**Detailed feature: prevent merge conflicts using local commits as indicators**

The idea of this feature is to gather information from every team member working on a same project. This information consists of monitoring the local machines as to when a commit is made. There might be cases where one or two local commits are made, and these have not been pushed to the cloud repository. Therefore, the other teammates cannot see these changes or what they consist of. If they could, merge conflicts could be avoided before they actually happen.

Every time a local commit happens, the information is sent to our middleware. The middleware contains a clone of the cloud repository and creates the same branches on which the clients are working on. With the local commit information, the branches are enriched and kept up-to-date. By using these branches to our advantage, we can test against every other branch for merge conflicts. If a conflict is found, the data is saved on our database (on which branch, on which commit, who is the branch owner).

Our goal with this feature is to be able to present a graph (git-graph) where every team member can see how far ahead their colleagues are from the HEAD of the cloud repository. The graph is the same for everybody. However, the leaves of the graph changes given who is the user that opened the graph. The user case is: “As a user, which other leaves (commits) are in conflict with me?”. By using this logic, the user can click on commits done by their teammates and assess if there is a conflict on that specified point. If so, which files are changed and where (which lines) the conflict is located.

An example of the shared graph can be seen here:

An example of the leaf visualization with the files changed can be seen here: