



Plan de Pruebas

CuentiApp

Versión: 1.0

Fecha: 13/07/2025

Autor: Matías Giambra

Tutor: Prof. Cristian Ciarallo

Release: Julio 2025

Tabla de contenidos

1. [Objetivo del Testing](#)
2. [Alcance del Testing](#)
3. [Herramientas para el Testing](#)
4. [Plan de Ejecución](#)
 - 4.1. [Configurar entorno de pruebas](#)
 - 4.2. [Definir casos de prueba clave](#)
 - 4.3. [Ejecutar pruebas manuales e integradas](#)
 - 4.4. [Registrar resultados](#)
5. [Plan de pruebas adicionales](#)

Plan de Pruebas: CuentiApp

1 - Objetivo del Testing:

Validar el correcto funcionamiento de las funcionalidades principales de CuentiApp, tanto en frontend como backend, verificando que las reglas de negocio y restricciones se cumplan, y que el sistema responda correctamente ante diferentes tipos de entrada.

2 - Alcance del Testing:

- Pruebas de endpoints de la API (registro, login, gastos, categorías, cuentas, grupo familiar).
- Pruebas de validación de formularios en el frontend.
- Verificación de respuestas esperadas ante errores y acciones exitosas.
- Pruebas de flujo de usuario: desde el registro hasta la visualización de estadísticas.

3 - Herramientas para el Testing:

- ✓ Pruebas individuales
 - Validación de cada endpoint de forma aislada.
 - Comprobación de mensajes de error y respuestas esperadas.
 - Pruebas de campos requeridos y tipos de datos.
- ✓ Pruebas funcionales integradas
 - Flujo completo de registro → login → crear cuentas → agregar gastos → ver estadísticas.
 - Interacción entre módulos: gastos afectan saldo de cuentas, se agrupan por categoría, etc.
- ✓ Pruebas de restricciones
 - Intento de registrar gasto con saldo insuficiente.
 - Agregar más de 1 invitado al grupo familiar.
 - Registrar usuario con email duplicado.

4 - Plan de Ejecución:

1. Configurar entorno de pruebas
 - BD en memoria para backend.
 - Mock localStorage y rutas en frontend.
2. Definir casos de prueba clave

Flujo completo de usuario.

Validaciones por campo.

Restricciones funcionales.

3. Ejecutar pruebas manuales e integradas

Frontend: pruebas visuales + comandos npm run test

Backend: correr test con npm test

4. Registrar resultados

Documentar errores, comportamientos inesperados, regresiones.

Repetir pruebas tras cada refactor o corrección.

5 - Plan de Pruebas Adicionales (Opcionales):

- Pruebas en dispositivos móviles.
- Pruebas de rendimiento básico (tiempo de respuesta de API).
- Validación con usuarios externos simulados.