

# User Service Test Coverage Summary

## Unit Tests

- `UserDtoMapperTest` - Tests DTO ↔ Domain mapping correctness. Ensures fields map properly and null safety.
- `UserModelAssemblerTest` - Tests HATEOAS link generation. Ensures correct self and collection links.
- `UserServiceTest` - Tests service logic (create, get, update, patch, delete) with mocked repository.
- `UserServiceExceptionTest` - Tests exception scenarios (`UserNotFoundException` thrown when not found).

## Integration Tests (Controller-level)

- `UserControllerIntegrationTest` - Uses `@WebMvcTest`. Covers endpoints, JSON payloads, status codes (200, 201, 204).
- `UserControllerValidationTest` - Tests validation errors (400 Bad Request). Covers invalid email, missing fields, short phone number.

## Full Integration Test (End-to-End with DB)

- `UserModuleIntegrationTest` - Uses `@SpringBootTest` with H2. Covers lifecycle: POST, GET, PUT, PATCH, DELETE. Ensures persistence, validation, mapping, repository, service, controller work together. Covers status codes: 200, 201, 204, 400, 404.

### Summary of Coverage

- Unit tests: Service logic, mapping, assembler, and exceptions.
- Integration tests: Controller endpoints, JSON, validation errors.
- Full integration: Entire stack with H2 DB, lifecycle flows, error handling.
- Status codes covered: 200, 201, 204, 400, 404.
- Missing: Explicit tests for JSON error body of `UserNotFoundException` (404).