# 📄 Documento de Pruebas del Aplicativo

## 1. Introducción

Este documento define el plan de pruebas del aplicativo descrito en las historias de usuario. Su objetivo es asegurar la calidad, funcionalidad y rendimiento del sistema mediante la aplicación de pruebas estructuradas, sistemáticas y documentadas.

## 2. Objetivos del Testing

- Validar que el sistema cumpla con las funcionalidades esperadas.  
- Detectar defectos antes de la puesta en producción.  
- Garantizar la estabilidad, usabilidad y seguridad del sistema.  
- Verificar el cumplimiento de los criterios de aceptación de las historias de usuario.

## 3. Tipos de pruebas a aplicar

|  |  |
| --- | --- |
| Tipo de prueba | Descripción |
| Pruebas funcionales | Verifican que cada historia de usuario se comporta según lo esperado. |
| Pruebas de regresión | Aseguran que nuevas funcionalidades no afecten las ya existentes. |
| Pruebas de rendimiento | Evalúan tiempos de respuesta, carga simultánea y consumo de recursos. |
| Pruebas de seguridad | Verifican que no existan vulnerabilidades críticas. |
| Pruebas de usabilidad | Evalúan la experiencia del usuario en flujos clave. |
| Pruebas de aceptación | Validación final con usuarios clave (administradores, docentes, estudiantes). |

## 4. Ambientes de prueba

- Ambiente de desarrollo: Para pruebas preliminares.  
- Ambiente de pruebas (QA): Réplica controlada del entorno de producción.  
- Ambiente de producción: Solo para pruebas de rendimiento si se cuenta con un entorno espejo.

## 5. Herramientas recomendadas

- Postman: para pruebas de API.  
- Selenium / Cypress: para pruebas automáticas de interfaz.  
- JMeter: para pruebas de carga y rendimiento.  
- Trello / Jira / TestLink: para seguimiento de casos de prueba.  
- Google Forms / Excel: para validaciones manuales simples.

## 6. Casos de prueba funcionales

Ejemplos de casos de prueba por funcionalidad:

|  |  |  |
| --- | --- | --- |
| Funcionalidad | Caso de prueba | Resultado esperado |
| Registro de usuario | Registrar con correo único | Cuenta creada + correo enviado |
| Registro de usuario | Registrar con correo repetido | Mostrar error: "Correo ya en uso" |
| Inicio de sesión | Ingresar con credenciales correctas | Acceso al sistema |
| Inicio de sesión | Ingresar con credenciales inválidas | Error: "Credenciales inválidas" |
| Recuperar contraseña | Solicitar restablecimiento con correo válido | Enlace enviado |
| Recuperar contraseña | Usar enlace vencido | Mostrar error: enlace expirado |
| Inscripción a diplomado | Inscripción con cupo disponible | Inscripción exitosa |
| Inscripción a diplomado | Inscripción cuando cupo lleno | Notificación de no disponibilidad |

## 7. Criterios de Aceptación para el Testing

- Ejecución exitosa del flujo principal  
- Mensajes de error apropiados en caso de entradas inválidas  
- Validación de datos obligatorios  
- Actualización/creación/consulta con consistencia en base de datos  
- Flujo de confirmación para acciones importantes (ej. eliminar, inactivar)

## 8. Ciclos de prueba sugeridos

|  |  |  |
| --- | --- | --- |
| Ciclo | Objetivo | Responsable |
| 1 | Pruebas unitarias | Desarrollador |
| 2 | Pruebas funcionales (QA) | Equipo de QA |
| 3 | Pruebas de aceptación (UAT) | Usuario final |
| 4 | Pruebas de regresión | QA / Automatización |
| 5 | Pruebas de carga | QA (JMeter) |

## 9. Roles involucrados

|  |  |
| --- | --- |
| Rol | Actividades principales |
| Tester QA | Ejecutar casos de prueba, documentar errores |
| Desarrollador | Corregir defectos y apoyar validaciones técnicas |
| Administrador | Validar flujos críticos de administración |
| Docente | Validar funcionalidad de foros, contenidos, clases |
| Estudiante | Validar flujos de inscripción, pagos, acceso |