Unit 1 SCM Recommendation

**Corey Crooks**

**Purdue University Global**

**IT481 – Advanced Software Development**

**Ahmad Kassem**

**April 26th, 2023**

**SCM Overview and Recommendation**

Software Configuration Management (SCM) tools are essential parts of any system’s design process. These tools allow a user to more coherently scan through activity to look for trends, and potential malicious entries. According to a guide published by the United States Department of Energy, “Configuration Management is practiced in one form or another as part of any software engineering project where several individuals or organizations have to coordinate their activities.” (2000). In each system, SCM tools allow developers to better coordinate their development efforts to share and conceptualize features for their software in a more seamless manner. These tools are imperative to the success of a proper development team, and will help ensure a higher quality of work than would be otherwise possible. This need for SCM tools leaves our company currently vulnerable to a lesser quality output, and must be addressed to preserve organization and cooperation among our development team.

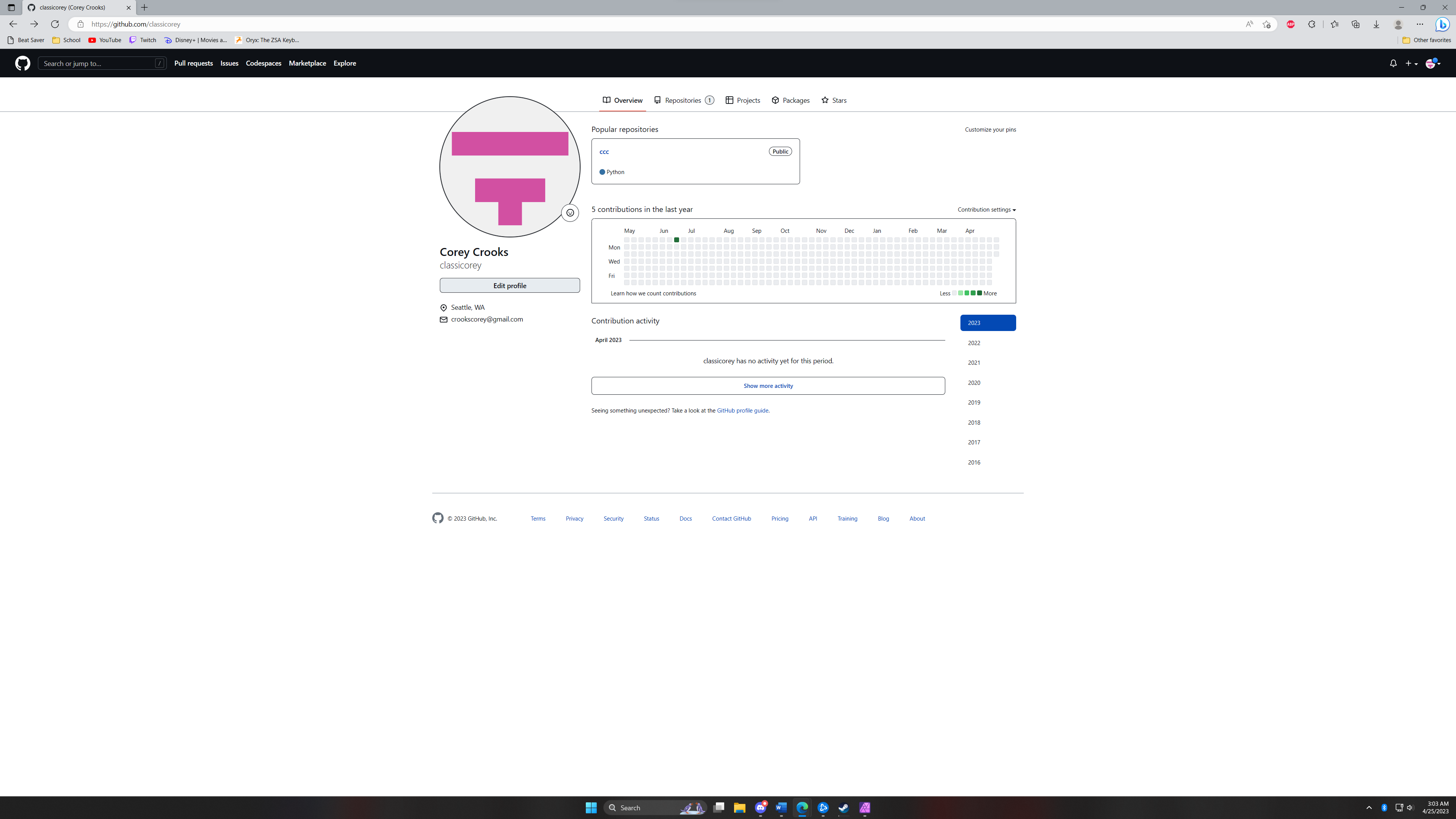
These systems are not all the same, however. Some offer features different from one another. Some may be free and open source, while others may have proprietary “creature comforts” to choose from and consider. For example, the massively popular Git suite of tools are completely free to use, distribute, and develop due to their nature of being completely open-source (git.com, n.d.). Being open-source, this application has extensive documentation available to help teams utilizing the SCM capabilities to understand the diverse feature set, and troubleshoot any issues that may arise on their own. This may be both a problem, and a feature. Due to the expansive community, issues are commonly responded to and some answer may be given. But given the nature of this interaction, these answers do not come from any kind of paid employee; as no such individual exists. Instead, this is community driven feedback that comes from an unrelated individual often from the top of their mind, and is in no way verifiable fact. Because of this, information about troubleshooting may not be a direct solution, or may not even be helpful at all. This is in stark contrast to a paid alternative such as SolarWinds SCM tools. Customer Service in this aspect is driven at the reputation of the company providing it, and as a relation stives to keep that reputation in good standing through the interactions it provides and solutions it delivers (SolarWinds, 2023). Although customer service is not the only difference between these distinct tools, this does highlight major areas of concern.

Pricing is another matter to look into. As monetary gains drive any company, the amount spent on SCM tools is definitely a factor worth considering. Opting for open-source solutions like the one mentioned above may be appealing in this category given that there is no cost to using such solutions. This is extremely advantageous, as many of these SCM tools do business in paid installations with monthly or annual payments. For example, SolarWinds offers a renewing subscription starting at $1,156 at the time of writing (SolarWinds, 2023). Another cloud hosting tool comes in the form of Auvik. Auvik provides advanced features such as topology maps, subnets of devices used, automated configuration backups, and more that are not seen—or seen in limited capacities—in competitors such as SolarWinds. The pricing of features depends on the “billable devices” on the company’s network to be secured, but may cost the company upwards of $150 per month (Auvik Networks, n.d.). This particular investment may need some additional consideration by the company into how exactly Auvik may need to adjust the price of the SCM tool depending on how many “billable devices” we own such as routers and modems. Additionally, it will need careful consideration to keep track in case the company needs to upscale the office, and buy additional devices such as routers and network extenders, as these would also increase the price of Auvik’s SCM tool suite; adding further complexity upon this investment.

One last option to consider is the availability of Cloud-based technologies. Cloud technologies come with their own set of benefits and drawbacks. For instance, operating with a cloud-based SCM suite may offer many benefits to adaptability and flexibility. Cloud centric devices may host source content and code in their own servers. This may eliminate the risk of the entire development being destroyed should our offices be compromised by a malicious attack, or natural disaster. Additionally, Cloud computing may cost a lesser amount depending on the usage types and performance demands our team meets and maintains (Amazon, 2023). This could give a better deal while using services we need, and not paying for additional services we don’t. This increased flexibility may come at a cost, however. Primarily, cloud-based services may be difficult to integrate. With the addition of new technologies, things like API’s and on-demand scaling as detailed previously may add additional complexity to an already technically demanding challenge (Morrow, 2018). Should the company not need these tools, this additional complexity may create undue risks of mismanagement and user error that potentially leads to serious concerns and data loss.

Overall, recommending a single SCM tool to utilize should be a decision put for our entire team to weigh in on, as it would affect all of us in our daily work. However, I do have a suggestion that I would recommend. As cost may be an extremely relevant factor in a small development studio of five overseen by a project manager, it would be wise to look into cost-mitigating solutions such as open-source technologies. These technologies would also give out studio the flexibility of experimentation, and a risk-free suite to develop multiple different projects as may be designated to us in the future, as expansion is common in today’s adapting environment. With these considerations, I believe a hyper-popular framework and SCM tool such as Git may be the wisest choice to consider. Given the popularity, it shouldn’t be an issue for our team to learn and adapt to. It also provides a simple and open-source front end to navigate and utilize to our teams need all while being completely free to scale however we require. This will give us an additional avenue to pursue while taking on new tasks and contracts as delegated to our team.

**Screenshot Proof of Git Account Creation**



# **References**

Amazon. (2023, April 25). *Amazon Web Services Auto-Scaling*. Retrieved from aws.amazon.com: https://aws.amazon.com/gamelift/pricing/

Auvik Networks. (n.d.). *Auvik Summary*. Retrieved from capterra.com: https://www.capterra.com/p/145245/Auvik/

git.com. (n.d.). *Git Home*. Retrieved from git-scm.com: https://git-scm.com/

Morrow, T. (2018, March 5). *12 Risks, Threats, & Vulnerabilities in Moving to the Cloud*. Retrieved from Software Engineering Institute: https://insights.sei.cmu.edu/blog/12-risks-threats-vulnerabilities-in-moving-to-the-cloud/

SolarWinds. (2023). *Server Configuration Monitor*. Retrieved from solarwinds.com: https://www.solarwinds.com/server-configuration-monitor

SolarWinds. (2023). *SolarWinds Academy Training and eLearning*. Retrieved from support.solarwinds.com: https://support.solarwinds.com/server-configuration-monitor

United States Department of Energy. (2000, April 25). *Software Configuration Management (SCM) A Practical Guide*. Retrieved from Energy.gov: https://www.energy.gov/sites/prod/files/cioprod/documents/scmguide.pdf