**Escuela Normal Superior**

**“Gral. Manuel Belgrano”**

**Tecnicatura Superior en Desarrollo de Software**

**Materia: Base de Datos II**

**Año 2024**

**Curso: 3°**

**Alumno: QUIROGA Carlos Eduardo**

**Trabajo Práctico N° 1**

Contenido

[INTRODUCCIÓN 3](#_Toc164276119)

[¿Qué es una base de datos? 3](#_Toc164276120)

[Tipos de base de datos 3](#_Toc164276121)

[1. Bases de datos relacionales 3](#_Toc164276122)

[2. Bases de datos NO relacionales 3](#_Toc164276123)

[3. Bases de datos en la nube 4](#_Toc164276124)

[Base de Datos Relacionales: REGLAS de Transformación. 4](#_Toc164276125)

[Diseño de una Base de Datos 4](#_Toc164276126)

[Etapa 1: Análisis: 4](#_Toc164276127)

[Etapa 2: Diseño Conceptual: 4](#_Toc164276128)

[Etapa 3: Diseño Lógico: 4](#_Toc164276129)

[Etapa 4: Diseño Físico: 5](#_Toc164276130)

[Transformación del MODELO ENTIDAD-RELACION al MODELO RELACIONAL 5](#_Toc164276131)

[Relaciones de UNO a UNO (1:1) 5](#_Toc164276132)

[Modelo ENTIDAD-RELACIÓN 5](#_Toc164276133)

[Modelo RELACIONAL: 6](#_Toc164276134)

[Relaciones de UNO a MUCHOS (1:N) 6](#_Toc164276135)

[Modelo ENTIDAD-RELACIÓN: 6](#_Toc164276136)

[Modelo RELACIONAL: 7](#_Toc164276137)

[Modelo FISICO: 7](#_Toc164276138)

[Relaciones de MUCHOS a MUCHOS (M:N) 7](#_Toc164276139)

[Modelo ENTIDAD-RELACIÓN: 7](#_Toc164276140)

[Modelo RELACIONAL: 8](#_Toc164276141)

[Modelo FÍSICO: 8](#_Toc164276142)

[Modelo de Ejemplo. 9](#_Toc164276143)

[Documento de Especificación de Requisitos (SRS) – 9](#_Toc164276144)

[Software de Visualización de Triunfos de Corredores de Ciclismo 9](#_Toc164276145)

[Bibliografía 11](#_Toc164276146)

# INTRODUCCIÓN

## ¿Qué es una base de datos?

Una base de datos es una colección organizada de información, con una estructura para ser accedida y gestionada. Los datos son administrados por medio de un sistema gestor (DataBase Management System o DBMS). El más común y popular Microsoft Excel es una base de datos que nos permite guardar datos con estructuras de filas y columnas.

## Tipos de base de datos

Existen diferentes tipos de base de datos, y estos se clasifican de acuerdo a diferentes características, según la flexibilidad para realizar cambios, tenemos acá estáticas o dinámicas, también se pueden clasificar según su organización. Centraremos nuestra atención en:

1. Bases de datos relacionales
2. Bases de datos NoSQL o no relacionales
3. Bases de datos en la nube
4. Otras

### 1. Bases de datos relacionales

Son aquellas que aplican el modelo relacional y se eligen cuando los datos que van a ser almacenados tienen una estructura planificada.

Son muy útiles cuando trabajamos con datos estructurados.

Ejemplos:

* MySQL
* Microsoft SQL Server
* Oracle Database
* PostgreSQL
* IBM Db2

### 2. Bases de datos NO relacionales

Una base de datos NO-SQL (Not Only - SQL) se utilizan para almacenar datos no estructurados o semiestructurados, son más flexibles y en la actualidad son una opción que se aplica para solucionar algunas limitaciones que tiene el modelo relacional. Este tipo de bases de datos es excelente para las organizaciones que buscan almacenar datos no estructurados o semiestructurados.

- Posibilidad de realizar cambios en producción, sin afectar a las aplicaciones que se encuentran utilizándola.

**Ejemplos:**

* MongoDB
* Redis
* Apache Cassandra
* Apache CouchDB
* CouchBase

### 3. Bases de datos en la nube

Estás bases de datos se entregan como un servicio desde la nube, por lo tanto la creación, mantenimiento y escalabilidad son competencia del proveedor de este servicio.

**Ejemplos:**

* Google Firebase
* Microsoft Azure SQL Database
* Amazon Relational Database Service
* Oracle Autonomous Database.

## Base de Datos Relacionales: REGLAS de Transformación.

### Diseño de una Base de Datos

El proceso de diseñar una base de datos es complejo, y abarca desde los requerimientos iniciales de los usuarios del sistema de información a la implementación de una base de datos.

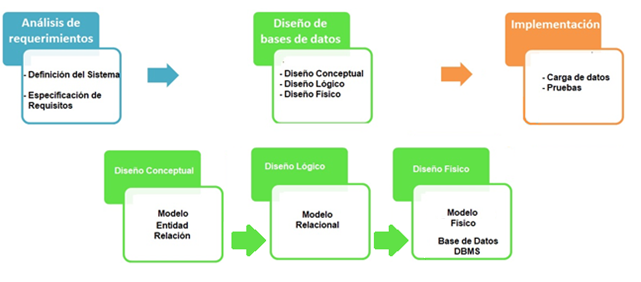
Para comenzar repasaremos las diferentes etapas y las fases que conforman el proceso de diseño de base de datos y especificaremos de manera clara los objetivos de cada una de estas etapas.

Etapa 1: Análisis: Establece el propósito de la base de datos, obtiene y organiza la información para obtener como resultado una lista de requisitos de usuarios.

Etapa 2: Diseño Conceptual: Del que resulta un esquema conceptual independiente de la tecnología que se utiliza para las siguientes etapas.

Etapa 3: Diseño Lógico: Transforma el modelo conceptual en un modelo lógico, más específicamente el modelo lógico relacional, es decir, el modelo lógico que utilizan las bases de datos relacionales.

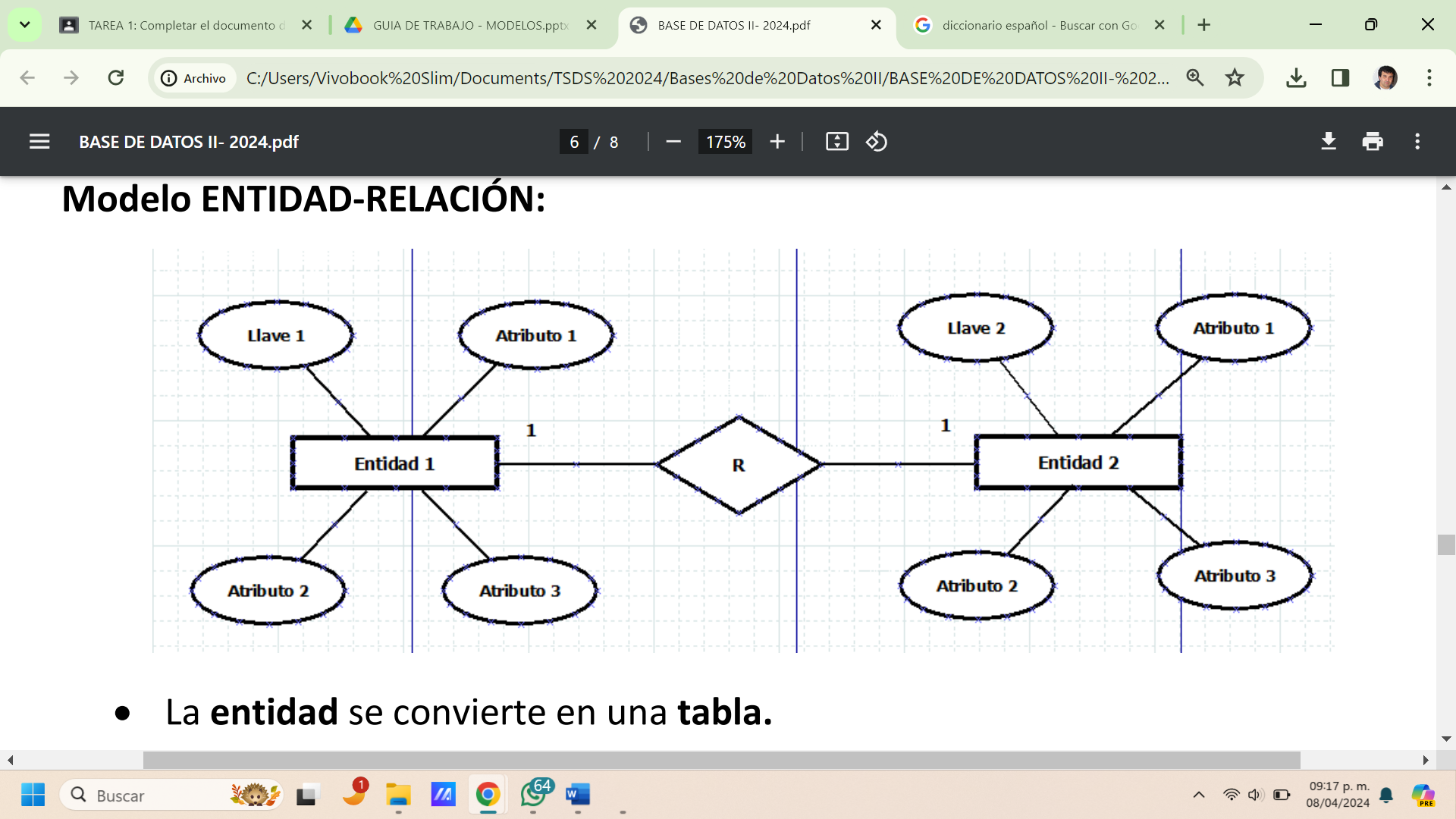
Etapa 4: Diseño Físico: Transforma el modelo lógico en un modelo físico, que nos permitirá obtener una implementación sobre un sistema gestor de bases de datos concreto.



## Transformación del MODELO ENTIDAD-RELACION al MODELO RELACIONAL

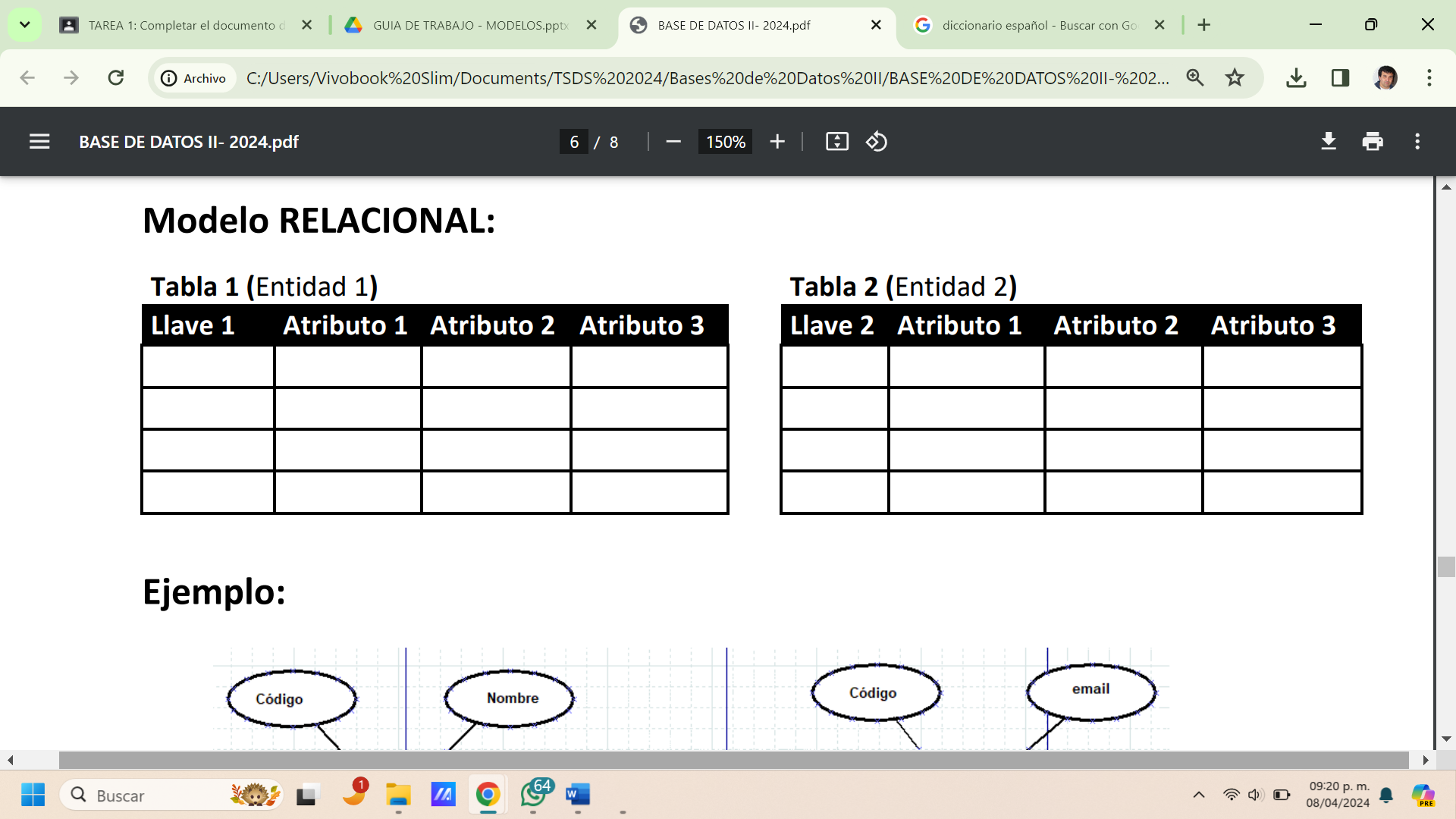
## **Relaciones de UNO a UNO (1:1)**

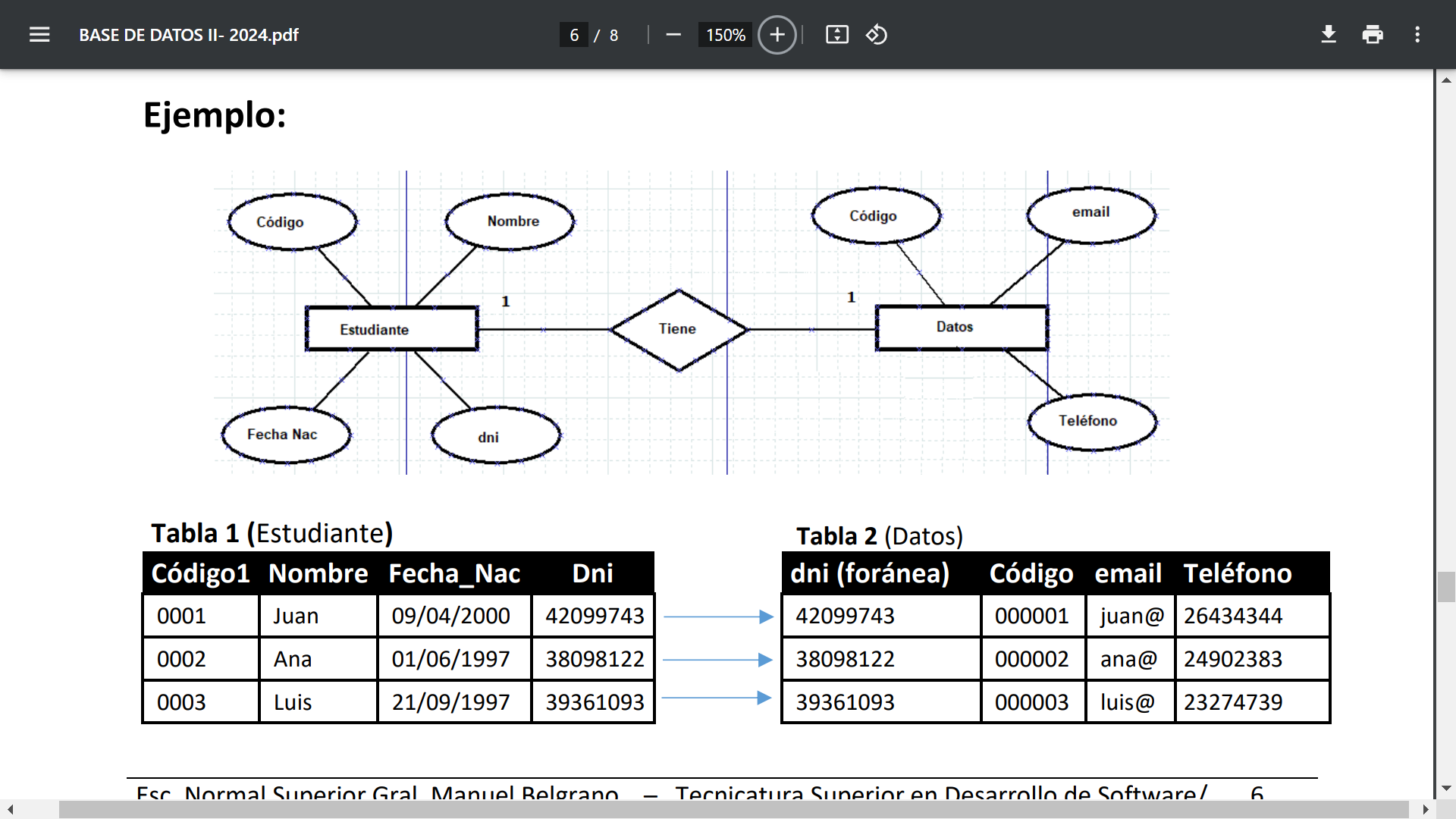
### Modelo ENTIDAD-RELACIÓN



* La entidad se convierte en una tabla.
* Cada atributo se convierte a un campo de la tabla.
* Se crea un atributo en una de las tablas que corresponde a la llave primaria de la otra. Esta es la llave foránea de la relación.

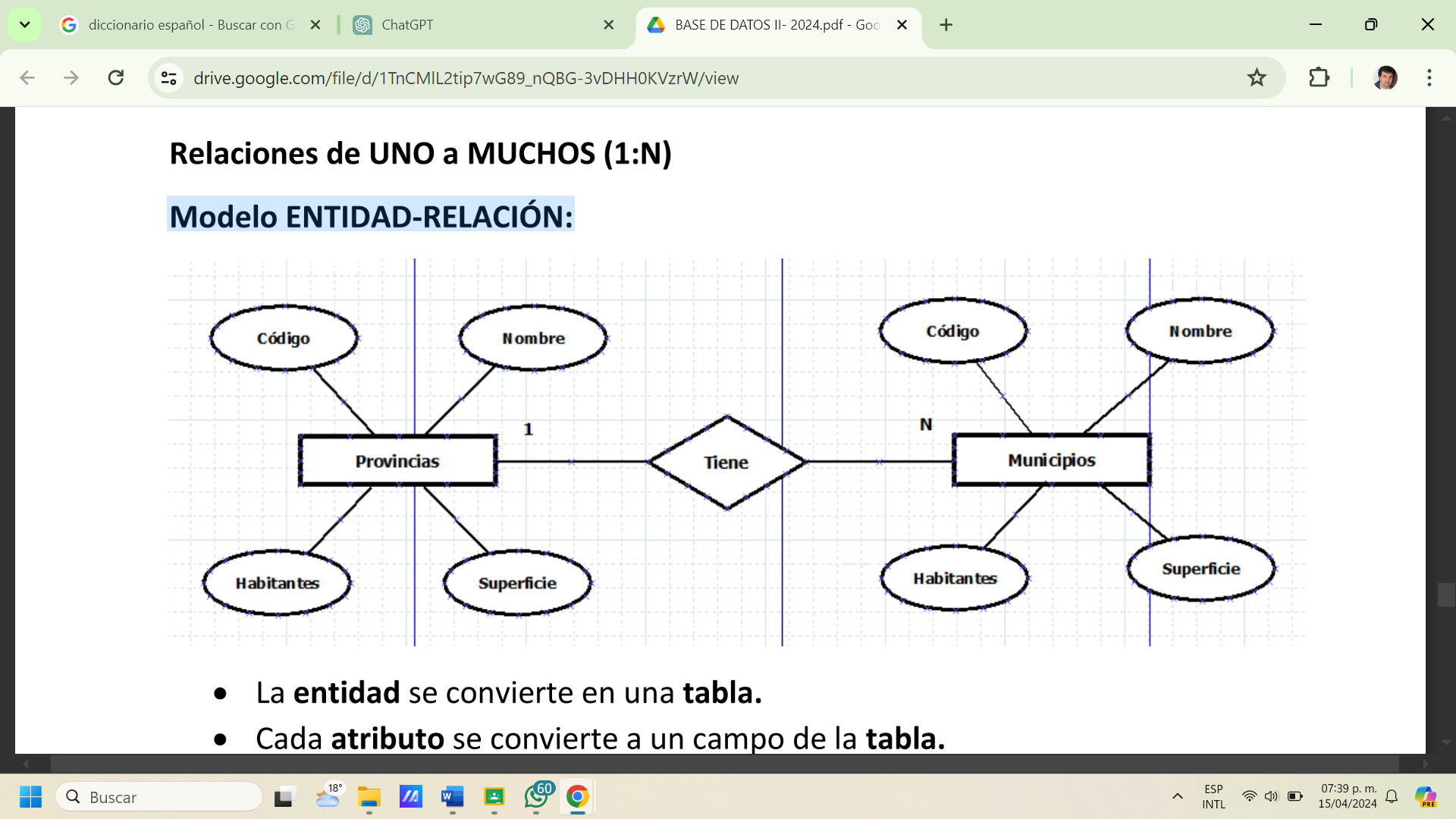
### Modelo RELACIONAL:



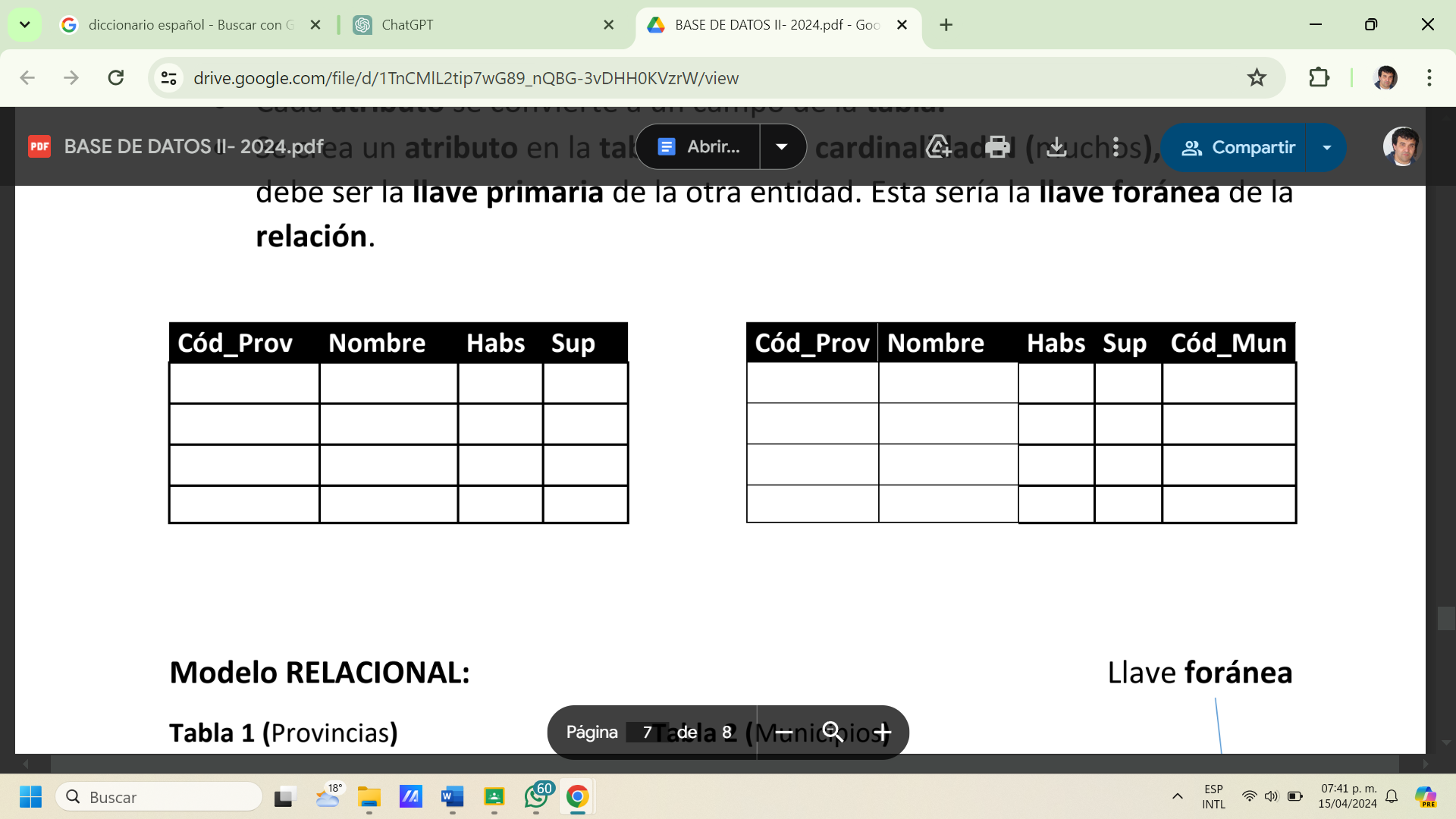


## Relaciones de UNO a MUCHOS (1:N)

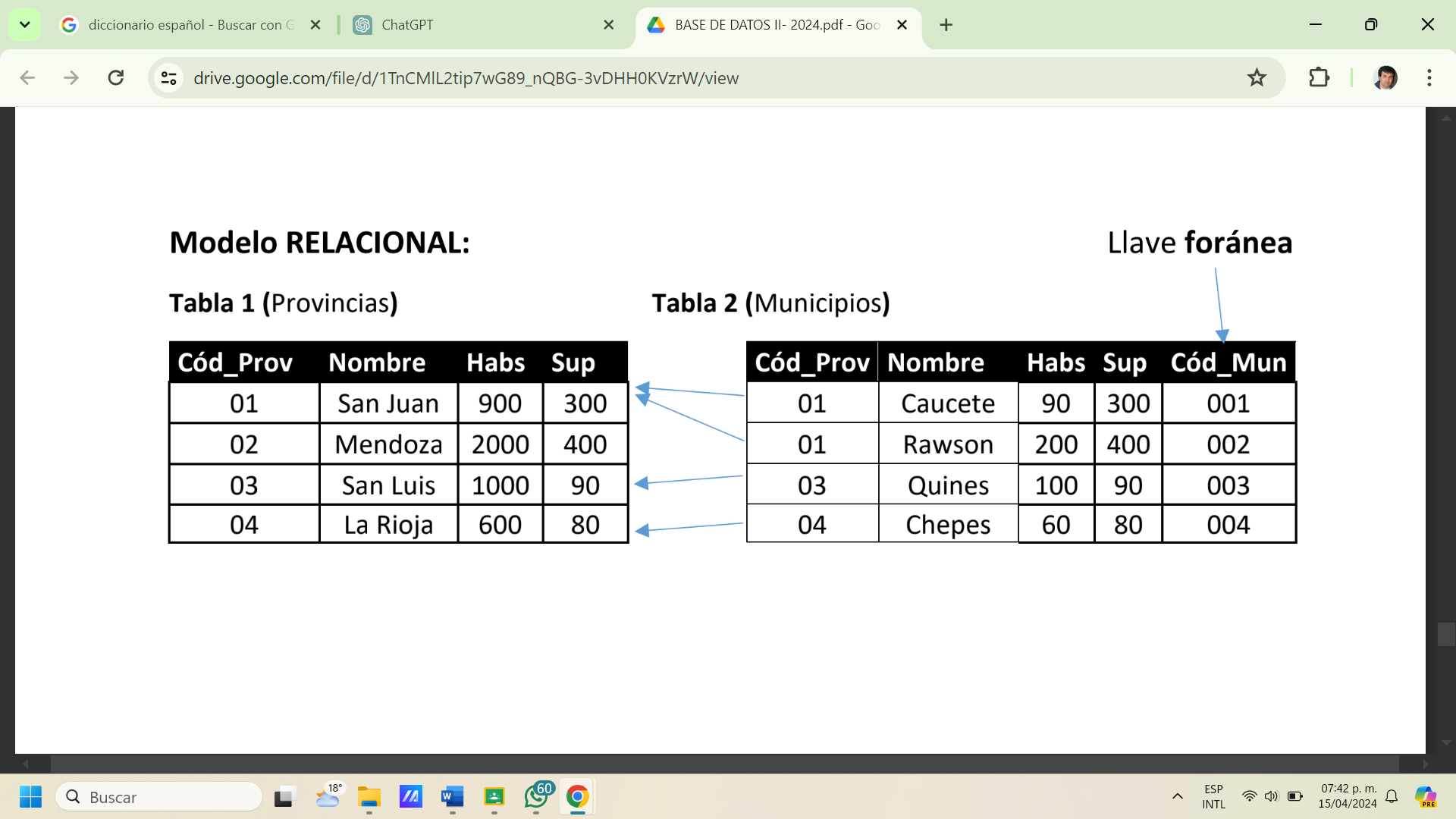
### Modelo ENTIDAD-RELACIÓN:



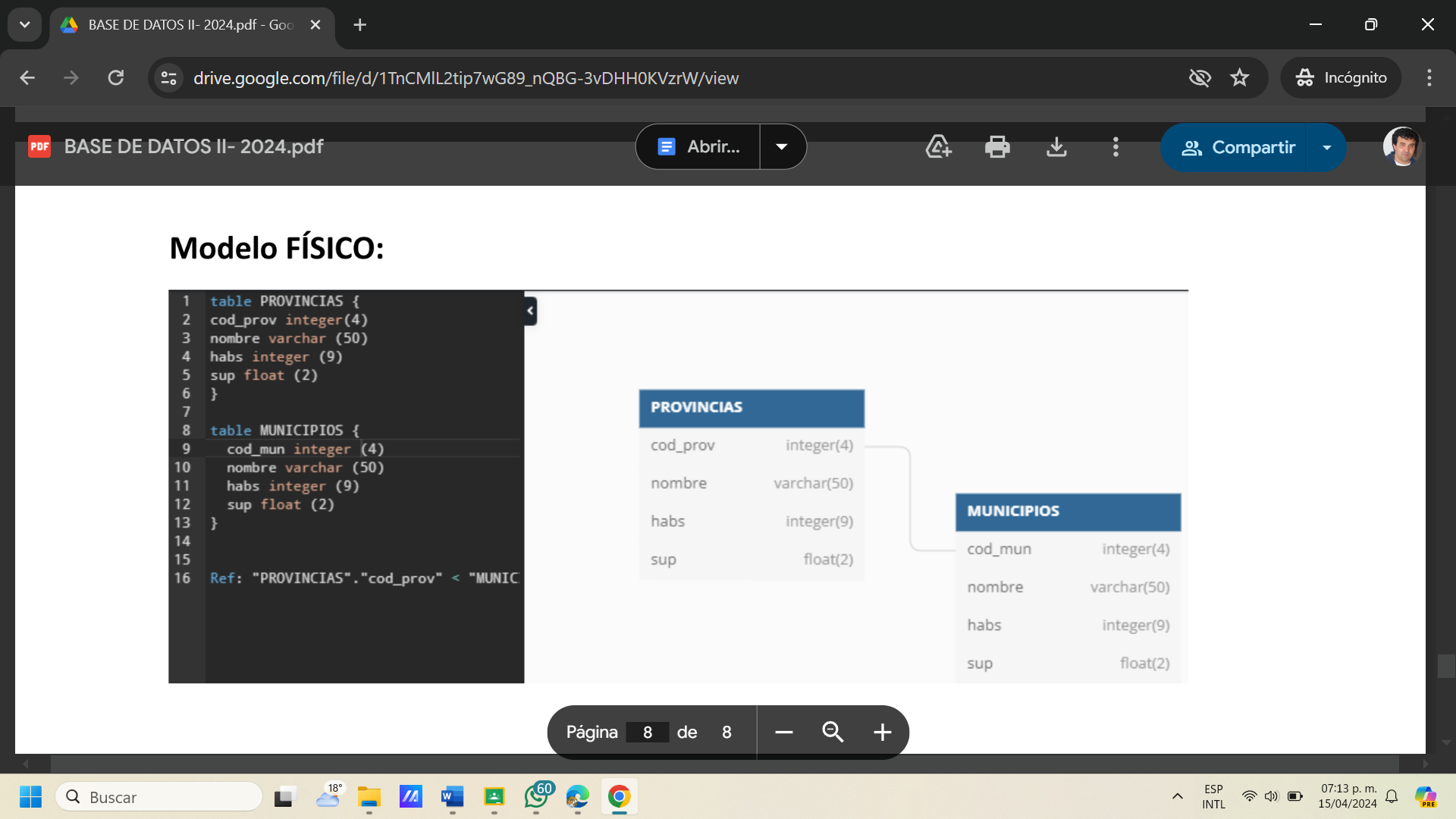
* La entidad se convierte en una tabla.
* Cada atributo se convierte a un campo de la tabla.
* Se crea un atributo en la tabla que tiene cardinalidad N (muchos), el cual debe ser la llave primaria de la otra entidad. Esta sería la llave foránea de la relación.



### Modelo RELACIONAL:



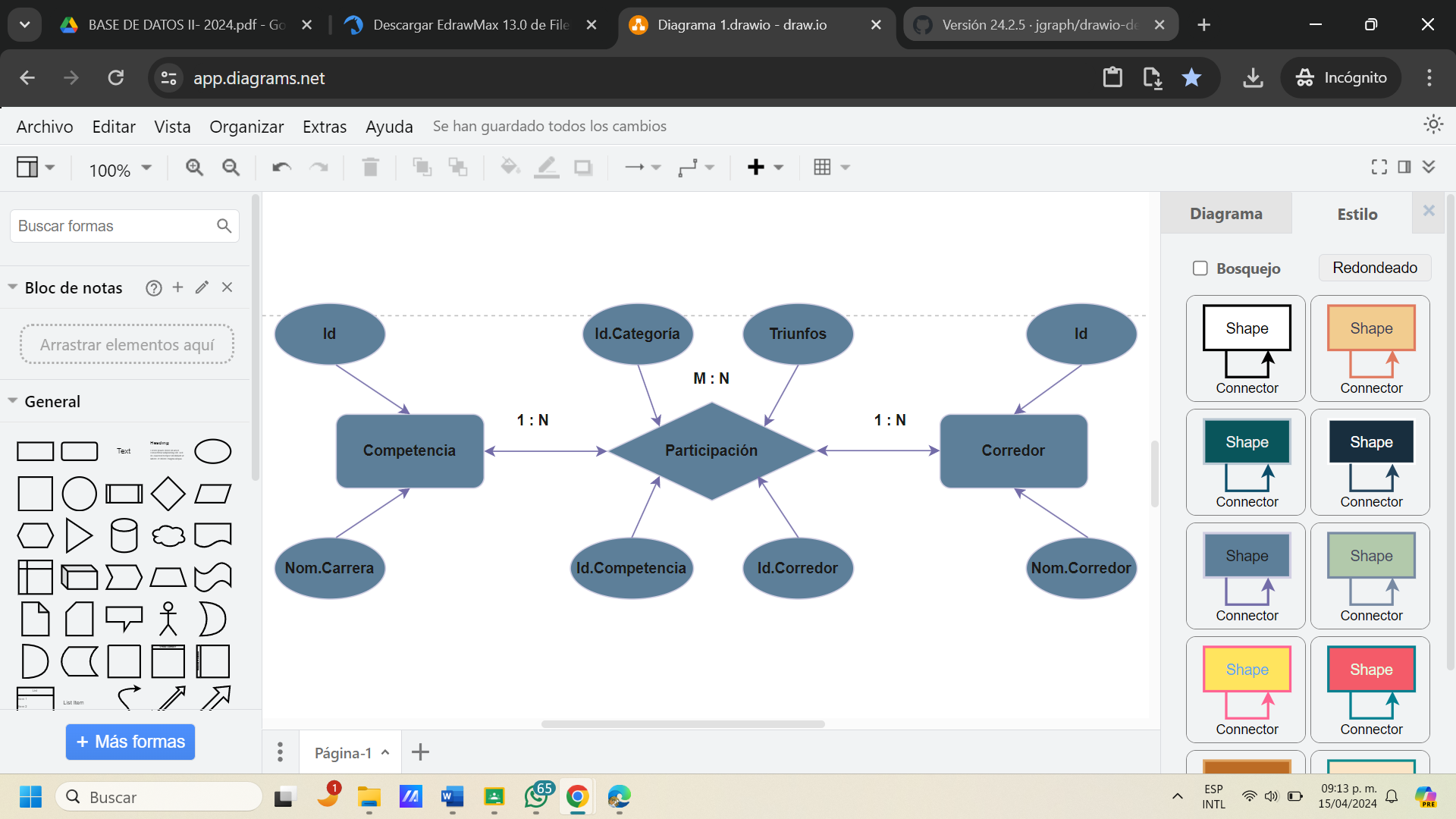
### Modelo FISICO:



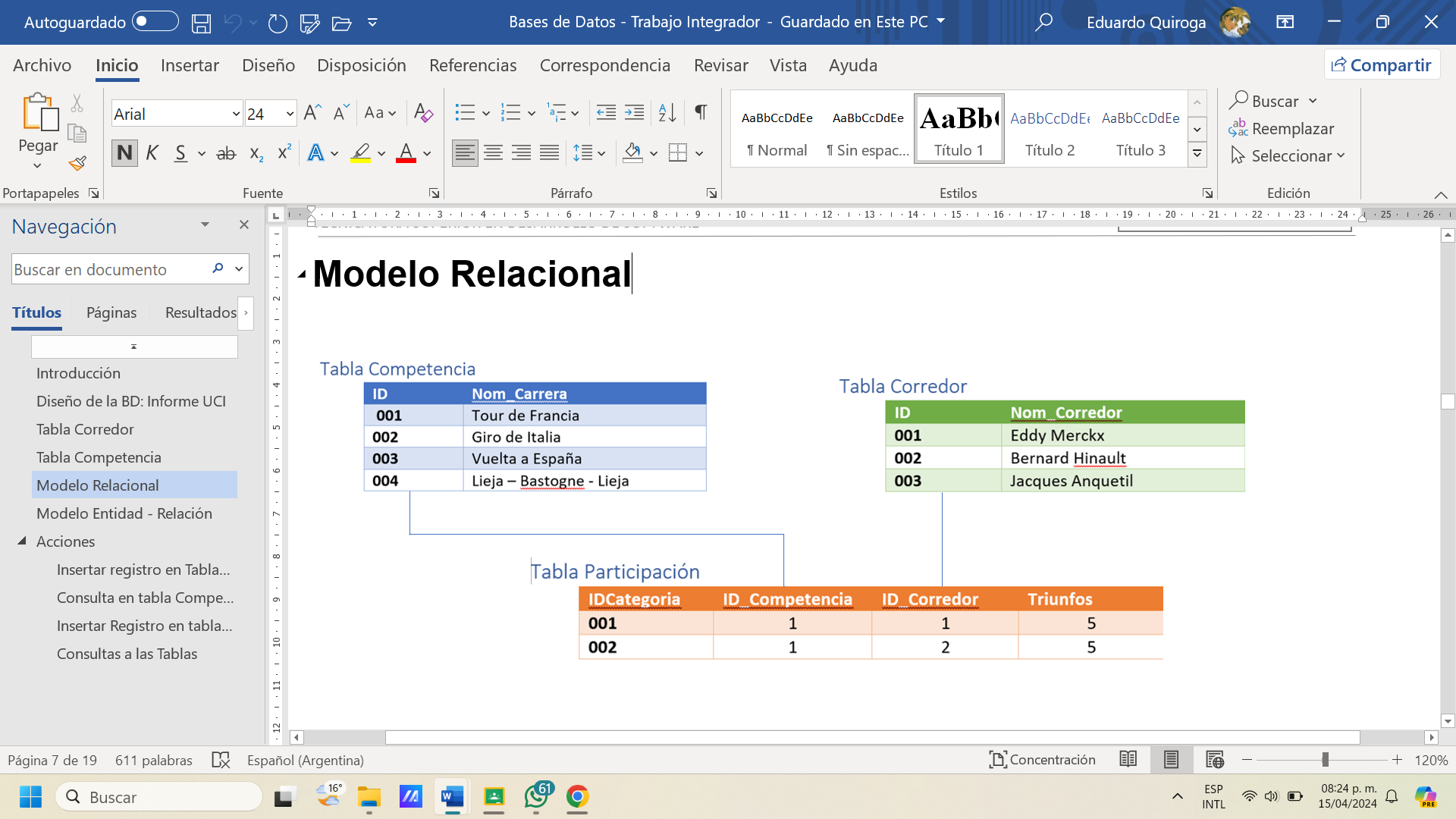
## Relaciones de MUCHOS a MUCHOS (M:N)

### Modelo ENTIDAD-RELACIÓN:

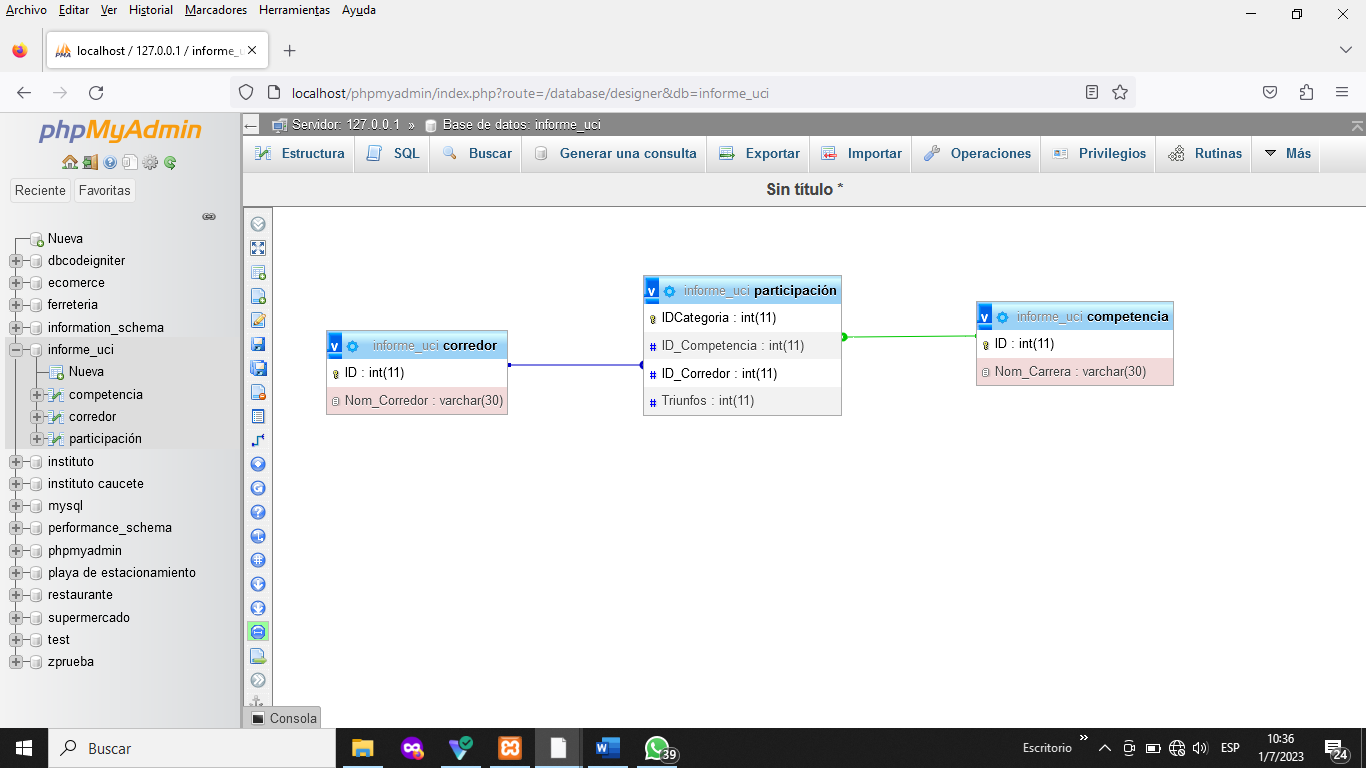
* La entidad se convierte en una tabla.
* Cada atributo se convierte a un campo de la tabla.
* Se crea una tabla adicional que relaciona las entidades primarias



### Modelo RELACIONAL:



### Modelo FÍSICO:



# Modelo de Ejemplo.

## Documento de Especificación de Requisitos (SRS) –

### Software de Visualización de Triunfos de Corredores de Ciclismo

#### **1. Introducción**

El presente documento describe los requisitos para el desarrollo de un software de visualización de triunfos de corredores de ciclismo para la UCI (Unión Ciclista Internacional). Este software permitirá identificar a los corredores con más victorias en las 9 competencias más relevantes del calendario ciclista internacional.

#### **2. Objetivos**

El objetivo principal del software es proporcionar a la UCI una herramienta que les permita visualizar de manera clara y concisa quiénes fueron los corredores con más triunfos en las competiciones ciclistas más importantes a nivel internacional. Los objetivos específicos incluyen:

* Recopilar datos de las 9 competiciones más importantes del calendario ciclista internacional.
* Analizar y procesar los datos para determinar los corredores con más victorias en cada competición.
* Presentar los resultados de manera visualmente atractiva y fácil de entender para los usuarios.

#### **3. Requisitos Funcionales**

* **3.1 Recopilación de Datos**

El software debe ser capaz de recopilar datos de las 9 competiciones ciclistas más importantes, incluyendo el Tour de Francia, Giro de Italia, Vuelta a España, y otras competiciones relevantes.

Los datos deben incluir información sobre los corredores participantes, sus equipos, las etapas de la competición y los resultados de cada etapa.

El sistema debe ser capaz de acceder a fuentes de datos confiables y actualizadas regularmente.

* **3.2 Procesamiento de Datos**

El software debe procesar los datos recopilados para determinar los corredores con más triunfos en cada competición.

Debe tener la capacidad de calcular el número de victorias de cada corredor en cada competición y clasificarlos en orden descendente según el número de victorias.

Se deben considerar criterios adicionales, como desempates en caso de igualdad de victorias.

* **3.3 Visualización de Datos**

El sistema debe presentar los resultados de manera visualmente atractiva, utilizando gráficos, tablas u otros medios adecuados.

Debe permitir a los usuarios filtrar los resultados por competición, año, corredor, equipo, entre otros criterios.

Debe proporcionar opciones de personalización para adaptarse a las preferencias de los usuarios.

#### **4. Requisitos No Funcionales**

* **4.1 Usabilidad**

La interfaz de usuario debe ser intuitiva y fácil de usar, incluso para usuarios con poca experiencia en el uso de software.

El software debe tener tiempos de respuesta rápidos para garantizar una experiencia fluida del usuario.

* **4.2 Fiabilidad**

El sistema debe ser confiable y preciso en el procesamiento y presentación de los datos.

Debe tener mecanismos de respaldo para proteger los datos en caso de fallos del sistema.

* **4.3 Seguridad**

El acceso al software debe estar restringido a usuarios autorizados dentro de la UCI.

Debe implementarse medidas de seguridad para proteger los datos recopilados y procesados.

#### **5. Conclusiones**

El desarrollo de este software proporcionará a la UCI una herramienta invaluable para analizar y visualizar los triunfos de los corredores en las competiciones ciclistas más importantes del mundo. Cumplir con los requisitos especificados garantizará un sistema robusto y confiable que satisfaga las necesidades de la organización.

# Bibliografía

* Wikipedia, Gestión de Bases de Datos.
* <https://gestionbasesdatos.readthedocs.io/es/latest/Tema2/Teoria.html>
* <https://diagramasuml.com/%e2%96%b7-diagrama-entidad-relacion-e-r-teoria-y-ejemplos/>