

TEST REPORT - hexagonal-architecture-demo

Date: 2026-01-10 13:03:29

Command: ./mvnw test

Summary

Total tests: 146

Passed: 146

Failures: 0

Errors: 0

Skipped: 0

Total time (s): 34.62

Pass rate: 100.00%

Scope

Unit tests for domain rules, services, and use cases.

Web layer tests with WebMvcTest for controllers and DTO mapping.

Security tests for JWT util and filters.

Configuration seed test for database bootstrap SQL.

Out of scope

Integration with real PostgreSQL and persistence adapters.

External email provider calls (Mailjet).

End-to-end UI or client SSE consumption.

Environment

OS: Windows-11-10.0.26100-SP0

Python: 3.14.0

Workspace: C:\Users\miche\Desktop\Proyecto Integrador\BackEnd\hexagonal-architecture-demo

Surefire reports: 56 file(s)

Recommendations

Add integration tests using Testcontainers for PostgreSQL.

Add e2e tests for notification delivery and email adapter.

Run controller tests with security filters enabled in a dedicated suite.

Consider performance tests for SSE under concurrent clients.

Test inventory (from README_TESTS.md)

TESTS GENERADOS

Este README documenta todos los tests agregados y el motivo de cada uno. El objetivo es cubrir reglas de negocio, validaciones, mapeos DTO<->dominio, controladores y seguridad sin depender de base de datos ni infraestructura externa.

COMO EJECUTAR

- Todos los tests: ‘./mvnw test’
- Un test puntual: ‘./mvnw -Dtest=NombreDeLaClaseTest test’
- Un metodo puntual (JUnit 5): ‘./mvnw -Dtest=NombreDeLaClaseTest#nombreDelMetodo test’

ESTRATEGIA

- Unit tests con Mockito para puertos y dependencias.
- WebMvcTest para controladores, sin filtros de seguridad (se mockean los puertos).
- Tests de seguridad para JWT util y filtros, con MockHttpServletRequest/Response.
- Sin DB embebida: se valida logica y contratos, no persistencia real.

HELPERS

- ‘src/test/java/com/example/demo/testsupport/TestDataFactory.java’
- Fabrica de datos reutilizables para usuarios, torneos y equipos con valores validos.

SMOKE TEST EXISTENTE

- ‘src/test/java/com/example/demo/DemoApplicationTests.java’
- Verifica que el contexto de Spring levanta. Se mantiene como smoke test.

CONFIG

- ‘src/test/java/com/example/demo/config/FormatSeederTest.java’
- Asegura que el seeder ejecuta el SQL idempotente esperado.

DOMINIO (SERVICIOS)

- ‘src/test/java/com/example/demo/core/domain/services/ValidateUruguayanIdServiceTest.java’
- Verifica CI uruguaya valida/invalida y formatos.
- ‘src/test/java/com/example/demo/core/domain/services/ValidateUserServiceTest.java’
- Verifica validaciones de email, password, fechas y perfil.

DOMINIO (MODELOS)

- ‘src/test/java/com/example/demo/core/domain/models/UserTest.java’
- Cubre validacion del constructor y actualizacion de perfil.
- ‘src/test/java/com/example/demo/core/domain/models/TournamentTest.java’
- Reglas de negocio: fechas, min/max, privacidad y participacion.
- ‘src/test/java/com/example/demo/core/domain/models/TeamTest.java’
- Participacion por id y por nationalId, y casos sin participantes.
- ‘src/test/java/com/example/demo/core/domain/models/OrganizerTest.java’
- Promedio de reputacion y manejo de nulos.

MAPPERS DTO

- ‘src/test/java/com/example/demo/adapters/in/api/mappers/UserMapperDtosTest.java’
- Mapeo registro/perfil y reputacion de organizador.

- 'src/test/java/com/example/demo/adapters/in/api/mappers/TournamentMapperTest.java'
 - Mapeo request->dominio y dominio->response (incluye status/organizerId).
- 'src/test/java/com/example/demo/adapters/in/api/mappers/TournamentSummaryMapperTest.java'
 - Mapeo de resumen y manejo de null.
- 'src/test/java/com/example/demo/adapters/in/api/mappers/DisciplineDtoMapperTest.java'
 - Mapeo de disciplina y manejo de null.
- 'src/test/java/com/example/demo/adapters/in/api/mappers/FormatDtoMapperTest.java'
 - Mapeo de formato y manejo de null.

SEGURIDAD JWT

- 'src/test/java/com/example/demo/adapters/in/api/security/JwtUtilTest.java'
 - Usa secreto fallback o secreto provisto segun longitud.
- 'src/test/java/com/example/demo/adapters/in/api/security/JwtValidationFilterTest.java'
 - Endpoints publicos sin token, rechazo de token invalido, y auth con token valido.
- 'src/test/java/com/example/demo/adapters/in/api/security/JwtAuthenticationFilterTest.java'
 - Validacion de credenciales faltantes, respuesta exitosa con token y error 401.

SERVICIOS DE APLICACION

- 'src/test/java/com/example/demo/core/application/service/NotificationSseServiceTest.java'
 - Registro de SSE, manejo de errores de serializacion y heartbeat.
- 'src/test/java/com/example/demo/core/application/service/NotificationHeartbeatSchedulerTest.java'
 - Envia heartbeat solo si hay conexiones activas.

CASOS DE USO

- 'src/test/java/com/example/demo/core/application/usecase/CreateTournamentUseCaseTest.java'
 - Valida dominio y persistencia del torneo.
- 'src/test/java/com/example/demo/core/application/usecase/RegisterUserUseCaseTest.java'
 - Guarda usuario y crea notificacion de bienvenida.
- 'src/test/java/com/example/demo/core/application/usecase/UpdateUserUseCaseTest.java'
 - Completa perfil y actualiza datos.
- 'src/test/java/com/example/demo/core/application/usecase GetUserByIdUseCaseTest.java'
 - Caso normal y comportamiento actual con null.
- 'src/test/java/com/example/demo/core/application/usecase GetUserByIdAndEmailUseCaseTest.java'
 - Validacion de email y mismatch id/email.
- 'src/test/java/com/example/demo/core/application/usecase ListUsersUseCaseTest.java'
 - Delegacion a repo.
- 'src/test/java/com/example/demo/core/application/usecase ListDisciplinesUseCaseTest.java'
 - Delegacion a repo.
- 'src/test/java/com/example/demo/core/application/usecase ListFormatsByDisciplineUseCaseTest.java'
 - Delegacion a repo.
- 'src/test/java/com/example/demo/core/application/usecase ListPublicTournamentsUseCaseTest.java'
 - Delegacion con filtros.
- 'src/test/java/com/example/demo/core/application/usecase ListTournamentsByStatusUseCaseTest.java'

- ava'
 - Null check y delegacion.
 - 'src/test/java/com/example/demo/core/application/usecase/GetAllTournamentsUseCaseTest.java'
 - Delegacion a repo.
 - 'src/test/java/com/example/demo/core/application/usecase/GetTournamentByIdUseCaseTest.java'
 - Delegacion a repo con teams.
 - 'src/test/java/com/example/demo/core/application/usecase/GetTournamentUseCaseTest.java'
 - Filtrado de torneos por nationalId.
 - 'src/test/java/com/example/demo/core/application/usecase/CancelTournamentUseCaseTest.java'
 - Permisos, estado, limpieza y notificaciones.
 - 'src/test/java/com/example/demo/core/application/usecase/StartTournamentUseCaseTest.java'
 - Validaciones de organizador/estado/fechas y notificacion.
 -
 - 'src/test/java/com/example/demo/core/application/usecase/RegisterToTournamentUseCaseTest.java'
 - Reglas de inscripcion, duplicados y torneo lleno.
 - 'src/test/java/com/example/demo/core/application/usecase/RegisterRunnerToTournamentUseCaseTest.java'
 - Inscripcion de carrera, nombre/cedula y torneo lleno.
 - 'src/test/java/com/example/demo/core/application/usecase/RegisterTeamToTournamentUseCaseTest.java'
 - Reglas por formato y notificacion.
 - 'src/test/java/com/example/demo/core/application/usecase/GenerateEliminationFixtureUseCaseTest.java'
 - Validaciones y generacion de bracket con byes.
 -
 - 'src/test/java/com/example/demo/core/application/usecase/GenerateLeagueFixtureUseCaseTest.java'
 - Round robin y doble ronda.
 - 'src/test/java/com/example/demo/core/application/usecase/ReportMatchResultUseCaseTest.java'
 - Validacion de ganador, avance de ronda y notificaciones.
 - 'src/test/java/com/example/demo/core/application/usecase/ReportLeagueMatchResultUseCaseTest.java'
 - Registro de resultado con empate permitido.
 - 'src/test/java/com/example/demo/core/application/usecase/GetLeagueStandingsUseCaseTest.java'
 - Calculo y ordenamiento de tabla.
 - 'src/test/java/com/example/demo/core/application/usecase/ReportRaceResultsUseCaseTest.java'
 - Validaciones, orden por tiempo y posiciones.
 - 'src/test/java/com/example/demo/core/application/usecase/GetRaceResultsUseCaseTest.java'
 - Persistidos vs placeholder de inscriptos.
 - 'src/test/java/com/example/demo/core/application/usecase/RemoveTeamFromTournamentUseCaseTest.java'
 - Permisos, estado, decremento y notificacion.
 - 'src/test/java/com/example/demo/core/application/usecase/ToOrganizerUseCaseTest.java'
 - Rol de organizador y manejo de usuario inexistente.
 - 'src/test/java/com/example/demo/core/application/usecase/CreateNotificationUseCaseTest.java'
 - Validaciones y envio SSE.
 -

- 'src/test/java/com/example/demo/core/application/usecase/GetUserNotificationsUseCaseTest.java'
 - Delegacion y validacion de userId.
- 'src/test/java/com/example/demo/core/application/usecase/MarkNotificationAsReadUseCaseTest.java'
 - Permisos y marcado individual/masivo.
- 'src/test/java/com/example/demo/core/application/usecase/RateOrganizerUseCaseTest.java'
 - Reglas de calificacion, participacion y stats.
- 'src/test/java/com/example/demo/core/application/usecase/GetOrganizerReputationUseCaseTest.java'
 - Distribucion y ultimas calificaciones.

CONTROLADORES (WEBMVCTEST)

- 'src/test/java/com/example/demo/adapters/in/api/controllers/UserControllerWebMvcTest.java'
 - Registro, perfil, listados y endpoints por id/email.
- 'src/test/java/com/example/demo/adapters/in/api/controllers/TournamentControllerWebMvcTest.java'
 - Creacion, fixtures, inscripciones, resultados, standings y errores.
- 'src/test/java/com/example/demo/adapters/in/api/controllers/DisciplineControllerWebMvcTest.java'
 - Listado de disciplinas y formatos.
 -
 - 'src/test/java/com/example/demo/adapters/in/api/controllers/OrganizerControllerWebMvcTest.java'
 - Calificacion y consulta de reputacion con auth.
 - 'src/test/java/com/example/demo/adapters/in/api/controllers/NotificationControllerWebMvcTest.java'
 - Crear, marcar, listar y contar notificaciones.
 - 'src/test/java/com/example/demo/adapters/in/api/controllers/PruebaControllerWebMvcTest.java'
 - Envio de correo de prueba.

NOTAS

- Los tests de WebMvcTest y el smoke test cargan contexto Spring; por eso pueden tardar mas.
- No se agrego DB embebida ni Testcontainers para mantener la ejecucion local simple.