

Programación:

CREATE DATABASE PRUEBA

```
CREATE TABLE IF NOT EXISTS PRU_PERSONAS (  
ID_PER INT AUTO_INCREMENT NOT NULL PRIMARY KEY,  
    PER_P_NOM VARCHAR(15) NOT NULL,  
    PER_S_NOM VARCHAR(15),  
    PER_T_NOM VARCHAR(15),  
    PER_P_APE VARCHAR(15) NOT NULL,  
    PER_S_APE VARCHAR(15),  
    PER_C_APE VARCHAR(15),  
    PER_DIR VARCHAR(80) NOT NULL,  
    PER_USU VARCHAR(10) NOT NULL,  
    PER_CON VARCHAR(10) NOT NULL,  
    PER_TEL INT(8),  
    PER_FECH DATE  
);
```

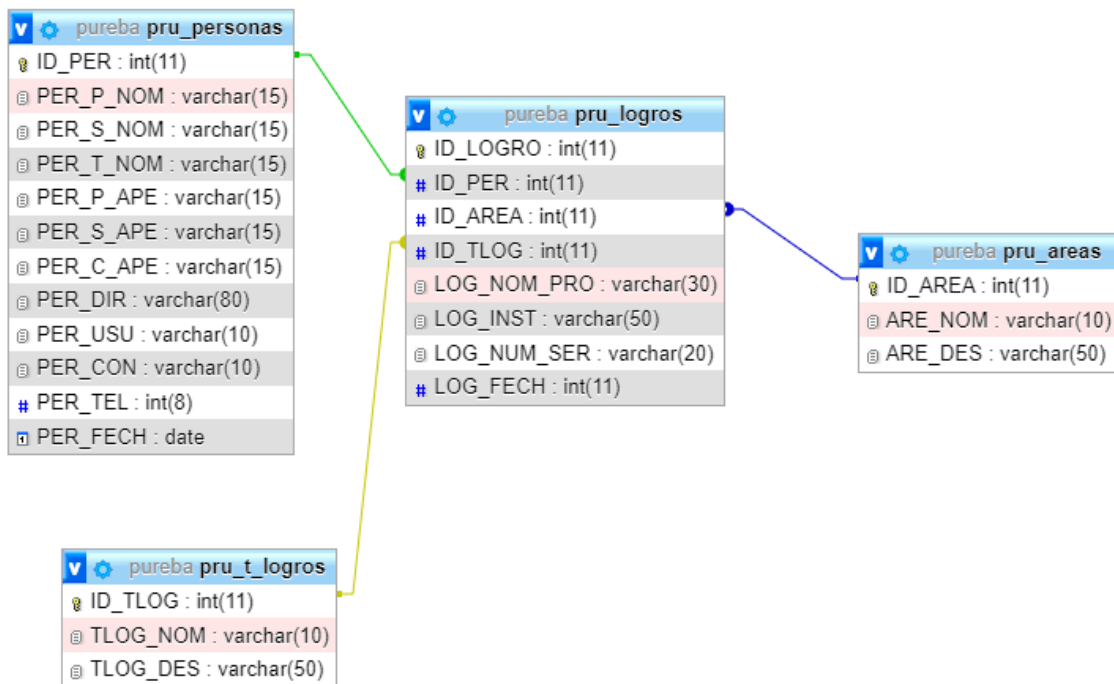
```
CREATE TABLE IF NOT EXISTS PRU_AREAS (  
ID_AREA INT NOT NULL AUTO_INCREMENT PRIMARY KEY,  
    ARE_NOM VARCHAR(10) NOT NULL,  
    ARE_DES VARCHAR(50)  
);
```

```
CREATE TABLE IF NOT EXISTS PRU_T_LOGROS (  
ID_TLOG INT NOT NULL AUTO_INCREMENT PRIMARY KEY,  
    TLOG_NOM VARCHAR(10) NOT NULL,  
    TLOG_DES VARCHAR(50)  
);
```

```

CREATE TABLE IF NOT EXISTS PRU_LOGROS (
ID_LOGRO INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
ID_PER INT NOT NULL,
ID_AREA INT NOT NULL,
ID_TLOG INT NOT NULL,
LOG_NOM_PRO VARCHAR(30),
LOG_INST VARCHAR(50),
LOG_NUM_SER VARCHAR(20),
LOG_FECH INT NOT NULL,
FOREIGN KEY (ID_PER) REFERENCES PRU_PERSONAS (ID_PER),
FOREIGN KEY (ID_AREA) REFERENCES PRU_AREAS (ID_AREA),
FOREIGN KEY (ID_TLOG) REFERENCES PRU_T_LOGROS (ID_TLOG)
);

```



INSER DE PRUEBA

USUARIO

```
INSERT INTO `pru_personas` (`ID_PER`, `PER_P_NOM`, `PER_S_NOM`, `PER_T_NOM`,  
`PER_P_APE`, `PER_S_APE`, `PER_C_APE`, `PER_DIR`, `PER_USU`, `PER_CON`, `PER_TEL`,  
`PER_FECH`) VALUES (NULL, 'MARIA', 'ALEJANDRA', '', 'MEJIA', 'MONTTOYA', NULL, '4TA AV 2-50  
RESIDENCIALES VALLE DE LA MARIPOSA', 'A.MEJIA', '12345', '44412584', '2000-03-17');
```

```
INSERT INTO `pru_personas` (`ID_PER`, `PER_P_NOM`, `PER_S_NOM`, `PER_T_NOM`,  
`PER_P_APE`, `PER_S_APE`, `PER_C_APE`, `PER_DIR`, `PER_USU`, `PER_CON`, `PER_TEL`,  
`PER_FECH`) VALUES (NULL, 'JUAN', 'ARMANDO', NULL, 'GONZALEZ', 'RAMIREZ', NULL, '6TA AV 8-9  
ZONA 1.', 'J.GONZALEZ', '12345', '66334497', '1993-04-09');
```

AREA

```
INSERT INTO `pru_areas` (`ID_AREA`, `ARE_NOM`, `ARE_DES`) VALUES (NULL, 'IT', 'TECNOLOGIA');
```

T_LOGRO

```
INSERT INTO `pru_t_logros` (`ID_TLOG`, `TLOG_NOM`, `TLOG_DES`) VALUES (NULL,  
'LICENCIATURA', NULL)
```

```
INSERT INTO `pru_t_logros` (`ID_TLOG`, `TLOG_NOM`, `TLOG_DES`) VALUES (NULL, 'DOCTORADO',  
NULL);
```

LOGRO

```
INSERT INTO `pru_logros` (`ID_LOGRO`, `ID_PER`, `ID_AREA`, `ID_TLOG`, `LOG_NOM_PRO`,  
`LOG_INST`, `LOG_NUM_SER`, `LOG_FECH`) VALUES (NULL, '3', '1', '1', 'INGENIERO EN SISTEMAS',  
'UNIVERSIDAD SAN CARLOS', NULL, '2007-05-07');
```

```
INSERT INTO `pru_logros` (`ID_LOGRO`, `ID_PER`, `ID_AREA`, `ID_TLOG`, `LOG_NOM_PRO`,  
`LOG_INST`, `LOG_NUM_SER`, `LOG_FECH`) VALUES (NULL, '1', '2', '2', 'BASES DE DATOS',  
'UNIVERSIDAD SAN CARLOS', NULL, '2009-05-07');
```

```
INSERT INTO `pru_logros` (`ID_LOGRO`, `ID_PER`, `ID_AREA`, `ID_TLOG`, `LOG_NOM_PRO`,  
`LOG_INST`, `LOG_NUM_SER`, `LOG_FECH`) VALUES (NULL, '3', '2', '2', 'terapeuta', 'UNIVERSIDAD  
SAN CARLOS', NULL, '2010-05-07');
```

CONSULTA

PRIMERA TABLA:

```
SELECT per.PER_P_NOM, per.PER_s_NOM, per.PER_t_NOM, per.PER_P_APE, per.PER_S_APE,  
per.PER_C_APE, LO.ID_TLOG , T.TLOG_NOM FROM PRU_PERSONAS per INNER JOIN PRU_LOGROS  
LO ON LO.ID_PER = per.ID_PER INNER JOIN PRU_T_LOGROS T ON LO.ID_TLOG = T.ID_TLOG;
```

SEGUNDA TABLA:

```
SELECT t2.ID_AREA, t1.ARE_NOM, COUNT(*) FROM pru_logros as t2, pru_areas AS t1  
WHERE t2.ID_AREA = t1.ID_AREA  
GROUP BY t2.ID_AREA, t1.ARE_NOM;
```

Para abrir proyecto:

Terminal:

Npm init -- yes

Cree el archivo package.json

Instale los módulos:

npm install

express (códigos ya probados)

mysql (poder conectarse a sql)

express-myconnection (conexión sql rapida)

morga (ver peticiones a servidor)

ejs (motor de plantilla)

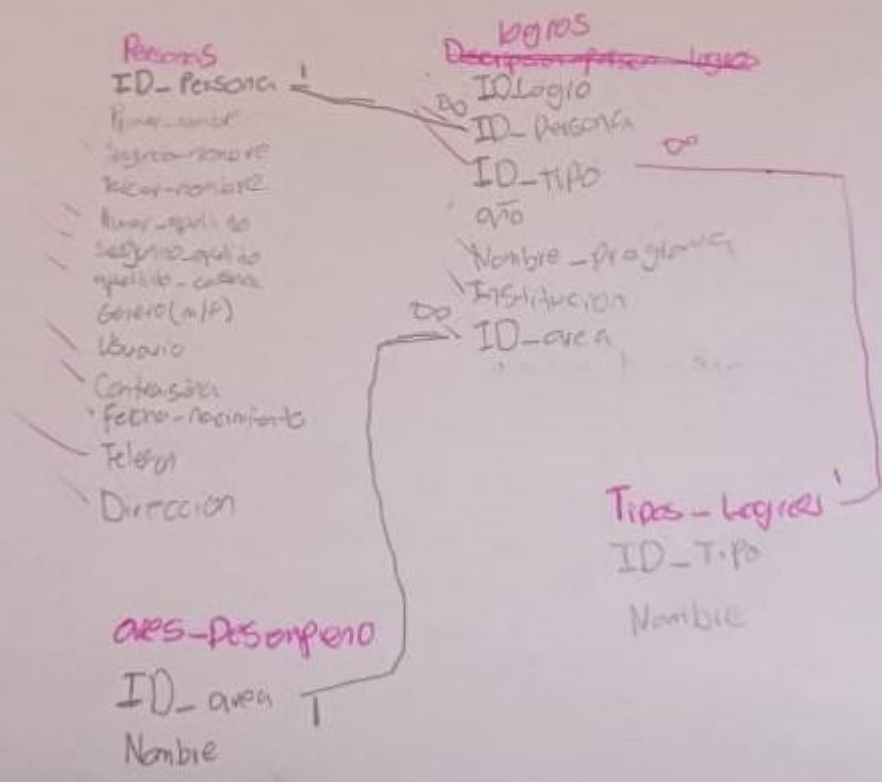
nodemon (reinicia el servidor automáticamente cada ves que hay un cambio)

node app (desde aquí inicia pruebas)

serie 1 - 20 pts.

Se tiene un registro de personas y se necesita el almacenamiento de sus datos personales como nombres, apellidos, teléfono, dirección, fecha de nacimiento y otro dato personal de relevancia. Estas personas pueden tener o no uno o varios logros académicos de los cuales pueden ser licenciatura, maestría, doctorado o también diplomados, certificaciones entre otros tipos de logros académicos, el logro académico debe de tener registrado año, título del programa, institución de la que lo obtuvo y otra información que sea necesaria y pertinente. De igual manera se desea que se registre sus áreas de desempeño de las personas, por ejemplo: ciencias de la computación, biología, ciencias económicas, entre otros, recordando que pueden tener varias áreas de desempeño. Estas personas registradas deben de contar con usuario y contraseña del sistema.

Se desea que se construya una base de datos relacional del anterior enunciado.



Select M From areas



GOBIERNO DE
GUATEMALA

SECRETARÍA
NACIONAL DE
CIENCIA Y TECNOLOGÍA

Serie II - 30 pts.

Se necesita que realice un frontend en una librería o framework basada en JQuery que tenga lo siguiente:

- Login para las personas registradas
 - Actualización de datos personales
 - Actualización de datos académicos
 - Actualizar área de desempeño
 - Agregar o quitar logros académicos
 - Agregar o quitar área de desempeño
- Una tabla (no se necesita un login para acceder) en la cual se pueda visualizar las personas registradas, así como el último grado académico que han conseguido con su respectiva información de dicho grado académico.
- Una tabla (no se necesita un login para acceder) en la cual se pueda visualizar las áreas de desempeño registradas, así como el número de personas que tienen esa área de desempeño.

1) Tabla

ID - Persona | Nombre | Grado

2) Tabla

~~ID - Área de Desempeño~~ | Número de personas



Serie III – 50 pts.

Se necesita que realice un *backend* de todo el manejo de base de datos (inserciones, actualizaciones, consultas, entre otros) y *lógica del negocio* por medio de APIs Rest en Node JS.

Aspectos que evaluar

- Modelo relacional creado con sentencias SQL
- Uso adecuado de llaves primarias, foráneas e índices (este último si es necesario)
- Uso adecuado de *framework* o librería JQuery, si lo necesita *puede apoyarse de PHP*.
- El acceso a datos, manejo de estos, lógica de negocio y demás debe de ser *desarrollado netamente en Node JS*.
- Es libre utilizar los módulos de Node JS que necesite para crear las rutas de las APIs Rest o demás módulos.
- Para consumir las *APIs Rest* puede hacerlo desde JavaScript o PHP.