

CREATE TABLE DATOSPERSONALES(

```
idperfil          int          primary key,
descripcionperfil varchar(50),
perfilactivo      int,
apellidos          varchar(60),
nombres            varchar(60),
nacionalidad       varchar(20),
lugarnacimiento    varchar(60),
fechanacimiento    date,
numerocedula       varchar(10),
sexo               varchar(1),
estadocivil        varchar(50),
licenciaconducir   varchar(6),
telefonoconvencional varchar(15),
telefonofijo       varchar(15),
direcciontrabajo   varchar(50),
direcciondomiciliaria varchar(50),
sitioweb           varchar(60),
constraint chk_datospersonales_sexo check(sexo in('H', 'M')),
constraint uq_datospersonales_ci unique(numerocedula)
)
```

CREATE TABLE EXPERIENCIALABORAL(

```
idexperiencialaboral int,
idperfilconqueestaactivo int,
cargodesempenado     varchar(100),
nombreempresa         varchar(50),
lugarempresa          varchar(50),
emailempresa          varchar(100),
sitiowebempresa        varchar(100),
nombrecontactoempresarial varchar(100),
telefonocontactoempresarial varchar(60),
fechainiciogestion     date,
fechafingestion        date,
descripcionfunciones   varchar(100),
activarparaqueseveaenfront bit          DEFAULT 1,
rutacertificado        varchar(100),
constraint fk_experiencialaboral_datospersonales foreign key(idperfilconqueestaactivo) references datospersonales(idperfil)
)
```

--BD DOCUMENTAL

PROGRAMAR

FRONT que no aparezcan los
FRONT programar aparezcan
FRONT con el idperfilconqueestaactivo
FRONT que no aparezcan las

CREATE TABLE RECONOCIMIENTOS(

--Base de datos documento

```
idreconocimiento    int,
idperfilconqueestaactivo int,
tiporeconocimiento   varchar(100),
fehareconocimiento   date,
descripcionreconocimiento varchar(100),
entidadpatrocinadora varchar(100),
```

```

nombrecontactoauspicio    varchar(100),
telefonocontactoauspicio  varchar(60),
activarparaqueseveaenfront bit          DEFAULT 1,
rutacertificado            varchar(100),
constraint chk_reconocimiento_tiporeconocimiento check(tiporeconocimiento in ('Académico', 'Público', 'Pr
)

```

```

CREATE TABLE CURSOSREALIZADOS(           --Base de datos documento
idcursorealizado          int,
idperfilconqueestaactivo  int,
nombrecurso               varchar(100),
fechainicio               date,
fechafin                  date,
totalhoras                 int,
descripcioncurso          varchar(100),
entidadpatrocinadora      varchar(100),
nombrecontactoauspicio    varchar(100),
telefonocontactoauspicio  varchar(60),
emailempresapatrocinadora varchar(60),
activarparaqueseveaenfront bit          DEFAULT 1,
rutacertificado            varchar(100),
constraint fk_cursosrealizados_datospersonales foreign key(idperfilconqueestaactivo) references datospers
)

```

```

CREATE TABLE PRODUCTOSACADEMICOS(
idproductoacademico       int,
idperfilconqueestaactivo  int,
nombrerecurso             varchar(100),
clasificador              varchar(100),
descripcion               varchar(100),
activarparaqueseveaenfront bit          DEFAULT 1,
constraint fk_productosacademicos_datospersonales foreign key(idperfilconqueestaactivo) references dato
)

```

```

CREATE TABLE PRODUCTOSLABORALES(
idproductoslaborales      int,
idperfilconqueestaactivo  int,
nombreproducto            varchar(100),
fechaproducto             date,
descripcion               varchar(100),
activarparaqueseveaenfront bit          DEFAULT 1,
constraint fk_productoslaborales_datospersonales foreign key(idperfilconqueestaactivo) references datospe
)

```

```

CREATE TABLE VENTAGARAGE(
idventagarage             int,
idperfilconqueestaactivo  int,
nombreproducto            varchar(100),

```

```
estadoproducto          varchar(40),
descripcion              varchar(100),
valordelbien             numeric(5,2),
activarparaqueseveaenfront bit          DEFAULT 1,
constraint chk_ventagarage_estadoproducto check(estadoproducto in ('Bueno', 'Regular')),
constraint fk_ventagarage_datospersonales foreign key(idperfilconqueestaactivo) references datospersonal
)
```

--CAMBIO EN ETAPA2

