Code coverage report

✅ What Worked Well

All 164 tests passed without any failures, errors, or skipped tests.
Complete test coverage achieved across all application layers.
JaCoCo agent was successfully attached and recorded test coverage during all test executions.

🏛 Controller Layer (Integration Tests)

- `AuthController`: 16 tests covering authentication endpoints

- `ProjectController`: 11 tests for project management

- `InvitationController`: 14 tests for invitation functionality

🔧 Service Layer (Unit Tests)

- `ProfileService`: 16 tests for user profile management

- `ProjectService`: 10 tests for project-related operations

- `InvitationService`: 9 tests for invitation handling

- `RoleApplicationService`: 9 tests for role application workflows

- `RefreshTokenService`: 10 tests for token management

- `ProjectRoleService`: 9 tests for project role operations

- `UserService`: 4 tests for user management logic

🗄 Repository Layer (Integration Tests)

- `ProfileRepository`: 12 tests with real database interactions

- `ProjectRepository`: 8 tests verifying project data persistence

- `UserRepository`: 5 tests for user data access

- `InvitationRepository`: complex integration tests covering invite persistence logic

🔐 Security Layer

- JwtUtil: 14 comprehensive tests validating JWT token generation and verification

🏗️ Application Layer

- BackendApplicationTests: confirmed full application context loads without error

📈 Quality Metrics

- Full test suite completed in ~35 seconds

- Environment: Java 21, Spring Boot 3.5.0, PostgreSQL (Testcontainers)

- Frameworks: JUnit 5, Mockito, Spring Boot Test, Testcontainers

- Coverage Tool: JaCoCo Maven Plugin successfully configured