

Documentación de Pruebas Unitarias del Frontend Sistema Descansos del Recuerdo SPA

César Rojas Ramos

Lucas Moncada

Profesor: Viviana Poblete

26 de noviembre de 2025

Índice

1. Introducción	3
2. Resumen de Ejecución de Pruebas	3
3. Detalle de Archivos Probados	3
3.1. Archivo <code>validators.js</code>	3
3.2. Archivo <code>AlertBadge.jsx</code>	4
4. Métricas de Cobertura	5
5. Conclusiones	5

1. Introducción

El presente documento describe las pruebas unitarias desarrolladas para el frontend del sistema *Descansos del Recuerdo SPA*. Las pruebas fueron realizadas con las herramientas **Vitest** y **@testing-library/react**, evaluando módulos críticos del flujo de usuario.

Toda la información incluida en este documento proviene del reporte de pruebas entregado por el estudiante, el cual contiene el número total de pruebas, cobertura, archivos probados y comportamientos verificados.

2. Resumen de Ejecución de Pruebas

Las pruebas fueron ejecutadas con el comando:

```
npm run test -- --coverage
```

El sistema reportó lo siguiente:

- **Archivos de prueba ejecutados:** 2
- **Total de pruebas:** 14
- **Resultado:** 14 pruebas aprobadas correctamente
- **Cobertura:** 100 % en:
 - Sentencias
 - Ramas
 - Funciones
 - Líneas

Los archivos sometidos a prueba fueron:

- `src/utils/validators.js`
- `src/components/common/AlertBadge.jsx`

3. Detalle de Archivos Probados

A continuación, se detalla el funcionamiento validado de cada módulo del frontend.

3.1. Archivo `validators.js`

Descripción

Este archivo contiene las funciones de validación usadas en formularios del sistema: login, registro, direcciones, gestión de urnas y validaciones administrativas.

Archivo de prueba asociado

```
src/utils/validators.test.js
```

Funciones probadas

Las pruebas verifican el correcto comportamiento de:

- `isValidEmail`
- `isValidPrice`
- `isValidStock`
- `isValidPassword`
- `isPositiveNumber`
- `isValidName`
- `isValidAddress`
- `isValidPhone`
- `isRequired`
- `validateProductForm`

Resultados

- Todas las funciones se comportan según lo esperado.
- No se detectaron errores de validación.
- Cobertura del archivo: **100 %**.

3.2. Archivo `AlertBadge.jsx`

Descripción

Componente visual encargado de mostrar el estado del stock mediante un distintivo (*badge*) dependiendo del valor disponible.

Archivo de prueba asociado

`src/components/common/AlertBadge.test.jsx`

Comportamientos validados

Las pruebas verifican que:

- Con stock igual a 0: se muestra “Agotado”.
- Con stock menor al umbral (por ejemplo, 3): se muestra “Stock bajo”.
- Con stock suficiente: se muestra “Stock: X” con la clase correcta.

Resultados

- Todos los escenarios se renderizaron correctamente.
- No se encontraron errores de interfaz.
- Cobertura del archivo: **100 %**.

4. Métricas de Cobertura

El siguiente reporte corresponde a la ejecución del comando con cobertura:

File	%Stmts	%Branch	%Funcs	%Lines
All files	100	100	100	100
components/common	100	100	100	100
AlertBadge.jsx	100	100	100	100
utils	100	100	100	100
validators.js	100	100	100	100

Este reporte confirma:

- Todas las rutas de ejecución fueron cubiertas.
- No existen líneas o funciones sin probar.
- Toda la lógica crítica del frontend está completamente validada.

5. Conclusiones

Las pruebas unitarias del frontend alcanzan un nivel de calidad excelente:

- Se cubren módulos críticos tanto de lógica como de presentación.
- Se logra 100 % de cobertura en todos los indicadores.
- No se encontraron fallos en las funciones evaluadas.
- Se garantiza estabilidad en el flujo principal del usuario.

Como recomendación, las pruebas deben extenderse conforme se añadan nuevas validaciones o componentes relevantes para mantener un estándar de calidad elevado.