Design document of VIRLab at Udel

This document aims to provide the key design patterns of VIRLab at University of Delaware. The document also includes the potential improvement of the current system.

1. **Web server, Database and backend codes**

   The VIRLab uses LAMP (Linux-Apache2-MySQL-PHP) as the basis of the backend server.

2. **The evaluation process**

   The evaluation process involves in the submission of the source code, the compilation of the source code, ranking using the submitted code and finally the evaluation of the ranking list. The problem of current system is that when the user of the system initiates a request of evaluating the ranking function against a specific collection (e.g. wt2g) the user interface freezes. This is because the backend is directly run the binary executable which is compiled from the source code. Apache is different from NodeJS, it just waits for the completion of the executable.

   A possible solution for this is to use Docker. The process can be described like this: when the user requests the evaluation of the ranking function, we start a Docker container. The Docker image on which the container is running contains the code base of the retrieval program but just without the actual ranking model. When the container starts it will first compile the binary executor and then evaluate the function against one collection that user has selected. Doing like this can avoid storing the binaries for each function. Using Docker is recommended also because of security reasons. We can choose which port(s) are opened to the hosting system and also confine the memory usage. After all, we have no idea what is in the submitted codes and they might cause potential damage to the server.

3. **The codes**

   At the beginning of the development, we (the initial developers) are not quite familiar with PHP so the PHP code is mixed with backend logic and the frontend UI. We think that it is better to separate them apart with APIs and JavaScript code which follows the RESTful API convention. While the APIs only response to the requests in JSON format, the JavaScript codes deals with the UI in the frontend. We believe this could make the logic of the code clearer. Also, this makes the codes more expandable if we are going to change the “frontend” to other media, e.g. phones.