

CSU33012 Software Engineering

Measuring Engineering Report

**Stephen Davis, Std# 18324401**

**November 14, 2021**

Contents

[1 Introduction 2](#_Toc88684101)

[2 How Can Software Engineering be Measured? 2](#_Toc88684102)

[2.1 Lines of Code 2](#_Toc88684103)

[2.2 Number/Frequency of Commits 2](#_Toc88684104)

[2.3 Number of Features Delivered 2](#_Toc88684105)

[2.4 Number of Code Reviews Done 2](#_Toc88684106)

[2.5 Leadtime 2](#_Toc88684107)

[2.6 Total Hours of Downtime 2](#_Toc88684108)

[3 What Platforms Can Be Used to Gather and Process Data? 3](#_Toc88684109)

[3.1 Pluralisation (Git Prime) 3](#_Toc88684110)

[3.2 WayDev 3](#_Toc88684111)

[3.3 Code Climate 3](#_Toc88684112)

[3.4 Hackystat 3](#_Toc88684113)

[3.5 Personal Software Process 3](#_Toc88684114)

[4 What Algorithms Can We Use? 3](#_Toc88684115)

[4.1 Halstead Complexity Measures 3](#_Toc88684116)

[4.2 Artificial Intelligence 3](#_Toc88684117)

[4.3 Computational Intelligence 4](#_Toc88684118)

[4.4 Cyclomatic Complexity 4](#_Toc88684119)

[5 Is This Ethical? 4](#_Toc88684120)

[5.1 Privacy 4](#_Toc88684121)

[5.2 Increasing Productivity 4](#_Toc88684122)

[6 Conclusion 4](#_Toc88684123)

# Introduction

This report will discuss why one would want to measure software engineering, how software engineering can be measured, what platforms can be used to measure software engineering and what algorithms we can use to carry out such measuring. This report will then discuss the ethics of measuring software engineering, before finally concluding.

# How Can Software Engineering be Measured?

There is much debate about whether software engineering can actually be measured. A popular view, and my personal view, is not whether software engineering can be measured, but how can software engineering be measured? It is worth noting that there are many ways we can measure software engineering, but not every way is accurate, insightful and fair. Most of the time the issue with these inaccurate measurements is that they lack context. It is worth noting that each measurement comes with its own advantages and disadvantages. Some examples of how to measure software engineering productivity which I will now discuss are:

* Lines of Code
* Number/Frequency of Commits
* The Number of Features Delivered
* Number of Code Reviews Done
* Leadtime – how long it takes you to go from idea to delivered software
* Total hours of downtime – dice used to count time spent on activity

## Lines of Code

## Number/Frequency of Commits

## Number of Features Delivered

## Number of Code Reviews Done

## Leadtime

## Total Hours of Downtime

With any measurement, if you measure the wrong things, you risk pushing the quality of code down, which tends to reward mediocrity.

# What Platforms Can Be Used to Gather and Process Data?

There are various platforms which can be used to gather and process data in our search to measure software engineering. The examples I will discuss in this report are:

* Pluralisation (Git Prime)
* WayDev
* Code Climate
* Hackystat
* Personal Software Process

## Pluralisation (Git Prime)

## WayDev

## Code Climate

## Hackystat

## Personal Software Process

# What Algorithms Can We Use?

* Halstead Complexity Measures
* Artificial Intelligence
* Computational Intelligence
* Cyclomatic Complexity

## Halstead Complexity Measures

## Artificial Intelligence

## Computational Intelligence

## Cyclomatic Complexity

# Is This Ethical?

* Moral Issues - Privacy
* Increasing Productivity

## Privacy

## Increasing Productivity

# Conclusion

Source:

<https://blog.pragmaticengineer.com/can-you-measure-developer-productivity/>