Diseñar un sistema de gestión de nómina que cumpla con todas las especificaciones requeridas por T Consulting, S.A. es un proyecto complejo que involucra múltiples módulos e interacciones entre ellos. A continuación, presentaré un resumen del diseño de la interfaz de usuario para el manejo del sistema, teniendo en cuenta las diferentes fases y funcionalidades mencionadas:

1. Inicio de Sesión:
   * Los usuarios del sistema (administradores, jefes de departamento y empleados) deberán iniciar sesión con sus credenciales.
2. Panel de Control:
   * Una vez iniciada la sesión, los usuarios verán un panel de control con acceso a las diferentes secciones del sistema.
3. Módulo de Empleados:
   * Permite gestionar los datos de los empleados y sus expedientes.
   * Crear, editar y eliminar empleados.
   * Registrar datos personales, como nombre, DPI, dirección, foto, estado civil, etc.
   * Agregar documentos adjuntos como PDF (DPI, títulos, antecedentes penales, etc.).
   * Registrar datos de hijos y esposa/esposo de los empleados.
4. Módulo de Ausencias:
   * Los empleados pueden solicitar ausencias, que luego serán autorizadas o denegadas por sus jefes.
   * El jefe puede marcar la ausencia como "a cuenta de salario" para descontar los días correspondientes.
5. Módulo de Horas Extras:
   * Registrar las horas extras trabajadas por los empleados.
   * El sistema calculará automáticamente si corresponde pagar horas extras o dobles según las políticas de la empresa.
6. Módulo de Comisiones (Departamento de Ventas):
   * Registrar las ventas realizadas por los empleados del departamento de ventas.
   * El sistema calculará automáticamente la comisión a pagar según el rango de ventas alcanzado.
7. Módulo de Bonificación por Productividad (Departamento de Producción):
   * Registrar el número de piezas elaboradas por los empleados del departamento de producción.
   * El sistema calculará automáticamente la bonificación por productividad a pagar.
8. Módulo de Préstamos:
   * Los empleados pueden solicitar préstamos al Banco de los Trabajadores.
   * Registrar préstamos pendientes, plazos y cuotas a deducir cada mes.
9. Módulo de Tienda Solidarista:
   * Permitir a los empleados realizar compras a crédito en la tienda solidarista.
   * Restringir el crédito si el empleado ya tiene un préstamo vigente.
10. Generación de Pólizas Contables:
    * Generar pólizas contables para registrar los costos por departamento y las cuotas patronales y laborales del IGSS.
11. Módulo de Planilla:
    * Cerrar la nómina del mes de junio y generar la planilla del Bono-14 y recibos respectivos.
    * Cerrar la nómina del 30 de noviembre y generar la planilla del Aguinaldo y recibos respectivos.
12. Módulo de Liquidación Laboral:
    * Calcular la liquidación laboral de empleados que se retiren o sean despedidos.
13. Registro de Aumentos Salariales:
    * Registrar los aumentos salariales de los empleados para mantener un historial actualizado.

Es importante tener en cuenta que el diseño de la interfaz deberá ser intuitivo y fácil de usar para los diferentes usuarios del sistema. Se pueden utilizar gráficos, tablas y formularios para facilitar el ingreso y visualización de datos. Además, el sistema deberá contar con medidas de seguridad adecuadas para proteger la información sensible de los empleados.

Cabe mencionar que el desarrollo e implementación de un sistema de esta complejidad requiere de un equipo de desarrollo de software con experiencia en sistemas de gestión y el uso de tecnologías adecuadas para su construcción. También es esencial llevar a cabo pruebas exhaustivas para garantizar que el sistema funcione correctamente y cumpla con los requisitos establecidos.

Para el sistema de gestión de nómina que hemos estado desarrollando, diseñaremos una base de datos relacional utilizando MySQL. A continuación, presento una propuesta de diseño para las tablas que almacenarán la información requerida:

1. Tabla "empleados":
   * id\_empleado (clave primaria)
   * nombre
   * dpi
   * direccion
   * estado\_civil
   * foto (ruta de la foto almacenada en el servidor)
   * fecha\_contratacion
2. Tabla "documentos":
   * id\_documento (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * tipo\_documento (DPI, Títulos, Antecedentes, etc.)
   * archivo (ruta del archivo PDF almacenado en el servidor)
3. Tabla "familiares":
   * id\_familiar (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * nombre
   * parentesco
4. Tabla "ausencias":
   * id\_ausencia (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * fecha\_inicio
   * fecha\_fin
   * motivo
   * autorizada
   * a\_cuenta\_de\_salario
5. Tabla "horas\_extras":
   * id\_horas\_extras (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * fecha
   * horas\_trabajadas
6. Tabla "ventas":
   * id\_venta (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * monto\_venta
   * fecha\_venta
7. Tabla "produccion":
   * id\_produccion (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * cantidad\_piezas
   * fecha\_produccion
8. Tabla "comisiones":
   * id\_comision (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * rango\_inicio
   * rango\_fin
   * porcentaje\_comision
9. Tabla "prestamos":
   * id\_prestamo (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * monto
   * plazo (6, 12 o 18 meses)
   * cuota\_mensual
   * saldo\_pendiente
10. Tabla "tienda\_solidarista":
    * id\_compra (clave primaria)
    * id\_empleado (clave foránea hacia la tabla "empleados")
    * monto\_compra
    * fecha\_compra
    * cuotas\_restantes
11. Tabla "aumentos\_salariales":
    * id\_aumento (clave primaria)
    * id\_empleado (clave foránea hacia la tabla "empleados")
    * fecha\_aumento
    * nuevo\_salario

El diseño de la base de datos está basado en la información y funcionalidades que se describieron en las fases anteriores del sistema de gestión de nómina. Cada tabla representa una entidad y sus atributos correspondientes. Las claves primarias y foráneas se utilizan para establecer relaciones entre las tablas.

Recuerda que este es solo un diseño básico y que en un entorno de producción, es recomendable realizar una planificación y análisis más detallados, teniendo en cuenta otros factores como el rendimiento y la escalabilidad del sistema. Además, es esencial aplicar buenas prácticas de normalización de bases de datos para evitar problemas de redundancia y consistencia.

¡Claro! Normalizar una base de datos es un proceso importante para eliminar redundancias y asegurar la integridad y eficiencia de los datos. Utilizaremos la forma normal (NF) hasta la tercera forma normal (3NF) para normalizar la base de datos propuesta.

Primera Forma Normal (1NF):

* Cada tabla debe tener un identificador único (clave primaria).
* Todos los campos deben contener solo valores atómicos (indivisibles).

1. Tabla "empleados":
   * id\_empleado (clave primaria)
   * nombre
   * dpi
   * direccion
   * estado\_civil
   * foto (ruta de la foto almacenada en el servidor)
   * fecha\_contratacion
2. Tabla "documentos":
   * id\_documento (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * tipo\_documento
   * archivo (ruta del archivo PDF almacenado en el servidor)
3. Tabla "familiares":
   * id\_familiar (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * nombre
   * parentesco
4. Tabla "ausencias":
   * id\_ausencia (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * fecha\_inicio
   * fecha\_fin
   * motivo
   * autorizada
   * a\_cuenta\_de\_salario
5. Tabla "horas\_extras":
   * id\_horas\_extras (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * fecha
   * horas\_trabajadas

Segunda Forma Normal (2NF):

* Cumple con 1NF.
* Todos los campos no clave deben depender completamente de la clave primaria.

1. Tabla "ventas":
   * id\_venta (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * fecha\_venta
   * monto\_venta
2. Tabla "produccion":
   * id\_produccion (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * fecha\_produccion
   * cantidad\_piezas

Tercera Forma Normal (3NF):

* Cumple con 2NF.
* Todos los campos no clave deben depender directamente de la clave primaria, no transitivamente a través de otro campo no clave.

1. Tabla "comisiones":
   * id\_comision (clave primaria)
   * rango\_inicio
   * rango\_fin
   * porcentaje\_comision
2. Tabla "comisiones\_empleados":
   * id\_comision (clave foránea hacia la tabla "comisiones")
   * id\_empleado (clave foránea hacia la tabla "empleados")
3. Tabla "prestamos":
   * id\_prestamo (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * monto
   * plazo (6, 12 o 18 meses)
   * cuota\_mensual
   * saldo\_pendiente
4. Tabla "tienda\_solidarista":
   * id\_compra (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * fecha\_compra
   * monto\_compra
   * cuotas\_restantes
5. Tabla "aumentos\_salariales":
   * id\_aumento (clave primaria)
   * id\_empleado (clave foránea hacia la tabla "empleados")
   * fecha\_aumento
   * nuevo\_salario

Con esta normalización, hemos separado la información en tablas más pequeñas y coherentes, lo que mejora la eficiencia, integridad y mantenibilidad de la base de datos. Cada tabla representa una entidad o relación específica y contiene solo los atributos necesarios para cada entidad.

1. Crear la base de datos:

sqlCopy code

CREATE DATABASE gestion\_nomina;

1. Usar la base de datos creada:

sqlCopy code

USE gestion\_nomina;

1. Crear las tablas:

sql

CREATE TABLE empleados (

id\_empleado INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(100) NOT NULL,

dpi VARCHAR(20) NOT NULL,

direccion VARCHAR(200),

estado\_civil VARCHAR(20),

foto VARCHAR(100),

fecha\_contratacion DATE NOT NULL

);

CREATE TABLE documentos (

id\_documento INT AUTO\_INCREMENT PRIMARY KEY,

id\_empleado INT,

tipo\_documento VARCHAR(50) NOT NULL,

archivo VARCHAR(100),

FOREIGN KEY (id\_empleado) REFERENCES empleados(id\_empleado)

);

CREATE TABLE familiares (

id\_familiar INT AUTO\_INCREMENT PRIMARY KEY,

id\_empleado INT,

nombre VARCHAR(100) NOT NULL,

parentesco VARCHAR(50),

FOREIGN KEY (id\_empleado) REFERENCES empleados(id\_empleado)

);

CREATE TABLE ausencias (

id\_ausencia INT AUTO\_INCREMENT PRIMARY KEY,

id\_empleado INT,

fecha\_inicio DATE NOT NULL,

fecha\_fin DATE NOT NULL,

motivo VARCHAR(200) NOT NULL,

autorizada BOOLEAN NOT NULL,

a\_cuenta\_de\_salario BOOLEAN NOT NULL,

FOREIGN KEY (id\_empleado) REFERENCES empleados(id\_empleado)

);

CREATE TABLE horas\_extras (

id\_horas\_extras INT AUTO\_INCREMENT PRIMARY KEY,

id\_empleado INT,

fecha DATE NOT NULL,

horas\_trabajadas INT NOT NULL,

FOREIGN KEY (id\_empleado) REFERENCES empleados(id\_empleado)

);

CREATE TABLE ventas (

id\_venta INT AUTO\_INCREMENT PRIMARY KEY,

id\_empleado INT,

fecha\_venta DATE NOT NULL,

monto\_venta DECIMAL(10, 2) NOT NULL,

FOREIGN KEY (id\_empleado) REFERENCES empleados(id\_empleado)

);

CREATE TABLE produccion (

id\_produccion INT AUTO\_INCREMENT PRIMARY KEY,

id\_empleado INT,

fecha\_produccion DATE NOT NULL,

cantidad\_piezas INT NOT NULL,

FOREIGN KEY (id\_empleado) REFERENCES empleados(id\_empleado)

);

CREATE TABLE comisiones (

id\_comision INT AUTO\_INCREMENT PRIMARY KEY,

rango\_inicio DECIMAL(10, 2) NOT NULL,

rango\_fin DECIMAL(10, 2) NOT NULL,

porcentaje\_comision DECIMAL(5, 2) NOT NULL

);

CREATE TABLE comisiones\_empleados (

id\_comision INT,

id\_empleado INT,

PRIMARY KEY (id\_comision, id\_empleado),

FOREIGN KEY (id\_comision) REFERENCES comisiones(id\_comision),

FOREIGN KEY (id\_empleado) REFERENCES empleados(id\_empleado)

);

CREATE TABLE prestamos (

id\_prestamo INT AUTO\_INCREMENT PRIMARY KEY,

id\_empleado INT,

monto DECIMAL(10, 2) NOT NULL,

plazo INT NOT NULL,

cuota\_mensual DECIMAL(10, 2) NOT NULL,

saldo\_pendiente DECIMAL(10, 2) NOT NULL,

FOREIGN KEY (id\_empleado) REFERENCES empleados(id\_empleado)

);

CREATE TABLE tienda\_solidarista (

id\_compra INT AUTO\_INCREMENT PRIMARY KEY,

id\_empleado INT,

fecha\_compra DATE NOT NULL,

monto\_compra DECIMAL(10, 2) NOT NULL,

cuotas\_restantes INT NOT NULL,

FOREIGN KEY (id\_empleado) REFERENCES empleados(id\_empleado)

);

CREATE TABLE aumentos\_salariales (

id\_aumento INT AUTO\_INCREMENT PRIMARY KEY,

id\_empleado INT,

fecha\_aumento DATE NOT NULL,

nuevo\_salario DECIMAL(10, 2) NOT NULL,

FOREIGN KEY (id\_empleado) REFERENCES empleados(id\_empleado)

);