

# **Universidad Politécnica Internacional**

Curso: técnicas de programación

Profesor: Luis Felipe Mora Umaña

Proyecto #2

SplitBuddies

Estudiantes

Daniel Benavides Robles

Freddy Martínez Espinoza

II Cuatrimestre 2025

## Contenido

1. Introducción .....	3
2. Decisiones de diseño .....	4
2.1. Cómo ejecutar el programa .....	5
1. Clonar o descargar el repositorio desde GitHub en la carpeta de “Proyecto #2” .....	5
2. Abrir la solución en Visual Studio 2022 o superior. ....	5
2.2. Manual de usuario .....	5
a) Pantalla de Login .....	5
b) Crear grupo .....	6
c) Calcular balance .....	8
d) Agregar gastos .....	9
4. Conclusiones y aprendizajes.....	16

## 1. Introducción

SplitBuddies es una aplicación desarrollada en C# que permite gestionar grupos de amigos o compañeros, registrar gastos compartidos y calcular automáticamente el balance de deudas entre los participantes. El sistema busca facilitar la organización de cuentas en actividades grupales como viajes, cenas o proyectos.

En esta segunda versión (v2), el sistema fue mejorado aplicando principios de Programación Orientada a Objetos, Clean Code y el patrón MVC, corrigiendo limitaciones de la versión inicial. Entre las nuevas funcionalidades destacan:

- Cada usuario puede visualizar únicamente sus grupos y acceder a ellos mediante invitaciones.
- Se permite la edición y borrado de gastos, incluyendo montos y participantes.
- Se muestra un balance detallado en cada grupo, indicando quién debe a quién y cuánto dinero.
- Se agregó un resumen general en la pantalla principal del usuario sobre sus deudas y créditos.
- Se implementó la generación de reportes filtrados por fechas, que pueden exportarse en CSV o PDF.

## 2. Decisiones de diseño

- Interfaz gráfica: Se mantuvo Windows Forms como tecnología para la capa de presentación, debido a su simplicidad y rapidez para construir interfaces de escritorio.
- Arquitectura: Se organizó el código siguiendo el patrón MVC para separar la lógica de negocio de la interfaz, facilitando la mantenibilidad y escalabilidad del sistema.
- Persistencia de datos: Se utilizan archivos JSON para almacenar usuarios, grupos y gastos, lo que permite simplicidad sin necesidad de una base de datos externa.
- Balance Calculator mejorado: Se extendió la funcionalidad para calcular no solo los saldos netos de cada usuario, sino también quién le debe a quién y cuánto dinero dentro de cada grupo.
- Gestión de grupos: Cada usuario solo puede ver y acceder a sus propios grupos mediante invitaciones, lo que mejora la privacidad y control.
- Gestión de gastos: Se agregó la opción de editar y borrar gastos existentes, modificando tanto montos como participantes.
- Reportes: Se incluyó la generación de reportes por usuario y por grupo, con posibilidad de filtrarlos por rangos de fechas y exportarlos en CSV (y PDF si aplica).
- Testing: Se realizaron pruebas unitarias sobre la lógica del negocio con el fin de garantizar la exactitud de los cálculos y la robustez del sistema.

- Buenas prácticas: Se aplicaron principios de POO, SOLID y Clean Code, utilizando nombres significativos, métodos con responsabilidad única, manejo de excepciones y reutilización de código.

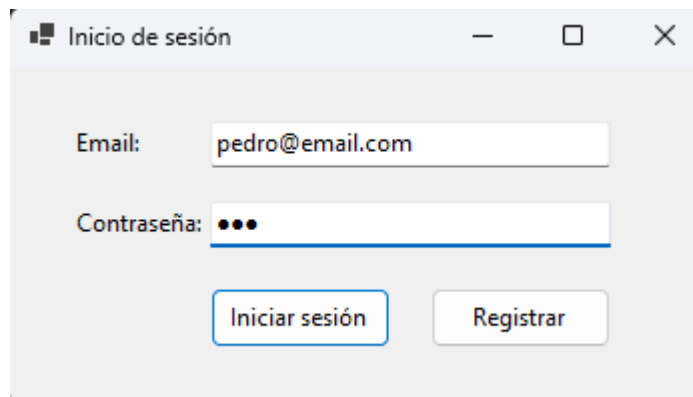
## 2.1. Cómo ejecutar el programa

1. Clonar o descargar el repositorio desde GitHub en la carpeta de “Proyecto #2”.
2. Abrir la solución en Visual Studio 2022 o superior.
3. Ejecutar el programar

## 2.2. Manual de usuario

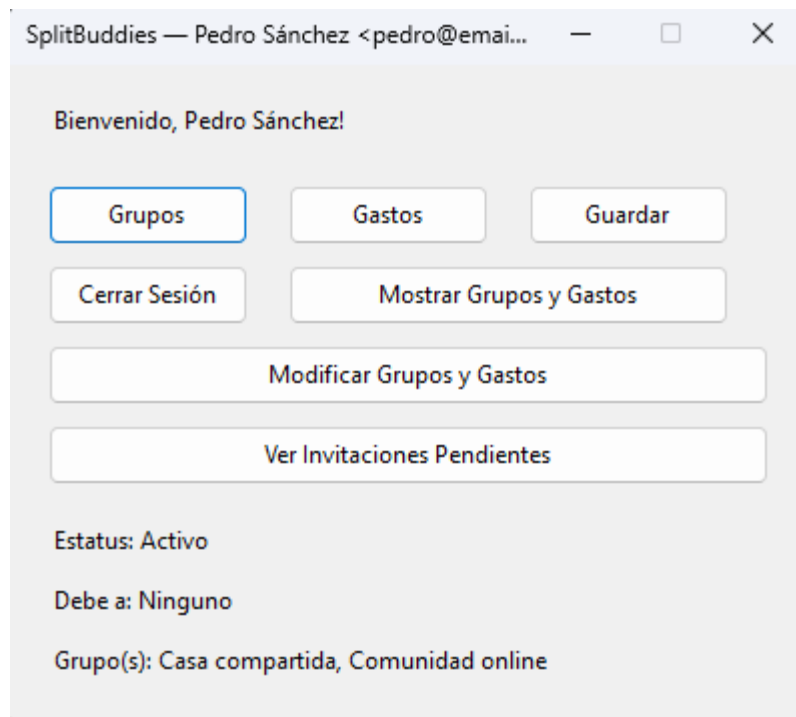
### a) Pantalla de Login

- Ingresar correo y contraseña.



The screenshot shows a login window titled "Inicio de sesión". It contains two input fields: "Email:" with the value "pedro@email.com" and "Contraseña:" with three dots indicating a password. Below the fields are two buttons: "Iniciar sesión" and "Registrar".

- Si son correctos, se abre el menú principal.



## b) Crear grupo

- Desde el menú, elegir Grupo.
- Asignar un nombre al grupo y opcionalmente una imagen.

Gestión de Grupos

Casa compartida  
Comunidad online

Nombre del grupo:  
Casa compartida

Ruta de imagen:  
casacompartida.jpg

Seleccionar imagen

Crear grupo

Eliminar grupo

Ver balances

Enviar

Desde: viernes , 22 de agosto ▼ Has viernes , 22 de agosto ▼ 

Generar reporte

Exportar CSV

d) Generar reporte

- Seleccionas una fecha desde a hasta y le das Generar reporte

Gestión de Grupos

Casa compartida

Comunidad online

Nombre del grupo:

Casa compartida

Ruta de imagen:

casacompartida.jpg

Crear grupo

Eliminar grupo

Seleccionar imagen

Ver balances

Enviar

Desde:

lunes , 1 de enerc

Has

ércoles, 31 de diciembre

Generar reporte

Exportar CSV

	Grupo	Descripción	Monto	Pagador	Usuario	Efecto	Fecha
▶	Casa com...	Compra e...	626,64	(Sin pagad...	(Sin pagad...	Recibe	5/07/2024
	Comunida...	Costo gru...	979,35	(Sin pagad...	(Sin pagad...	Recibe	15/07/2024
	Comunida...	Pago nece...	230,24	(Sin pagad...	(Sin pagad...	Recibe	15/07/2024
	Comunida...	Pago com...	242,27	(Sin pagad...	(Sin pagad...	Recibe	15/07/2024
	Comunida...	Parte del p...	638,44	(Sin pagad...	(Sin pagad...	Recibe	15/07/2024
	Comunida...	Gasto gen...	223,68	(Sin pagad...	(Sin pagad...	Recibe	15/07/2024
*							

podrás exportar el reporte en CSV

### c) Calcular balance

- Al abrir el grupo, selecciona un grupo y haz clic en ver Balance.



Balance del Grupo

	Descripción	Monto	Pagador	Usuario	Efecto	Fecha
	Compra entre todos	626,64	(Sin pagador)	(Sin pagador)	Recibe	5/07/2024
▶	Pago del agua	500,0	maria@email.com	maria@email.com	Recibe	22/08/2025 10:27 p. m.
	Pago del agua	500,0	maria@email.com	pedro@email.com	Debe	22/08/2025 10:27 p. m.
*						

	Email	Saldo (+ recibe / - debe)
▶	(Sin pagador)	626,64
	maria@email.com	500,0
	pedro@email.com	-500,0
*		

	De	Para	Monto
▶	pedro@email.com	maria@email.com	500,0
*			

Exportar CSV

Podrás ver toda la información y podrás exportar en CSV

#### d) Agregar gastos

- Seleccionar un grupo y registrar un gasto: grupo, nombre, monto, quien debe o recibe, descripción, quién paga y quiénes participan

**Agregar Gasto**

Grupo: Casa compartida

Pagado por: pedro@email.com

Miembros incluidos: ☒ pedro@email.com  
☒ maria@email.com

Nombre:

Descripción:

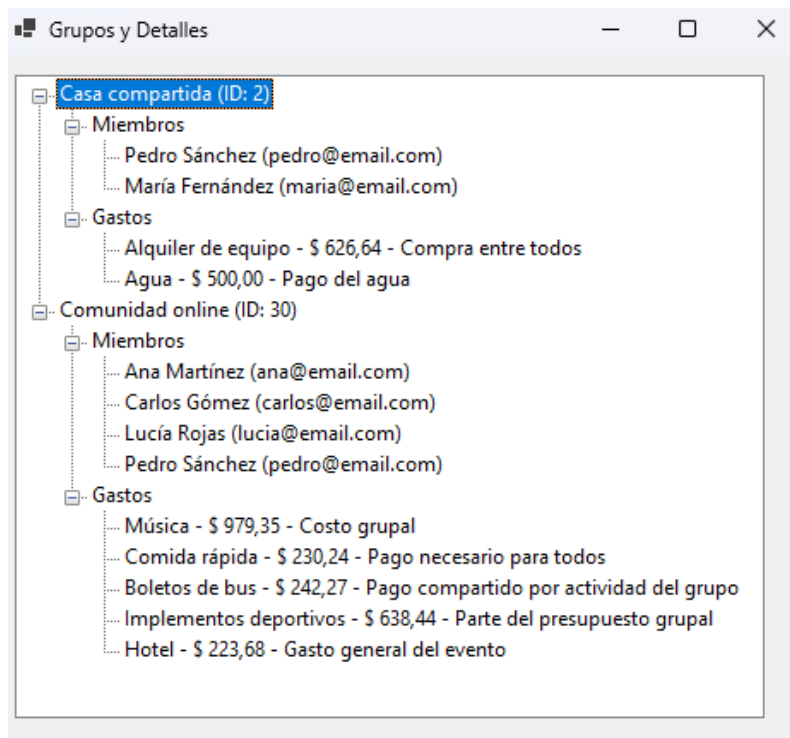
Monto:

Efecto: Debe

Agregar Gasto

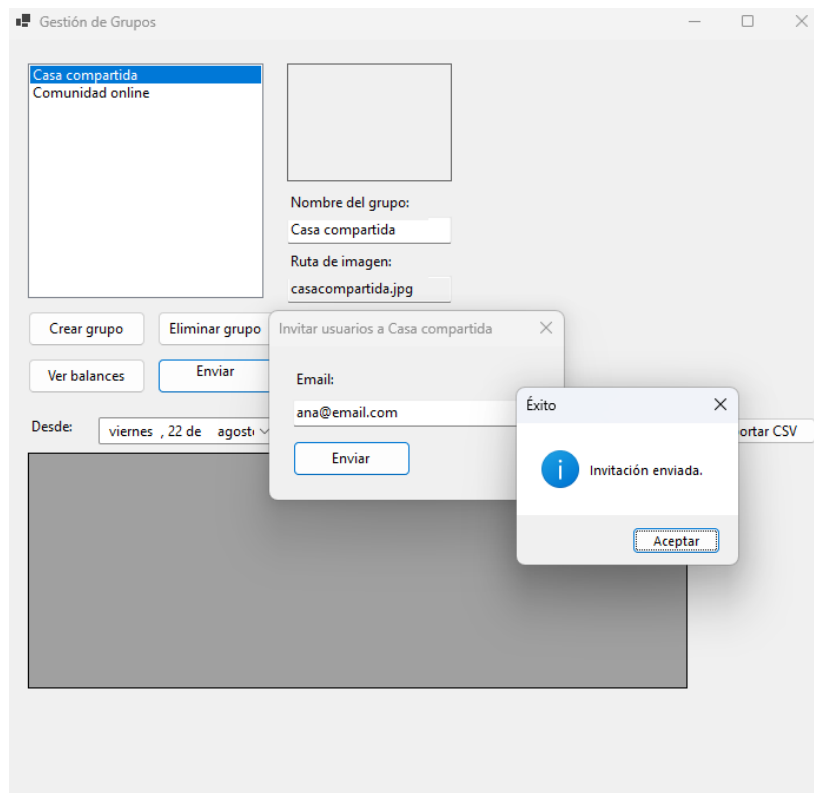
#### e) Mostrar Grupos y Gastos

- **podrás visualizar todos los grupos y gastos que pertenece el usuario en general**

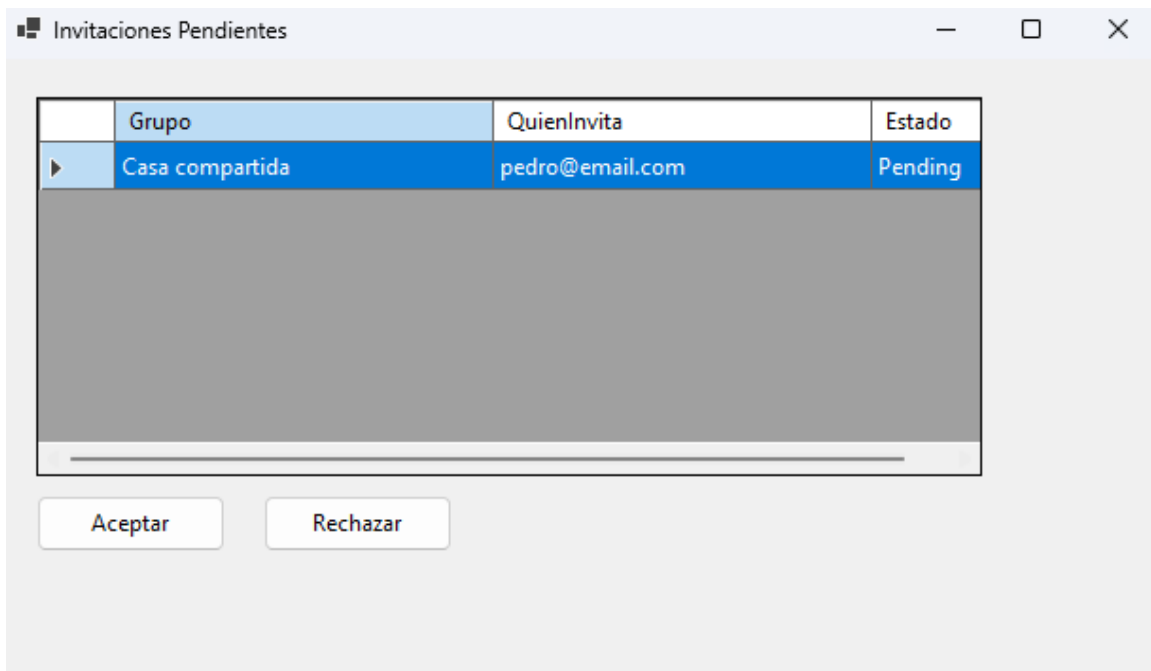


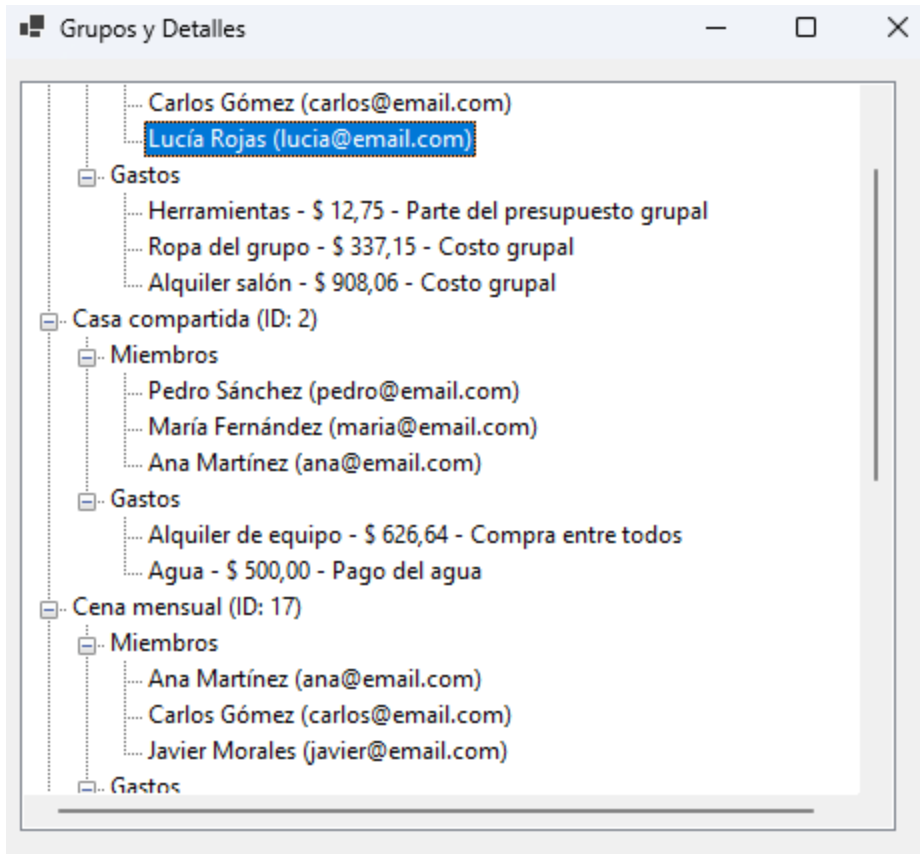
#### f) Invitaciones

- En grupos si le das al botón de enviar podrás enviar una solicitud a un usuario poniendo su correo



- Luego la persona invitada podrá visualizar la invitación en la pestaña de “Ver invitaciones pendientes” y podrá aceptarlo o rechazarlo





g) Editar grupos y gastos

- Pestaña donde podrás ver los grupos que esta en el usuario y poder editar o borrar miembros y gastos

Editar Grupos y Gastos

Viaje de verano

Casa compartida

Cena mensual

Comunidad online

SplitBuddies.Models.Expense

SplitBuddies.Models.Expense

Nombre del Grupo:  
Casa compartida

Miembros (separados por coma):  
pedro@email.com, maria@email.cor

Guardar Cambios

Alquiler de equipo

626,64

Compra entre todos

☐ pedro@email.com

☐ maria@email.com

☐ ana@email.com

Editar Gasto

Borrar Gasto

### 3. Análisis de resultados

El programa cumple con la mayoría de los objetivos establecidos para la versión 2:

- Gestión de grupos: Cada usuario únicamente puede ver y acceder a sus propios grupos mediante invitaciones, lo cual asegura mayor privacidad.
- Gestión de gastos: Se pueden registrar, editar y eliminar gastos, modificando montos y participantes de forma sencilla.

- Cálculo de balances: El sistema calcula automáticamente los saldos individuales, así como quién debe a quién y en qué grupo.
- Reportes: Se implementó la generación de reportes tanto a nivel de grupo como de usuario, con la posibilidad de filtrarlos por fechas y exportarlos en formato CSV.
- Pruebas unitarias: Se alcanzó una cobertura de pruebas mayor al 60% en la capa de lógica de negocio, validando principalmente el registro de gastos, el cálculo de balances y la exportación de datos.
- Documentación del código: Se generó documentación automática utilizando Doxygen, lo que permite contar con una referencia clara de las clases, métodos y relaciones del sistema, mejorando la mantenibilidad y escalabilidad.

**A pesar de estos avances, se identificaron algunas áreas de mejora:**

- Gestión de contraseñas: Actualmente se almacenan de forma básica; se recomienda implementar encriptación y validaciones más robustas.
- Invitaciones de grupos: Aunque los usuarios reciben invitaciones, estas podrían mejorarse con confirmación explícita o notificaciones más dinámicas.
- Interfaz gráfica: La aplicación cumple con su propósito, pero la experiencia de usuario podría enriquecerse con un diseño más moderno y fluido.
- Multi-idioma: Actualmente la aplicación está en un único idioma; implementar soporte multilingüe ampliaría su alcance.

#### 4. Conclusiones y aprendizajes

El desarrollo de este proyecto permitió aplicar conocimientos de programación en C# y Windows Forms, integrando conceptos de manejo de datos, eventos y control de errores. La experiencia adquirida no solo mejoró las habilidades técnicas, sino también la capacidad de resolver problemas de manera autónoma y de diseñar aplicaciones prácticas y funcionales.

- Se aprendió a trabajar con Windows Forms en C#, creando interfaces gráficas funcionales e interactivas.
- Se aplicó la serialización JSON para la persistencia de datos.
- El proyecto mostró la importancia de planificar antes de programar, lo que ayuda a evitar errores de integración y facilita la organización del código.
- Se comprendió la importancia de escribir código limpio, modular y reutilizable, siguiendo buenas prácticas de programación.
- Se mejoró la capacidad de diseñar interfaces amigables y funcionales, enfocadas en la experiencia del usuario.
- Se adquirió experiencia en la gestión y manipulación de datos, usando archivos JSON de manera eficiente.
- Se fortaleció la capacidad de aprendizaje autónomo, investigando y aplicando nuevas herramientas según las necesidades del proyecto.
- Se generó la **documentación técnica con Doxygen**, permitiendo mantener un registro estructurado del código y facilitando su comprensión para futuros mantenimientos y mejoras.