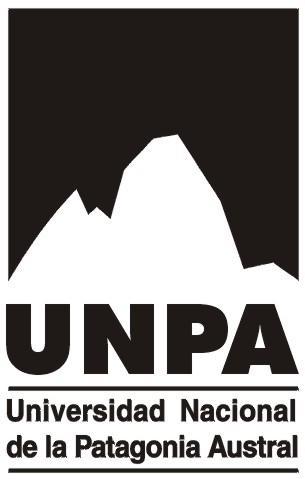


Modelo de Diseño

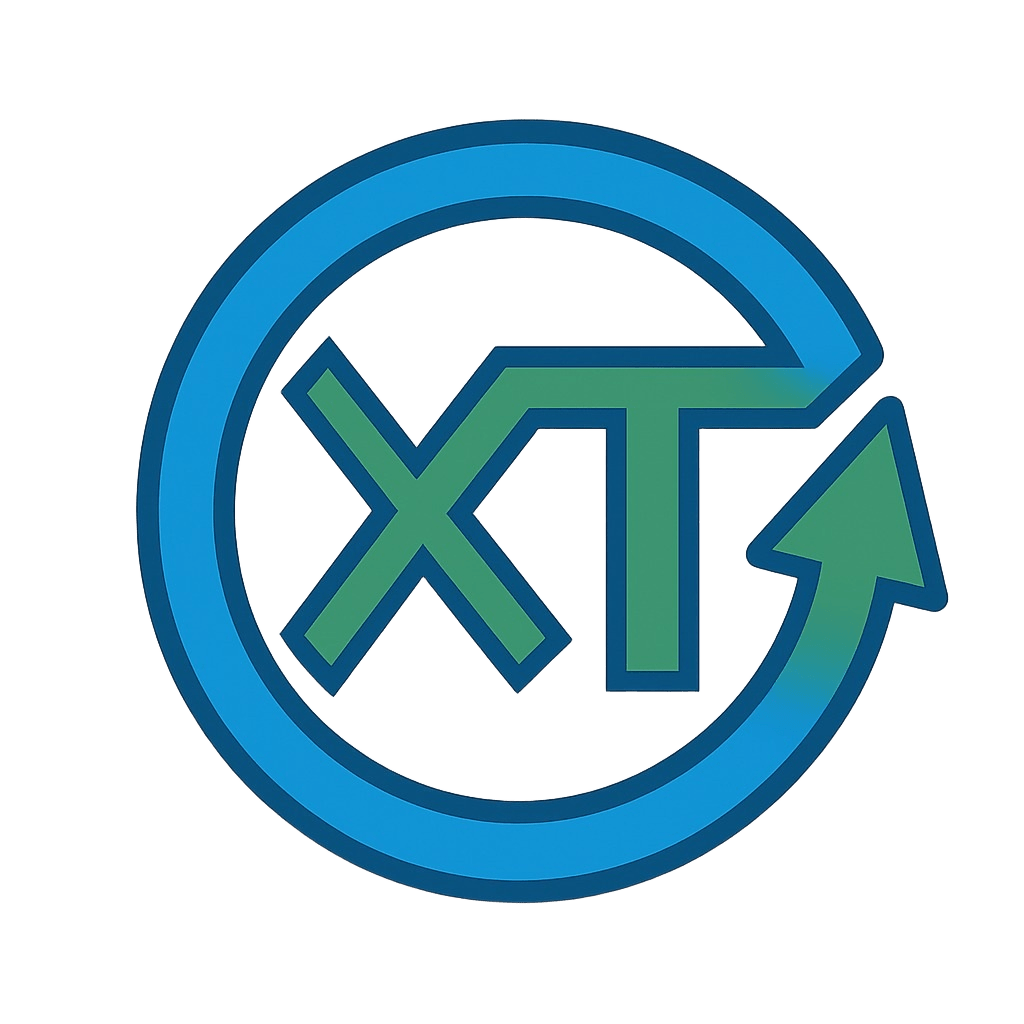
Kairos

NexTech

Centurión Valeria, Escalante Guillermo, Maldonado Agustina, Mendez Florencia, Ulloa Gonzalo.

**

**



**Tabla de contenido**

[**Introducción 4**](#_heading=h.3u5nqdqsvyb)

[*Propósito 4*](#_heading=h.yjzs0boni4oi)

[*Alcance 4*](#_heading=h.tm643w4f2p7b)

[*Referencias 4*](#_heading=h.i8vp42tneczd)

[*Visión general 5*](#_heading=h.suh40ihe4czm)

[**Diseño de Casos de Uso 5**](#_heading=h.ge39bntkxxur)

[*Caso de Uso CU 01: Iniciar Sesión 5*](#_heading=h.iwnjkdy4yune)

[*Especificación de caso de uso de diseño 5*](#_heading=h.qj6rcdk71ufp)

[*Caso de Uso CU 02: Exportar Información 7*](#_heading=h.4nj6hhb1uuiz)

[*Especificación de caso de uso de diseño 7*](#_heading=h.sfrv74y59shd)

[*Caso de Uso CU 05: Crear Proyecto 10*](#_heading=h.8nefth3w9f51)

[*Especificación de caso de uso de diseño 10*](#_heading=h.8hlp4lyzh7ug)

[*Caso de Uso 20: Registro de tiempo por cronómetro 12*](#_heading=h.ba57zouvno8x)

[*Especificación de caso de uso de diseño 12*](#_heading=h.ucrph4lbx3tb)

[*Diagrama de Secuencia 14*](#_heading=h.cn690lenm6wb)

[*Caso de Uso 21: Registro de tiempo manual 15*](#_heading=h.3aw9xpgfw2r3)

[*Especificación de caso de uso de diseño 15*](#_heading=h.v78bz0cdypkk)

[*Diagrama de Secuencia 16*](#_heading=h.vp4xaxctqzuf)

[*Diagrama de componentes 17*](#_heading=h.dqc39p9uv7js)

[*Diagrama de clases 18*](#_heading=h.hvjv02f347ga)

Modelo de Diseño

# Introducción

En este documento se detalla el modelo de diseño del sistema Kairos. Su propósito es transformar los resultados del análisis en una especificación técnica que oriente la implementación del software, definiendo la arquitectura, los principales componentes y el diseño de los casos de uso prioritarios.

## Propósito

El modelo de diseño tiene como objetivo detallar la estructura del sistema desde una perspectiva técnica, definiendo las clases, paquetes, interacciones y flujos que permitirán materializar los requerimientos funcionales identificados en el análisis. Este documento representará una base sólida para la implementación del software, incluyendo su codificación y realización de pruebas.

## Alcance

El modelo de diseño abarca la totalidad de los casos de uso que serán implementados en el sistema Kairos. Además, especifica los componentes estructurales del sistema, la forma en que estos interactúan entre sí y cómo se modularizan. Asimismo, incluye la gestión de usuarios, los escenarios de prueba, la estructura de la base de datos y las integraciones con sistemas externos, ofreciendo una visión clara de la interacción entre el back-end y el front-end.

## Referencias

En este documento se hace referencia a los siguientes documentos desarrollados por el equipo NexTech:

* Modelo de Datos.
* Especificación de Casos de Uso.
* Modelo de Casos de Uso.

## Visión general

Cada caso de uso presentado a partir de este punto, contará con su diseño; diagrama de paquetes, diagrama de interacción, diseño de flujo de eventos, y se especificará aquel o aquellos requerimiento/s no funcional/es que estén relacionados, si es que posee.

Por último, se incluirá al final del documento los diagramas de clases, diagramas de secuencia, diagramas de paquetes y diagrama de colaboración.

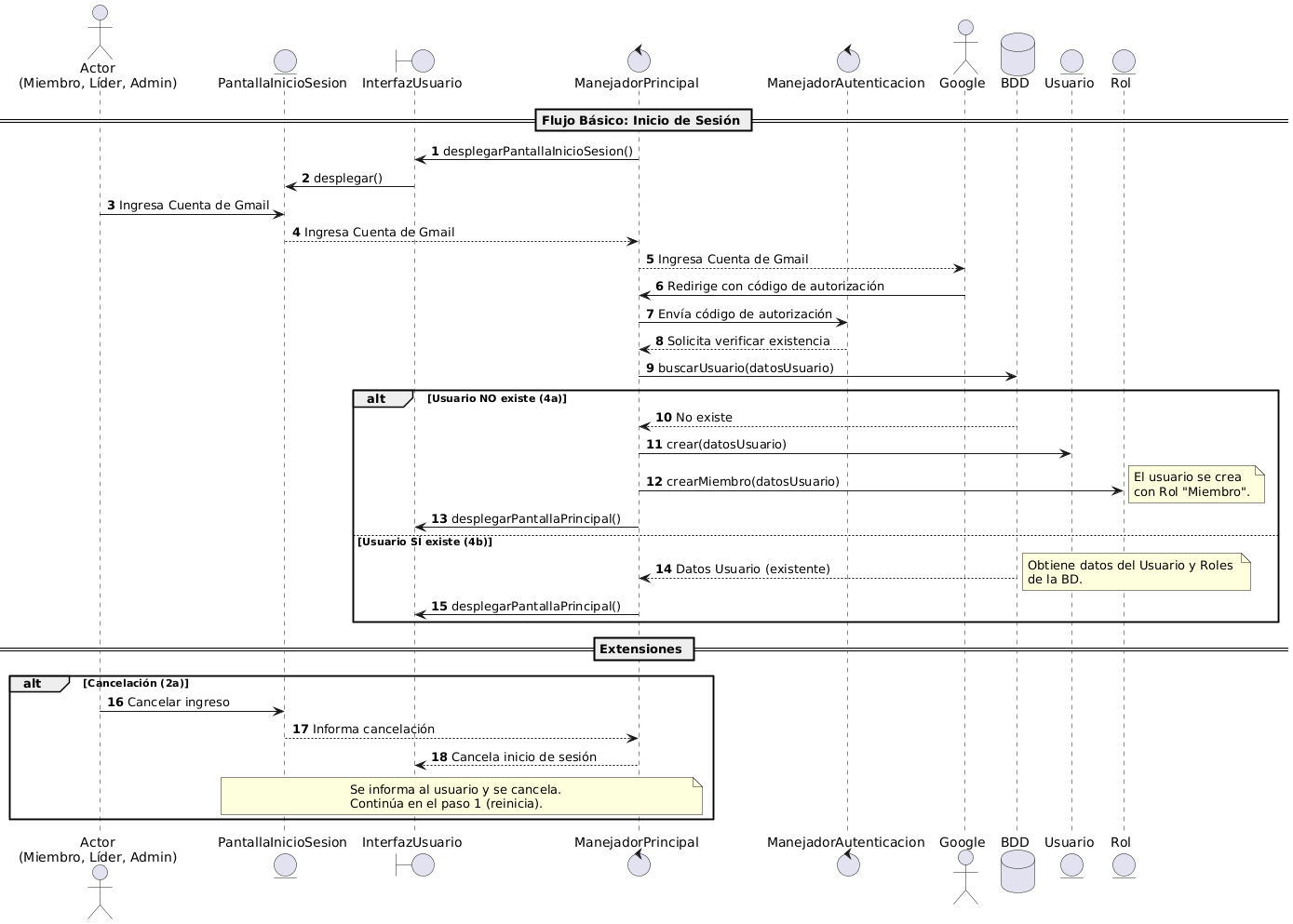
# Diseño de Casos de Uso

## Caso de Uso CU 01: Iniciar Sesión

### Especificación de caso de uso de diseño

| CASO DE USO #01 | | Iniciar sesión |
| --- | --- | --- |
| Actor Principal | | Administrador, Líder de Proyecto, Miembro |
| Precondiciones | | El actor debe tener una cuenta de Gmail. |
| Disparador | | El actor ingresa al sistema Kairos y presiona el botón “Ingresar con Google”. |
| Descripción | Paso | Acción |
|  | 1 | El ManejadorPrincipal solicita a la InterfazUsuario desplegar la PantallaInicioSesion. La InterfazUsuario solicita a la PantallaInicioSesion desplegarse. La PantallaInicioSesion se despliega. |
| 2 | El actor ingresa su cuenta de Gmail.  Google autentica el usuario y redirige al sistema con un código de autorización. |
| 3 | El ManejadorPrincipal recibe el código de autorización.  El ManejadorPrincipal envía el código de autorización al ManejadorAutenticacion. |
|  | 4 | El ManejadorAutenticacion obtiene los datos del Usuario.  El ManejadorAutenticacion solicita al ManejadorPrincipal verificar existencia en la Base de Datos. |
|  | 5 | El ManejadorPrincipal solicita a la InterfazUsuario desplegar la PantallaPrincipal |
| Extensiones | | Acción de Derivación |
|  | 2a | El usuario cancela el ingreso:  Se informa al usuario y se cancela el inicio de sesión.  Continúa en el CU01, paso 1. |
|  | 4a | El usuario no existe en la base de datos:  Se crea el Usuario con los datos obtenidos, se le asigna el Rol de “Miembro” dentro del sistema.  Continúa en el CU01, paso 5. |
|  | 4b | El usuario existe en la base de datos:  Se obtienen los datos e ingresa al sistema.  Continúa en el CU01, paso 5. |

**Diagrama de Secuencia**

**

**Diseño de Flujo de eventos**

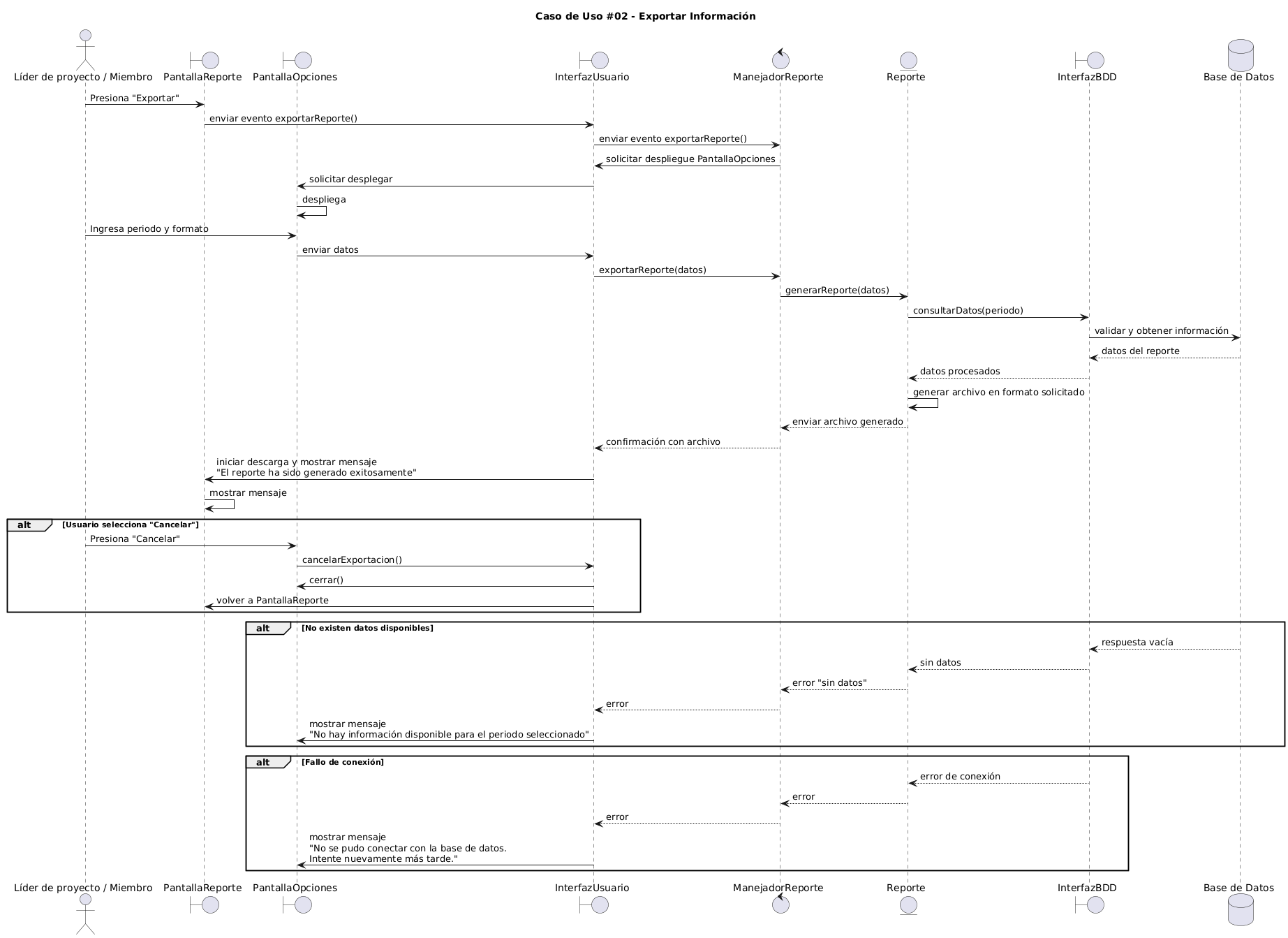
**Requerimientos especiales o de implementación**

## Caso de Uso CU 02: Exportar Información

### Especificación de caso de uso de diseño

| CASO DE USO #02 | | Exportar información |
| --- | --- | --- |
| Actor Principal | | Líder de proyecto, miembro |
| Disparador | | El actor presiona el botón “Exportar” desde la pantalla de reportes. |
| Precondiciones | | El actor debe estar autenticado en el sistema.  Debe existir información generada (proyectos, tareas, reportes). |
| Descripción | Paso | Acción |
|  | 1 | La InterfazUsuarioenvía el evento *exportarReporte()* al ManejadorReporte. El ManejadorReporte solicita a la InterfazUsuario desplegar la PantallaOpciones. La PantallaOpciones se despliega. |
| 2 | El actor ingresa los datos ‘periodo’ y ‘formato*’*, y presiona el botón **“**Generar**”**. La PantallaOpciones envía los datos ingresados a la InterfazUsuario, que los reenvía al ManejadorReporte mediante *exportarReporte(datos)*. |
| 3 | El ManejadorReporte valida los datos recibidos y los envía a la entidad Reporte con el evento *generarReporte(datos*). La entidad Reporte envía una solicitud a la InterfazBDD para consultar la información correspondiente en la base de datos (*consultarDatos(periodo*)). |
| 4 | La BDD valida los datos del periodo y responde a la InterfazBDD con los resultados. La InterfazBDD envía los datos procesados a la entidad Reporte, que genera el archivo del reporte en el formato solicitado. La entidad Reporte envía el resultado al ManejadorReporte con un mensaje de confirmación. |
|  | 5 | El ManejadorReporte envía el archivo a la InterfazUsuario, que solicita a la PantallaReporte iniciar la descarga. La PantallaOpciones muestra el mensaje “El reporte ha sido generado exitosamente” y el archivo se descarga al dispositivo del usuario. |
| Extensiones | |  |
|  | 2a | El usuario cancela la operación:  El actor presiona “Cancelar”. La PantallaOpciones envía el evento *cancelarExportacion()* a la InterfazUsuario. La InterfazUsuario solicita cerrar la PantallaOpciones y volver a la PantallaReporte. |
|  | 4a | No existen datos para el periodo seleccionado:  La BDD devuelve una respuesta vacía a la InterfazBDD. La InterfazBDD envía un mensaje de error a la entidad Reporte, que lo reenvía al ManejadorReporte . El ManejadorReporte envía el error a la InterfazUsuario, la cual solicita a la PantallaOpciones mostrar el mensaje “No hay información disponible para el periodo seleccionado”. |
|  | 4b | Error de conexión con la base de datos:  La InterfazBDD detecta un fallo de conexión y envía un mensaje de error a la entidad Reporte. La entidad Reporte envía el error al ManejadorReporte, que lo pasa a la InterfazUsuario. La InterfazUsuario solicita a la PantallaOpciones mostrar el mensaje “No se pudo conectar con la base de datos. Intente nuevamente más tarde.” |

**Diagrama de Secuencia**

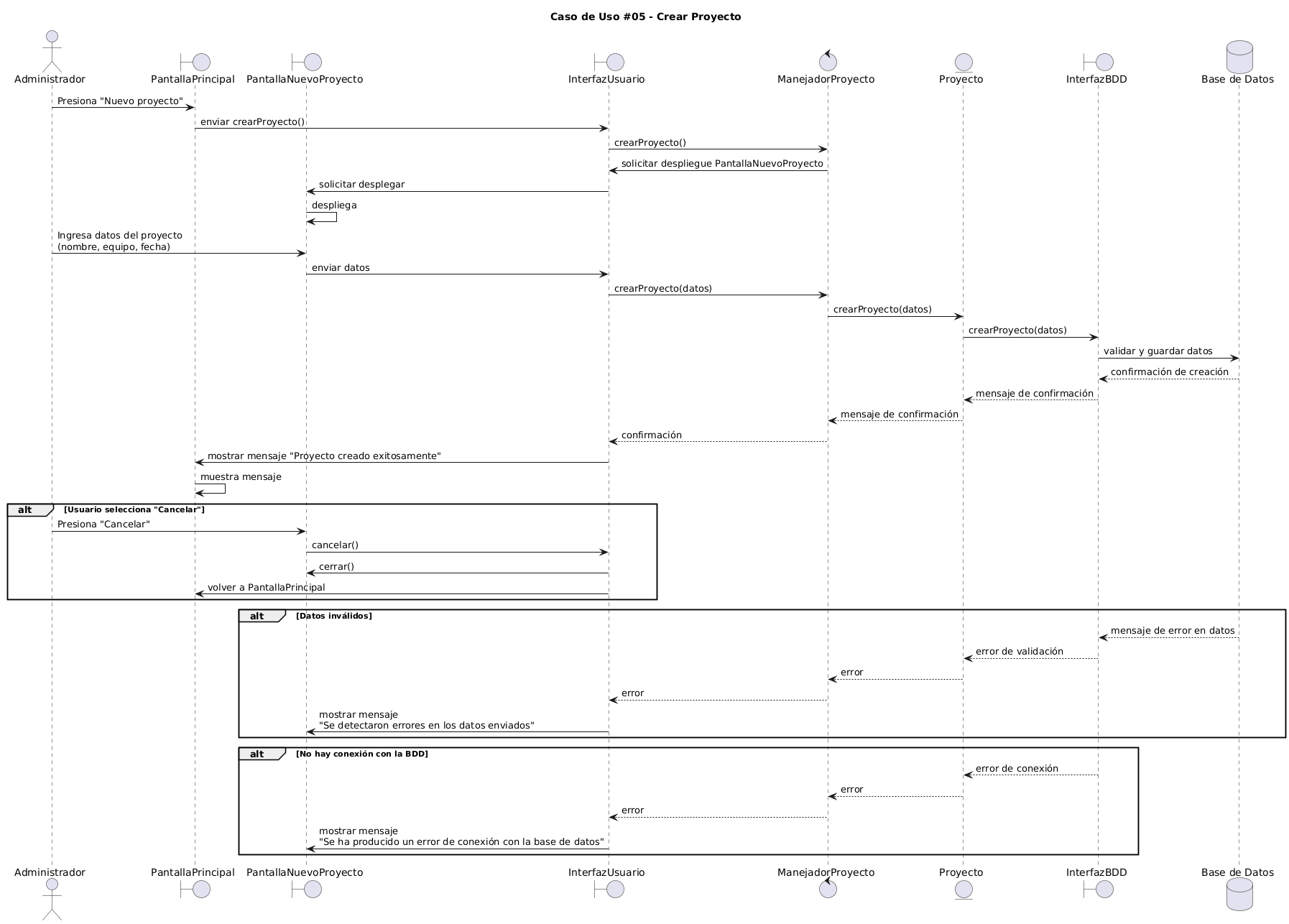


## Caso de Uso CU 05: Crear Proyecto

### Especificación de caso de uso de diseño

| CASO DE USO #05 | | Crear proyecto |
| --- | --- | --- |
| Actor Principal | | Administrador |
| Precondiciones | | El actor debe estar autenticado en el sistema. |
| Disparador | | El actor presiona el botón “Nuevo proyecto” desde la pantalla principal. |
| Descripción | Paso | Acción |
|  | 1 | La InterfazUsuario envía el evento *crearProyecto* al ManejadorProyecto. El ManejadorProyecto solicita a la InterfazUsuario desplegar la PantallaNuevoProyecto. La PantallaNuevoProyecto se despliega. |
| 2 | El actor ingresa la información en los campos obligatorios ‘nombre de proyecto’, ‘nombre del equipo’, y ‘fecha de inicio’, y presiona el botón “Crear”. La PantallaNuevoProyecto envía los datos a la InterfazUsuario. La InterfazUsuario envía el evento *crearProyecto(datos)* al ManejadorProyecto. |
| 3 | El ManejadorProyecto envía el evento a la entidad Proyecto. La entidad Proyecto envía el evento a la InterfazBDD, quien le solicita a la BDD *crearProyecto(datos)*. |
|  | 4 | La BDD valida los datos, guarda la información y envía un mensaje de confirmación con los datos del nuevo proyecto a la InterfazBDD. La InterfazBDD envía el mensaje a la entidad Proyecto. La entidad Proyecto envía el mensaje de confirmación al ManejadorProyecto, quien envía el mensaje a la InterfazUsuario. La InterfazUsuario solicita a la PantallaPrincipal mostrar el mensaje “El proyecto ha sido creado exitosamente”. |
| Extensiones | | Acción de Derivación |
|  | 2a | El usuario selecciona la opción “Cancelar”:  La PantallaNuevoProyecto envía el evento *cancelar* a la InterfazUsuario. La InterfazUsuario solicita cerrar la PantallaNuevoProyecto, volviendo a la PantallaInicio. |
|  | 4a | Los datos ingresados son incorrectos:  La BDD detecta errores en los datos enviados y cancela la operación. La BDD envía un mensaje de error a la InterfazBDD. La InterfazBDD envía el mensaje de error a la entidad Proyecto. La entidad Proyecto envía el mensaje de error al ManejadorProyecto. El Manejador Proyecto envía el mensaje de error a la InterfazUsuario. La InterfazUsuario solicita a la PantallaNuevoProyecto mostrar el mensaje “Se detectaron errores en los datos enviados”. La PantallaNuevoProyecto muestra el mensaje. Continúa en el CU05, paso 2. |
|  | 4b | La InterfazBDD no logra conectarse con la BDD:  La InterfazBDD envía el mensaje de error de conexión a la entidad Proyecto. La entidad Proyecto envía el mensaje al ManejadorProyecto. El Manejador Proyecto envía el mensaje a la InterfazUsuario. La InterfazUsuario solicita a la PantallaNuevoProyecto mostrar el mensaje “Se ha producido un error de conexión con la base de datos”. La PantallaNuevoProyecto muestra el mensaje. Continúa en el CU05, paso 2. |

**Diagrama de Secuencia**



**Diseño de Flujo de eventos**

**Requerimientos especiales o de implementación**

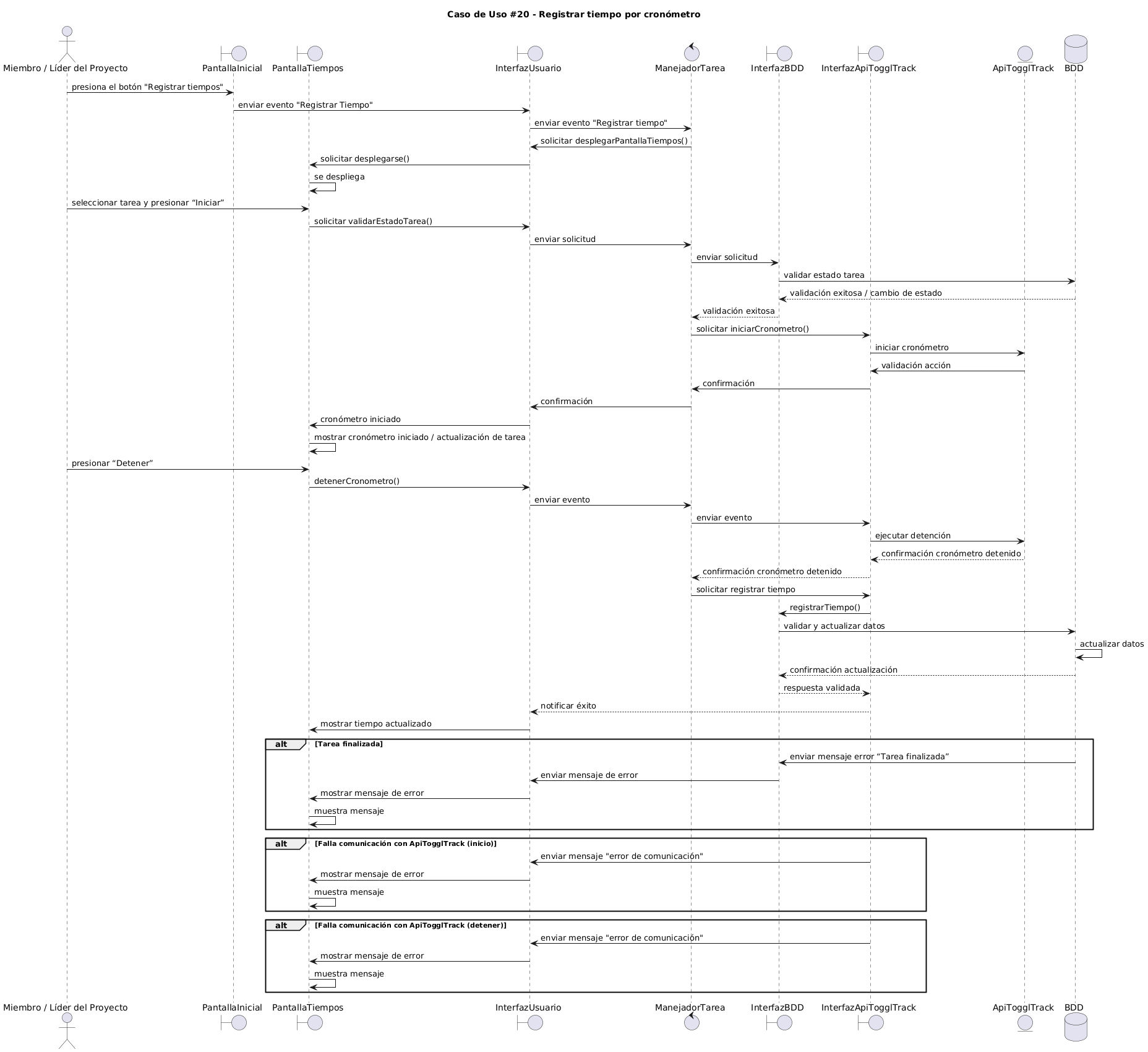
## Caso de Uso 20: Registro de tiempo por cronómetro

### Especificación de caso de uso de diseño

## 

| CASO DE USO #20 | | Registrar tiempo por cronómetro |
| --- | --- | --- |
| Actor Principal | | Miembro, Líder del Proyecto |
| Precondiciones | | El actor debe estar autenticado en el sistema.  La tarea debe estar registrada y asignada a un miembro. |
| Disparador | | El actor selecciona la opción “Registrar tiempo”. |
| Descripción | Paso | Acción |
|  | 1 | El actor selecciona la opción “Registrar Tiempo” desde la PantallaInicio. La PantallaInicio envía el evento a la InterfazUsuario. La InterfazUsuario envía el evento al ManejadorTarea. |
| 2 | El ManejadorTarea solicita desplegarPantallaTiempos al InterfazUsuario.  El InterfazUsuario solicita a la PantallaTiempos desplegarse. La PantallaTiempos se despliega. |
|  | 3 | El actor selecciona una tarea y presiona el botón “Iniciar”. La PantallaTiempos solicita validarEstadoTarea a InterfazUsuario. La InterfazUsuario envía la solicitud al ManejadorTarea. El ManejadorTarea envía la solicitud a InterfazBDD. La InterfazBDD envía la solicitud a la BDD y ésta valida la petición, cambiando el estado de la tarea. |
|  | 4 | El ManejadorTarea solicita a la InterfazApiTogglTrack iniciarCronometro. La InterfazApiTogglTrack envía la solicitud a la ApiTogglTrack, y ésta valida la acción. |
|  | 5 | La ApiTogglTrack inicia el cronómetro y envía a la InterfazApiTogglTrack un mensaje de confirmación. La InterfazApiTogglTrack envía el mensaje a la InterfazUsuario y ésta refleja el cronómetro iniciado en la PantallaTiempos. |
|  | 6 | El actor presiona el botón “Detener” y la PantallaTiempos envía el evento pausarCronometro a la InterfazUsuario. La InterfazUsuario envía el evento al ManejadorTarea, quien lo envía a la InterfazApiTogglTrack . La InterfazApiTogglTrack envía el evento a la ApiTogglTrack y ésta lo ejecuta. La ApiTogglTrack envía un mensaje de confirmación a la InterfazApiTogglTrack. La InterfazApiTogglTrack envía un mensaje de confirmación al ManejadorTarea. El ManejadorTarea envía el evento registrarTiempo a la InterfazBDD. La InterfazBDD envía el evento a la BDD. La BDD lo valida y actualiza los datos. |
| Extensiones | | Acción de Derivación |
|  | 3a | La BDD detecta que la tarea sobre la que se solicita registrar tiempo se encuentra con el estado ‘finalizado’. La BDD envía un mensaje de error a la InterfazBDD. La InterfazBDD envía el mensaje a la InterfazUsuario. La InterfazUsuario envía el mensaje a la PantallaTiempos . En la PantallaTiempos se muestra el mensaje de error. Continúa en el CU20, paso 3. |
|  | 4a | La InterfazApiTogglTrack no logra comunicarse con la ApiTogglTrack. La InterfazApiTogglTrack envía un mensaje de error a la InterfazUsuario. La InterfazUsuario envía el mensaje de error a la PantallaTiempos. Continúa en el CU20, paso 3. |
|  | 6a | La InterfazApiTogglTrack no logra comunicarse con la ApiTogglTrack. La InterfazApiTogglTrack envía un mensaje de error a la InterfazUsuario. La InterfazUsuario envía el mensaje de error a la PantallaTiempos. Continúa en el CU20, paso 3. |

### Diagrama de Secuencia

**Figura 01 - Diagrama de secuencia CU20: Registrar tiempo por cronómetro**

**Diseño de Flujo de eventos**

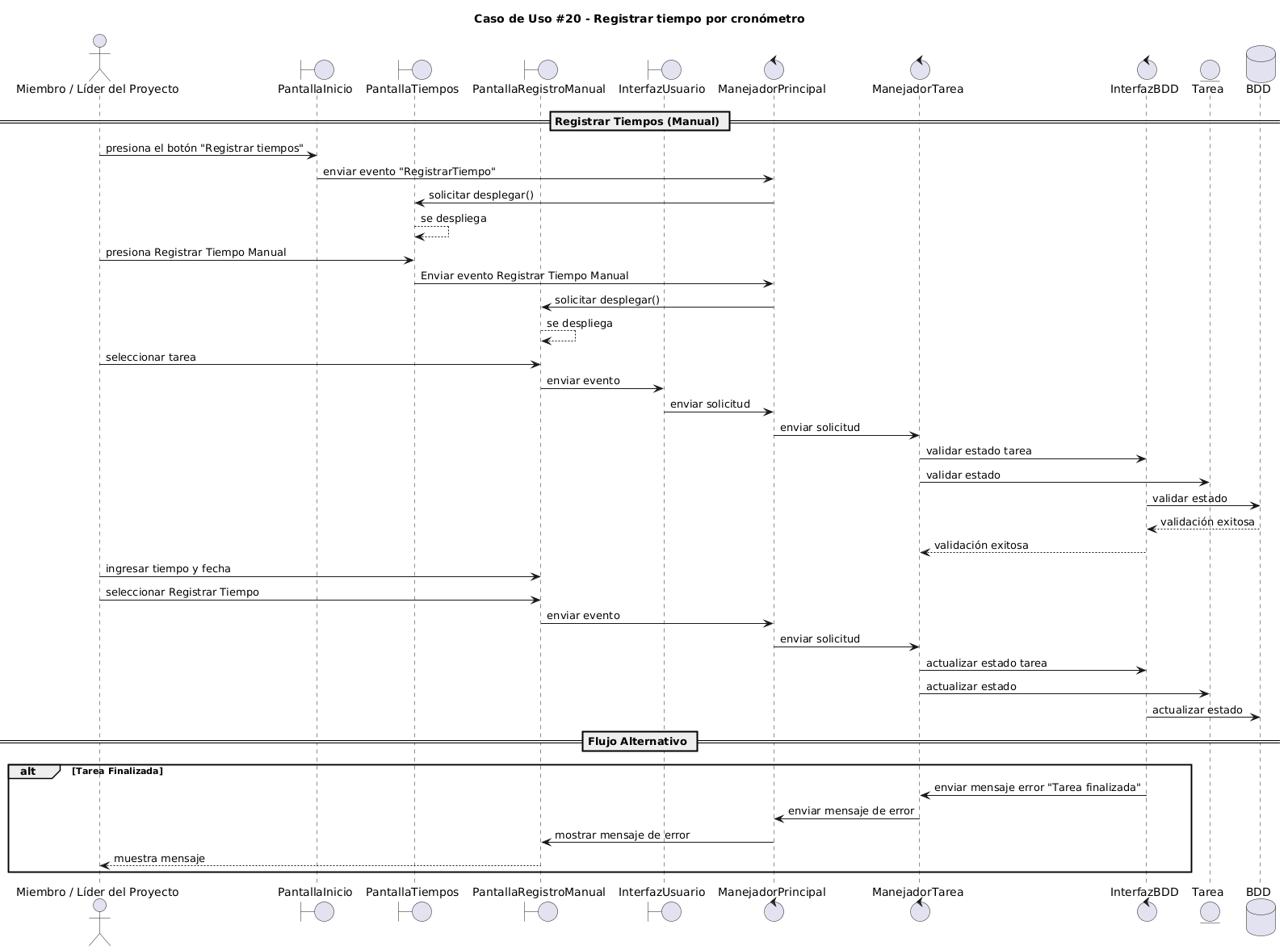
## Caso de Uso 21: Registro de tiempo manual

### Especificación de caso de uso de diseño

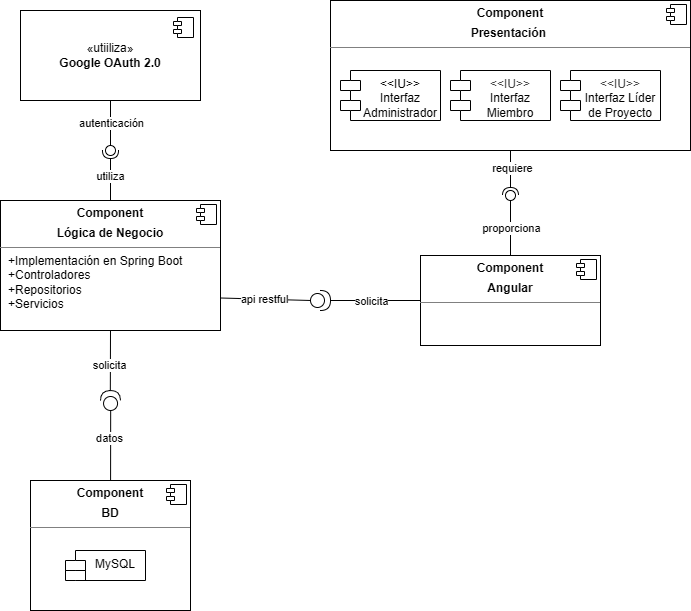
## 

| CASO DE USO #21 | | Registrar tiempo manual |
| --- | --- | --- |
| Actor Principal | | Miembro, Líder del Proyecto |
| Precondiciones | | El actor debe estar autenticado en el sistema.  La tarea debe estar registrada y asignada a un miembro. |
| Disparador | | El actor selecciona la opción “Registrar tiempo”. |
| Descripción | Paso | Acción |
|  | 1 | El actor selecciona la opción “Registrar Tiempo” desde la PantallaInicio. La PantallaInicio envía el evento a la InterfazUsuario. La InterfazUsuario envía el evento al ManejadorTarea. |
| 2 | El ManejadorTarea solicita desplegar PantallaTiempos al InterfazUsuario.  El InterfazUsuario solicita a la PantallaTiempos desplegarse. La PantallaTiempos se despliega. |
|  | 3 | El actor selecciona “Registrar Tiempo Manual”. La PantallaTiempos solicita desplegar PantallaRegistroManual a InterfazUsuario. |
|  | 4 | El actor selecciona la tarea a registrar.  La PantallaRegistroManual envía el evento a ManejadorTarea. |
|  | 5 | El actor ingresa el tiempo invertido en la tarea y la fecha.  El actor selecciona la opción “Registrar Tiempo”  El ManejadorTarea envía la solicitud a InterfazBDD. La InterfazBDD envía la solicitud a la BDD y ésta valida la petición, cambiando el estado de la tarea. |
| Extensiones | | Acción de Derivación |
|  | 4a | La BDD detecta que la tarea sobre la que se solicita registrar tiempo se encuentra con el estado ‘finalizado’. La BDD envía un mensaje de error a la InterfazBDD. La InterfazBDD envía el mensaje a la InterfazUsuario. La InterfazUsuario envía el mensaje a la PantallaTiempos. En la PantallaTiempos se muestra el mensaje de error. Continúa en el CU21, paso 3. |

### Diagrama de Secuencia

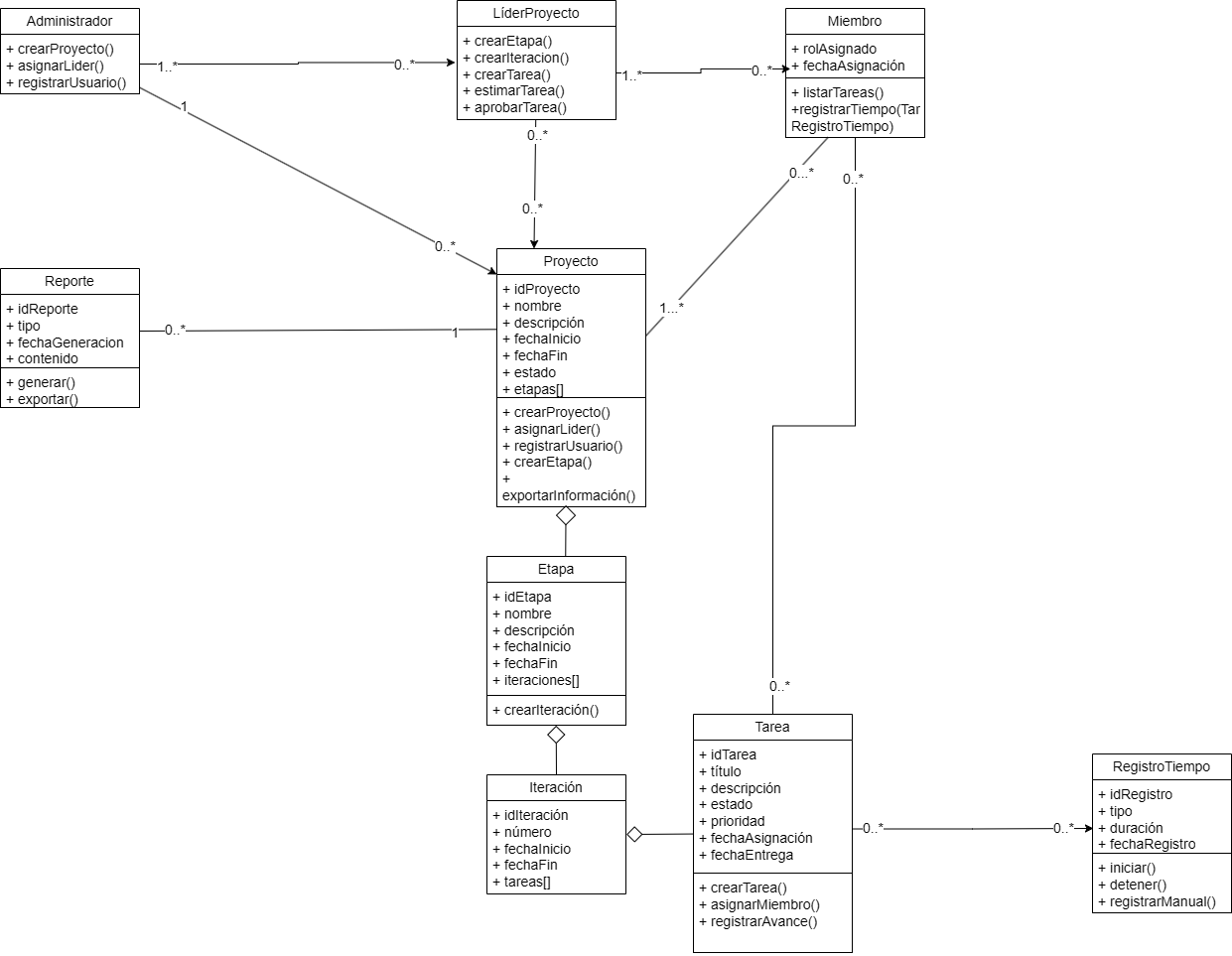
**

## Diagrama de componentes

****

**Fig. 2.1: Modelo de componentes**.

## Diagrama de clases



**Fig. 2.2: Diagrama de Clases.**