

ReviewServiceExtendedTest Documentation

ReviewServiceExtendedTest Documentation

Overview:

This document covers three additional unit test cases added to the ReviewService class to enhance validation and robustness.

The newly added test methods focus on corner cases such as null inputs, proper average rating recalculation, and decline behavior.

New Test Cases:

1. testPostReview_RecalculatesAverageRatingCorrectly

Purpose: Ensure that the average rating of a product is updated accurately after a new review is posted.

Scenario:

- Product already has one review with rating 5.0
- A new review with rating 3.0 is posted
- The average is recalculated and updated on the product

Assert:

- Product's rating is set to 4.0 (average of 5.0 and 3.0)
- The review ID is properly linked
- Product save is invoked

2. testDeclineReview_Successful

Purpose: Validate that an existing review can be successfully marked as unverified.

Scenario:

- The review is retrieved by ID

- Its verified field is set to false and saved

Assert:

- Returned review has verified = false
- Save is called once

3. testDeclineReview_NotFound

Purpose: Ensure the service throws an exception when trying to decline a non-existent review.

Scenario:

- Review ID does not exist in the repository

Assert:

- IllegalArgumentException is thrown
- Repository save is never invoked

Mocking Strategy:

- ReviewRepository and ProductRepository are mocked for data access
- Ratings are simulated using mock return values
- Verification ensures business logic is triggered as expected

Conclusion:

These extended test cases strengthen the validation logic for product-review interactions.

They ensure accurate data integrity for both review verification status and product rating consistency.