

## 2. Análisis del Sistema Contable "Comercial el mejor vendedor S.A."

A continuación, se presenta un análisis detallado de la arquitectura, funcionalidad y especificaciones del sistema contable, basado en el código fuente proporcionado.

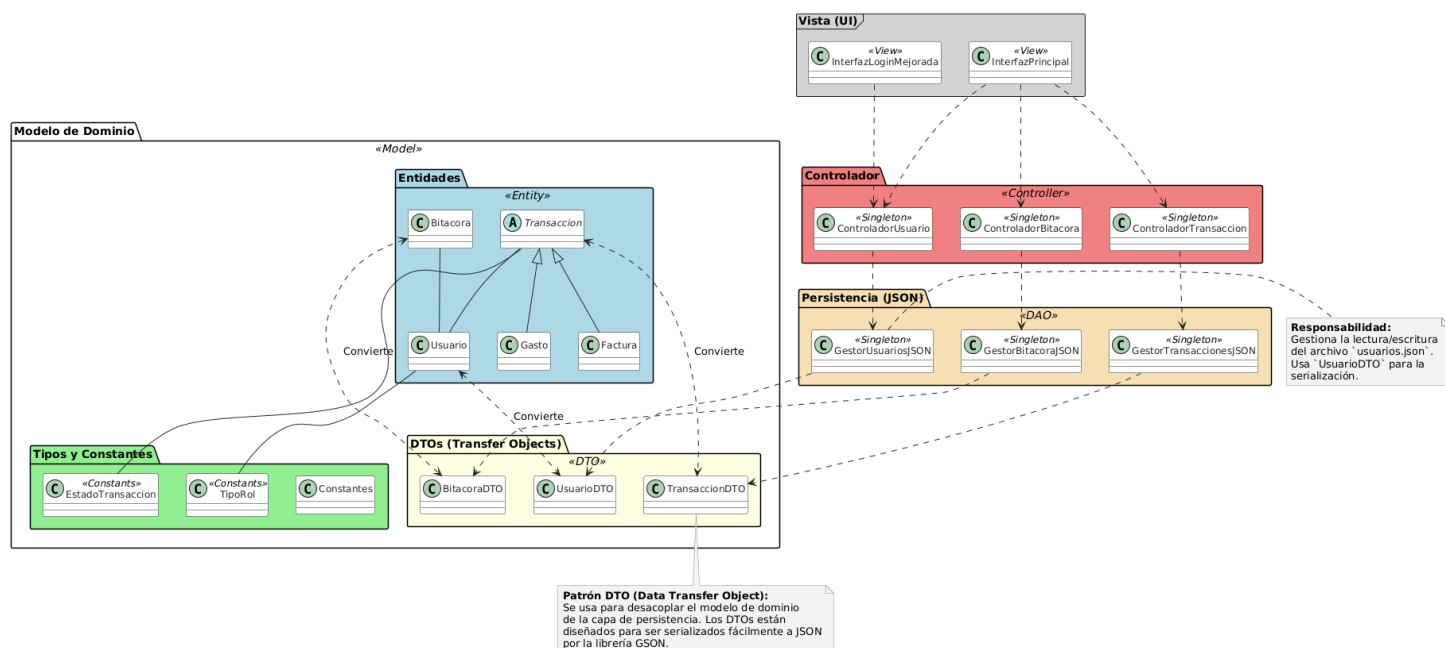
### 2.1 Modelo Lógico del Dominio (Diagrama de Clases UML)

El sistema está diseñado siguiendo una arquitectura de capas que se asemeja al patrón **Modelo-Vista-Controlador (MVC)**. El siguiente diagrama de clases UML ilustra esta estructura, mostrando cómo las responsabilidades se separan en componentes lógicos para facilitar el mantenimiento y la escalabilidad.

#### Descripción de la Arquitectura:

- **Capa de Vista (UI):** Compuesta por las clases `InterfazLoginMejorada` e `InterfazPrincipal`, es responsable de presentar la información al usuario y capturar sus interacciones. No contiene lógica de negocio.
- **Capa de Controlador:** Actúa como intermediario entre la Vista y el Modelo. Las clases `ControladorUsuario`, `ControladorTransaccion` y `ControladorBitacora` reciben las solicitudes de la vista y orquestan las operaciones necesarias, delegando la persistencia y la lógica de negocio a las capas inferiores.
- **Capa de Modelo y Datos:** Es el corazón del sistema y se subdivide en:
  - **Lógica del Dominio:** Contiene las entidades principales (`Usuario`, `Transaccion`, `Factura`, `Gasto`, `Bitacora`) que representan los conceptos del negocio.
  - **Acceso a Datos (DAO):** Los gestores (`Gestor...JSON`) se encargan exclusivamente de la persistencia, es decir, de leer y escribir los datos en los archivos JSON.
  - **Objetos de Transferencia (DTO):** Clases "planas" (`UsuarioDTO`, `TransaccionDTO`) diseñadas específicamente para facilitar la serialización y deserialización de datos hacia y desde el formato JSON.

#### Código del Diagrama (PlantUML):

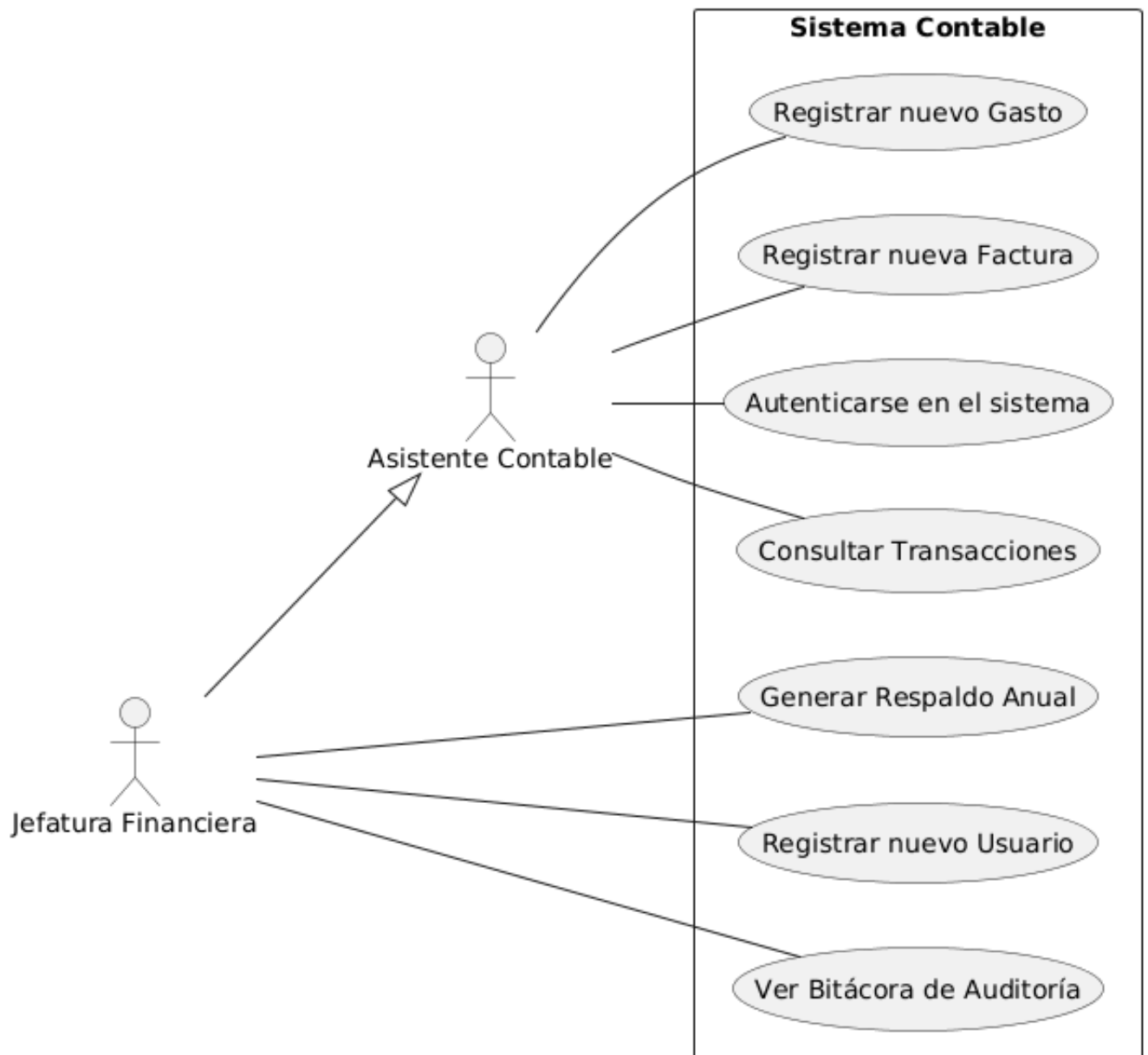


## 2.2 Diagrama de Casos de Uso y Escenarios Principales

El diagrama de casos de uso identifica los actores del sistema y las funcionalidades que pueden realizar.

### Actores:

1. **Asistente Contable:** Usuario operativo encargado del registro diario de facturas y gastos.
2. **Jefatura Financiera:** Usuario con privilegios administrativos, responsable de la supervisión, gestión de usuarios y generación de reportes. Este rol hereda todas las capacidades del Asistente Contable.



### Escenarios Principales:

- **Escenario 1: Registro de una nueva factura (Asistente Contable)**
  1. **Precondición:** El asistente ha iniciado sesión exitosamente.
  2. **Flujo Principal:**
    - El actor selecciona la opción "Registrar Factura" en la interfaz principal.
    - El sistema presenta un formulario para ingresar los detalles de la factura.
    - El actor completa los campos: cliente, RUC, monto, descripción y número de factura.

- Al confirmar, el ControladorTransaccion recibe los datos, crea un objeto Factura y lo valida.
- El controlador delega al GestorTransaccionesJSON la persistencia de la nueva factura en el archivo transacciones.json.
- El ControladorBitacora registra el evento de creación.
- El sistema actualiza la tabla de transacciones en la vista y muestra un mensaje de éxito.

- **Escenario 2: Aprobación de un Gasto (Jefatura Financiera)**

1. **Precondición:** La jefatura ha iniciado sesión y existe al menos un gasto en estado "Registrado".

2. **Flujo Principal:**

- El actor selecciona un gasto de la tabla de transacciones que se encuentra en estado "Registrado".
- El actor presiona el botón "Aprobar Transacción".
- La InterfazPrincipal notifica al ControladorTransaccion sobre la acción, pasando el ID de la transacción.
- El controlador verifica que el usuario actual tenga el rol JEFATURA\_FINANCIERA.
- Si el permiso es válido, el controlador actualiza el estado de la transacción a "Aprobado" y lo persiste a través del GestorTransaccionesJSON.
- Se registra la aprobación en la bitácora.
- La vista actualiza el estado de la transacción en la tabla.

## 2.3 Especificación de Datos (Esquemas JSON)

El sistema utiliza archivos en formato JSON para la persistencia de datos, lo que garantiza que sean legibles tanto por humanos como por máquinas.

- [usuarios.json](#): Almacena la información de los usuarios del sistema.

```
[
  {
    "idUsuario": 1,
    "nombreUsuario": "admin",
```

```

        "contrasena": "admin123",
        "nombreCompleto": "Administrador del Sistema",
        "rol": "JEFATURA_FINANCIERA"
    }
]

```

- **transacciones.json:** Contiene el registro histórico de todas las facturas y gastos. Un campo tipo diferencia cada registro.

```

[
  {
    "tipo": "factura",
    "idTransaccion": 101,
    "fecha": "2025-11-10",
    "monto": 1725.86,
    "estado": "APROBADO",
    "proveedorCliente": "Empresa XYZ S.A.",
    "idUsuarioRegistro": 1,
    "subtotal": 1500.75,
    "iva": 225.11
  },
  {
    "tipo": "gasto",
    "idTransaccion": 102,
    "fecha": "2025-11-11",
    "monto": 350.00,
    "estado": "REGISTRADO",
    "proveedorCliente": "Papelería El Estudiante",
    "idUsuarioRegistro": 2,
    "deducible": true,
    "ivaCompra": 45.65
  }
]

```

- **bitacora.json:** Guarda un registro de auditoría de las acciones clave realizadas en el sistema.

```

[
  {
    "idRegistro": 1,
    "idUsuario": 1,
    "fechaHora": "2025-11-11T10:05:30.123456",
    "accion": "INICIO_SESION",
    "descripcion": "Inicio de sesión exitoso."
  }
]

```

]

## 2.4 Especificación de Criterios de Aceptación (Gherkin)

Los criterios de aceptación definen el comportamiento esperado del sistema desde la perspectiva del usuario, utilizando el formato Dado-Cuando-Entonces.

### Funcionalidad: Autenticación de Usuario

- **Criterio 1: Inicio de sesión exitoso**
  - **Dado** un usuario con credenciales válidas ("admin", "admin123") en la pantalla de login.
  - **Cuando** ingresa su nombre de usuario y contraseña y presiona "Ingresar".
  - **Entonces** el sistema le da acceso y muestra la interfaz principal.
- **Criterio 2: Inicio de sesión con credenciales incorrectas**
  - **Dado** un usuario en la pantalla de login.
  - **Cuando** ingresa un nombre de usuario "admin" y una contraseña incorrecta "password".
  - **Entonces** el sistema debe mostrar un mensaje de error indicando que las credenciales son incorrectas y no debe permitir el acceso.

### Funcionalidad: Registro de Nuevo Usuario (Exclusivo de Jefatura Financiera)

- **Criterio 1: Registro exitoso de un nuevo asistente**
  - **Dado** que un usuario con rol JEFATURA\_FINANCIERA ha iniciado sesión.
  - **Cuando** va a la sección de registro, ingresa un nuevo nombre de usuario "carlos", una contraseña "carlos123", confirma la contraseña y selecciona el rol ASISTENTE\_CONTABLE.
  - **Entonces** el sistema crea el nuevo usuario, lo persiste en [usuarios.json](#) y muestra un mensaje de "Usuario 'carlos' registrado exitosamente".
- **Criterio 2: Intento de registro con contraseñas no coincidentes**
  - **Dado** que la JEFATURA\_FINANCIERA está en el formulario de registro.
  - **Cuando** ingresa una contraseña "clave123" y una confirmación de contraseña "clave456".

- **Entonces** el sistema debe mostrar un mensaje de error indicando que "Las contraseñas no coinciden" y no debe crear el usuario.

### Funcionalidad: Gestión de Transacciones

- **Criterio 1: Registrar un gasto con datos válidos**
  - **Dado** que un Asistente Contable ha iniciado sesión.
  - **Cuando** navega a "Registrar Gasto", completa todos los campos obligatorios y confirma.
  - **Entonces** el sistema debe registrar el nuevo gasto en transacciones.json con estado "Registrado", mostrar un mensaje de éxito y actualizar la tabla de transacciones.
- **Criterio 2: Intento de aprobar transacción con rol no autorizado**
  - **Dado** que existe una transacción en estado "Registrado".
  - **Cuando** un usuario con rol ASISTENTE\_CONTABLE intenta aprobar dicha transacción.
  - **Entonces** el sistema debe denegar la acción, ya sea deshabilitando el botón "Aprobar" o mostrando un mensaje de "No tiene permisos para realizar esta acción".