



# UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



# FIME

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

## Facultad Autónoma de Nuevo León

### Facultad de Ingeniería Mecánica y Eléctrica

#### AF5: Diccionario de Datos y el Diseño de Pseudocódigos

**Materia :** Programación Orientada a Objetos

Nombre	Matricula	Carrera
Alfonso Alejandro Virgen Delgado	2177999	ITS
Angel Martin Gonzalez Esquivel	2096529	ITS
Christian Sebastian Moreno Gonzalez	1950199	ITS

**Hora :** N3

**Grupo :** 017

**Docente :** Ana Karen Antopia Barrón

**Fecha :** 30 de Marzo de 2025

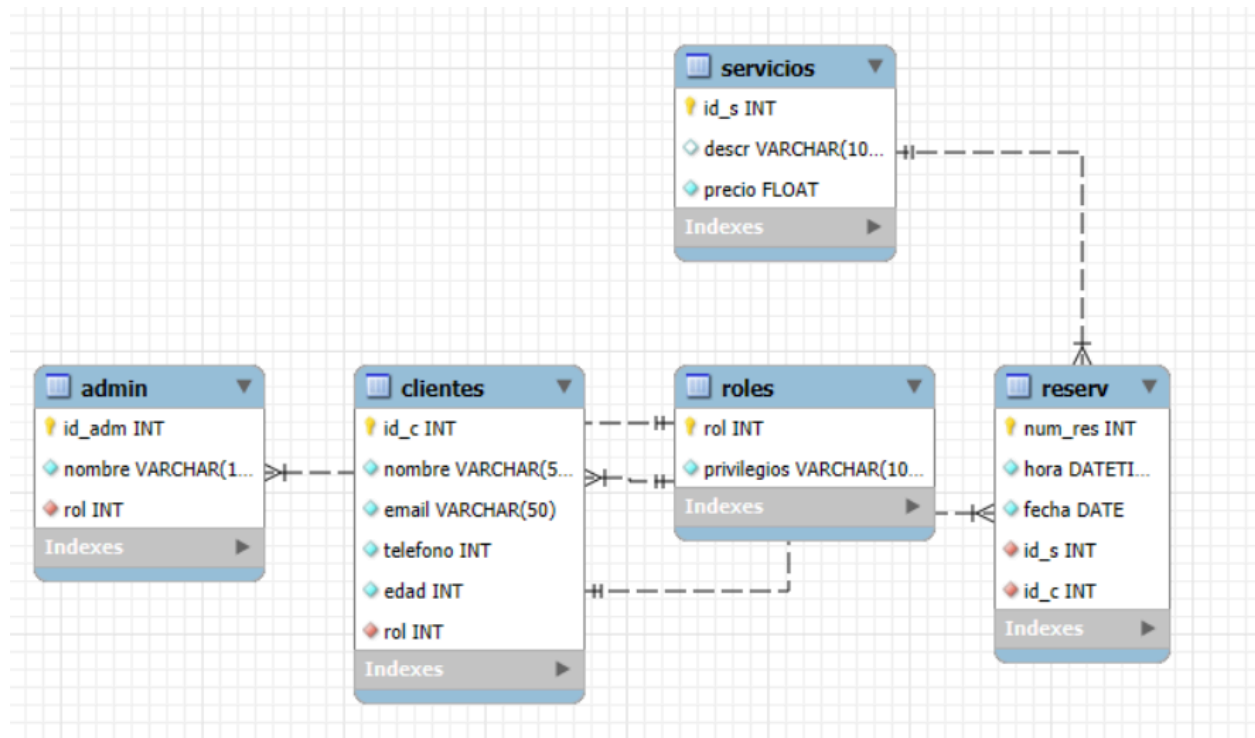
# 1. Diccionario de Datos

**Tabla 1: Definición de Datos del Sistema**

Nombre	Tipo de Variable	Largo	Descripción
id_c	entero	2	Número de identificación unico del cliente
nombre	varchar	50	Nombre del cliente, administrador o barbero
email	varchar	50	correo electrónico del cliente
contraseña	varchar	20	Contraseña de un usuario para acceder a su sesión ya sea administrador, cliente, o barbero
teléfono	varchar	10	Número de teléfono del cliente
rol	entero	1	Número de rol del usuario, esto para identificar si es un cliente o un administrador
id_adm	entero	2	Número de identificación unico de administrador
privilegios	varchar	100	Permisos que tendrá el usuario, determinando qué acciones puede hacer en la página
id_s	entero	2	Número de identificación único de un servicio
descr	varchar	200	Descripción sobre el servicio
precio	flotante	10	Precio del servicio
num_res	entero	2	Número de identificación de una reservación
fecha_act	date time	10	Día del mes y hora en la cual se hizo la cita

fecha_cit	date time	10	Dia del mes y hora para el cual se programó la cita
id_bar	entero	2	Número de identificación del barbero
esp_bar	varchar	100	Especialidad del barbero y los servicios que puede hacer
disp_bar	boolean	1	Estado del barbero, ya sea ocupado o disponible

## Evidencias



## 2. Diseño de Pseudocódigos

"DAR DE ALTA A UN CLIENTE"

INICIO

IMPRIMIR "ALTA DE CLIENTE - BARBERÍA ONLINE"

// Datos personales

IMPRIMIR "Ingrese su nombre completo:"

LEER nombre\_completo

IMPRIMIR "Ingrese su teléfono:"

LEER telefono

IMPRIMIR "Ingrese su correo electrónico:"

LEER correo

// Validar correo único

MIENTRAS correo EXISTE EN base\_de\_datos

IMPRIMIR "Este correo ya está registrado. Ingrese otro:"

LEER correo

FIN MIENTRAS

// Credenciales de acceso

IMPRIMIR "Ingrese su contraseña:"

LEER contrasena

ENCRIPTAR contraseña

// Información adicional para barbería

IMPRIMIR "Seleccione su tipo de corte preferido (1-Caballero, 2-Dama, 3-Niño):"

LEER tipo\_corte\_preferido

IMPRIMIR "¿Desea recibir promociones? (S/N):"

LEER recibir\_promociones

// Guardar en base de datos

INSERTAR EN clientes (

nombre\_completo,

telefono,

correo,

contrasena,

tipo\_corte\_preferido,

recibir\_promociones,

fecha\_registro = HOY()

)

// Confirmación

IMPRIMIR "¡Registro exitoso! Bienvenido a nuestra barbería online."

IMPRIMIR "Se ha enviado un correo de confirmación a: ", correo

FIN

## Explicación:

Este algoritmo en pseudocódigo registra clientes en nuestra página de una barbería. Solicita datos personales, valida que el correo sea único, encripta la contraseña y guarda la información en la base de datos. También permite seleccionar un tipo de corte preferido y recibir promociones. Finalmente, confirma el registro y envía un correo de bienvenida.

### "RESERVACIÓN DE CITAS"

#### INICIO

IMPRIMIR "=== RESERVA DE CITA - BARBERÍA ONLINE ==="

// Autenticación

IMPRIMIR "Ingrese su correo electrónico:"

LEER correo

IMPRIMIR "Ingrese su contraseña:"

LEER contrasena

ENCRIPtar contraseña

// Validar credenciales

SI (correo, contrasena) NO EXISTE EN base\_de\_datos ENTONCES

IMPRIMIR "Credenciales incorrectas. Intente nuevamente o regístrese."

FIN

// Mostrar servicios disponibles

IMPRIMIR "Servicios disponibles:"

IMPRIMIR "1. Corte de caballero - \$150"

IMPRIMIR "2. Corte y barba - \$200"

IMPRIMIR "3. Corte de niño - \$120"

IMPRIMIR "4. Afeitado clásico - \$100"

IMPRIMIR "Seleccione el servicio (1-4):"

LEER servicio\_seleccionado

// Mostrar barberos disponibles

IMPRIMIR "Barberos disponibles:"

CONSULTAR barberos\_disponibles DESDE base\_de\_datos

IMPRIMIR barbero.id, "-", barbero.nombre, "- Especialidad:", barbero.especialidad

IMPRIMIR "Seleccione su barbero preferido:"

LEER barbero\_seleccionado

// Selección de fecha y hora

IMPRIMIR "Ingrese la fecha deseada (DD/MM/AAAA):"

LEER fecha\_cita

IMPRIMIR "Horarios disponibles para", fecha\_cita, ":"

CONSULTAR horarios\_disponibles DESDE base\_datos

```

IMPRIMIR horario.hora_inicio, "-", horario.hora_fin
IMPRIMIR "Seleccione el horario deseado:"
LEER horario_seleccionado
// Confirmación
IMPRIMIR "Resumen de su cita:"
IMPRIMIR "Servicio:", servicio_seleccionado
IMPRIMIR "Barbero:", barbero_seleccionado
IMPRIMIR "Fecha:", fecha_cita
IMPRIMIR "Hora:", horario_seleccionado
IMPRIMIR "¿Confirmar cita? (S/N):"
LEER confirmacion

SI confirmacion = "S" ENTONCES
    INSERTAR EN citas (
        cliente = correo,
        servicio = servicio_seleccionado,
        barbero = barbero_seleccionado,
        fecha = fecha_cita,
        hora = horario_seleccionado,
        estado = "Pendiente"
    )

    IMPRIMIR "¡Cita reservada con éxito!"
    IMPRIMIR "Se ha enviado un correo de confirmación a:", correo
SI confirmación es = N
    IMPRIMIR "Reserva cancelada"

```

FIN

## Explicación

Este algoritmo en pseudocódigo permite a los clientes reservar citas en una barbería.

Primero, autentica al usuario validando su correo y contraseña. Luego, muestra los servicios y barberos disponibles, permitiendo al cliente seleccionar uno. A continuación, solicita la fecha y hora deseada y muestra los horarios disponibles. Finalmente, tras confirmar la cita, la guarda en la base de datos y envía un correo de confirmación. Si el cliente no confirma, la reserva se cancela.

### 3. Dificultades Encontradas y Próximos Pasos

Dificultades detectadas:

Optimización de la estructura del diccionario de datos para mejorar la eficiencia de las consultas.

Ajustes en la lógica de ejecución del código dentro del sistema.

Integración de los algoritmos con las estructuras de base de datos.

Optimización de la estructura del diccionario de datos para mejorar la eficiencia de las consultas:

- La base de datos podría optimizarse con índices en campos clave como correo (para validaciones rápidas).
- Usar estructuras normalizadas para evitar redundancias en información como barberos y servicios.
- Posible uso de caché o consultas optimizadas para mejorar tiempos de respuesta.

Ajustes en la lógica de ejecución del código dentro del sistema:

- Manejo de errores más robusto para evitar interrupciones en el flujo.
- Mejor estructuración del proceso de validación, evitando bucles innecesarios.
- Implementación de una mejor gestión de sesiones para la autenticación.

Integración de los algoritmos con las estructuras de base de datos:

- Asegurar que las consultas SQL sean seguras y eficientes (uso de Prepared Statements).
- Validar que los formatos de datos sean compatibles entre el código y la base de datos.
- Implementar triggers o procedimientos almacenados para automatizar ciertas verificaciones (como disponibilidad de horarios).

Acciones a seguir:

- Refinar los pseudocódigos para asegurar compatibilidad con el lenguaje de implementación.
- Implementar una primera versión del sistema basada en los algoritmos diseñados.
- Realizar pruebas de validación en la base de datos.

Optimización del Diccionario de Datos

Acciones:

- Indexar campos clave: Agregar índices en correo, teléfono y fecha\_cita para consultas más rápidas.
- Normalización de datos:
  - Crear una tabla de servicios en lugar de almacenar valores repetidos en cada cita.
  - Usar claves foráneas en lugar de guardar información redundante en la tabla de citas.
- Optimizar estructuras de datos en código:
  - Si manejas datos en memoria antes de insertarlos en la base de datos, usa diccionarios en lugar de listas para accesos más rápidos.

Fecha estimada de finalización de la siguiente fase: [24/04/2025]