**Run Tests**

bash

*# Run all tests*

npm test

*# Run specific test suite*

npm run test:auth

*# Run with coverage report*

npm run test:coverage

*# Watch mode for development*

npm run test:watch

**Key Testing Features**

**In-Memory Database**

* Uses MongoDB Memory Server for isolated testing
* No external database dependencies
* Fast test execution
* Clean state between tests

**Comprehensive Test Helpers**

* **User Creation**: createTestUser(), createTestUsers()
* **Data Setup**: createTestIssue(), createTestComment()
* **API Helpers**: registerUser(), loginUser(), createIssueViaAPI()
* **Assertions**: assertSuccessResponse(), assertErrorResponse()
* **Auth Headers**: getAuthHeader(token)

**Real File Testing**

* Creates actual test files for upload testing
* Tests file validation and security
* Verifies file cleanup and integrity
* Tests multiple file upload scenarios

**Edge Case Coverage**

* Invalid data validation
* Permission and authorization checks
* Rate limiting verification
* Error scenario testing
* Boundary condition testing

**Test Statistics Summary**

| **Test Suite** | **Test Count** | **Coverage Areas** |
| --- | --- | --- |
| **Authentication** | 25+ tests | Registration, Login, JWT, Security |
| **Issues** | 35+ tests | CRUD, Search, Filter, Permissions |
| **Comments** | 25+ tests | Threading, Validation, Access Control |
| **Files** | 25+ tests | Upload, Download, Validation, Stats |
| **Total** | **110+ tests** | **Complete API Coverage** |

**Test Categories**

**Happy Path Tests**

* Successful operations with valid data
* Normal user workflows
* Expected functionality verification

**Error Scenario Tests**

* Invalid input handling
* Authentication failures
* Permission denials
* Resource not found cases

**Edge Case Tests**

* Boundary conditions
* Rate limiting
* File size limits
* Concurrent operations

**Integration Tests**

* End-to-end workflows
* Multi-step processes
* Data consistency checks
* Relationship integrity