



**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



# **Instituto Tecnológico de León**

## **Integrantes:**

Luis Adrian Casillas Ornelas

Leonardo Missael Hernandez Calvillo

Luis Brandon Torres Rocha

## **Profesor:**

Luz del Carmen Ruiz Gaytán

## **Taller de Base de Datos**

## **Proyecto Final: Conectividad a la BD**

**7/07/2021**

## Contenido

<b>Enunciado semántico:</b> .....	3
<b>Modelo entidad relación:</b> .....	3
<b>Modelo relacional:</b> .....	4
<b>Script:</b> .....	4
<b>Conexión a la BD con un lenguaje de programación:</b> .....	15
<b>Pagina de inicio:</b> .....	15
<b>Consulta:</b> .....	16
<b>Enunciado:</b> .....	16
<b>Modelo entidad relación</b> .....	17
<b>Modelo relacional:</b> .....	17
<b>Código de la conexión a la BD:</b> .....	18

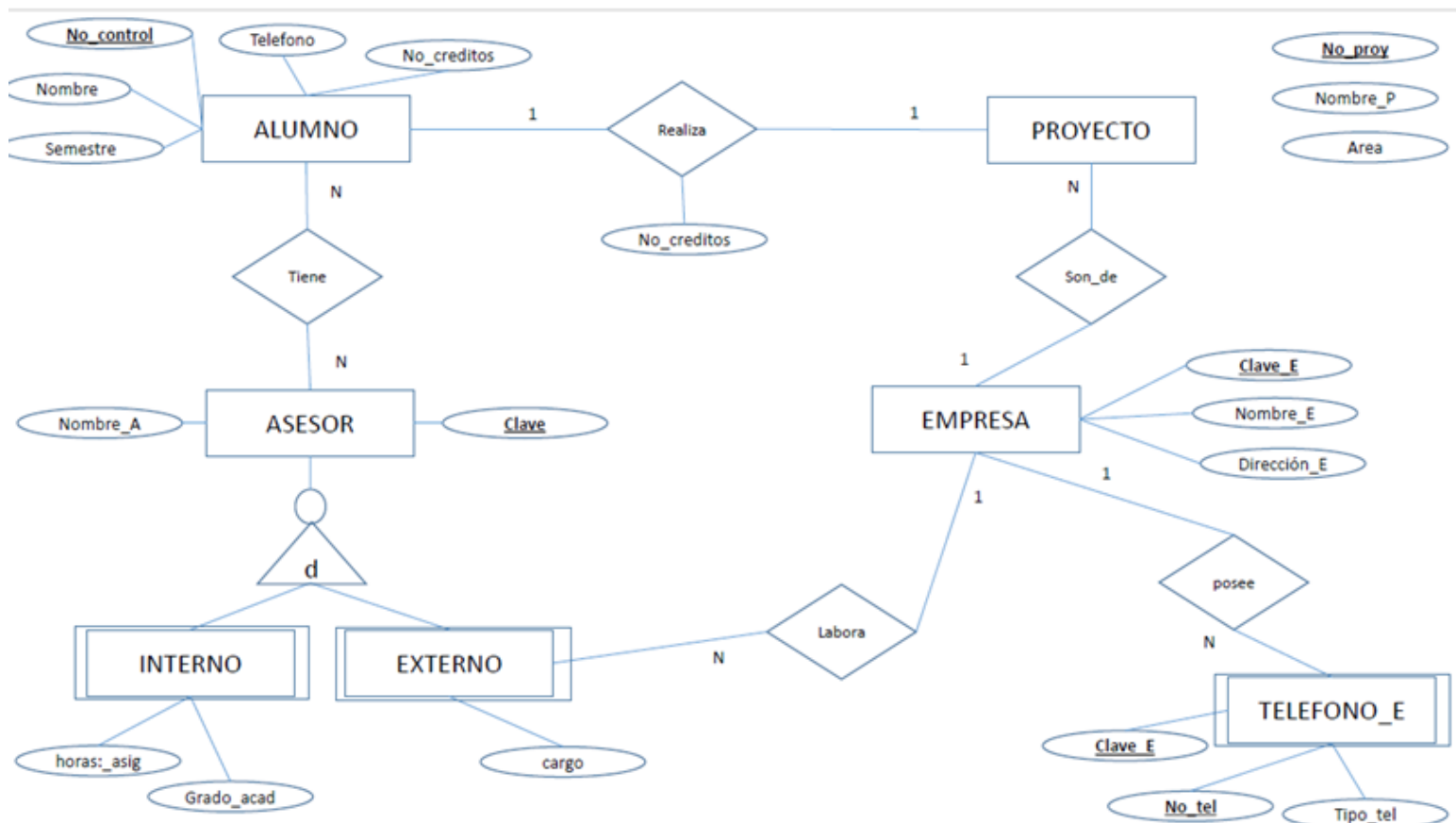


## Enunciado semántico:

Se requiere llevar el control de los alumnos que llevan residencias profesionales. El departamento de Sistemas y Computación del Instituto Tecnológico de León. Se deberán contemplar los siguientes requisitos:

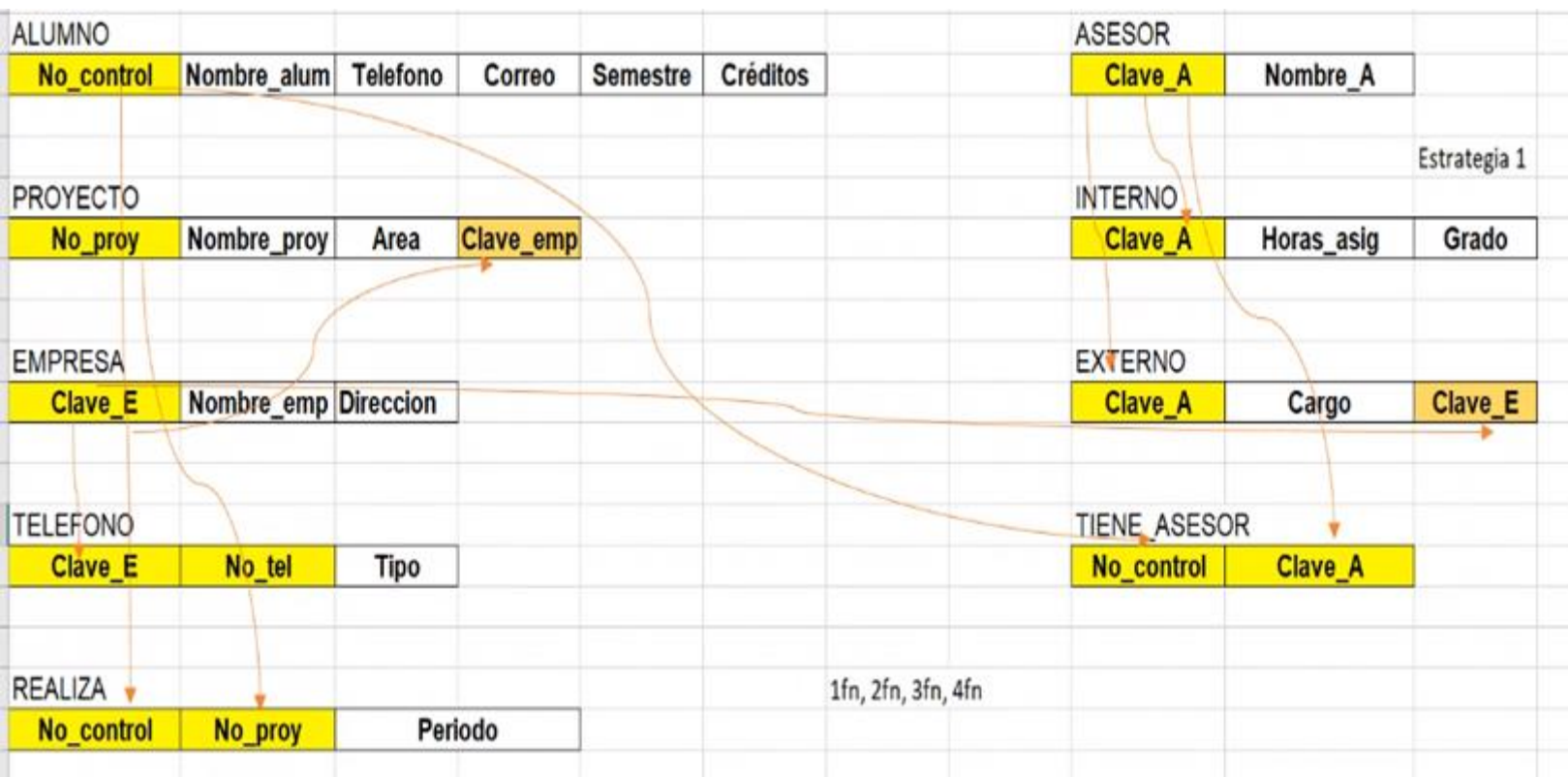
- ❑ La información que hay que registrar por alumno es: número de control, nombre completo, teléfono, semestre, cantidad de créditos aprobados y correo.
- ❑ Los residentes o alumnos tendrán que realizar un proyecto del cual hay que registrar: número de proyecto, nombre del proyecto, área a la que pertenece (Base de datos, redes, software, electrónica, etc.)
- ❑ Los proyectos se realizan en empresas de las cuales se registra: clave de empresa, nombre de la empresa, dirección, teléfonos, nombre del asesor externo, su clave y su cargo.
- ❑ Los alumnos residentes tendrán dos asesores: asesor externo (el de la empresa) y un asesor interno (docente de sistemas).
- ❑ De los docentes nos interesa registrar: clave de maestro, nombre, horas de asignación y grado académico.
- ❑ Se debe contemplar que un residente solo puede realizar un proyecto y que el proyecto solo es realizado por un residente.
- ❑ Los docentes podrán ser asesores de más de un residente y un residente solo puede tener un docente como asesor.
- ❑ También interesa registrar el periodo en que el residente realizó el proyecto en determinada empresa para evitar que se pueda repetir las residencias. Estos periodos pueden ser: Enero-Junio o Agosto-Diciembre de cada año. Solo podrán realizar residencias una sola vez en toda su carrera.

## Modelo entidad relación:





## Modelo relacional:



## Script:

```
use master
create database residencias;
use residencias;
```

### -----Creacion de la BD-----

```
--Se crea la tabla alumno
create table alumno(
No_control char(8) not null,
Nombre_alum varchar(30),
Telefono char(10),
Correo varchar(50),
Semestre char(2) not null,
Creditos int not null
)
--Se establece la llave primaria
alter table alumno add constraint pk_alumno primary key(No_control)

--Se llena la tabla alumno
select * from alumno
```



**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



```

insert into alumno values
('17240666','Adrian','4778964523','a@gmail.com','10',190)
insert into alumno values
('17425464','Andrea','4778568887','b@gmail.com','9',200)
insert into alumno values
('19795486','Eduardo','4774865403','c@gmail.com','11',200)
insert into alumno values
('16856489','Paola','4778963245','d@gmail.com','10',195)
insert into alumno values
('17986536','Miguel','4798546326','e@gmail.com','12',205)

--Se crea la tabla empresa
create table empresa(
Clave_E char(5) not null,
Nombre_emp varchar(30),
Direccion varchar(30))

--Se llena la tabla Empresa
select * from empresa
insert into empresa values
('E-896','INDESYTH','Av. Miguel Hidalgo #2')
insert into empresa values
('E-892','INDEX','Av. Republica #452')
insert into empresa values
('E-452','SEIQ S. DE R.L. DE C.V','CIRCUITO EL MARQUÉS SUR # 12')
insert into empresa values
('E-798','IHFES','ADMINISTRACIÓN DE FINANC.')
insert into empresa values
('E-895','AURIMÉDICA S DE SR DE CV','ARMANDO BIRLAIN 2001')

--Se establece su llave primaria
alter table empresa add constraint pk_empresa primary key(Clave_E)

--Se crea la tabla proyecto
create table proyecto(
No_proy char(5) not null,
Nombre_proy varchar(30),
area varchar(30),
clave_emp char(5))

--Se llena la tabla proyecto
select * from proyecto
insert into proyecto values
('Q-546','MINI DRONE DE USO RECREATIVO','Software','E-896')
insert into proyecto values
('Q-547','PROGRAMACION IOS PARA DRONES','Software','E-892')
insert into proyecto values
('Q-548','ELABORACION SITIO WEB CASE IH','Sitio Web','E-452')
insert into proyecto values
('Q-549','PRESUPUESTO AUTORIZADO','Software','E-798')
insert into proyecto values
('P-256','CONSULTA ON LINE','Software','E-895')

--Se establece la llave primaria
alter table proyecto add constraint pk_proyecto primary key(No_proy)

```



**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



```
--Se establece la llave foranea
alter table proyecto add constraint fk_proyecto
foreign key (clave_emp) references empresa (Clave_E)

--Se crea la tabla telefono
create table telefono(
Clave_E char(5) not null,
No_tel char(10) not null,
tipo varchar(20)
)

--Se llena la tabla telefono
select * from telefono
insert into telefono values ('E-896', '4778965423', 'Oficina')
insert into telefono values ('E-892', '4778964789', 'Oficina')
insert into telefono values ('E-452', '4785463259', 'Oficina')
insert into telefono values ('E-798', '4795969872', 'Oficina')
insert into telefono values ('E-895', '4775786423', 'Oficina')

--Se establece la llave primaria
alter table telefono add constraint pk_telefono primary key(No_tel)

--Se establece la llave foranea
alter table telefono add constraint fk_telefono
foreign key (Clave_E) references empresa (Clave_E)

--Se crea la tabla realiza
create table realiza(
No_control char(8) not null,
No_proy char(5) not null,
periodo varchar(20)
)

--Se establece la llave foranea
alter table realiza add constraint fk_alumno
foreign key (No_control) references alumno (No_control)

alter table realiza add constraint fk_realiza2
foreign key (No_proy) references proyecto (No_proy)

--Se llena la tabla telefono
select * from realiza
insert into realiza values ('17240666', 'Q-546', 'Enero-Junio')
insert into realiza values ('17425464', 'Q-547', 'Enero-Junio')
insert into realiza values ('19795486', 'Q-548', 'Agosto-Diciembre')
insert into realiza values ('16856489', 'Q-549', 'Agosto-Diciembre')
insert into realiza values ('17986536', 'P-256', 'Enero-Junio')

--Se crea la tabla asesor
create table asesor(
Clave_A char(5) not null,
Nombre_A char(40) not null,
)

--Se establece la llave primaria
alter table asesor add constraint pk_asesor primary key (Clave_A)
```





**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



```
--Se llena la tabla asesor
select * from asesor
insert into asesor values ('A-001', 'Laura Juarez Guerra')
insert into asesor values ('A-002', 'Gerardo Carpio')
insert into asesor values ('A-003', 'Ruth Saenz')
insert into asesor values ('A-004', 'Cirino Silva')
insert into asesor values ('A-005', 'Edna Militza')
insert into asesor values ('E-789', 'Moises Elias Perez')
insert into asesor values ('E-790', 'Pedro Gonzalez')
insert into asesor values ('E-791', 'Omar Palomares')
insert into asesor values ('E-792', 'Martin Mora')
insert into asesor values ('E-793', 'Jorge Cosileon')

--Se crea la tabla interno
create table interno(
Clave_A char(5) not null,
Horas_asig char(3) not null,
Grado varchar(20)
)

--Llenar la tabla interno
select * from interno
insert into interno values ('A-001', '100', 'Maestria')
insert into interno values ('A-002', '120', 'Doctorado')
insert into interno values ('A-003', '70', 'Maestria')
insert into interno values ('A-004', '100', 'Maestria')
insert into interno values ('A-005', '105', 'Maestria')

--Se estableve la llave primaria
alter table interno add constraint fk_interno
foreign key(Clave_A) references asesor(Clave_A)

--Se crea la tabla externo
create table externo(
Clave_A char(5) not null,
cargo varchar(30) not null,
Clave_E char(5)
)

--Llenar la tabla externo
select * from externo
insert into externo values ('E-789', 'Director', 'E-896')
insert into externo values ('E-790', 'Jefe de Sistemas', 'E-892')
insert into externo values ('E-791', 'Asesor Web', 'E-452')
insert into externo values ('E-792', 'Encargado Sistemas', 'E-798')
insert into externo values ('E-793', 'Jefe de Sistemas', 'E-895')

--Se crea llave foranea
alter table externo add constraint fk_externo
foreign key(Clave_A) references asesor(Clave_A)

alter table externo add constraint fk_externo2
foreign key(Clave_E) references empresa(Clave_E)

--Se crea la tabla tiene_asesor
```



**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



```
create table tiene_asesor(
No_control char(8) not null,
Clave_A char(5) not null
)

--Se llena tabla tiene_asesor
select * from tiene_asesor
insert into tiene_asesor values ('17240666', 'A-001')
insert into tiene_asesor values ('17425464', 'A-002')
insert into tiene_asesor values ('19795486', 'A-003')
insert into tiene_asesor values ('16856489', 'A-004')
insert into tiene_asesor values ('17986536', 'A-005')
insert into tiene_asesor values ('17240666', 'E-789')
insert into tiene_asesor values ('17425464', 'E-790')
insert into tiene_asesor values ('19795486', 'E-791')
insert into tiene_asesor values ('16856489', 'E-792')
insert into tiene_asesor values ('17986536', 'E-793')

--Se crea llave foranea
alter table tiene_asesor add constraint fk_tiene_asesor1
foreign key(No_control) references alumno(No_control)

alter table tiene_asesor add constraint fk_tiene_asesor2
foreign key(Clave_A) references asesor(Clave_A)
```

#### -----Creacion de Vistas-----

```
--Muestra el numero de control, el nombre del alumno y el nombre proyecto
CREATE VIEW vwAlumno_Proyecto
WITH ENCRYPTION
AS
SELECT RE.No_control, Nombre_alum, Nombre_proy
FROM (alumno AS AL JOIN realiza AS RE ON AL.No_control = RE.No_control)
JOIN proyecto AS PO ON RE.No_proy = PO.No_proy
--Consulta
select * from vwAlumno_Proyecto

--Muestra el numero de control, el nombre del proyecto, y quien asesora
el proyecto
CREATE VIEW vwAsesor_Proyecto
WITH ENCRYPTION
AS
SELECT RE.No_control, AE.Nombre_A AS 'Nombre Asesor', PO.Nombre_proy
FROM realiza AS RE
JOIN tiene_asesor AS TA ON TA.No_control = RE.No_control
JOIN asesor AS AE ON TA.Clave_A = AE.Clave_A
JOIN proyecto AS PO ON PO.No_proy = RE.No_proy
--Consulta
SELECT * FROM vwAsesor_Proyecto

--Muestra empresa, proyecto y asesor de la empresa
CREATE VIEW vwEmpresa_Proyecto
WITH ENCRYPTION
AS
SELECT Nombre_emp, Nombre_proy, Nombre_A
```





**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



```

FROM (empresa AS EM JOIN proyecto AS PO ON EM.Clave_E = PO.clave_emp)
JOIN externo AS EX ON EM.Clave_E = EX.Clave_E
JOIN asesor AS AE ON EX.Clave_A = AE.Clave_A
--Consulta
SELECT * FROM vwEmpresa_Proyecto

--Muestra el numero de control, nombre del proyecto y el periodo en el
que se realiza
CREATE VIEW vwProyecto_Periodo
WITH ENCRYPTION
AS
SELECT RE.No_control, PO.Nombre_proy, RE.periodo
FROM realiza AS RE JOIN proyecto AS PO ON RE.No_proy = PO.No_proy
--Consulta
SELECT * FROM vwProyecto_Periodo

--Muestra nombre del alumno, el numero de creditos y el semestre que
curso
CREATE VIEW vwInfo_Alumno
WITH ENCRYPTION
AS
SELECT Nombre_alum, Creditos, Semestre
FROM alumno
--Consulta
SELECT * FROM vwInfo_Alumno

--Muestra nombre de la empresa, direccion y numero de contacto
CREATE VIEW vwInfo_Empresa
WITH ENCRYPTION
AS
SELECT EM.Nombre_emp, EM.Direccion, TE.No_tel
FROM empresa AS EM JOIN telefono AS TE ON EM.Clave_E = TE.Clave_E
--Consulta
SELECT * FROM vwInfo_Empresa

--Muestra la informacion de contacto del alumno
CREATE VIEW vwContacto_Alumno
WITH ENCRYPTION
AS
SELECT Nombre_alum, Telefono, Correo
FROM alumno
--Consulta
SELECT * FROM vwContacto_Alumno

--Muestra los asesores internos y el grado academico que tienen
CREATE VIEW vwAsesorInt_Info
WITH ENCRYPTION
AS
SELECT AR.Nombre_A, IR.Grado
FROM interno AS IR JOIN asesor AS AR ON IR.Clave_A = AR.Clave_A
--Consulta
SELECT * FROM vwAsesorInt_Info

```



**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



## -----Creacion de Usuarios-----

```
--Se crean los usuarios de la base de datos
CREATE LOGIN escuela WITH PASSWORD = '1234';
CREATE LOGIN docente WITH PASSWORD = '1234';
CREATE LOGIN alumno WITH PASSWORD = '1234';

--Se crean los usuarios correspondientes a la BD
CREATE USER institucion FOR LOGIN escuela;
CREATE USER profesor FOR LOGIN docente;
CREATE USER estudiante FOR LOGIN alumno;

--Se crea un rol dueño para el usuario 'institucion'
CREATE ROLE escuela AUTHORIZATION db_owner;

--Se agrega el usuario al rol que le corresponde
EXEC sp_addrolemember 'escuela', 'institucion';

--Se crea un rol escritor para el usuario 'profesor'
CREATE ROLE docente AUTHORIZATION db_datawriter;
--Se crea un rol lector para el usuario 'profesor'
CREATE ROLE docenteL AUTHORIZATION db_datareader;

--Se agrega el usuario al rol que le corresponde
EXEC sp_addrolemember 'docente', 'profesor';
EXEC sp_addrolemember 'docenteL', 'profesor';

--Se establecen los permisos
DENY ALTER ON alumno to docente;
DENY ALTER ON asesor to docente;
DENY ALTER ON empresa to docente;
DENY ALTER ON externo to docente;
DENY ALTER ON interno to docente;
DENY ALTER ON proyecto to docente;
DENY ALTER ON realiza to docente;
DENY ALTER ON telefono to docente;
DENY ALTER ON tiene_asesor to docente;

--Se crea un rol dueño para el usuario 'profesor'
CREATE ROLE alumno AUTHORIZATION db_datareader;

--Se agrega el usuario al rol que le corresponde
EXEC sp_addrolemember 'alumno', 'estudiante';
```

## -----TEST DE PRIVILEGIOS-----

```
USE residencias;

--El Usuario3 (estudiante intenta cambiar datos de la BD)
SELECT * FROM alumno;
UPDATE alumno SET Nombre_alum = 'Luis' WHERE Nombre_alum = 'Adrian'

--El Usuario1 (institucion intenta cambiar datos de la BD)
SELECT * FROM alumno;
UPDATE alumno SET Nombre_alum = 'Luis' WHERE Nombre_alum = 'Adrian'
```



**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



```
--El Usuario2 (profesor intenta borrar o cambiar una base de datos)
DROP TABLE alumno;
ALTER TABLE alumno ADD Email2 varchar(255);
```

### -----Transacciones-----

```
/*Se maneja una transacción simple debido a que en la base de datos
no se encontró una situación donde se necesitara una transacción.*/
BEGIN TRANSACTION;
USE residencias
SELECT * FROM alumno
SELECT * FROM realiza
COMMIT;
```

### -----Procedimientos almacenados-----

```
--Procedimiento ingresar registros a la tabla alumno
CREATE PROCEDURE ingresarAlumno
@No_control char(8),
@Nombre_alum varchar(30),
@Telefono char(10),
@Correo varchar(50),
@Semestre char(2),
@Creditos int
AS
insert into alumno values (@No_control,@Nombre_alum,@Telefono,
                           @Correo,@Semestre, @Creditos)
GO
```

```
--Procedimiento ingresar registros a la tabla realiza
CREATE PROCEDURE ingresarRealiza
@No_control char(8),
@No_proy char(5),
@periodo varchar(20)
AS
insert into realiza values (@No_control,@No_proy,@periodo)
GO
```

```
--Procedimiento ingresar registros a la tabla proyecto
CREATE PROCEDURE ingresarProyecto
@No_proy char(5),
@Nombre_proy varchar(30),
@area varchar(30),
@clave_emp char(5)
AS
insert into proyecto values (@No_proy,@Nombre_proy,@area, @clave_emp)
GO
```

```
--Procedimiento ingresar registros a la tabla empresa
CREATE PROCEDURE ingresarEmpresa
@Clave_E char(5),
@Nombre_emp varchar(30),
@Direccion varchar(30)
AS
insert into empresa values (@Clave_E,@Nombre_emp,@Direccion)
```



**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



GO

--Procedimiento ingresar registros a la tabla telefono

CREATE PROCEDURE ingresarTelefono

@Clave\_E char(5),

@No\_tel char(10),

@tipo varchar(20)

AS

insert into telefono values (@Clave\_E,@No\_tel,@tipo)

GO

--Procedimiento ingresar registros a la tabla tiene\_asesor

CREATE PROCEDURE ingresarTieneA

@No\_control char(8),

@Clave\_A char(5)

AS

insert into tiene\_asesor values (@No\_control, @Clave\_A)

GO

--Procedimiento ingresar registros a la tabla asesor

CREATE PROCEDURE ingresarAsesor

@Clave\_A char(5),

@Nombre\_A char(40)

AS

insert into asesor values (@Clave\_A,@Nombre\_A)

GO

--Procedimiento ingresar registros a la tabla externo

CREATE PROCEDURE ingresarExterno

@Clave\_A char(5),

@cargo varchar(30),

@Clave\_E char(5)

AS

insert into externo values (@Clave\_A,@cargo,@Clave\_E)

GO

--Procedimiento ingresar registros a la tabla interno

CREATE PROCEDURE ingresarInterno

@Clave\_A char(5),

@Horas\_asig char(3),

@Grado varchar(20)

AS

insert into interno values (@Clave\_A,@Horas\_asig,@Grado)

GO

--PROCEDIMIENTOS PARA ELIMINAR--

--Procedimiento eliminar registros a la tabla alumno

CREATE PROCEDURE EliminarRegistroAlumno

@No\_control char(8)

AS

DELETE FROM alumno

WHERE No\_control = @No\_control

GO

--Procedimiento eliminar registros a la tabla empresa

CREATE PROCEDURE EliminarRegistroEmpresa



**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



```
@Clave_E char(5)
AS
DELETE FROM empresa
        WHERE Clave_E = @Clave_E
GO

--Procedimiento eliminar registros a la tabla proyecto
CREATE PROCEDURE EliminarRegistroProyecto
@No_proy char(5)
AS
DELETE FROM proyecto
        WHERE No_proy = @No_proy
GO

--Procedimiento eliminar registros a la tabla telefono
CREATE PROCEDURE EliminarRegistroTelefono
@No_tel char(10)
AS
DELETE FROM telefono
        WHERE No_tel = @No_tel
GO

--Procedimiento eliminar registros a la tabla realiza
CREATE PROCEDURE EliminarRegistroRealiza
@No_control char(8),
@No_proy char(5)
AS
DELETE FROM realiza
        WHERE No_control = @No_control AND No_proy = @No_proy
GO

--Procedimiento eliminar registros a la tabla asesor
CREATE PROCEDURE EliminarRegistroAsesor
@Clave_A char(5)
AS
DELETE FROM asesor
        WHERE Clave_A = @Clave_A
GO

--Procedimiento eliminar registros a la tabla asesor interno
CREATE PROCEDURE EliminarRegistroAInterno
@Clave_A char(5)
AS
DELETE FROM interno
        WHERE Clave_A = @Clave_A
GO

--Procedimiento eliminar registros a la tabla asesor externo
CREATE PROCEDURE EliminarRegistroAExterno
@Clave_A char(5)
AS
DELETE FROM externo
        WHERE Clave_A = @Clave_A
GO

--Procedimiento eliminar registros a la tabla tiene Asesor
```



**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



```
CREATE PROCEDURE EliminarRegistroTAsesor
@No_control char(8),
@Clave_A char(5)
AS
DELETE FROM tiene_asesor
        WHERE No_control = @No_control AND Clave_A = @Clave_A
GO
```

-----Creacion de trigger-----

```
create trigger CheckAlumnoInsert on alumno for insert
as
    if (select Creditos from inserted) < 0
    begin
        print 'Los créditos del alumno no pueden ser negativos.'
        rollback
    end
go
```





**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO

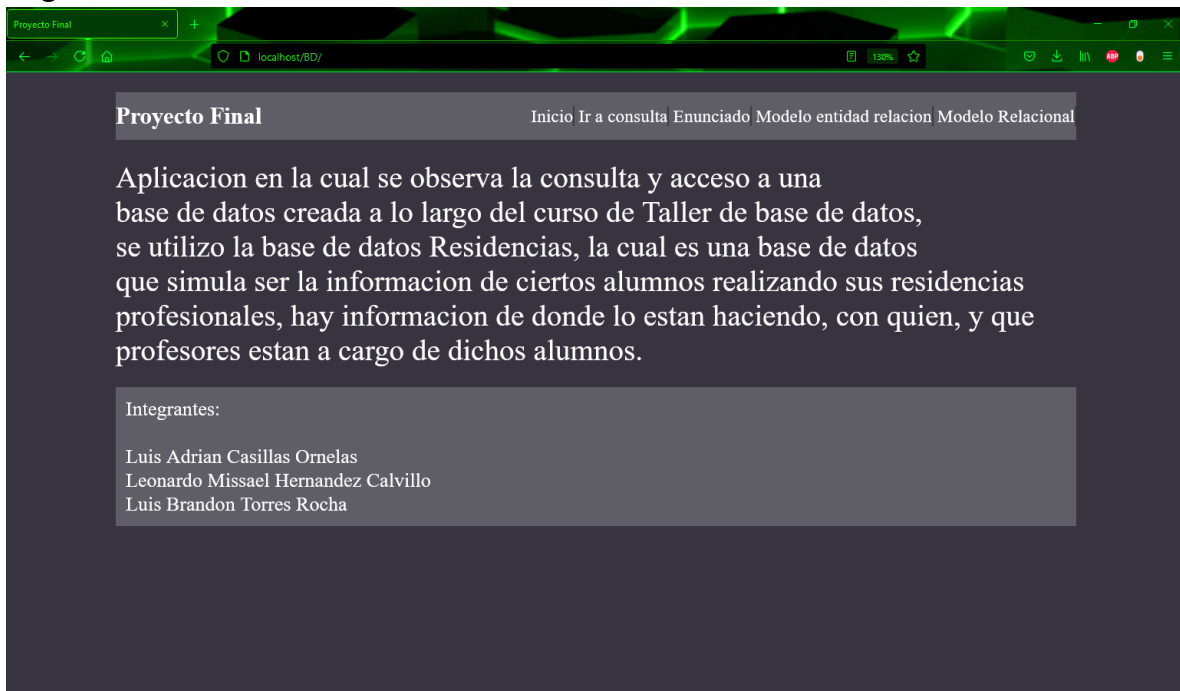


## Conexión a la BD con un lenguaje de programación:

Se realizó la conexión a la base de datos en PHP en el cual se tiene una pagina de inicio donde se explica de que trata el proyecto, así como contar con pestañas para ver una consulta previamente establecida en el lenguaje de programación, otra pestaña para ver el enunciado. El modelo entidad relación y una última para ver el modelo relacional.

En la pestaña de la consulta se puede observar a qué base de datos está conectado, con cual usuario se ingresó (establecido en el lenguaje), cuál fue la consulta que se realizó y la tabla resultante.

### Pagina de inicio:





**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



**TECNOLÓGICO  
NACIONAL DE MÉXICO**



## Consulta:

Conexion a una base de datos

localhost/BD/resultado.php

110%

Proyecto Final

Inicio Ir a consulta Enunciado Modelo entidad relacion Modelo Relacional

**Conexion establecida a la base de datos: Residencias**  
Conectado al usuario: docente  
La consulta realizada fue:  
**SELECT \* FROM alumno**

No_control	Nombre_alum	Telefono	Correo	Semestre	Creditos
16856489	Paola	4778963245	d@gmail.com	10	195
17240666	Adrian	4778964523	a@gmail.com	10	190
17425464	Andrea	4778568887	b@gmail.com	9	200
17986536	Miguel	4798546326	e@gmail.com	12	205
19795486	Eduardo	4774865403	c@gmail.com	11	200

Integrantes:  
Luis Adrian Casillas Ornelas  
Leonardo Missael Hernandez Calvillo  
Luis Brandon Torres Rocha

## Enunciado:

Proyecto Final

localhost/BD/enunciado.php

120%

Proyecto Final

Inicio Ir a consulta Enunciado Modelo entidad relacion Modelo Relacional

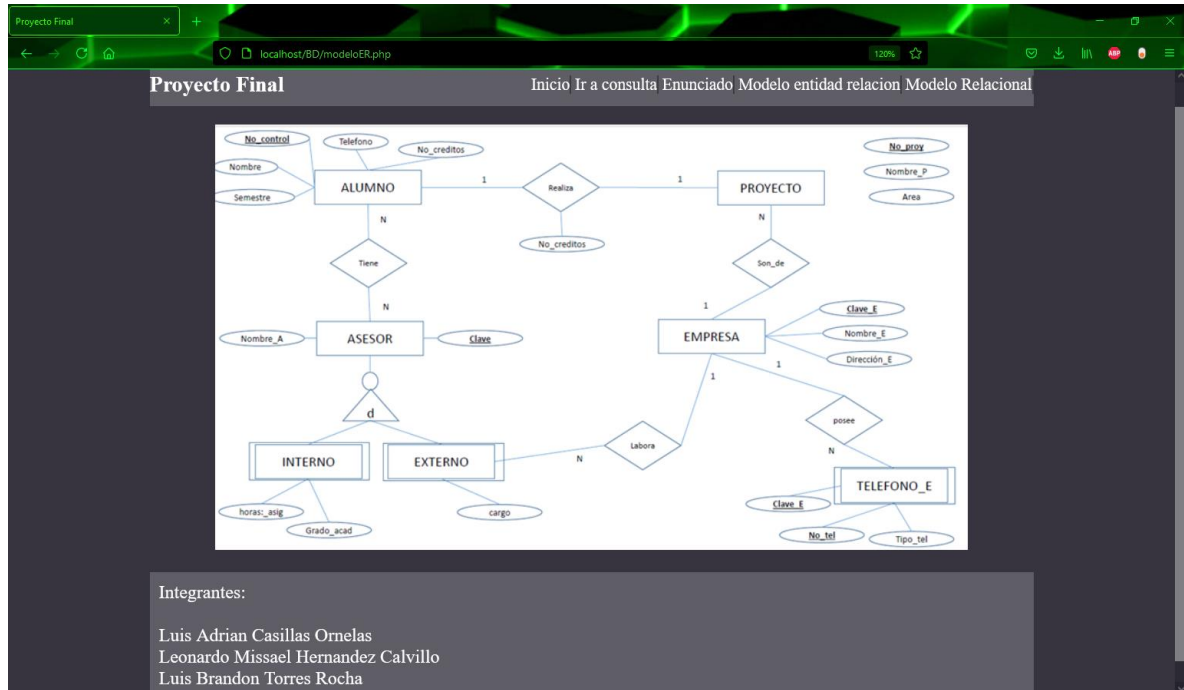
Se requiere llevar el control de los alumnos que llevan residencias profesionales. El departamento de Sistemas y Computación del Instituto Tecnológico de León. Se deberán contemplar los siguientes requisitos:

- La información que hay que registrar por alumno es: número de control, nombre completo, teléfono, semestre, cantidad de créditos aprobados y correo.
- Los residentes o alumnos tendrán que realizar un proyecto del cual hay que registrar: número de proyecto, nombre del proyecto, área a la que pertenece (Base de datos, redes, software, electrónica, etc.)
- Los proyectos se realizan en empresas de las cuales se registra: clave de empresa, nombre de la empresa, dirección, teléfonos, nombre del asesor externo, su clave y su cargo.
- Los alumnos residentes tendrán dos asesores: asesor externo (el de la empresa) y un asesor interno (docente de sistemas).
- De los docentes nos interesa registrar: clave de maestro, nombre, horas de asignación y grado académico.
- Se debe contemplar que un residente solo puede realizar un proyecto y que el proyecto solo es realizado por un residente.
- Los docentes podrán ser asesores de más de un residente y un residente solo puede tener un docente como asesor.
- También interesa registrar el periodo en que el residente realizó el proyecto en determinada empresa para evitar que se pueda repetir las residencias. Estos periodos pueden ser: Enero-Junio o Agosto-Diciembre de cada año. Solo podrán realizar residencias una sola vez en toda su carrera.

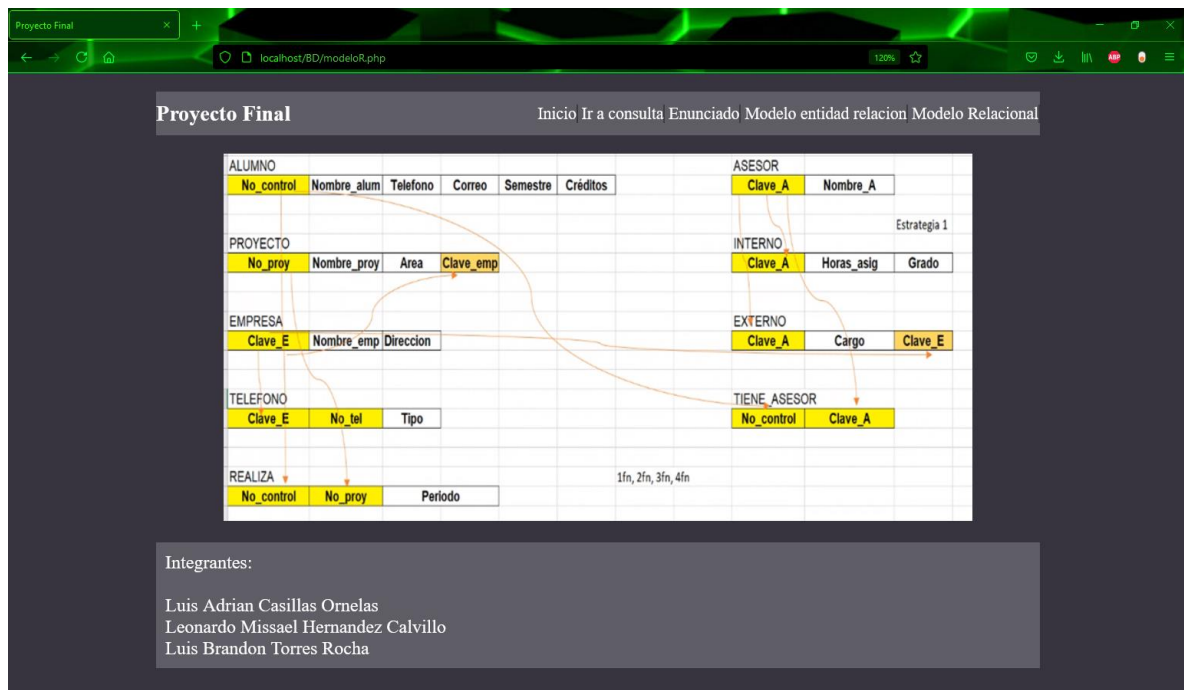
Integrantes:  
Luis Adrian Casillas Ornelas  
Leonardo Missael Hernandez Calvillo  
Luis Brandon Torres Rocha



## Modelo entidad relación



## Modelo relacional:





**MÉXICO**  
PRESIDENCIA DE LA REPÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



Código de la conexión a la BD:

```
<?php
$nombreServer = 'DESKTOP-L4BS0N0\SQLEXPRESS';
$infoDeConexion = array("Database" => "Residencias", "UID"=>"docente","PWD"=>"1234");
$conexion = sqlsrv_connect($nombreServer, $infoDeConexion);
$sql = "";

function validacion($conexion,$infoDeConexion){
    if ($conexion) return true;
    else {
        die('No se pudo conectar a la base de datos, revise las credenciales');
        return false;
    }
}

function consulta($conexion){
    global $sql;
    $sql= "SELECT * FROM alumno";
    $stmt = sqlsrv_query($conexion, $sql);
    return $stmt;
}
?>
```