

Casos de uso

1. Contexto y límites

- **Actor principal:** Persona usuaria (aplicación de escritorio, mono-usuario).
- **Límite del sistema:** Cliente que consume el servidor de la práctica conforme al swagger (Friends, Expenses y Friends↔Expenses solo **lectura** de participantes).
- **Fuera de alcance (otro equipo):** añadir/editar/borrar **amigos** y **participantes de un gasto**.

2. Lista de casos de uso

UC-01	Ver lista de gastos
UC-02	Crear gasto
UC-03	Editar gasto
UC-04	Eliminar gasto
UC-05	Ver detalle de un gasto (lectura de participantes y balances del gasto)
UC-06	Ver lista de amigos (lectura)
UC-07	Ver detalle de un amigo (gastos por amigo y balances por gasto)
UC-08	Ver balance general

3. Detalle de casos de uso

UC-01 – Ver lista de gastos

Campo	Descripción
Objetivo	Consultar todos los gastos con <code>description</code> , <code>date</code> (YYYY-MM-DD), <code>amount</code> , <code>num friends</code> y <code>total credit balance</code> .
Precondiciones	Conectividad disponible.
Flujo principal	1) Solicitar la lista de gastos. 2) Mostrar lista scrolleable con los campos indicados y acceso al detalle. 3) Permitir orden/filtrado local .
Alternativas/errores	A1: Sin datos → estado “No hay gastos”. A2: Error/timeout → mensaje y acción “Reintentar”.
Postcondición	La lista queda visible y navegable.

Campo	Descripción
API (lectura)	GET /expenses

UC-02 – Crear gasto

Campo	Descripción
Objetivo	Registrar un nuevo gasto con <code>description</code> , <code>date</code> , <code>amount</code> .
Precondiciones	Datos válidos: <code>description</code> no vacía; <code>date</code> formato YYYY-MM-DD ; <code>amount</code> numérico > 0.
Flujo principal	1) Mostrar formulario. 2) Validar. 3) Crear el gasto. 4) Refrescar lista y/o abrir detalle.
Alternativas/errores	A1: Datos inválidos → validación y bloqueo. A2: Error servidor → mensaje + reintento/cancelar.
Postcondición	El nuevo gasto existe y es visible.
API (creación)	POST /expenses

UC-03 – Editar gasto

Campo	Descripción
Objetivo	Modificar <code>description</code> , <code>date</code> o <code>amount</code> de un gasto existente.
Precondiciones	Gasto existente seleccionado; datos válidos.
Flujo principal	1) Mostrar formulario precargado. 2) Validar cambios. 3) Guardar. 4) Refrescar vista.
Alternativas/errores	Validaciones como en UC-02; error servidor → mensaje + reintento.
Postcondición	Gasto actualizado y visible.
API (actualización)	PUT /expenses/{expense_id}

UC-04 – Eliminar gasto

Campo	Descripción
Objetivo	Borrar un gasto existente.
Precondiciones	Gasto existente seleccionado; confirmación de la persona usuaria.
Flujo principal	1) Confirmar borrado. 2) Eliminar. 3) Actualizar lista.

Campo	Descripción
Alternativas/errores	Error/timeout → mensaje + reintento/cancelar.
Postcondición	El gasto deja de mostrarse.
API (borrado)	<code>DELETE /expenses/{expense_id}</code>

UC-05 – Ver detalle de un gasto (lectura de participantes)

Campo	Descripción
Objetivo	Consultar participantes ya asignados a ese gasto y sus saldos: <code>credit balance</code> (ha pagado) y <code>debit balance</code> (debe), además del balance total del gasto .
Precondiciones	Gasto existente.
Flujo principal	1) Mostrar resumen del gasto (<code>id</code> , <code>description</code> , <code>date</code> , <code>amount</code> , <code>total credit balance</code>). 2) Mostrar tabla de participantes con <code>name</code> , <code>credit balance</code> , <code>debit balance</code> . 3) Mostrar totales/reparto si aplica.
Alternativas/errores	A1: Sin participantes → mensaje informativo. A2: Error/timeout → mensaje + reintento.
Postcondición	Información visible; sin modificar participantes.
API (lectura)	<code>GET /expenses/{expense_id}</code> + <code>GET /expenses/{expense_id}/friends</code>

UC-06 – Ver lista de amigos (solo lectura)

Campo	Descripción
Objetivo	Consultar todos los amigos con <code>id</code> , <code>name</code> , <code>total credit balance</code> , <code>total debit balance</code> .
Precondiciones	Conectividad disponible.
Flujo principal	1) Solicitar la lista de amigos. 2) Mostrar lista con campos clave y acceso al detalle. 3) Permitir orden/filtrado local .
Alternativas/errores	Sin datos → estado vacío; error → mensaje + reintento.
Postcondición	Lista visible.
API (lectura)	<code>GET /friends</code>

UC-07 – Ver detalle de un amigo (solo lectura)

Campo	Descripción
Objetivo	Consultar la ficha del amigo (<code>id</code> , <code>name</code> , totales) y la lista de gastos en los que participa con sus saldos por gasto (<code>credit/debit balance</code> , <code>num friends</code> , <code>amount</code> , <code>description</code>).
Precondiciones	Amigo existente.
Flujo principal	1) Mostrar datos del amigo y su balance agregado (crédito/débito/neto). 2) Listar gastos en los que participa con su saldo por gasto.
Alternativas/errores	Sin gastos asociados → estado vacío; error → mensaje + reintento.
Postcondición	Información visible.
API (lectura)	<code>GET /friends/{friend id} + GET /friends/{friend_id}/expenses</code>

UC-08 – Ver balance general

Campo	Descripción
Objetivo	Visualizar, por amigo, <code>total credit balance</code> , <code>total debit balance</code> y neto .
Precondiciones	—
Flujo principal	1) Agregar información por amigo. 2) Mostrar tabla/resumen con opción de orden (por neto o alfabético).
Alternativas/errores	Sin datos → estado vacío; error → mensaje + reintento.
Postcondición	Resumen global visible.
API (lectura)	<code>GET /friends</code>

Diseño

09:52 AM



SPLIT WITH ME

FRIENDS

EXPENSES

09:52 AM



SPLIT WITH ME

FRIENDS



Friend 1

SHOW ALL

Friend 2

SHOW ALL

Friend 3

SHOW ALL

Friend 4

SHOW ALL

Friend 5

SHOW ALL

EXPENSES

REFRESH

09:52 AM



SPLIT WITH ME

FRIENDS



Friend 1



SHOW ALL

Friend 2



SHOW ALL

Friend 3



SHOW ALL

Friend 4



SHOW ALL

Friend 5



SHOW ALL

USER'S EXPENSES	
EXPENSE NAME	IMPORTE
GASTO 1	\$\$
GASTO 2	\$\$

Description

Fecha



EXPENSES

REFRESH

09:52 AM



SPLIT WITH ME

EXPENSES



EXPENSE 1

SHOW ALL

EXPENSE 2

SHOW ALL

EXPENSE 3

SHOW ALL

EXPENSE 4

SHOW ALL

EXPENSE 5

SHOW ALL

FRIENDS

REFRESH

09:52 AM



SPLIT WITH ME

EXPENSES



EXPENSE 1

SHOW ALL

EXPENSE 2

SHOW ALL

EXPENSE 3

SHOW ALL

EXPENSE 4

SHOW ALL

EXPENSE 5

SHOW ALL

EXPENSE NAME	
USER NAME	IMPORTE
USER 1	\$\$
USER 2	\$\$

Description

Fecha



FRIENDS

REFRESH

09:52 AM



SPLIT WITH ME

CREAR GASTO

Description

Fecha

Monto

Participantes:

☐ FRIEND 1

☐ FRIEND 2

☐ FRIEND 3



CONFIRMAR

09:52 AM



SPLIT WITH ME

MODIFICAR GASTO

Description

Fecha

Monto

Participantes:

☐ FRIEND 1

☐ FRIEND 2

☐ FRIEND 3

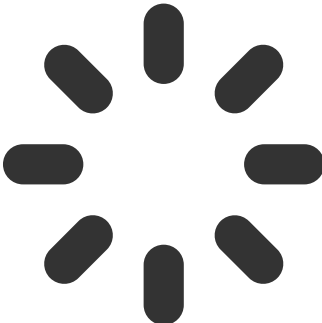


CONFIRMAR

09:52 AM



SPLIT WITH ME



09:52 AM



SPLIT WITH ME



ERROR!

SPLIT WITH ME

FRIENDS

EXPENSES

SPLIT WITH ME

FRIENDS

+

Friend 1



SHOW ALL

Friend 2



SHOW ALL

Friend 3



SHOW ALL

Friend 4



SHOW ALL

Friend 5



SHOW ALL

EXPENSES

REFRESH

SPLIT WITH ME

FRIENDS

+

Friend 1

SHOW ALL

Friend

ALL

Friend

ALL

Friend

ALL

Friend

ALL

USER'S EXPENSES	
EXPENSE NAME	IMPORTE
GASTO 1	\$\$
GASTO 2	\$\$

Description

Fecha



EXPENSES

REFRESH

SPLIT WITH ME

EXPENSES

+

Expense 1



SHOW ALL

Expense 2



SHOW ALL

Expense 3



SHOW ALL

Expense 4



SHOW ALL

Expense 5



SHOW ALL

FRIENDS

REFRESH

SPLIT WITH ME

EXPENSES

+

Expense 1

SHOW ALL

Expense

ALL

Expense

ALL

Expense

ALL

Expense

ALL

EXPENSE NAME	
USER NAME	IMPORTE
USER 1	\$\$
USER 2	\$\$

Description

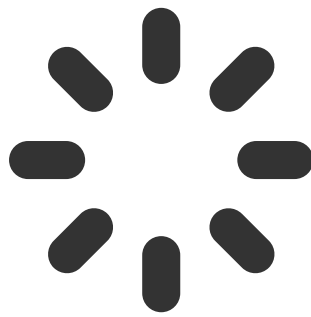
Fecha



FRIENDS

REFRESH

SPLIT WITH ME



SPLIT WITH ME



ERROR!

SPLIT WITH ME

CREAR GASTO

Description

Fecha



Monto

Participantes:

☐ FRIEND 1

☐ FRIEND 2

☐ FRIEND 3

CONFIRMAR

SPLIT WITH ME

MODIFICAR GASTO

Description

Fecha



Monto

Participantes:

☐ FRIEND 1

☐ FRIEND 2

☐ FRIEND 3

CONFIRMAR