|  |
| --- |
| Software Deployment |
| Assessment 1 |
| Portfolio Assessment 1 |

|  |
| --- |
| Kyle Chamberlain  2-11-2021 |

Table of Contents

[Optimisation Report 0](#_Toc74900296)

[What is a code optimiser? 0](#_Toc74900297)

[Machine Independent Optimisation 0](#_Toc74900298)

[Machine Dependent optimisation 0](#_Toc74900299)

# Optimisation Report

## What is a code optimiser?

Code optimization is a transformation technique used to improve code. This can be done by reducing the resources being used. Some goals that should be achieved during code optimisation are;

* The optimisation of code should not change the way the program functions as a whole.
* It should increase the speed and overall performance of code.
* Compiler time must be relatively quick and does not increase the delay for overall compiling.

Code should be optimised at the end stage of the development process. Since we are nearing an end for this project the code will be getting optimised. For this we will be looking at various types of optimisation.

### Machine Independent Optimisation

This is optimisation phase improves intermediate code so as to get a better target code being output.

### Machine Dependent optimisation

This is done after the target code has been generated.

## Examples of Code Optimisers

* PHP CodeSniffer
* Atoum
* PHPSpec

## PHPCodeSniffer

For this project we will be using the PEAR Code Standard. This is an industry standard that both optimises code as well as maintaining a certain readability for developers. Each time that the code is optimised a report is made showing where all non-pear code standard blocks are and how they need to be fixed. This is a great tool for developers.