

Universidad Politécnica Internacional

Curso: Técnicas de Programación

Profesor: Luis Felipe Mora Umaña

Proyecto 1

SplitBuddies

Estudiantes

Daniel Benavides

Freddy Martinez Espinoza

II Cuatrimestre, 2025

Índice

1. Introducción	4
2. Decisiones de diseño	5
2.1. Cómo usar y ejecutar el programa	5
2.2. Capturas de pantalla con explicación de interacciones	6
3. Análisis de resultados.....	10
4. Conclusiones y aprendizaje	11

Índice de Ilustraciones

Ilustración 1. Registro de usuario: Lista de usuarios disponibles para seleccionar o crear uno nuevo.	6
Ilustración 2. Panel de grupos: Visualización y creación de grupos.	7
Ilustración 3. Formulario de gasto: Registro de nombre, monto y participantes del gasto.	8
Ilustración 4. Vista de balances: Balance de gastos y deudas entre los miembros del grupo.	9

1. Introducción

SplitBuddies es una aplicación de escritorio desarrollada en C# utilizando el paradigma de programación orientada a objetos y el patrón MVC.

Su objetivo es permitir a los usuarios gestionar gastos compartidos dentro de grupos, mostrando balances, estadísticas y reportes de gastos. Esta es la primera iteración del sistema y sienta la base para futuras ampliaciones.

2. Decisiones de diseño

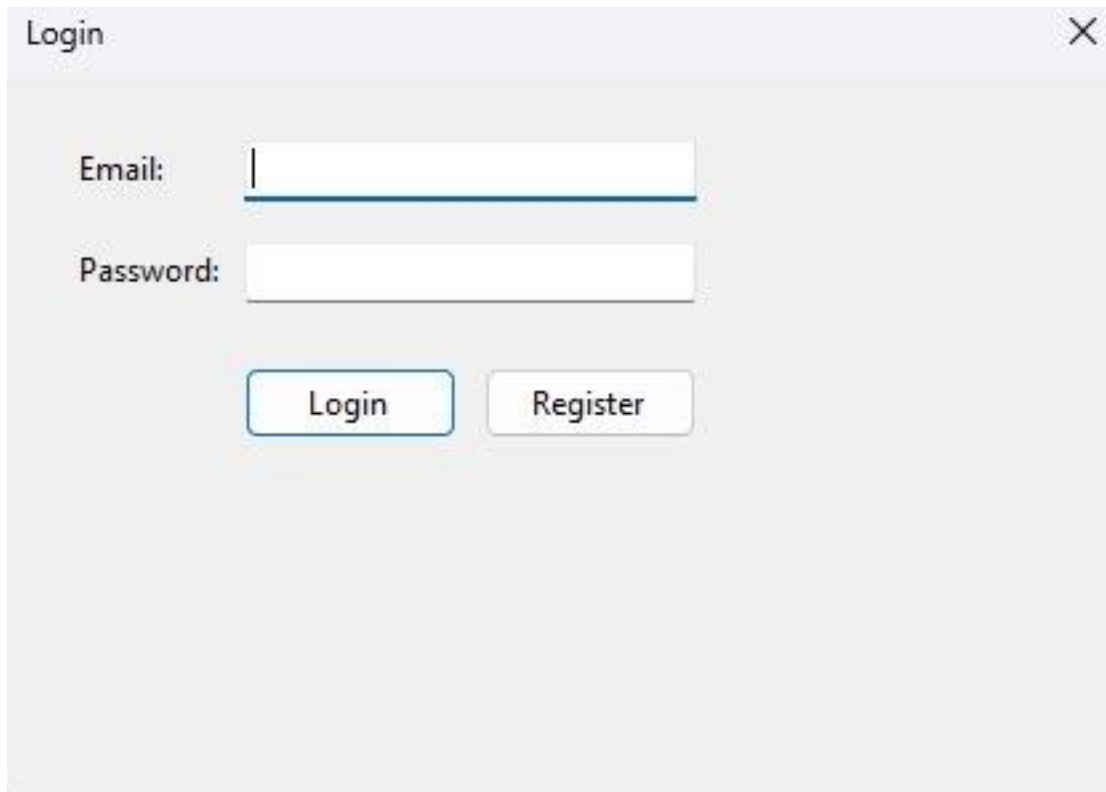
Entre las decisiones de diseño se encuentran los siguientes:

- Lenguaje: C# con Windows Forms
- Arquitectura: Patrón MVC
- Persistencia: Archivos JSON
- Extensibilidad: Código modular preparado para futuras iteraciones

2.1. Cómo usar y ejecutar el programa

- Abrir el proyecto en Visual Studio
- Ejecutar la solución (F5)
- Registrarse como usuario nuevo
- Crear o unirse a un grupo
- Agregar gastos
- Consultar balances desde el panel de estadísticas
- Explorar reportes filtrando por fechas

2.2. Capturas de pantalla con explicación de interacciones



A screenshot of a 'Login' dialog box. The dialog has a title bar with the word 'Login' on the left and a close button (an 'X' icon) on the right. Inside the dialog, there are two input fields. The first is labeled 'Email:' and has a text input field with a vertical cursor. The second is labeled 'Password:' and has a password input field. Below these fields are two buttons: 'Login' and 'Register'. The 'Login' button is highlighted with a blue border, while the 'Register' button has a grey border. The background of the dialog is light grey.

Ilustración 1. Registro de usuario: Lista de usuarios disponibles para seleccionar o crear uno nuevo.



A screenshot of a 'Register' dialog box. The title bar at the top is light gray and contains the word 'Register' on the left and a close button (an 'X' icon) on the right. The main content area has a light gray background. At the top of this area, the text 'Registro de usuario' is centered in a bold, black font. Below this, there are four labels on the left, each followed by a white rectangular input field: 'Nombre:', 'Correo electrónico:', 'Contraseña:', and 'Confirmar'. The 'Nombre:' field has a blue horizontal line at its bottom. At the bottom of the dialog, there are two buttons: 'Registrar' and 'Cancelar', both with rounded corners and a light gray background.

Register

Registro de usuario

Nombre:

Correo electrónico:

Contraseña:

Confirmar

Registrar Cancelar

Ilustración 2. Panel de grupos: Visualización y creación de grupos.

The image shows a software window titled "Agregar Gasto" (Add Expense). It contains the following fields:

- Grupo:** A dropdown menu.
- Pagado por:** A dropdown menu.
- Miembros incluidos:** A large text area for listing members.
- Nombre:** A text input field.
- Descripción:** A text input field.
- Monto:** A text input field.

At the bottom of the form is a button labeled "Agregar Gasto".

Ilustración 3. Formulario de gasto: Registro de nombre, monto y participantes del gasto.

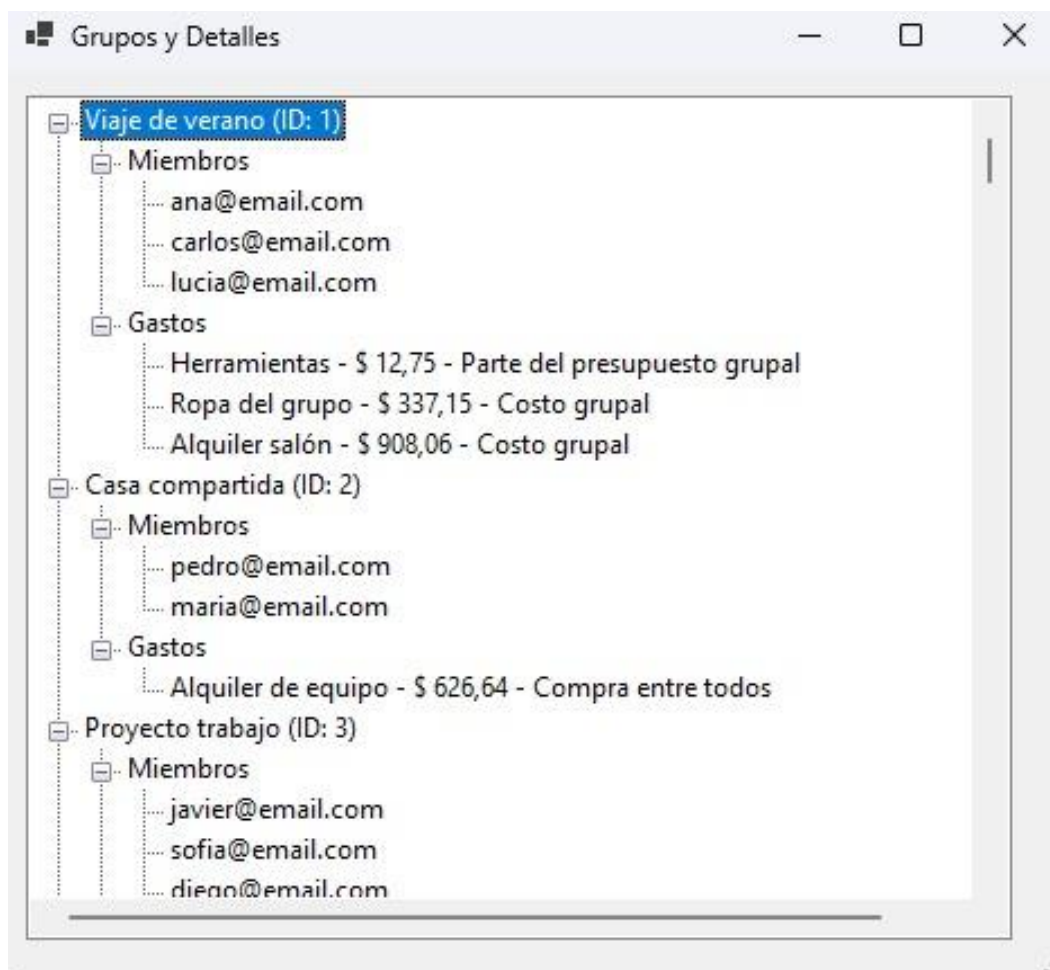


Ilustración 4. Vista de balances: Balance de gastos y deudas entre los miembros del grupo.

3. Análisis de resultados

Se logra implementar de forma exitosa la base del sistema, incluyendo usuarios, grupos y gastos. El sistema calcula correctamente los balances por grupo y por usuario. La estructura del código sigue principios de POO y Clean Code.

4. Conclusiones y aprendizaje

El desarrollo de este proyecto nos permitió aplicar principios de diseño y programación orientada a objetos, además del uso práctico del patrón MVC.

Con la elaboración de este proyecto se aprende a trabajar en equipo a estructurar el proyecto desde el inicio y mantener un código limpio, reutilizable y mantenible, utilizando las herramientas necesarias.