

# Predicción del Rendimiento Académico Universitario: Un Enfoque Multifactorial con Estadística Bayesiana.

Pinzón, Angela <sup>1</sup>, Cervantes , Daniel<sup>2</sup>, Yañez, Andrés<sup>3</sup>

Dpto. de Matemáticas,  
Universidad Externado de Colombia  
Pregrado Ciencia de Datos  
Curso de Bases de Datos  
Bogotá, Colombia

2 de noviembre de 2023

## Índice

<b>1. Introducción</b>	<b>3</b>
<b>2. Características del proyecto de investigación que hace uso de Bases de Datos</b>	<b>3</b>
2.1. Título del proyecto de investigación . . . . .	3
2.2. Objetivo general . . . . .	3
2.2.1. Objetivos específicos . . . . .	3
2.3. Alcance . . . . .	4
2.4. Pregunta de investigación . . . . .	4
2.5. Hipótesis . . . . .	4
<b>3. Reflexiones sobre el origen de datos e información</b>	<b>5</b>
3.1. ¿Cuál es el origen de los datos e información? . . . . .	5
3.2. ¿Cuáles son las consideraciones legales o éticas del uso de la información? . . . . .	5
3.3. ¿Cuáles son los retos de la información y los datos que utilizará en la base de datos en términos de la calidad y la consolidación?	5
3.4. ¿Qué espera de la utilización de un sistema de Bases de Datos para su proyecto? . . . . .	5
<b>4. Diseño del Modelo de Datos del SMBD (Sistema Manejador de Bases de Datos)</b>	<b>6</b>

4.1. Características del SMBD (Sistema Manejador de Bases de Datos) para el proyecto . . . . .	6
4.2. Diagrama modelo de datos . . . . .	6
4.3. Código SQL - lenguaje de definición de datos (DDL) . . . . .	6
4.4. Código SQL - Manipulación de datos (DML) . . . . .	8
4.5. Código SQL + Resultados: Vistas . . . . .	9
4.6. Código SQL + Resultados: Triggers . . . . .	10
4.7. Código SQL + Resultados: Funciones . . . . .	10
4.8. Código SQL + Resultados: procedimientos almacenados . . . . .	11
<b>5. Bases de Datos No-SQL</b>	<b>12</b>
5.1. Diagrama de la Base de Datos NoSQL . . . . .	12
5.2. SMBD Utilizado para la Base de Datos NoSQL . . . . .	14
<b>6. Aplicación de ETL (Extract, Transform, Load) y Bodega de Datos (<i>Tercera entrega</i>)</b>	<b>17</b>
6.1. Ejemplo de aplicación de ETL y Bodega de Datos ( <i>Tercera entrega</i> )	17
6.2. Automatización de Datos ( <i>Tercera entrega</i> ) . . . . .	17
6.3. Integración de Datos ( <i>Tercera entrega</i> ) . . . . .	17
<b>7. Próximos pasos (<i>Tercera entrega</i>)</b>	<b>18</b>
<b>8. Lecciones aprendidas (<i>Tercera entrega</i>)</b>	<b>19</b>
<b>9. Bibliografía</b>	<b>20</b>

## **1. Introducción**

El objetivo principal de este proyecto de investigación es determinar cómo una variedad de factores, ya sean personales o profesionales son capaces de influir la nota de un grupo de estudiantes de la Facultad de Ingeniería y la Facultad de Ciencias de la Educación en 2019. A partir de esta determinación, se propone desarrollar un modelo estadístico con el fin de predecir el rendimiento académico de los estudiantes. Esta herramienta permitirá establecer rutas iniciales para alcanzar resultados académicos deseados, teniendo en cuenta las diferentes influencias de familia o hábitos de estudio, entre otros factores. Nuestro objetivo general es determinar cómo estos factores interactúan y afectan el rendimiento académico de los estudiantes universitarios y, a partir de esta comprensión, desarrollar un modelo estadístico efectivo que pueda predecir con precisión este rendimiento. La herramienta resultante no solo ofrecerá una visión integral de los determinantes académicos, si no que también nos dará datos importantes para la toma de decisiones frente a los aspectos académicos.

## **2. Características del proyecto de investigación que hace uso de Bases de Datos**

### **2.1. Título del proyecto de investigación**

El Título sugerido inicialmente por nosotros, sin tener definido inicialmente el modelo utilizado de estadística bayesiana: "Predicción del Rendimiento Académico Universitario: Un Enfoque Multifactorial con Estadística Bayesiana."

### **2.2. Objetivo general**

Determinar cómo diversos factores, personales, de estudio, familiares, entre otros afectan en el rendimiento académico. Desarrollar un modelo estadístico con enfoque bayesiano para predecir el rendimiento calificativo de los estudiantes.

#### **2.2.1. Objetivos específicos**

- Identificar los factores determinantes del rendimiento académico: Se buscarán y determinarán los factores más influyentes en el rendimiento académico de los estudiantes.
- Analizar la relación entre factores y rendimiento académico: Se llevará a cabo un análisis estadístico de los datos recolectados para determinar la influencia de factores en el rendimiento académico de los alumnos.
- Proponer rutas de mejora académica: Utilizando el modelo desarrollado, se ofrecerán sugerencias concretas para mejorar el desempeño académico de los estudiantes.

### **2.3. Alcance**

En un ambiente universitario, la búsqueda de métodos efectivos para mejorar el rendimiento académico de los estudiantes universitarios es una prioridad. Este proyecto de investigación surge con el propósito de abordar esta cuestión desde una perspectiva multifactorial y con el apoyo de la estadística bayesiana. Nuestra investigación se centra en comprender cómo una amplia gama de factores, que van más allá de las aulas, influyen en el desempeño académico de los alumnos. Estos factores incluyen aspectos familiares, profesionales, el cómo estudia el individuo entre otros elementos.

### **2.4. Pregunta de investigación**

”¿Cómo puede influir diversos factores personales o profesionales en el rendimiento académico de los estudiantes de la Facultad de Ingeniería y la Facultad de Ciencias de la Educación en 2019?”

### **2.5. Hipótesis**

Al revisar otros estudios frente al tema, no encontramos modelos sencillos para lograr una predicción inicial ÚNICAMENTE con variables relacionadas directamente con las notas y variables, es posible que el modelo crezca en complejidad proporsional a que datos se valla a utilizar y se deba investigar sobre técnicas o modelos muy específicos que se han usado para otros temas para realizar una extrapolación a nuestro problema.

### **3. Reflexiones sobre el origen de datos e información**

En esta sección, se abordarán cuestiones relacionadas con el origen de los datos e información que se utilizarán en la investigación, así como las consideraciones éticas y los desafíos asociados. También se discutirá la importancia de la calidad y consolidación de los datos y las expectativas relacionadas con el uso de un sistema de bases de datos como el paso inicial junto con la verificación del estado del arte.

#### **3.1. ¿Cuál es el origen de los datos e información?**

Los datos se obtuvieron de una universidad en Sekeroglu, donde los estudiantes de la Facultad de Ingeniería y la Facultad de Ciencias de la Educación en 2019 brindaron esta información.

#### **3.2. ¿Cuáles son las consideraciones legales o éticas del uso de la información?**

El uso de datos académicos y personales implica consideraciones éticas y legales importantes. Se respetarán todas las regulaciones de privacidad y no se trabajará con ninguna información perjudicial. Además, se protegerá la confidencialidad de la información personal y se seguirán los procedimientos éticos establecidos por la institución.

#### **3.3. ¿Cuáles son los retos de la información y los datos que utilizará en la base de datos en términos de la calidad y la consolidación?**

El mayor reto de la información es organizar de manera correcta, pues las distintas relaciones entre las variables, pues el cómo estas demuestran o no una interacción puede diferenciar el modelo.

#### **3.4. ¿Qué espera de la utilización de un sistema de Bases de Datos para su proyecto?**

El uso de un sistema de bases de datos permitirá una gestión eficiente de los datos recopilados, su almacenamiento seguro y la realización de consultas y análisis de manera efectiva. Se espera que el sistema de bases de datos facilite la manipulación de datos y contribuya a la robustez de la investigación al proporcionar una estructura organizada para el almacenamiento y acceso a la información. Inicialmente se plantea el uso de manejar un sistema estructurado de datos, sin embargo en la práctica creemos que sea mas versátil el uso de un sistema no estructurado, se determinará cuál sistema servirá más en posteriores entregas.

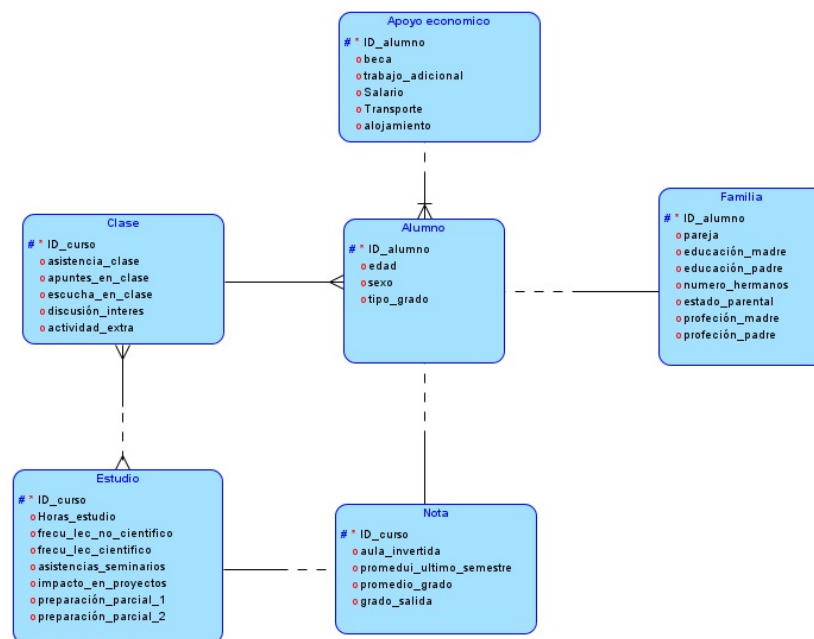
## 4. Diseño del Modelo de Datos del SMBD (Sistema Manejador de Bases de Datos)

### 4.1. Características del SMBD (Sistema Manejador de Bases de Datos) para el proyecto

El modelo se creó en Oracle SQL Developer Data Modeler, dentro de las características más importantes que nos hizo elegir esta herramienta están:

- Fuerte soporte para consultas complejas y operaciones de base de datos.
- Extensiones propias con soporte de múltiples lenguajes de programación, incluidos python, lo que nos permite interactuar con código en python y usar lenguaje SQL dentro de este.

### 4.2. Diagrama modelo de datos



### 4.3. Código SQL - lenguaje de definición de datos (DDL)

```
1 CREATE TABLE alumno (  
2     id_alumno INTEGER NOT NULL,  
3     edad      INTEGER,  
4     Sexo      VARCHAR(10) ,  
5     tipo_grado VARCHAR(10));  
6
```

```

7 ALTER TABLE alumno ADD CONSTRAINT alumno_pk
8 PRIMARY KEY ( id_alumno );
9
10 CREATE TABLE apoyoeconomico (
11     id_alumno          INTEGER NOT NULL,
12     beca                VARCHAR(10) ,
13     trabajo_adicional  INTEGER,
14     Salario             VARCHAR(15) ,
15     transporte         VARCHAR(30) ,
16     alojamiento        VARCHAR(15));
17
18 CREATE TABLE Nota (
19     id_curso            INTEGER NOT NULL,
20     aula_invertida      VARCHAR(15) ,
21     promedio_ult_semestre VARCHAR(20) ,
22     promedio_grado      VARCHAR(20) ,
23     grado_salida        VARCHAR(5));
24
25 CREATE TABLE estudio (
26     id_curso            INTEGER NOT NULL,
27     horas_estudio       VARCHAR(25) ,
28     frecu_lec_no_cientifico VARCHAR(20) ,
29     frecu_lec_cientifico  VARCHAR(20) ,
30     asistencias_seminarios VARCHAR(2) ,
31     impacto_proyec       VARCHAR(20) ,
32     preparacion_parcial_1 VARCHAR(10) ,
33     preparacion_parcial_2 VARCHAR(30));
34
35 ALTER TABLE nota ADD CONSTRAINT curso_pk
36 PRIMARY KEY ( id_curso );
37
38 CREATE TABLE clase (
39     id_curso            INTEGER NOT NULL,
40     asistencia_clase     VARCHAR(10) ,
41     apuntes_en_clase     VARCHAR(10) ,
42     escucha_en_clase     VARCHAR(10) ,
43     discusion_interes    VARCHAR(10) ,
44     actividad_extra      VARCHAR(2));
45
46 CREATE TABLE familia (
47     id_alumno           INTEGER NOT NULL,
48     pareja              VARCHAR(2) ,
49     educacion_madre     VARCHAR(15) ,
50     educacion_padre     VARCHAR(15) ,
51     num_hermanos        INTEGER,
52     estado_parental     VARCHAR(35) ,
53     profecion_madre     VARCHAR(40) ,
54     profecion_padre     VARCHAR(40));
55
56 ALTER TABLE familia
57     ADD CONSTRAINT familia_alumno_fk FOREIGN KEY ( id_alumno )
58     REFERENCES alumno ( id_alumno );
59
60 ALTER TABLE apoyoeconomico
61     ADD CONSTRAINT table_6_alumno_fk FOREIGN KEY ( id_alumno )
62     REFERENCES alumno ( id_alumno );
63

```

```

64 ALTER TABLE estudio
65     ADD CONSTRAINT table_7_curso_fk FOREIGN KEY ( id_curso )
66     REFERENCES curso ( id_curso );
67
68 ALTER TABLE clase
69     ADD CONSTRAINT table_7_curso_fk FOREIGN KEY ( id_curso )
70     REFERENCES curso ( id_curso );

```

#### 4.4. Código SQL - Manipulación de datos (DML)

```

1  — Insertar un nuevo alumno
2  INSERT INTO alumno (id_alumno, edad, sexo, tipo_grado)
3  VALUES (1, 20, 'M', 'Tcnico');
4
5  — Insertar un nuevo registro de apoyo economico
6  INSERT INTO apoyoeconomico (id_alumno, beca, trabajo_adicional,
7  salario, transporte, alojamiento)
8  VALUES (1, 'S', 50, '10000', 'Autob s',
9  'Alojamiento_universitario');
10
11 — Insertar un nuevo registro de nota
12 INSERT INTO nota (id_curso, aula_invertida,
13 promedio_ult_semestre, promedio_grado, grado_salida)
14 VALUES (1, 'S', '8.5', '9.0', 'Licenciado');
15
16 — Insertar un nuevo registro de estudio
17 INSERT INTO estudio (id_curso, horas_estudio,
18 frecu_lec_no_cientifico, frecu_lec_cientifico,
19 asistencias_seminarios, impacto_proyec, preparacion_parcial_1,
20 preparacion_parcial_2)
21 VALUES (1, '10', '3', '5', '2', 'Alto', 'Bueno', 'Excelente');
22
23 — Insertar un nuevo registro de clase
24 INSERT INTO clase (id_curso, asistencia_clase, apuntes_en_clase,
25 escucha_en_clase, discusion_interes, actividad_extra)
26 VALUES (1, 'S', 'S', 'S', 'S', 'S');
27
28 — Insertar un nuevo registro de familia
29 INSERT INTO familia (id_alumno, pareja, educacion_madre,
30 educacion_padre, num_hermanos, estado_parental, profecion_madre,
31 profecion_padre)
32 VALUES (1, 'No', 'Secundaria', 'Secundaria', 2, 'Casado',
33 'Enfermera', 'Obrero');
34
35 — Actualizar el sexo del alumno con ID 1
36 UPDATE alumno
37 SET sexo = 'F'
38 WHERE id_alumno = 1;
39
40 — Actualizar el salario del apoyo economico del alumno
41 con ID 1
42 UPDATE apoyoeconomico
43 SET salario = '15000'
44 WHERE id_alumno = 1;
45

```



```

46  — Actualizar el promedio del ltimo semestre de la nota
47  con ID 1
48  UPDATE nota
49  SET promedio_ult_semestre = '9.5'
50  WHERE id_curso = 1;
51
52  — Actualizar las horas de estudio del estudio con ID 1
53  UPDATE estudio
54  SET horas_estudio = '15'
55  WHERE id_curso = 1;
56
57  — Actualizar la asistencia a clase de la clase con ID 1
58  UPDATE clase
59  SET asistencia_clase = 'No'
60  WHERE id_curso = 1;
61
62  — Actualizar el estado parental de la familia con ID 1
63  UPDATE familia
64  SET estado_parental = 'Divorciado'
65  WHERE id_alumno = 1;
66
67  — Eliminar al alumno con ID 1
68  DELETE FROM alumno
69  WHERE id_alumno = 1;
70
71  — Eliminar el registro de apoyo economico del alumno con ID 1
72  DELETE FROM apoyoeconomico
73  WHERE id_alumno = 1;
74
75  — Eliminar el registro de nota con ID 1
76  DELETE FROM nota
77  WHERE id_curso = 1;
78
79  — Eliminar el registro de estudio con ID 1
80  DELETE FROM estudio
81  WHERE id_curso = 1;
82
83  — Eliminar el registro de clase con ID 1
84  DELETE FROM clase
85  WHERE id_curso = 1;
86
87  — Eliminar el registro de familia con ID 1
88  DELETE FROM familia
89  WHERE id_alumno = 1;

```

## 4.5. Código SQL + Resultados: Vistas

```

1  CREATE VIEW promedio_ult_por_curso AS
2  SELECT id_curso , AVG ( promedio_ult_semestre ) as promedio_ult
3  FROM nota
4  GROUP BY id_curso ;
5
6  CREATE VIEW promedio_grado_por_curso AS
7  SELECT id_curso , AVG ( promedio_grado ) as promedio_grado
8  FROM nota

```

```

9  GROUP BY id_curso ;
10
11
12  CREATE VIEW curso_y_alumno AS
13  SELECT c . id_curso , a . id_alumno
14  FROM id_curso c
15  JOIN alumno a ON c . alumno = a . id_alumno ;

```

#### 4.6. Código SQL + Resultados: Triggers

```

1  CREATE OR REPLACE TRIGGER actualizar_promedio
2  AFTER INSERT ON nota
3  FOR EACH ROW
4  BEGIN
5  UPDATE alumno
6  SET promedio = ( SELECT AVG( promedio_ult_semestre )
7  FROM nota
8  WHERE id_alumno = : NEW . id_alumno )
9  WHERE id_alumno = : NEW . id_alumno ;
10 END ;
11
12
13 CREATE OR REPLACE TRIGGER verificar_edad
14 BEFORE INSERT ON alumno
15 FOR EACH ROW
16 BEGIN
17 IF : NEW . edad < 18 THEN
18 RAISE_APPLICATION_ERROR ( -20001 ,
19   El alumno debe ser mayor de 15 años . ) ;
20 END IF ;
21 END ;
22
23 CREATE OR REPLACE TRIGGER actualizar_numero_estudiantes
24 AFTER INSERT ON estudio
25 FOR EACH ROW
26 BEGIN
27 UPDATE clase
28 SET asistencia_clase = asistencia_clase + 1
29 WHERE id_curso = : NEW . curso ;
30 END ;

```

#### 4.7. Código SQL + Resultados: Funciones

```

1  CREATE OR REPLACE FUNCTION obtener_promedio_alumno
2  ( p_id_alumno INTEGER ) RETURN FLOAT promedio FLOAT ;
3  BEGIN
4  SELECT AVG ( promedio_grado ) INTO promedio
5  FROM nota
6  WHERE id_alumno = p_id_alumno ;
7  RETURN promedio ;
8  END ;
9
10 CREATE OR REPLACE FUNCTION obtener_id_curso

```

```

11  ( p-id_alumno INTEGER ) RETURN VAR28 id_curso VARCHAR (255) ;
12  BEGIN
13  sSELECT pc . id_curso INTO id_curso
14  FROM curso c
15  JOIN alumno a ON c . id_curso = a . curso
16  WHERE a . id_alumno = p-id_alumno ;
17
18  RETURN id_curso ;
19  END ;

```

#### 4.8. Código SQL + Resultados: procedimientos almacenados

```

1  CREATE PROCEDURE insert_alumno(
2      IN id_alumno INTEGER,
3      IN edad INTEGER,
4      IN sexo VARCHAR(10),
5      IN tipo-grado VARCHAR(10)
6  )
7  AS
8  BEGIN
9      INSERT INTO alumno (id_alumno, edad, sexo, tipo-grado)
10     VALUES (id_alumno, edad, sexo, tipo-grado);
11  END;
12
13  CREATE PROCEDURE update-sexo_alumno(
14      IN id_alumno INTEGER,
15      IN sexo VARCHAR(10)
16  )
17  AS
18  BEGIN
19      UPDATE alumno
20      SET sexo = sexo
21      WHERE id_alumno = id_alumno;
22  END;
23
24  CREATE PROCEDURE delete_alumno(
25      IN id_alumno INTEGER
26  )
27  AS
28  BEGIN
29      DELETE FROM alumno
30      WHERE id_alumno = id_alumno;
31  END;

```

## 5. Bases de Datos No-SQL

En esta sección, se presenta un modelo NoSQL que se desarrolló a partir de una base de datos SQL preexistente. El objetivo es explicar la organización de las lecturas en la base de datos NoSQL, en particular utilizando MongoDB.

### 5.1. Diagrama de la Base de Datos NoSQL

Inicialmente, se creó un diagrama general para visualizar la organización de las lecturas que se realizarían en el Sistema de Gestión de Bases de Datos (SMBD) seleccionado, en este caso, MongoDB. La Figura 1 muestra este diagrama de lectura NoSQL:

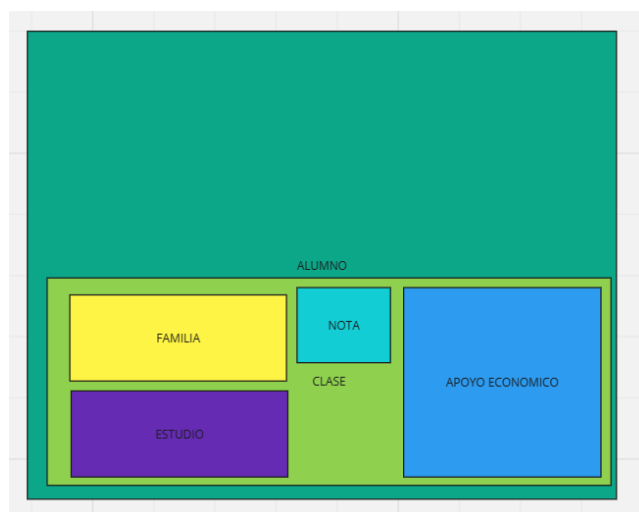


Figura 1: Diagrama de Lectura NoSQL

La Figura 1 presenta el Diagrama de Lectura NoSQL, que corresponde al Meta-Modelo Físico de nuestra base de datos. Este diagrama se centra en la organización de los datos en MongoDB, proporcionando detalles específicos sobre cómo se almacenan y se pueden acceder los datos en el nivel de implementación. Aquí, se esquematizan las colecciones, documentos y las relaciones entre ellos, lo que permite una representación visual de cómo se traducen los conceptos lógicos a estructuras físicas en la base de datos NoSQL.

El diagrama ilustra cómo se manejan las condiciones generales y por clase para cada alumno. Esto se logra aprovechando la flexibilidad de modelado sin filas y columnas característica de las bases de datos NoSQL. Para una explicación más detallada, se ha desarrollado un diagrama que utiliza los mismos colores que el diagrama de lectura NoSQL (Figura 2).

La Figura 2, denominada Diagrama NoSQL Detallado, representa el Meta-Modelo Conceptual de nuestra base de datos NoSQL. A este nivel, se abstrae la estructura

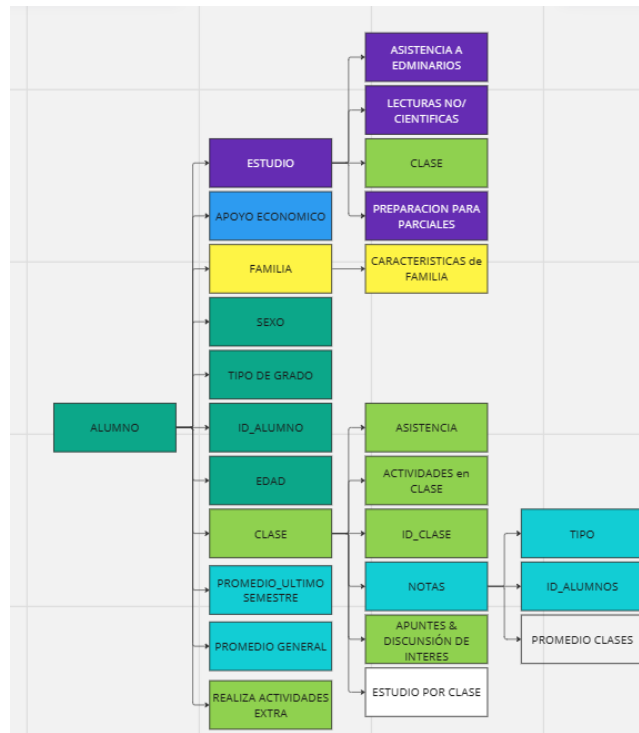


Figura 2: Diagrama NoSQL Detallado

física y se enfoca en las entidades, relaciones y flujos de información en un formato más comprensible y conceptual. En este diagrama, destacamos cómo los elementos clave de la base de datos se relacionan entre sí. Proporciona una visión más clara de la organización de los datos y cómo estos datos se asocian con conceptos más amplios en el contexto de la aplicación. Este enfoque conceptual es fundamental para la comprensión y el diseño efectivo de la base de datos, donde se ilustra de una mejor manera como se puede acceder a lo que llamaríamos entidades en SQL que pueden ver en la sección **4.2 Diagrama modelo datos**.

La Figura 3, conocida como Meta-Modelo Lógico, representa la capa intermedia entre el modelo conceptual y la implementación física de la base de datos. En este diagrama, se enfatiza la estructura de alto nivel de la base de datos y se definen las relaciones esenciales entre las entidades y los atributos. A través de esta representación, se capturan las conexiones lógicas que existen entre los componentes clave de la base de datos, lo que permite comprender cómo se relacionan los datos y cómo los usuarios interactúan con ellos en un nivel más abstracto. El Meta-Modelo Lógico es crucial para definir una estructura de base de datos que se adapte a las necesidades de la aplicación sin entrar en detalles específicos de la implementación física.

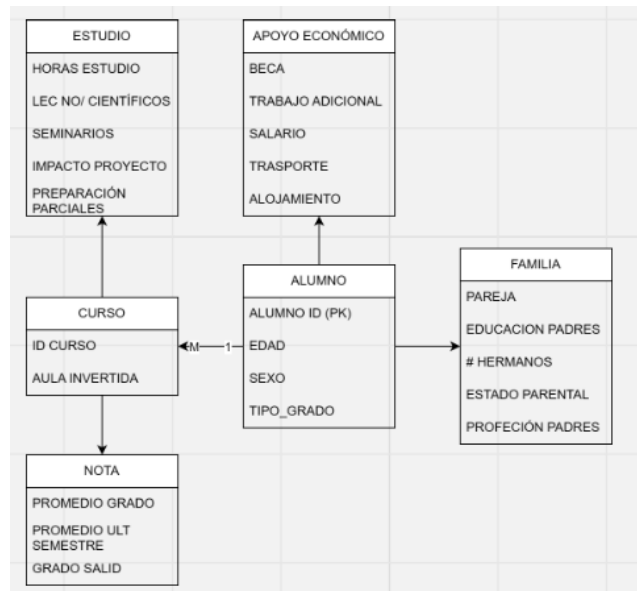


Figura 3: Meta-Modelo Lógico

## 5.2. SMBD Utilizado para la Base de Datos NoSQL

Para implementar este modelo NoSQL, hemos utilizado MongoDB. A continuación, presentamos un ejemplo de cómo ingresamos datos para un alumno en MongoDB. Esto ilustra cómo la información se organiza en documentos, que son las unidades fundamentales en MongoDB. Cada documento almacena detalles sobre un estudiante, incluyendo su identificación, edad, género, tipo de grado, apoyo económico, clases, notas, estudios y detalles familiares.

El código que sigue a continuación representa un ejemplo de datos de un estudiante almacenados en MongoDB. Cada campo tiene una descripción, y esto es solo un ejemplo de cómo se pueden estructurar los datos.

Este enfoque nos permite aprovechar la flexibilidad y escalabilidad de MongoDB en comparación con un sistema de base de datos relacional tradicional, lo que es esencial para cumplir con los requisitos específicos de nuestro proyecto.

Esta mejora ayuda a proporcionar contexto y una comprensión más profunda del contenido que se está presentando, lo que es esencial en un entorno técnico o académico.

```

1  {
2    "alumno_id": 1,
3    "edad": 20,
4    "Sexo": "M",
5    "tipo_grado": "Licenciatura",
6    "apoyoeconomico": {
7      "beca": "Completa",
8      "trabajo_adicional": 1,

```

```

9      "Salario": "$1000",
10     "transporte": "Autob s",
11     "alojamiento": "Residencia"
12   },
13   "clases": [
14     {
15       "id_curso": 101,
16       "aula_invertida": "S ",
17       "notas": [
18         {
19           "promedio_ult_semestre": "85",
20           "promedio_grado": "88",
21           "grado_salida": "Bachiller"
22         }
23       ],
24       "estudios": [
25         {
26           "horas_estudio": "20 horas por semana",
27           "frecu lec_no_cientifico": "3 horas por semana",
28           "frecu lec_cientifico": "5 horas por semana",
29           "asistencias_seminarios": "10",
30           "impacto_proyec": "Alto",
31           "preparacion_parcial_1": "Buena",
32           "preparacion_parcial_2": "Excelente"
33         }
34       ]
35     },
36     {
37       "id_curso": 102,
38       "aula_invertida": "No",
39       "notas": [
40         {
41           "promedio_ult_semestre": "78",
42           "promedio_grado": "82",
43           "grado_salida": "Bachiller"
44         }
45       ],
46       "estudios": [
47         {
48           "horas_estudio": "15 horas por semana",
49           "frecu lec_no_cientifico": "2 horas por semana",
50           "frecu lec_cientifico": "4 horas por semana",
51           "asistencias_seminarios": "8",
52           "impacto_proyec": "Medio",
53           "preparacion_parcial_1": "Regular",
54           "preparacion_parcial_2": "Bueno"
55         }
56       ]
57     }
58   ],
59   "familia": {
60     "pareja": "No",
61     "educacion_madre": "Bachiller",
62     "educacion_padre": "Licenciatura",
63     "num_hermanos": 2,
64     "estado_parental": "Casados",
65     "profecion_madre": "Enfermera",

```

```
66     "profesion_padre": "Ingeniero"  
67 }  
68 }
```



- 6. **Aplicación de ETL (Extract, Transform, Load)  
y Bodega de Datos** (*Tercera entrega*)
  - 6.1. **Ejemplo de aplicación de ETL y Bodega de Datos**  
(*Tercera entrega*)
  - 6.2. **Automatización de Datos** (*Tercera entrega*)
  - 6.3. **Integración de Datos** (*Tercera entrega*)

## 7. Próximos pasos (*Tercera entrega*)

## 8. Lecciones aprendidas (*Tercera entrega*)

## 9. Bibliografía