

Here's an automated AI-generated review to support your development workflow.

Summary

- Refactors payment flow into a dedicated module with clearer separation of concerns
- Centralizes currency conversion and validation in one place
- Improves testability by injecting dependencies for external payment APIs

Potential Issues

- Error handling in the refund path does not distinguish between transient and permanent failures; retries may not be appropriate for all cases
- Edge cases for currency conversion (e.g. rounding, negative amounts) are not explicitly documented or tested
- Some payment provider responses are logged at INFO level; consider moving sensitive fields to DEBUG only

Potential Optimizations

- Consider a small state machine or status enum for payment lifecycle so edge cases are explicit
- Add idempotency keys for payment and refund operations to support safe retries
- Document expected currency precision and rounding rules for multi-currency support

Suggestions

- Error handling improvements suggested for refund and webhook paths
- Edge cases for currency conversion should have unit tests with boundary values
- Test coverage recommendations: add integration tests for the new module and mock the provider

— This review was generated by [Gitzoid](#), an AI-powered code review assistant.