Alex Nielsen

CSE 210

4/28/23

Version Control

Version Control is the system or process used in virtually all programs that are being developed, especially by teams of people. It is the process of creating ‘copies’ of a version of a program, to then be modified by a developer, and then ‘pushing’ the new modified version into the actual program, thus updating it; This creates an easy way to manage changes in a program. This opens a wide world of possibilities for programmers… the most notable of which is the ease for teams to develop the same project at once. With one main program that several people can copy to their local computers, change, and then push to update the real code allows everyone to efficiently update programs and test them before it is implemented into the real program. Version control also includes saving old or previous versions of the code so that if there are errors or problems in the future, you can roll back to a previous working version. Another major benefit is the ability to create branches, which allows separate versions of the same code are being updated simultaneously, and later possibly merged. Likely the easiest example I could give is if a group of say 4 friends are working on developing a game. Using version control the actual game’s code is stored on a server of some kind, and each of the 4 friends can make a copy of the code from that server onto their own computer. They then modify the copy, test it, debug it, etc. Finally once it is working properly, they can push the new code onto the server, and now anytime the others make a copy it will ‘pull’ the changes made and they can continue to develop from the updated version. However, say a piece of code implemented a week earlier made a bug in a certain game mechanic, the group could restore the old working version and work back from there to prevent the bug from ever happening thanks to version control. A simple example of how someone would ‘push’ an updated copy of code to the server would be by first ‘staging’ the code or getting it ready to push with the command *git add ‘filename’*. They can then ‘commit’ the changes with a message that explains the purpose of the changes for documentation and future management, using *git commit -m “message here”*. Finally, to actually send the changes to the main code on the server, they would give the command *git push*.