

## Assignment-2

### Introduction

The purpose of this report is to analyze and compare the structural quality of three open-source Python projects: S3cmd, Requests, and Python-Twitter. I focused on three key metrics: coupling, git changes (modifiability/changeability), and maintainability index. These metrics give insight into the dependency between modules, the effort required to modify the systems, and how easy it is to maintain the codebase.

### Coupling

Coupling measures the level of dependency between different modules within a system. In general, lower coupling is preferred, as highly coupled systems tend to be more complex and harder to maintain or extend.

- **S3cmd:** The coupling results show that S3cmd has moderate to high coupling in some areas, with peaks in the 11, 9, and 6 ranges. This means some modules in the system depend on several others, which could make modifications more challenging. However, there are also many modules with lower coupling, so the system may still have well-isolated areas.
- **Requests:** The coupling analysis for Requests reveals a more varied distribution. The system has both very low and higher coupling in different modules. Some modules show significant dependencies (peaking at 10), which could complicate maintenance. On the whole, Requests seems moderately coupled.
- **Python-Twitter:** The coupling for Python-Twitter is consistently low, with most modules having a coupling value of 1 or 2. This suggests a well-structured design with minimal dependencies between modules, making it easier to change or extend.

From these results, I conclude that Python-Twitter has the most modular design in terms of coupling, followed by Requests and S3cmd, which have areas of higher dependency.

### Git Changes (Modifiability/Changeability)

Modifiability or changeability is a measure of how many files need to be changed during a typical git commit. A system with high modifiability allows changes to be local, affecting fewer files.

- **S3cmd:** The histogram for git changes shows a broad spread, with many commits affecting 1-3 files but some impacting up to 23 files. This indicates that while some changes are localized, certain features or bug fixes require touching multiple files, suggesting that the system's design may need better modularization in some areas.

- **Requests:** Requests has a similar pattern to S3cmd but shows more peaks in commits affecting 1-3 files. It also has some high-impact commits affecting up to 22 files. This means that Requests is somewhat modifiable, but there are still occasional changes that require widespread file modifications.
- **Python-Twitter:** Most commits in Python-Twitter only involve a single file, with very few exceptions. This suggests that the system is highly modifiable, and changes are typically localized to specific modules or files, making it easier to introduce new features or fix bugs.

I find that Python-Twitter demonstrates the highest modifiability, with most changes confined to individual files. Requests and S3cmd are similar, though both have instances of large, system-wide changes.

### Maintainability Index

Maintainability index measures how easy it is to maintain or change the code over time. Higher values indicate better maintainability.

- **S3cmd:** The maintainability scores for S3cmd show a wide range, with a few modules scoring a perfect 100, indicating excellent maintainability. However, there are also modules with very low scores, such as 0 and 12, which may indicate more problematic code that is difficult to maintain or change. This inconsistency points to varying levels of code quality across the project.
- **Requests:** Requests has a generally high maintainability index, with many modules scoring above 80, indicating that the project is relatively easy to maintain. There are some lower scores, such as 21 and 47, but these are the exception rather than the rule.
- **Python-Twitter:** Most modules in Python-Twitter score highly, with maintainability indexes in the 50-100 range. This suggests that the system is well-designed and relatively easy to maintain. However, there are a few modules with mid-level scores, but overall the project shows strong maintainability.

From these observations, I conclude that Requests and Python-Twitter both have good maintainability overall, while S3cmd has more variability, with some modules that may need refactoring to improve maintainability.

### Conclusion

In this comparative analysis of three open-source projects S3cmd, Requests, and Python-Twitter, I found that Python-Twitter consistently scores the highest in terms of structural quality. It exhibits low coupling, high modifiability, and good maintainability, making it a well-structured system overall. Requests performs well but shows more variance in coupling and

modifiability, while S3cmd, though it has some well-designed areas, demonstrates higher coupling and lower maintainability in certain modules. This analysis shows how these key metrics can provide insight into a project's long-term viability and ease of modification, which is essential for ongoing development and maintenance.

## Graphs







