

COEN 6761
Software Validation and Testing
Winter 2026

Submitting to:
Prof. Yan Liu

Submitted by:
Azan Ali Shah Syed (40241942)

Github Issues:

The screenshot shows a GitHub Issues page for the repository 'asaji1999 / Coen7671-assign1'. The search bar contains 'is:issue state:closed'. There are three closed issues listed:

- #3 - Implement Fail-Soft policy and document failure semantics (closed 20 minutes ago)
- #2 - Implement Fail-Partial Policy (closed 19 minutes ago)
- #1 - Implement a Fail-Fast Policy (closed 19 minutes ago)

Figure 1: GitHub issues created to track implementation of failure policies and documentation.

Issue 1:

Implement a Fail-Fast Policy #1

This issue tracks the implementation of the fail-fast (atomic) failure handling policy for concurrent microservice calls using CompletableFuture.

asaji1999 opened 5 hours ago

asaji1999 closed this as completed 21 minutes ago

asaji1999 10 minutes ago

The fail-fast (atomic) policy has been implemented and verified through unit testing.

Issue 2:

Implement Fail-Partial Policy #2

 Closed

 asaji1999 opened 5 hours ago Owner ...

This issue tracks the implementation of the fail-partial (best-effort) failure handling policy.

  asaji1999 closed this as completed 22 minutes ago

 asaji1999 10 minutes ago Owner Author ...

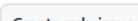
The fail-partial (best-effort) policy has been implemented and validated through unit tests.



Issue 3:

 asaji1999 opened 6 hours ago Owner ...

This issue tracks the implementation of the fail-soft policy along with required documentation of failure semantics.

  asaji1999 closed this as completed 1 hour ago

 asaji1999 48 minutes ago Owner Author ...

The fail-soft policy and the required documentation (failure-semantics.md) have been completed.



Feature Branches:

The screenshot shows a Git interface with two sections: 'Default' and 'Your branches'. In the 'Default' section, there is one branch named 'main' which was updated 48 minutes ago. In the 'Your branches' section, there are two feature branches: 'feature/readability' and 'feature/docs-update', both updated 5 hours ago. Each branch has a status bar indicating '3 | 0' and a pull request button labeled '#5' and '#4' respectively.

Branch	Updated	Check status	Behind	Ahead	Pull request
main	48 minutes ago	Default			

Branch	Updated	Check status	Behind	Ahead	Pull request
feature/readability	5 hours ago		3	0	#5
feature/docs-update	5 hours ago		3	0	#4

Main branch and two Feature Branches.

Pull Requests:

The screenshot shows a list of pull requests. There are two open pull requests: '#5 by asaji1999' and '#4 by asaji1999'. Both were merged 51 and 50 minutes ago respectively. The merge commit hash for #5 is 207f42a.

Author	Label	Projects	Milestones	Reviews	Assignee	Sort
asaji1999						

0 Open 2 Closed

Pull requests used to merge feature branches into main.

PR # 1:

The screenshot shows the details of pull request #5. It includes a comment from 'asaji1999' 5 hours ago stating: 'This pull request adds a small documentation comment to improve code readability in AsyncProcessor.' Below the comment is a reply from 'asaji1999' with the commit message 'Code: minor readability comment'. The merge commit hash is 207f42a. A revert button is also visible.

asaji1999 commented 5 hours ago

This pull request adds a small documentation comment to improve code readability in AsyncProcessor.

asaji1999 Code: minor readability comment 207f42a

asaji1999 merged commit 1af4ba5 into main 52 minutes ago

Revert

PR# 2:

This screenshot shows a GitHub Pull Request (PR# 2) with the following details:

- Comment by asaji1999:** Small documentation update for clarity.
- Commit:** Docs: minor clarification (commit hash: 849f06c)
- Review by Goddey99:** Approved these changes 4 hours ago. A link to "View reviewed changes" is provided.
- Comment by Goddey99:** Reviewed the documentation update. The clarification is clear and improves readability. No issues found.

Peer Code Review:

Docs: minor clarification #4

Merged asaji1999 merged 1 commit into main from feature/docs-update 53 minutes ago

This screenshot shows a GitHub Peer Code Review (PR# 4) with the following details:

- Comment by asaji1999:** Small documentation update for clarity.
- Commit:** Docs: minor clarification (commit hash: 849f06c)
- Review by Goddey99:** Approved these changes 4 hours ago. A link to "View reviewed changes" is provided.
- Comment by Goddey99:** Reviewed the documentation update. The clarification is clear and improves readability. No issues found.

Peer review was done on Pull Request Docs:Minor Clarification by Godred.

Failure Semantics:

A failure-semantics.md file was created in the docs folder. Below is a snapshot of it:

The screenshot shows a code editor interface with a sidebar on the left and a main content area on the right. The sidebar has a 'Files' tab and displays a file tree with the following structure:

- main
- docs (selected)
- failure-semantics.md (highlighted)
- src
- pom.xml

The main content area has tabs at the top: Preview, Code, Blame. The preview tab is active. The content is as follows:

Failure Semantics for Concurrent Microservice Calls

This project implements three failure-handling policies for concurrent microservice calls using `CompletableFuture`.

Context

Given a list of microservices and a corresponding list of messages (same size), each microservice is called asynchronously using `retrieveAsync(message)`.

The policies define how the overall computation behaves when one or more microservices fail.

1) Fail-Fast (Atomic)

Method: `processAsyncFailFast(services, messages)`

Rule: If *any* microservice call fails, the entire operation fails.

Result:

AI Usage Claim:

The AI tool ChatGPT Version 5.2 was used for proofreading of failure-semantics.md and guidance regarding markdown styling for .md files. Further it was used for reviewing of code and gaining insight on how to improve the code by the author. All code changes , decisions were done by the author whereas ChatGPT was used mostly as a learning aid and as a reviewing tool.

Prompts:

“For the code as given below, I need to differentiate between the three policies Fail-Fast, Fail-Partial, Fail-Soft. I have to create it in a .md file format , how should I structure it and inform me of the various markdown commands and tips to have a clean and neat .md report.”

“Explain fail-fast vs fail-partial vs fail-soft for the following assignment which I have uploaded as a pdf. How should I start , and what I should look out for each step.”