En especial, en la carrera de Ingeniería en Software de la ULEAM, extensión El Carmen, se ha identificado que factores como la desmotivación, la falta de seguimiento oportuno y las exigencias laborales físicas, que en su mayoría están relacionadas con situaciones socioeconómicas, influyen negativamente en el desempeño de los estudiantes.

Por lo antes planteado resulta necesario implementar estrategias académicas que solucionen esta problemática con la finalidad de detectar a tiempo casos de rendimiento académico bajo y así poder brindar apoyo adecuado. De ahí que se propone el desarrollo de un sistema informático basado en árboles de decisión que ayude a solucionar la problemática planteada mediante el análisis de notas y los niveles de asistencia. Para ello resulta necesario utilizar técnicas de minería de datos. Ya que estas herramientas facilitan identificar a los estudiantes que se encuentran en riesgo y permite actuar a tiempo para evitar pérdidas y rezago estudiantil.

Para el desarrollo de la investigación se consideran los siguientes capítulos:

- Capítulo I: En este capítulo se presenta una introducción a la problemática de estudio. Luego se detalla el tema de investigación, se da un contexto sobre los aspectos más importantes para la elección del tema. También se presenta la problemática observada, los objetivos, la justificación y los impactos esperados.
- El Capítulo II: En este apartado se detalla el marco teórico que sirvió como sustento y base para la investigación. En este se desarrollaron aspectos sobre los antecedentes históricos, antecedentes de investigaciones relacionadas para conocer los aspectos importantes de los árboles de decisión y el desempeño académico. En la categoría conceptual se desarrollan los aspectos que ayuden a tener una fundamentación teórica sólida para el correcto desarrollo del sistema. Entre los principales temas desarrollados se tiene los sistemas informáticos con árboles de decisión, la arquitectura de software para DSS, la teoría de árboles de

decisión, los algoritmos de clasificación, la optimización con Random Forest. Y todos los temas relacionados con el rendimiento académico.

- El Capítulo III: en este capítulo se desarrolla la metodología utilizada, se detallan aspectos como el tipo de investigación, métodos de investigación, fuentes de información de datos, la estrategia operacional para la recolección de datos y finalmente el análisis y presentación de resultados.
- El Capítulo IV: En este capítulo se desarrolla el marco propositivo de la investigación para lo cual se elabora la propuesta en donde se detalla los requisitos, la descripción de la propuesta, la determinación de recursos, las etapas de acción para el desarrollo del sistema.
- El Capítulo V: Una vez presentados los resultados se presenta la evaluación de los resultados, aquí también se realiza un monitoreo y una interpretación objetiva.
- El Capítulo VI: Finalmente en este capítulo se presentan las conclusiones y recomendaciones de acuerdo a los objetivos planteados, para cumplir con el alcance de la investigación.

## **1.2 Presentación del tema**

El presente trabajo surge de la necesidad de mejorar los medios de monitoreo y evaluación del rendimiento académico de los estudiantes de la carrera de Ingeniería en Software de la ULEAM, extensión El Carmen. Ya que se ha observado una escasez de herramientas tecnológicas que faciliten la detección oportuna de estudiantes con bajo desempeño académico. Esto impide la implementación de medidas adecuadas para prevenir consecuencias como la deserción y la desmotivación estudiantil.

De ahí que para solucionar esta problemática se eligió este tema ya que responde al interés investigativo y brinda una solución mediante el uso de técnicas de minería de datos. Debido a la estructura del sistema se consideró que el uso de árboles de decisión es la mejor opción. Debido a que mediante esta se procesa la información mediante el reconocimiento de patrones que permitan identificar alertas de bajo rendimiento académico para encontrar soluciones oportunas. Este sistema informático permite automatizar estos procesos para realizar un análisis académico adecuado, que permita fortalecer la toma de decisiones institucionales y apoyar el acompañamiento individualizado de los estudiantes.

### **1.3 Ubicación y contextualización de la problemática**

La Universidad Laica Eloy Alfaro de Manabí (ULEAM), extensión El Carmen, fue creada en el año 1985 por el Congreso de esa época. Su matriz se encuentra en la ciudad de Manta y, dos años después de su fundación, se aprobó la creación de una extensión en el cantón más distante de la sede principal, ofreciendo inicialmente carreras vinculadas a las áreas de Educación, Agropecuaria y Contabilidad. En la universidad se han incorporado nuevas carreras con la finalidad de satisfacer la demanda educativa. Esto ha ayudado a contribuir al desarrollo académico y profesional de Manabí.

Para el desarrollo del proyecto se considera la carrera de Ingeniería en Software, la cual es una de las carreras más recientes que se ha incorporado a la universidad y actualmente cuenta con aproximadamente 159 estudiantes. Las actividades académicas se desarrollan en los diferentes laboratorios y aulas de la Uleam El Carmen.

Este contexto permitió analizar de cerca las problemáticas que enfrentan los estudiantes con bajo rendimiento académico, lo que dio paso al diseño de estrategias tecnológicas que fomenten un mejor desempeño estudiantil, garanticen la continuidad en los estudios y contribuyan a mejorar la experiencia académica.

### **1.4 Planteamiento del problema**

#### **1.4.1 Problematización**

Una vez conocido que los estudiantes de la carrera de Ingeniería en Software de la ULEAM, extensión El Carmen, presentan diferentes condiciones socioeconómicas, el autor del presente trabajo se plantea una serie de preguntas con el objetivo de identificar los posibles problemas que enfrentan los alumnos, así como las causas que podrían originarlos. Para ello, se formuló el siguiente cuestionario:

- ¿Cómo identificar el bajo rendimiento académico en los estudiantes?
- ¿Cuántos estudiantes se han retirado por bajo rendimiento académico?
- ¿Por qué los estudiantes abandonan sus estudios en los primeros niveles?
- ¿Cuántos estudiantes se ven afectados por la falta de notificación oportuna de sus calificaciones?
- ¿Qué consecuencias tiene el trabajo físico exigente en el desempeño académico?
- ¿Por qué no se detecta a tiempo a los estudiantes con problemas académicos?

- ¿Cuántos estudiantes están en riesgo cada semestre por falta de intervención temprana?

### **1.4.2 Génesis del problema**

El rendimiento académico de los estudiantes universitarios ha sido siempre un tema de interés para las instituciones de educación superior. En el caso específico de la Universidad Laica Eloy Alfaro de Manabí (ULEAM), extensión El Carmen, se ha evidenciado un alto porcentaje de estudiantes de la carrera de Ingeniería en Software que presentan dificultades en su desempeño académico. Lo que ha causado la pérdida de materias y en muchos casos la deserción estudiantil.

Entre los principales factores que afectan al rendimiento académico se tiene la situación económica, la distribución de los estudios junto a trabajos que les permita cubrir los gastos universitarios. Gran parte de estudiantes reparten su tiempo entre el trabajo y sus estudios ya que son jornadas agotadoras haciendo que el estudio quede de lado. Lo que repercute negativamente en su rendimiento académico. A la situación planteada se le agrega la escasa notificación de calificaciones, así como la falta de herramientas tecnológicas que permitan identificar de manera oportuna a los estudiantes en situación de riesgo.

Esto ha provocado estudiantes desmotivados ya que su rendimiento se ha visto afectado lo que en muchas ocasiones a concluido en deserción estudiantil. En conjunto estos factores han causado niveles bajos de aprendizaje y bajos indicadores de calidad de la carrera y de la universidad.

### **1.4.3 Estado actual del problema**

Actualmente, muchos estudiantes enfrentan dificultades para hacer un seguimiento adecuado de sus calificaciones. A lo largo del semestre, no siempre tienen claridad sobre cómo van en cada materia, ya sea porque no llevan un control personal, o porque no reciben información actualizada de parte de sus docentes. Esto ha ocasionado que los estudiantes se confíen y no se den cuenta del estado de su situación académica. Conllevando muchas veces a la pérdida de las materias y retraso en sus estudios.

Al ofertar la universidad facilidades en sus estudios muchos de los estudiantes también distribuyen su tiempo con su trabajo y obligaciones en sus hogares. Lo que afecta directamente a su desempeño académico, además que repercute en su estado físico y emocional. A esta

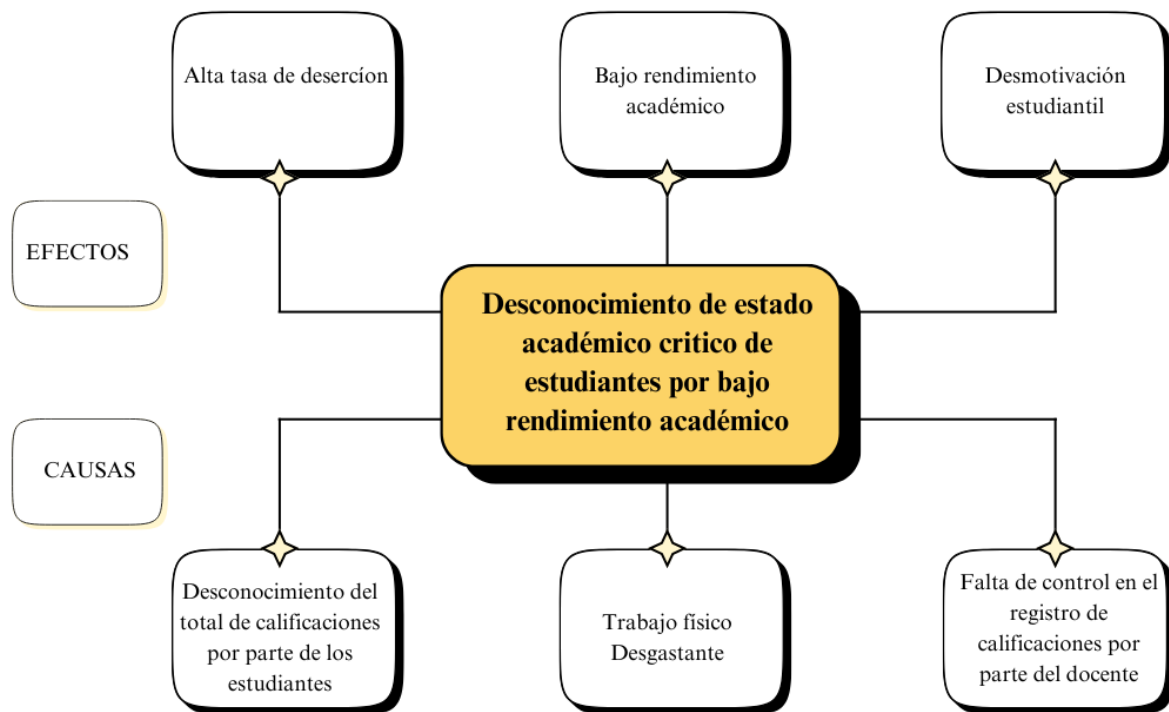
problemática se le suma la falta de seguimiento y acompañamiento docente, además de la falta de alertas y notificaciones ante el culmino de tiempo de tareas.

De ahí que es evidente que los estudiantes que viven en estas situaciones presentan problemas en su rendimiento académico. Muchas veces reprueban materias consecutivas lo que les lleva a una desmotivación frecuente y en algunas ocasiones a una deserción estudiantil. Es por ello que la Universidad ULEAM se ha visto preocupada ante esta problemática ya que limita las oportunidades de mejorar el proceso educativo y apoyar a quienes más lo necesitan. Por ello es importante buscar soluciones prácticas que permitan tener un mejor control del rendimiento académico y tomar decisiones antes de que sea demasiado tarde

### 1.5 Diagrama causa – efecto del problema

**Figura 1**

*Diagrama de causa y efecto del problema*



*Nota.* En la figura se detalla el diagrama de causa efecto en donde se pone como punto central el desconocimiento de estado académico crítico de estudiantes por bajo rendimiento académico. Elaborado por la autora, 2025.

## **1.6 Objetivos**

### **1.6.1 Objetivo general**

Desarrollar un sistema informático basado en técnicas de minería de datos, mediante árboles de decisión, que permita evaluar y mejorar el rendimiento académico de los estudiantes de la carrera de Ingeniería en Software de la ULEAM, extensión El Carmen.

### **1.6.2 Objetivos específicos**

- Definir los objetivos del proyecto, plantear la problemática que sea el punto de partida al desarrollo del proyecto e dictaminar los impactos esperados del mismo.
- Fundamentar bibliográficamente con la norma APA7 conceptos de sistemas informáticos como también de rendimiento académico para establecer los fundamentos teóricos del estudio.
- Diseñar un enfoque metodológico fundamentado en el paradigma cuantitativo, empleando instrumentos como encuestas estructuradas y técnicas estadísticas descriptivas e inferenciales, que permitan recopilar, analizar e interpretar datos precisos y relevantes sobre el rendimiento académico de los estudiantes, con el fin de sustentar técnicamente la viabilidad del desarrollo del sistema informático propuesto.
- Proponer un sistema informático con árboles de decisión mediante técnicas de análisis y diseño de software para dar una solución tecnológica que pueda detectar estudiantes con bajo rendimiento académico en la carrera de Ingeniería en Software de la ULEAM.
- Analizar los resultados obtenidos de la implementación del sistema propuesto mediante pruebas funcionales y validación con usuarios para verificar su eficacia en el desarrollo del seguimiento académico.

## **1.7 Justificación**

Este proyecto nace del interés por contribuir al mejoramiento del desempeño académico de los estudiantes. A lo largo de los semestres, se ha observado la importancia de que los estudiantes mantengan una visión clara y continua sobre su progreso académico. Contar con información precisa y a tiempo favorece la toma de decisiones, fortalece la motivación personal y promueve la responsabilidad en el proceso de aprendizaje. Es importante tener en cuenta que

identificar a tiempo las necesidades académicas de los estudiantes de la carrera de Ingeniería en Software de la ULEAM ayuda a mejorar los procesos académicos, así como impulsar una cultura de superación y compromiso con la formación profesional de esta institución.

Por consiguiente, esta investigación busca ayudar a los estudiantes que se encuentran en situaciones vulnerables para que así no deserten de sus estudios o no se retrasen en cada semestre ante la pérdida de materias. Ya que al ver alertas en su rendimiento estos pueden concienciar acerca del riesgo en el cual se encuentran si no toman medidas correctivas y así pueden esforzarse y dedicar mayor tiempo a las tareas y actividades de cada materia. Esto no solo mejora su rendimiento, sino que también puede motivarlos a seguir adelante y terminar su carrera.

Además, este proyecto también pretende reforzar la comunicación entre docentes y estudiantes, permitiendo que haya más interés en apoyar a quienes lo necesiten. De esta manera, no solo se busca mejorar calificaciones, sino también fortalecer el compromiso con el proceso educativo y con el futuro profesional de cada estudiante. Por ello, este proyecto representa una oportunidad significativa tanto para el desarrollo estudiantil como para el fortalecimiento institucional.

## **1.8 Impactos esperados**

### **1.8.1 Impacto tecnológico**

El impacto tecnológico de este proyecto se basa en la posibilidad de fomentar una cultura institucional más alineada con el uso estratégico de tecnologías de la información en el ámbito educativo. En la actualidad, las tecnologías digitales representan un recurso clave para mejorar la gestión académica, el acceso a la información y la toma de decisiones fundamentadas dentro de las instituciones de educación superior.

Ya que mediante el uso de estas herramientas se puede integrar herramientas tecnológicas que mejoren el proceso enseñanza aprendizaje. Y a través de ello fortalece el desarrollo de competencias digitales en la comunidad universitaria, incentivando el uso responsable y provechoso de la tecnología en el entorno académico. Es así que el impacto tecnológico resulta significativo ya que genera un ambiente propicio para la innovación. Esto contribuye al fortalecimiento institucional y a la mejora continua de los procesos de enseñanza y aprendizaje mediante el aprovechamiento de los recursos tecnológicos disponibles.

### **1.8.2 Impacto social**

El impacto social de este proyecto se manifiesta en la contribución al fortalecimiento del acompañamiento académico, promoviendo una relación más cercana y comprometida entre la institución, los estudiantes y sus familias. Al generar conciencia sobre la importancia del seguimiento académico, se favorece un entorno educativo que valora el desarrollo integral del estudiante, reforzando su autoestima, confianza y motivación para continuar su formación profesional.

Este enfoque también aporta a la construcción de una comunidad universitaria más solidaria, donde se reconoce la importancia de atender oportunamente los factores que inciden en el rendimiento académico, promoviendo así la permanencia estudiantil y el bienestar colectivo.

### **1.8.3 Impacto ecológico**

Este proyecto también genera un impacto positivo en el cuidado del medio ambiente, ya que al mejorar el rendimiento académico y evitar que los estudiantes repitan materias o semestres, se reduce la necesidad de movilizarse constantemente a clases adicionales. Esto contribuye indirectamente a una menor contaminación ambiental asociada al uso del transporte y, por ende, a la disminución en la emisión de gases contaminantes.

De esta manera, el proyecto no solo promueve el fortalecimiento del proceso educativo, sino que también aporta a la sostenibilidad, fomentando un uso más consciente y responsable de los recursos, y promoviendo prácticas que respetan y cuidan el entorno natural.

## CAPÍTULO II

### 2 MARCO TEÓRICO DE LA INVESTIGACIÓN

#### 2.1 Antecedentes históricos

##### 2.1.1 1. Evolución de los Sistemas Computacionales y la Lógica de Programación

###### **El origen de los sistemas informáticos se remonta a 1822**

Con la máquina diferencial de Babbage, la cual utilizaba tarjetas perforadas para automatizar cálculos complejos Posteriormente, entre 1935 y 1945, Alan Turing estableció las bases de la computación moderna al proponer un modelo teórico capaz de resolver problemas mediante algoritmos, mientras que John Von Neumann introdujo el concepto de programa almacenado en memoria, facilitando la reutilización del hardware. Estos avances, sumados al desarrollo de la lógica simbólica de George Boole, permitieron la creación de las estructuras de decisión que hoy son esenciales en la programación actual. (Raya & Raya, 2015)

##### 2.1.2 Consolidación de la Minería de Datos y el Proceso KDD

La minería de datos emergió a finales de los años 80 y se consolidó en la década de 1990 como una disciplina crítica para extraer información oculta en grandes volúmenes de datos. Fayyad et al. (1996) la definieron como un subproceso dentro del Descubrimiento de Conocimiento en Bases de Datos (KDD), orientado a identificar patrones válidos, novedosos y útiles. El aumento de la capacidad de procesamiento y almacenamiento en esta época permitió que instituciones y empresas pasaran de simples consultas de datos a análisis complejos de patrones de comportamiento y fidelización, utilizando software especializado como Weka o SPSS (Sangüesa, 2016).

##### 2.1.3 Desarrollo de los Árboles de Decisión y Algoritmos Evolutivos

Dentro del aprendizaje automático, los árboles de decisión se posicionaron como los algoritmos precursores, destacando los modelos ID3 y C4.5 propuestos por Quinlan en 1986 y 1993, respectivamente. A medida que la disciplina evolucionó, surgieron enfoques más complejos como los árboles oblicuos y multivariados para superar limitaciones de precisión (Breiman et al., 1984; Llorà & Garrell, 2000). Finalmente, la introducción de Algoritmos Evolutivos (AE) como GALE permitió una inducción de árboles más robusta y versátil,

facilitando su aplicación en diversos dominios, incluyendo el análisis del rendimiento académico en la educación superior (Fogel, 1992).

#### **2.1.4 Evolución de la Minería de Datos Educativos (EDM) y su Impacto Institucional**

A partir de la década de 2010, la minería de datos se especializó en el ámbito académico, dando lugar a la Minería de Datos Educativos (EDM por sus siglas en inglés). Esta disciplina se centra en desarrollar métodos para explorar datos provenientes de entornos educativos y utilizarlos para comprender mejor a los estudiantes (González et al., 2003).

En la actualidad, el uso de sistemas informáticos con alertas tempranas y algoritmos de clasificación ha permitido que las instituciones de educación superior identifiquen patrones de riesgo de deserción o bajo rendimiento antes de que ocurran. Esto ha transformado la gestión académica, pasando de un modelo reactivo a uno preventivo, donde la toma de decisiones se fundamenta en la evidencia extraída directamente de los registros históricos de los estudiantes (Sangüesa, 2016).

#### **2.1.5 Evolución del Software de Aplicación y la Gestión de Datos Masivos**

A partir de la década de 2010, el desarrollo de sistemas informáticos experimentó un cambio hacia la arquitectura responsiva y el intercambio de datos mediante APIs, facilitando que el software se adapte a diversos dispositivos y plataformas (Coenen, 2011). En el contexto académico, esto permitió la transición de simples bases de datos a sistemas de gestión de aprendizaje que soportan la "Carga Masiva de Datos".

Según Sangüesa (2016), esta evolución técnica ha sido crucial para procesar grandes volúmenes de calificaciones y registros de asistencia en tiempo real. Actualmente, la integración de estas capacidades tecnológicas con algoritmos de clasificación permite que las instituciones educativas no solo almacenen información, sino que generen conocimiento predictivo para mejorar la toma de decisiones administrativa y pedagógica.

## 2.2 Antecedentes de investigaciones relacionadas al tema presentado

Como antecedentes de investigaciones previas se presenta el trabajo de investigación realizado por Puga (2023) titulado “Mapeo sistemático sobre el seguimiento del aprendizaje de estudiantes mediante el uso de la minería de datos educativos”. En este trabajo se utiliza los árboles de decisión para evaluar el rendimiento académico la recolección de datos se la realiza mediante el método de bola de nieve con la finalidad de ampliar la búsqueda inicial e incluir estudios adicionales, los algoritmos utilizados son ID3, C4.5 y CART con la finalidad de dividir los datos de manera óptima.

En el artículo realizado Díaz-Landa et al. (2021) cuyo tema fue el “Rendimiento Académico de Estudiantes en Educación Superior: Predicciones de Factores Influyentes a partir de Árboles de Decisión”. Se elaboró una predicción del rendimiento académico de estudiantes de maestrías en educación. Para ello se empleó la técnica de árbol de decisión y minería de datos y herramientas para construir un modelo con el algoritmo J48 del software WEKA, teniendo en cuenta factores educacionales, familiares, socioeconómicos, de hábitos y costumbres. Los resultados dan cuenta de una metodología capaz de entrenar un sistema para clasificar a un estudiante, a partir de una de las categorías del rendimiento académico.

Otro de los trabajos considerados fue el realizado por Hernández (2024) que se centra en el tema “Modelo de dominio específico para análisis y minería de datos educativos”. En este se propone el uso de técnicas de análisis de datos para apoyar los procesos educativos, al igual que en otros dominios, busca optimizar la toma de decisiones y la planificación. Las tecnologías de la información y la comunicación contribuyen a estos procesos de análisis. En este trabajo se encontraron dificultades en el procesamiento de los datos. También se pudo observar que los modelos genéricos presentan poco ajuste en cada dominio de datos.

El siguiente proyecto de titulación fue desarrollado por Rico y Gaytán (2022). cuyo título fue “Modelos predictivos del rendimiento académico a partir de características de estudiantes de ingeniería”. El objetivo de esta investigación fue “proponer una metodología para construir modelos predictivos del rendimiento académico mediante características de estudiantes de ingeniería del país y comparar los modelos utilizando diferentes métricas de evaluación”. Los datos fueron recolectados a través de tres técnicas de aprendizaje automático. Los modelos considerados fueron “Naïve Bayes, k vecinos más cercanos y árbol de decisión C4.5”. El árbol de decisión C4.5 se utilizó con validación cruzada “con 10 particiones en el

entrenamiento y con nueve particiones se construye el modelo predictivo para predecir la aprobación de la partición restante para calcular la exactitud” (Rico y Gaytán, 2022).

El estudio elaborado por Morales (2024) en el marco de la Maestría en Inteligencia de Negocios y Ciencia de Datos, desarrolló el proyecto titulado “Deserción escolar en Ecuador 2022: Modelo basado en árboles de decisión”. En este trabajo se desarrolló un modelo basado en árboles de decisión orientado a la identificación temprana de estudiantes en riesgo de deserción escolar en Ecuador. Esta propuesta permitió aplicar estrategias de intervención oportunas que contribuyeron significativamente a la mejora de los índices de deserción académica.

Una vez analizados todos los trabajos antes mencionados se puede concluir que el uso de técnica de minería de datos para mejorar el rendimiento académico es cada vez más usado. Ya que mediante estas se puede prevenir la deserción y fomentar la permanencia estudiantil mediante estrategias como alertas tempranas, informes de seguimiento, entrega anticipada de calificaciones. Así como el monitoreo de la satisfacción estudiantil. Sin embargo, en cada investigación se consideran diferentes algoritmos, población y herramientas. Tal y como se puede ver en los estudios en los que se utilizaron algoritmos específicos como J48 en el software WEKA (Díaz-Landa et al., 2021).

Otros estudios optaron por enfoques más amplios de análisis de datos educativos (Hernández, 2024) o por la implementación de modelos predictivos basados en características contextuales y académicas (Rico y Gaytán, 2022). Las principales diferencias entre el trabajo propuesto y las investigaciones presentadas se dan en que este no será un mapeo ni revisión sistemática, sino una implementación práctica de un sistema informático basado en árbol de decisión.

Además, este estudio se centrará en crear un sistema informático aplicable en tiempo real y con usuarios. Con respecto a la investigación realizada por Rico y Gaytán se diferencia en que en su investigación los autores comparan técnicas, mientras que en este trabajo se propone implementar un sistema basado únicamente en árboles de decisión orientado a la toma de decisiones académicas y no solo a evaluación de modelos como ellos lo hicieron.

Asimismo, se observan diferencias en los niveles de estudio abordados ya que las investigaciones varían desde estudiantes de maestría hasta estudiantes universitarios de pregrado. Por otro lado, también existe diferencia en los entornos geográficos, lo que aporta una visión más diversa sobre la aplicabilidad de estas tecnologías en distintos contextos

educativos. Ya que esta investigación será aplicada a los estudiantes de la carrera de Ingeniería en Software de ULEAM El Carmen. De ahí que, en conjunto, estas investigaciones evidencian el potencial de la analítica educativa para apoyar la toma de decisiones en instituciones de educación superior y hace novedoso y viable la propuesta realizada.

## **2.3 Categoría conceptual**

### **2.3.1 Sistema Informático con Árbol de Decisión**

#### **2.3.1.1 *Sistemas Informáticos***

Se considera que los sistemas informáticos son conjuntos organizados de componentes interrelacionados que permiten el procesamiento, almacenamiento, transmisión y gestión de información. Están compuestos por hardware, software, datos, usuarios y procedimientos, funcionando de manera coordinada para automatizar tareas, apoyar la toma de decisiones y mejorar la eficiencia de procesos en distintos contextos, incluido el ámbito educativo (Laudon y Laudon, 2020).

Por otro lado, el término sistema de información (SI) se refiere a un sistema de personas, registros de datos y actividades que procesan los datos y la información en una organización, e incluye los procesos manuales y automatizados de la organización. Por lo tanto, un SI incluye las Tecnologías de la Información (Paul, 2010).

En sentido estricto, el término sistemas informáticos se refieren al software de aplicación específico que se utiliza para almacenar registros de datos en un sistema informático y automatiza algunas de las actividades de procesamiento de información de la organización. Los sistemas de información informáticos pertenecen al campo de las tecnologías de la información.

#### **2.3.1.2 *Árbol de decisión***

Los árboles de decisión, como concepto en el aprendizaje automático (ML), tienen una historia que se remonta a mediados del siglo XX. Los primeros estudios sobre árboles de decisión fueron iniciados por Charles J. Clopper y Egon S. Pearson en 1934, quienes introdujeron el concepto de procesos de decisión binarios (Gupta y Ramdas, 2021).

Sin embargo, la implementación moderna de los árboles de decisión en el contexto del aprendizaje automático comenzó décadas después (Mazurek et al., 2023). Por su lado, Breiman (1996) desarrolló el algoritmo CART en 1984 e introdujo conceptos como el índice de Gini y la división binaria que ahora están muy extendidos en el diseño de árboles de decisión.

Por otro lado, Quinlan (1986) fue el primero en desarrollar el algoritmo ID3 y este hasta la actualidad es uno de los más notables y utilizados. Además, el mismo autor Quinlan (1993) mejoró ID3 mediante la introducción del árbol de decisión C4.5. Estos algoritmos en conjunto con los bosques aleatorios y los algoritmos de boosting son considerados como algoritmos fundamentales en el aprendizaje automático (Amat, 2020).

El proceso de aprendizaje de los árboles de decisión “implica una serie de pasos en los que los datos se dividen en subconjuntos homogéneos” (Breiman et al., 1984). En donde el nodo raíz que es el punto de partida del árbol representa todo el conjunto de datos. El algoritmo identifica la característica y el umbral que conduce a la mejor división según un criterio específico (Mienye et al., 2019).

El proceso continúa recursivamente y cada subconjunto de datos se divide en cada nodo hijo. Esto continúa hasta que se alcanza un criterio de parada, generalmente cuando los nodos son puros o cuando se alcanza una profundidad predefinida del árbol. Los nodos donde termina el árbol, llamados nodo hoja o nodo terminal, representan los resultados o las etiquetas de clase. La decisión de dividir en cada nodo se toma mediante formulaciones matemáticas como la ganancia de información, la impureza de Gini o la reducción de la varianza (Barceló, 2008).

De ahí que se toma en cuenta varios factores para el éxito de las técnicas de árboles de decisión. Estos factores incluyen la calidad de los datos, la profundidad del árbol, los criterios de división y el método de poda del árbol. La efectividad de los árboles de decisión depende en gran medida de la calidad de los datos de entrenamiento. Por lo tanto, es necesario utilizar datos limpios o preprocesados que no contengan valores faltantes ni valores atípicos, lo que puede mejorar significativamente el rendimiento de los modelos resultantes. Además, la selección y la ingeniería de características son necesarias porque la introducción de características relevantes y bien transformadas puede generar divisiones más eficientes y precisas (Piramuthu, 2008).

### ***2.3.1.3 Componentes críticos***

Los árboles de decisión (DT) por su nombre en inglés, representan modelos predictivos bien conocidos, propuestos originalmente por Breiman (1996) hace más de 20 años. Los DT son estructuras de árbol con raíz, donde las hojas representan clasificaciones y los nodos, pruebas de características que conducen a dichas clasificaciones. Normalmente, los DT se implementan en software (Domor y Jere, 2016).

La implementación de hardware de los DT no se ha investigado a fondo hasta la fecha. Solo existen dos artículos disponibles en la literatura (Bermak y Martínez, 2003). El artículo propone la implementación de hardware de DT oblicuos utilizando la arquitectura de matriz sistólica bidimensional, mientras que el artículo (López-Estrada y Cumplido, 2006) presenta la implementación de hardware de DT de ejes paralelos para un problema específico (clasificación del estado del mar de texturas).

López-Estrada y Martínez (2003) propone varias arquitecturas digitales adecuadas para la implementación de hardware de DT arbitrarios. Estas arquitecturas se pueden personalizar fácilmente para adaptarse a una amplia variedad de requisitos de aplicación, lo que las convierte en excelentes candidatas para los componentes básicos de los diseños de Sistemas en Chip (SIM) de sistemas embebidos.

### ***2.3.1.3.1 Implementación de hardware de árboles de decisión***

Cuando se habla de aprendizaje de DT es importante considerar que la función objetivo a aprender se representa mediante un árbol de decisión de profundidad finita. En donde cada nodo del árbol involucra a uno o más atributos de la función objetivo y cada rama que desciende de un nodo coincide con uno de los posibles resultados de la prueba en el nodo considerado. Las pruebas asociadas son muy usadas para clasificar una instancia. Para clasificar una instancia para ello se comienza con el nodo raíz y se termina con un nodo hoja. Si solo se permiten atributos numéricos, el DT resultante será binario. La mayoría de los algoritmos de árboles de decisión solo permiten atributos numéricos (Struharik, 2011).

Según el conocimiento de Breiman (1996) y (Domor y Jere, 2016), existen pocos ejemplos disponibles en la literatura sobre la implementación de DT en hardware. Un enfoque sencillo para la implementación de DT en hardware sería implementar cada nodo de DT como un módulo independiente, como se propone en (Domor y Jere, 2016). Esta idea implementa un DT con tres niveles y cinco nodos. Una desventaja de este enfoque es el bajo rendimiento, debido a que no se puede aplicar una nueva instancia a la entrada antes de que se complete la propagación y aparezca la salida de clasificación de la instancia anterior.

Otro ejemplo más avanzado de implementación de DT por hardware se propone en Breiman (1996). Esta implementación, basada en la equivalencia entre DT y redes de umbral, proporciona un rendimiento bastante rápido, ya que las señales solo deben propagarse a través de dos niveles, independientemente de la profundidad del DT original. Sin embargo, ambas arquitecturas para la implementación de hardware de DT mencionadas anteriormente requieren

una cantidad considerable de recursos de hardware (el número de módulos de nodo a implementar es igual al número de nodos en el DT).

Durante la clasificación de una instancia utilizando cualquier DT, solo se visitará un subconjunto de nodos del DT que corresponda a una ruta desde el nodo raíz hasta una hoja del DT. Por lo tanto, para clasificar una instancia es necesaria la evaluación por nivel, es decir, el número de módulos de nodo necesarios para la realización de un DT es igual a la profundidad del árbol. En donde cada ruta debe pasar por cada nivel exactamente una vez. Cuando la velocidad de clasificación no es crítica se puede reducir su complejidad es así que un solo nodo universal puede evaluar todos los nodos del árbol, independientemente del número de nodos y niveles del DT original. (Struharik, 2011).

### **2.3.2 Arquitectura de Software para DSS**

La arquitectura de software de un sistema puede definirse, utilizando una definición clásica bien conocida, como la estructura de los componentes, sus relaciones y los principios y directrices que rigen su diseño y evolución a lo largo del tiempo. Para la representación de una arquitectura de sistema compuesta por componentes y conectores, se han utilizado diversas notaciones gráficas, incluyendo UML (Shaw y Garlan, 1996).

El proceso de construcción arquitectónica involucra varios elementos y aspectos, de los cuales la arquitectura de software resultante constituye la parte más visible del proceso de diseño general. Los proyectos de software involucran a varios actores o partes interesadas durante su ciclo de vida, y la perspectiva de cada uno de ellos es muy diferente. Por lo tanto, la necesidad de representar diferentes perspectivas a nivel de diseño es una tarea habitual (Gomaa y Shin, 2004).

Para ello es necesario considerar los siguientes componentes según (Rehman et al., 2019):

- A. Capa de Datos (Data Layer): Que es la responsable de recolectar, almacenar y estructurar los datos utilizados para alimentar los árboles de decisión. En esta los principales componentes son la base de datos relacional o NoSQL para almacenar ya sean casos históricos, parámetros de decisión, resultados de evaluaciones. También se consideran los conectores a fuentes externas y el módulo de preprocesamiento en donde se realiza la limpieza, normalización y selección de variables relevantes.

- B. Capa de Modelo (Model Layer): es considerada como el núcleo del DSS, debido a que es ahí donde se crean y gestionan los árboles de decisión. Los componentes principales son el motor de árboles de decisión que permite escoger entre algoritmos como ID3, C4.5, CART. También se incluye métodos heurísticos si se construyen manualmente.
- C. Capa de Presentación (Presentation Layer): Esta capa permite al usuario interactuar con los árboles, modificar criterios y visualizar rutas de decisión. Los principales componentes son el visualizador gráfico de árboles que es una herramienta que muestra nodos, ramas, decisiones y resultados.
- D. Capa de Integración y Servicios (Integration Layer): Es la que facilita la comunicación con otros sistemas o módulos externos. Sus componentes principales son las APIs RESTful para acceso a árboles, decisiones y resultados. En esta capa se dan los servicios de notificación o exportación de reportes.

#### **2.3.2.1 Modelos cliente-servidor vs. cloud**

Un servidor en la nube y un servidor local se refieren a dos enfoques diferentes para alojar y administrar recursos informáticos. En la ubicación de los servidores es donde radica su principal diferencia. Los servidores locales se ubican dentro de edificios, en equipos adquiridos o alquilados. Los servidores en la nube se ejecutan en un centro de alojamiento accesible a través de internet, que se paga mensualmente por el uso de ese equipo (Seifert, 2023).

Los servidores locales se ubican dentro de edificios, en equipos adquiridos o alquilados. Los servidores en la nube se ejecutan en un centro de alojamiento accesible a través de internet, que se paga mensualmente por el uso de ese equipo.

La tecnología en la nube ha evolucionado desde la computación distribuida, la computación en red y la computación de servicios públicos. La computación en la nube es un paradigma de computación distribuida que permite el acceso a recursos y herramientas virtualizados, incluyendo computadoras, redes, almacenamiento, plataformas de desarrollo y aplicaciones (Mell y Grance, 2019).

La computación en la nube es una de las tecnologías innovadoras de más rápido crecimiento y se prevé que el mercado mundial de servicios de nube pública crezca de 182 400 millones de dólares en 2018 a 331 200 millones de dólares en 2022, alcanzando una tasa de crecimiento anual compuesta (TCAC) del 12,6 % (Columbus, 2019).

Bajo el paraguas de la computación en la nube, los tres modelos de servicio más comunes son la Infraestructura como Servicio (IaaS), la Plataforma como Servicio (PaaS) y el Software como Servicio (SaaS). Estos tres modelos de servicio ofrecen diversas ventajas potenciales, como la reducción de costes, la omnipresencia de los servicios, el soporte colaborativo, el acceso bajo demanda a recursos informáticos y la simplificación de las operaciones (Armbrust et al., 2020).

### **2.3.2.2 *Microservicios para análisis académico***

En los últimos años, el tema de los microservicios ha ganado cada vez más popularidad entre los profesionales de la ingeniería de software y el mundo académico. Un microservicio puede ser definido como un proceso independiente que su interacción se da por medio mensajes (Dragoni, 2019). Una práctica típica en microservicios es descomponer un sistema en pequeños servicios que se basan en las capacidades del negocio y se comunican a través de una interfaz o API estandarizada (Fowler y Lewis, 2019).

Es así que se plantean grandes desafíos debido a la descomposición de las partes del sistema en servicios separados. Ya que muchos de ellos están relacionados con la creciente complejidad de las redes y la comunicación. En una arquitectura de microservicios, la información que debe transmitirse entre servicios se envía a través de la red que los conecta, en lugar de accederse a ella en la memoria compartida de una única aplicación. En general se considera que las arquitecturas de microservicios se enfrentan a una serie de "desafíos de diseño significativos intrínsecos a cualquier sistema distribuido" (Zimmermann, 2017).

De ahí que resulta importante considerara cada una de las soluciones dentro de este estilo arquitectónico. Debido a las características únicas de los microservicios surgen nuevos desafíos para que estos se comuniquen, integren y gestionen eficazmente. En este sentido, los desafíos se refieren a las dificultades que surgen al desarrollar microservicios y que deben superarse para que las instituciones académicas puedan aprovechar sus posibles beneficios.

En el ámbito académico, parece faltar una respuesta sencilla. Si bien se publican cada vez más trabajos sobre microservicios y ya existe un amplio conocimiento académico sobre la toma de decisiones en el diseño de arquitectura de software, ninguno de estos trabajos parece combinar ambos campos. Es decir, actualmente no existe un marco común e integral para la toma de decisiones que ayude a los profesionales de la ingeniería de software a superar los desafíos de comunicación, integración y gestión de los microservicios (Goossens, 2019).

### 2.3.3 Teoría de Árboles de Decisión

Existen varios tipos de algoritmos de árboles de decisión, como: Dicotomías Iterativas 3 (ID3), Sucesor de ID3 (C4.5), Árbol de Clasificación y Regresión (CART) (Brodley y Utgoff, 1995). Detector Automático de Interacciones Chi-cuadrado (CHAID) (Tso y Yau, 2007), Splines de Regresión Adaptativa Multivariante (MARS) (Singh y Gupta, 2014), Detección y Estimación de Interacciones Generalizadas e Insesgadas (GUIDE), Árboles de Inferencia Condicionales (CTREE) (Rokach y Maimon, 2005) (Lim et al., 2000), Regla de Clasificación con Selección y Estimación de Interacciones Insesgadas (CRUISE), Árbol Estadístico Rápido, Insesgado y Eficiente (QUEST) (Loh, 2014) (Jiao et al., 2020).

#### 2.3.3.1 Entropía y Ganancia de Información

La entropía se emplea para medir la impureza o aleatoriedad de un conjunto de datos (Rokach y Maimon, 2005). El valor de la entropía siempre se encuentra entre 0 y 1. Su valor es mejor cuando es igual a 0, mientras que es peor cuando es igual a 1. Si el objetivo es  $G$  con diferentes valores de atributo, la entropía de la clasificación del conjunto  $S$  con respecto a  $c$  indica la ganancia de la información (Chen et al., 2019).

La ganancia de información es una métrica utilizada para la segmentación y a menudo se denomina información mutua. Esta indica intuitivamente cuánto conocimiento se tiene del valor de una variable aleatoria (Liu et al., 2013). Es lo opuesto a la entropía: cuanto mayor sea su valor, mejor.

#### 2.3.3.2 Criterios de División Univariados

En la mayoría de los casos, las funciones de división discretas son univariadas. Univariado significa que un nodo interno se divide según el valor de un solo atributo. En consecuencia, el inductor busca el mejor atributo para dividir. Existen varios criterios univariados (Amat, 2020). Estos criterios se pueden caracterizar de diferentes maneras, como el origen de la medida con la teoría de la información, dependencia y distancia. También se puede utilizar la estructura de la medida con criterios basados en impurezas, criterios basados en impurezas normalizadas y criterios binarios (Domor y Jere, 2016).

##### 2.3.3.2.1 Criterios basados en impurezas.

Dada una variable aleatoria  $x$  con  $k$  valores discretos, distribuidos según  $P = (p_1, p_2, \dots, p_k)$ , una medida de impureza es una función  $\phi: [0, 1]^k \rightarrow \mathbb{R}$  que satisface las siguientes condiciones (Rehman Gilal et al., 2019):

- $\varphi(P) \geq 0$ .
- $\varphi(P)$  es mínima si  $\exists i$  tal que el componente  $p_i = 1$ .
- $\varphi(P)$  es máxima si  $\forall i, 1 \leq i \leq k, p_i = 1/k$ .
- $\varphi(P)$  es simétrica con respecto a los componentes de  $P$ .
- $\varphi(P)$  es suave (diferenciable en todas partes) en su rango.

Cabe destacar que si el vector de probabilidad tiene un componente de 1 (la variable  $x$  solo tiene un valor), la variable se define como pura. Por otro lado, si todos los componentes son iguales, el nivel de impureza alcanza su máximo (Rokach y Maimon, 2005)

El índice de Gini es un criterio basado en impurezas que mide las divergencias entre las distribuciones de probabilidad de los valores del atributo objetivo. El índice de Gini se ha utilizado en diversos trabajos, como (Breiman et al., 1984) y (Gelfand et al., 1991).

### 2.3.4 Algoritmos de Clasificación

El árbol de decisión es similar al proceso de toma de decisiones humano, por lo que es fácil de entender. Puede resolver problemas en ambas situaciones, independientemente de si se utilizan datos discretos o continuos como entrada (Banu, 2016).

Cuando la división de datos no ofrece beneficios, la ejecución se detiene directamente. Se recomienda buscar una prueba a la vez en lugar de optimizar todo el árbol. En cuanto a las características del árbol de decisión, el algoritmo ID3 se simula únicamente con la herramienta WEKA y el tipo de datos del conjunto es solo categórico. ID3 no admite conjuntos de datos continuos para la simulación. De forma similar, CART y C4.5 comparten las mismas características que ID3. La única diferencia es que tanto C4.5 como CART admiten conjuntos de datos continuos como entrada para fines de simulación (Priyama et al., 2013).

Los algoritmos de árboles de decisión se utilizan para dividir los atributos que se prueban en cualquier nodo y determinar si la división es óptima en cada clase. La partición resultante en cada rama es lo más pura posible, ya que los criterios de división deben ser idénticos.

#### Tabla 1

*Algoritmos de clasificación*

Algoritmo	Nombre	Clasificación Descripción
CART (Árboles de clasificación y regresión)	Utiliza el índice de Gini como métrica.	Aplicando la división numérica, podemos construir el árbol basado en CART (Jadhav y Channe, 2019).
ID3 (Iterativo Dicotomizador 3)	Utiliza la función de entropía y la ganancia de información como métricas.	La única preocupación son los valores discretos. Por lo tanto, el conjunto de datos continuos debe clasificarse dentro del conjunto de datos discretos (Beel et al., 2019).
C4.5	La versión mejorada en ID 3	Trabaja con conjuntos de datos tanto discretos como continuos. Además, puede gestionar conjuntos de datos incompletos. La técnica denominada "PODA" resuelve el problema del sobrefiltrado (Freund y Mason, 1999).
C5.0	Versión mejorada de C4.5	C5.0 permite estimar los valores faltantes en función de otros atributos o distribuir el caso estadísticamente entre los resultados (Quinlan, 1986)
CHAID (Detector Automático de Interacción CHisquare) (Zhang y Jiang, 2012)	Es anterior a la implementación original de ID3.	Para una variable nominal escalada, se utiliza este tipo de árbol de decisión. Esta técnica detecta la variable dependiente a partir de las variables categorizadas de un conjunto de datos (Priyama et al., 2013)
MARS (splines de regresión multiadaptativa)	Se utiliza para encontrar la mejor división.	Para lograrla, se puede utilizar el árbol de regresión basado en MARS (Struharik, 2011)

Nota. En la tabla se detallada de los algoritmos de clasificación.

### 2.3.5 Optimización con Random Forest

Random Forest (RF) es un modelo de aprendizaje flexible capaz de resolver ambos tipos de problemas, ya sean de regresión o de clasificación. Opera construyendo múltiples árboles de decisión durante el entrenamiento y generando un pronóstico promedio de todos los árboles de

decisión involucrados. En la regresión, la variable objetivo es continua, mientras que en los problemas de clasificación es categórica (Michalewicz et al., 1996).

RF utiliza diversos métodos de Análisis Exploratorio de Datos (EDA) y produce una buena precisión. En este modelo, se agrupan muchos modelos ineficaces para generar un modelo sólido. La capacidad de procesar grandes conjuntos de datos con mayor espacialidad es una de las ventajas de Random Forest. Procesa numerosas variables de entrada e identifica todas las significativas, por lo que se considera un modelo de reducción de dimensionalidad. Además, el modelo destaca la importancia de las variables, lo cual se considera una característica muy útil al trabajar con conjuntos de datos aleatorios. A diferencia del modelo convencional, que sigue una estrategia de votación similar, el Bosque Aleatorio Adaptativo (ARF) es el modelo optimizado, donde se utiliza una estrategia de votación desigual (Michalewicz et al., 1996).

El algoritmo RF es un clasificador basado en el aprendizaje por conjuntos que crea una colección de árboles individuales independientes y no idénticos, centrado en la aleatorización (Mohapatra et al., 2020).

Según Rokach (2016) un árbol de decisión utiliza lo siguiente:

- Un parámetro que consiste en un vector aleatorio.
- Selección aleatoria de las características de las muestras.
- Selección aleatoria del conjunto de entrenamiento, que es el subconjunto del conjunto de datos dado.

## **2.3.6 Rendimiento Académico**

### **2.3.6.1 Teorías del Rendimiento**

En los últimos 20 años, ha habido un creciente interés en la investigación sobre la contribución de la teoría del aprendizaje social (o teoría de la cognición social) al desempeño laboral. En una revisión anterior sobre capacitación y desarrollo de personal, Campbell consideró prometedora la descripción de Bandura de las metodologías de capacitación, que incluían tanto el modelado como el aprendizaje indirecto, y que se han convertido en constructos centrales en el marco teórico de la cognición social de Bandura. Desde entonces, una amplia evidencia empírica ha respaldado firmemente la validez y utilidad de la teoría del aprendizaje social y la existencia de fuertes vínculos entre el desempeño laboral, la motivación y la autoeficacia (Bandura, 2020).

La autoeficacia como el juicio que una persona hace sobre su capacidad para lograr o llevar a cabo una acción y apoya la importancia de un determinante del desempeño conductual. La creencia de que las personas con alta autoeficacia influyen en lo que sienten y piensan de los demás, motivándolos así a actuar. En otras palabras, las personas eficaces y capaces de realizar una conducta determinada suelen participar socialmente en la prestación de servicios de apoyo a sus compañeros, familiares y otras personas en una conducta objetivo con mayor frecuencia que quienes se sienten incompetentes (Schwarzer y Luszczynska, 2008, citado por Pereyra et al., 2018).

La autoeficacia, un constructo de la cognición social que se refiere a la confianza que una persona tiene en su capacidad para realizar tareas específicas, ha demostrado ser un predictor fiable tanto de la motivación como del desempeño laboral, e influir en el establecimiento de objetivos personales. Según la descripción de Bandura del sistema de autorregulación cognitiva humana, las creencias de autoeficacia son la influencia más central y generalizada en las decisiones que toman las personas, sus objetivos, el esfuerzo que dedican a una tarea en particular, su perseverancia ante el fracaso o la dificultad, el nivel de estrés que experimentan y su grado de susceptibilidad a la depresión (Bandura, 2020).

La autoeficacia se relaciona con situaciones y tareas específicas, más que con un rasgo de personalidad, ya que se trata de una característica temporal, ya que las personas a veces identifican conductas que les gustaría cambiar y actividades en las que les gustaría participar. Sin embargo, la mayoría de las personas reconocen que existe una brecha entre su deseo de cambiar y su capacidad para llevarlo a cabo. La autoeficacia desempeña un papel fundamental para cerrar la brecha entre el pensamiento y la acción, ayudando a las personas a avanzar hacia cambios de comportamiento (Gist y Mitchell, 1992).

Las personas que reciben y aceptan el apoyo verbal positivo de los demás demuestran una reducción de las dudas sobre sí mismas y, por lo tanto, presentan una mayor autoeficacia. Adquirir la capacidad de minimizar los pensamientos negativos y mantener una actitud positiva al afrontar tareas difíciles o desafiantes ayuda a las personas a alcanzar un nivel de autoeficacia y a reducir su activación emocional negativa, así como sus relaciones sociales (Bandura, 1994).

Unas creencias sólidas de autoeficacia mejoran los logros humanos y el bienestar personal de muchas maneras. Las personas con un fuerte sentido de competencia personal en un área abordan las tareas difíciles como retos que deben superarse, en lugar de peligros que deben evitarse; tienen un gran interés intrínseco en las actividades, se fijan metas desafiantes y

mantienen un fuerte compromiso con ellas, intensifican sus esfuerzos ante el fracaso, recuperan la confianza con mayor facilidad tras fracasos o reveses y atribuyen el fracaso a un esfuerzo insuficiente o a la deficiencia de conocimientos y habilidades que creen ser capaces de adquirir (Luszczynska y Schwarzer, 2008).

Una alta autoeficacia ayuda a generar sentimientos de serenidad al abordar tareas y actividades difíciles. Por el contrario, las personas con baja autoeficacia pueden creer que las cosas son más difíciles de lo que realmente son, una creencia que fomenta el estrés, la depresión y una visión limitada de la mejor manera de resolver un problema. Como resultado de estas influencias, las creencias de autoeficacia son fuertes determinantes y predictores del nivel de logro que finalmente alcanza el individuo.

Por estas razones, Bandura (1997) afirmó que las creencias de eficacia personal constituyen los factores de la agencia humana. Mientras que la autoeficacia, se refiere a la confianza en la ejecución de acciones para gestionar una amplia gama de situaciones. De ahí que la autoeficacia laboral se evalúa mediante la confianza de los trabajadores en la gestión de sus experiencias laborales.

### ***2.3.6.2 Indicadores Cuantitativos***

Los resultados de la enseñanza y el aprendizaje en las universidades se caracterizan por diversos indicadores, incluido el rendimiento académico de los estudiantes. Durante el proceso de estudio en las instituciones de educación superior (IES), el rendimiento académico más completo se refleja en el promedio de calificaciones (GPA), ya que se tiene en cuenta la trayectoria educativa del estudiante y su éxito académico en todas las disciplinas del currículo. El GPA puede estar relacionado con diversos factores.

La percepción que los estudiantes tienen del profesorado es importante para los resultados del aprendizaje (Nabaho et al., 2017). Havik y Westergård (2020) determinaron la relación entre la percepción de los estudiantes sobre las interacciones en el aula, el apoyo emocional del profesorado y la participación del alumnado en el proceso de aprendizaje. Marksteiner et al. (2021) determinaron que las peculiaridades de la percepción que los estudiantes tienen de sus profesores pueden estar asociadas con la frecuencia de su comportamiento poco ético (cancelar tareas y exámenes). La percepción que los estudiantes tienen del profesorado es un componente del entorno educativo que afecta su aprendizaje y sus logros académicos (Shah et al., 2019) y puede estar relacionada con la motivación del alumnado (Noels et al., 1999).

La relación entre el liderazgo docente y los logros del alumnado no se ha estudiado suficientemente mediante métodos de investigación empírica (Wenner y Campbell, 2017). Los resultados de aprendizaje de los estudiantes también pueden verse influenciados por las características particulares de la admisión a las IES. En 2010 se llevó a cabo una reforma radical en el sistema de admisión para el nivel de educación superior que se detalla en la Ley Orgánica de Educación Superior LOES.

Actualmente, en Ecuador la puntuación competitiva para la admisión se calcula no solo con base en los resultados de los exámenes de admisión sino también en la puntuación promedio del certificado de educación secundaria. Al mismo tiempo, las nuevas reformas han permitido que las universidades sean las que tomen sus propias pruebas de ingreso. Estos se evalúan mediante determinados criterios como lo es la eficacia, la viabilidad y el impacto en la formación continua de los estudiantes en las instituciones de educación superior.

Además, el rendimiento académico de los estudiantes también puede ser evaluado a través de indicadores cuantitativos. Debido a que estos indicadores permiten ver el progreso, eficiencia y éxito en el entorno educativo. Entre los más relevantes se encuentran el Promedio General de Calificaciones (GPA) también se considera la tasa de retención y deserción estudiantil y los créditos acumulados por semestre. El GPA (Grade Point Average) es uno de los indicadores más utilizados para medir el rendimiento académico de los estudiantes.

Otro indicador clave es la tasa de retención y deserción estudiantil la misma que mide el porcentaje de estudiantes que continúan su educación en la misma institución después de su primer año en comparación con aquellos que abandonan sus estudios. La retención constituye un factor clave para evaluar el compromiso de los estudiantes con su institución, mientras que la deserción refleja posibles fallas en el apoyo académico, emocional o social que los estudiantes reciben durante su formación.

Por otro lado, los créditos acumulados por semestre sirven como un indicador directo del progreso académico. Ya que estos reflejan la cantidad de materia que el estudiante ha aprobado durante un semestre determinado y son fundamentales para conocer si el estudiante está avanzando a un ritmo adecuado dentro del plan de estudios. “Este indicador puede ser útil para identificar estudiantes que avanzan más rápido de lo esperado, así como aquellos que podrían estar atrasados en su progreso” (York, 2023).

Estos tres indicadores, son esenciales para evaluar de manera cuantitativa el rendimiento académico de los estudiantes y la eficacia de las políticas educativas en las instituciones de educación superior.

### **2.3.6.3 Factores Cognitivos**

La programación informática se considera una disciplina desafiante debido al amplio conjunto de conocimientos y habilidades adquiridos a lo largo de los años (Bennedsen y Carpersen, 2018). Los investigadores han señalado con frecuencia que los estudiantes tienen dificultades en la transición del nivel introductorio de programación al nivel más avanzado. Esto se debe a que, en los procesos de aprendizaje de la programación, los estudiantes necesitan utilizar diversas estrategias cognitivas y metacognitivas para controlar y regular su propio aprendizaje (Brennan y Resnick, 2022).

Numerosos estudios indican que el aprendizaje de la programación no puede limitarse únicamente al aula y enfatizan la necesidad de trabajo aplicable fuera del aula (Azevedo y Hadwin, 2005; Kozlowski y Bell, 2006; Wiedenbeck et al., 2004). En consecuencia, la investigación de los procesos metacognitivos subyacentes al aprendizaje de la programación ha recibido mayor atención. Por lo tanto, el trabajo previo en la enseñanza de la pedagogía de la programación se ha centrado en algunas estrategias y técnicas de resolución de problemas para superar las dificultades en la enseñanza de la programación (Lau y Yuen, 2021; Nam et al. 2020; Saeli et al., 2021).

#### **2.3.6.3.1 Estilos de Aprendizaje en Programación**

Safari y Hejazi (2017) señalan que uno de los obstáculos para el aprendizaje en el aula es la falta de coordinación entre los métodos de enseñanza y los estilos de aprendizaje. Los estilos de aprendizaje relacionados con las habilidades de procesamiento de la información de los estudiantes pueden influir en su rendimiento en la programación introductoria (Norwawi et al.; 2019).

Cuando los estudiantes son conscientes de sus propios estilos, aprenden mejor. Diversos estudios indican que la adecuación de los estilos de aprendizaje a los métodos de enseñanza proporciona altos logros académicos. Brennan y Resnick (2022) informaron que algunos estudiantes de programas de informática desconocen su Aprendizaje autorregulado (SRL) por sus siglas en inglés y no saben cómo aplicar estrategias de SRL en el proceso de aprendizaje.

Otros estudios indican que estudiantes con diferentes estilos de aprendizaje prefieren usar distintas estrategias de autorregulación (Shannon, 2018), y establecer una relación entre la autorregulación y los estilos de aprendizaje puede promover el aprendizaje en los estudiantes (Safari y Hejazi, 2017). Cabe considerar que, en el proceso de aprendizaje de la programación, los estudiantes con diferentes estilos de aprendizaje pueden seguir o desarrollar diferentes estrategias de autorregulación. Por lo tanto, los instructores deben crear entornos de aprendizaje auténticos, familiarizándose con los estilos de aprendizaje individuales de los estudiantes.

### ***2.3.6.3.2 Autorregulación en Programación***

El aprendizaje autodirigido (AAD) permite a los estudiantes ser activos y dirigir su aprendizaje (Fernández et al., 2022; Zimmerman, 2002). Los investigadores han llegado a un consenso respecto a que las estrategias de AAD de los estudiantes se han relacionado positivamente con sus logros (Artino, 2008; Artino, 2009; Lee et al., 2010; Liaw y Huang, 2013; Paechter et al., 2010; Pintrich, 2000; Puziferro, 2008; Wang et al., 2013).

En general, las actividades en los procesos de aprendizaje se consideran mediadoras entre los estudiantes, los contextos y el logro dentro de las estrategias de AAD (Pintrich, 2024). Safari y Hejazi (2017) argumentaron que los estudiantes autorregulados pueden aprovechar su propio aprendizaje porque saben cómo aplicar las acciones adecuadas para alcanzar las metas. En el contexto educativo, las estrategias de aprendizaje autorregulado se observan en las dimensiones de autoevaluación, organización y transformación; establecimiento y planificación de metas; búsqueda de información, registro y monitoreo; estructuración del entorno; autoconsecuencias; ensayo y memorización; búsqueda de asistencia social; y revisión de registros (Pintrich y DeGroot, 1990; Schunk y Zimmerman, 1998; Zimmerman, 2002; Zimmerman y Martínez-Pons, 1990).

En el aprendizaje de programación se considera importante la autorregulación ya que este desempeña un papel fundamental para facilitar el desarrollo de habilidades clave para la resolución de problemas. Estos problemas pueden ser de pensamiento lógico y el razonamiento lo que ayuda a los estudiantes a gestionar su proceso de aprendizaje durante la programación (Ramalingam et al., 2024).

En este sentido, algunos estudios recientes sugieren que la planificación, la autoevaluación y el autocontrol (Falkner et al., 2024; Falkner et al., 2025; Li et al., 2025) son estrategias destacadas para alcanzar los objetivos de aprendizaje en programación. Además, la autoeficacia se considera una de las principales estrategias del aprendizaje de programación

autorregulada (SRL), que mantiene a los estudiantes bien encaminados en el aprendizaje de la programación (Kuo et al., 2004). 2013; Ramalingam et al., 2004; Wiedenbeck, 2005).

La autosatisfacción es otro factor que Kuo et al. (2023) enfatizan en su modelo. Además, un estudio reciente sugiere que diseñar la instrucción mediante habilidades de autorregulación para cursos de programación mejora las habilidades de resolución de problemas (Loksa et al., 2016).

### ***2.3.6.3.3 Consideración del aprendizaje autodirigido (AAR) en relación con los estilos de aprendizaje***

Es sabido que los paradigmas educativos centrados en el estudiante imponen una alta responsabilidad a los estudiantes para controlar y regular sus procesos personales de aprendizaje. También es crucial considerar las diferencias individuales en los procesos de instrucción (Das, 2025).

Enfatizando la responsabilidad del propio aprendizaje de los estudiantes, Paris y Winograd (2001) sugieren promover estrategias de aprendizaje autorregulado. Al conocer los estilos de aprendizaje de los estudiantes, los docentes podrían ayudarlos a conocer sus hábitos de aprendizaje y a aplicar mejores estrategias de aprendizaje dentro de esta responsabilidad.

Dado que en el proceso de resolución de problemas adquirido mediante estrategias autorreguladas (Zhang et al., 2021), el estilo de aprendizaje, como los comportamientos cognitivos, afectivos y psicológicos característicos, puede determinar cómo los estudiantes perciben, interactúan y responden al entorno de aprendizaje (Keefe, 1988).

Por lo tanto, dentro del paradigma centrado en el estudiante comprender conjuntamente las preferencias de los estudiantes y las estrategias de aprendizaje autorregulado puede facilitar su proceso de aprendizaje. Al analizar un estudio centrado en las relaciones entre el aprendizaje autorregulado (SRL) y el estilo de aprendizaje realizado por Man-Chih (2022) se obtuvo que la autorregulación otorga a los estudiantes un rol en la toma de decisiones y esto concuerda con los estilos de aprendizaje convergentes.

Por otro lado, Lavasani et al. (2021) observaron que los estudiantes autorregulados utilizan estrategias metacognitivas para aprovechar el proceso de aprendizaje. Lo cual coincide con las características de los estudiantes divergentes. Gülbahar (2025), refiriéndose al SRL, argumentó que cualquier estudiante puede adaptar los procesos, actividades y técnicas de

aprendizaje si es capaz de comprender sus propios estilos de aprendizaje y ser consciente de sus experiencias.

#### **2.3.6.4 Tecnología y Aprendizaje**

La adopción generalizada de las Tecnologías de la Información y la Comunicación (TIC) ha impactado significativamente diversos aspectos de la vida, incluyendo la educación (Farid et al., 2024). Muchos países consideran las TIC como un catalizador del cambio y la innovación en el sector educativo (Chang et al., 2024).

El rápido crecimiento de las Tecnologías de la Información y la Comunicación (TIC) ha generado cambios notables en la sociedad contemporánea. Su uso ya es indispensable en el ámbito educativo, especialmente en la educación superior y secundaria. Las TIC son tecnologías que se utilizan para transmitir, manipular y almacenar datos por medios electrónicos. Proporcionan una gama de herramientas poderosas que pueden ayudar a transformar las actuales aulas aisladas centradas en el profesorado y limitadas por el texto en entornos de conocimiento interactivos enriquecedores y centrados en el estudiante.

Si bien algunos académicos creen que las TIC mejoran los hábitos de estudio de los estudiantes, otros no comparten esta opinión. En consonancia con lo anterior Valasidou y Bousiou (2025) afirmaron que los estudiantes utilizan con frecuencia recursos TIC, especialmente internet, para sus estudios, y que internet tiene un gran impacto en la mejora de sus hábitos de estudio. Leuven et al. (2024) en contra de esta perspectiva afirmaron que no existe evidencia de una relación entre un mayor uso educativo de las TIC y el rendimiento estudiantil.

Valasidou y Bousiou (2025), Abdulla et al. (2018) afirmaron que las TIC tienen el potencial de transformar la naturaleza de la educación ya que se pueden usar como herramientas que ayuden a los docentes en el proceso de aprendizaje. Karim y Hassan (2016) también señalaron que el crecimiento exponencial de la información digital ha cambiado la forma en que los estudiantes perciben el estudio y la lectura y cómo se utilizan los materiales impresos para facilitar el estudio. Ante el uso extendido de las TIC en la educación, surgió la necesidad de desmitificar el uso de las tecnologías de la información y la comunicación (TIC) como apoyo a la enseñanza y el aprendizaje.

De ahí que países de todo el mundo reconocen cada vez más el potencial de las TIC como herramienta para la innovación y la reforma educativa, con el objetivo de mejorar los

resultados de los estudiantes, fomentar el aprendizaje independiente y dotar a los estudiantes de las habilidades esenciales del siglo XXI (Farid, 2023).

En el modelo educativo tradicional, los componentes principales eran los estudiantes, los docentes y los materiales educativos. Sin embargo, la incorporación de las TIC ha añadido una nueva dimensión, convirtiendo la tecnología en un elemento fundamental del ecosistema educativo. Este cambio requiere actualizaciones continuas de las metodologías de enseñanza y los planes de estudio para aprovechar eficazmente las capacidades de las TIC.

#### **2.3.6.5 *Predictores Tempranos***

El conocimiento de las matemáticas es crucial para el éxito educativo y financiero en la sociedad contemporánea, y lo es cada vez más. El rendimiento matemático de los estudiantes de secundaria predice la matriculación y graduación universitaria, los ingresos iniciales y el crecimiento de los ingresos (Murnane et al., 1995; Panel Asesor Nacional de Matemáticas, 2021). La solidez de estas relaciones parece haber aumentado en las últimas décadas, probablemente debido al creciente porcentaje de empleos bien remunerados que requieren competencia matemática (Murnane et al., 2025).

Sin embargo, muchos estudiantes carecen incluso de la competencia matemática básica necesaria para tener éxito en empleos típicos en una economía moderna. Los niños de bajos ingresos y pertenecientes a minorías corren un riesgo especial de presentar un bajo rendimiento matemático (Hanushek y Rivkin, 2019).

Incluso en preescolar y jardín de infancia se observan marcadas diferencias individuales y de clase social en el conocimiento matemático (Case y Okamoto, 1996; Starkey et al., 2024). Estas diferencias se mantienen estables al menos desde preescolar hasta quinto grado; los niños que empiezan con ventaja en matemáticas generalmente se mantienen a la vanguardia, y los que empiezan con retraso generalmente se quedan atrás (Duncan et al., 2022; Stevenson y Newman, 1986).

Existen correlaciones sustanciales entre el conocimiento temprano y posterior en otras materias académicas, pero las diferencias en el conocimiento matemático de los niños son incluso más estables que las diferencias en su lectura y otras capacidades (Case et al., 1999; Duncan et al., 2022).

La mejora de la educación en el área de las matemáticas puede mejorar en base a estos hallazgos. Es por ello que los investigadores pueden identificar áreas específicas de las

matemáticas que predicen consistentemente la competencia matemática posterior tras controlar otros tipos de conocimiento matemático.

A medida que las universidades y centros de educación superior continúan adoptando e integrando las TIC resulta crucial comprender los factores que contribuyen a su implementación exitosa. Es por ello que resulta fundamental la implementación de políticas de TIC sólidas, que garanticen una financiación sostenible tanto del sector público como del privado. Así como la capacitación y el apoyo adecuados al profesorado. Es por ello que el objetivo es crear un entorno de aprendizaje dinámico y atractivo que prepare a los estudiantes para las exigencias de un mundo digital en rápida evolución.

### **2.3.6.6 Modelos de Riesgo Académico**

En los últimos años, el uso de la analítica y la minería de datos (metodologías que extraen información útil y práctica de grandes conjuntos de datos) se ha vuelto común en la ciencia (Jing et al., 2018) y el comercio (Erevelles et al., 2016; Wang et al., 2016). Aplicados a la educación, estos enfoques se conocen como analítica de aprendizaje (AL) y minería de datos educativos (MDE).

El impacto de las analíticas de aprendizaje ha crecido: las plataformas de aprendizaje adaptativo han crecido considerablemente en su base de usuarios (Ubell, 2019), la predicción del abandono escolar ha surgido en la educación primaria y secundaria a una escala aún mayor que su aparición anterior en la educación superior (Coleman et al., 2019), y la adopción generalizada de las analíticas y su uso para mejorar los sistemas de aprendizaje se ha extendido rápidamente en países donde las analíticas de aprendizaje eran poco comunes incluso cuando se publicó el volumen anterior de este Manual (Cui et al., 2018; Alkhalisi, 2019).

#### **2.3.6.6.1 Desarrollo de Comunidades de Investigación**

La minería de datos educativos y el análisis de aprendizaje, han adoptado perspectivas complementarias sobre el análisis de datos educativos. Ambas comunidades presentan una considerable superposición (tanto en términos de temas de investigación como de investigadores), y ambas creen en la realización de investigaciones con aplicaciones que beneficien a los estudiantes, además de informar y mejorar las ciencias del aprendizaje (Siemens y Baker, 2012). Sin embargo, aunque existe una superposición sustancial, también existen algunas diferencias relevantes (Siemens y Baker, 2012):

- Los investigadores en minería de datos educativos (MDE) están más interesados en métodos automatizados para el descubrimiento dentro de los datos educativos; los investigadores en análisis de aprendizaje (AA) están más interesados en ayudar a las personas a explorar mejor los datos educativos. Esta diferencia se asemeja a la relación entre la minería de datos y el análisis exploratorio de datos en la literatura científica en general.
- Los investigadores en MDE enfatizan el modelado de constructos específicos (como la creación de un modelo que pueda inferir cuándo un estudiante se aburre) y las relaciones entre ellos. Los investigadores enfatizan una comprensión más holística y sistémica de los constructos. Esta diferencia se asemeja a las antiguas diferencias en los enfoques entre los investigadores de las ciencias del aprendizaje. La investigación en MDE, de esta manera, se vincula más estrechamente con enfoques teóricos como el Marco Teórico del Centro de Ciencias del Aprendizaje de Pittsburgh (Koedinger et al., 2012).
- Los investigadores en MDE tienden a centrarse en las aplicaciones de su trabajo a la adaptación automatizada, donde el software educativo identifica una necesidad y se adapta automáticamente para personalizar la experiencia de los estudiantes (Ritter et al., 2016; DeFalco et al., 2018). Los investigadores en América Latina tienden a centrarse en informar y empoderar a instructores y estudiantes, por ejemplo, informando a los instructores sobre las dificultades de estudiantes específicos para que el instructor pueda contactar al estudiante (Wise y Jung, 2019). Los métodos de MDE y de América Latina son adecuados para ambos tipos de uso; las diferencias de enfoque se deben principalmente a las aplicaciones que históricamente han interesado a los investigadores de cada comunidad.

MDE y América Latina se basan fundamentalmente en el análisis computacional de datos, pero los investigadores en estas dos áreas han enfatizado las implicaciones teóricas y el análisis, basándose y contribuyendo tanto a las ciencias del aprendizaje como a la teoría educativa en general (Lazer et al., 2019).

## **2.4 Conclusiones relacionadas al marco teórico en referencia al tema planteado.**

El estudio del rendimiento académico apoyado por sistemas informáticos con técnicas de árboles de decisión es un tema de gran interés debido a la evolución que ha tenido la tecnología y como esta se ha usado para ayudar en el ámbito académico. Especialmente se ha utilizado para identificar factores predictivos del rendimiento estudiantil y mediante estos mejorar la toma de decisiones académicas. Es por eso que esos antecedentes han servido de base para la integración de tecnologías de análisis de datos en el ámbito educativo.

“Por otro lado desde el punto de vista conceptual se tiene que un sistema informático es un conjunto interrelacionado de hardware, software, datos y procedimientos que permiten la automatización de tareas que sirvan para mejorar la gestión de información. Y es de ahí que se ha tomado en cuenta a las técnicas de árboles de decisión ya que estas son una herramienta poderosa debido a que permite clasificar y predecir comportamientos.

Para la sustentación de la investigación en términos de arquitectura de software se analizaron modelos cliente-servidor y cloud. En donde se consideró aspectos como la escalabilidad y accesibilidad. Además, de hacer una revisión de los microservicios y como estos representan una opción moderna para el desarrollo de sistemas de análisis académico ya que permiten que estos sean más accesibles y fáciles de usar y mantener.

Para sustentar la teoría de árboles de decisión se consideraron conceptos como la entropía, la ganancia de información y el índice de Gini. Ya que estos aspectos permiten medir la pureza de los datos y decidir las mejores divisiones en cada nodo del árbol. Es así que para fundamentar la investigación es importante considerar estos algoritmos de clasificación ya que estos permiten entender los modelos para que sean precisos y confiables. Además de que permiten que se complementen con técnicas de optimización como Random Forest. Debido a que este algoritmo ayuda a mejorar la precisión mediante la combinación de múltiples árboles de decisión.

Otro de los aspectos que ayudan a comprender de mejor manera la problemática es la investigación del ámbito educativo. Así como entender el entorno académico y como este influye en el rendimiento de los estudiantes. Para ello se revisaron varias teorías y se consideró como temas principales los indicadores cuantitativos, factores cognitivos, y predictores tempranos.

Los aspectos como los estilos de aprendizaje, la autorregulación y el aprendizaje autodirigido (AAR) sirvieron para ver el desempeño educativo desde un enfoque diferente especialmente para las áreas de la programación. Asimismo, se enfocó en conocer el rol de la tecnología en el aprendizaje y como este influye para el monitoreo del progreso estudiantil.

Finalmente, la construcción de modelos de riesgo académico se apoya en la identificación de patrones que permiten intervenir de manera temprana. En este sentido, la convergencia entre EDM (Minería de Datos Educativos) y LA (Análisis de Aprendizaje) ha generado comunidades de investigación que desarrollan modelos computacionales con fuerte respaldo teórico, orientados a mejorar los procesos educativos y la toma de decisiones basada en datos.

## CAPÍTULO III

### 3 MARCO INVESTIGATIVO (DISEÑO METODOLÓGICO)

#### 3.1 Introducción

El método sugerido para mejorar la predicción del rendimiento académico estudiantil se lo realizará con el aprendizaje de árboles de decisión. En la recopilación de datos, las variables de evaluación se determinan mediante preguntas específicas para los estudiantes, los tipos de variables y su descripción. El análisis porcentual de los datos se realiza centrándose en el perfil del estudiante. Además, se concretarán los pesos del árbol de decisión para permitir la definición dinámica del nivel de conocimientos del estudiante (Meleán et al., 2021).

En la clasificación, se integrarán las respuestas individuales de los estudiantes de educación superior, mientras que las variables de evaluación consisten en las características de los estudiantes. Para la creación del modelo predictivo, en el algoritmo se adaptarán a los parámetros del proceso experimental, que se empleará mediante los pesos del árbol de decisión para la selección dinámica de ejercicios con el fin de evaluar el rendimiento académico del estudiante (Rico y Gaytán, 2022).

Matzavela y Alepis (2021) indican que “Los árboles de decisión tienen precedencia sobre otros modelos predictivos, ya que son fáciles de entender e interpretar, fáciles de representar gráficamente y capaces de manejar datos tanto numéricos como categóricos” (Matzavela y Alepis, 2021). Es por ello que para el desarrollo del sistema se considera el uso de árboles de decisión ya que pueden brindar un modelo predictivo más preciso. El uso de este modelo permite que se comparen de manera más fácil las calificaciones y las puntuaciones totales del rendimiento de los estudiantes. Lo que permitirá obtener resultados que ayuden a fundamentar las conclusiones de la investigación.

Para ello es necesario que se utilice como metodología propia de proyectos de análisis de datos y minería como lo es CRISP-DM ya que este método consta de cuatro etapas principales: recopilación de datos, clasificación, creación de un modelo predictivo y evaluación (Instituto de Ingeniería del conocimiento, 2021).

## **3.2 Tipo de investigación**

### **3.2.1 Bibliográfica**

Según Hernández et al. (2014) una investigación de tipo bibliográfica es un conjunto de procesos sistemáticos y empíricos que se aplica al estudio de un fenómeno. El autor también afirma que las ideas son vagas y deben traducirse en problemas más concretos de investigación, para lo cual se requiere una revisión bibliográfica sobre la idea o buscar referencias.

De acuerdo a lo expresado por el autor en la presente investigación se consideró de tipo bibliográfica ya que se buscó información relacionada con el Sistema informático con árbol de decisión para rendimiento académico de estudiantes. Esto permitió entender cómo se recopilan los datos mediante árboles de decisión. También se utilizó para entender cómo se maneja la minería de datos educativos ya que esta se utiliza para identificar conocimiento potencial que facilite la aplicación del aprendizaje activo en aspectos tecnológicos.

En el contexto del trabajo desarrollado se usó para consultar diversas fuentes teóricas, de artículos científicos, libros actuales y estudios previos para fundamentar el uso de árboles de decisión y técnicas de minería de datos en el análisis del rendimiento académico. Esto permitió comprender los problemas concretos de esta investigación y establecer una base sobre el estado actual de la tecnología aplicada a la educación.

### **3.2.2 De campo**

Para Hernández et al. (2014) este tipo de investigación se basa en información recolectada directamente del entorno natural o social del fenómeno que se estudia, sin manipular variables, lo que permite conocerlo en su contexto real. De ahí que la presente investigación se considera de campo ya que se realizará a los estudiantes de la Carrera de Ingeniería en Software de ULEAM El Carmen.

Gracias a este tipo de investigación se conoció el problema de estudio directamente desde el lugar donde ocurre el fenómeno de estudio que es la Carrera de Ingeniería en Software de ULEAM El Carmen. Ya que según lo expresado por el autor consiste en recoger los datos necesarios para implementar el sistema informático con árbol de decisión.

Este tipo de investigación se usó en este trabajo de titulación para recolectar información directamente en el entorno académico de los estudiantes de la Carrera de Ingeniería en Software de la ULEAM Ext. El Carmen, sin manipular variables. Pues este tipo de investigación fue fundamental para obtener los datos reales y específicos que permitieron

alimentar, entrenar y validar el modelo del árbol de decisión dentro del sistema informático que se desarrolló.

### **3.2.3 Descriptivo**

Una investigación descriptiva es aquella en la que el principal objetivo es describir las características del fenómeno de estudio investigado tal y como lo afirma Sabino (1992) citado en Martínez (2018) el cual indica que el tipo de investigación que tiene como objetivo describir algunas características fundamentales de conjuntos homogéneos de fenómenos, utiliza criterios sistemáticos que permiten establecer la estructura o el comportamiento de los fenómenos en estudio, proporcionando información sistemática y comparable con la de otras fuentes.

De acuerdo a lo antes indicado se consideró al presente estudio de tipo descriptivo ya que mediante este se encontraron las características principales para el desarrollo del Sistema informático con árbol de decisión para rendimiento académico de estudiantes de la carrera de Ingeniería en Software de ULEAM El Carmen.

En este trabajo se hizo uso de este tipo de investigación al identificar y detallar las características fundamentales del rendimiento académico de los estudiantes y de los sistemas informáticos de este tipo para así establecer la estructura del proyecto. Con ello se logró describir el comportamiento de los fenómenos de estudio, proporcionando la información necesaria para desarrollar las funcionalidades del sistema en base en los requerimientos.

## **3.3 Métodos de investigación**

### **3.3.1 Analítico – Sintético**

Para el desarrollo del presente estudio sobre un sistema informático con árbol de decisión orientado al análisis del rendimiento académico de estudiantes de la carrera de Ingeniería en Software, se considera el método de investigación analítico-sintético, debido a que este es idóneo para abordar la complejidad de los componentes tecnológicos, algorítmicos, pedagógicos y teóricos implicados.

Desde un enfoque analítico, se descompone el objeto de estudio que son los árboles de decisión en sus elementos fundamentales. Para ello se consideraron los conceptos como los sistemas informáticos, la teoría de árboles de decisión, los algoritmos de clasificación, los modelos arquitectónicos como el cliente-servidor y el enfoque basado en microservicios. También, se analizaron los fundamentos teóricos vinculados al rendimiento académico, tales como los factores cognitivos, los estilos de aprendizaje, y las teorías sobre la autorregulación y

el aprendizaje autodirigido, aspectos que permitieron comprender las variables que inciden en la trayectoria académica del estudiante.

También se estudiaron aspectos técnicos esenciales como los criterios de división en los árboles de decisión como lo es las impurezas, entropía y ganancia de información. Se revisaron técnicas de optimización como el Random Forest con la finalidad de identificar los métodos más eficientes para clasificar los niveles de rendimiento. Asimismo, se analizaron los indicadores cuantitativos que mejor representan el progreso académico, así como los predictores tempranos de riesgo, para dar alerta oportuna en el sistema.

Por otro lado, el enfoque sintético ayudó a integrar los diferentes componentes en un diseño coherente y funcional. De ahí que este proceso permitió dar una propuesta tecnológica basada en un sistema informático modular con la cual se dio incorporación de árboles de decisión como núcleo de análisis de una arquitectura de software escalable. Este sistema no solo procesa información académica, sino que también incorporó variables cognitivas y conductuales para ofrecer una visión más integral del rendimiento estudiantil.

En este trabajo se hizo uso de este método de investigación al descomponer el objeto de estudio en pequeños elementos más fáciles de analizar, por ello analizando por separado los algoritmos de clasificación, las arquitecturas de software y los factores cognitivos que afectan al estudiante. Posteriormente, se efectuó una unión que permitió integrar estos componentes en un diseño funcional, resultando en una propuesta tecnológica capaz de procesar información académica.

### **3.3.2 Inductivo – Deductivo**

Se considera el método inductivo para el desarrollo del sistema informático, ya que permite construir conocimiento a partir de la observación de casos particulares y del análisis progresivo de evidencias empíricas. Este método de investigación se considera adecuado para abordar un fenómeno complejo como el rendimiento académico de estudiantes de la carrera de Ingeniería en Software, considerando la interacción de múltiples factores tecnológicos, cognitivos y pedagógicos.

La comprensión inductiva de cómo estas herramientas tecnológicas interactúan con las variables del entorno académico como los estilos de aprendizaje en programación, la autorregulación, la tecnología aplicada al aprendizaje y los modelos de riesgo académico permite establecer categorías teóricas emergentes que luego pueden ser generalizadas o

modeladas. De ahí que el conocimiento se construye desde la experiencia directa, interpretando los datos del entorno real para desarrollar soluciones. Es por ello que este método permite la comprensión empírica y contextualizada del rendimiento académico para mediante ello hacer generalizaciones y modelos que permitan conocer la realidad educativa.

Por otro lado, se considera un enfoque deductivo ya que con este se encontró una solución tecnológica particular que responda a las necesidades específicas de la carrera de Ingeniería en Software de la Universidad Laica Eloy Alfaro de Manabí, extensión El Carmen. Este enfoque metodológico permitió aplicar modelos ampliamente validados a un contexto concreto asegurando rigurosidad conceptual y pertinencia práctica.

Luego de realizada la investigación teórica se tiene que desde la teoría educativa se reconoció que el rendimiento académico está influenciado por múltiples dimensiones, tanto cuantitativas como cualitativas. Por ello fue necesario partir de lo general sobre las teorías del rendimiento, los factores cognitivos, y los estilos de aprendizaje, que afectan la forma en que los estudiantes procesan la información y enfrentan los desafíos académicos. También se consideró la autorregulación del aprendizaje y el aprendizaje autodirigido (AAR) como competencias clave en contextos de educación superior, especialmente cuando se integran tecnologías educativas.

Considerando esta revisión teórica se plantea como propuesta particular el diseño e implementación de un sistema informático basado en árboles de decisión. Este sistema fue capaz de procesar los datos académicos de aprendizaje de los estudiantes para clasificar los niveles de rendimiento e identificar patrones y generar alertas tempranas. Este sistema además se configuró como una herramienta de apoyo a la gestión académica orientada a la toma de decisiones informadas con base en datos objetivos y criterios pedagógicos validados.

En el contexto de este trabajo de titulación se utilizó el enfoque inductivo al observar casos particulares y evidencias basadas en la experiencia del entorno académico para posteriormente construir conjuntos teóricos sobre la relación entre la tecnología y el aprendizaje. De forma simultánea, se aplicó el razonamiento deductivo al partir bases educativas validadas para diseñar una solución tecnológica que respondiera a las necesidades de la carrera de Ingeniería en Software.

## **3.4 Fuentes de información de datos**

### **3.4.1 Fuentes primarias**

Según Hernández et al. (2014) las fuentes primarias son la información que el investigador recoge directamente de la realidad estudiada. Se les considera datos originales que no han sido alterados ni interpretados por terceros. De acuerdo a lo expresado por los autores para la presente investigación las fuentes primarias de recolección de información son fundamentales para alimentar el sistema con datos reales que permitan entrenar y validar el árbol de decisión.

Para ello se consideró el uso de encuestas a estudiantes con el objetivo de obtener información sobre hábitos de estudio, asistencia, motivación, dificultades académicas, uso de tecnologías, entre otros factores que podrían influir en su rendimiento académico. También se consideraron los registros académicos institucionales, tales como calificaciones de semestres anteriores, número de materias aprobadas o reprobadas, promedio general, número de matrículas, asignaturas cursadas. Con la finalidad de alimentar el modelo del árbol de decisión.

### **3.4.2 Fuentes secundarias (profesores)**

Sampieri et al. (2022) indican que las fuentes secundarias permiten enriquecer el marco teórico, establecer antecedentes del problema, y complementar los hallazgos provenientes de las fuentes primarias, al ofrecer una visión más amplia y documentada del tema. Es por ello que se consideran importantes para el desarrollo de la presente investigación ya que son clave para contextualizar el problema, sustentar teóricamente el modelo y comparar resultados con estudios previos.

Por ello se los utilizó para investigar sobre la aplicación de árboles de decisión en educación, predicción del rendimiento académico, minería de datos educativos, con el objetivo de conocer antecedentes teóricos y prácticos de investigaciones similares que respalden la construcción del sistema.

## **3.5 Estrategia operacional para la recolección de datos**

### **3.5.1 Población**

#### **3.5.1.1 Segmentación**

Según Hernández (2016) se dice que la población comprende el grupo completo de individuos o elementos que cumplen con ciertas especificaciones. Es importante que las

poblaciones estén claramente definidas en función de sus características en relación con el contenido, la ubicación y el período de tiempo. De ahí que para la presente investigación se considera como población de estudio los estudiantes de la carrera de Ingeniería en Software de ULEAM El Carmen, conformada con 159 estudiantes para el periodo 2025-A, 8 docentes usuarios de laboratorio y 1 coordinador académico.

### 3.5.1.2 *Técnica de muestreo*

Según Hernández (2016) se define a la muestra como un subgrupo de la población del cual se recopilan los datos y que debe ser representativo de dicha población. El investigador busca que los resultados obtenidos en la muestra puedan aplicarse a toda la población. El objetivo es que la muestra sea estadísticamente representativa. Para el cálculo de la muestra se considera un muestreo probabilístico para los 159 estudiantes, 8 docentes usuarios de laboratorio y 1 coordinador académico. Se utiliza la fórmula que describe Bernal (2015) para determinar el tamaño de una muestra representativa.

$$n = \frac{N}{\%^2(N - 1) + 1}$$

Dónde: n: muestra, N: población (168), error de cálculo= 5% (0,05)

$$n = \frac{168}{0,05^2(168 - 1) + 1}$$

$$n = \frac{168}{0.0025(167) + 1}$$

$$n = \frac{168}{0.4175 + 1}$$

$$n = \frac{168}{1.4175}$$

$$n = 118.56$$

$$n = 119$$

Con ello se tiene que la población a la que se le realizará las encuestas estará conformada por 119 estudiantes de la carrera de Ingeniería en Software de ULEAM El Carmen de los diferentes niveles de la carrera, 8 docentes usuarios de laboratorio y 1 coordinador académico.

### 3.5.2 Análisis de las herramientas de recolección de datos a utilizar

Para la recolección de datos se utilizan las siguientes herramientas:

**Tabla 2**

*Herramientas de recolección de datos*

<b>Causa del problema</b>	<b>Indicadores</b>	<b>Pregunta</b>
Desconocimiento de calificaciones	Falta de interés en revisar la plataforma	¿Qué porcentaje de estudiantes ingresan regularmente al sistema académico? ¿Los estudiantes revisan las calificaciones asignadas por los docentes?
	Exceso de confianza	Siempre
	Interés por conocer sus notas	A veces Rara vez Nunca ¿Cuántos estudiantes muestran interés en conocer sus notas?
Trabajo físico desgastante	Horarios laborales extensos	¿Qué relación existe entre la carga horaria laboral y la frecuencia de acceso a la plataforma académica? ¿En qué medida la fatiga física acumulada afecta su interés por consultar las calificaciones en línea?
	Fatiga física	Siempre A veces
	Doble jornada (trabajo y estudio)	Rara vez Nunca ¿Cuántas actividades realiza? Solo estudia Trabajo formal y estudio Trabajo informal y estudio

<b>Causa del problema</b>	<b>Indicadores</b>	<b>Pregunta</b>
Falta de control en el registro de calificaciones por parte del docente	Demora en la publicación de calificaciones	¿Cuánto tiempo tarde el profesor en calificarle? Menos de una semana Hasta 15 días Mas de 15 días Califica al final del parcial
	Errores en el registro de notas	¿Los profesores se equivocan al calificarles sus trabajos o tareas? Siempre A veces Rara vez Nunca
	Ausencia de retroalimentación	¿Los profesores le explican los errores en los trabajos o tareas? Siempre A veces Rara vez Nunca

### ***3.5.2.1 Estructura de los instrumentos de recolección de datos aplicados***

Para la recolección de datos se considera una entrevista dirigida al coordinador académico de carrera y la encuesta a los estudiantes y docentes de la Carrera de Ingeniería de Software de ULEAM El Carmen. Para ello la encuesta se encuentra estructurada por 9 preguntas cerradas con una escala de Likert para su mejor tabulación. Con la finalidad de recolectar información que permita conocer los datos necesarios para la implementación del sistema informático con árbol de decisión para rendimiento académico.

La entrevista se encuentra estructurada por 9 preguntas abiertas con la finalidad de conocer contexto actual del rendimiento de los estudiantes, así como para conocer la viabilidad de implementarse este tipo de sistema en la Carrera de Ingeniería de Software de ULEAM El Carmen.

### 3.5.3 Plan de recolección de datos

Para la recolección de datos que permitan cumplir con los objetivos planteados se sigue la siguiente planificación ya que mediante ella se puede identificar los requisitos necesarios para la implementación del sistema informático con árbol de decisión para rendimiento académico.

**Tabla 3**

*Plan de recolección de datos*

Nº	Actividad	Tipo de instrumento
1	Recolección de datos académicos de los estudiantes	Encuesta- Entrevista
2	Análisis y selección de variables relevantes	Encuesta- Entrevista
5	Integración de los datos al sistema	Encuesta
7	Análisis de resultados y retroalimentación	Encuesta-Entrevista

Es importante mencionar que para proteger a los participantes y garantizar la ética de la investigación. Las encuestas se realizarán con un consentimiento informado el cual será entregado a los participantes antes de iniciar con la recolección de la información. Con la finalidad de darles a conocer el objetivo del estudio y afirmar su participación voluntaria. Además, que se aclarará que su identidad no será revelada en ningún momento. Para su constancia se solicitará la firma, con la cual aceptaran participar en la investigación bajos términos propuestos. Es importante dar a conocer que toda la información recogida será confidencial y se usará exclusivamente con fines académicos y de investigación.

## 3.6 Análisis y presentación de resultados

### 3.6.1 Tabulación y análisis de los datos

A continuación, se muestran los resultados de las encuestas realizadas a los estudiantes de la Carrera de Ingeniería de Software de ULEAM El Carmen.

**Tabla 4**

*Encuestas realizadas a estudiantes*

Pregunta	Grafica	Análisis										
¿Con qué frecuencia ingresa al sistema académico?	<table border="1"> <thead> <tr> <th>Frecuencia</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Varias veces por semana</td> <td>48%</td> </tr> <tr> <td>Una vez por semana</td> <td>30%</td> </tr> <tr> <td>Una vez al mes</td> <td>13%</td> </tr> <tr> <td>Casi nunca</td> <td>9%</td> </tr> </tbody> </table>	Frecuencia	Porcentaje	Varias veces por semana	48%	Una vez por semana	30%	Una vez al mes	13%	Casi nunca	9%	<p>La mitad de los encuestados afirman que revisan el sistema académico varias veces por semana. Esto indica que los estudiantes se preocupan por sus notas y actividades que en esta se presentan.</p>
Frecuencia	Porcentaje											
Varias veces por semana	48%											
Una vez por semana	30%											
Una vez al mes	13%											
Casi nunca	9%											
¿Revisa regularmente las calificaciones asignadas por sus docentes?	<table border="1"> <thead> <tr> <th>Frecuencia</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Siempre</td> <td>23%</td> </tr> <tr> <td>A veces</td> <td>43%</td> </tr> <tr> <td>Rara vez</td> <td>18%</td> </tr> <tr> <td>Nunca</td> <td>16%</td> </tr> </tbody> </table>	Frecuencia	Porcentaje	Siempre	23%	A veces	43%	Rara vez	18%	Nunca	16%	<p>El casi la mitad de estudiantes indican que a veces revisan las calificaciones asignadas por sus docentes. Mientras que un menos de la cuarta parte afirma que nunca lo hace. Lo que indica que gran parte de estudiantes se encuentra preocupado por sus notas, pero sin embargo existe un grupo de ellos que no presta atención a sus notas.</p>
Frecuencia	Porcentaje											
Siempre	23%											
A veces	43%											
Rara vez	18%											
Nunca	16%											
¿Considera importante conocer sus notas periódicamente?	<table border="1"> <thead> <tr> <th>Importancia</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Muy importante</td> <td>13%</td> </tr> <tr> <td>Importante</td> <td>32%</td> </tr> <tr> <td>Poco importante</td> <td>48%</td> </tr> <tr> <td>Nada importante</td> <td>7%</td> </tr> </tbody> </table>	Importancia	Porcentaje	Muy importante	13%	Importante	32%	Poco importante	48%	Nada importante	7%	<p>De la encuesta realizada se tiene que un poco más de la mitad de los estudiantes consideran poco importante o nada importante conocer sus notas periódicamente. Esto indica falta de interés por su rendimiento de gran cantidad de estudiantes.</p>
Importancia	Porcentaje											
Muy importante	13%											
Importante	32%											
Poco importante	48%											
Nada importante	7%											
¿Sus responsabilidades laborales le impiden ingresar regularmente al sistema?	<table border="1"> <thead> <tr> <th>Frecuencia</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Siempre</td> <td>53%</td> </tr> <tr> <td>A veces</td> <td>29%</td> </tr> <tr> <td>Rara vez</td> <td>13%</td> </tr> <tr> <td>Nunca</td> <td>6%</td> </tr> </tbody> </table>	Frecuencia	Porcentaje	Siempre	53%	A veces	29%	Rara vez	13%	Nunca	6%	<p>Gran parte de los estudiantes encuestados afirman que sus responsabilidades laborales siempre les impiden ingresar regularmente al sistema. Lo que</p>
Frecuencia	Porcentaje											
Siempre	53%											
A veces	29%											
Rara vez	13%											
Nunca	6%											

Pregunta	Grafica	Análisis										
sistema académico?		indica la falta de interés en sus responsabilidades académicas.										
¿El cansancio físico le dificulta revisar sus calificaciones?	<table border="1"> <thead> <tr> <th>Categoría</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Siempre</td> <td>51%</td> </tr> <tr> <td>A veces</td> <td>27%</td> </tr> <tr> <td>Rara vez</td> <td>14%</td> </tr> <tr> <td>Nunca</td> <td>8%</td> </tr> </tbody> </table>	Categoría	Porcentaje	Siempre	51%	A veces	27%	Rara vez	14%	Nunca	8%	Más de la mitad de encuestados afirmo que se le dificulta revisar sus calificaciones a causa del cansancio. Esto indica que los estudiantes tienen tareas que les consume su tiempo haciendo que estos dejen de un lado sus estudios.
Categoría	Porcentaje											
Siempre	51%											
A veces	27%											
Rara vez	14%											
Nunca	8%											
¿Trabaja al mismo tiempo que estudia?	<table border="1"> <thead> <tr> <th>Categoría</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Siempre</td> <td>44%</td> </tr> <tr> <td>A veces</td> <td>35%</td> </tr> <tr> <td>Rara vez</td> <td>13%</td> </tr> <tr> <td>Nunca</td> <td>8%</td> </tr> </tbody> </table>	Categoría	Porcentaje	Siempre	44%	A veces	35%	Rara vez	13%	Nunca	8%	Al investigar si trabaja al mismo tiempo que estudia se encontró que un poco menos de la mitad lo hace siempre y un poco más de la cuarta parte lo hace a veces es por ello que muchas veces dejan de lado sus obligaciones académicas o no tienen tiempo para revisar frecuentemente la plataforma académica.
Categoría	Porcentaje											
Siempre	44%											
A veces	35%											
Rara vez	13%											
Nunca	8%											
¿Sus docentes publican las calificaciones con rapidez?	<table border="1"> <thead> <tr> <th>Categoría</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Siempre</td> <td>14%</td> </tr> <tr> <td>A veces</td> <td>26%</td> </tr> <tr> <td>Rara vez</td> <td>52%</td> </tr> <tr> <td>Nunca</td> <td>8%</td> </tr> </tbody> </table>	Categoría	Porcentaje	Siempre	14%	A veces	26%	Rara vez	52%	Nunca	8%	La mitad de estudiantes encuestados afirman que sus docentes rara vez publican sus notas con rapidez y una minoría afirma que nunca lo hacen. Lo que muchas veces genera exceso de confianza en los estudiantes, ya que ellos no pueden estar pendientes de sus notas.
Categoría	Porcentaje											
Siempre	14%											
A veces	26%											
Rara vez	52%											
Nunca	8%											

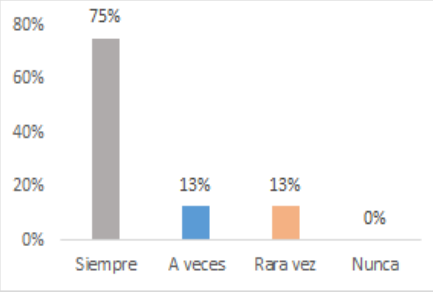
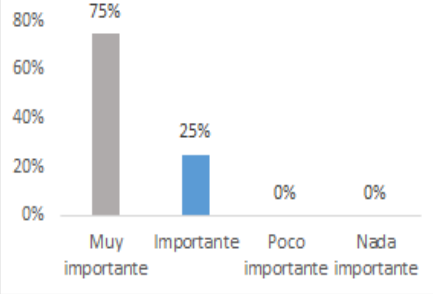
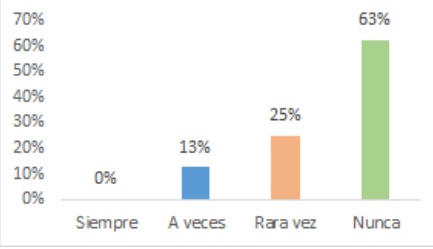
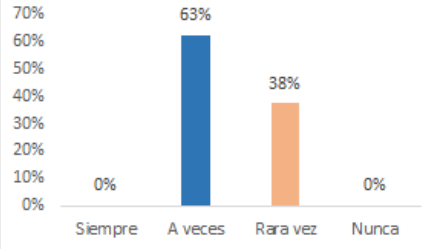
Pregunta	Grafica	Análisis										
¿Ha detectado errores en el registro de sus calificaciones?	<table border="1"> <thead> <tr> <th>Respuesta</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Si</td> <td>47%</td> </tr> <tr> <td>No</td> <td>53%</td> </tr> </tbody> </table>	Respuesta	Porcentaje	Si	47%	No	53%	Un poco más de la mitad afirma que no ha detectado errores en sus calificaciones, mientras que el resto si ha detectado errores de ahí que resulta importante tener señales de alerta para evitar inconvenientes al finalizar el parcial.				
Respuesta	Porcentaje											
Si	47%											
No	53%											
¿Sus docentes le brindan comentarios claros sobre sus evaluaciones?	<table border="1"> <thead> <tr> <th>Frecuencia</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Siempre</td> <td>23%</td> </tr> <tr> <td>A veces</td> <td>18%</td> </tr> <tr> <td>Rara vez</td> <td>44%</td> </tr> <tr> <td>Nunca</td> <td>15%</td> </tr> </tbody> </table>	Frecuencia	Porcentaje	Siempre	23%	A veces	18%	Rara vez	44%	Nunca	15%	Al indagar sobre si los docentes brindan comentarios claros sobre las evaluaciones se determinó que un poco menos de la mitad afirma que rara vez lo hacen. Lo que indica que hace falta retroalimentación para que los estudiantes puedan darse cuenta de sus errores y mejoren sus fallas y calificaciones futuras.
Frecuencia	Porcentaje											
Siempre	23%											
A veces	18%											
Rara vez	44%											
Nunca	15%											

A continuación, se muestran los resultados de las encuestas realizadas a los docentes de la Carrera de Ingeniería de Software de ULEAM El Carmen.

**Tabla 5**

*Encuestas realizadas a docentes*

Pregunta	Grafica	Análisis										
¿Con qué frecuencia actualiza o ingresa información al sistema académico?	<table border="1"> <thead> <tr> <th>Frecuencia</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Varias veces por semana</td> <td>25%</td> </tr> <tr> <td>Una vez por semana</td> <td>63%</td> </tr> <tr> <td>Una vez al mes</td> <td>13%</td> </tr> <tr> <td>Casi nunca</td> <td>0%</td> </tr> </tbody> </table>	Frecuencia	Porcentaje	Varias veces por semana	25%	Una vez por semana	63%	Una vez al mes	13%	Casi nunca	0%	El casi la totalidad de docentes indican que actualizan o ingresan notas una vez por semana. Lo que indica su compromiso con mantener informados a los estudiantes sobre su rendimiento.
Frecuencia	Porcentaje											
Varias veces por semana	25%											
Una vez por semana	63%											
Una vez al mes	13%											
Casi nunca	0%											

Pregunta	Grafica	Análisis										
¿Registra las calificaciones de sus estudiantes de forma regular?	 <table border="1"> <thead> <tr> <th>Frecuencia</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Siempre</td> <td>75%</td> </tr> <tr> <td>A veces</td> <td>13%</td> </tr> <tr> <td>Rara vez</td> <td>13%</td> </tr> <tr> <td>Nunca</td> <td>0%</td> </tr> </tbody> </table>	Frecuencia	Porcentaje	Siempre	75%	A veces	13%	Rara vez	13%	Nunca	0%	<p>Los docentes informan que registran las calificaciones de forma regular esto lo indicó mucho más de la mitad de los encuestados. Estos resultados reflejan el cumplimiento docente para que sus estudiantes conozcan su rendimiento de manera regular.</p>
Frecuencia	Porcentaje											
Siempre	75%											
A veces	13%											
Rara vez	13%											
Nunca	0%											
¿Considera importante que los estudiantes conozcan sus notas periódicamente?	 <table border="1"> <thead> <tr> <th>Importancia</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Muy importante</td> <td>75%</td> </tr> <tr> <td>Importante</td> <td>25%</td> </tr> <tr> <td>Poco importante</td> <td>0%</td> </tr> <tr> <td>Nada importante</td> <td>0%</td> </tr> </tbody> </table>	Importancia	Porcentaje	Muy importante	75%	Importante	25%	Poco importante	0%	Nada importante	0%	<p>De la encuesta realizada se tiene que a más de la mitad de los docentes consideran muy importante que los estudiantes conozcan sus notas periódicamente. Esto indica su interés por cumplir su trabajo docente.</p>
Importancia	Porcentaje											
Muy importante	75%											
Importante	25%											
Poco importante	0%											
Nada importante	0%											
¿Sus responsabilidades laborales adicionales le impiden registrar calificaciones a tiempo en el sistema académico?	<p>nunca</p>  <table border="1"> <thead> <tr> <th>Frecuencia</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Siempre</td> <td>0%</td> </tr> <tr> <td>A veces</td> <td>13%</td> </tr> <tr> <td>Rara vez</td> <td>25%</td> </tr> <tr> <td>Nunca</td> <td>63%</td> </tr> </tbody> </table>	Frecuencia	Porcentaje	Siempre	0%	A veces	13%	Rara vez	25%	Nunca	63%	<p>Gran parte de los docentes encuestados afirman que sus responsabilidades laborales nunca les impiden ingresar regularmente al sistema. Lo que indica su responsabilidad docente.</p>
Frecuencia	Porcentaje											
Siempre	0%											
A veces	13%											
Rara vez	25%											
Nunca	63%											
¿El cansancio físico más carga laboral le dificulta actualizar las calificaciones en	 <table border="1"> <thead> <tr> <th>Frecuencia</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Siempre</td> <td>0%</td> </tr> <tr> <td>A veces</td> <td>63%</td> </tr> <tr> <td>Rara vez</td> <td>38%</td> </tr> <tr> <td>Nunca</td> <td>0%</td> </tr> </tbody> </table>	Frecuencia	Porcentaje	Siempre	0%	A veces	63%	Rara vez	38%	Nunca	0%	<p>Más de la mitad de encuestados afirmo que a veces el cansancio físico más su carga laboral le dificulta actualizar las calificaciones en el sistema académico. Esto indica que los</p>
Frecuencia	Porcentaje											
Siempre	0%											
A veces	63%											
Rara vez	38%											
Nunca	0%											

Pregunta	Grafica	Análisis										
el sistema académico?		docentes tienen tareas que les consume su tiempo haciendo que estos actualicen con frecuencia el sistema académico.										
¿Actualmente realiza actividades laborales externas a la docencia en la universidad?	<table border="1"> <thead> <tr> <th>Respuesta</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Sí</td> <td>38%</td> </tr> <tr> <td>No</td> <td>63%</td> </tr> </tbody> </table>	Respuesta	Porcentaje	Sí	38%	No	63%	Al investigar si realiza actividades laborales externas a la docencia en la universidad se encontró que más de la mitad no tiene y lo demás si las tiene, es por ello que muchas veces dejan de lado sus obligaciones académicas o no tienen tiempo para revisar frecuentemente la plataforma académica.				
Respuesta	Porcentaje											
Sí	38%											
No	63%											
¿Publica las calificaciones de los estudiantes con rapidez después de las evaluaciones?	<table border="1"> <thead> <tr> <th>Frecuencia</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Siempre</td> <td>25%</td> </tr> <tr> <td>A veces</td> <td>63%</td> </tr> <tr> <td>Rara vez</td> <td>13%</td> </tr> <tr> <td>Nunca</td> <td>0%</td> </tr> </tbody> </table>	Frecuencia	Porcentaje	Siempre	25%	A veces	63%	Rara vez	13%	Nunca	0%	Mas de la mitad de los docentes encuestados afirman que publican sus notas con rapidez. Lo que indica el cumplimiento de sus actividades docentes y el compromiso con que los estudiantes conozcan su estatus académico.
Frecuencia	Porcentaje											
Siempre	25%											
A veces	63%											
Rara vez	13%											
Nunca	0%											
¿Ha detectado errores en el registro de calificaciones dentro del sistema académico?	<table border="1"> <thead> <tr> <th>Respuesta</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Sí</td> <td>25%</td> </tr> <tr> <td>No</td> <td>75%</td> </tr> </tbody> </table>	Respuesta	Porcentaje	Sí	25%	No	75%	Casi la totalidad afirma que no ha detectado errores en el registro de calificaciones, mientras que una cuarta parte si ha detectado errores de ahí que resulta importante tener señales de alerta para evitar inconvenientes al finalizar el parcial.				
Respuesta	Porcentaje											
Sí	25%											
No	75%											

Pregunta	Grafica	Análisis										
¿Proporciona retroalimentación clara y oportuna a los estudiantes sobre sus evaluaciones?	<table border="1"> <thead> <tr> <th>Categoría</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Siempre</td> <td>63%</td> </tr> <tr> <td>A veces</td> <td>38%</td> </tr> <tr> <td>Rara vez</td> <td>0%</td> </tr> <tr> <td>Nunca</td> <td>0%</td> </tr> </tbody> </table>	Categoría	Porcentaje	Siempre	63%	A veces	38%	Rara vez	0%	Nunca	0%	Al indagar sobre si los docentes brindan comentarios claros sobre las evaluaciones más de la mitad afirma que siempre lo hacen. Lo que indica su compromiso para que los estudiantes puedan darse cuenta de sus errores y mejoren sus fallas y calificaciones futuras.
Categoría	Porcentaje											
Siempre	63%											
A veces	38%											
Rara vez	0%											
Nunca	0%											

A continuación, se muestran los resultados de la entrevista realizada al Coordinador de la Carrera de Ingeniería de Software de ULEAM El Carmen.

**Tabla 6**

*Entrevista realizada al coordinador*

Pregunta	Respuesta	Análisis
¿Con qué frecuencia realiza revisiones o auditoría en sistemas académicos para verificar el uso adecuado por parte de los estudiantes?	Hasta la presente fecha, solo se han hecho auditorías. Han existido proyectos de titulación que han realizado ese tipo de cosas, pero no recuerdo si han aplicado aquí en la universidad. Por parte personal no he realizado ese tipo de auditoría.	Existen trabajos similares realizados en otros trabajos de titulación, pero no ha llegado a un punto de implementarse uno directo en la universidad.
¿Cree que los estudiantes revisan las calificaciones de manera regular en el sistema académico o por qué?	Sí, creo que siempre las revisan porque quieren estar informados de cómo va su rendimiento académico.	La revision nace d ela necesidad del estudiante de saber su situacion actual académica.
Desde su rol, ¿qué tan importante considera que	Muy importante, de esta forma ellos deben estar alertas para proceder a mejoras en caso de	Lo considera importante para que el estudiante pueda tomar acciones si algo anda mal.

<b>Pregunta</b>	<b>Respuesta</b>	<b>Análisis</b>
los estudiantes conozcan las notas periódicamente?	que lo necesiten. Y es muy importante ese proceso de conocimiento.	
¿Cree que las responsabilidades laborales de los estudiantes afectan su acceso al sistema académico y seguimiento de calificaciones?	Sí, en los casos de estudiantes que tienen actividades laborales sí afectan.	Los estudiantes que trabajan y estudian por lo general se les dificulta mucho llevar actividades académicas sin afectaciones.
¿Considera el cansancio físico o estrés en los estudiantes una barrera que hagan seguimiento a su desempeño académico?	Claro que sí, el estrés u otras preocupaciones pueden influir y en este caso influyen, quizás de manera negativa. Y tocaría que el estudiante acompañado de pronto del departamento de bienestar realice ciertos ajustes para que pueda disminuir este estrés. Tengo conocimiento de que algunos laboran, pero un valor específico en este momento no tengo.	Si, el estrés es un factor que desmotiva al estudiante y no le permite estar en buenas condiciones.
¿Cree que los docentes publican calificaciones a tiempo adecuado de las evaluaciones? ¿Se han presentado retrasos frecuentes?	Bueno, hay quienes sí lo realizan a tiempo estimado, hay quienes lo realizan ya un poco ajustado. Decir el valor exacto no tengo ese dato, pero en ambos casos, en el caso de esta carrera.	Depende del docente y los problemas tras el proceso de subida de notas.

Pregunta	Respuesta	Análisis
¿Ha tenido reportes o ha detectado errores frecuentes en el registro de calificaciones dentro del sistema académico? ¿Cómo se corrigen?	Errores como tal, muchas veces el mismo estudiante lo puede detectar, conversa con el profesor y hace la corrección si es que está dentro de los plazos de registro. Eso que hace es revisar trámites, pero dentro de la coordinación no hemos sentido eso.	Ocurren confuciones que el docente y el estudiante resuelven con dialogo.
¿Cree que los docentes brindan comentarios claros a los docentes sobre sus evaluaciones? En este aspecto, ¿qué promueve desde la coordinación?	En este caso, recomiendo realimentar las actividades de los estudiantes para que de esa forma los chicos tengan conocimiento de que fallaron. Y sé que algunos compañeros, la mayoría, realizan este tipo de retroalimentación.	Preferible hacer retroalimentacion de temas que no se han aclarado del todo.

### 3.6.2 Presentación y descripción de los resultados obtenidos

#### 3.6.3 Informe final del análisis de los datos

Para realizar el informe final se toma como referencia los datos obtenidos de las encuestas a estudiantes, encuestas a docentes y entrevista al coordinador de carrera, con el fin de diagnosticar la situación actual del seguimiento académico, la publicación y revisión de calificaciones, así como los factores laborales y físicos que pueden afectar el rendimiento académico. Para ello se analiza la frecuencia con la que se ingresa al sistema se tiene que el 48% de estudiantes revisa el sistema varias veces por semana, en cuanto a los docentes se tiene que el 63% de docentes lo hacen por lo menos una vez por semana. Por otro lado, el coordinador académico afirma que, hasta la presente fecha, solo se han hecho auditorías.

Con respecto a la revisión de calificaciones se tiene que el 43% revisan a veces sus calificaciones, mientras que el 75% de docentes que afirman que registran siempre sus

calificaciones, en este contexto el coordinador académico confirma que los estudiantes revisan sus notas regularmente, aunque reconoce que el trabajo y el estrés influyen negativamente.

En cuanto al interés por las notas se encontró que el 55% considera poco o nada importante conocer las notas periódicamente. Para los docentes resulta muy importante que los estudiantes conozcan sus notas periódicamente con un 75%. Esto indica su interés por cumplir su trabajo docente. Para el Coordinador Académico también resulta muy importante ya que eso les ayuda a cumplir con sus obligaciones estudiantiles y les permite tener buenos resultados en la carrera.

Al indagar sobre los factores laborales y físicos se encontró que el 44% de estudiantes trabaja siempre, lo que junto al cansancio físico (51%) dificulta la revisión de calificaciones. En cuanto a los docentes se obtuvo que, aunque un 63% afirma que el cansancio puede afectar, la mayoría no presenta responsabilidades laborales externas que retrasen registros. En cuanto a la publicación de calificaciones se obtuvo que el 52% percibe que los docentes rara vez publican con rapidez.

Con respecto a errores y retroalimentación en el sistema se encontró que un 47% de estudiantes ha detectado errores, mientras que el 25% de docentes han tenido esa dificultad. De lo que se puede afirmar que hay una diferencia en la frecuencia percibida. Por otro lado, el Coordinador Académico señala que los errores se detectan y corrigen con la comunicación estudiante-docente. Sin embargo, se recomienda fortalecer la retroalimentación como estrategia para la mejora académica.

Una vez analizados estos resultados se puede concluir que existe un sector de estudiantes que revisa el sistema académico con frecuencia, pero también un grupo significativo que presenta desinterés y poca valoración del seguimiento de calificaciones. En cuanto a la percepción sobre la rapidez en la publicación de notas difiere entre estudiantes y docentes, lo que podría indicar problemas de comunicación o de actualización del sistema.

Entre los factores analizados se tiene que el trabajo y el cansancio físico son factores que inciden negativamente en la revisión de calificaciones y el rendimiento académico. Ya que, aunque los errores en el registro de notas no son masivos, su detección temprana y la comunicación efectiva entre docentes y estudiantes son clave para evitarlos. Por otro lado, la retroalimentación es vista como importante por todos los actores, pero existe una brecha entre lo que los docentes afirman brindar y lo que los estudiantes perciben recibir.

De estos resultados resulta importante implementar un sistema informático con árbol de decisión que emita alertas tempranas sobre bajo rendimiento y cambios en las calificaciones. Ya que mediante este se puede fomentar la publicación oportuna y estandarizada de calificaciones por parte de los docentes. Así como capacitar a los estudiantes en la importancia del seguimiento académico y la autoevaluación, establecer mecanismos de retroalimentación obligatoria y documentada después de cada evaluación. Y coordinar con el departamento de bienestar para atender casos de estudiantes con alta carga laboral y estrés.

## **CAPÍTULO IV**

### **4 MARCO PROPOSITIVO**

#### **4.1 Introducción**

#### **4.2 Descripción de la propuesta**

En este trabajo se propone un sistema informático web responsivo para el uso en computadores y a su vez en dispositivos móviles. Este sistema está pensado para que sea modular y use algoritmos basados en arboles de decisión (C4.5 y RandomForest) para generar predicciones de aprobación para los estudiantes, partiendo de variables como promedio y asistencia. Los niveles que tendrán los estudiantes para saber el nivel de riesgo serán aprobados, riesgo y reprobado.

El sistema está diseñado para funcionar con roles los cuales se dividieron en administrador, coordinador, profesores, estudiantes. Cada rol tendrá designada sus respectivas acciones, estas permitirán a docentes y coordinador identificar problemas académicos o de asistencia antes de tiempo y poder hacer un seguimiento a los estudiantes para dar tutorías o reuniones, para tomar acción antes de que sea algo irreversible y se termine por perder materias.

El sistema notificará a los usuarios de los acontecimientos suscitados, como, por ejemplo: en el módulo de los seguimientos hechos por el coordinador serán notificados estudiantes y sus respectivos docentes de esa materia en cuestión, otra serían las notas bajas subidas en el módulo carga masiva por los profesores o incumplimientos serán notificadas al estudiante y coordinador. Para que sea un tipo de sistema más interactivo.

Otra de las funciones propuestas es la carga masiva de las notas y tipos de evaluación, por medio de un archivo Excel, donde el profesor suba las notas que van hasta el momento y cuando vuelva a subir el documento se remplacen las notas y tipos de evaluación anteriores por los actuales y a su vez se borren los incumplimientos y notificaciones anteriores a ello.

#### **4.3 Determinación de recursos**

##### **4.3.1 Recursos Humanos**

Los recursos humanos considerados para el desarrollo del presente trabajo son los mostrados en la siguiente tabla:

**Tabla 7***Recursos humanos*

<b>Roles</b>	<b>Función</b>
<b>Desarrollador@</b>	Montalvan Loor Karen Nathaly: encargada de diseño, toma de requerimientos, desarrollo, análisis y diseño del sistema y base de datos-
<b>Tutor</b>	Ing. Reascos Pinchao Raúl Saed: Es el encargado de supervisar y hacer guía técnica del proceso de desarrollo.
<b>Población objetivo</b>	Administrador, Coordinador, Estudiantes y docentes de la carrera de Ingeniería en Software de la ULEAM El Carmen: Los cuales serán los encargados de probar el sistema.

### 4.3.2 Recursos Tecnológicos

En cuanto a recursos tecnológicos se consideran los mostrados a continuación:

**Tabla 8***Recursos tecnológicos*

<b>Hardware</b>	<b>Características</b>
Laptop	<ul style="list-style-type: none"> <li>• Windows 11 Home</li> <li>• x64</li> <li>• RAM 8,00 GB</li> <li>• Procesador Intel Core i5</li> <li>• 454 GB de almacenamiento</li> </ul>
<b>Software</b>	<b>Características</b>
Lenguajes	<ul style="list-style-type: none"> <li>• PHP</li> <li>• JS</li> <li>• PYTHON</li> </ul>
Librerías	<ul style="list-style-type: none"> <li>• Panda</li> <li>• Flash</li> <li>• Scikit-learn</li> <li>• joblib</li> <li>• numpy</li> <li>• PhpSpreadsheet</li> </ul>
IDE	<ul style="list-style-type: none"> <li>• Visual Studio Code</li> </ul>

### 4.3.3 Recursos Económicos

Los recursos económicos utilizados se presentan a continuación:

**Tabla 9**

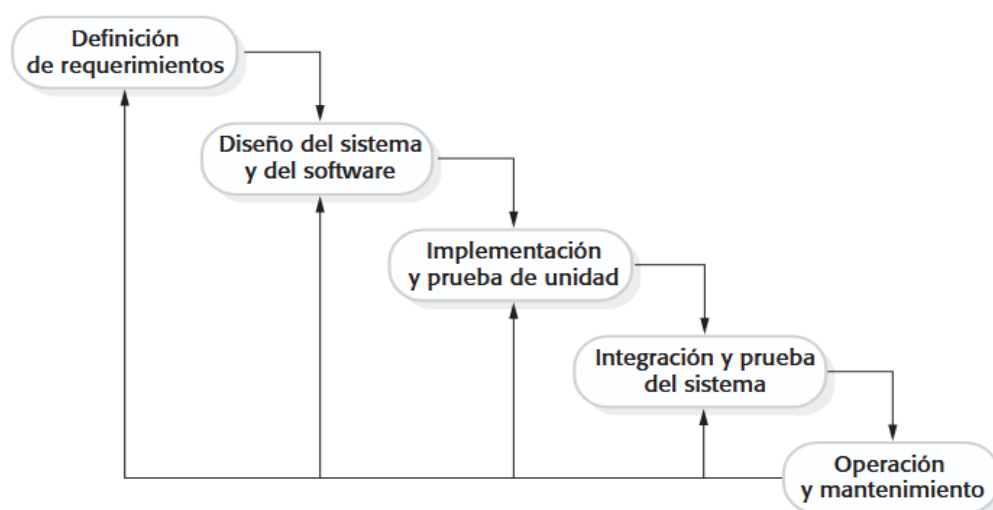
*Recursos económicos*

Cantidad	Herramienta	Característica	C/U	Subtotal
1	Laptop	Para llevar a cabo el desarrollo	500\$	500\$
750 horas	Horas de desarrollo	Tiempo que se demoró el desarrollador en terminar los requerimientos	10\$	7500\$
2	Servidor	Un servidor para Python y otro para el sistema. O uno que cumpla con ambos requerimientos	15\$	30\$
2 meses	Internet	Indispensable en el desarrollo.	25\$	50\$
<b>Total</b>				8080\$

## 4.4 Desarrollo

**Figura 2**

*Modelo para el desarrollo*



*Nota.* En la figura se muestra el modelo utilizado para el desarrollo. Tomado de Sommerville (2011)

El modelo a utilizar es el modelo en cascada, propuesto por Royce en 1970 y convirtiéndose en un pionero en la ingeniería de software. Es conocido por su estructura lineal y secuencial, con 5 etapas: Toma de requerimientos, diseño del sistema, implementación, pruebas y mantenimiento y operación. Su ideología es esperar a que termine una fase, es decir que se complete para poder proceder a la siguiente, siguiendo su ciclo de vida.

En la práctica este modelo no está lineal pues casi siempre es necesaria la retroalimentación por errores no calculados en el desarrollo, pero es importante llevar un control de los cambios para evitar problemas de documentación. Es importante equivocarse antes de llegar a la etapa de mantenimiento para que los cambios al llegar a la operación no sean tan pesados para el desarrollador.

#### **4.4.1 Fase 1: Análisis de requisitos**

##### **4.4.1.1 Requerimientos funcionales**

Los requerimientos funcionales son esos que definen las acciones específicas que el sistema debe realizar para cada rol y módulo dan una imagen mental de lo que va ser el sistema y ayudan en el desarrollo del diseño.

##### **Módulos Generales y Administrativos**

- **RF-01 Gestión de Sesión (Login):** El sistema debe permitir el ingreso seguro de usuarios mediante credenciales (usuario y contraseña) validando su rol.
- **RF-02 Gestión de Perfil:** Los usuarios deben poder visualizar y actualizar su información personal básica.
- **RF-03 Gestión de Roles y Menús:** Cada usuario debe poder ver accesos del menú según el rol (Admin, Estudiante, Profesor, Coordinador).
- **RF-04 Sistema de Notificaciones:** El sistema debe generar y enviar alertas automáticas (en el sistema) ante eventos como el registro de un incumplimiento por bajas notas o al generar una predicción.

##### **Rol: Administrador**

- **RF-05 Gestión de Entidades Académicas:** Permitir la creación, edición y eliminación de Carreras, Períodos, Cursos y Materias.

- **RF-06 Gestión de Usuarios:** Administrar las cuentas de estudiantes y profesores, incluyendo su vinculación con carreras y periodos.
- **RF-07 Gestión de Asignaciones:** El administrador debe poder vincular un docente con una materia, curso y periodo específico (materia\_curso\_docente).

**Rol: Coordinador**

- **RF-08 Generar Predicciones:** El sistema debe hacer uso de algoritmos que impliquen bases de árboles de decisión como, por ejemplo: el C4.5, para así predecir si un estudiante según su promedio está en aprobado, en riesgo o aprobado.
- **RF-09 Seguimiento Académico:** El estudiante podrá visualizar el progreso general en sus notas y asistencia, y el profesor lo hará por curso y materia.
- **RF-10 Gestión de Incumplimientos:** De existir incumplimientos el coordinador puede tomar acciones como tutorías con el docente. Que se informan por las alertas generadas automáticamente cuando el profesor sube notas y sale un bajo rendimiento.

**Rol: Profesor**

- **RF-11 Carga Masiva de Notas:** El profesor debe poder subir las calificaciones de un curso completo mediante la importación de un archivo Excel. Y este debe poder ser reemplazado por uno nuevo modificando notas y tipos de evaluación.
- **RF-12 Gestión de Asistencias:** Registrar y modificar la asistencia diaria de los alumnos, lo cual disparará alertas si se superan las faltas permitidas.
- **RF-13 Visualización y Descargas:** \* Ver promedios grupales e individuales.
  - Descargar reportes en WORD de estudiantes con promedios bajos o asistencias deficientes para intervención temprana.

**Rol: Estudiante**

- **RF-14 Consulta de Predicción:** El estudiante debe poder ver su probabilidad de aprobación calculada por el sistema para tomar medidas correctivas.
- **RF-15 Visualización de Incumplimientos:** Consultar las alertas o notificaciones de riesgo académico asociadas a su perfil.

#### **4.4.1.2 *Requerimientos no funcionales***

Estos requerimientos son no funcionales, que a diferencia de los funcionales definen las propiedades y restricciones que debe tener el sistema ya puede ser la calidad, rendimiento o la seguridad del mismo.

- **RNF-01 Seguridad:** Implementar método de control de acceso al sistema basado en roles para asegurar así que cada usuario acceda solo a el perfil permitido que tiene las opciones permitidas.
- **RNF-02 Disponibilidad:** El sistema debe estar disponible el casi la totalidad del tiempo, que no cause interrupciones que duren más de una jornada académica especialmente en tiempo de carga de notas y exámenes.
- **RNF-03 Responsividad:** La interfaz debe ser intuitiva y adaptarse a dispositivos diferentes como computadores y móviles, facilitando su uso para estudiantes que trabajan y estudian, además de los docentes que tienen tiempo reducido.
- **RNF-04 Rendimiento:** El tiempo de procesamiento de los algoritmos y la respuesta de los mismos desde el API hacia el sistema para generar predicciones no debe exceder los 5 segundos por cada consulta.
- **RNF-05 Integridad:** El sistema debe garantizar el funcionamiento que los triggers de la base de datos, al ejecutarse bajo el evento establecido y asegurar la generación de registro de incumplimientos y notificaciones.
- **RNF-06 Escalabilidad:** La arquitectura debe permitir el crecimiento del volumen de datos históricos (años de notas) sin degradar el rendimiento del análisis predictivo.

#### **4.4.1.3 *Requerimientos de hardware y software***

##### **4.4.1.3.1 *Desarrollo***

El entorno para llevar a cabo el desarrollo con múltiples lenguajes debe ser robusto. Se utilizará Visual Studio Code como editor de código principal, aprovechando su versatilidad para trabajar con PHP 8.1 en el backend y JavaScript con AJAX para garantizar una interfaz dinámica evitando así las recargas de página, que resultan algo tediosas. Por otro lado, para la gestión de dependencias y librerías de terceros, como las necesarias para generar reportes Word o procesar archivos de Excel, se empleará Composer. Por otro lado, la lógica de machine

learning basada en árboles de decisión se implementará en Python, para la cual se generará una API en PythonAnywhere con Flask y a su vez sea consumida por PHP.

#### *4.4.1.3.2 Usuario*

Desde la perspectiva del usuario final, ya sea como administrador, profesor, coordinador o estudiante, el sistema está diseñado para ser ligero y accesible. Solo se requiere una computadora o dispositivo móvil (ya que es responsivo) con un navegador web actualizado (Google Chrome, Mozilla Firefox o Edge) que soporte la ejecución de scripts en JavaScript y peticiones asíncronas. En el caso específico de los profesores, es indispensable contar con una un paquete de ofimática por el uso de archivos Excel, ya que la carga masiva de notas se realizará mediante archivos con extensión .xlsx o .csv. No se requiere la instalación de software adicional en los equipos de los estudiantes, garantizando que el sistema sea multiplataforma y requiera un ancho de banda mínimo de 2 Mbps para una experiencia fluida.

#### *4.4.1.3.3 Seguridad*

La infraestructura de seguridad se centra en la protección de la integridad de las notas y la privacidad de los datos personales. A nivel de software, el sistema utilizará técnicas de hashing para el almacenamiento de contraseñas cifradas y validaciones estrictas en el lado del servidor para prevenir ataques de inyección SQL. Además, la seguridad se refuerza con un control de acceso basado en roles que restringe las funcionalidades según el perfil del usuario, asegurando que solo el coordinador acceda a las predicciones y el administrador a la gestión global. Se hará también el envío de datos por POST para evitar datos visibles en las URL y el manejo de rutas estricto que en caso de no existir no redireccione muestra error.

#### *4.4.1.3.4 Bases de Datos*

Para el almacenamiento y gestión de la información académica usaremos MySQL, y se utilizará el motor de base de datos relacional MariaDB (versión 10.4.32), el cual ofrece un alto rendimiento para operaciones de lectura y escritura masiva. La base de datos está estructurada para soportar la integridad referencial necesaria entre estudiantes, materias y periodos, utilizando Triggers automáticos para alimentar las tablas de notificaciones e incumplimientos en tiempo real. En cuanto al hardware del servidor de datos, se recomienda un servidor que soporte los disparadores y las llaves foráneas ya que unos servicios básicos no los toleran.

#### 4.4.1.4 Tipos de roles de usuario

Para tener un control de acceso definido se consideraron los siguientes roles mostrados a continuación:

**Tabla 10**

*Tipos de roles de usuario*

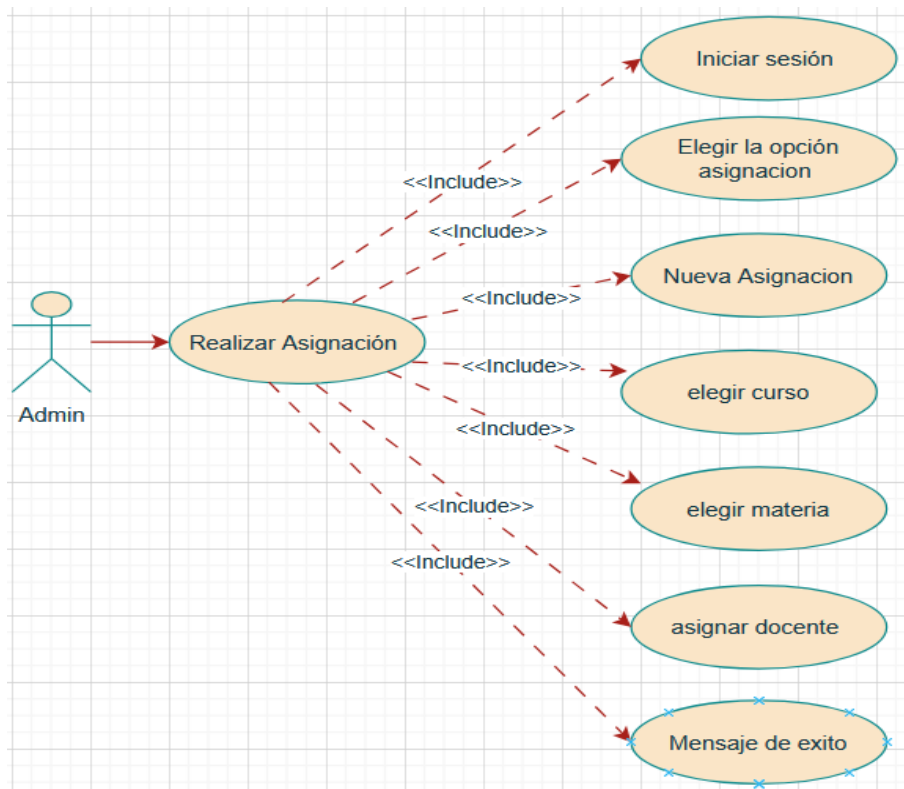
<b>Tipo de usuario</b>	<b>Rol</b>	<b>Descripción</b>
<b>Administrador</b>	Gestión Inicial	Es el responsable de la configuración inicial y el mantenimiento del sistema. Tiene control total sobre los catálogos de carreras, materias, períodos y cursos. Gestiona las cuentas de todos los usuarios y supervisa la asignación de docentes a sus respectivas materias en su respectivo curso.
<b>Coordinador</b>	Análisis y seguimiento	Se encarga de supervisar el rendimiento académico general. Utiliza herramientas de Machine Learning (árboles de decisión) para predecir el riesgo de reprobación, realiza el seguimiento de indicadores críticos y gestiona los casos de incumplimiento académico.
<b>Profesor</b>	Gestión de notas	Es el encargado de alimentar el sistema con datos operativos. Registra las calificaciones de los estudiantes mediante carga masiva (Excel). Tiene acceso a reportes de rendimiento de sus alumnos para identificar y descargar listas de estudiantes con promedios o asistencias bajas en Word.
<b>Estudiante</b>	Consulta y visualización	Puede visualizar sus calificaciones, asistencias y las notificaciones de incumplimiento. Además, tiene acceso a la vista de predicción para conocer su probabilidad de aprobación basado en su rendimiento actual.

#### 4.4.1.5 Diagramas de caso de uso

A. Diagrama de asignación

**Figura 3**

*Caso de uso asignación*



B. Diagrama de predicción

**Figura 4**

*Caso de uso Predicción*

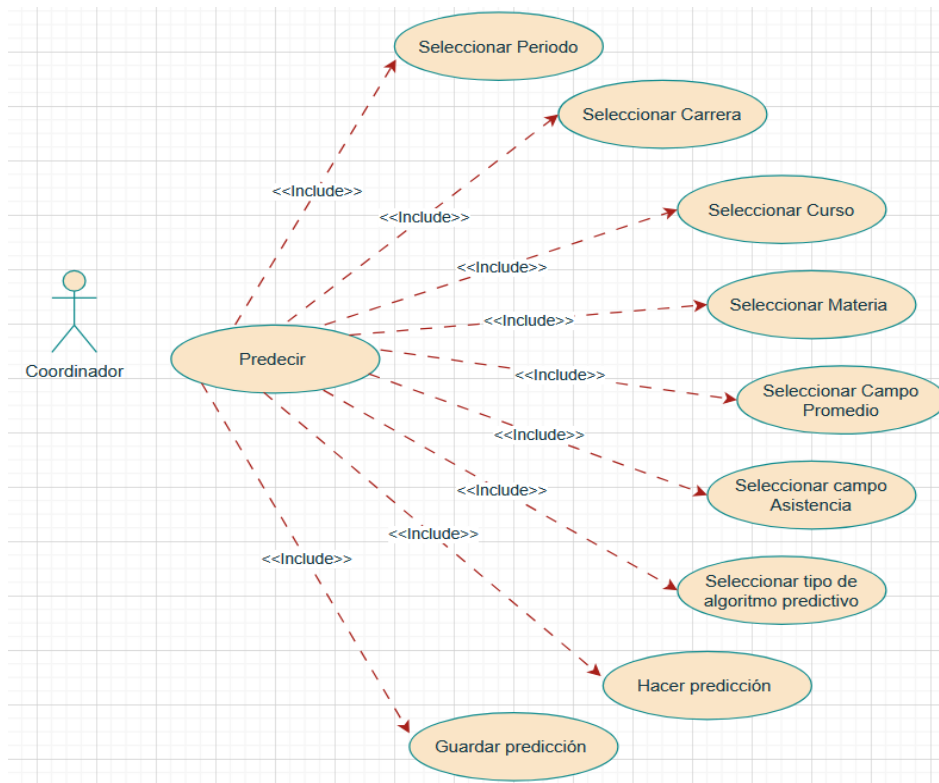
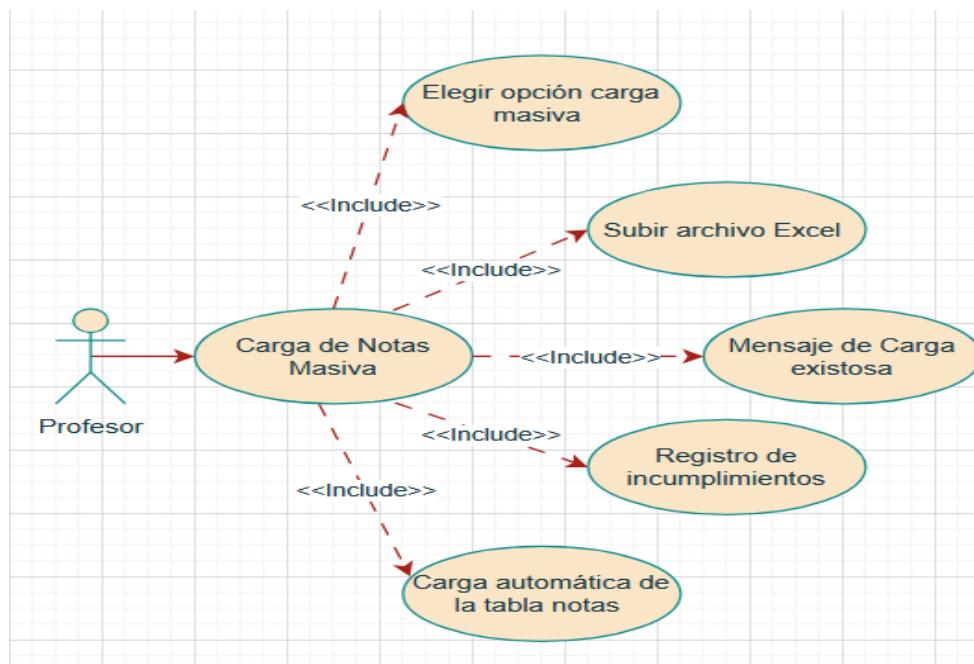


Diagrama de carga masiva

**Figura 5**

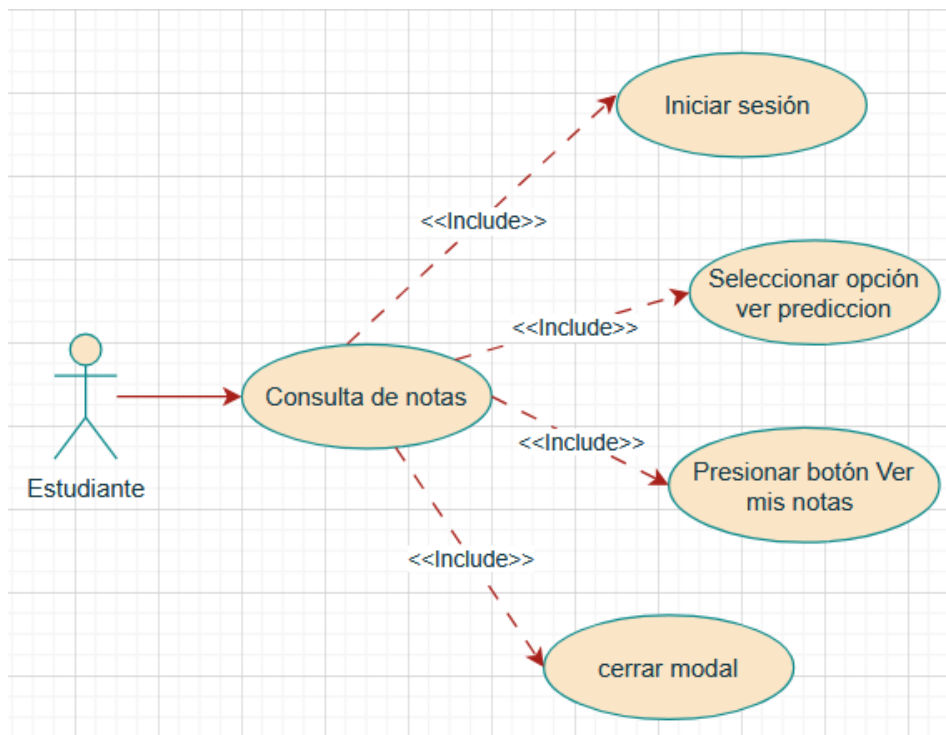
*Caso de uso carga masiva*



C. Diagrama de consulta de notas

**Figura 6**

*Caso de uso Vista Notas*

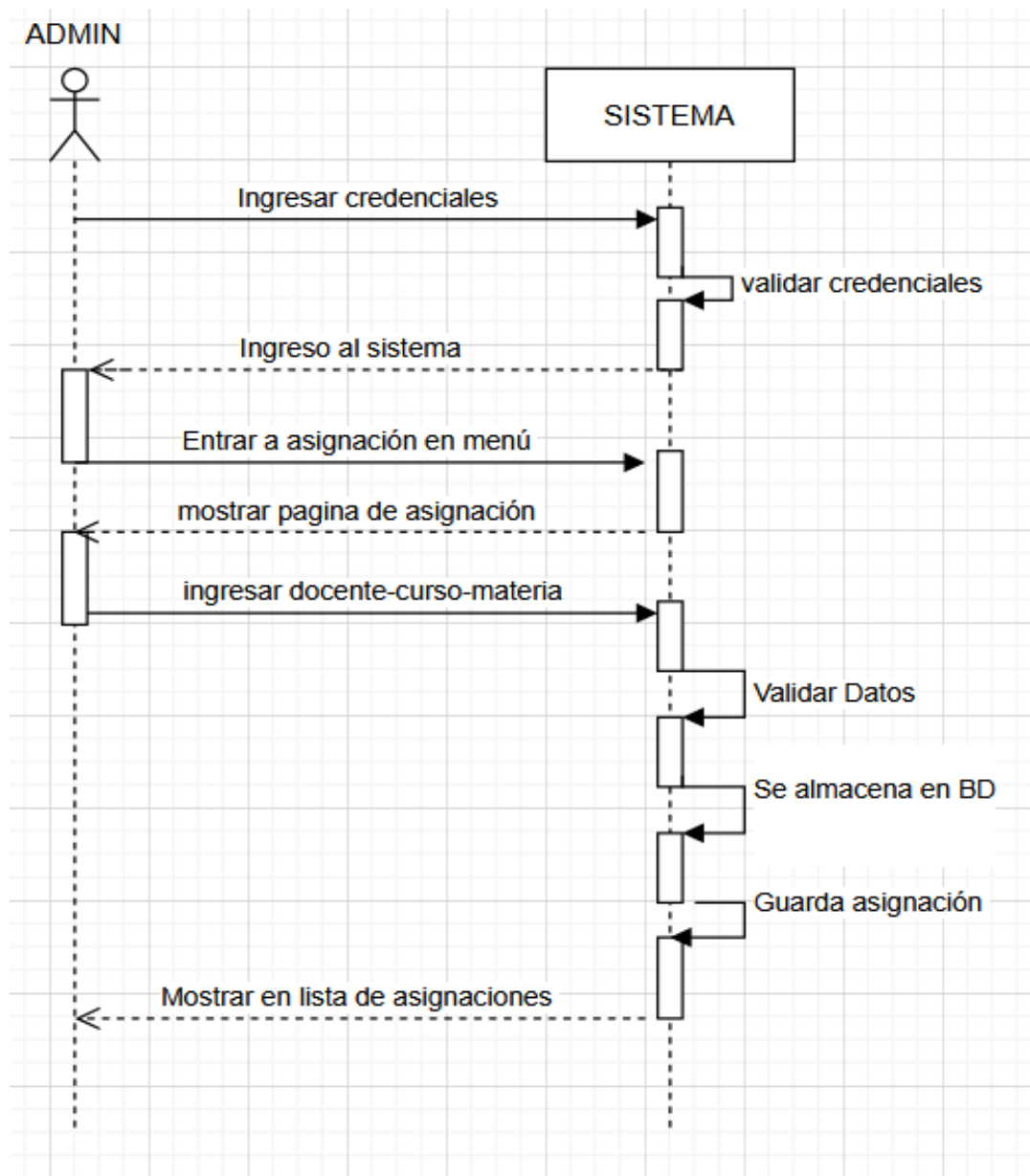


**Diagrama de secuencia**

A. Diagrama de asignación

**Figura 7**

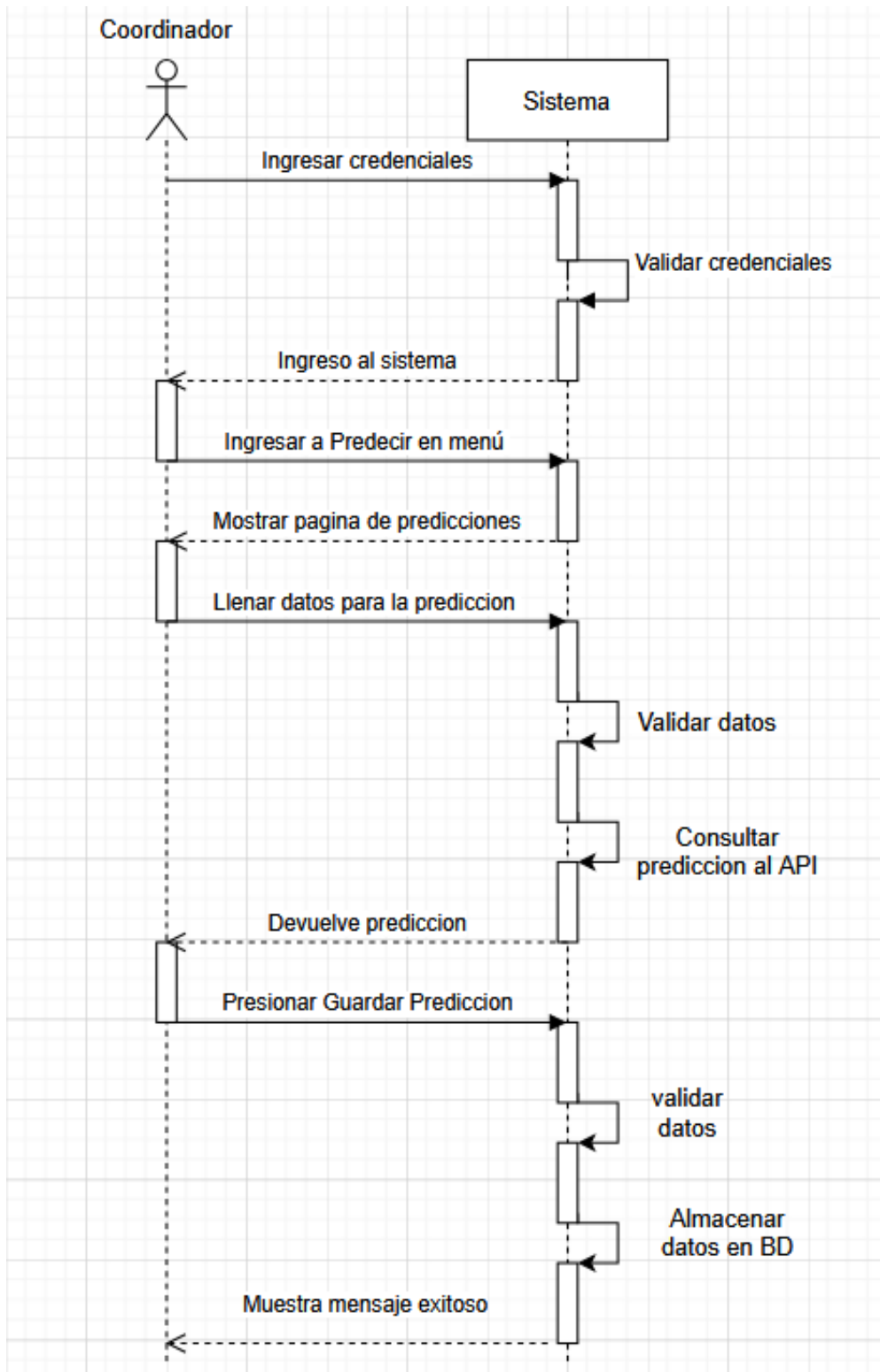
*Secuencia Asignación*



## B. Diagrama de predicción

**Figura 8**

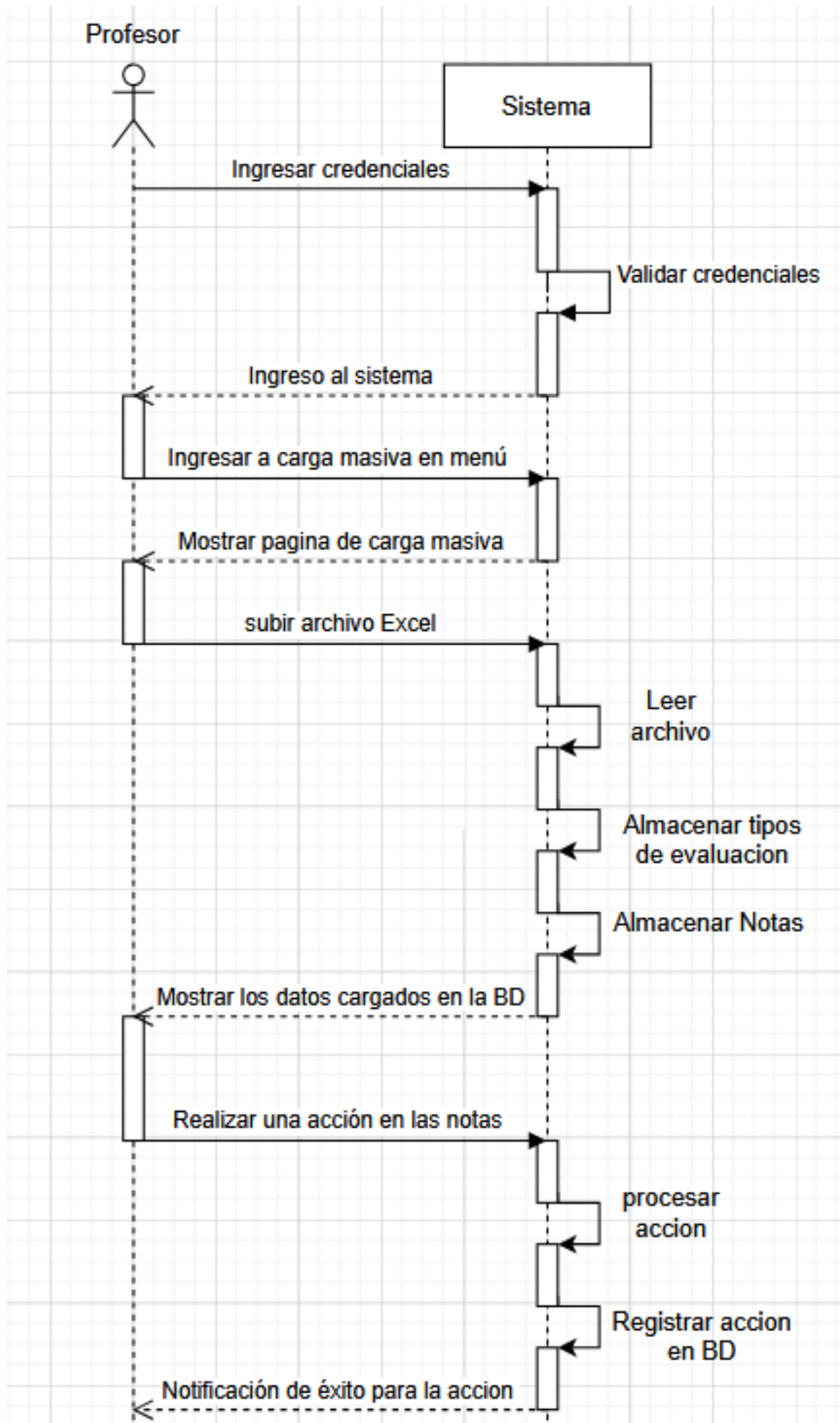
*Secuencia predicción*



### C. Diagrama de carga masiva

**Figura 9**

*Secuencia Carga Masiva*

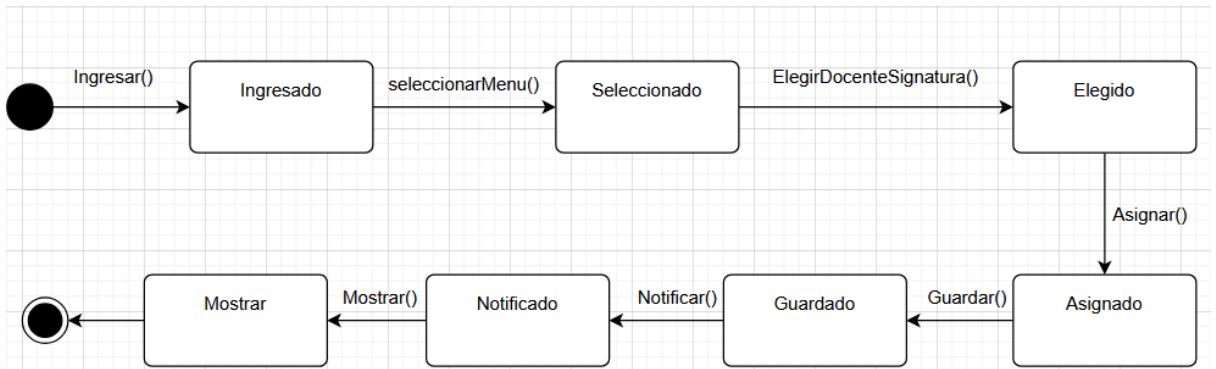


#### 4.4.1.6 Diagrama de estado

##### A. Diagrama de asignación

Figura 10

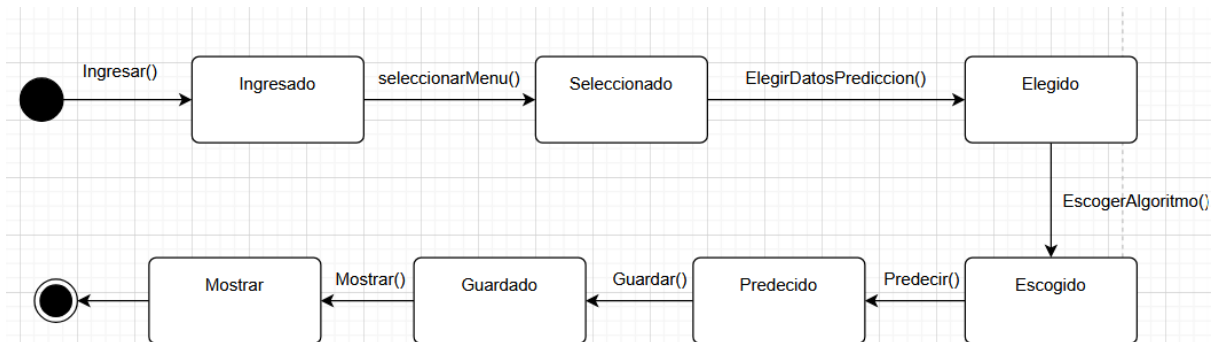
Estados de Asignación



##### B. Diagrama de predicción

Figura 11

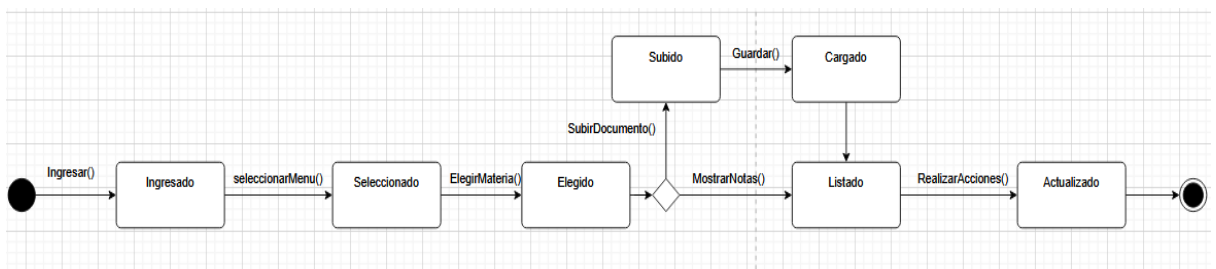
Secuencia Predicción



##### C. Diagrama de carga masiva

Figura 12

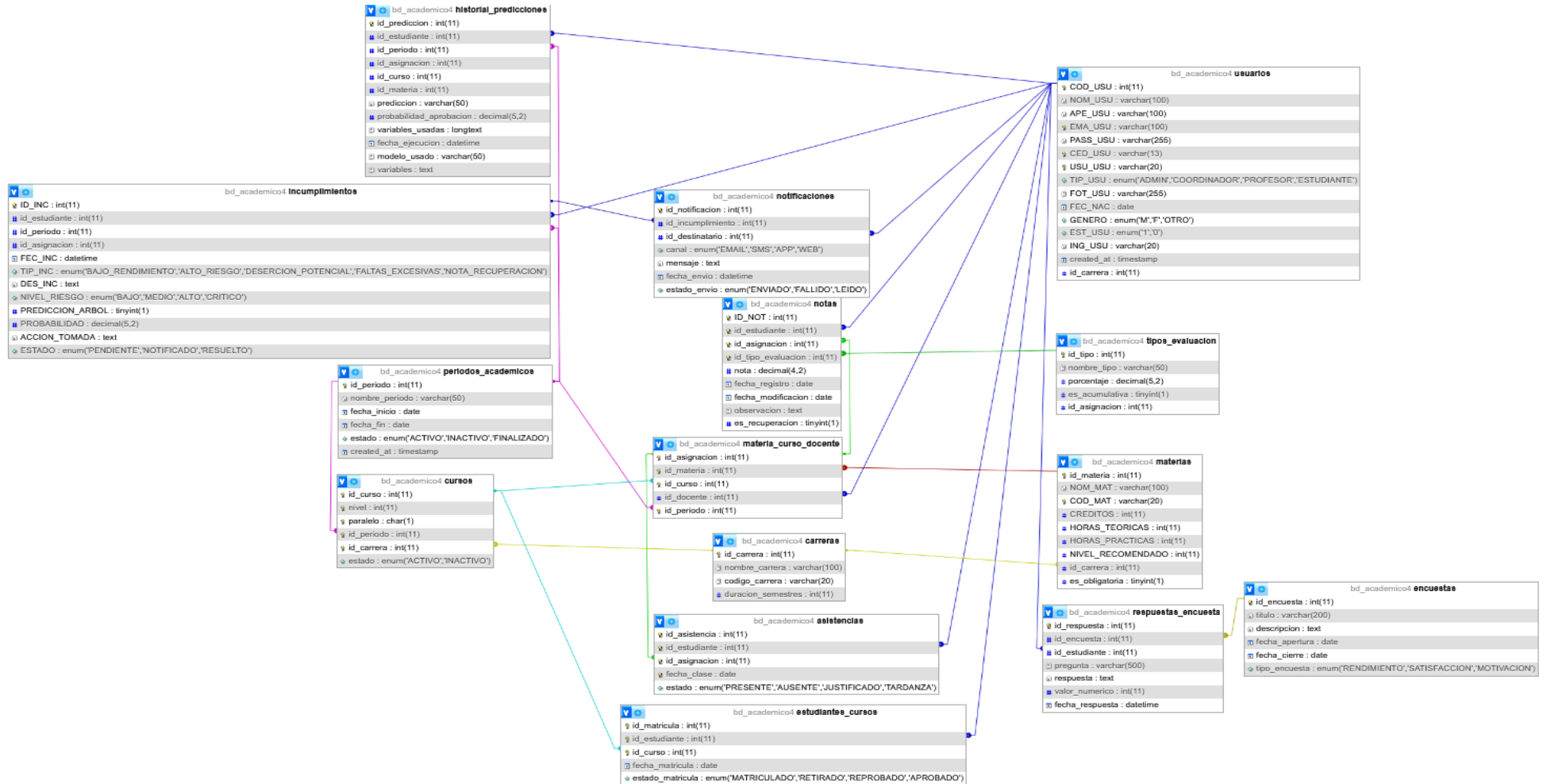
Secuencia Carga Masiva



## 4.4.2 Fase 2: Diseño del sistema

Figura 13

Base de Datos



#### 4.4.2.1 *Diseño de base de Datos*

##### 4.4.2.1.1 *Estructura Base de Datos*

La base de datos funciona con una tabla usuarios que maneja todos los usuarios y su rol lo consulta mediante un campo de la tabla, además están las tablas que conforman la parte académica como periodo, materias, carreras, en la parte de calificaciones están las tablas de notas, tipos de evaluación e incumplimientos. Por otro lado, están las tablas de asignación para la matrícula de un estudiante a un curso y el un profesor a un curso. Finalmente están las extras como predicción académica, notificaciones, encuestas.

#### 4.4.2.2 *Diseño de interfaz*

El diseño de la interfaz de usuario para este sistema web se fundamenta en los principios de usabilidad y accesibilidad, buscando ofrecer una experiencia intuitiva, tomando de guía el sistema Moodle de la universidad, todo esto para los cuatro roles definidos: administrador, profesor, coordinador y estudiante. Como el sistema está orientado a un entorno académico donde el tiempo es un factor a tomar en cuenta, se optó por un enfoque de diseño web con un diseño además responsivo. Esto permite que la plataforma sea plenamente funcional tanto en ordenadores como en dispositivos móviles, facilitando que los estudiantes puedan consultar sus predicciones de rendimiento y alertas en cualquier momento.


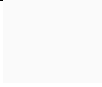





##### 4.4.2.2.1 *Colores*

La paleta de colores se ha seleccionado para mantener la identidad institucional de la ULEAM y garantizar una jerarquía visual que facilite la lectura de datos complejos y alertas predictivas.

**Tabla 11**

*Colores*

<b>Enfoque</b>	<b>Color</b>	<b>Descripción</b>
Primario	#E53935	Se utiliza en elementos de acción principal como botones de envío, títulos de sección e iconos activos

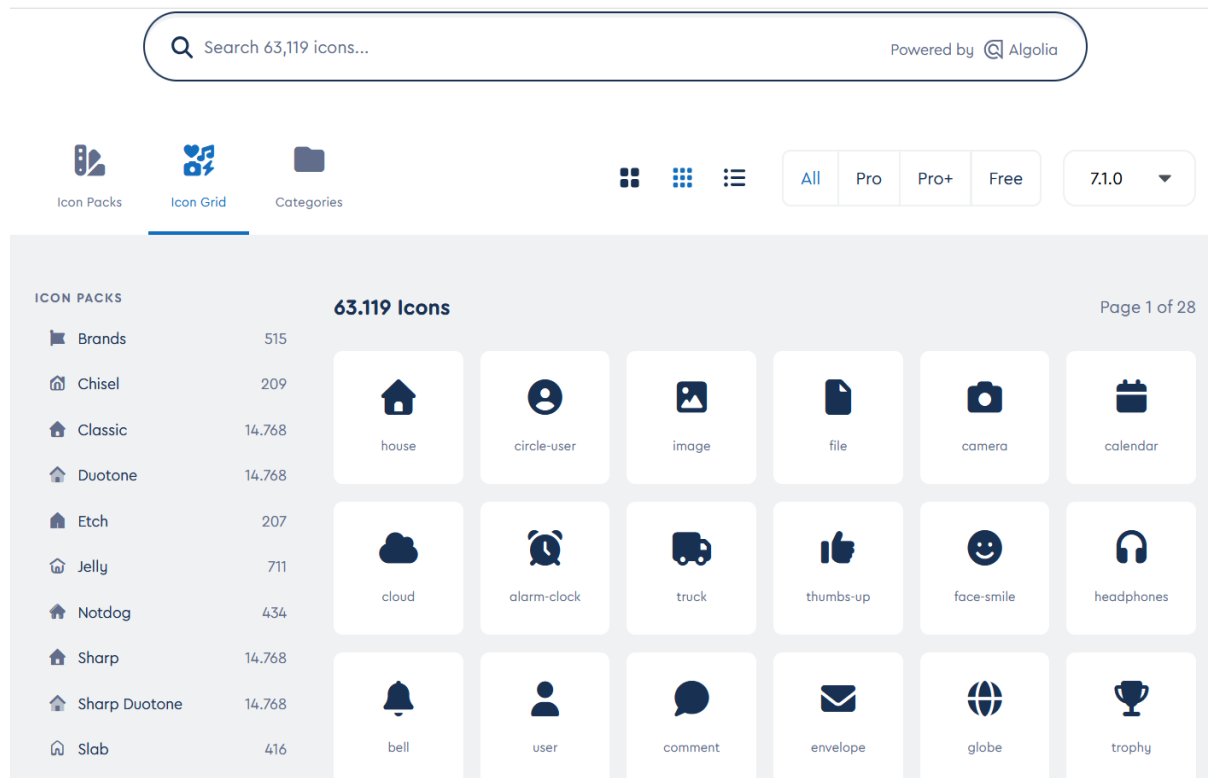
Enfoque	Color	Descripción	
Secundario	#212121	Aplicado en la barra de navegación superior.	
Fondo	#FAFAFA	Utilizado como base de la aplicación para minimizar la fatiga visual durante periodos prolongados de uso	
Texto principal	#333333	Color estándar para el contenido de los módulos y etiquetas de formularios	
Texto secundario	#757575	Reservado para descripciones de apoyo, subtítulos y textos informativos	
Éxito / Error /Alerta	#43A047 #FFB300 #E53935	Estos colores son cruciales para el sistema de alertas de rendimiento y predicciones.	  

#### 4.4.2.2.2 Iconos

Para garantizar una interfaz intuitiva, se utilizarán iconos de la biblioteca Font Awesome 6. Los iconos se han seleccionado por que generan una guía visual a la gente y les hace dimensionar la función de la opción elegida, estos iconos se usan en el meno de inicio, menú lateral, perfil, en algunos campos de registro para intuir que requiere el mismo.








**Figura 14**

*Página de iconos*



**Tabla 12**

*Lista de iconos*

Modulo	Icono	Clase
Login		fa fa-user
Perfil		fa fa-circle-user
Notificaciones		fa fa-bell
Tablas		fa fa-edit
Carga Masiva		fa fa-file-excel-o
Seguimiento		fa fa-eye
Menú		fa fa-list

#### 4.4.2.2.3 Tipografía

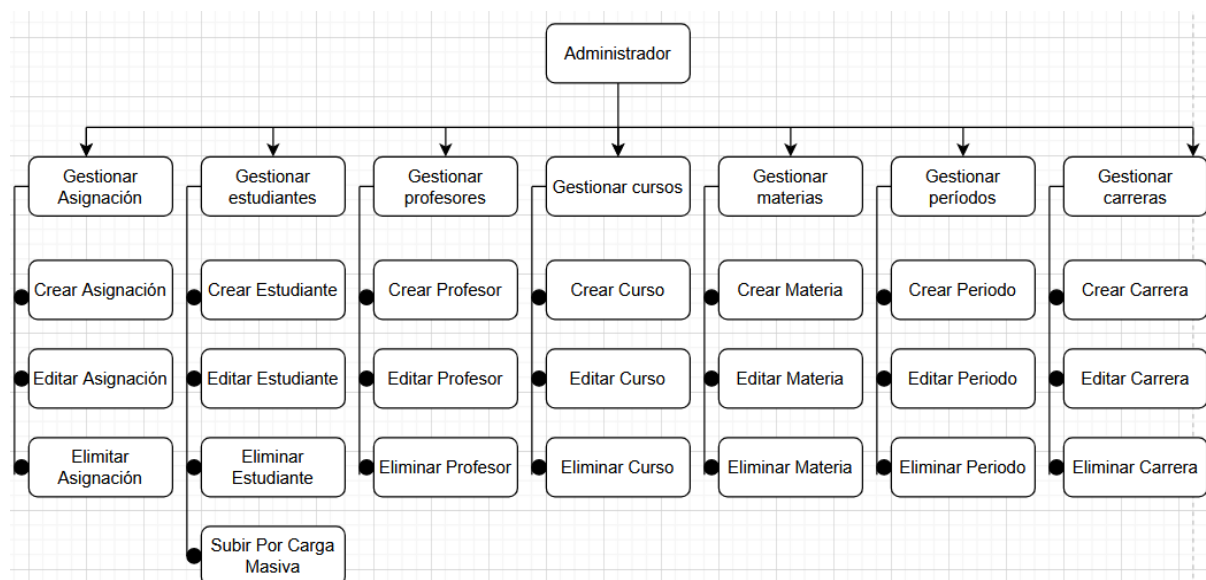
La tipografía fue elegida pensando en los usuarios finales, para garantizar una legibilidad alta tanto en interfaces de escritorio como para los dispositivos móviles, considerando que el usuario interactuará con tablas con muchos datos. Se ha definido el uso de las fuentes Roboto y Open Sans, las cuales son reconocidas por su diseño y su alta eficiencia en pantallas. Para establecer patrones visuales claros. En cuanto al tamaño se emplearán distintos tamaños: los títulos principales varían entre 24 px con negrita a 30 px, los subtítulos en 18 px con negrita, y el texto del cuerpo en 14 px o 12 px. Esta configuración tipográfica aporta una estética profesional y moderna al sistema.

#### 4.4.2.3 Pantallas

En este apartado se muestran los módulos que se manejan en cada rol (administrador, coordinador, profesor, estudiante) y consiguiente las acciones más relevantes de cada uno. Muchas de las acciones se realizan a través de formulación implementados en modales para disminuir el número de pantallas.

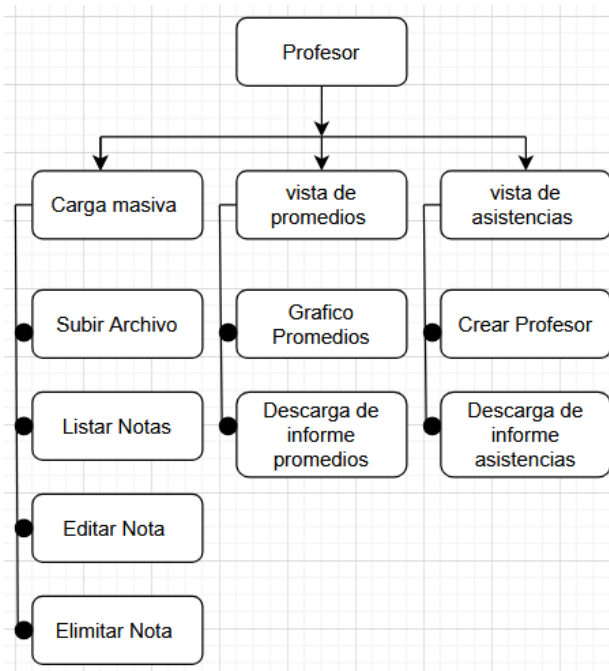
**Figura 15**

*Pantallas Administrador*



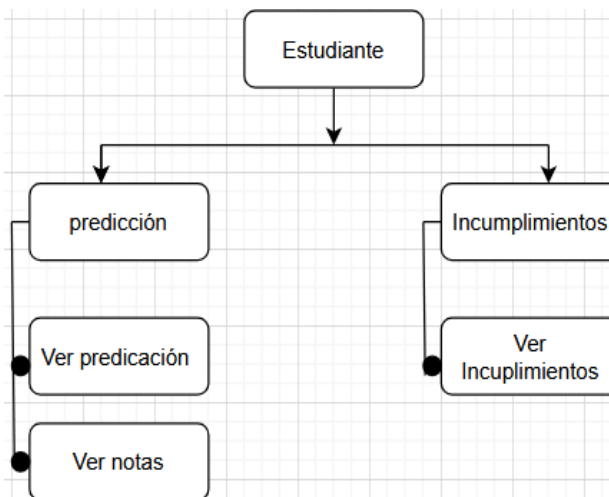
**Figura 16**

*Pantallas profesor*



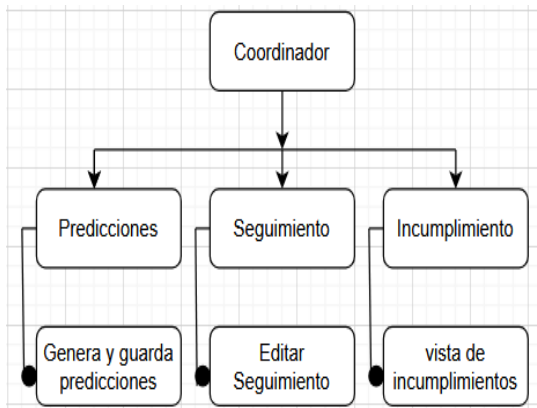
**Figura 17**

*Pantallas Coordinador*



**Figura 18**

*Pantallas estudiantas*

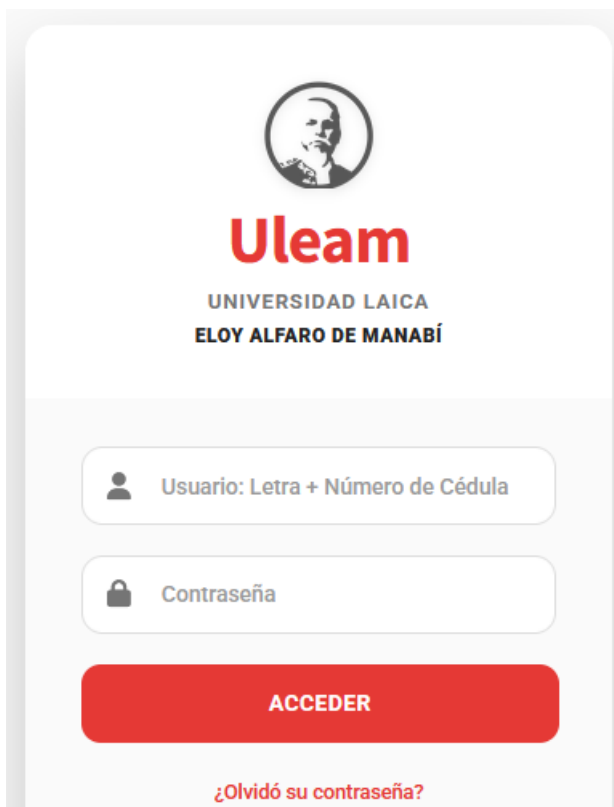


#### 4.4.2.3.1 Sistema Web

El sistema inicia con una pantalla para el inicio de sesión para que los usuarios puedan acceder al sistema según su rol.

**Figura 19**

*Login*

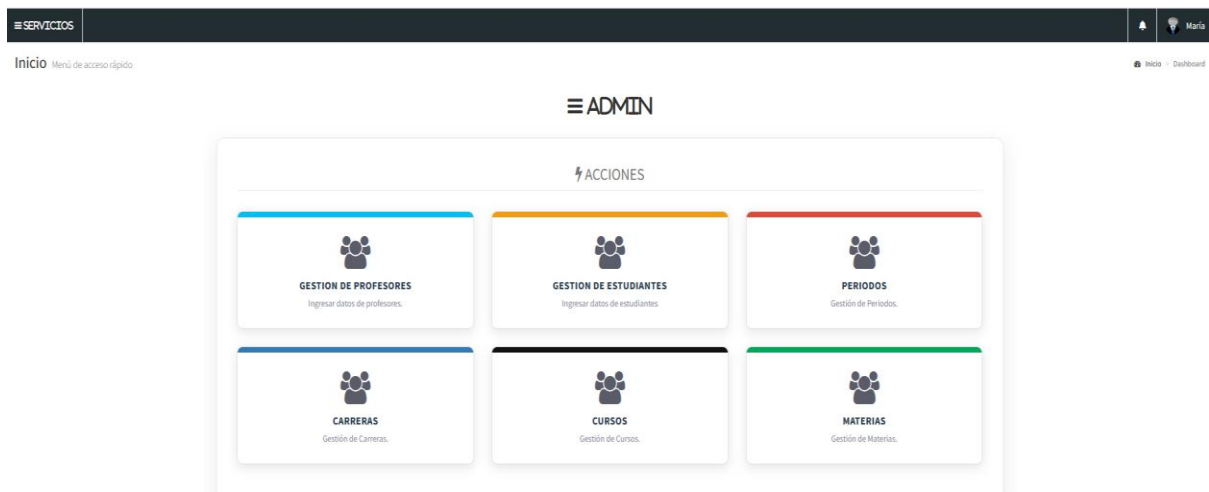


#### 4.4.2.3.1.1 Administrador

Entre las pantallas destacadas del rol de administrador encontramos el panel inicial con el menú de inicio con los diferentes módulos asignados a este rol. Lo mismo se mostrará a cada rol con sus respectivos módulos cada uno.

**Figura 20**

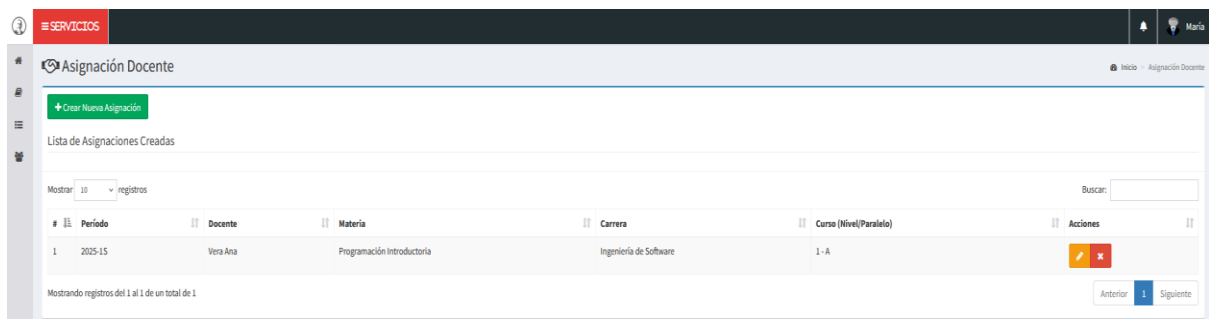
*Menú*



El diseño principal para los módulos del administrador es el siguiente ya que hace una gestión académica de cursos, materias, carreras y periodos.

**Figura 21**

*Asignación*



#### 4.4.2.3.2 Coordinador

Las pantallas más importantes del coordinador están conformadas por predicción, incumplimientos y seguimiento de incumplimientos.

## Figura 22

### Predicción

Resultados de Predicción

[Guardar Predicciones en Historial](#)

**Resultados de la Predicción Actual**  
Predicción de la nota final (Aprobado/Reprobado) para los estudiantes del curso seleccionado.

Mostrar: 10 registros Buscar:

#	Cédula	Estudiante	Materia	Predicción	Probabilidad	Acciones
1	1350207427	JUAN DAVID ALCIVAR TACURI	Programación Introdutoria	APROBADO	100%	<a href="#">Ver Detalle</a>
2	1351288640	JAMEL JOSUE ANCHUNDIA SEPA	Programación Introdutoria	REPROBADO	100%	<a href="#">Ver Detalle</a>
3	2300573454	AXEL ARIEL AVILA AVILA	Programación Introdutoria	APROBADO	100%	<a href="#">Ver Detalle</a>
4	1317774295	JOSE FERNANDO BRIONES MOREIRA	Programación Introdutoria	APROBADO	100%	<a href="#">Ver Detalle</a>
5	2300751589	JACSON ALEXANDER CAMPOVERDE CAMPOVERDE	Programación Introdutoria	APROBADO	100%	<a href="#">Ver Detalle</a>
6	1317551594	JOSE DAVID CEDEÑO CHAVEZ	Programación Introdutoria	APROBADO	100%	<a href="#">Ver Detalle</a>
7	1317872909	DEIVIS STEVEN CEDEÑO CHILA	Programación Introdutoria	APROBADO	100%	<a href="#">Ver Detalle</a>
8	0503393134	CRISTOPHER STALIN CHUBA ALVA	Programación Introdutoria	APROBADO	100%	<a href="#">Ver Detalle</a>
9	1313426429	LUIS MIGUEL ESTRADA VELEZ	Programación Introdutoria	APROBADO	100%	<a href="#">Ver Detalle</a>
10	1350169247	KAREN LUISIANA GARCIA NIJARRETE	Programación Introdutoria	APROBADO	100%	<a href="#">Ver Detalle</a>

Mostrando registros del 1 al 10 de un total de 25 Anterior 1 2 3 Siguiente

En el caso de los incumplimientos permite ver el número de incumplimientos por estudiante y al presionar el botón ver incumplimientos me deja ver los incumplimientos.

## Figura 23

### Incumplimientos

**Listado de Estudiantes del Curso/Materia Seleccionada**

Mostrar: 10 registros Buscar:

#	Cédula	Estudiante	Curso	Materia	Total Incumplimientos	Acciones
1	2300587983	ARIEL SEBASTIAN JUMBO RAMOS	1A	Programación Introdutoria	1	<a href="#">Ver Incumplimientos</a>
2	2300573454	AXEL ARIEL AVILA AVILA	1A	Programación Introdutoria	1	<a href="#">Ver Incumplimientos</a>
3	1316409976	CRISTHIAN JEAMPIERE RUIZ ZAMBRANO	1A	Programación Introdutoria	1	<a href="#">Ver Incumplimientos</a>
4	0503393134	CRISTOPHER STALIN CHUBA ALVA	1A	Programación Introdutoria	1	<a href="#">Ver Incumplimientos</a>
5	2350656647	DAVID JAHIR PARRAGA MITES	1A	Programación Introdutoria	1	<a href="#">Ver Incumplimientos</a>
6	1317872909	DEIVIS STEVEN CEDEÑO CHILA	1A	Programación Introdutoria	1	<a href="#">Ver Incumplimientos</a>
7	1313665729	EDWIN ARMANDO OSTAZA SOLORZANO	1A	Programación Introdutoria	1	<a href="#">Ver Incumplimientos</a>
8	2300012461	FRANCISCO JAVIER MOREIRA VILLACIS	1A	Programación Introdutoria	1	<a href="#">Ver Incumplimientos</a>
9	2300751589	JACSON ALEXANDER CAMPOVERDE CAMPOVERDE	1A	Programación Introdutoria	1	<a href="#">Ver Incumplimientos</a>
10	0958031999	JAIRO VINICIO PEÑAFIEL PALACIO	1A	Programación Introdutoria	1	<a href="#">Ver Incumplimientos</a>

Mostrando registros del 1 al 10 de un total de 25 Anterior 1 2 3 Siguiente

El módulo seguimiento permite tomar acciones sobre el estudiante y notificar al profesor y al estudiante por cualquier acción.

**Figura 24**  
Seguimiento

Seguimiento de Incumplimientos

1. Periodo: 2025-15 | 2. Carrera: Ingeniería de Software | 3. Curso: IA | 4. Materia: Programación Introdutoria

Mostrar: 10 registros | Buscar: \_\_\_\_\_

#	Estudiante	Tipo Incumplimiento	Fecha	Riesgo	Estado Actual	Acción Actual	Acciones
1	ALCNAR TACURI JUAN DAVID JUAN DAVID ALCNAR TACURI	BAJO_RENDIMIENTO	2026-01-03 23:01:45	BAJO	PENDIENTE	Sin acción	Ver Seguimiento
2	ALCNAR TACURI JUAN DAVID JUAN DAVID ALCNAR TACURI	BAJO_RENDIMIENTO	2026-01-03 23:01:45	BAJO	PENDIENTE	Sin acción	Ver Seguimiento
3	ALCNAR TACURI JUAN DAVID JUAN DAVID ALCNAR TACURI	BAJO_RENDIMIENTO	2026-01-03 23:01:45	BAJO	PENDIENTE	Sin acción	Ver Seguimiento
4	ALCNAR TACURI JUAN DAVID JUAN DAVID ALCNAR TACURI	BAJO_RENDIMIENTO	2026-01-03 23:01:45	BAJO	PENDIENTE	Sin acción	Ver Seguimiento
5	ALCNAR TACURI JUAN DAVID JUAN DAVID ALCNAR TACURI	BAJO_RENDIMIENTO	2026-01-03 23:01:45	BAJO	PENDIENTE	Sin acción	Ver Seguimiento
6	ALCNAR TACURI JUAN DAVID JUAN DAVID ALCNAR TACURI	BAJO_RENDIMIENTO	2026-01-03 23:01:45	BAJO	PENDIENTE	Sin acción	Ver Seguimiento
7	ANCHUNDIA SEPH JAMEL JOSUE JAMEL JOSUE ANCHUNDIA SEPA	BAJO_RENDIMIENTO	2026-01-03 23:01:46	BAJO	PENDIENTE	Sin acción	Ver Seguimiento
8	ANCHUNDIA SEPH JAMEL JOSUE JAMEL JOSUE ANCHUNDIA SEPA	BAJO_RENDIMIENTO	2026-01-03 23:01:46	BAJO	PENDIENTE	Sin acción	Ver Seguimiento
9	ANCHUNDIA SEPH JAMEL JOSUE JAMEL JOSUE ANCHUNDIA SEPA	BAJO_RENDIMIENTO	2026-01-03 23:01:46	BAJO	PENDIENTE	Sin acción	Ver Seguimiento
10	ANCHUNDIA SEPH JAMEL JOSUE JAMEL JOSUE ANCHUNDIA SEPA	BAJO_RENDIMIENTO	2026-01-03 23:01:46	BAJO	PENDIENTE	Sin acción	Ver Seguimiento

Mostrando registros del 1 al 10 de un total de 123

Anterior 1 2 3 4 5 ... 13 Siguiente

#### 4.4.2.3.3 Profesor

El profesor principalmente usa el módulo de carga masiva para subir las notas de los estudiantes con un Excel, Y listar las mismas y permitir editar, crear y eliminar. Al subir las notas se crean incumplimientos de tener notas vacías o bajas.

**Figura 25**  
Carga Masiva

Subir Notas Masivas (Excel Moodle)

Seleccione la materia para habilitar la carga masiva.

Mostrar: 10 registros | Buscar: \_\_\_\_\_

#	Cédula	Estudiante	HSP-Control de Lectura (Real)	HSP-Cuestionario Políticas y Sílabo de Asignatura	HSP-Cuestionario Resolución de problemas (Real)	HSP-Cuestionario Problemas Algoritmos y Jerarquía	HSP-Cuestionario Sentencias PSeInt (Real)
1	2300587983	ARIEL SEBASTIAN JUMBO RAMOS	10.00	10.00	10.00	10.00	10.00
2	2300573454	AXEL ARIEL AVILA AVILA	10.00	10.00	10.00	10.00	10.00
3	1316409976	CRISTHIAN JEAMPIERE RUIZ ZAMBRANO	+ Agregar	10.00	10.00	10.00	10.00
4	0503393134	CRISTOPHER STALIN CHUBA ALVIA	10.00	10.00	10.00	10.00	10.00
5	2350656647	DAVID JAHIR PARRAGA MITES	10.00	10.00	10.00	10.00	7.00
6	1317872909	DEMIS STEVEN CEDERO CHILA	10.00	10.00	10.00	10.00	10.00
7	1313665729	EDWIN ARMANDO OSTAIZA SOLORZANO	10.00	10.00	10.00	10.00	10.00
8	2300012461	FRANCISCO JAVIER MOREIRA VILLACIS	10.00	10.00	10.00	10.00	10.00
9	2300751589	JACSON ALEXANDER CAMPOVERDE CAMPOVERDE	10.00	10.00	10.00	10.00	10.00
10	0958031999	JAIRO VINICIO PEÑAFIEL PALACIO	10.00	10.00	10.00	10.00	9.00

Mostrando registros del 1 al 10 de un total de 25

Anterior 1 2 3 Siguiente

Además, el profesor puede ver los promedios y generar un gráfico para ver que estudiantes se encuentran en peligro y descargar en Word los promedios más bajos.



### 4.4.3 Fase 3: Implementación (Codificación)

#### 4.4.3.1 Herramientas de programación

**Tabla 13**

*Lenguajes para el desarrollo*

<b>Herramienta</b>	<b>Función</b>	<b>Tipo</b>
<b>PHP 8.1</b>	Conecta la base de datos con la interfaz	Lenguaje Frontend
<b>JS</b>	Da interacción a la página	Lenguaje Backend
<b>CSS</b>	Usado para editar el diseño de los objetos de la página	Lenguaje para el diseño
<b>PYTHON</b>	Python actúa como un servicio especializado del backend para procesar el Machine Learning	Lenguaje Frontend
<b>MYSQL</b>	La base de datos donde se almacena todo de forma permanente	Lenguaje de base de datos
<b>HTML</b>	Define la estructura de la página.	Lenguaje de etiquetas

**Tabla 14**

*Librerías usadas*

<b>Herramienta</b>	<b>Función</b>	<b>Tipo</b>
<b>Bootstrap (3.3.7)</b>	El framework de CSS y JS principal para el diseño de rejilla, botones, formularios y componentes modales.	Librería CSS y JS
<b>DataTables</b>	Permite convertir tablas HTML comunes en tablas dinámicas con funciones de búsqueda, paginado en tiempo real y ordenamiento de columnas.	Librería JS

<b>Herramienta</b>	<b>Función</b>	<b>Tipo</b>
<b>Chart.js (2.9.3)</b>	Utilizada para la generación de gráficos estadísticos	Librería JS
<b>Moment.js</b>	Una librería esencial para manipular, validar y dar formato a fechas y horas en JavaScript de forma sencilla.	Librería JS
<b>Font Awesome (6.5.0)</b>	Librería de iconos vectoriales utilizada para los símbolos del menú y botones de acción.	Librería HTML
<b>SweetAlert2</b>	Reemplaza las ventanas de alerta clásicas del navegador por ventanas emergentes elegantes	Librería JS
<b>Toastr</b>	Se usa para mostrar notificaciones flotantes no intrusivas	Librería JS
<b>Bootstrap Daterangepicker / DatePicker</b>	Permite al usuario seleccionar rangos de fechas o fechas individuales mediante un calendario desplegable en los formularios.	Librería JS
<b>jQuery</b>	Permite modificar el contenido de la página y base para manejar otras librerías basadas en js	Librería JS
<b>PhpSpreadsheet</b>	Permite leer y procesar archivos de Excel (.xlsx o .csv).	Librería PHP
<b>scikit-learn</b>	Es la librería principal de IA.	Librería PYTHON
<b>Joblib</b>	Permite guardar los modelos entrenados en archivos. pkl	Librería PYTHON
<b>Pandas</b>	Fundamental para el manejo de estructuras de datos.	Librería PYTHON
<b>NumPy</b>	Se utiliza para operaciones matemáticas y generación de datos aleatorios.	Librería PYTHON

**Tabla 15**

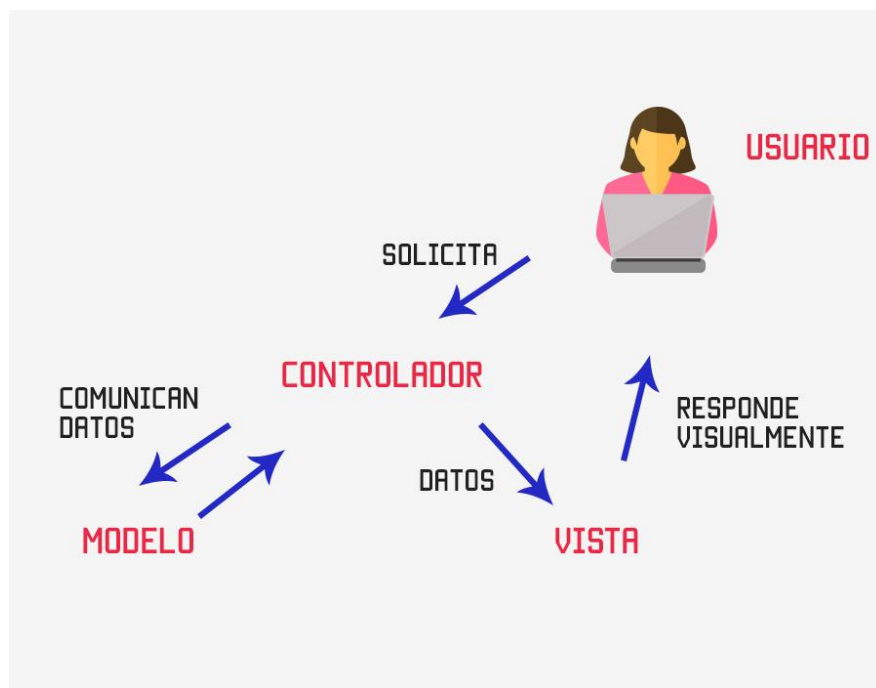
*Herramientas extras*

Herramienta	Función	Tipo
AdminLTE (2.4)	Es la plantilla base sobre la cual se construye toda la interfaz.	Plantilla
Visual Studio Code (1.108)	Editor de código fuente ligero, pero extremadamente potente desarrollado por Microsoft	Editor de código
Xampp (8.2.12)	Servidor web local	Servidor Local
PythonAnyware	Servidor web Python	Servidor web Python

#### 4.4.3.2 Arquitectura

**Figura 28**

*Patrón MVC. Extraída de: (Hernández U. , 2015)*



El sistema fue desarrollado bajo una arquitectura de software modular y escalable, todo bajo el patrón de desarrollo Modelo – Vista - Controlador o MVC. Este permite una separación entre la vista del usuario, el tratamiento de los datos y manejo de la base de datos, facilitando la actualización de los componentes. Para mejorar la experiencia del usuario, se implementa

AJAX, lo que garantiza dinamismo al usuario final, que evita recargas innecesarias de la página al realizar peticiones al servidor. El sistema también tiene un enfoque responsivo, asegurando su total funcionamiento tanto en computadoras como celulares. Además, para documentar este sistema, se utiliza un modelo de desarrollo de software en cascada para garantizar el cumplimiento secuencial de las etapas de requisitos, diseño, implementación y pruebas.

#### **4.4.3.3 Seguridad**

Para el tema de la seguridad del sistema se dio prioridad a la protección de los datos personales y la integridad de las notas de los estudiantes usando capas de defensa. En primer lugar, se implementa un mecanismo de encriptación y hashing para el guardar las contraseñas de los usuarios, asegurando que los datos de acceso de los usuarios finales no sean vulnerables. Para la él envío de datos por otro lado, se hace usando el método POST, así evitando que datos sensibles queden expuestos en las URLs del navegador y reduciendo el riesgo de ataques.

Además, como extra, el sistema cuenta con un manejo estricto de rutas y un control de acceso basado en roles. Esto garantiza que cada usuario según su rol (administrador, coordinador, profesor o estudiante) acceda exclusivamente a las funciones permitidas para su perfil, bloqueando cualquier navegación hacia rutas inexistentes o restringidas. Para prevenir amenazas comunes en aplicaciones web, se aplican validaciones en el lado del servidor también y técnicas para mitigar ataques SQL.

#### **4.4.3.4 Clases y Métodos**

##### **4.4.3.4.1 Clases**

- **Usuario:** Se encarga de gestionar la información personal y los datos de acceso de todos los usuarios que interactúan con el sistema (administrador, docente, estudiante o coordinador).
- **Periodo:** Gestiona los intervalos de tiempo académicos. Es fundamental para organizar el historial académico y las vigencias de las materias.
- **Carrera:** Almacena la información de las carreras ofertadas por la institución,
- **Curso:** Define los grupos específicos que vinculan un nivel con una carrera y periodo determinado.
- **Materia:** Se encarga de la gestión de las unidades de aprendizaje o asignaturas que forman parte del plan de estudios de las diferentes carreras.

- **Predicción:** Clase encargada de procesar los datos predictivos generados a partir de variables seleccionadas (promedio y asistencia) mediante algoritmos de Árboles de Decisión para determinar el riesgo de reprobar.
- **Notas:** Gestiona los registros de las calificaciones obtenidas por los estudiantes en las diferentes evaluaciones.
- **Incumplimiento:** Se utiliza para registrar las faltas académicas o de asistencia.
- **Seguimiento:** Clase orientada a la supervisión del desempeño del estudiante, permite tomar acciones sobre los incumplimientos.
- **Asignación (docente\_curso\_materia):** Actúa como la entidad de relación que vincula a un docente con una materia y un curso en un periodo determinado.

#### 4.4.3.4.2 Métodos

**Tabla 16**

*Métodos*

<b>Clase</b>	<b>Método</b>	<b>Descripción</b>
<b>Usuario</b>	ctrEditarUsuarioPerfil()	Gestiona los usuarios. Permite la autenticación al sistema, la actualización de perfil y la creación de cuentas según el rol (docente o estudiante) con sus respectivos privilegios.
	ctrIngresoUsuario()	
	ctrBorrarUsuario()	
	ctrCrearUsuarioProfesor()	
	ctrCrearUsuarioEstudiante()	
<b>Periodo</b>	ctrEditarPeriodo()	Administra los lapsos de tiempo académicos. Se encarga de registrar, modificar y eliminar los semestres.
	ctrIngresoPeriodo()	
	ctrBorrarPeriodo()	
<b>Carrera</b>	ctrEditarCarrera()	Controla la información de los programas de estudio. Gestiona el nombre, código y facultades de las carreras profesionales.
	ctrIngresoCarrera()	
	ctrBorrarCarrera()	
<b>Curso</b>	ctrEditarCurso()	Gestiona la creación de niveles y paralelos. Permite organizar físicamente a los alumnos dentro de una carrera y un periodo.
	ctrIngresoCurso()	
	ctrBorrarCurso()	

<b>Clase</b>	<b>Método</b>	<b>Descripción</b>
<b>Materia</b>	ctrEditarMateria()	Los métodos gestionan las materias desde su ingreso, alguna modificación y su eliminación de la base de datos.
	ctrIngresoMateria()	
	ctrBorrarMateria()	
<b>Notas</b>	ctrEditarNotas()	Encargados del registro masivo de calificaciones desde un archivo .CSV o Excel. Además, permite la gestión de las mismas notas subidas y su remplazo total de ser subidas nuevamente.
	ctrIngresoNotas()	
	ctrEliminarNotas()	
	ctrSubirNotasMasivoDinamico() ()	
<b>Predicción</b>	ctrEjecutarPrediccionAPI() ctrMostrarDetallePrediccion()	Clase encargada generar la predicción haciendo uso de los algoritmos C4.5 y Random Forest. Envía los datos académicos del curso, a la API de Python para que se encargue de valorar el riesgo y mostrar los resultados analíticos por curso y estudiante.
	ctrEditarIncumplimiento() ctrIngresoIncumplimiento () TriggerRegistrarIncumplimientos	Lista los incumplimientos de cada estudiante o por curso, y el hilo por su parte registra automáticamente luego de insertar notas en la tabla de notas, y de cumplir como baja, crea un registro y crea además su notificación.
<b>Seguimiento</b>	ctrEditarSeguimiento() ctrIngresoSeguimiento()	Registra las acciones tomadas por el coordinador y notifica a docentes, mientras que a los estudiantes los notifica tutorías.
<b>Asignación (docente_curso_materia)</b>	ctrEditarAsignacion() ctrIngresoAsignacion() ctrBorrarAsignacion()	Gestiona la carga horaria docente. Establece la relación lógica entre un profesor, una materia y un curso determinado para un periodo vigente.

### 4.4.3.5 Codificación

#### 4.4.3.5.1 Login

El método `ctrIngresoUsuario()` se encarga de recibir las credenciales por POST (usuario, contraseña) luego consulta si ese usuario existe y si la contraseña es válida, luego se valida si la cuenta esta activada y de todo ser correcto registra el ingreso y direcciona a la página de inicio que le pertenece a ese usuario según su rol.

Figura 29

Código Login 1

```
static public function ctrIngresoUsuario(){
if(isset($_POST["ingUsuario"])){
    // 1. Validación de caracteres permitidos
    if(preg_match('/^[a-zA-Z0-9]+$/ ', $_POST["ingUsuario"]) &&
        preg_match('/^[a-zA-Z0-9]+$/ ', $_POST["ingPassword"])){
        $tabla = "usuarios";
        $item = "USU_USU";
        $valor = $_POST["ingUsuario"];

        $respuesta = ModeloUsuarios::MdlMostrarUsuarios($tabla, $item, $valor);

        // 2. Verificación de credenciales con password_verify
        if ($respuesta && password_verify($_POST["ingPassword"], $respuesta["PASS_USU"])) {

            // 3. Verificación de estado de cuenta
            if($respuesta["EST_USU"] != 1){
                echo '<br><div class="alert alert-warning">Cuenta desactivada. Contacte soporte.</div>';
                return;
            }

            // 4. Manejo de Intentos / Caducidad
            $intentos = $respuesta["ING_USU"];
            $maximoIntentos = 4000;

            $intentos++;
            $datosUpdate = array("usuarioActual" => $respuesta["USU_USU"], "actualizarIntentos" => $intentos);
            ModeloUsuarios::intentosUsuarioModel($datosUpdate, "usuarios");

            if($intentos >= $maximoIntentos){
                echo '<script>>window.location = "index.php?ruta=actualizarContrasena&usuario='.$valor.'";</script>';
                return;
            }
        }
    }
}
```

Además, guarda los datos más importantes del usuario en variables de sesión, para usarlas más adelante y evitar constantes consultas de datos a la base de datos.

Figura 30

Código Login 2

```
switch ($respuesta["TIP_USU"]) {  
  
    case "ADMIN":  
        $_SESSION['ACCESO'] = "1";  
        $_SESSION["CEDULA"] = $respuesta["CED_USU"];  
        $_SESSION["EMAIL"] = $respuesta["EMA_USU"];  
        $_SESSION["PASS"] = $respuesta["PASS_USU"]; // El Admin usualmente  
        break;  
  
    case "COORDINADOR":  
        $_SESSION['ACCESO'] = "2";  
        $_SESSION["CEDULA"] = $respuesta["CED_USU"];  
        $_SESSION["EMAIL"] = $respuesta["EMA_USU"];  
        $_SESSION["PASS"] = $respuesta["PASS_USU"];  
        break;  
  
    case "PROFESOR":  
        $_SESSION['ACCESO'] = "3";  
        $_SESSION["CEDULA"] = $respuesta["CED_USU"];  
        $_SESSION["EMAIL"] = $respuesta["EMA_USU"];  
        $_SESSION["PASS"] = $respuesta["PASS_USU"];  
        break;  
  
    case "ESTUDIANTE":  
        $_SESSION['ACCESO'] = "4";  
        $_SESSION["CEDULA"] = $respuesta["CED_USU"];  
        $_SESSION["EMAIL"] = $respuesta["EMA_USU"];  
        $_SESSION["PASS"] = $respuesta["PASS_USU"];  
        break;  
  
    default:  
        echo '<br><div class="alert alert-danger">Rol no reconocido.</div>';  
        return;  
}
```

#### 4.4.3.5.2 Predicción Python

Para generar la predicción se genera primero un archivo de 1000 datos para luego con esos datos entrenar el modelo de los dos algoritmos seleccionados como se muestra en el siguiente código, para hacerlo usamos Sklearn, librería de Python para IA y otras como pandas y jolib.

Figura 31

Código Python Predicción 1

```
import pandas as pd
from sklearn.ensemble import RandomForestClassifier
from sklearn.tree import DecisionTreeClassifier
import joblib

# Cargar tus datos (sintéticos o reales)
df = pd.read_csv('datos_entrenamiento_balanceado_1000.csv')

X = df[['promedio_notas', 'porcentaje_asistencia']]
y = df['resultado']

print("Distribución en entrenamiento:")
print(y.value_counts())

# Entrenar Random Forest (mejor precisión)
rf = RandomForestClassifier(n_estimators=200, random_state=42)
rf.fit(X, y)
joblib.dump(rf, 'modelo_randomforest.pkl')

# Entrenar C4.5 (Árbol de decisión con entropía)
c45 = DecisionTreeClassifier(criterion='entropy', max_depth=5, random_state=42)
c45.fit(X, y)
joblib.dump(c45, 'modelo_c45.pkl')

print("¡Modelos reentrenados y guardados con datos mejorados!")
```

Una vez obtenidos los archivos PKL se los sube al servidor y el archivo main.py que es el siguiente se encarga de hacer la predicción partido de los algoritmos seleccionados y los datos enviados desde la vista y se valida con las reglas expuestas. Para luego dar respuesta al sistema.

Figura 32

Código Python Predicción 2

```
from flask import Flask, request, jsonify
import joblib
import numpy as np
import os

app = Flask(__name__)

# Directorio donde están tus modelos .pkl
SCRIPT_DIR = os.path.dirname(os.path.abspath(__file__))

def obtener_regla(promedio, asistencia):
    if asistencia < 70:
        return "Asistencia < 70% -> REPROBAR"
    elif promedio >= 70 and asistencia >= 70:
        return "Promedio >= 70 y Asistencia >= 70% -> APROBAR"
    else:
        return "Cumple asistencia pero Promedio < 70 -> RIESGO"

@app.route('/predict', methods=['POST'])
def predict():
    try:
        data = request.get_json()

        # Extraer datos que envía PHP
        algoritmo = data.get('algoritmo', 'RandomForest')
        promedio = float(data.get('promedio', 0))
        asistencia = float(data.get('asistencia', 0))
```

**Figura 33**

*Código Python Predicción 3*

```
# Seleccionar modelo
if 'C4.5' in algoritmo or 'J48' in algoritmo:
    model_name = 'modelo_c45.pkl'
else:
    model_name = 'modelo_randomforest.pkl'

model_path = os.path.join(SCRIPD_DIR, model_name)
model = joblib.load(model_path)

# Predicción
X = np.array([[promedio, asistencia]])
pred_label = model.predict(X)[0]

# Probabilidad
probas = model.predict_proba(X)[0]
class_index = list(model.classes_).index(pred_label)
probabilidad = round(probas[class_index] * 100, 2)

return jsonify({
    "status": "success",
    "prediccion": pred_label,
    "probabilidad": probabilidad,
    "regla": obtener_regla(promedio, asistencia)
})

except Exception as e:
    return jsonify({"status": "error", "message": str(e)}), 400

# Esta parte es para que funcione en PythonAnywhere
if __name__ == '__main__':
    app.run()
```

*4.4.3.5.3 Predicción PHP*

En el método `ctrEjecutarPrediccionAPI()` se reciben los datos necesarios para consultas internas y para la predicción. Primero se consulta las predicciones guardadas si existen, para proceder a mostrarlas. Luego mediante `url_api` se establece la conexión del API que nos devuelve la predicción, el mismo ya configurado antes en el servidor. Luego se prepara los datos que se va enviar al API y se envían en formato Json.

**Figura 34**

*Código PHP predicción 1*

```
static public function ctrEjecutarPrediccionAPI($id_periodo, $id_curso, $id_materia, $id_tipo_nota, $id_tipo_asistencia, $algoritmo)
{
    // 1. Extraer datos de la BD
    $datos_raw = ModeloPredicciones::mdlExtraerDatosExcelParaPrediccion($id_periodo, $id_curso, $id_materia, $id_tipo_nota, $id_t

    if (!$datos_raw || count($datos_raw) == 0) {
        return [
            "error" => true,
            "mensaje" => "No se encontraron calificaciones registradas para los parámetros seleccionados."
        ];
    }

    $resultados_prediccion = [];

    // --- NUEVA CONFIGURACIÓN DE API ---
    // Reemplaza 'tu_usuario' por tu nombre de usuario de PythonAnywhere
    $url_api = "http://Karen14ml.pythonanywhere.com/predict";

    foreach ($datos_raw as $estudiante) {

        $promedio_notas = ($estudiante['nota_promedio'] != null) ? round((float)$estudiante['nota_promedio'], 2) : 0;
        $promedio_notas = $promedio_notas * 10; // Convertir escala 10 a 100
        $porcentaje_asistencia = ($estudiante['asistencia_promedio'] != null) ? round((float)$estudiante['asistencia_promedio'],

        // Preparar datos para la API
        $input_data = [
            'algoritmo' => $algoritmo_seleccionado,
            'promedio' => $promedio_notas, // Cambiado para coincidir con el nombre en la API Flask
            'asistencia' => $porcentaje_asistencia
        ];
    }
}
```

Luego la recupero la respuesta del servidor y de existir un error lo informa, luego las respuestas las pasamos por el estándar de calificaciones y asistencias.

**Figura 35**

*Código PHP predicción 2*

```

$ch = curl_init($url_api);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($input_data));
curl_setopt($ch, CURLOPT_HTTPHEADER, ['Content-Type: application/json']);
curl_setopt($ch, CURLOPT_TIMEOUT, 10); // Tiempo limite de espera

$json_output = curl_exec($ch);
$http_code = curl_getinfo($ch, CURLINFO_HTTP_CODE);
curl_close($ch);
$python_result = json_decode(trim($json_output), true);

// == VALIDACIÓN DE LA PREDICCIÓN ==
$prediccion = 'ERROR_PREDICCIÓN';
$confiianza = 0;
$regla_aplicada = 'Fallo al conectar con la API';

if ($http_code == 200 && is_array($python_result) && isset($python_result['status']) && $python_result['status'] == 'success')
{
    $pred_raw = strtoupper(trim($python_result['prediccion']));

    // Estandarizar respuesta
    if ($pred_raw == 'APROBAR' || $pred_raw == 'APROBADO') {
        $prediccion = 'APROBADO';
    } elseif ($pred_raw == 'REPROBAR' || $pred_raw == 'REPROBADO') {
        $prediccion = 'REPROBADO';
    } elseif ($pred_raw == 'RIESGO') {
        $prediccion = 'RIESGO';
    } else {
        $prediccion = $pred_raw;
    }

    $confiianza = $python_result['probabilidad'];
    $regla_aplicada = $python_result['regla'];
} else {

```

Para finalmente enviar todos los datos recibidos a la vista a través de un Json.

**Figura 36**

*Código PHP predicción 3*

```

} else {
    $regla_aplicada = 'Error en API: HTTP ' . $http_code;
}

if ($prediccion != 'ERROR_PREDICCIÓN') {
    $variables_usadas = json_encode([
        'id_tipo_nota' => $id_tipo_nota,
        'id_tipo_asistencia' => $id_tipo_asistencia,
        'valor_nota' => $promedio_notas,
        'valor_asistencia' => $porcentaje_asistencia,
        'regla_aplicada' => $regla_aplicada
    ], JSON_UNESCAPED_UNICODE);

    $resultados_prediccion[] = [
        'id_estudiante' => $estudiante['id_estudiante'],
        'cedula_estudiante' => $estudiante['cedula_estudiante'],
        'nombre_estudiante' => $estudiante['nombre_estudiante'],
        'nombre_materia' => $estudiante['nombre_materia'] ?? 'N/A',
        'algoritmo_usado' => $algoritmo_seleccionado,
        'prediccion_final' => $prediccion,
        'probabilidad' => $confiianza,
        'regla_aplicada' => $regla_aplicada,
        'promedio_notas_parciales' => $promedio_notas,
        'porcentaje_asistencia' => $porcentaje_asistencia,
        'nota_final_real' => $estudiante['nota_final_real'] ?? 0,
        'variables_usadas' => $variables_usadas,
        'id_asignacion' => $estudiante['id_asignacion'] ?? null,
        'id_periodo' => $id_periodo,
        'id_curso' => $id_curso,
        'id_materia' => $id_materia
    ];
}
}

```

#### 4.4.3.5.4 Carga Masiva

En el método ctrSubirNotasMasivoDinamico() en este método primero se valida que los filtros y el archivo se hayan seleccionado y subido. Después se empieza la lectura del documento las respectivas filas y columnas. Y la limpieza de los datos anteriores de la base de datos porque van a ser remplazados por los nuevos, como se planteó.

**Figura 37**

*Código Carga Masiva 1*

```
static public function ctrSubirNotasMasivoDinamico()
{
    if (isset($_POST["btnSubirNotasMasivoDinamico"])) {

        $id_asignacion = $_POST["id_asignacion_masiva"] ?? null;

        if (empty($id_asignacion)) {
            echo '<script>swal("Error", "No se recibió la materia seleccionada.", "error");</script>';
            return;
        }

        if (isset($_FILES["archivoNotas"]) || empty($_FILES["archivoNotas"]["tmp_name"])) {
            echo '<script>swal("Error", "Debe seleccionar un archivo Excel.", "error");</script>';
            return;
        }

        try {
            $rutaArchivo = $_FILES["archivoNotas"]["tmp_name"];
            $spreadsheet = IOFactory::load($rutaArchivo);
            $sheet = $spreadsheet->getActiveSheet();
            $highestRow = $sheet->getHighestRow();
            $highestColumn = $sheet->getHighestColumn();

            // LIMPIAR notas y tipos de esta asignación
            $limpieza = ModeloCargaMasiva::mdlLimpiarNotasYTiposPrevios($id_asignacion);
            if ($limpieza != "ok") {
                throw new Exception("Error limpiando datos previos: " . $limpieza);
            }

            // Obtener curso
            $id_curso = ModeloCargaMasiva::mdlObtenerIdCursoPorAsignacion($id_asignacion);
            if (!$id_curso) {
                throw new Exception("No se encontró el curso para esta materia.");
            }
        }
    }
}
```

Con los datos recibidos consultamos la asignación, y procedemos a insertar primero los estudiantes de no existir y generar su matrícula automática a ese curso.

**Figura 38**

*Código Carga Masiva 2*

```
$insertados = 0;
$errores = 0;
$mapa_tipos = [];

for ($row = 2; $row <= $highestRow; $row++) {

    $nombres = trim($sheet->getCell('A' . $row)->getValue() ?? '');
    $apellidos = trim($sheet->getCell('B' . $row)->getValue() ?? '');
    $email = trim($sheet->getCell('C' . $row)->getValue() ?? '');

    if (empty($nombres) || empty($apellidos) || empty($email)) {
        $errores++;
        continue;
    }

    if (!preg_match('/^e(\d+)@/', $email, $matches)) {
        $errores++;
        continue;
    }
    $cedula = $matches[1];

    // Crear/obtener estudiante
    $id_estudiante = ModeloCargaMasiva::mdlCrearEstudianteSiNoExiste($cedula, $nombres, $apellidos, $email);
    if (!$id_estudiante) {
        $errores++;
        continue;
    }

    // Matricular
    ModeloCargaMasiva::mdlMatricularEstudianteEncurso($id_estudiante, $id_curso);

    // Procesar evaluaciones
    $colIndex = 6;
    $maxCol = \PhpOffice\PhpSpreadsheet\Cell\Coordinate::columnIndexFromString($highestColumn);
```

Finalmente vamos a insertar los tipos de evaluación primero y después las notas de cada estudiante por cada tipo de evaluación registrado. O remplazar en caso ya haya sido subido un documento anteriormente.

**Figura 39**

*Código Carga Masiva 3*

```

for (; $colIndex < $maxCol; $colIndex++) {
    $colLetter = \PhpOffice\PhpSpreadsheet\Cell\Coordinate::stringFromColumnIndex($colIndex + 1);
    $nombre_tipo = trim($sheet->getCell($colLetter . '1')->getValue() ?? '');
    if (empty($nombre_tipo)) continue;
    // Crear tipo especifico para esta asignación
    if (isset($mapa_tipos[$nombre_tipo])) {
        $id_tipo = ModeloCargaMasiva::mdlObtenerOCrearTipoEvaluacion($nombre_tipo, $id_asignacion);
        if (!$id_tipo) throw new Exception("Error creando tipo: $nombre_tipo");
        $mapa_tipos[$nombre_tipo] = $id_tipo;
    }
    $id_tipo = $mapa_tipos[$nombre_tipo];
    $valor = $sheet->getCell($colLetter . $row)->getValue();

    if (is_numeric($valor)) {
        $nota = floatval($valor);
        if ($nota > 10) $nota = 10;
        if ($nota < 0) $nota = 0;

        $pdo = Conexion::conectar();
        $stmt = $pdo->prepare("
            REPLACE INTO notas
            (id_estudiante, id_asignacion, id_tipo_evaluacion, nota, fecha_registro)
            VALUES (:id_est, :id_asig, :id_tipo, :nota, NOW())
        ");
        $stmt->bindParam(":id_est", $id_estudiante, PDO::PARAM_INT);
        $stmt->bindParam(":id_asig", $id_asignacion, PDO::PARAM_INT);
        $stmt->bindParam(":id_tipo", $id_tipo, PDO::PARAM_INT);
        $stmt->bindParam(":nota", $nota, PDO::PARAM_STR);

        if ($stmt->execute()) {
            $insertados++;
        } else {
            $errores++;
        }
    }
}

```

#### 4.4.4 Fase 4: Pruebas

##### 4.4.4.1 Pruebas de Datos en frio

###### 4.4.4.1.1 Login

**Tabla 17**

*Pruebas de Datos en frio login*

Objeto	Tipo de objeto	Comportamiento	Observación
Usuario	Caja de texto	Ingresa datos alfanuméricos	Funciona Bien
Contraseña	Caja de texto	Ingresa datos alfanuméricos	Necesita una opción de ver contraseña.
Botón Ingresar	Botón	Click para ingresar	Ingresa y muestra error correctamente

#### 4.4.4.1.2 Registrar Docente/Estudiante

**Tabla 18**

*Pruebas de Datos en frío Registro Docente*

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Comportamiento</b>	<b>Observación</b>
Nombre	Caja de texto	Ingresa datos alfabéticos	No existe un límite (min-max) en la caja de texto.
Apellido	Caja de texto	Ingresa datos alfabéticos	No existe un límite (min-max) en la caja de texto.
Contraseña	Caja de texto	Ingresa datos alfanuméricos	Funciona bien
Correo	Caja de texto	Ingresa datos alfanuméricos	Funciona bien
Usuario	Caja de texto	Ingresa datos alfanuméricos	No existe un límite (min-max) en la caja de texto.
Género	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Botón Registrar	Botón	Click para Registrar	Muestra mensaje correctamente y registra

#### 4.4.4.1.3 Registrar Materia

**Tabla 19**

*Pruebas de Datos en frío Registro materia*

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Comportamiento</b>	<b>Observación</b>
Nombre	Caja de texto	Ingresa datos alfabéticos	No existe un límite (min-max) en la caja de texto.
Código	Caja de texto	Ingresa valores enteros	No existe un límite (min-max) en la caja de texto.
Créditos	Caja de texto	Ingresa valores enteros	Funciona bien

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Comportamiento</b>	<b>Observación</b>
Horas teóricas	Caja de texto	Ingresa valores enteros	Evitar que tome valores muy altos
Horas Practicas	Caja de texto	Ingresa valores enteros	Evitar que tome valores muy altos
Carrera	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Tipo	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Botón Registrar	Botón	Click para Registrar	Registra bien y notifica correctamente

#### 4.4.4.1.4 Registrar asignación

**Tabla 20**

*Pruebas de Datos en frio Registro Asignación*

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Comportamiento</b>	<b>Observación</b>
Periodo	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Carrera	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Curso	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Materia	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Docente	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien, pero debe filtrar solo docentes de la carrera
Botón Asignar	Botón	Click para Asignar	Asigna correctamente y notifica

#### 4.4.4.1.5 Carga Masiva

**Tabla 21**

*Pruebas de Datos en frio Registro Masivo*

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Comportamiento</b>	<b>Observación</b>
Periodo	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Carrera	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Curso	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Materia	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Subir Archivo	Caja de Selección de archivos	Seleccionar un objeto del escritorio	El formato debería estar mejor explicado
Botón Cargar	Botón	Click para Cargar	Carga bien y muestra cuantos datos se cargaron.

#### 4.4.4.1.6 Predicción

**Tabla 22**

*Pruebas de Datos en frio Predicción*

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Comportamiento</b>	<b>Observación</b>
Periodo	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Carrera	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Curso	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Comportamiento</b>	<b>Observación</b>
Materia	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Promedio	Caja de Selección	Seleccionar un objeto de una lista desplegable	Debe solo mostrar el campo de promedio.
Asistencia	Caja de Selección	Seleccionar un objeto de una lista desplegable	Debe sol mostrar el campo asistencia.
Algoritmo	Caja de Selección	Seleccionar un objeto de una lista desplegable	Funciona bien
Botón Predicción	Botón	Click para generar	Predice y muestra la predicción bien

#### 4.4.4.2 Pruebas de Datos reales

##### 4.4.4.2.1 Login

**Tabla 23**

*Pruebas de Datos Reales Login*

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Observación</b>
Usuario	Caja de texto	Funciona Bien
Contraseña	Caja de texto cifrada	La contraseña no se muestra al escribir
Botón Ingresar	Botón	Ingresar al sistema de ser correctas las credenciales y de no da mensajes de error

##### 4.4.4.2.2 Registrar Docente

**Tabla 24**

*Pruebas de Datos Reales Registro Docente*

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Observación</b>
Nombre	Caja de texto	Funciona bien
Apellido	Caja de texto	Funciona bien
Contraseña	Caja de texto	Tiene un mínimo y funciona bien
Correo	Caja de texto	Funciona bien
Usuario	Caja de texto	Funciona Bien

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Observación</b>
Género	Caja de Selección	Funciona Correctamente
Botón Registrar	Botón	Registra al docente / estudiante y lo muestra en la lista

#### 4.4.4.2.3 Registrar Materia

**Tabla 25**

*Pruebas de Datos Reales Registro Materia*

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Observación</b>
Nombre	Caja de texto	Funciona bien
Código	Caja de texto	Funciona bien
Créditos	Caja de texto	Funciona bien
Horas teóricas	Caja de texto	Funciona bien
Horas Practicas	Caja de texto	Funciona bien
Carrera	Caja de Selección	Funciona correctamente
Tipo	Caja de Selección	Funciona correctamente
Botón Registrar	Botón	Registra y muestra en la lista materias.

#### 4.4.4.2.4 Registrar asignación

**Tabla 26**

*Pruebas de Datos Reales Asignación*

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Observación</b>
Periodo	Caja de Selección	Funciona Correctamente
Carrera	Caja de Selección	Funciona Correctamente
Curso	Caja de Selección	Funciona Correctamente
Materia	Caja de Selección	Funciona Correctamente
Docente	Caja de Selección	Funciona Correctamente
Botón Asignar	Botón	Guarda y muestra la asignación

#### 4.4.4.2.5 Carga Masiva

**Tabla 27**

*Pruebas de Datos Reales Carga Masiva*

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Observación</b>
Periodo	Caja de Selección	Funciona Correctamente
Carrera	Caja de Selección	Funciona Correctamente
Curso	Caja de Selección	Funciona Correctamente
Materia	Caja de Selección	Funciona Correctamente
Subir Archivo	Caja de subida de archivos	Sube y muestra el archivo subido
Botón Cargar	Botón	Guarda, reemplaza y muestra las notas

#### 4.4.4.2.6 Predicción

**Tabla 28**

*Pruebas de Datos Reales Predicción*

<b>Objeto</b>	<b>Tipo de objeto</b>	<b>Observación</b>
Periodo	Caja de Selección	Funciona Correctamente
Carrera	Caja de Selección	Funciona Correctamente
Curso	Caja de Selección	Funciona Correctamente
Materia	Caja de Selección	Funciona Correctamente
Promedio	Caja de Selección	Funciona Correctamente
Asistencia	Caja de Selección	Funciona Correctamente
Algoritmo	Caja de Selección	Funciona Correctamente
Botón Predicción	Botón	Genera la Predicción y la muestra

### 4.4.5 Fase 5: Mantenimiento

#### 4.4.5.1 manual de usuario

##### 4.4.5.1.1 Servidor PythonAnywhere

Creemos un servidor para que se encargue de responder a las predicciones del sistema y lo haremos en la página: <https://www.pythonanywhere.com> y usaremos Flask que la proporciona la misma página y que proporciona las herramientas esenciales para crear aplicaciones web que puedan ser consumidas o en pocas palabras un API.

**Figura 40**

*Flash y Pythonanywhere*



Primero vamos a iniciar sesión o crear una cuenta de ser primera vez.

**Figura 41**

*Crear cuenta Pythonanyware*

**Figura 42**

*Dashboard Pythonanyware*

**Paso 1:** Crear un servicio WEB, para ello vamos a ir al apartado web:

**Figura 43**

*Menú Pythonanyware*

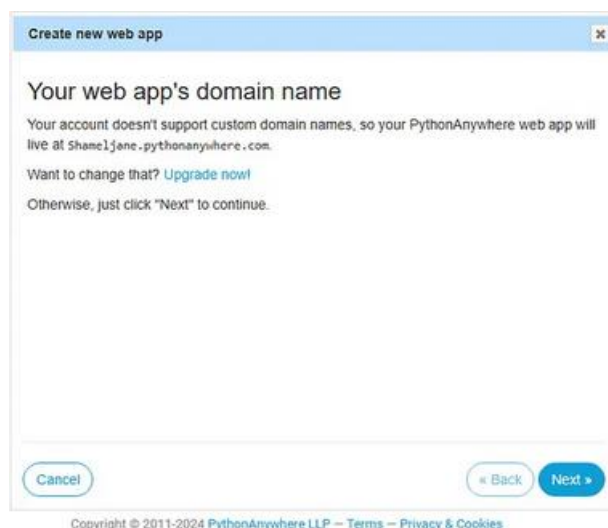
[Dashboard](#) [Consoles](#) [Files](#) **Web** [Tasks](#) [Databases](#)

**Paso 2:** Haz clic en ese botón "Add a new web app".

**Paso 3:** Te saldrá un mensaje sobre dominios (puntos extra), solo dale a Next.

**Figura 44**

*Mensaje de Dominios*



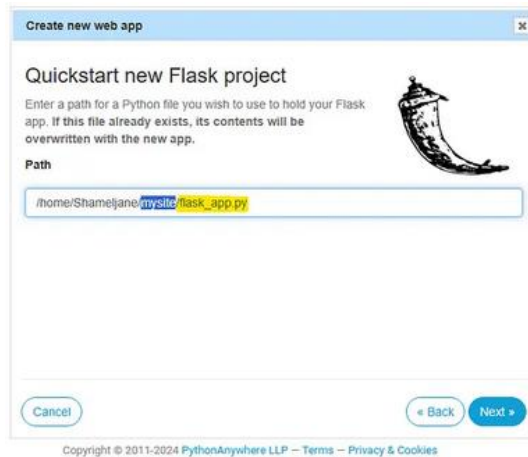
**Paso 4:** Selecciona Flask como tú "Python Web Framework".

**Paso 5:** Selecciona Python 3.10 (o la versión más alta que te salga).

**Paso 6:** Te preguntará por la ruta del archivo ("Path to a Flask file"). Borra lo que salga ahí y escribe esto exactamente: /home/tu\_usuario/mi\_proyecto/main.py (Cambia tu\_usuario por tu nombre de usuario real).

**Figura 45**

*Crear Proyecto Web Flask*



**Paso 7:** Dale a Next.

**Paso 8:** Luego ve a instalar en consola (en el menú hay una opción console) las librerías joblib, pandas y scikit-learn(copia y pega el siguiente comando en consola):

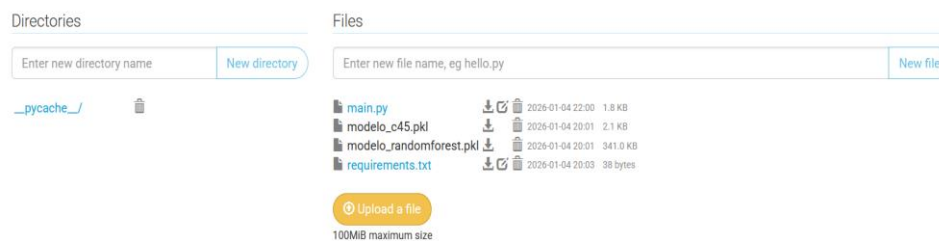
**`pip3 install --user flask joblib scikit-learn pandas numpy`**

**Paso 9:** Espera que termine de instalar

**Paso 10:** subir archivos a “files”. Primero crear carpeta mi proyecto y dentro poner los archivos pkl y el main.py que va manejar la predicción.

## Figura 46

*Gestión de archivos Pythonanyware*



**Paso 11:** Ahora sí, ajustar el archivo WSGI en WEB en el apartado Code:

## Figura 47

*Code->WSGI*

WSGI configuration file: `/var/www/`

**Paso 12:** Debe estar justo así, cambiar variable `tu_usuario` por el usuario real)

## Figura 48

*Configuración WSGI*

```

import sys

# add your project directory to the sys.path
project_home = '/home/tu_usuario/mi_proyecto'
if project_home not in sys.path:
    sys.path = [project_home] + sys.path

# import flask app but need to call it "application" for WSGI to work
from main import app as application # noqa

```

**Paso 13:** Ya configurado el servidor se recarga el servidor cada cierto tipo para que no lo duerman, ya que es un sitio gratuito.

## Figura 49

### *Web Creada Pythonanywhere y Flask*

The screenshot displays the PythonAnywhere dashboard interface. At the top, there is a navigation bar with links for 'Send feedback', 'Forums', 'Help', 'Blog', 'Account', and 'Log out'. Below this, the PythonAnywhere logo and 'by ANACONDA' are visible on the left, and a navigation menu with 'Dashboard', 'Consoles', 'Files', 'Web', 'Tasks', and 'Databases' is on the right. A green notification banner at the top of the main content area reads 'All done! Your web app is now set up. Details below.' The main content area is titled 'Configuration for Shameljane.pythonanywhere.com'. It features a 'Reload' section with a green button labeled 'Reload Shameljane.pythonanywhere.com'. Below this is a 'Best before date' section with a warning: 'This site will be disabled on Sunday 23 February 2025'. A yellow button labeled 'Run until 3 months from today' is positioned below the warning. At the bottom of the configuration area, a small note states: 'Pavinn users' sites stay on forever without any need to log in to keep them running.'

# CAPÍTULO V

## 5 EVALUACIÓN DE RESULTADOS

### 5.1 Introducción

Como parte del proceso de validación del sistema informático con árbol de decisión para el análisis del rendimiento académico de los estudiantes de la carrera de Ingeniería en Software de la ULEAM Extensión El Carmen. Para ello se realizaron pruebas funcionales y no funcionales donde se utilizaron datos en frío y datos reales, con el fin de verificar que las funcionalidades del sistema tengan un correcto desempeño, como el ver si los campos tenían validaciones de entrada, o si la respuesta del sistema era rápida y confiable con los resultados generados.

Estas pruebas se aplicaron a los principales procesos del sistema, tales como el inicio de sesión, el registro de docentes y estudiantes, el registro de materias, la asignación académica, la carga masiva de información y el módulo de predicción del rendimiento académico. Los resultados obtenidos permitieron identificar fortalezas en la operatividad del sistema, así como oportunidades de mejora relacionadas principalmente con la validación de rangos de entrada, la usabilidad de algunos campos y la optimización de filtros. Garantizando así que el sistema funcione de manera adecuada tanto en escenarios controlados como en condiciones reales de uso.

### 5.2 Presentación y monitoreo de resultados

Para el monitoreo de resultados se consideran las predicciones realizadas con Python y PHP, además de la carga masiva de datos. Considerando que se usó un modelo en cascada con árboles de decisión con algoritmos C4.5 y Random Forest. Para ello se consideran los siguientes resultados:

**Tabla 29**

*Resultados generales de las predicciones académicas*

<b>Algoritmo</b>	<b>Lenguaje</b>	<b>Variables utilizadas</b>	<b>Tipo de datos</b>	<b>Resultado de la predicción</b>
C4.5	Python	Promedio, Asistencia	Datos reales	Aprobado / Reprobado
C4.5	PHP	Promedio, Asistencia	Datos reales	Aprobado / Reprobado
Random Forest	Python	Promedio, Asistencia	Datos reales	Aprobado / Reprobado
Random Forest	PHP	Promedio, Asistencia	Datos reales	Aprobado / Reprobado

En esta tabla se puede ver que ambos lenguajes y algoritmos generan predicciones coherentes utilizando las mismas variables académicas. Con lo que se considera ideal la elección de estos algoritmos para el sistema informático con árbol de decisión para el análisis del rendimiento académico de los estudiantes.

**Tabla 30**

*Monitoreo de predicciones por algoritmo*

Algoritmo	Total de registros evaluados	Predicciones de aprobación	Predicciones de reprobación	Observación
C4.5	n	n	n	Presenta reglas claras y comprensibles
Random Forest	n	n	n	Mayor estabilidad ante variaciones de datos

Según las pruebas realizadas se obtuvo que Random Forest mostró mayor robustez, mientras que C4.5 facilitó la interpretación de las decisiones.

**Tabla 31**

*Comparación del comportamiento de los algoritmos según las variables*

Algoritmo	Promedio académico	Asistencia	Sensibilidad a cambios	Interpretabilidad
C4.5	Alta	Alta	Media	Alta
Random Forest	Alta	Alta	Baja	Media

Según los datos analizados se observó que C4.5 permite comprender fácilmente las reglas de decisión, mientras que Random Forest reduce el riesgo de sobreajuste.

**Tabla 32**

*Resultados de la carga masiva de datos para predicción*

Proceso	Cantidad de registros	Estado del proceso	Resultado
Carga masiva de notas	n	Correcto	Datos almacenados
Ejecución del modelo	n	Correcto	Predicciones generadas
Visualización	n	Correcto	Resultados mostrados

En esta tabla se muestra que al realizar la carga masiva se pudo procesar grandes volúmenes de información sin afectar la integridad del sistema.

**Tabla 33***Resumen de validación del sistema de predicción*

<b>Criterio</b>	<b>Resultado</b>
Correcta ejecución del modelo	Cumple
Uso adecuado de variables académicas	Cumple
Integración Python–PHP	Cumple
Generación de predicciones	Cumple
Soporte para carga masiva	Cumple

Los resultados obtenidos demuestran que el sistema informático implementado, basado en algoritmos de árboles de decisión C4.5 y Random Forest, permite generar predicciones confiables sobre la aprobación académica de los estudiantes a partir de variables clave como el promedio y la asistencia. La integración de Python y PHP, junto con la carga masiva de datos y el modelo en cascada, garantiza un procesamiento eficiente y una correcta visualización de los resultados, validando la funcionalidad y pertinencia del sistema propuesto.

### **5.3 Interpretación objetiva**

Los resultados obtenidos a partir de la aplicación del modelo en cascada para su desarrollo, utilizando algoritmos basados en árboles de decisión como C4.5 y Random Forest, evidencian un desempeño adecuado en la generación de predicciones sobre la aprobación académica de los estudiantes.

Al considerar variables clave como el promedio académico y la asistencia. Las predicciones generadas tanto en el entorno Python como en PHP mostraron coherencia y estabilidad. Esto confirma una buena adaptación del sistema al problema, que lo convierte en una posible solución que se debe pulir. Su buen manejo de los datos masivos y de generación de predicciones garantiza la integridad y consistencia de la información utilizada.

En términos de probabilidad el algoritmo Random Forest presentó una menor robustez y precisión frente a C4.5, aunque la diferencia es mínima y cambia por un 2% a un 5%, por lo que el uso de cualquiera de los dos presentara los resultados muy parecidos en conjunto, estos resultados confirman que el sistema informático desarrollado constituye una herramienta viable para apoyar la toma de decisiones académicas, al permitir la identificación temprana de estudiantes con riesgo de no aprobación.

# CAPÍTULO VI

## 6 CONCLUSIONES Y RECOMENDACIONES

### 6.1 Conclusiones

En la presente investigación se logró desarrollar un sistema informático basado en técnicas de minería de datos, utilizando algoritmos de árboles de decisión como C4.5 y Random Forest. El cual permitió evaluar el rendimiento académico de los estudiantes de la carrera de Ingeniería en Software de la ULEAM extensión El Carmen. El sistema demostró ser una herramienta eficaz para apoyar el seguimiento académico y la identificación temprana de estudiantes con riesgo de bajo rendimiento.

Para fundamentar teóricamente la investigación se lo hizo a partir de fuentes bibliográficas actualizadas y citadas bajo la normativa APA 7, permitió establecer conceptos claros sobre sistemas informáticos, minería de datos y rendimiento académico. Esta base conceptual fortaleció el sustento científico del estudio y orientó adecuadamente el diseño y desarrollo de la solución tecnológica propuesta.

El enfoque metodológico usado se sustentó en el paradigma cuantitativo, posibilitó la recopilación de información relevante mediante encuestas y técnicas estadísticas. Con los datos obtenidos se pudo observar mejor las necesidades y ver el problema desde diferentes puntos, así a su vez detectar que el sistema era viable y encajaba muy bien y llegaba a dar una solución.

Por ello el sistema propuesto se diseñó a partir de los resultados de las técnicas de análisis y bajo un modelo de diseño de software modelo-vista-controlador, también incorporó módulos como registro, carga masiva de datos y predicción académica, logrando así detectar a estudiantes con bajo rendimiento antes de que sea tarde y se puedan tomar acciones preventivas. Todo esto partiendo de variables como promedio y asistencia.

Esto evidencia que el uso de árboles de decisión es una alternativa válida para la toma de decisiones en el ámbito académico. Para validar la funcionalidad del sistema el mismo fue expuesto a pruebas funcionales y no funcionales, junto con validación de usuarios finales y usuarios de prueba para comprobar un correcto manejo de los roles y funcionalidades del sistema y así generar aceptación por parte de docentes y estudiantes.

## 6.2 Recomendaciones

Se recomienda a la ULEAM extensión El Carmen considerar la implementación del sistema informático desarrollado como una herramienta de apoyo institucional para el seguimiento académico, ya que esta permite la detección temprana de estudiantes con riesgo de bajo rendimiento y facilitando la toma de decisiones oportunas.

Es aconsejable ampliar el conjunto de variables utilizadas en el modelo predictivo, incorporando factores adicionales como aspectos socioeconómicos, hábitos de estudio o resultados de evaluaciones parciales con el fin de mejorar la precisión y el alcance de las predicciones académicas.

Se sugiere continuar con la actualización y entrenamiento periódico de los modelos de árboles de decisión utilizando nuevos datos académicos. Lo que permitirá mantener la vigencia y confiabilidad del sistema frente a cambios en los patrones de comportamiento estudiantil.

Se recomienda capacitar a los docentes y personal administrativo en el uso del sistema informático, para garantizar un aprovechamiento adecuado de sus funcionalidades y promover una cultura de toma de decisiones basada en datos dentro de la institución.

## 7 BIBLIOGRAFÍA

- Alan Neill, D., y Cortez Suárez, L. (2017). *Procesos y Fundamentos de la investigación Científica*. Editorial Utmach.  
<https://repositorio.utmachala.edu.ec/bitstream/48000/14232/1/Cap.4-Investigaci%C3%B3n%20cuantitativa%20y%20cualitativa.pdf>
- Aluja Banet, T. (2001). La minería de datos, entre la estadística y la inteligencia artificial. *Questiò: Quaderns d'Estadística, Sistemes, Informàtica i Investigació Operativa*, 25(3), 479-498. <https://dialnet.unirioja.es/servlet/articulo?codigo=2364961>
- Amat, R. J. (octubre de 2020). *Árboles de decisión, random forest, gradient boosting y C5.0*. [cienciadedatos.net/](http://cienciadedatos.net/):  
[https://cienciadedatos.net/documentos/33\\_arboles\\_decision\\_random\\_forest\\_gradient\\_boosting\\_c50](https://cienciadedatos.net/documentos/33_arboles_decision_random_forest_gradient_boosting_c50)
- Anderson, A. J. (1994). *Foundations of computer technology*. CRC Press.  
<https://doi.org/10.1201/9781003072164>
- Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., . . . Zaharia, M. (2020). A View of Cloud Computing. *Communications of the ACM*, 53(4), 50-58.
- Bandura, A. (1977). Self-efficacy: Toward a Unifying Theory of Behavioral Change. *Psychological Review*, 84(2), 191-215. <https://doi.org/10.1037/0033-295x.84.2.191>
- Bandura, A. (1994). Exercise of Personal and Collective Efficacy in Changing Societies. In A. Bandura (Ed.). *Selfefficacy in Changing Societies*, 1-45.
- Bandura, A. (2020). Impacto en la educación de la teoría cognitiva social del aprendizaje. *Revista Padres y Maestros / Journal of Parents and Teachers*, 384, 72-76.  
<https://dialnet.unirioja.es/servlet/articulo?codigo=7677756>
- Banu, G. (2016). A Role of decision Tree classification data Mining Technique in Diagnosing Thyroid disease. *International Journal of Computer Sciences and Engineering*, 4(11), 111-115.
- Barceló, M. (2008). *Una historia de la informática*. Editorial UOC.  
[https://www.torrossa.com/digital/toc/2012/2516910\\_TOC.pdf](https://www.torrossa.com/digital/toc/2012/2516910_TOC.pdf)

- Beel, J., Langer, S., Genzmehr, M., y Nürnberger, A. (2019). Introducing Docear's research paper recommender system. *In Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries*, 459-460.
- Bennedsen, J., y Carpersen, M. E. (2018). Exposing the programming process. Bennedsen, J., Carpersen, M. E. ve Kolling, M. (Eds.). *Reflection on the theory of programming: Methods and implementation*, 6-16.
- Bermak, A., y Martinez, D. (2003). A compact 3D VLSI classifier using bagging threshold network ensembles. *IEEE Trans. on Neural Networks*, 4(5), 1097-1109.
- Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(8), 123–140.
- Breiman, L., Friedman, J., Olshen, R., y Stone, C. (1984). *Classification and Regression Trees*. Wadsworth International Group.
- Brennan, K., y Resnick, M. (2022). Using artifact-based interviews to study the development of computational thinking in interactive media design. *Paper presented at annual American Educational Research Association meeting*.
- Brodley, C. E., y Utgoff, P. E. (1995). Multivariate decision trees. *Machine learning*, 19(1), 45–77.
- Chen, X., Yang, Z., y Lou, W. (2019). Fault Diagnosis of Rolling Bearing Based on the Permutation Entropy of VMD and Decision Tree. *Technology and Computer Engineering (EITCE), Xiamen, China*, 1911–1915. <https://doi.org/10.1109/EITCE47263.2019.9095187>.
- Coenen, F. (2011). Data mining: past, present and future. *The Knowledge Engineering Review*, 26(1), 25-29. <https://doi.org/10.1017/S0269888910000378>
- Columbus, L. (. (2019). Public Cloud Soaring To \$331B by 2022 According to Gartner. *Forbes*. <https://www.forbes.com/sites/louiscolombus/2019/04/07/publiccloud-soaring-to-331b-by-2022-according-togartner/?sh=1480d1de5739>
- De Mántaras, R. L. (1991). A distance-based attribute selection measure for decision tree induction. *Machine learning*, 6(1), 81–92.
- Díaz-Landa, B., Meleán-Romero, R., y Marín-Rodríguez, W. (2021). Rendimiento académico de estudiantes en educación superior: Predicciones de factores influyentes a partir de

- árboles de decisión. *TELOS: Revista de Estudios Interdisciplinarios en Ciencias Sociales*, 616–639. <https://doi.org/10.36390/telos233.13>
- Diego Garcia, E. d. (1995). *Historia de la industria en España: La electrónica y la informática*. ACTAS y escuela de Organización Industrial. [https://www.academia.edu/77284473/Emilio\\_De\\_Diego\\_Historia\\_De\\_La\\_Industria\\_e\\_n\\_Espana\\_La\\_Electronica\\_y\\_La\\_Informatica](https://www.academia.edu/77284473/Emilio_De_Diego_Historia_De_La_Industria_e_n_Espana_La_Electronica_y_La_Informatica)
- Domor Miernye, I., y Jere, N. (2016). A Survey of Decision Trees: Concepts, Algorithms, and Applications. *IEEE Access*, 4, 1-15. <https://doi.org/https://creativecommons.org/licenses/by/4.0/>
- Dragoni, N. (2019). Microservices: yesterday, today, and tomorrow. 1–17.
- Fogel, D. B. (1992). *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*. *IEEE Press*.
- Fowler, M., y Lewis, J. (2019). *Microservices*. <https://martinfowler.com/articles/microservices.html>
- Freund, Y., y Mason, L. (1999). The alternating decision tree learning algorithm. *Inicml*, 99, 124-133.
- Gist, M. E., y Mitchell, T. R. (1992). Self-Efficacy: A Theoretical Analysis of its Determinants and Malleability. *The Academy of Management Review*, 17(2).
- Gomaa, H., y Shin, E. (2004). A Multiple View Meta-modeling Approach for Variability Management in Software Product Lines. *Eighth International Conference on Software Reuse: Methods, Techniques and Tools*.
- González Barahona, J., Seoane Pascual, J., y Robles, G. (2003). *Introducción al Software libre*. <https://biblio.flacsoandes.edu.ec/libros/digital/40241.pdf>
- Goossens, B. (2019). *Decision -Making in a microservice architecture*. University of twente.
- Gupta, C., y Ramdas, A. (2021). Distribution-free calibration guarantees for histogram binning without sample splitting. *International conference on machine learning*, 3942–3952.
- Havik, T., y Westergård, E. (2020). Estudio sobre percepción de interacciones en el aula y apoyo emocional del profesorado. *Revista/Editorial*.
- Heath, D., Kasif, S., y Salzberg, S. (1993). *Learning oblique decision trees*. Morgan Kaufmann.

- Hernández Leal, E. J. (2024). *Modelo de dominio específico para análisis y minería de datos educativos*. Universidad Nacional de Colombia, Ingeniería - Industria y Organizaciones, Bogotá. <https://repositorio.unal.edu.co/handle/unal/85413>
- Hernández, R., Fernández, C., y Baptista, P. (2014). *Metodología de la investigación*. McGRAW-HILL. <https://www.esup.edu.pe/wp-content/uploads/2020/12/2.%20Hernandez,%20Fernandez%20y%20Baptista-metodolog%C3%ADa%20Investigacion%20Cientifica%206ta%20ed.pdf>
- Hernández, U. (2015). *codigofacilito*. <https://codigofacilito.com/articulos/mvc-model-view-controller-explicado>
- Instituto de Ingeniería del conocimiento. (2021). *La metodología CRISP-DM en ciencia de datos*. La metodología CRISP-DM en ciencia de datos
- Jadhav, S., y Channe, H. (2019). Efficient recommendation system using decision tree classifier and collaborative filtering. *Int. Res. J. Eng. Technol*, 3(21), 13-18.
- Jiao, S. R., Song, J., y Liu, B. (2020). A Review of Decision Tree Classification Algorithms for Continuous Variables. *Journal of Physics: Conference Series*, 1651(1), 012083.
- Laudon, K. C., y Laudon, J. P. (2020). *Sistemas de información gerenciales (15.ª ed.)*. Pearson Educación.
- Lim, T.-S., Loh, W.-Y., y Shih, Y.-S. (2000). A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. *Machine learning*, 40(3), 203-228.
- Liu, Y., Hu, L., Yan, F., y Zhang, B. (2013). Information Gain with Weight Based Decision Tree for the Employment Forecasting of Undergraduate. *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, 2210–2213. <https://doi.org/10.1109/GreenCom-iThingsCPSCom.2013.417>
- Llor`a, X., y Garrell, J. M. (2000). Automatic Classification and Artificial Life Models. *Proceedings of Learning00 Workshop*, 1-6.
- Loh, W.-Y. (2014). Fifty Years of Classification and Regression Trees. *International Statistical Review*, 82. <https://doi.org/10.1111/insr.12016>

- Lopez-Estrada, S., y Cumplido, R. (2006). Decision tree based FPGA architecture for texture sea state classification. *Reconfigurable Computing and FPGA's ReConFig 2006 IEEE International Conference*, 1-7.
- Luszczynska, A., y Schwarzer, R. (2008). Planning and SelfEfficacy in the adoption and maintenance of breast study examination: A longitudinal study of self-regulatory cognitions. *Psychology and health*(18), 93-108.
- Marksteiner, T., Nishen, A., y Dickhäuser, O. (. (2021). Students' perception of teachers' reference norm orientation and cheating in the classroom. *Frontiers in Psychology*, 12, 614199. <https://doi.org/https://doi.org/10.3389/fpsyg.2021.614199>
- Martínez, C. (2018). *Investigación descriptiva: definición, tipos y características*. <https://www.lifeder.com/investigacion-descriptiva>
- Matzavela, V., & Alepis, E. (2021). Decision tree learning through a Predictive Model for Student Academic Performance in Intelligent M-Learning environments. *Computers and Education: Artificial Intelligence*, 2. <https://doi.org/https://doi.org/10.1016/j.caeai.2021.100035>
- Mazurek, F., Tschand, A., Wang, Y., Pajic, M., y Sorin, D. (2023). Rigorous evaluation of computer processors with statistical model checking. *MICRO '23: 56th Annual IEEE/ACM International Symposium on Microarchitecture, ACM*, 10.
- Meleán, R., Marín, W., & Díaz, B. (2021). Rendimiento académico de estudiantes en Educación Superior: predicciones de factores influyentes a partir de árboles de decisión. *Telos*, 23(3), 616-639. <https://doi.org/https://doi.org/10.36390/telos233.08>
- Mell, P., y Grance, T. (2019). The NIST Definition of Cloud Computing. *Technical Report 15*.
- Michalewicz et al. (1996). *Genetic Algorithms + Data Structures = Evolution Programs*. 3rd edn.
- Mienye, I. D., Sun, Y., y Wang, Z. (2019). Prediction performance of improved decision tree-based algorithms: a review. *Procedia Manufacturing*, 35, 698–703.
- Mobasher, B. (2007). 2007. *School of Computer Science, Telecommunication, and Information Systems puter Science, Telecommunication, and Information Systems*, 1(1), 90–135. [https://doi.org/DOI: 10.1007/978-3-540-72079-9\\_3](https://doi.org/DOI: 10.1007/978-3-540-72079-9_3)

- Mohapatra, N., Shreya, K., y Chinmay, A. (2020). Optimization of the Random Forest Algorithm. *Lecture Notes on Data Engineering and Communications Technologies 37*, 201-208. [https://doi.org/https://doi.org/10.1007/978-981-15-0978-0\\_19](https://doi.org/https://doi.org/10.1007/978-981-15-0978-0_19)
- Morales Gavilanes, C. S. (2024). *Deserción escolar en Ecuador: Modelo basado en árboles de decisión*. Tesis de maestría, Universidad de Las Américas.
- Nabaho, S., Nkapyia, A. M., y Ssentamu, A. (2017). Estudio sobre la percepción estudiantil del profesorado. *Revista/Editorial*.
- Paul, R. J. (2010). What an Information System Is, and Why Is It Important to Know This. *Journal of Computing and Information Technology - CIT 18, 2*, 95–99. <https://doi.org/10.2498/cit.1001800>
- Pereyra Girardi, C. I., Ronchieri Pardo, C. d., Rivas, A., Trueba, D. A., Mur, J. A., y Páez Vargas, N. (2018). Autoeficacia: una revisión aplicada a diversas áreas de la psicología. *Scielo, 16(2)*, 299-325. [http://www.scielo.org.bo/pdf/rap/v16n2/v16n2\\_a04.pdf](http://www.scielo.org.bo/pdf/rap/v16n2/v16n2_a04.pdf)
- Piramuthu, S. (2008). Input data for decision trees. *Expert Systems with Applications, 34*, 1220–1226.
- Priyama, A., Abhijeeta, R., Ratheeb, A., y Srivastavab, S. (2013). Comparative analysis of decision tree classification algorithms. *International Journal of Current Engineering and Technology, 3(2)*, 334-337.
- Puga Paredes, J. L. (2023). *Mapeo Sistemático sobre el seguimiento del aprendizaje de estudiantes mediante el uso de la minería de datos educativos*. Universidad Politécnica Salesiana. <https://dspace.ups.edu.ec/bitstream/123456789/26641/4/UPS-GT004893.pdf>
- Quinlan, R. (1986). Induction of decision trees. *Machine learning, 1(1)*, 81–106.
- Quinlan, R. (1993). *C4.5: Programs for Machine Learning*. . *Morgan Kaufmann Publishers*.
- Raya Cabrera, J. L., & Raya González, L. (2015). *Sistemas informáticos*. Madrid. <https://elibro.net/es/ereader/uleam/62481?page=12>
- Rehman Gilal, A., Omar, M., Gilal, R., Waqas, A., Afridi, S., y Jaafar, J. (2019). A Decision Tree Model for Software Development Teams. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, 8(5S), 241-245.

- Rico Páez, A., y Gaytán Ramírez, N. D. (2022). Modelos predictivos del rendimiento académico a partir de características de estudiantes de ingeniería. *ie Revista de Investigación Educativa de la Rediech*, 13.
- Rokach, L. (2016). Decision forest: twenty years of research. *Inf. Fusion* 27, 111–125.
- Rokach, L., y Maimon, O. (2005). *Decision Trees*. Tel-Aviv University.
- Rokach, L., y Maimon, O. (2005). Top-Down Induction of Decision Trees Classifiers—A Survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions*, 35, 476–487. <https://doi.org/10.1109/TSMCC.2004.843247>
- Safari, E., y Hejazi, M. (2017). Learning styles and self-regulation: an associational study on high school students. *Mediterranean Journal of Social Sciences*, 8(1), 463-469.
- Sangüesa, R. (2016). *El proceso de descubrimiento de conocimiento a partir de datos*. Universidad Abierta de Cataluña. <https://openaccess.uoc.edu/server/api/core/bitstreams/1611ca88-76ab-4e8b-b9dd-dd8d6164b5de/content>
- Schwarzer, R. E., y Luszczynska, A. (2008). Self-Efficacy. *M. Gerrard & K. O. McCaul (Eds). Health Behaviour Constructs: Theory, Measurement and Research. National Cancer Institute Website*.
- Seifert. (28 de septiembre de 2023). *¿Qué es mejor? ¿Tener servidores locales o en la nube?* <https://www.seifert.com/which-is-better-having-servers-on-premise-or-in-the-cloud>
- Shaw, M., y Garlan, D. (1996). *Software Architecture*. Prentice Hall.
- Singh, S., y Gupta, P. (2014). Comparative study ID3, cart and C4. 5 decision tree algorithm: a survey. *International Journal of Advanced Information Science and Technology (IJAIST)*, 27(27), 97-103.
- Sommerville, I. (2011). *Software engineering*. Pearson Education, Inc. [https://gc.scalahed.com/recursos/files/r161r/w25469w/ingdelsoftwarelibro9\\_compressed.pdf](https://gc.scalahed.com/recursos/files/r161r/w25469w/ingdelsoftwarelibro9_compressed.pdf)
- Struharik, J. (2011). Implementing Decision Trees in Hardware . *1 IEEE 9th International Symposium on Intelligent Systems and Informatics*, 8(16), 40-46.

- Trasobares, A. H. (2003). Los sistemas de información: evolución y desarrollo. *Proyecto Social: Revista de Relaciones Laborales*, 149-165.
- Tso, G. K., y Yau, K. K. (2007). Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks. *Energy*, 32(9), 1761–1768. <https://doi.org/10.1016/j.energy.2006.11.010>
- Zhang, X., y Jiang, S. (2012). A Splitting Criteria Based on Similarity. *Decision Tree Learning*, 7(8), 1775-1782.
- Zimmermann, O. (2017). Microservices tenets: Agile approach to service development and deployment. *Comput. Sci. - Res. Dev*, 3(4), 301–310.

# ANEXOS

## Anexo A. Aprobación de tema

Estimad@  
Docente y Estudiante  
Uleam

En cumplimiento de lo establecido en la Ley, el Reglamento de Régimen Académico y las disposiciones estatutarias de la Uleam, por medio de la presente se oficializa la dirección y tutoría en el desarrollo del Trabajo de Integración curricular / Trabajo de Titulación del siguiente estudiante:

**Tema:** SISTEMA INFORMÁTICO CON ÁRBOL DE DECISIÓN PARA RENDIMIENTO ACADÉMICO DE ESTUDIANTES DE LA CARRERA DE INGENIERÍA EN SOFTWARE DE ULEAM EXTENSIÓN EL CARMEN

**Estado de aprobación:** Aprobado

**Tipo de titulación:** Trabajo de Integración Curricular

**Tipo de proyecto:** Trabajo de Integración Curricular / Trabajo de titulación se articula con proyectos y programas de Investigación.

**Apellidos y nombres del tutor asignado:** REASCOS PINCHAO RAUL SAED

**Apellidos y nombres del estudiante:** MONTALVAN LOOR KAREN NATHALY

**Carrera:** TECNOLOGÍAS DE LA INFORMACIÓN 2022 (EL CARMEN)

**Periodo de inducción:** Periodo 2025-1

Sírvase(n) cumplir con lo dispuesto en el Manual de Procedimientos de TITULACIÓN DE ESTUDIANTES DE GRADO: TRABAJO DE INTEGRACIÓN CURRICULAR Y UNIDAD DE TITULACIÓN

## **Anexo B. Instrumento entrevista**

### **Entrevista**

**Título:** Sistema Informático con árbol de decisión para rendimiento académico de estudiantes de la Carrera de Ingeniería en Software de ULEAM El Carmen.

**Autora:** Montalván Loor Karen Nathaly

**Dirigida A:** Coordinador académico de la Carrera de Ingeniería en Software de ULEAM El Carmen.

1. ¿Cómo evalúa actualmente el rendimiento académico de los estudiantes de la carrera?
2. ¿Qué herramientas tecnológicas se utilizan actualmente para el seguimiento del rendimiento académico?
3. ¿Qué limitaciones ha identificado en el sistema actual de control y análisis académico?
4. Desde su perspectiva ¿cree que un sistema informático basado en árbol de decisión podría mejorar la gestión del rendimiento académico?
5. ¿Qué tipo de datos considera esenciales para que un sistema de este tipo sea eficiente?
6. ¿Qué dificultades podrían surgir en la implementación de un sistema con estas características?
7. ¿Qué beneficios espera que este sistema aporte a los docentes, estudiantes de la carrera?
8. ¿Cómo cree que se debería capacitar al personal para el uso adecuado de esta herramienta?
9. ¿Estaría dispuesto a apoyar la implementación y evaluación de un proyecto piloto de este sistema?

## Anexo C. Instrumentos de encuesta

### Encuesta dirigida a estudiantes

**Título:** Sistema Informático con árbol de decisión para rendimiento académico de estudiantes de la Carrera de Ingeniería en Software de ULEAM El Carmen.

**Autora:** Montalván Loor Karen Nathaly

**Dirigida A:** Estudiantes de la Carrera de Ingeniería en Software de ULEAM El Carmen.

#### Preguntas:

1. ¿Con qué frecuencia ingresa al sistema académico?
  - Varias veces por semana
  - Una vez por semana
  - Una vez al mes
  - Casi nunca
2. ¿Revisa regularmente las calificaciones asignadas por sus docentes?
  - Siempre
  - A veces
  - Rara vez
  - Nunca
3. ¿Considera importante conocer sus notas periódicamente?
  - Muy importante
  - Importante
  - Poco importante
  - Nada importante
4. ¿Sus responsabilidades laborales le impiden ingresar regularmente al sistema académico?
  - Siempre
  - A veces
  - Rara vez
  - Nunca
5. ¿El cansancio físico le dificulta revisar sus calificaciones?
  - Siempre
  - A veces
  - Rara vez
  - Nunca
6. ¿Trabaja al mismo tiempo que estudia?
  - Si
  - No
7. ¿Sus docentes publican las calificaciones con rapidez?
  - Siempre

- A veces
  - Rara vez
  - Nunca
8. ¿Ha detectado errores en el registro de sus calificaciones?
- Si
  - No
9. ¿Sus docentes le brindan comentarios claros sobre sus evaluaciones?
- Siempre
  - A veces
  - Rara vez
  - Nunca

## Encuesta dirigida a docentes

**Título:** Sistema Informático con árbol de decisión para rendimiento académico de estudiantes de la Carrera de Ingeniería en Software de ULEAM El Carmen.

**Autora:** Montalván Loor Karen Nathaly

**Dirigida A:** Docentes de la Carrera de Ingeniería en Software de ULEAM El Carmen.

### Preguntas:

1. ¿Con qué frecuencia actualiza o ingresa información al sistema académico?
  - Varias veces por semana
  - Una vez por semana
  - Una vez al mes
  - Casi nunca
2. ¿Registra las calificaciones de sus estudiantes de forma regular?
  - Siempre
  - A veces
  - Rara vez
  - Nunca
3. ¿Considera importante que los estudiantes conozcan sus notas periódicamente?
  - Muy importante
  - Importante
  - Poco importante
  - Nada importante
4. ¿Sus responsabilidades laborales adicionales le impiden registrar calificaciones a tiempo en el sistema académico?
  - Siempre
  - A veces
  - Rara vez
  - Nunca
5. ¿El cansancio físico más carga laboral le dificulta actualizar las calificaciones en el sistema académico?
  - Siempre
  - A veces
  - Rara vez

- Nunca
6. ¿Actualmente realiza actividades laborales externas a la docencia en la universidad?
    - Sí
    - No
  7. ¿Publica las calificaciones de los estudiantes con rapidez después de las evaluaciones?
    - Siempre
    - A veces
    - Rara vez
    - Nunca
  8. ¿Ha detectado errores en el registro de calificaciones dentro del sistema académico?
    - Sí
    - No
  9. ¿Proporciona retroalimentación clara y oportuna a los estudiantes sobre sus evaluaciones?
    - Siempre
    - A veces
    - Rara vez
    - Nunca

**Anexo D. Fotografías**

**Figura 50**

Imagen del proceso de desarrollo





## GLOSARIO

**Árbol de Decisión:** Se trata de una técnica usada en machine learning para el aprendizaje automático, la cual emplea nodos estructurados y ramas para clasificar datos y así predecir casos futuros.

**Algoritmo C4.5:** Se trata de un procedimiento organizado para clasificar información y que usa árboles de decisión para predecir usando la información previa. Por ello en el contexto de este trabajo, el algoritmo nos plasmaría si un estudiante aprobará, reprobará o se estuviera en riesgo.

**Random Forest:** es otro tipo de algoritmo de predicción que combina múltiples árboles de decisión para mejorar la precisión y evitar errores al momento de manejar los datos y arrojar la predicción.

**Descubrimiento de Conocimiento en Bases de Datos:** Es un proceso que se usa para identificar patrones novedosos y útiles dentro de grandes cantidades de datos.

**Minería de Datos:** es una rama de la inteligencia artificial que usa software

junto con técnicas y algoritmos para hallar información oculta o patrones no vistos a simple vista.

**Modelo-Vista-Controlador:** Es un Patrón de diseño para la arquitectura de un software, que en resumen separa los datos de la aplicación, en lo que ve el usuario final y la lógica, haciendo más ordenado su desarrollo.

**Rendimiento Académico:** Nivel de que puntea el logro alcanzado por un estudiante en su proceso académico, medido comúnmente a través de notas y asistencia.

**Sistema Informático:** Conjunto de hardware, software y personal que interactúan entre sí para procesar, almacenar y distribuir información que apoye la toma de decisiones.

**Responsivo:** Es la capacidad de un sistema informático para adaptarse a cualquier tamaño de pantalla, en diversos dispositivos, sin dejar de ser funcional.

**Tecnologías de la Información:**

Recursos tecnológicos utilizados para gestionar información, esenciales procesos de enseñanza y aprendizaje. En el contexto del trabajo realizado es la carrera académica en la que se aplica el mismo.

**Entropía:** Trata de medir el grado de desorden en un conjunto de datos sin clasificación, en minería de datos nos dice que tan desordenados están los datos dentro de un nodo.

**Alertas Tempranas:** Es una forma en la que el sistema se encarga de notificar

a los diferentes roles sobre las acciones tomadas sobre su rol, lo que indica alguna situación de riesgo académico.

**Carga Masiva de Datos:** Es una de las funcionalidades del sistema que permite importar grandes volúmenes de datos que en este caso eran notas y estudiantes todo para facilitar el proceso.

**API:** Hablamos de un intercambio o comunicación que se lleva a cabo entre distintas aplicaciones a través del uso de protocolos.