



CSU330312 Software Engineering

Measuring Software Engineering Report

Liam Junkermann - 19300141

December 21, 2021

Contents

1	Measurable Data	2
	Commits and code frequency	2
	Testing Coverage	2
	Pull Request creation or review frequency	2
	Task and time tracking systems	3
	Health and Activity Tracking	3
2	Connecting and performing Calculations on Activity Data sets	4
	Integrated Platforms	4
	Custom Pipelines	4
3	Computation to profile software engineers	5
4	Morality, Ethics, and Legality of Collecting and processing engineer measurement	5

Introduction

Software engineering is a notoriously difficult to track for time and productivity. Some engineers may be more efficient with some technologies and not others, and tasks may have varying difficulty making engineering activity and productivity particularly difficult to accurately and consistently track productivity. This report will discuss various ways engineering activity can be measured, appropriate data collected, and then processed to accurately profile engineers. Finally, the moral, ethical, and legal consequences of these methods will be explored.

1 Measurable Data

There are a number of metrics which can be used to determine engineering activity depending on the way a given repository or organisation is set up.

Commits and code frequency

The simplest way to track engineering productivity could be based on the commit and code frequency charts provided by most version control providers. These charts allow a supervisor, or even the engineer themselves, to see the number of commits they make over a period of time, as well as the number of lines added or removed from the files within the repository. This method of measuring engineering activity is rudimentary and does not account for tasks which may take more time to research, plan, or structure – all subtasks which would not be reflected in a simple commit or code frequency chart. In a team it is incredibly important for team members to make and label commits appropriately in order to allow other team members to see which tasks are being completed, how they are being completed, and to more easily find the necessary changes in lines or files during code reviews.

Testing Coverage

An important part of any project which looks to see a production environment is the testability and rigorous testing to ensure that the solution can run effectively in that production environment. Any solutions a developer writes must be tested, and many production pipelines provide engineers graphs or statistics about code coverage – as in which lines of code were tested and which were not. These statistics allow engineers to catch bugs and ensure their solutions is being tested appropriately.

Pull Request creation or review frequency

Version control is used to allow many people to work on a project together at the same time. Organisations will use pull requests to combine the work of various engineers in order to begin the production pipeline (the flow within which the code is built and packaged for release to production or testing groups). While commit frequency and volume are important to track activity, and break down tasks, ultimately contribution to production environments is what dictates success and progress. Pull requests are the measure for those production contributions, they

allow the engineer who has worked on a feature or task to share that work with their team, and allows the rest of the team to make comments about the functionality or approach used to solve the problem presented in the task or feature. It is important, for both the team and the product they are working on, that the new features sent to the production pipeline will work and solve the problem stated. Tracking engineering activity through pull requests is an effective way to track both individual engineering contributions to a production product, and the contributions from other team members through the code review process.

Task and time tracking systems

One of the more effective and holistic ways to track engineering activity and performance leverages and expands upon the previous two solutions. Using a system like JIRA, Azure Dev Ops (ADO) or even certain functions of GitHub allow teams to track tasks that need to be completed and track progress through tasks broken down to different degrees of complexity, as well as link progress with commits and completed pull requests. This method of measuring engineering performance allows progress on tasks to be tracked in reference with actual effort and hours of work put into a given task or subtask. This also allows the management of the team to improve as time estimation becomes more accurate with iterations of using this kind of solution.

Health and Activity Tracking

There are many health tracking solutions on the market today which can be used to track engineers daily activity, sleep effectiveness and hygiene, among a myriad of other potential metrics to track. These health tracking solutions can provide insight into how certain habits affect overall health acutely – on a day-to-day or "once off" basis – or more chronically, effects that may build over time. Collecting and understanding this data can be very helpful for some people to see and change their behaviour to achieve a more balanced life, which can only improve productivity.

Health tracking options include common and integrated devices like Apple or Samsung watches, these watches allow users to very easily track metrics like sleep, day to day activity, exercise distance and intensity metrics, and some other health metrics depending on the model, all while enhancing the users experience and productivity with their attached mobile devices. These watches can serve engineers with reminders to drink water, stand up and walk around (particularly important given that most software developers spend many hours a day sitting), and even can have reminders to eat or drink if that is something a user may struggle with.

There are also more targeted devices which can be used, such as an Oura ring or Whoop strap. These devices are much more targeted in their intended use, and carry a different price tag accordingly. The Oura ring is geared toward analysing and improving sleep, reportedly being nearly as accurate as a full scale sleep lab in analysing your sleep. Boasting a high accuracy in measuring sleeping heart rate, heart rate variability, respiratory rate, body temperature, with more features such as sleeping blood oxygen coming soon, this option allows engineers to optimise their sleep and habits around sleep to maximise their productivity and recovery every day, particularly if they engage in consistent high intensity exercise.

The Whoop Band is much more geared toward athletes. Whoop is a low profile strap worn around the wrist or bicep (depending on the user's preference) which records heart rate throughout the day and night, with the intention of being worn, like the Oura ring, 24/7. The data from the strap is then uploaded to the Whoop app, before then being analysed to determine the athlete's strain throughout the day. Whoop would be an especially good tool for engineers who exercise and are looking for an activity tracking solution which requires minimal effort. Whoop and Oura have the functionality to share activity data which is more universally accessible than the smart-watch options noted above which may be useful for an organisation looking to build an engineers profile using their activity data. The morality and legality of this will be discussed later on in this report.

2 Connecting and performing Calculations on Activity Data sets

Integrated Platforms

Code analytics There are various ways to connect and collect activity data with platforms which can analyse that data, many times these platforms – which handle collection and analysis of code productivity data – are the same. Platforms like GitHub, JIRA, and ADO provide many charts and analytics options for specific repositories. GitHub also provides an API which allows anybody to view the commit, PR and Code review statistics of any given user. These can be collected and passed to other analytics services or to a custom service or pipeline which executes analytics on datasets. These integrated systems also produce data and graphics which can be used to determine the time it takes to complete a given task or the progress of tasks and epics through commits and PRs. Utilising a complete system like this (or with native integrations like JIRA and Bitbucket), can streamline the process of finding collecting and performing analytics on a set of data.

Custom Pipelines

Code Analytics Popular version control providers, including many self-hosted solutions, provide APIs and dataset hooks to access data. This allows for many data analytics options. There are a few options for data analytics solutions.

Excel A basic Excel spreadsheet can be used to generate charts and data visualisations that may be used to profile an engineer. Excel also allows for data to be used offline and manipulated more interactively than the integrated platforms allow.

Pandas Data analysis Pandas is a python data analytics library which allows the users to manipulate and complete operations on data in a more extendable way than Excel. It also can be used alongside libraries like Matplotlib to generate charts and show any relationships found within the Pandas datasets.

3 Computation to profile software engineers

4 Morality, Ethics, and Legality of Collecting and processing engineer measurement