



UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO
FACULTAD DE INGENIERÍA

DIVISIÓN DE INGENIERÍA ELÉCTRICA
COORDINACIÓN DE COMPUTACIÓN

BASES DE DATOS

Tarea 3

**Diseño Conceptual de una Base de
Datos**

Nombre del alumno: Yukioayax Canek Gabriel Hernández

Profesor: Fernando Arreola Franco

Grupo: 1

Semestre en curso: 2026-2

1. Creación de usuario con límites y vigencia

Para crear un usuario en PostgreSQL con una contraseña, un límite de conexiones y una fecha de expiración, se utiliza el comando `CREATE USER` (que es un alias de `CREATE ROLE` con atributo `LOGIN`), la sintaxis incluye las cláusulas `ENCRYPTED PASSWORD`, `CONNECTION LIMIT` y `VALID UNTIL`.

1.1. Comando para crear el usuario

Supongamos que deseamos crear el usuario `alumno_tarea` con:

- Contraseña: `'MiClave2026'`
- Límite de conexiones: 5
- Vigencia: 1 mes a partir de una fecha determinada (por ejemplo, hasta el 18 de abril de 2026 si hoy es 18 de marzo).

El comando sería:

```
CREATE USER alumno_tarea
WITH
    ENCRYPTED PASSWORD 'MiClave2026'
    CONNECTION LIMIT 5
    VALID UNTIL '2026-04-18';
```

1.2. Verificación

Para comprobar los atributos del usuario creado se puede consultar la vista `pg_roles`:

```
SELECT rolname, rolconlimit, rolvaliduntil
FROM pg_roles
WHERE rolname = 'alumno_tarea';
```

2. Creación de un rol y asignación de permisos sobre la tabla estudiante

En PostgreSQL, un **rol** puede ser un usuario (si tiene login) o un grupo de permisos, aquí crearemos un rol sin login (grupo) para otorgarle permisos específicos sobre la tabla `estudiante`.

2.1. Crear el rol

```
CREATE ROLE rol_edicion_estudiante;
```

2.2. Asignar permisos sobre la tabla estudiante

Se asumen los permisos de lectura (SELECT), actualización (UPDATE) y borrado (DELETE) sobre la tabla `estudiante` (la cual debe existir previamente en la base de datos).

```
GRANT SELECT, UPDATE, DELETE ON TABLE estudiante TO rol_edicion_estudiante;
```

Si se desea que el rol también pueda aplicar estos permisos a otros (opción `WITH GRANT OPTION`), se añade al final, aunque no es requerido en el enunciado.

3. Asignación del rol al usuario

Finalmente, se otorga el rol creado al usuario `alumno_tarea`, para que este herede los permisos sobre la tabla `estudiante`.

```
GRANT rol_edicion_estudiante TO alumno_tarea;
```

A partir de este momento, el usuario `alumno_tarea` puede conectarse a la base de datos (hasta 5 conexiones simultáneas, y solo hasta la fecha indicada) y, gracias al rol asignado, puede consultar, modificar y eliminar registros de la tabla `estudiante`.

3.1. Verificación de los permisos

Para listar los permisos efectivos del usuario se puede utilizar:

```
SELECT * FROM information_schema.role_table_grants  
WHERE grantee = 'rol_edicion_estudiante';
```

O consultar directamente los privilegios sobre la tabla:

```
\dp estudiante
```

(comando `psql`).

4. Consideraciones adicionales

- La tabla `estudiante` debe existir en la base de datos donde se ejecutan estos comandos, si no existe, se puede crear con una estructura simple, por ejemplo:

```
CREATE TABLE estudiante (  
    id SERIAL PRIMARY KEY,  
    nombre VARCHAR(100),  
    carrera VARCHAR(50)  
);
```

- Los comandos mostrados son específicos de PostgreSQL. En otros motores de bases de datos la sintaxis puede variar.
- La fecha de vigencia debe estar en formato `YYYY-MM-DD` y se considera la hora `00:00:00` de ese día a menos que se especifique la hora.

Referencias

- [1] C. J. Date, *An Introduction to Database Systems*, 8th ed. Massachusetts: Addison Wesley, 2003.
- [2] R. Elmasri y S. Navathe, *Fundamentos de sistemas de bases de datos*, 3ra ed. Pearson Prentice Hall, 2003.
- [3] D. Kroenke, *Procesamiento de bases de datos*, 8a ed. México: Pearson / Prentice Hall, 2003.
- [4] P. Rob y C. Coronel, *Database Systems: Design, Implementation and Management*, 6th ed. Course Technology, 2004.
- [5] A. De Miguel Martínez, M. Piattini et al., *Diseño de bases de datos relacionales*. México: Alfamaga, 2000.
- [6] J. Johnson, *Bases de datos, modelos, lenguajes, diseño*. México: Oxford University Press, 2000.