5.2 Construct a CI Pipeline

My pipeline

I gathered inspiration for my CI pipeline from an article I found covering continuous integration of Java projects using GitHub Actions (Krzywiec, 2020).

The basic steps of any CI pipelines are:

- Build in a fresh environment
- Test

Building and testing in a clean environment ensures that no individual developer's local setup influences test outcomes, making the results repeatable and reliable.

The CI pipeline constructed for this project implements these principles by automatically building the project and running the full test suite on each push to the repository. This provides that the software continues to function as required, even with changes.

Next Steps

Whilst my pipeline is appropriate for a smaller project, in production or team there would be improvement required.

The first step for me would be adding a linting stage. This article (Madrazo, 2024) provides good information on why linting is crucial. Linting improves code quality and gives a more uniform style, improving readability and maintainability.

Adding a linting stage is an is an important step for any pipeline, as it acknowledges that a project is more serious and may have a larger team working on it over time. As a first step, I would add checkstyle. It would provide a good foundation for additional linting and would enforce programming standards.

References

Krzywiec, W. (2020, January 20). Retrieved from Medium: https://faun.pub/continuous-integration-of-java-project-with-github-actions-7a8a0e8246ef?gi=5c9c8b20491f

Madrazo, E. (2024, December 19). Retrieved from Medium: https://medium.com/@emadrazo/more-than-10-reasons-to-have-a-linting-stage-in-your-pipeline-4303c033c77f