

SYS-201/SEN-201 Software Engineering
Laboratory Exercise 6
Software Configuration Management

Throughout the lab, our primary focus was on gaining experience in version control Git skills, specifically utilizing the Git command line over graphical interfaces or integrated development environments (IDEs). This approach aimed to give us a deeper understanding of programming version control and collaborative software development. The lab helped us steer away from common mistakes such as adding unnecessary files, reading errors from the command line, and managing branches in Git.

The utilization of the Git command line forced us to place emphasis on the commands executed, making us place more awareness on Git command syntax and the directory formats. Reading informative messages in the command line, often limited in IDEs, became a crucial skill, offering insights into errors that might be harder to read and understand but carry crucial information. This emphasis on command line messages not only enhanced our proficiency but also provided a more comprehensive understanding of the development process.

The lab also changed the overall working conditions, having us work on different computers (and different operating systems in our case). This addresses the challenges of offsite work and adapting to Git usage on different computers. This practical approach mirrored real-world scenarios, preparing us for working environments where it is not always possible to meet with teammates in person or where everyone in the team uses a different operating system.

As the lab progressed, we went into other version control aspects, with a particular focus on branching and conflict resolution. Creating and managing branches became a key skill, allowing for parallel development efforts. This was a very important aspect because it is simply more effective for different members to resolve different problems in different but parallel branches. The experience of merging branches emphasized the necessity for clear communication within the team, as miscommunication in merging can have significant setbacks on the software development.

The final part of the lab involved repository cleanup, where unnecessary files or directories were removed using Git commands. This task reinforced our command knowledge and also taught us the importance of a well organized repository.

In summary, the lab introduced using Git through the command line and utilizing Git commands. It also helped us understand that the command line proves more insight in preventing errors and reading informative messages. We also tackled working in different environments and practiced important concepts like branching and conflict resolution. Using version control Git seemed to be quite intimidating and less useful than IDE's at first, but proved itself to be even simpler and more useful in developing software as a team overall after finishing this lab.