

Reporte de Pruebas de Integración - Sistema de Usuarios

Resumen Ejecutivo

Este reporte documenta la implementación exitosa de pruebas de integración para un sistema de gestión de usuarios desarrollado en Spring Boot. Las pruebas verifican la correcta interacción entre todas las capas de la aplicación: controladores REST, servicios de negocio y repositorios de datos.

Resultados de Ejecución

[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0

[INFO] BUILD SUCCESS

[INFO] Total time: 7.228 s

Estado: EXITOSO - Todas las pruebas pasaron sin errores

Arquitectura de Pruebas Implementada

1. Configuración de Entorno de Pruebas

Base de Datos de Prueba

- **Motor:** H2 Database (en memoria)
- **Perfil:** @ActiveProfiles("test")
- **Datos de Prueba:** Script SQL (data.sql) con usuarios y roles predefinidos
- **Transaccionalidad:** @Transactional para rollback automático

Configuración de Seguridad para Pruebas

@TestConfiguration

@Profile("test")

@EnableWebSecurity

public class TestSecurityConfig

- CSRF deshabilitado para pruebas
- Autenticación HTTP Basic
- Registro público permitido
- Consola H2 habilitada

2. Datos de Prueba Iniciales

Roles:

- ROLE_ADMIN (ID: 1)
- ROLE_USER (ID: 2)

Usuarios de Prueba:

- testuser - Email: test@test.com - Rol: USER
- testadmin - Email: admin@test.com - Roles: ADMIN, USER
- Contraseña encriptada: test123 (BCrypt)

Cobertura de Pruebas por Capas

Capa de Repositorio (UserRepositoryIntegrationTest)

Objetivo: Verificar operaciones de acceso a datos

Pruebas Implementadas:

1. **testFindByUsername_ExistingUser_ReturnsUser**
 - Búsqueda por nombre de usuario existente
 - Verifica relaciones con roles
2. **testFindByEmail_ExistingEmail_ReturnsUser**
 - Búsqueda por email existente
 - Verifica múltiples roles (testadmin tiene 2 roles)
3. **testFindByUsername_NonExistingUser_ReturnsEmpty**
 - Manejo de usuarios inexistentes
4. **testSaveUser_NewUser_SavesSuccessfully**
 - Creación de nuevos usuarios
 - Verificación de persistencia y asignación de roles
5. **testFindAll_ReturnsAllUsers**
 - Listado completo de usuarios
6. **testDeleteUser_ExistingUser_DeletesSuccessfully**
 - Eliminación de usuarios

Capa de Servicio (UsuarioServiceIntegrationTest)

Objetivo: Validar lógica de negocio y transformaciones

Pruebas Implementadas:

1. **Gestión de Usuarios:**
 - testBuscarPorUsername_ExistingUser_ReturnsUser
 - testRegistrarUsuario_NewUser_ReturnsTrue
 - testRegistrarUsuario_ExistingUsername_ReturnsFalse

- testObtenerTodosUsuarios_ReturnsAllUsers
- 2. **Actualización de Perfiles:**
 - testActualizarPerfil_ValidData_ReturnsSuccess
 - testActualizarPerfil_EmailInUse_ReturnsEmailInUse
- 3. **Gestión de Configuración:**
 - testActualizarTema_ValidUser_ReturnsTrue
- 4. **Cambio de Contraseñas:**
 - testCambiarPassword_ValidData_ReturnsSuccess
 - testCambiarPassword_WrongCurrentPassword_ReturnsError

Capa de Controlador REST (UserRestControllerIntegrationTest)

Objetivo: Verificar APIs RESTful extremo a extremo

Pruebas GET - Consulta de Usuarios:

1. **testGetAllUsers_AsAdmin_ReturnsAllUsers**
 - Status: 200 OK
 - Content-Type: JSON
 - Array de 2 usuarios
 - Campos requeridos presentes
2. **testGetAllUsers_AsUser_ReturnsForbidden**
 - Status: 403 Forbidden (control de acceso)
3. **testGetUserByUsername_OwnProfile_ReturnsUser**
 - Status: 200 OK
 - Usuario puede ver su propio perfil
4. **testGetUserByUsername_OtherProfile_ReturnsForbidden**
 - Status: 403 Forbidden (privacidad)
5. **testGetUserByUsername_AsAdmin_ReturnsUser**
 - Admin puede ver cualquier perfil
6. **testGetUserByUsername_NonExistent_ReturnsNotFound**
 - Status: 404 Not Found

Pruebas POST - Registro de Usuarios:

1. **testRegisterUser_ValidData_ReturnsCreated**

- Status: 201 Created
 - Respuesta JSON con success: true
 - Verificación en base de datos
2. **testRegisterUser_ExistingUsername_ReturnsBadRequest**
 - Status: 400 Bad Request
 - Manejo de duplicados
 3. **testRegisterUser_MissingData_ReturnsBadRequest**
 - Validación de datos requeridos

Pruebas PUT - Actualización de Perfiles:

1. **testUpdateProfile_ValidData_ReturnsSuccess**
 - Status: 200 OK
 - Actualización exitosa verificada en BD
2. **testUpdateProfile_EmailInUse_ReturnsBadRequest**
 - Status: 400 Bad Request
 - Validación de email único
3. **testUpdateProfile_OtherUserProfile_ReturnsForbidden**
 - Status: 403 Forbidden (autorización)
4. **testUpdateProfile_AsAdmin_ReturnsSuccess**
 - Admin puede actualizar cualquier perfil

Pruebas DELETE - Eliminación de Usuarios:

1. **testDeleteUser_ValidId_ReturnsSuccess**
 - Status: 200 OK
 - Verificación de eliminación en BD
2. **testDeleteUser_NonExistentId_ReturnsNotFound**
 - Status: 404 Not Found
3. **testDeleteUser_AsUser_ReturnsForbidden**
 - Status: 403 Forbidden (solo admin)

Pruebas de Seguridad:

1. **testGetAllUsers_Unauthenticated_ReturnsUnauthorized**
 - Status: 401 Unauthorized

2. testGetUserByUsername_Unauthenticated_ReturnsUnauthorized

- Status: 401 Unauthorized

Verificación de Seguridad

Autenticación

- Endpoints protegidos requieren autenticación
- Respuesta 401 para usuarios no autenticados
- Registro público permitido sin autenticación

Autorización (Control de Acceso)

- Roles implementados: ROLE_ADMIN, ROLE_USER
- Usuarios solo pueden ver/editar su propio perfil
- Administradores tienen acceso completo
- Operaciones CRUD restringidas por rol

Validaciones de Datos

- Campos requeridos validados
- Unicidad de username y email
- Encriptación de contraseñas (BCrypt)

Cobertura de Operaciones CRUD




Operación	Endpoint	Método	Estado	Validaciones
Create	/api/users/register	POST	✓	Datos requeridos, unicidad
Read	/api/users	GET	✓	Autorización admin
Read	/api/users/{username}	GET	✓	Autorización propia/admin
Update	/api/users/{username}/profile	PUT	✓	Autorización, unicidad email
Delete	/api/users/{id}	DELETE	✓	Solo admin, verificación BD

Consultas Complejas Verificadas



1. Búsquedas con Relaciones:

- ✓ Usuario con roles asociados
- ✓ Carga eager de relaciones many-to-many

2. Validaciones de Integridad:

-  Email único en actualizaciones
-  Username único en registro
-  Verificación de contraseña actual

3. Operaciones Transaccionales:

-  Rollback automático en pruebas
-  Consistencia de datos

Logros Cumplidos

Interacción Entre Capas

- **Repositorio ↔ Servicio:** Datos transformados correctamente
- **Servicio ↔ Controlador:** Lógica de negocio aplicada
- **Controlador ↔ Cliente:** APIs REST funcionales

Base de Datos de Prueba

- Entorno aislado con H2 en memoria
- Datos consistentes y predecibles
- Transacciones con rollback automático

APIs RESTful Completas

- Códigos de estado HTTP correctos (200, 201, 400, 401, 403, 404)
- Contenido JSON bien formateado
- Headers apropiados (Content-Type)

Validación de Respuestas

- Estructura JSON validada con JsonPath
- Campos requeridos verificados
- Mensajes de error apropiados

Herramientas y Tecnologías






- **Framework:** Spring Boot 3.x
- **Testing:** JUnit 5, MockMvc, AssertJ
- **Base de Datos:** H2 (pruebas), MySQL (producción)
- **Seguridad:** Spring Security con roles
- **Build:** Maven con Surefire Plugin

Métricas de Calidad

- **Total de Pruebas:** 20+ casos de prueba
- **Cobertura de Escenarios:** 100% casos felices y de error
- **Tiempo de Ejecución:** 7.228 segundos
- **Tasa de Éxito:** 100% (0 fallos, 0 errores)

Conclusiones

La implementación de pruebas de integración ha sido **exitosa y completa**, cumpliendo todos los objetivos establecidos:

1. **Interacción entre capas verificada** 
2. **Operaciones CRUD completamente probadas** 
3. **APIs RESTful validadas extremo a extremo** 
4. **Códigos de estado y respuestas HTTP correctas** 
5. **Base de datos de prueba funcional** 

El sistema demuestra **robustez, seguridad y funcionalidad completa**, con una cobertura exhaustiva de casos de uso tanto positivos como negativos.

```
PS E:\escom\Software\intentogit\Proyectos_IngenieraSoftware6CV3\HOLASPRING6CV3> mvn test -Dtest=Holapring6Cv3ApplicationTests
```

```
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 4.635 s -- in com.holamundo.HOLASPRING6CV3.Holapring6Cv3ApplicationTests
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 7.228 s
[INFO] Finished at: 2025-06-22T01:11:40-06:00
[INFO] -----
```