[**CU 05 - Crear Proyecto**](#_xcyb6xsxknag) **1**

[**CU 02 - Exportar Información**](#_kq3okftx97v) **3**

[**CU 20 - Registrar Tiempo por Cronómetro**](#_y5ugs1qfi6cu) **5**

# CU01 - Iniciar Sesión

@startuml CU01\_IniciarSesion

actor "Actor\n(Miembro, Líder, Admin)" as A

entity PantallaInicioSesion as PIS

boundary InterfazUsuario as IU

control ManejadorPrincipal as MP

control ManejadorAutenticacion as MA

actor Google

database "BDD" as BD

entity Usuario as U

entity Rol

autonumber

== Flujo Básico: Inicio de Sesión ==

MP -> IU: desplegarPantallaInicioSesion()

IU -> PIS: desplegar()

A -> PIS: Ingresa Cuenta de Gmail

PIS --> MP: Ingresa Cuenta de Gmail

MP --> Google: Ingresa Cuenta de Gmail

Google -> MP: Redirige con código de autorización

MP -> MA: Envía código de autorización

MA --> MP: Solicita verificar existencia

MP -> BD: buscarUsuario(datosUsuario)

alt Usuario \*\*NO\*\* existe (4a)

BD --> MP: No existe

MP -> U: crear(datosUsuario)

MP -> Rol: crearMiembro(datosUsuario)

note right: El usuario se crea \ncon Rol "Miembro".

MP -> IU: desplegarPantallaPrincipal()

else Usuario \*\*SÍ\*\* existe (4b)

BD --> MP: Datos Usuario (existente)

note right: Obtiene datos del Usuario y Roles \nde la BD.

MP -> IU: desplegarPantallaPrincipal()

end

== Extensiones ==

alt Cancelación (2a)

A -> PIS: Cancelar ingreso

PIS --> MP: Informa cancelación

MP --> IU: Cancela inicio de sesión

note over MP, PIS: Se informa al usuario y se cancela. \nContinúa en el paso 1 (reinicia).

end

@enduml

# CU 05 - Crear Proyecto

@startuml

title Caso de Uso #05 - Crear Proyecto

actor A as "Administrador"

boundary IP as "PantallaPrincipal"

boundary INP as "PantallaNuevoProyecto"

boundary IU as "InterfazUsuario"

control M as "ManejadorProyecto"

entity P as "Proyecto"

boundary IBDD as "InterfazBDD"

database BDD as "Base de Datos"

A -> IP : Presiona "Nuevo proyecto"

IP -> IU: enviar crearProyecto()

IU -> M : crearProyecto()

M -> IU : solicitar despliegue PantallaNuevoProyecto

IU -> INP : solicitar desplegar

INP -> INP : despliega

A -> INP : Ingresa datos del proyecto\n(nombre, equipo, fecha)

INP -> IU : enviar datos

IU -> M : crearProyecto(datos)

M -> P : crearProyecto(datos)

P -> IBDD : crearProyecto(datos)

IBDD -> BDD : validar y guardar datos

BDD --> IBDD : confirmación de creación

IBDD --> P : mensaje de confirmación

P --> M : mensaje de confirmación

M --> IU : confirmación

IU -> IP : mostrar mensaje "Proyecto creado exitosamente"

IP -> IP : muestra mensaje

alt Usuario selecciona "Cancelar"

A -> INP : Presiona "Cancelar"

INP -> IU : cancelar()

IU -> INP : cerrar()

IU -> IP : volver a PantallaPrincipal

end

alt Datos inválidos

BDD --> IBDD : mensaje de error en datos

IBDD --> P : error de validación

P --> M : error

M --> IU : error

IU -> INP : mostrar mensaje\n"Se detectaron errores en los datos enviados"

end

alt No hay conexión con la BDD

IBDD --> P : error de conexión

P --> M : error

M --> IU : error

IU -> INP : mostrar mensaje\n"Se ha producido un error de conexión con la base de datos"

end

@enduml

# CU 02 - Exportar Información

@startuml

title Caso de Uso #02 - Exportar Información

actor A as "Líder de proyecto / Miembro"

boundary PR as "PantallaReporte"

boundary PO as "PantallaOpciones"

boundary IU as "InterfazUsuario"

control MR as "ManejadorReporte"

entity R as "Reporte"

boundary IBDD as "InterfazBDD"

database BDD as "Base de Datos"

A -> PR : Presiona "Exportar"

PR -> IU : enviar evento exportarReporte()

IU -> MR : enviar evento exportarReporte()

MR -> IU : solicitar despliegue PantallaOpciones

IU -> PO : solicitar desplegar

PO -> PO: despliega

A -> PO : Ingresa periodo y formato

PO -> IU : enviar datos

IU -> MR : exportarReporte(datos)

MR -> R : generarReporte(datos)

R -> IBDD : consultarDatos(periodo)

IBDD -> BDD : validar y obtener información

BDD --> IBDD : datos del reporte

IBDD --> R : datos procesados

R -> R : generar archivo en formato solicitado

R --> MR : enviar archivo generado

MR --> IU : confirmación con archivo

IU -> PR : iniciar descarga y mostrar mensaje\n"El reporte ha sido generado exitosamente"

PR -> PR : mostrar mensaje

alt Usuario selecciona "Cancelar"

A -> PO : Presiona "Cancelar"

PO -> IU : cancelarExportacion()

IU -> PO : cerrar()

IU -> PR : volver a PantallaReporte

end

alt No existen datos disponibles

BDD --> IBDD : respuesta vacía

IBDD --> R : sin datos

R --> MR : error "sin datos"

MR --> IU : error

IU -> PO : mostrar mensaje\n"No hay información disponible para el periodo seleccionado"

end

alt Fallo de conexión

IBDD --> R : error de conexión

R --> MR : error

MR --> IU : error

IU -> PO : mostrar mensaje\n"No se pudo conectar con la base de datos.\nIntente nuevamente más tarde."

end

@enduml

# CU 20 - Registrar Tiempo por Cronómetro

@startuml

title Caso de Uso #20 - Registrar tiempo por cronómetro

actor "Miembro / Líder del Proyecto" as Actor

boundary "PantallaInicial" as PI

boundary "PantallaTiempos" as PT

boundary "InterfazUsuario" as IU

control "ManejadorTarea" as MT

boundary "InterfazBDD" as IBDD

boundary "InterfazApiTogglTrack" as IApiTT

entity "ApiTogglTrack" as ApiTT

database "BDD" as BDD

Actor -> PI : presiona el botón "Registrar tiempos"

PI -> IU : enviar evento "Registrar Tiempo"

IU -> MT : enviar evento "Registrar tiempo"

MT -> IU : solicitar desplegarPantallaTiempos()

IU -> PT : solicitar desplegarse()

PT -> PT : se despliega

Actor -> PT : seleccionar tarea y presionar “Iniciar”

PT -> IU : solicitar validarEstadoTarea()

IU -> MT : enviar solicitud

MT -> IBDD : enviar solicitud

IBDD -> BDD : validar estado tarea

BDD --> IBDD : validación exitosa / cambio de estado

IBDD --> MT : validación exitosa

MT -> IApiTT : solicitar iniciarCronometro()

IApiTT -> ApiTT : iniciar cronómetro

ApiTT -> IApiTT : validación acción

IApiTT -> MT : confirmación

MT -> IU : confirmación

IU -> PT : cronómetro iniciado

PT -> PT : mostrar cronómetro iniciado / actualización de tarea

Actor -> PT : presionar “Detener”

PT -> IU : detenerCronometro()

IU -> MT : enviar evento

MT -> IApiTT : enviar evento

IApiTT -> ApiTT : ejecutar detención

ApiTT --> IApiTT : confirmación cronómetro detenido

IApiTT --> MT : confirmación cronómetro detenido

MT -> IApiTT : solicitar registrar tiempo

IApiTT -> IBDD : registrarTiempo()

IBDD -> BDD : validar y actualizar datos

BDD -> BDD : actualizar datos

BDD --> IBDD : confirmación actualización

IBDD --> IApiTT : respuesta validada

IApiTT --> IU : notificar éxito

IU -> PT : mostrar tiempo actualizado

alt Tarea finalizada

BDD -> IBDD : enviar mensaje error “Tarea finalizada”

IBDD -> IU : enviar mensaje de error

IU -> PT : mostrar mensaje de error

PT -> PT : muestra mensaje

end

alt Falla comunicación con ApiTogglTrack (inicio)

IApiTT -> IU : enviar mensaje "error de comunicación"

IU -> PT : mostrar mensaje de error

PT -> PT : muestra mensaje

end

alt Falla comunicación con ApiTogglTrack (detener)

IApiTT -> IU : enviar mensaje "error de comunicación"

IU -> PT : mostrar mensaje de error

PT -> PT : muestra mensaje

end

@enduml