Sistema de Gestión de Conciertos – StageLink – Modelos G01 a G05

# G01. Propósito

StageLink es una empresa dedicada a brindar soluciones tecnológicas para la industria musical, enfocándose en la gestión de eventos en estadios y arenas. Su objetivo principal es ofrecer herramientas que optimicen la organización de conciertos y espectáculos en grandes recintos.

Actualmente, la coordinación entre bandas, representantes y administradores de estadios suele ser desorganizada y propensa a errores. Las reservas se manejan de forma manual o mediante canales informales, lo que genera múltiples dificultades: confusión entre los involucrados, superposiciones de fechas que obligan a reprogramar o cancelar eventos, o mala asignación de sectores que afecta la experiencia del público, y perdida directa de oportunidades de venta. Esta falta de sistematización compromete la eficiencia del proceso y pone en riesgo el éxito del evento.

Para resolver este problema, StageLink propone una plataforma digital integral que centraliza la gestión de conciertos: permite registrar solicitudes, validar reservas, configurar sectores y vender entradas de forma controlada. Todo el proceso es gestionado desde un sistema unificado.

Con esta solución, se busca agilizar la logística, mejorar la precisión en la disponibilidad de fechas y sectores, automatizar la emisión de entradas, y brindar soporte a los distintos actores involucrados en la organización de un evento musical.

# G02. Descripción funcional y alcance

## RFN1. Gestión de reservas de conciertos

### PN1 - Agendar concierto

1. El proceso inicia cuando la Banda contacta a StageLink para solicitar una fecha para un concierto. Esta solicitud puede realizarse por correo, teléfono u otro medio habilitado. EL vendedor recopila los datos esenciales del solicitante para dar inicio al registro.  
   **Atributos registrados:**  
   Nombre\_Banda, NombreRepresentante, EmailRepresentante, TelefonoRepresentante.
2. Una vez recopilados los datos de la Banda, el **vendedor** ingresa al sistema StageLink y accede al módulo de agendamiento. Allí selecciona el estadio propuesto por la banda (o uno disponible) y la fecha sugerida para el evento. También puede ingresar una breve descripción del espectáculo.  
   **Atributos ingresados:**  
   Estadio, Fecha, Hora, DescripcionEvento.
3. El sistema valida automáticamente si el estadio seleccionado se encuentra disponible en la fecha y horario solicitados. Para ello, revisa posibles eventos ya agendados y restricciones operativas. Luego, muestra al **vendedor** un resumen con el estado de disponibilidad.  
   **Atributos mostrados:**  
   Estadio, Fecha, Capacidad, EstadoReserva.
4. Si el estadio se encuentra disponible, el **vendedor** procede a registrar la información técnica necesaria para el análisis del evento. Esto incluye el rider técnico (especificaciones de sonido, iluminación, etc.) y cualquier otra documentación provista por la Banda, y la validación general de la solicitud. Durante esta etapa, se verifica que toda la información este completa y que los requisitos técnicos y de disponibilidad estén correctamente cumplidos. Finalmente, el vendedor aprueba o rechaza la solicitud de agendamiento.  
   **Atributos ingresados y actualizados:**  
   RiderTecnico, DocumentacionAdicional, EstadoReserva, FechaValidacion, ObservacionesAdministrador.
5. Una vez completados los datos y cargada la documentación, el sistema registra la solicitud de agendamiento como una nueva entrada en la base de datos y la envía al área técnica para su revisión. En este punto, la reserva queda en estado **"Pendiente"**.  
   **Atributos registrados:**  
   EstadoReserva.

## RFN2. Gestión de entradas

### PN2 - Configuración y venta

1. El administrador configura los sectores y precios del estadio para el concierto programado 🡪 Atributos: Sector, Fila, Asiento, PrecioBase, CapacidadSector.
2. El sistema publica internamente las entradas disponibles para su venta (no publicadas, sino listas para empleados en boletería o sistema interno) 🡪 Atributos: FechaPublicacion, Concierto, Precio, Cantidad.
3. Un empleado de venta realiza la carga de la venta de entradas en el sistema, al momento de que un cliente compra presencialmente 🡪 Atributos: NombreCliete, EmailCliente, CantidadEntradas, MontoToal, QRTicket, FechaComra.
4. El sistema genera el Ticket (con código QR) y actualiza automáticamente la disponibilidad 🡪 Atributos: CodigoEntrada, QRTicket, FechaCompra.

# G03. Definiciones, acrónimos y abreviaciones

**Banda:** Grupo musical que solicita agendar un concierto en la plataforma a través de un vendedor.

**Concierto:** Evento musical en un estadio o arena.

**Estadio:** Recinto donde se realiza el concierto, con capacidad definida.

**Rider Técnico:** Documento con requerimientos técnicos del show.

**Entrada:** Documento digital o físico que habilita el acceso al concierto.

**QR Ticket:** Código único de entrada escaneable en el ingreso.

Acrónimos

- QR: Quick Response  
- ID: Identificador único  
- PDF: Portable Document Format

Abreviaciones

- Prec.: Precio  
- Cant.: Cantidad  
- Resp.: Responsable  
- Desc.: Descripción

# G04. Roles y accesos

|  |  |  |
| --- | --- | --- |
| Nombre | Rol | Accesos |
| Lara Torres | Vendedor | RFN1 - Carga de reservas, validación técnica preliminar, registro de datos de evento y contacto, asignación de estadio inicial.  RFN2 – Carga de ventas presenciales. |
| Tomás Frías | Técnico de soporte | RFN1 – Soporte en verificación de rider técnico, revisión de documentación, asistencia en validación técnica.  RFN2 – Soporte ante fallas del sistema de emisión de tickets. |
| Mateo Gutiérrez | Desarrollador del sistema | RFN1 - Mantenimiento del backend de gestión de reservas.  RFN2 – Desarrollo del módulo de ventas y control de stock de tickets. |
| Iván Urso | Analista funcional | RFN1 - Diseño del flujo de reserva, validaciones de negocio, documentación de requisitos.  RFN2 – Especificación funcional de venta de entradas. |
| Dana Pérez | Project Manager | RFN1 – Supervisión de avances, revisión de funcionalidades implementadas.  RFN2 – Coordinación de los entregables del modulo de ventas. |

# G05. Otros requisitos

Del producto:  
- Basado en ISO/IEC 25010 (calidad de software).  
- Interfaz moderna, clara y adaptable.  
  
Requisitos del sistema:  
- Windows 10/11  
- .NET Framework actualizado  
- 4 GB RAM, 2 GHz CPU, 1 GB disco  
- SQL Server 2019+  
- Soporte para almacenamiento y BackUp  
  
Requisitos de desempeño:  
- Respuesta en tiempo real  
- Fluidez en operación y carga de eventos  
  
Entorno operativo:  
- Uso LAN para administración local  
- Integración con dispositivos de escaneo QR  
- Exclusivo para Windows escritorio  
  
Documentación:  
- Manual de usuario con ejemplos de uso  
- Guía de instalación paso a paso  
- Correo de soporte + número de atención

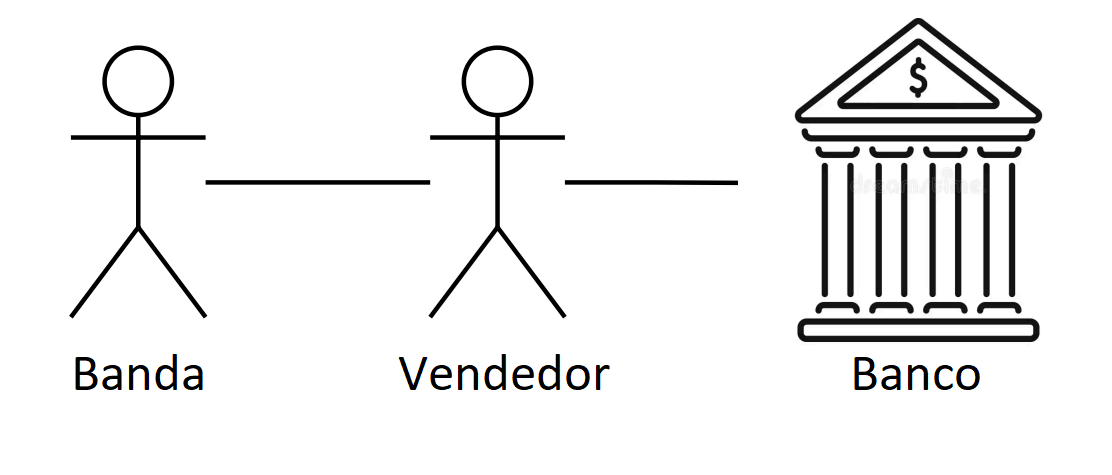
# N01

## N01.RF1 – Gestión de reservas de conciertos.

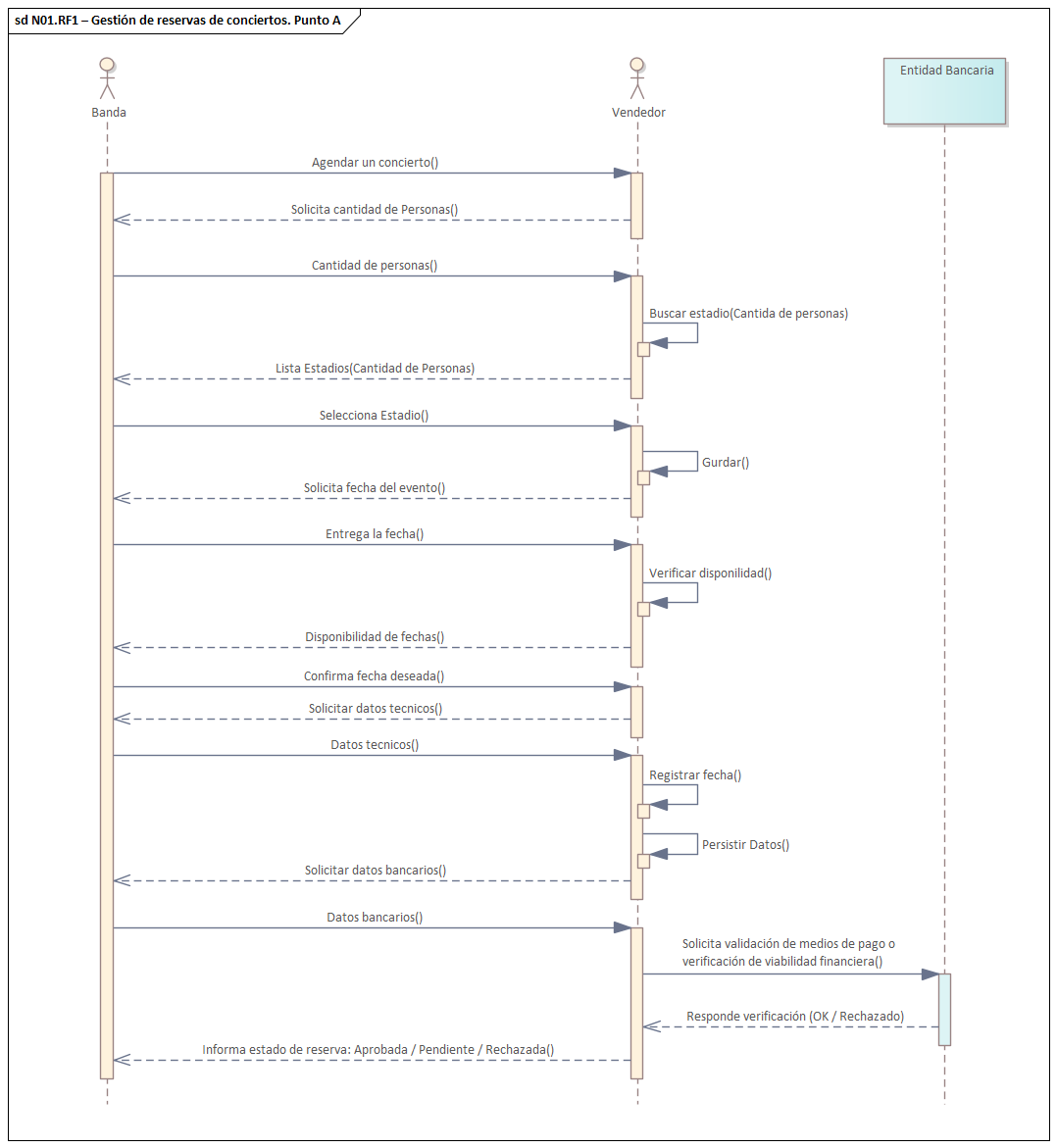
Punto A. Identificación de roles intervinientes.

* Banda (Persona – No es Actor Directo – No usa GUI)
* Empleado (Persona – Primario – Usa la GUI)
* Banco (Sistema – Se conecta con el sistema de ventas - Remoto)

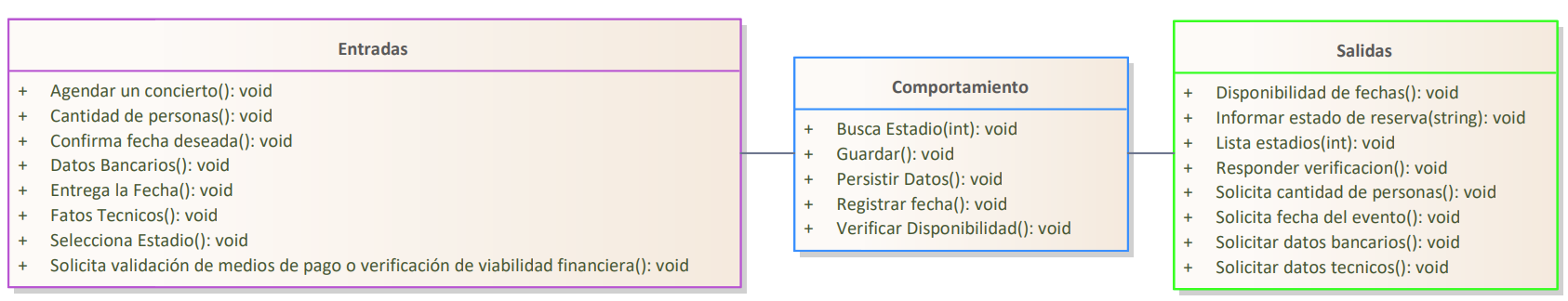
### Diagrama de roles:



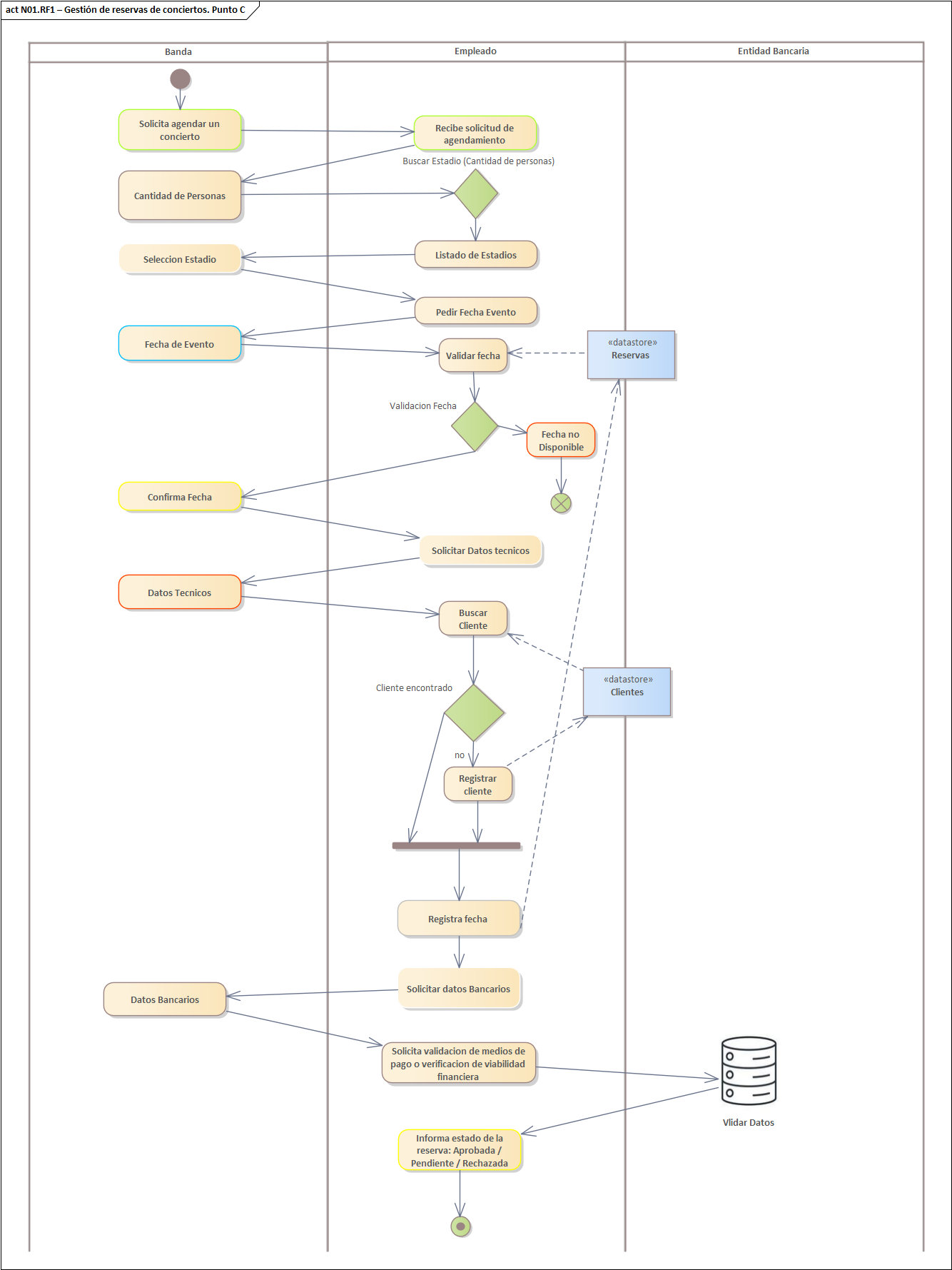
### Diagrama de secuencia de Roles:



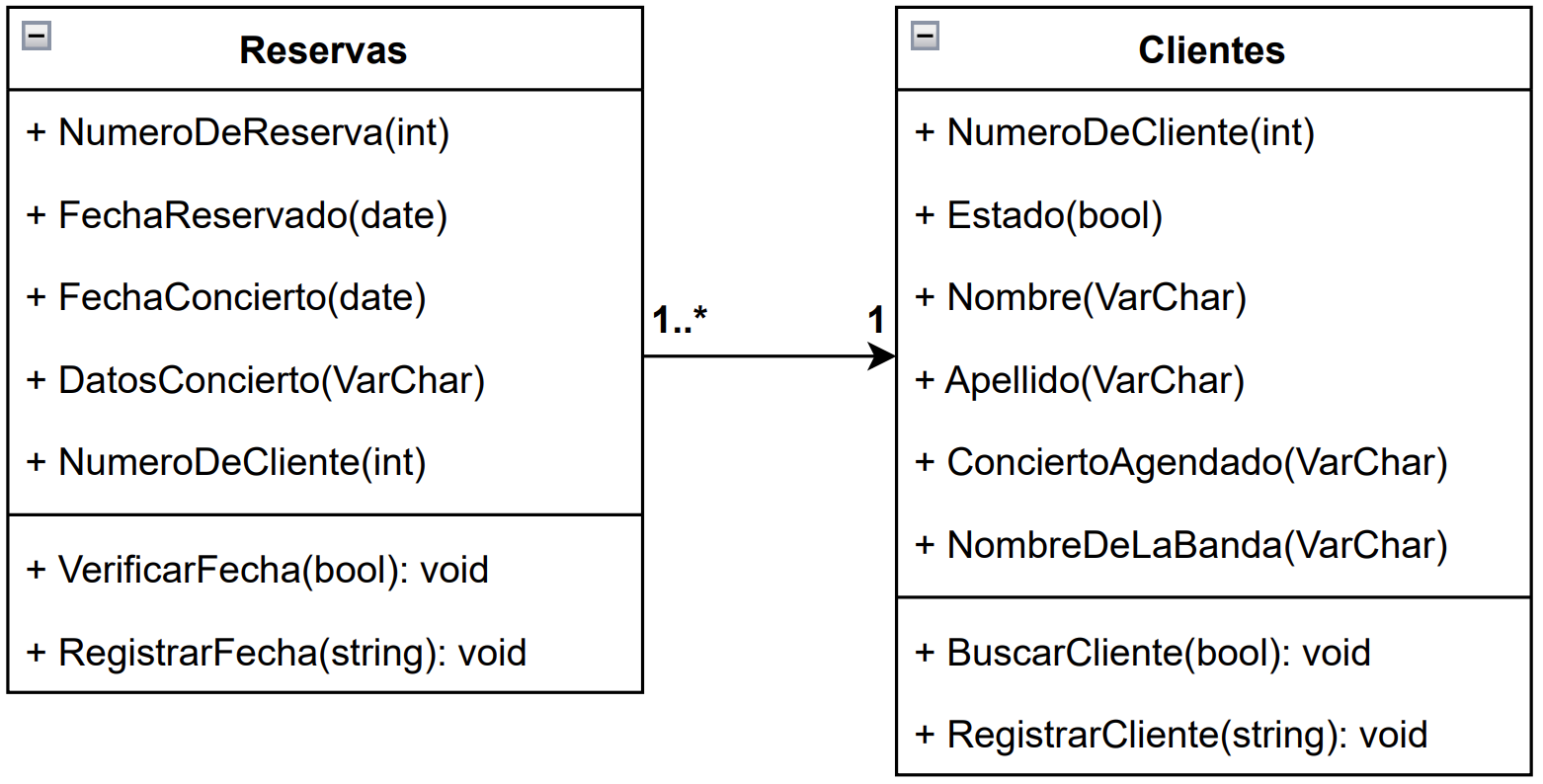
### Descripción Funcional del Proceso.



### Diagrama de proceso.



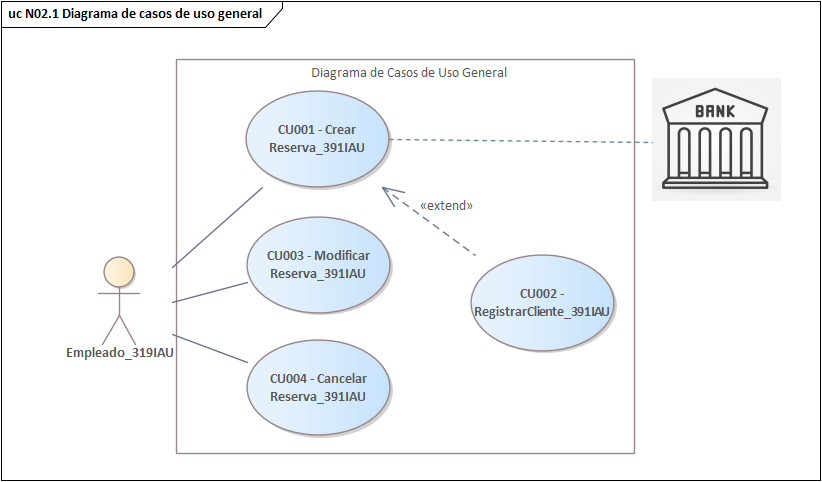
### Diagrama Conceptual.



# N02. Especificaciones de Casos de Uso.

Se procede a especificar con mayor detalle el pinto G02, documentando el requerimiento funcional RFN1 – “Gestión de reservas de Conciertos”

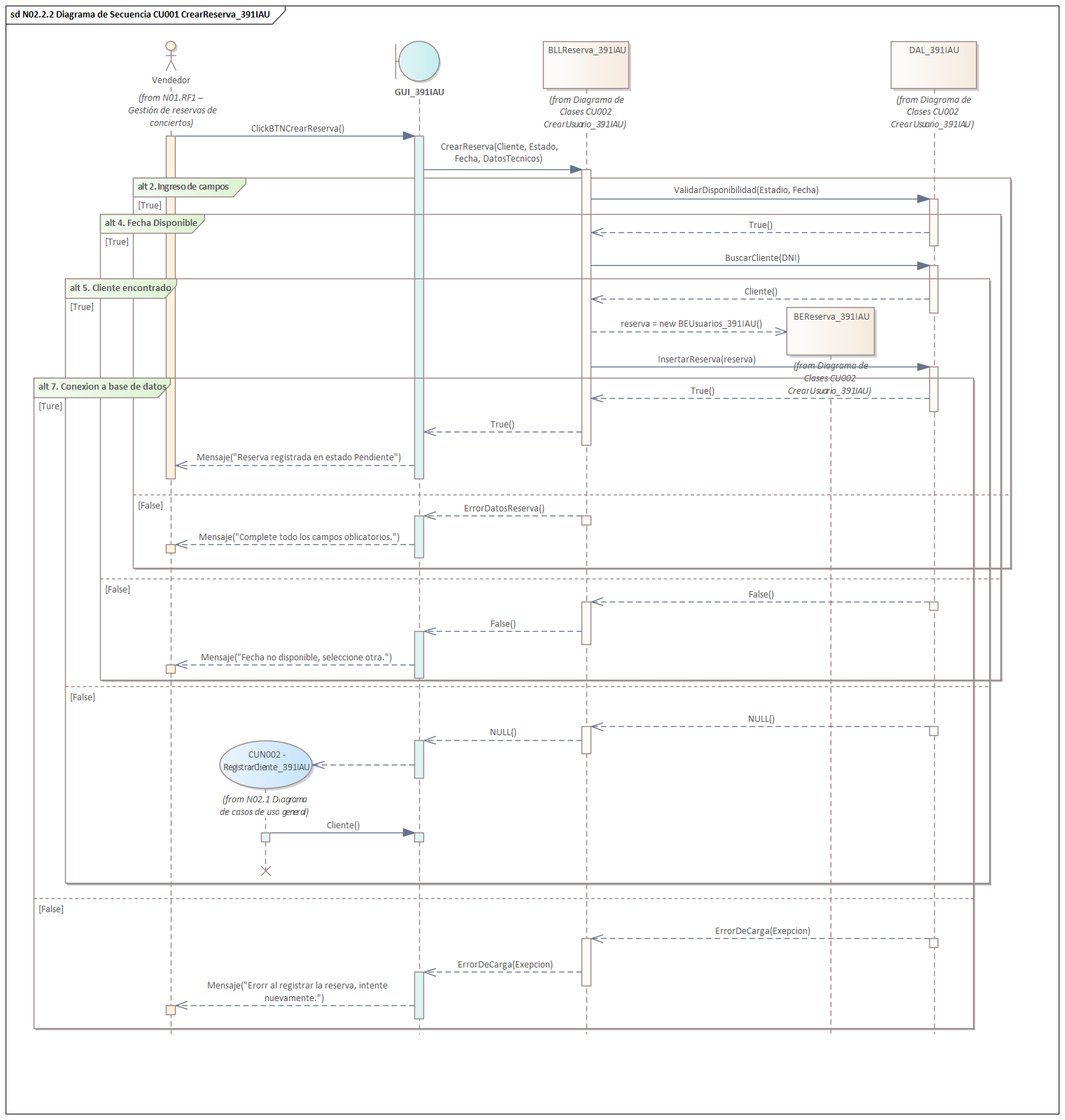
## N02.1 Diagrama de Casos de Uso General.



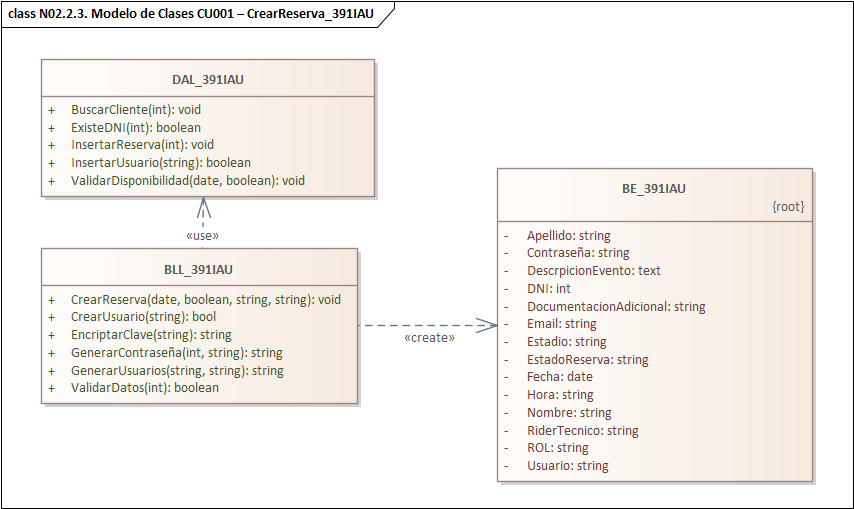
## N02.2.1 Especificación CU001 CrearReserva\_391IAU

|  |  |
| --- | --- |
| **ID y Nombre: CU001 – CrearReserva\_391IAU** | |
| **Objetivo:** Registrar una nueva solicitud de reserva para un concierto solicitado por un cliente, almacenando los datos del evento y documentación técnica, y dejándola lista para su validación. | |
| **Actor principal: Empleado\_391IAU** | **Actor secundario:** - |
| **Precondiciones:**   * El empelado debe haber iniciado sesión correctamente. | |
| **Punto de extensión:** CU002 – RegistrarCliente\_391IAU | **Punto de inclusión: -** |
| **Disparador:** El cliente solicita agendar un concierto mediante un canal de contacto (Teléfono, Mail, Local Físico) | |
| **Postcondiciones:**   * Se crea la reserva en estado “Pendiente”. * La documentación queda registrada en el sistema. * La reserva queda lista para su validación técnica y administrativa. | |
| **Escenario principal:**   1. El cliente solicita la reserva de una fecha. 2. El empleado recibe la solicitud e ingresa al sistema. 3. El empleado selecciona el estadio y fecha deseada. 4. El sistema valida que la fecha esté disponible. 5. El empleado ingresa los datos técnicos y adjunta la documentación. 6. El sistema registra la solicitud como “Pendiente”. 7. Se notifica al cliente del estado inicial de la resera. | |
| **Escenario alternativo:**  **2a.** Error de ingreso de datos en la solicitud.  **2b.** Ingresar los datos pedidos y obligatorios  **4a.** La fecha no está disponible.  **4b.** Se muestra un mensaje con una lista de fechas disponibles.  **4c.** El empleado selecciona una de las fechas de esa lista.  **5a.** El cliente no existe.  **5b.** CU002 – RegistrarCliente\_391IAU.  **7a.** Hay error de conexión.  **7b.** Se muestra “Error al registrar reserva, intente nuevamente”. | |

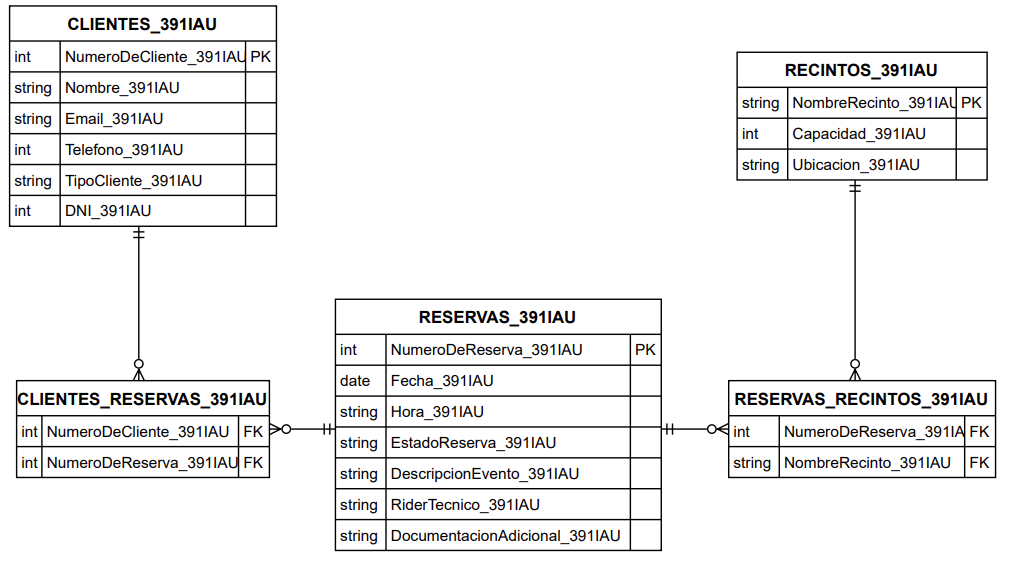
### N02.2.2 Diagrama de Secuencia CU001 CrearReserva\_391IAU



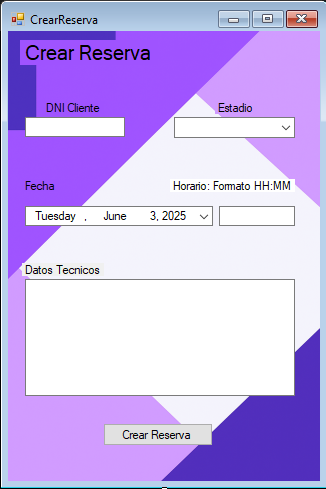
### N02.2.2. Modelo de clases C001 – CrearReserva\_391IAU



### N02.2.4. Modelo DER CU001 – CrearReserva\_391IAU



### N02.2.5. GUI - CU001 – CrearReserva\_391IAU

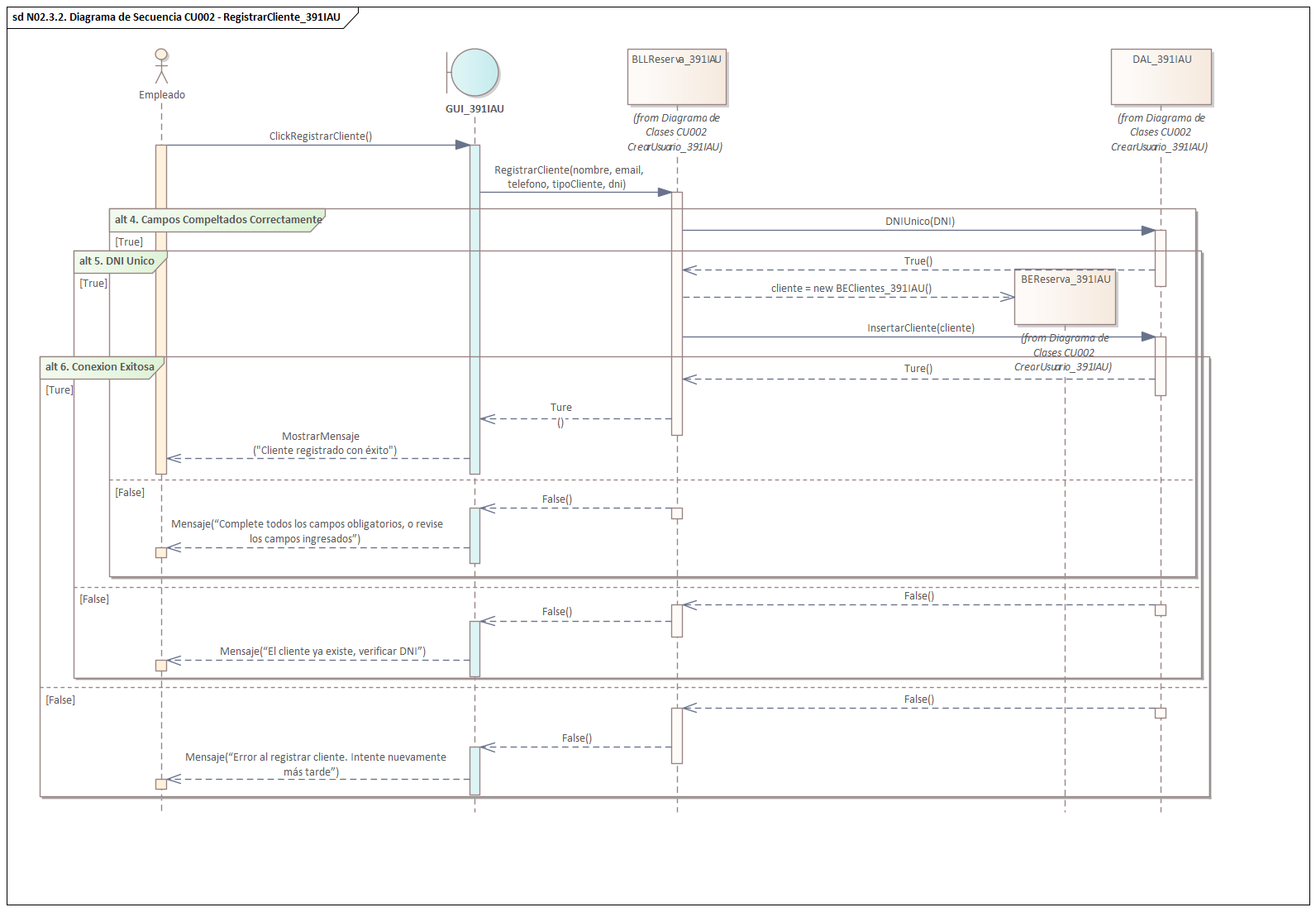


## N02.3. CU002 “RegistrarCliente\_391IAU”

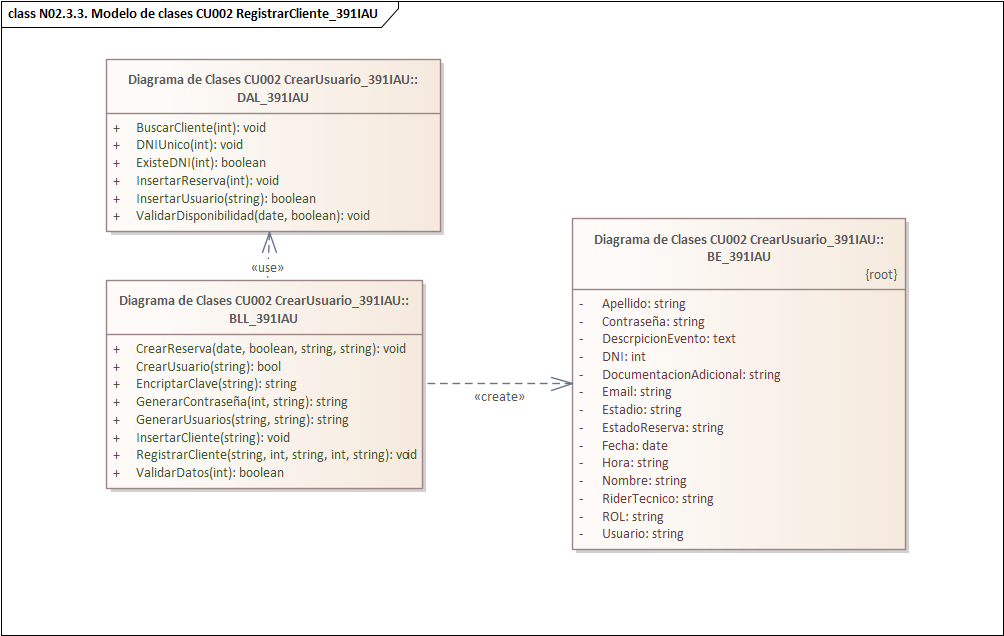
### N02.3.1. Especificación CU002 RegistrarCliente\_391IAU

|  |  |
| --- | --- |
| **ID y Nombre: CU002 RegistrarCliente\_391IAU** | |
| **Objetivo:** Permitir al empleado registrar un nuevo cliente en el sistema, almacenando sus datos de contacto y clasificación antes de completar la reserva. | |
| **Actor principal:** Empleado\_391IAU | **Actor secundario:** - |
| **Precondiciones:**   * El empleado debe haber iniciado sesión correctamente. * No debe existir un cliente registrado con el mismo DNI. | |
| **Punto de extensión: -** | **Punto de inclusión:** CU001 – CrearReserva\_391IAU |
| **Disparador:** Durante el alta de una reserva, se detecta que el cliente no se encuentra registrado en el sistema. | |
| **Postcondiciones:**   * El nuevo cliente queda registrado y disponible para asociarse a reservas. * El sistema asocia la reserva pendiente al nuevo cliente. | |
| **Escenario principal:**   1. El empleado detecta que el cliente no se encuentra registrado. 2. El sistema solicita los datos del cliente. 3. El empleado completa los campos requeridos (Nombre, Email, Teléfono, Tipo de Cliente). 4. El sistema valida que los datos estén completos y en el formato correcto. 5. El sistema valida que el cliente no exista por medio el DNI. 6. El sistema registra al cliente en la base de datos. 7. Se notifica al empleado que el cliente fuer registrado correctamente. 8. El sistema retorna al flujo de alta de reserva. | |
| **Escenario alternativo:**  **4a.** Falta algún campo obligatorio.  **4b.** El sistema muestra el mensaje“Complete todos los campos obligatorios, o revise los campos ingresados”.  **5a.** El DNI ya existe en la base de datos.  **5b.** El sistema muestra el mensaje “El cliente ya existe, verificar DNI”  **6a.** Error de conexión.  **6b.** El sistema muestra el mensaje “Error al registrar cliente. Intente nuevamente más tarde”. | |

### N02.3.2. Diagrama de secuencia CU002 RegistrarCliente\_391IAU



### N02.3.3. Modelo de clases CU002 RegistrarCliente\_391IAU



### N02.3.4. Modelo DER CU002 RegistrarCliente\_391IAU



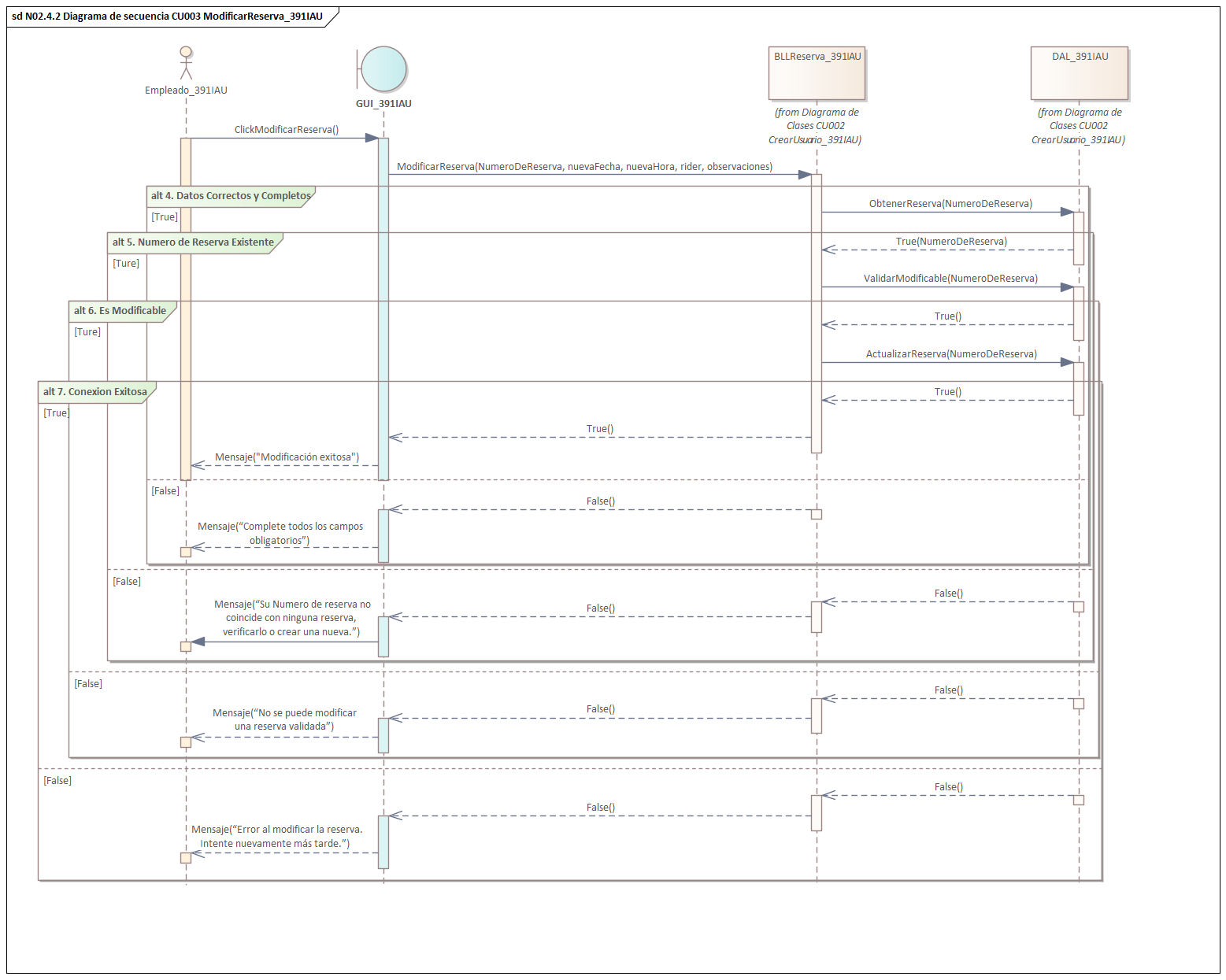
### N02.3.5. GUI - CU002– RegistrarCliente\_391IAU



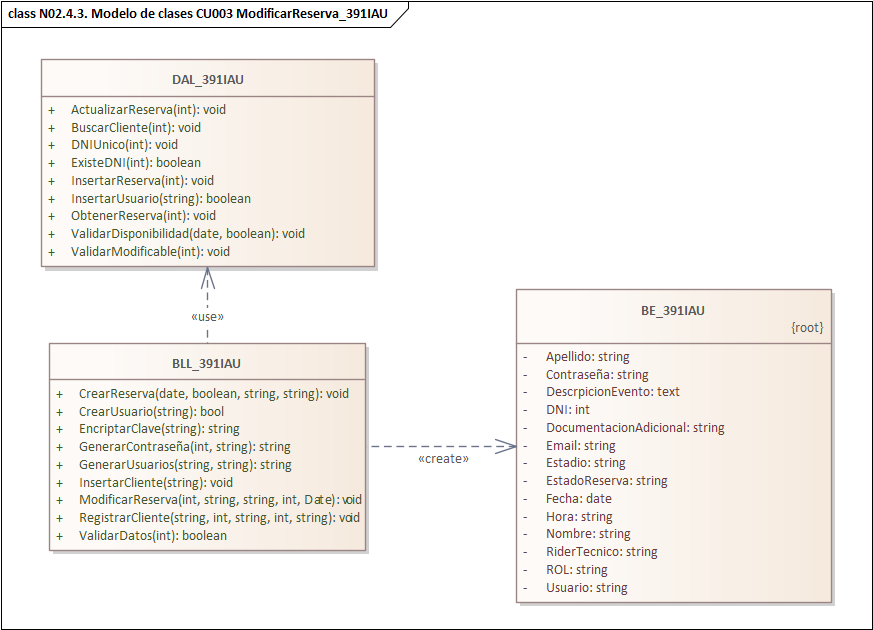
### N02.4.1 Especificación CU003 ModificarReserva\_391IAU

|  |  |
| --- | --- |
| **ID y Nombre: CU003 ModificarReserva\_391IAU** | |
| **Objetivo:** Permitir al empleado modificar una reserva existente que aún no fue validada, cambiando sus datos técnicos fecha u observaciones si es necesario. | |
| **Actor principal:** Empleado\_391IAU | **Actor secundario:** - |
| **Precondiciones:**   * El empleado debe haber iniciado sesión correctamente. * La reserva debe estar en estado “Pendiente”. | |
| **Punto de extensión: -** | **Punto de inclusión: -** |
| **Disparador:** El empleado selecciona una reserva en estado “Pendiente” y presiona la opción “Modificar”. | |
| **Postcondiciones:**   * Los datos de la reserva quedan actualizados en la base de datos. * La reserva conserva el estado “Pendiente”. | |
| **Escenario principal:**   1. El empelado accede al módulo de reservas y selecciona una en estado “Pendiente”. 2. El sistema carga los datos actuales de la reserva. 3. El empleado modifica uno o más campos campos permitidos (Fecha, hora, rider técnico, observaciones). 4. El sistema valida que los nuevos datos estén completos y en el formato correcto. 5. El sistema valida que la reserva exista mediante el NumeroDeReserva. 6. El sistema validad que la reserva no haya sido validada. 7. El sistema actualiza la reserva en la base de datos. 8. El sistema informa que la modificación fue exitosa | |
| **Escenario alternativo:**  **4a.** Falta completar campos obligatorios.  **4b.** El sistema muestra el mensaje “Complete todos los campos obligatorios”.  **5a.** El NumeroDeReserva no coincide con ninguna reserva registrada.  **5b.** El sistema muestra el mensaje “Su Número de reserva no coincide con ninguna reserva, verificarlo o crear una nueva.”.  **6a.** La reserva ya fue validada.  **6b.** El sistema muestra el mensaje “No se puede modificar una reserva validada”.  **7a.** Error de conexión o actualización.  **7b.** El sistema muestra el mensaje “Error al modificar la reserva. Intente nuevamente más tarde.”. | |

### N02.4.2 Diagrama de secuencia CU003 ModificarReserva\_391IAU



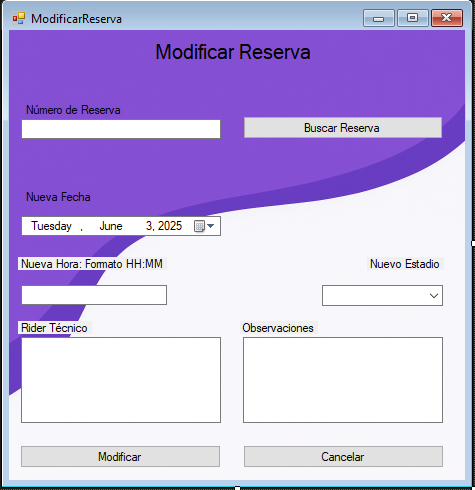
### N02.4.3. Modelo de clases CU003 ModificarReserva\_391IAU



### N02.4.4. Modelo de clases CU003 ModificarReserva\_391IAU

### 

### N02.4.5. GUI - CU003 – ModificarReserva\_391IAU

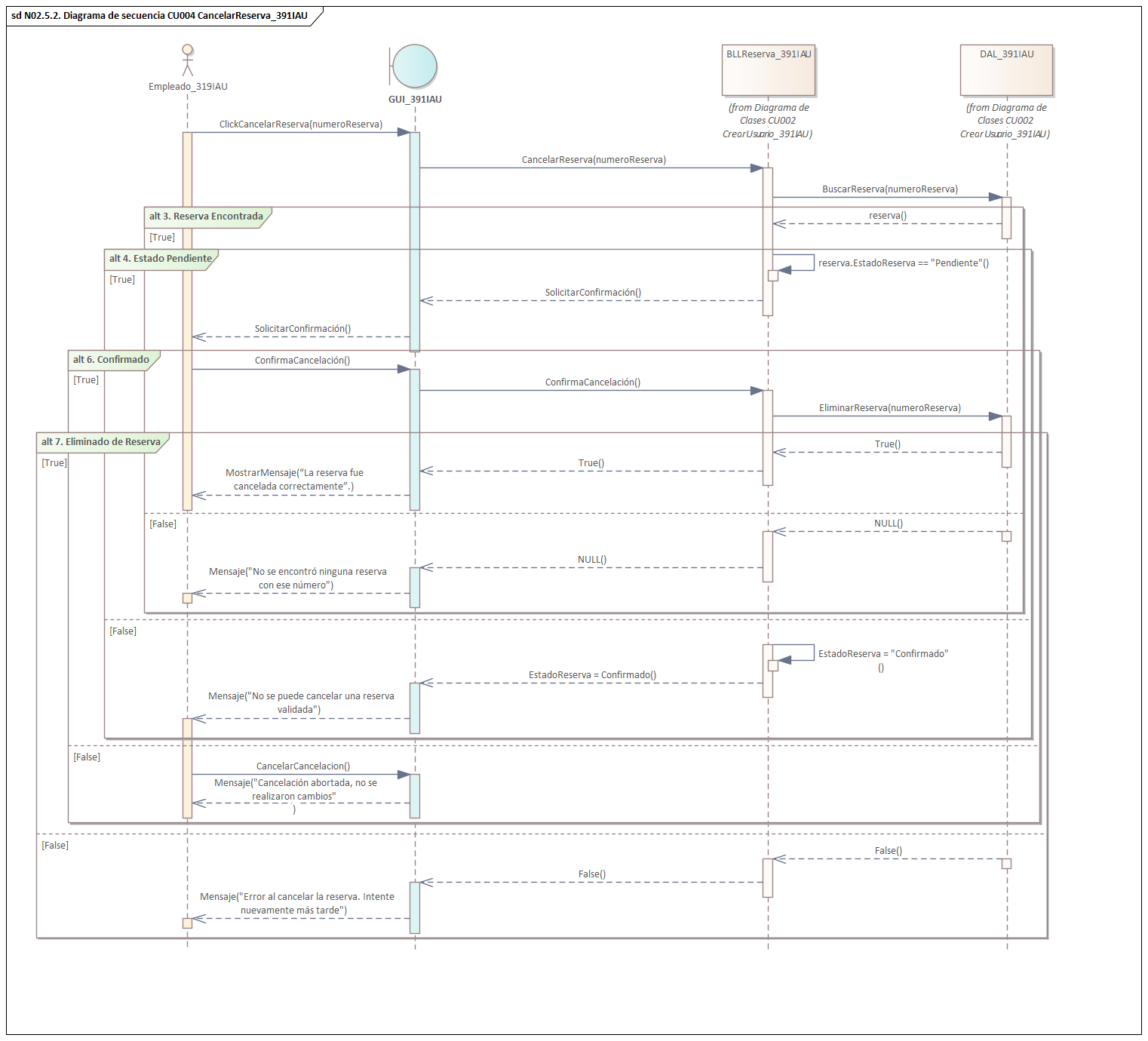


## N02.5. CU004 “CancelarReserva\_391IAU”

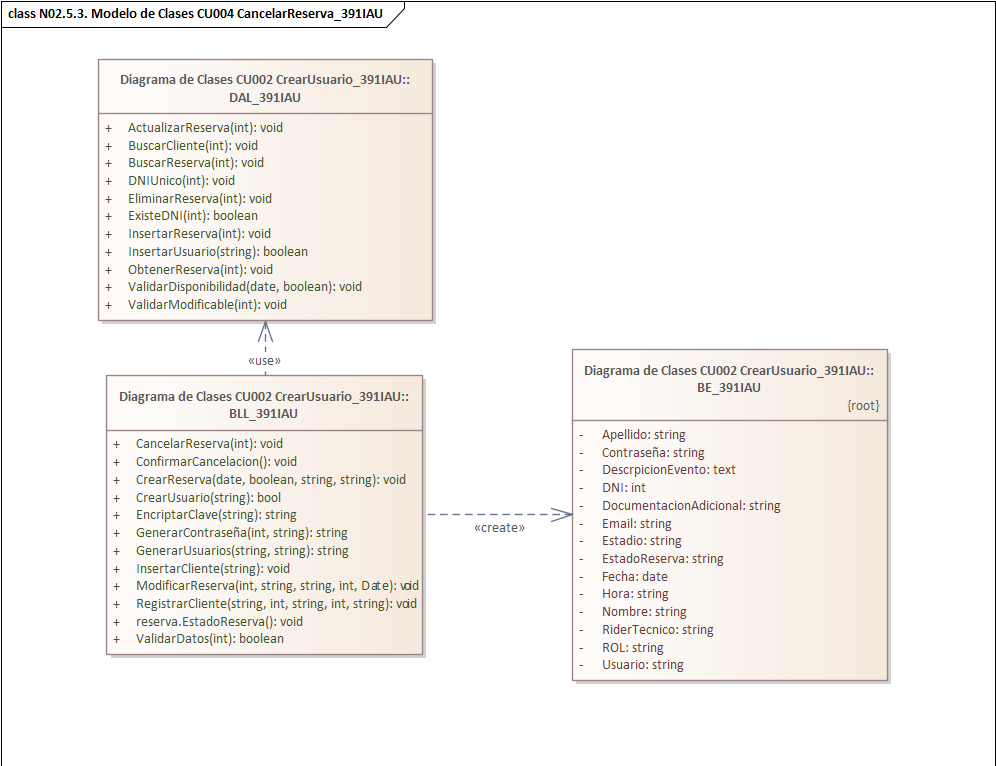
### N02.5.1. Especificación CU004 CancelarReserva\_391IAU

|  |  |
| --- | --- |
| **ID y Nombre: CU004 CancelarReserva\_391IAU** | |
| **Objetivo:** Permitir al empleado cancelar una reserva pendiente de validación, eliminándola del sistema antes de su confirmación. | |
| **Actor principal:** Empleado\_391IAU | **Actor secundario:** - |
| **Precondiciones:**   * El empleado debe haber iniciado sesión correctamente. * La reserva debe encontrarse en estado “Pendiente”. | |
| **Punto de extensión: -** | **Punto de inclusión: -** |
| **Disparador:** EL empleado selecciona una reserva pendiente y presiona la opción “Cancelar”. | |
| **Postcondiciones:**   * Lareserva es eliminada del sistema. * Se libera la disponibilidad del recinto reservado. | |
| **Escenario principal:**   1. El empleado acede al módulo de gestión de reservas. 2. Selecciona una reserva en estad “Pendiente”. 3. El sistema valida que la reserva existe. 4. El sistema verifica que no haya sido validada. 5. El sistema solicita confirmación para cancelar. 6. El empleado confirma la cancelación. 7. EL sistema elimina la reserva de la base de datos. 8. Se muestra el mensaje “La reserva fue cancelada correctamente”. | |
| **Escenario alternativo:**  **3a.** La reserva no existe.  **3b.** El sistema muestra el mensaje “No se encontró ninguna reserva con ese número.”.  **4a.** La reserva fue validada.  **4b.** El sistema muestra el mensaje “No se puede cancelar una reserva validada.”.  **6a.** El empleado cancela la acción.  **6b.** El sistema regresa al módulo de reservas sin efectuar cambios.  **7a.** Error de conexión o fallo en el borrado.  **7b.** El sistemamuestra el mensaje “Error al cancelar la reserva. Intente nuevamente más tarde.”. | |

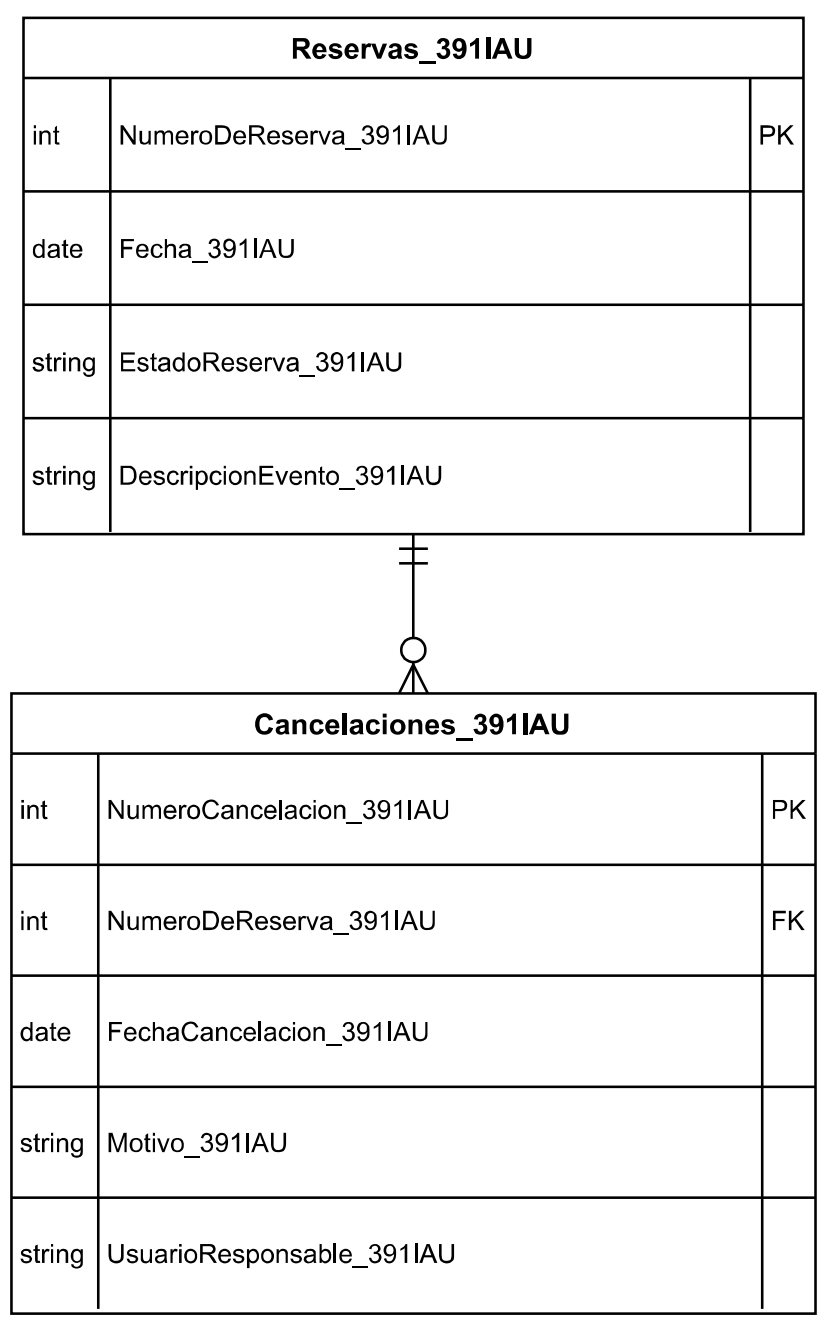
### N02.5.2. Diagrama de secuencia CU004 CancelarReserva\_391IAU



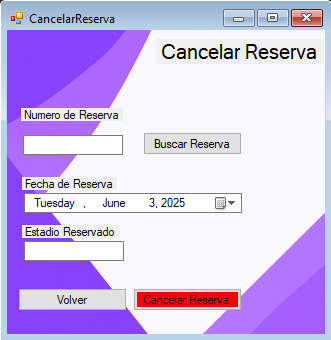
### N02.5.3. Modelo de Clases CU004 CancelarReserva\_391IAU



### N02.5.4. Modelo DER CU004 CancelacionReserva\_391IAU

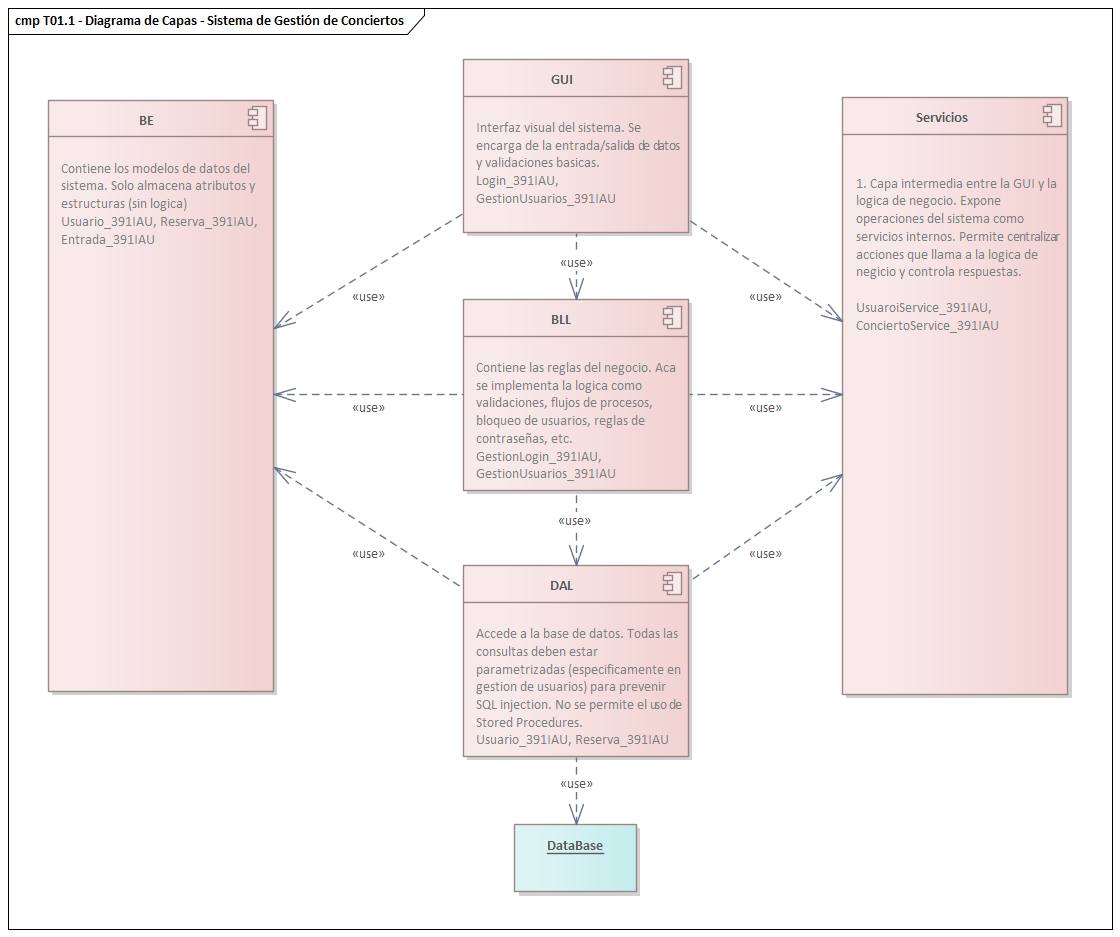


### N02.5.5. GUI – CU004 CancelacionReserva\_391IAU

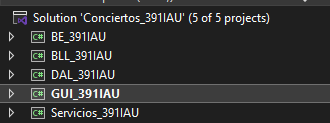


# T01

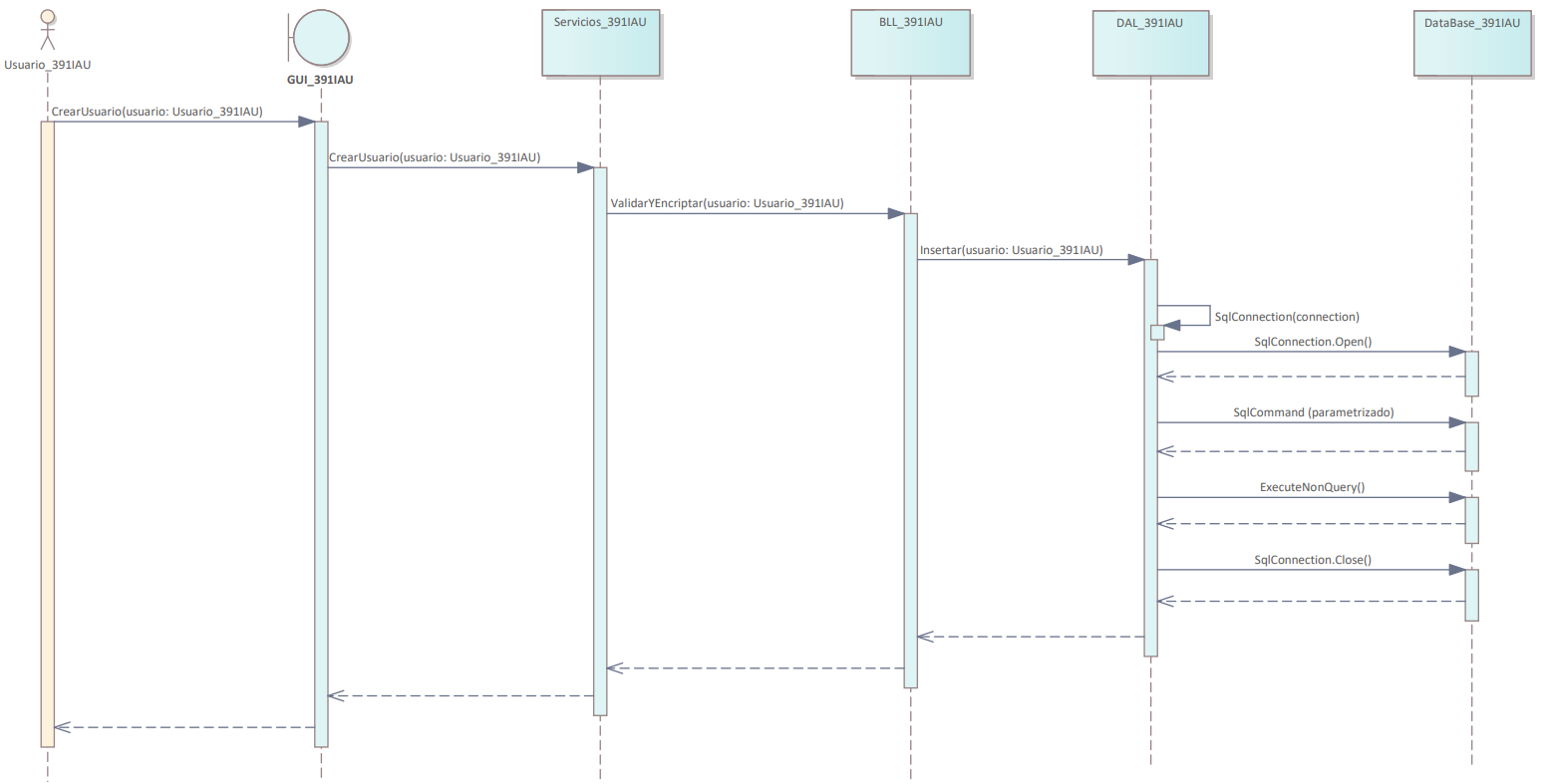
## T01.1 Diagrama de Capas



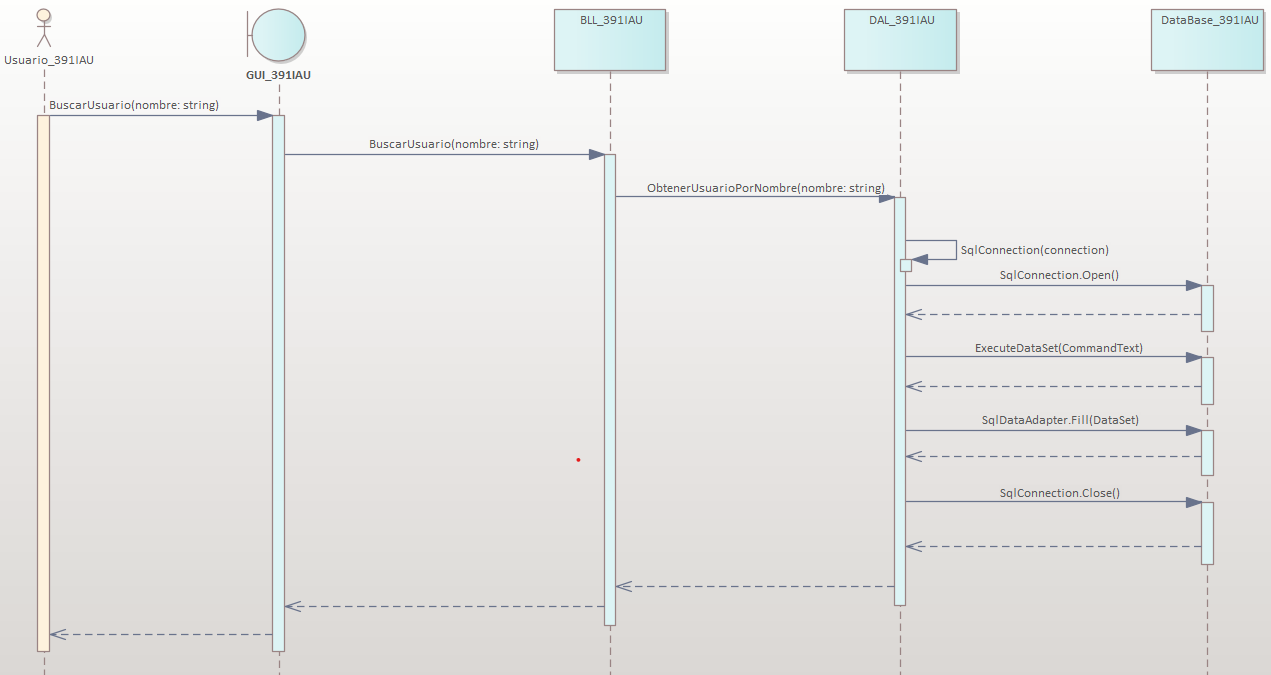
## T01.2 Captura de Pantalla de Capas creadas en el IDE



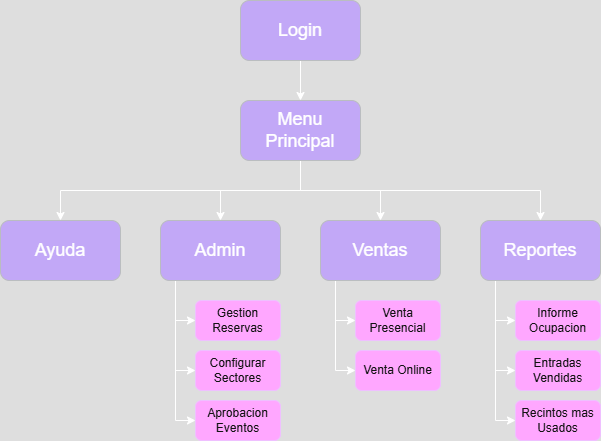
## T01.3 Diagrama de Secuencia de Persistencia en la Base de Datos



## T01.4 Diagrama de Secuencia de Consulta en la Base de Datos



## T01.5 Diagrama de Secuencia de Consulta en la Base de Datos



# T02 Gestión de Usuarios

**T02.1.1 – Descomposición funcional CU001 – CrearUsuario\_391IAU**

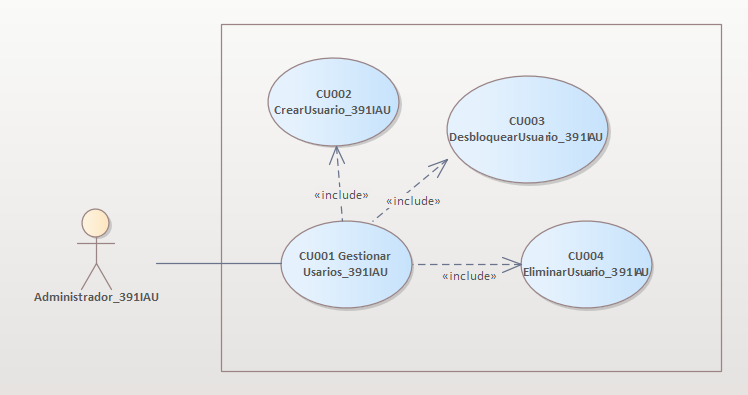
Escenario Funcional Paso a paso

1. El administrador\_391IAU inicia sesión y accede al menú principal.
2. Dentro del menú, se dirige al módulo “Administración” y selecciona la opción “Gestionar Usuarios”.
3. El sistema muestra la pantalla de gestión de usuarios, que contiene una grilla con los usuarios ya registrados y una barra de botones (Crear, Modificar, Desbloquear, Eliminar).
4. El administrador selecciona el botón “Crear”.
5. El sistema habilita los campos para la carga:
   1. DNI
   2. Nombre
   3. Apellido
   4. Email
   5. Rol
6. El administrador completa los campos requeridos y presiona el botón “Aplicar”.
7. El sistema valida que:
   1. Ningún acampo obligatorio este vacío.
   2. El DNI no este repetido en la Base de Datos.
   3. El email tenga formato valido.
   4. El rol sea uno de los definidos en el sistema (Admin, Técnico, Promotor, ETC).
8. Si todo es válido:
   1. El sistema genera automáticamente el nombre de usuario (Nombre.Apellido) y la contraseña (DNI+Apellido).
   2. Encripta la contraseña con SHA-256.
   3. Inserta el nuevo registro en la base de datos.
9. El sistema muestra un mensaje de confirmación:
   1. “El usuario fue creado correctamente”.
10. La grilla se actualiza y se visualiza el nuevo usuario.

Lista de atributos Involucrados:

* DNI
* NOMBRE
* APELLIDO
* EMAIL
* ROL
* USUARIO (Autogenerado)
* CONTRASEÑA (Autogenerado + encriptado SHA-256)

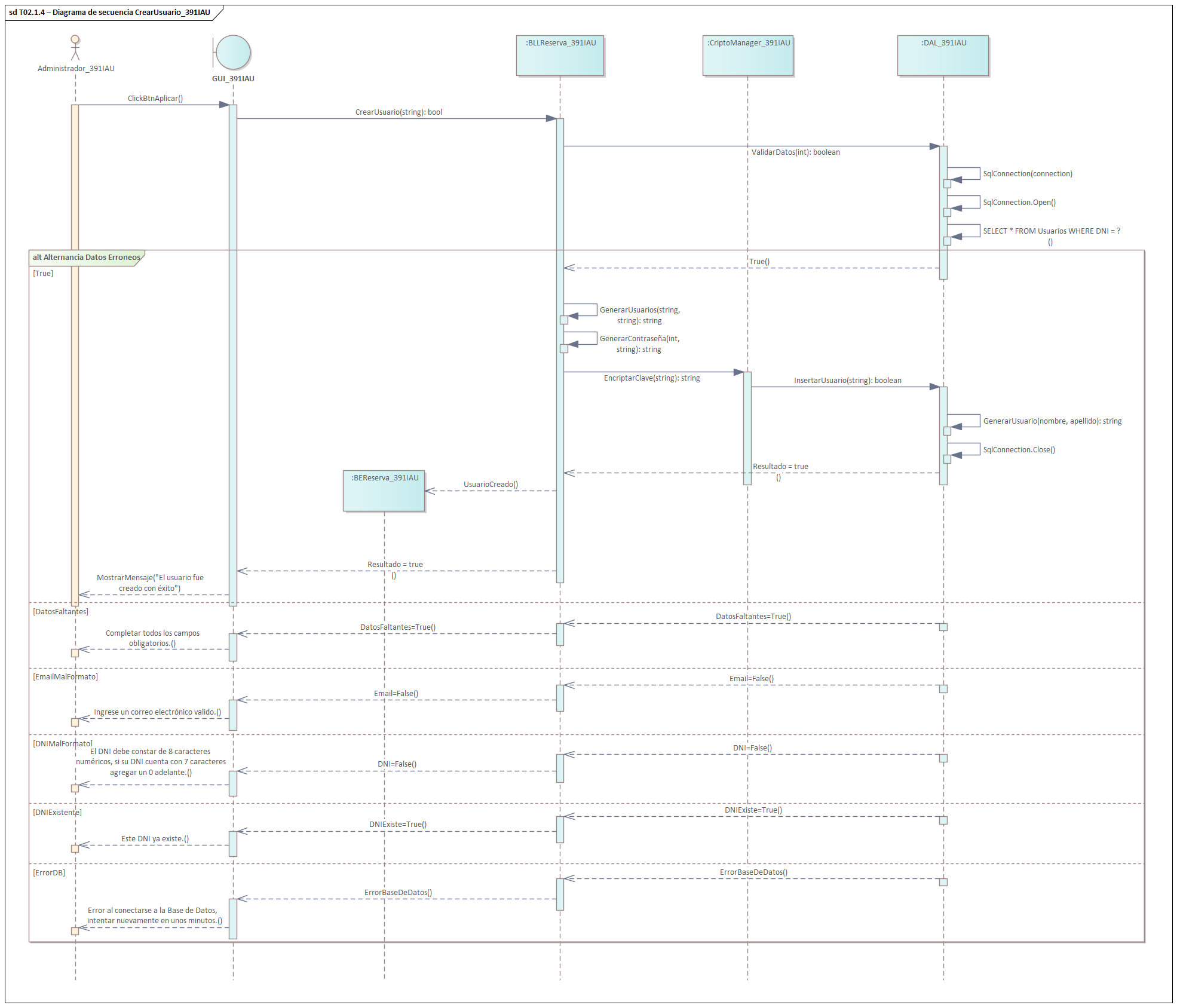
## T02.1.2 Diagrama de Casos de Uso CU001 – CrearUsuario\_391IAU



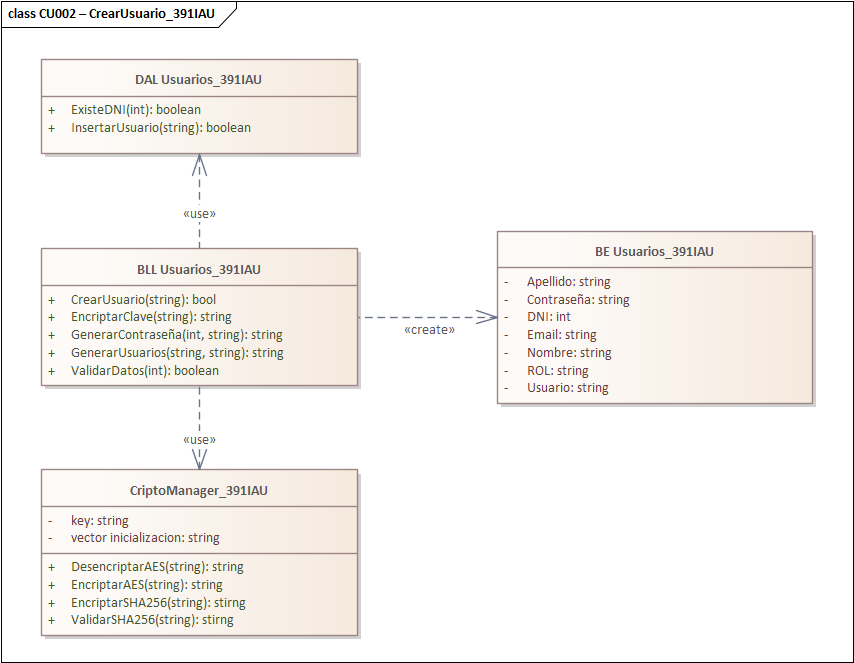
## T02.1.3 – Especificación del Caso de Uso: CU002 CrearUsuario\_391IAU

|  |
| --- |
| **ID y Nombre:** CU002 – CrearUsuario\_391IAU |
| **Objetivo:** Permitir al administrador crear un nuevo usuario en el sistema, garantizando que los datos sean válidos, únicos y que la contraseña sea encriptada con SHA-256 antes de almacenarse. |
| **Actor principal: Administrador\_391IAU** |
| **Precondiciones:**   * El Administrador debe haber iniciado sesión correctamente. * Debe estar posicionado en el módulo “Gestión de Usuarios” |
| **Punto de extensión:** |
| **Disparador:** El administrador selecciona la opción “Crear Usuario” desde el menú de gestión de usuarios. |
| **Postcondiciones:**   * El usuario se registró en el sistema y es visualizado en la grilla. * La contraseña queda almacenada y encriptada (SHA-256) * Se muestra mensaje de confirmación “El usuario fue creado con éxito” |
| **Escenario principal:**   1. **El administrador selecciona “Crear Usuario”.** 2. **El sistema muestra el formulario vacío.** 3. **El administrador ingresa los campos: Nombre, Apellido, DNI, Email, ROL.** 4. **El sistema valida:**  * **Que los campos obligatorios se encuentren completos.** * **Que el formato del mail sea correcto.** * **Que el DNI sea formato numérico INT de 8 caracteres.** * **Que el usuario no exista en la base de datos, se fija que el DNI sea único en la base.**  1. **El sistema genera:**  * **Usuario = Nombre.Apellido** * **Contraseña = DNI + Apellido**  1. **El sistema encripta la contraseña con SHA-256** 2. **Se guarda el nuevo usuario en la base de datos.** 3. **Se actualiza la grilla y se muestra mensaje de éxito.** |
| **Escenario alternativo:**  **4.1. Faltan campos obligatorios.**  **4.2. El sistema muestra el mensaje “Completar todos los campos obligatorios.”.**  **4.3. Email con formato invalido.**  **4.4. El sistema notifica “Ingrese un correo electrónico valido.”.**  **4.5. DNI formato incorrecto.**  **4.6. El sistema muestra el mensaje “El DNI debe constar de 8 caracteres numéricos, si su DNI cuenta con 7 caracteres agregar un 0 adelante.”**  **4.7. El usuario ya existe.**  **4.8. El sistema muestra el mensaje “Este DNI ya existe.”.**  **4.9. Error de conexión o falla en la BD.**  **4.10. Se muestra un mensaje que dice “Error al conectarse a la Base de Datos, intentar nuevamente en unos minutos.”.** |
| **Puntos de extensión:**   * **CU001 GestionarUsuarios\_391IAU** |

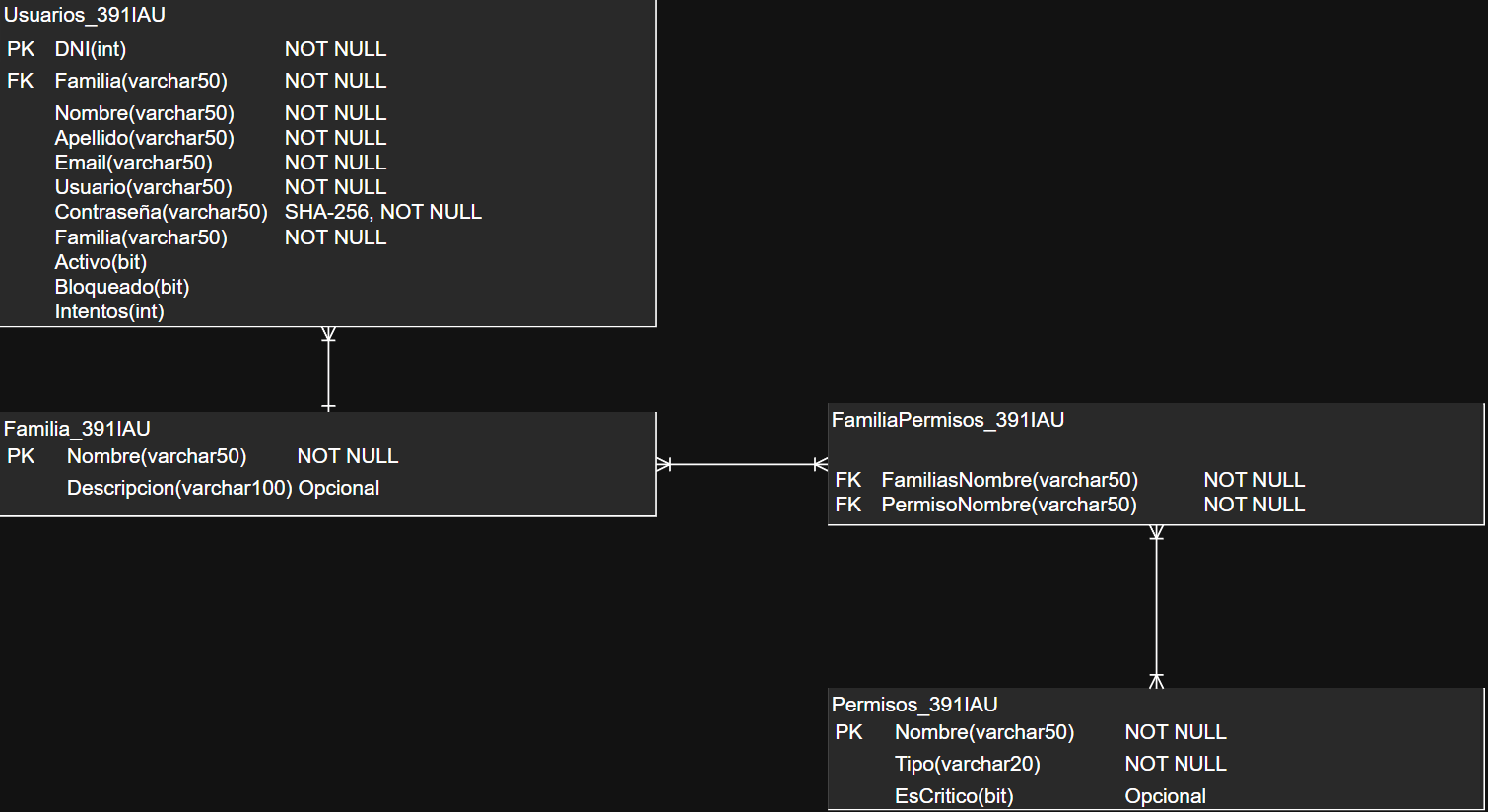
## T02.1.4 Diagrama de Secuencia CU002 CrearUsuario\_391IAU



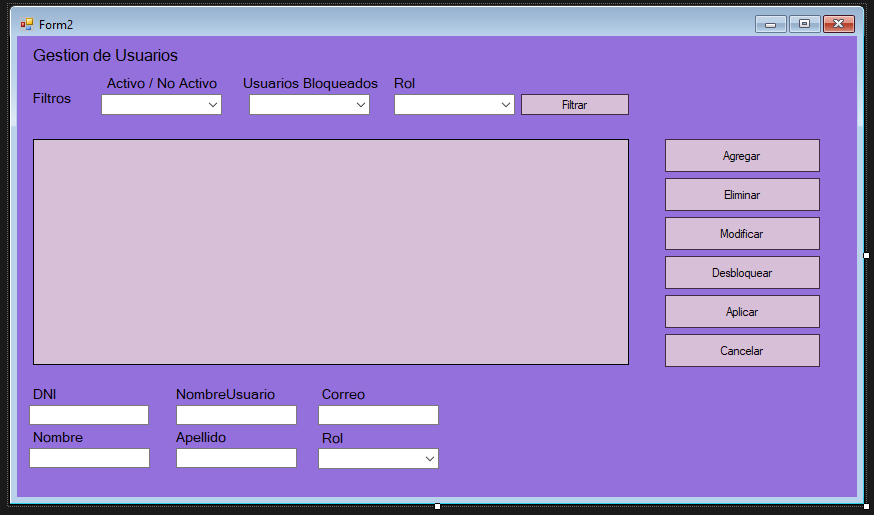
## T02.1.5 Diagrama de Clases CU002 CrearUsuario\_391IAU



## T02.1.6 Diagrama DER CU002 CrearUsuario\_391IAU



## T02.1.7 GUI CU002 CrearUsuario\_391IAU



T02.2.1