

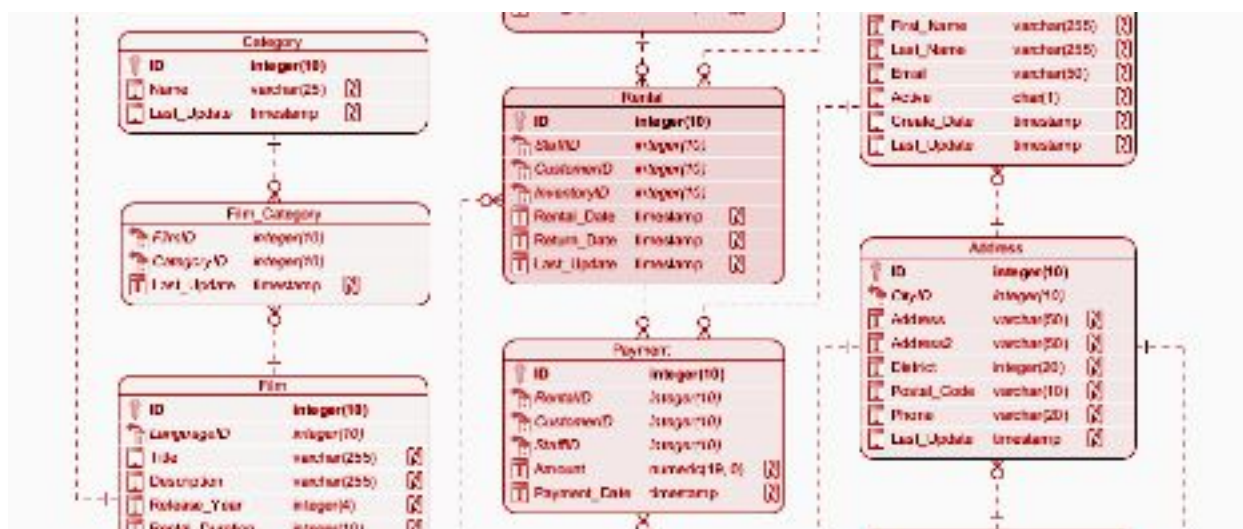
## Desarrollo Web Full Stack Node

Trabajo integrador - M06C01

# Ejercitación - Bases de datos 1

## Introducción

Al igual que hicimos con el armado de wireframes y bocetos antes de comenzar a trabajar con HTML y CSS, es siempre recomendable que pensemos en las necesidades de datos que va a tener nuestro sistema y que diseñemos la base de datos primero con lápiz y papel (o su equivalente digital 🧐💡) antes de ponernos a crear tablas.



## Consignas

A continuación les planteamos diferentes escenarios que necesitan de una base de datos para poder almacenar la información del sistema. Su rol será el de diseñar una base de datos junto con sus tablas, campos y relaciones para satisfacer esas necesidades. Es importante que interpreten lo que leen y no tomen todo línea por línea 🙄△.

Para esta tarea les recomendamos utilizar la herramienta de [diagrams.net](https://diagrams.net). Si lo hacen, deben seleccionar: “Save to device” → “New diagram” → “Entity Relationship Diagram”.

Luego cuando terminen les recomendamos exportar el resultado como html (“File” → “Export as” → “HTML”), de esa manera podrán volver a importarlo a la herramienta si quieren seguir trabajándolo.

Algunas sugerencias de cómo encarar el ejercicio:

- Empiecen por determinar las entidades principales que se desprendan de los requisitos, pueden ser usuarios, productos, etc.
- Cuando ya las tengan piensen en qué datos necesitarán almacenar de cada una de ellas.
- Sigán por determinar qué información necesitan separar en tablas secundarias, normalmente son cosas como categorías, temáticas, colores, etc.
- Finalmente establezcan las relaciones entre las tablas que crearon

Terminado todo este ejercicio, imaginen la base poblada de datos y funcionando para ver si detectan algún fallo en el diseño que armaron.

Ya lo dijimos antes... **las etapas de planificación son de las más importantes de cualquier proyecto**. De hecho, muchos proyectos fallan al salir a producción por no haber hecho bien esta tarea 🧐👉✨.

## Escenario 1 – Playground

Vamos con un viejo conocido. Para entrar en calor, les pedimos que modelen la base de datos que necesitaría una versión simplificada de Playground.



### Desafío 1 – Los usuarios y los cursos

Queremos tener usuarios, los usuarios tendrán nombre, apellido, email, contraseña y categoría. Los usuarios podrán tener categoría de estudiantes, docentes, editores o administradores.

Lo siguiente que queremos es poder almacenar los cursos, que tendrán un título, una descripción, una imagen, una fecha de inicio, una fecha de finalización y un cupo máximo.

Los cursos tendrán unidades (para organizar el contenido) que tendrán un título, una descripción y una fecha de inicio.

Los usuarios (de cualquier tipo) podrán estar asociados a cursos.

### Desafío 2 – El contenido del curso

Las unidades contendrán clases que también tendrán un título, una descripción, una fecha de inicio, y una marca de visibilidad (si el bloque está visible o no).

Las clases contendrán bloques. Los bloques tendrán un título y una marca de visibilidad. Los bloques podrán ser de diferente tipo: texto, video, presentación, PDF o archivo.

Los bloques también tendrán que poder guardar el contenido en caso de que el tipo sea texto y la url en caso de que sea video, presentación, PDF o archivo.

## Escenario 2 – Bazar Digital

Queremos modelar el sistema de un Bazar. Acá se pone un poco más difícil porque hay parte de los requerimientos que no son para la base de datos y los podrán resolver en el sistema luego. Les queda a ustedes determinar cuáles se resuelven en la base 🤔💡✨.



### Desafío 1 – Los artículos y los empleados

En charlas con los dueños entendemos que el bazar va a vender vajilla, juguetes y artículos de jardín, y van a tener empleados comunes, un coordinador, empleados de mostrador y repositores.

Todos los artículos tendrán nombre, precio, descripción, stock y aclararán si son para uso profesional o no.

Todos los empleados tendrán nombre, apellido, dni, sueldo y una comisión general.

### Desafío 2 – El stock y las ventas

Los empleados de mostrador son los únicos que podrán cobrar y tendrán una caja donde guardar lo que cobran, y los repositores podrán ir al depósito de mercaderías. Para eso esperamos poder saber cuantos productos están exhibidos y / o en el depósito.

Los empleados pueden vender un artículo generando una comisión adicional del 10% del valor del artículo. Para las ventas esperamos poder almacenar fecha, artículos, medio de pago y total de la compra.

## Cierre

Como toda habilidad, se requiere práctica para mejorar. No se preocupen si al principio es difícil armar una estructura coherente de base de datos. Es cuestión de ir probando y con el tiempo van a encontrar las estructuras que mejor funcionan para cada caso que tengan que resolver.

No olviden esta práctica que pronto les tocará ponerse a diseñar la base de datos de su proyecto integrador 🤓🚀??.