**Kerntaak 4.5: Manages content**

When setting up our developer environments, we decided to use Git and GitHub. The advantage of this is that all three of us could work on the same code without accidentally blocking progress of others (eg. by deleting code others had written). The disadvantage was that that meant that there were bound to be conflicts in the code which would have to be resolved by one person and merged back into the master branch.

Nevertheless we decided to push through that minor inconvenience and stick with this way of contributing code to the project, both for fixes and for completely new content. Erik would be the one resolving conflicts and merging code as he was always the one signing off on all the code anyway.

The added advantage was that all our developer environments included a way to easily push our changes to our GitHub repository: on Windows through the GitHub's own separate application and on Linux through the Android Studio IDE. This made receiving code a breeze as well, as you could just as easily pull code as you could push it. Having no other other ways to contribute code meant that there was no need to do any conversion work or complicated and time-hungry copy-pasting labour.

Our patches were not just our work, though. They were dependent on third parties. The developers of the original website had implemented new endpoints for our mobile app to connect to. The implemented end points were documented on a special website which included forms that would test the relevant calls and display what the end point would return. This way we could catalogue what data we would have access to before we actually implemented the call to the API into our application. All the necessary data was delivered to us in this fashion without fail.