

Creación de una BBDD relacional

Equipo No Mongo

Full Stack & Data Science - Septiembre 23

Contenidos

- 1 - Introducción
- 2 - Equipo y contexto
- 3 - Primeros pasos
- 4 - Modelos y normalización
- 5 - Querys
 - Creación
 - Introducción
- 6 - Testing
- 7 - Next Steps

Una primera aproximación a los requisitos generales del proyecto.

Creación de una base de datos relacional sobre unos datasets proporcionados.

Requisitos

- Normalización de datos.
- Modelo entidad-relación.
- Modelo lógico.
- Despliegue de la bbdd.
- Querys de consulta.

El equipo

Data Science

Daniel Gouveia

Daniel Manso

Alex Campos

Full Stack

Cristina Mateo

Adrián Ortiz

Softwares

PgAdmin 4

PostgreSQL

Elephant SQL

Python

Numpy & Pandas

VSC

Excel

Primeros pasos

Con las herramientas de Elephant y PgAdmin se hace un seteo de un servidor online y un visualizador.



Modelo ER

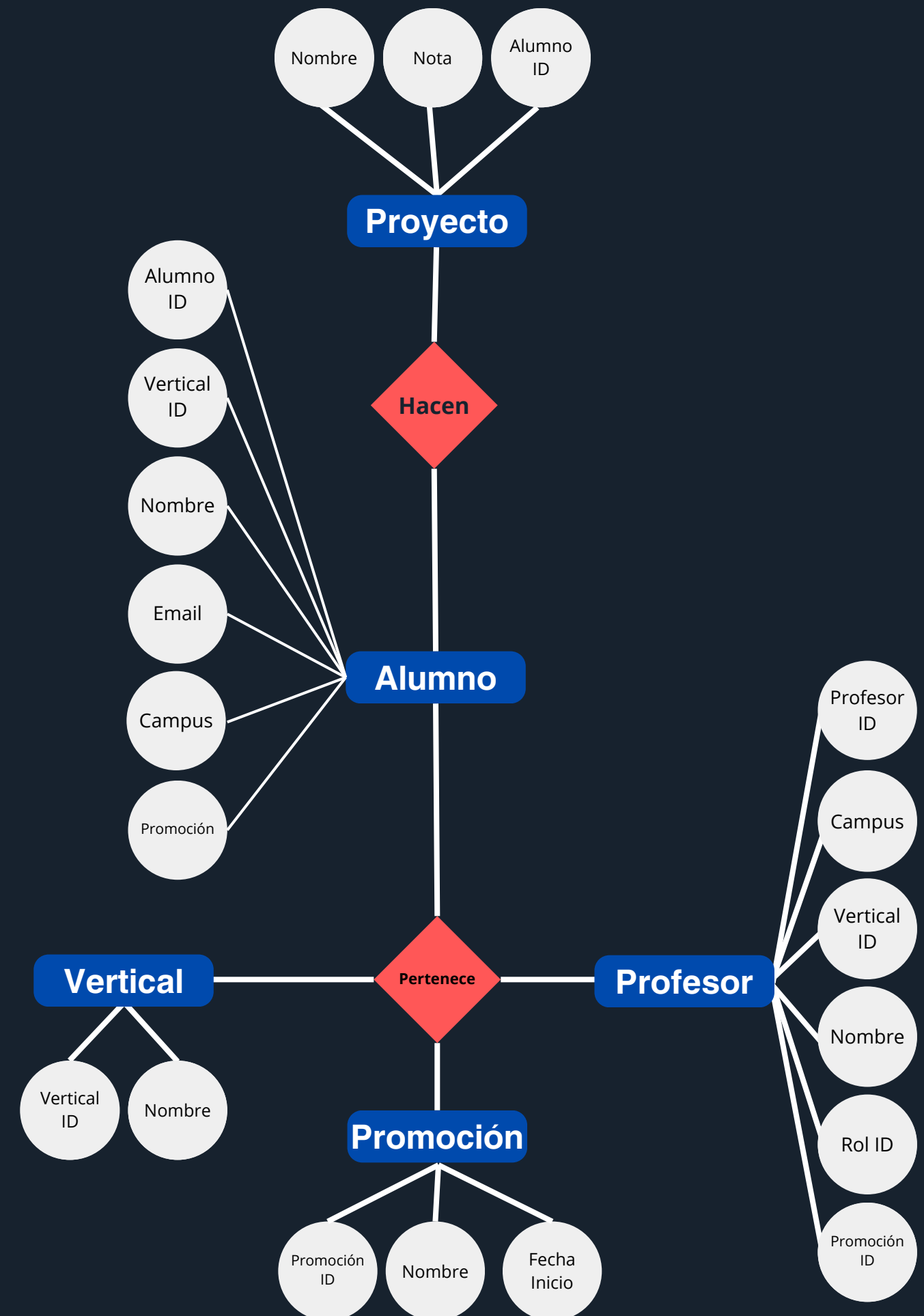
Comenzamos con un estudio de las posibles **entidades** y **relaciones** para empezar a crear el esqueleto de los modelos.

La normalización la hacemos parte del proceso de creación de los diagramas.

Normalización

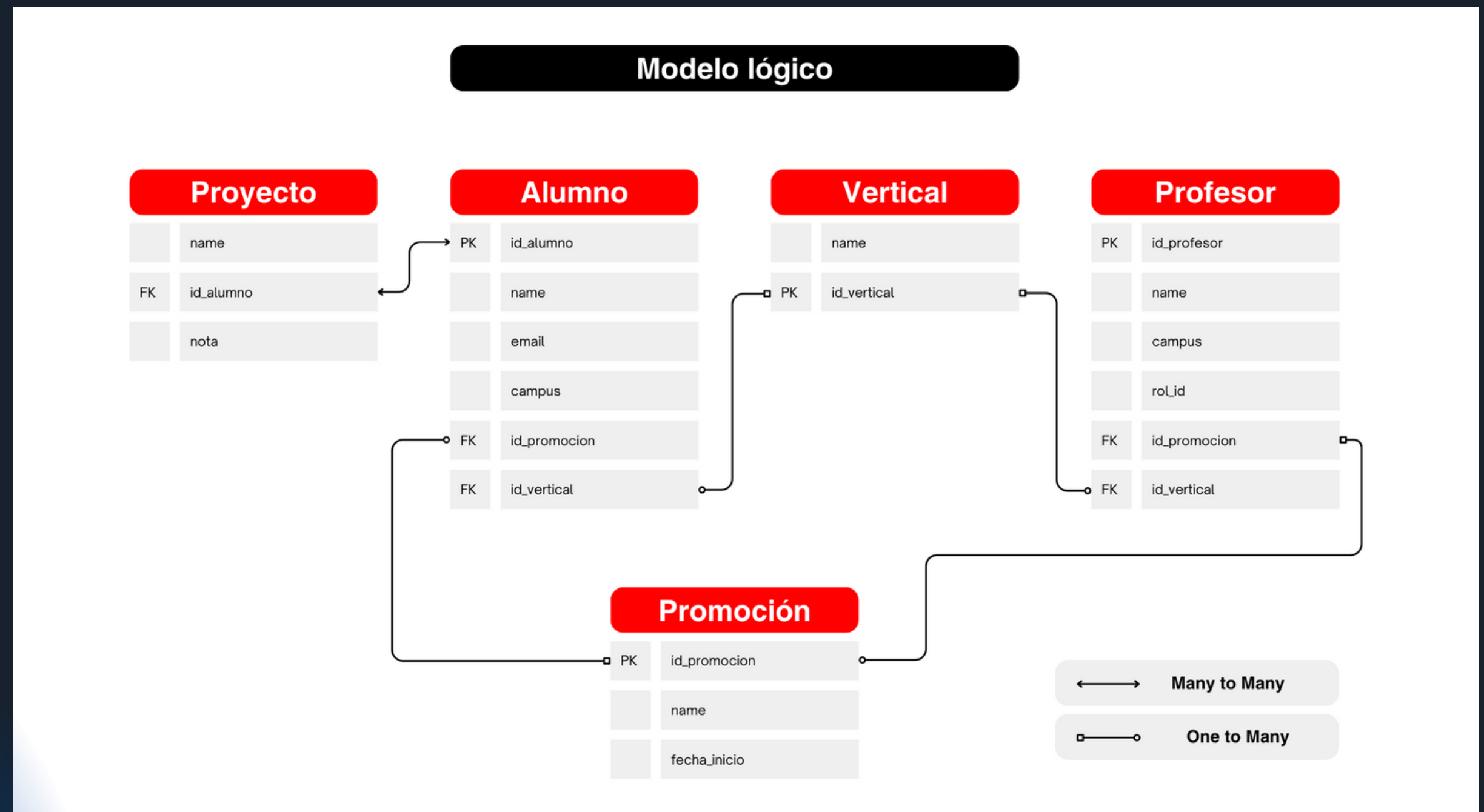
El proceso de **normalización** ha sido un **proceso abstracto** en el que hemos ido pasando de una forma a otra de **manera orgánica** a lo largo de la creación de los modelos

Diagrama ER



Modelo lógico

A partir del modelo ER anterior creamos el modelo lógico, definiendo las PK y FK de cada entidad y el tipo de relaciones que hay entre ellas.



Querys – creación

En PgAdmin usamos las **querys** aprendidas en clase para crear la tabla, definiendo el nombre de la misma y las columnas que contiene, así como los “**Constraints**” que ponemos a cada columna.

Se fueron modificando los errores sobre la marcha.

```
CREATE TABLE alumnos (  
  id_alumno serial NOT NULL PRIMARY KEY,  
  name varchar(45) NOT NULL,  
  email varchar(45) NOT NULL UNIQUE,  
  id_vertical int,  
  category varchar(15),  
  FOREIGN KEY (id_vertical) REFERENCES vertical(id_vertical),  
  campus varchar(10) NOT NULL,  
  promocion varchar(10) NOT NULL,  
  fecha_inicio varchar(10) NOT NULL  
);
```

-- Crear tabla proyectos

```
CREATE TABLE proyectos (  
  id_proyecto serial NOT NULL PRIMARY KEY,  
  name varchar(45) NOT NULL,  
  id_alumno int,  
  category varchar(15),  
  FOREIGN KEY (id_alumno) REFERENCES alumnos(id_alumno),  
  nota varchar(10) NOT NULL  
);
```

-- Crear tabla profes

```
CREATE TABLE profesores (  
  id_profe serial NOT NULL PRIMARY KEY,  
  name varchar(45) NOT NULL
```


Querys – introducción

En la misma plataforma se introducen los datos limpios usando los comandos de query de SQL necesarios.

Se define la tabla y columna en la que se introducen y se añaden los valores.

```
-- Insertar datos en tabla promocion
```

```
INSERT INTO promocion(id_promocion,name,fecha_inicio)
VALUES
('DS2309','Sep23','2023/09/18');
```

```
-- Insertar datos en tabla promocion
```

```
INSERT INTO promocion(id_promocion,name,fecha_inicio)
VALUES
('FS2309','Sep23','2023/09/18'),
('FS2402','Feb24','2024/02/12'),
('DS2402','Feb24','2024/02/12')
;
```

```
-- Insertar datos en tabla vertical
```

```
INSERT INTO vertical(id_vertical,name)
VALUES
('FS','Full Stack'),
('DS','Data Science')
;
```

Querys – introducción

```
-- Insertar datos en tabla alumnos
INSERT INTO alumnos(name,email,campus,id_promocion,id_vertical)
VALUES
('Jafet Casals','Jafet_Casals@gmail.com','Madrid','DS2309','DS'),
('Jorge Manzanares','Jorge_Manzanares@gmail.com','Madrid','DS2309','DS'),
('Onofre Adadia','Onofre_Adadia@gmail.com','Madrid','DS2309','DS'),
('Merche Prada','Merche_Prada@gmail.com','Madrid','DS2309','DS'),
('Pilar Abella','Pilar_Abella@gmail.com','Madrid','DS2309','DS'),
('Leoncio Tena','Leoncio_Tena@gmail.com','Madrid','DS2309','DS'),
('Odalys Torrijos','Odalys_Torrijos@gmail.com','Madrid','DS2309','DS'),
('Eduardo Caparrós','Eduardo_Caparrós@gmail.com','Madrid','DS2309','DS'),
('Ignacio Goicoechea','Ignacio_Goicoechea@gmail.com','Madrid','DS2309','DS'),
('Clementina Santos','Clementina_Santos@gmail.com','Madrid','DS2309','DS'),
('Abraham Vélez','Abraham_Vélez@gmail.com','Madrid','DS2309','DS'),
('Maximiliano Menéndez','Maximiliano_Menéndez@gmail.com','Madrid','DS2309','DS'),
('Eli Casas','Eli_Casas@gmail.com','Madrid','DS2309','DS')
;
```

Testing

Para comprobar que los datos son accesibles, hacemos algunas queries de prueba y observamos los resultados que nos devuelve.

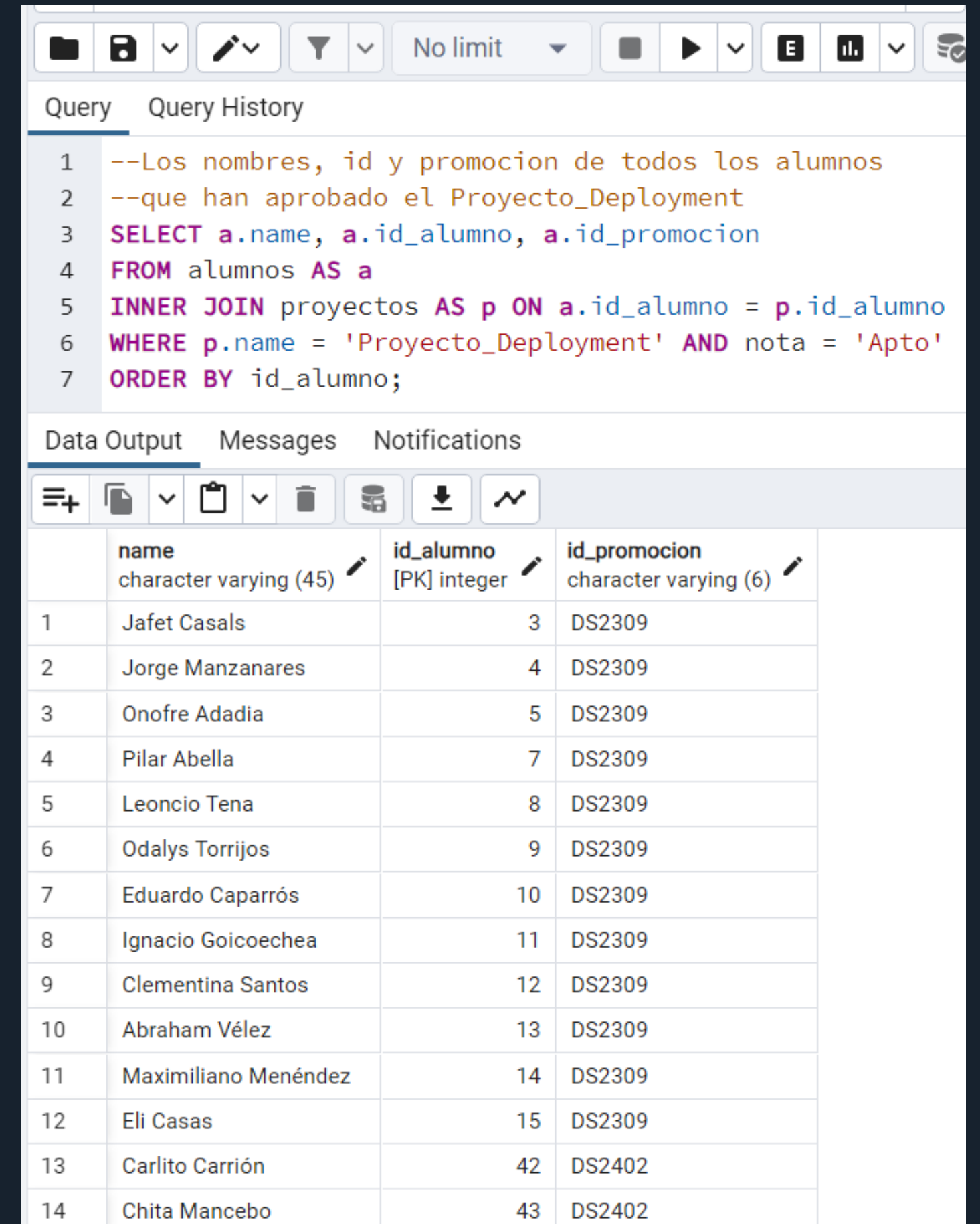
The screenshot shows a database query tool interface. At the top, there is a toolbar with icons for file operations, filters, and execution. Below the toolbar, there are tabs for 'Query' and 'Query History'. The 'Query' tab is active, displaying a SQL query with line numbers 1 through 5. The query is a SELECT statement that retrieves the names and IDs of students from the 'alumnos' table, filtered by the promotion 'DS2309'. Below the query editor, there are tabs for 'Data Output', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing a table with two columns: 'name' (character varying (45)) and 'id_alumno' ([PK] integer). The table contains 15 rows of data, listing students and their IDs. The interface also includes a toolbar for the data output table with icons for expand, copy, delete, and other actions.

```
1 --Todos los nombres y los id de los alumnos de
2 --la promoción de septiembre de data
3 SELECT name, id_alumno
4 FROM alumnos
5 WHERE id_promocion = 'DS2309';
```

	name character varying (45)	id_alumno [PK] integer
1	Jafet Casals	3
2	Jorge Manzanares	4
3	Onofre Adadia	5
4	Merche Prada	6
5	Pilar Abella	7
6	Leoncio Tena	8
7	Odalys Torrijos	9
8	Eduardo Caparrós	10
9	Ignacio Goicoechea	11
10	Clementina Santos	12
11	Abraham Vélez	13
12	Maximiliano Menéndez	14
13	Eli Casas	15
14	Daniela Falcó	76
15	Anita Heredia	77

Testing

Para comprobar que los datos son accesibles, hacemos algunas queries de prueba y observamos los resultados que nos devuelve.



The screenshot shows a database query editor interface. At the top, there's a toolbar with icons for file operations, query execution, and settings. Below the toolbar, the 'Query' tab is active, displaying a SQL query. The query is a SELECT statement that joins the 'alumnos' table with the 'proyectos' table, filtering for the project 'Proyecto_Deployment' and the status 'Apto'. The results are ordered by 'id_alumno'.

```
1 --Los nombres, id y promocion de todos los alumnos
2 --que han aprobado el Proyecto_Deployment
3 SELECT a.name, a.id_alumno, a.id_promocion
4 FROM alumnos AS a
5 INNER JOIN proyectos AS p ON a.id_alumno = p.id_alumno
6 WHERE p.name = 'Proyecto_Deployment' AND nota = 'Apto'
7 ORDER BY id_alumno;
```

Below the query, the 'Data Output' tab is active, showing a table with 14 rows of results. The table has four columns: 'name', 'id_alumno', and 'id_promocion'. The 'name' column is of type 'character varying (45)', 'id_alumno' is of type '[PK] integer', and 'id_promocion' is of type 'character varying (6)'. The results list 14 students who have passed the 'Proyecto_Deployment' project.

	name character varying (45)	id_alumno [PK] integer	id_promocion character varying (6)
1	Jafet Casals	3	DS2309
2	Jorge Manzanares	4	DS2309
3	Onofre Adadia	5	DS2309
4	Pilar Abella	7	DS2309
5	Leoncio Tena	8	DS2309
6	Odalys Torrijos	9	DS2309
7	Eduardo Caparrós	10	DS2309
8	Ignacio Goicoechea	11	DS2309
9	Clementina Santos	12	DS2309
10	Abraham Vélez	13	DS2309
11	Maximiliano Menéndez	14	DS2309
12	Eli Casas	15	DS2309
13	Carlito Carrión	42	DS2402
14	Chita Mancebo	43	DS2402

Demo en directo

Next Steps

Algunas de las ideas que se nos han ido ocurriendo durante y tras nuestro aprendizaje realizando este proyecto que por cuestiones de tiempo no hemos podido implementar.

- Asegurar escalabilidad proyecto
- Tratar el atributo Campus como entidad para mejorar las relaciones entre entidades.
- Añadir id_vertical como atributo en la columna proyectos para mejorar su relación
- Estudiar mejor el funcionamiento de la escuela y añadir más entidades como por ejemplo:
 - Modalidad (Full time o Part Time)
 - Formato (Remoto online, Presencial, Remoto diferido)
- Y más

Gracias por vuestra atención

Equipo “No-Mongo”

Full Stack & Data Science – Septiembre 23

3 de Noviembre de 2023