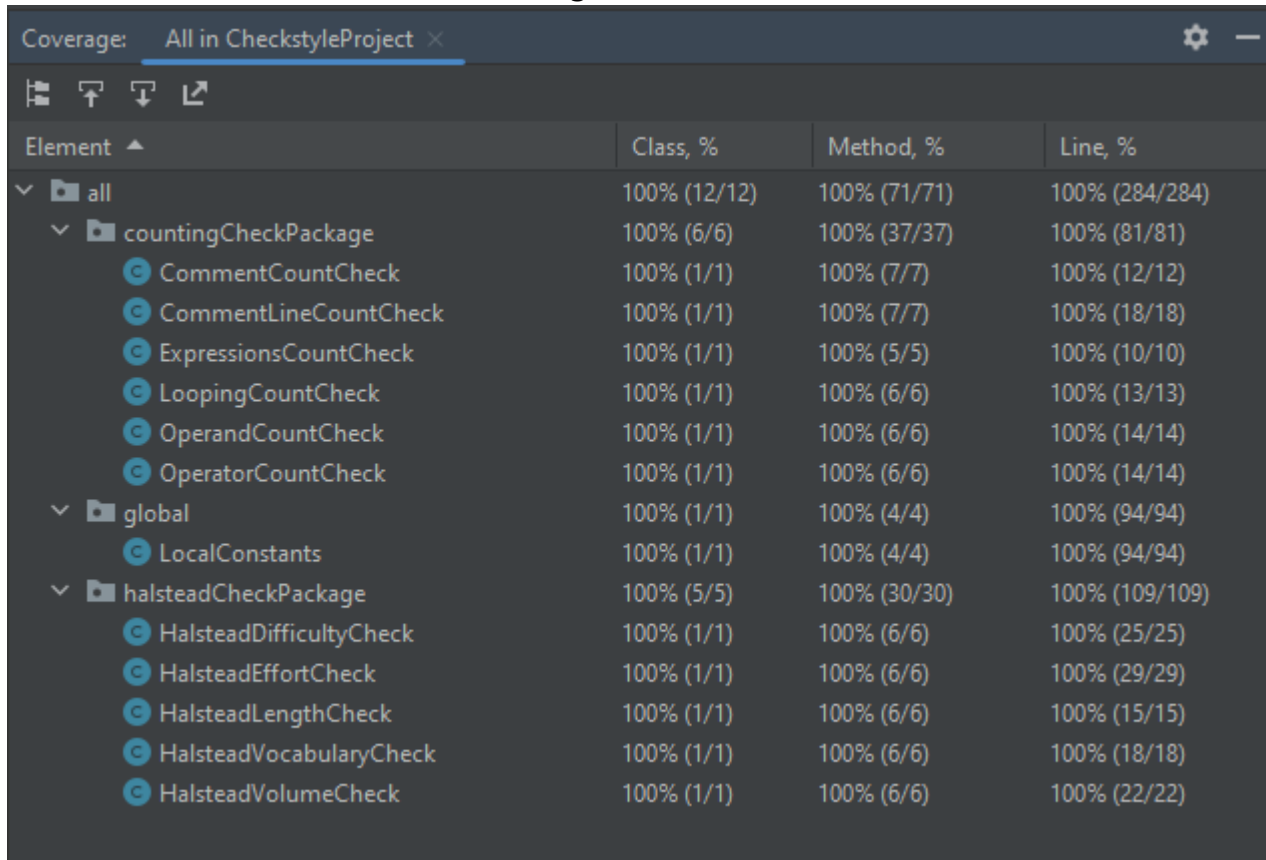


# 422 Milestone 2 report

In my defense this report was not talked about in depth ever, and I don't know what I'm doing.

Also this is a screenshot of the coverage



The screenshot shows a code coverage tool interface with a dark theme. At the top, it says 'Coverage: All in CheckstyleProject'. Below this is a table with four columns: 'Element', 'Class, %', 'Method, %', and 'Line, %'. The table lists various packages and classes, all of which show 100% coverage. The packages listed are 'all', 'countingCheckPackage', 'global', and 'halsteadCheckPackage'. Each package contains several classes, all of which are marked with a green circle icon indicating 100% coverage.

Element	Class, %	Method, %	Line, %
all	100% (12/12)	100% (71/71)	100% (284/284)
countingCheckPackage	100% (6/6)	100% (37/37)	100% (81/81)
CommentCountCheck	100% (1/1)	100% (7/7)	100% (12/12)
CommentLineCountCheck	100% (1/1)	100% (7/7)	100% (18/18)
ExpressionsCountCheck	100% (1/1)	100% (5/5)	100% (10/10)
LoopingCountCheck	100% (1/1)	100% (6/6)	100% (13/13)
OperandCountCheck	100% (1/1)	100% (6/6)	100% (14/14)
OperatorCountCheck	100% (1/1)	100% (6/6)	100% (14/14)
global	100% (1/1)	100% (4/4)	100% (94/94)
LocalConstants	100% (1/1)	100% (4/4)	100% (94/94)
halsteadCheckPackage	100% (5/5)	100% (30/30)	100% (109/109)
HalsteadDifficultyCheck	100% (1/1)	100% (6/6)	100% (25/25)
HalsteadEffortCheck	100% (1/1)	100% (6/6)	100% (29/29)
HalsteadLengthCheck	100% (1/1)	100% (6/6)	100% (15/15)
HalsteadVocabularyCheck	100% (1/1)	100% (6/6)	100% (18/18)
HalsteadVolumeCheck	100% (1/1)	100% (6/6)	100% (22/22)

Overall I did not have any lines that I could not get covered and because most of the methods that I am testing are void I just make sure that they run at least once. In hindsight I probably should have maybe pulled some of the functionality out of some of the methods to make an easier testing experience. The reason that I did not do that is because it would have been pulled out into a private method so I could not test it anyway.