

## **Desarrollo de proyecto.**

### **Etapas 1 – Inicio del proyecto.**

- Crear un proyecto nuevo llamado Negocio, usando PHP.
- Crear el repositorio git y commitear el inicio del proyecto.
- Crear un repositorio github o equivalente.

### **Etapas 2 – Armado de Base de datos.**

- Montar en un servidor de base datos la siguiente BD.
- El código DDL de la base debe ser parte dentro del proyecto.
- Armar un DER de la base y documentar el DER en el proyecto.
- La base de datos puede ser cualquier motor conocido del mercado, se recomienda MySQL, MariaDB o SQLite.
- Generar registros DML de prueba para la base de datos, se puede usar una IA.
- Generar consultas de prueba para base, se puede usar una IA.

```
drop database if exists negocio;  
create database negocio;  
use negocio;
```

```
create table if not exists clientes (  
codigo integer auto_increment,  
nombre varchar(20) not null,  
apellido varchar(20) not null,  
cuit char(13),  
direccion varchar(50),  
comentarios varchar(140),  
primary key (codigo)  
);
```

```
create table facturas(  
letra char(1),  
numero integer,  
fecha date,  
monto double,  
primary key (letra,numero)  
);
```

```
create table articulos(  
codigo integer auto_increment,  
descripcion varchar(50),  
precio double,  
stock integer,  
primary key (codigo)  
);
```

```
alter table facturas add codigocliente integer not null;
```

```
alter table facturas  
add constraint FK_facturas_codcli  
foreign key (codigocliente)  
references clientes(codigo)  
on delete cascade;
```

```
create table detalles(  
letra char(1) not null,  
numero int not null,  
codigo int not null,
```

```
cantidad int unsigned not null,  
primary key(letra,numero,codigo)  
);  
  
alter table detalles  
add constraint FK_facturas_letra_numero  
foreign key (letra,numero)  
references facturas(letra,numero)  
on delete cascade;
```

### **Etapas 3 – Armado de la aplicación.**

- Diseñar estructuras de paquetes del proyecto, siguiendo las buenas prácticas.
- Armar una clase Connector que permita manejar las conexiones a la BD.
- Armar un test de objetos MOCKs para el conector.
- Armar el patrón DAO que permita el acceso a la base dada (Interface e implementación de repositorio PDO).
- Armar un test de objetos MOCKs de los repositorios.
- Crear una GUI (HTML – Bootstrap - JS) que brinde un ABM de cada entidad. (debe tener Validaciones, ser responsiva y ser fácil de usar)

### **Etapas 4 – Mejora de aplicación.**

- Debatir en grupo que mejoras realizaría a la aplicación, y armar una lista de posibles mejoras, Ejemplo, mejoras en la base, mejoras en el patrón DAO, seguridad o mejoras en la GUI.
- Armar un plan de trabajo para realizar dichas mejoras y ejecutarlas.