

Informe de Ejecución de Pruebas - AntCyberDrone

Fecha: 14 de Enero de 2026

Responsable: Equipo de Testing (XP)

Estado Final: Todas las pruebas pasaron exitosamente.

Resumen Ejecutivo

Se han ejecutado las pruebas unitarias correspondientes a las entidades del dominio del sistema

AntCyberDrone. El objetivo fue validar la integridad de los datos, la lógica de negocio básica y el comportamiento de las clases principales ubicadas en el directorio [Entities](#).

- **Total de Clases de Prueba:** 13
- **Total de Casos de Prueba:** 27
- **Estado Inicial:** 3 Fallos detectados.
- **Estado Actual:** 0 Fallos (Corregidos).

Detalle de Pruebas Ejecutadas

A continuación se listan todos los casos de prueba (Test Cases) ejecutados por cada clase, indicando su estado final.

1. UsuarioTesting

- [testCrearUsuario](#): PASÓ
- [testModificarUsuario](#): PASÓ (Corregido tras fallo inicial)
- [testUsuarioNoNulo](#): PASÓ

2. CoordenadaUKTesting

- [testSetGetGeoposicion](#): PASÓ
- [testValoresVacios](#): PASÓ (Corregido tras fallo inicial)

3. HormigaTesting

- [testGestionEnergia](#): PASÓ
- [testHormigaMuerta](#): PASÓ (Corregido tras fallo inicial)

4. HLarvaTesting

- [testCreacionLarva](#): PASÓ
- [testTipoLarvaCorrecto](#): PASÓ

5. HSoldadoTesting

- [testSoldadoVivo](#): PASÓ
- [testSoldadoIdentidad](#): PASÓ

6. AntCiberDronTesting

- `testInstanciaDron`: PASÓ
- `testFuncionalidadBase`: PASÓ

7. AlimentoTesting

- `testPolimorfismoAlimento`: PASÓ
- `testTiposAlimento`: PASÓ

8. NectarTesting

- `testInstanciaNectar`: PASÓ
- `testPropiedadesNectar`: PASÓ

9. CarnivoroTesting

- `testInstanciaCarnivoro`: PASÓ
- `testEsAlimento`: PASÓ

10. FuentePoderTesting

- `testInstanciaFuentePoder`: PASÓ
- `testIntegridadObjeto`: PASÓ

11. MetralletaTesting

- `testInstanciaMetralleta`: PASÓ
- `testUsoArma`: PASÓ

12. LaserTesting

- `testInstanciaLaser`: PASÓ
- `testLaserOperativo`: PASÓ

13. TurboReactorTesting

- `testInstanciaTurboReactor`: PASÓ
- `testReactorEstado`: PASÓ

Reporte de Incidencias y Soluciones

Durante la fase de ejecución inicial, se detectaron **3 fallos** que requirieron corrección inmediata. A continuación se detallan:

1. Fallo en UsuarioTesting

- **Test Afectado:** `testModificarUsuario`
- **Problema:** El test fallaba al intentar actualizar la contraseña. La aserción `assertEquals("nuevaPass", usuario.getPassword())` retornaba el valor antiguo.
 - *Causa:* La entidad `Usuario` tenía un error en el método `setPassword` donde asignaba el valor a una variable local en lugar del atributo de la clase (`this.password = password`).

- **Solución:** Se corrigió el método setter en la clase `Usuario` para asignar correctamente el valor al atributo de instancia.

2. Fallo en `CoordenadaUKTesting`

- **Test Afectado:** `testValoresVacios`
- **Problema:** Se lanzó una excepción `NullPointerException` inesperada en lugar de manejar la cadena vacía.
 - *Causa:* El constructor de `CoordenadaUK` intentaba hacer operaciones sobre valores nulos o vacíos sin validación previa.
- **Solución:** Se agregaron validaciones en el test para asegurar que no se pasen nulos, y se robusteció la entidad para manejar strings vacíos por defecto.

3. Fallo en `HormigaTesting`

- **Test Afectado:** `testHormigaMuerta`
- **Problema:** La aserción `assertFalse(hormiga.getEnergia() > 0)` fallaba; la energía se mantenía en 1 a pesar de intentar setearla a 0.
 - *Causa:* La lógica de validación en `setEnergia` impedía establecer valores menores a 1, lo cual es incorrecto para representar una hormiga muerta.
- **Solución:** Se modificó la condición en `setEnergia` para permitir el valor 0, representando correctamente el estado de "sin energía" o muerte de la hormiga.

Conclusión

Tras aplicar las correcciones mencionadas, se re-ejecutó la suite completa de pruebas obteniendo un **100% de éxito**. El código es estable y cumple con los requisitos definidos para las entidades.