# Functional Testing for uploadResources function

**Pre-condition violations:**

* The mime-type and size of the files to be uploaded are checked incorrectly due to logic errors in the coding.

**Post-condition violations:**

**Data structure requirements:**

# Non-functional testing / assessment

**The use case was tested against the following list of architectural requirements:**

**Usability:**

Usage of this function is fairly easy and obvious. The selected files from the file selector only need to be sent through to the function and it handles all the rest of the heavy lifting. It can handle multiple files at once and tests each file individually.

**Performance:**

The performance of this function would rely on network speed and stable connection to the database, since it does not require much computational time or pc resource allocation.

**Scalability:**

This function allows multiple files to be uploaded together, with no specified limit of files. The size of each file is limited by the relevant mime-type entry in the database, thus, theoretically, any file size should be able to be uploaded, if the database permits it. Furthermore, multiple users can use the function concurrently, because of the way the server can be set up.

**Testability:**

The testability of this function is not hard at all, since it allows only a certain number of things to happen. If a file is not allowed, it shouldn't be uploaded, if it is, it should. If the file size is too large, likewise. It is quite easy to send a query to the database to see if the correct information was stored by the function, thereby further enhancing the testability thereof.

**Security:**

This function does not handle any type of security (except not allowing harmful mime-types). The connection to the database, for instance, is created somewhere else and the database object is only used by this function.

**Reliability:**

In the cases tested, the function is not reliable as it should be. It allows multiple files to be uploaded, as it should, however - it allows ALL files of any size to be uploaded due to a logic error in the code. The size is tested the wrong way around (a test is done to see if the max size is smaller than the file size, which is wrong). Not only that, a function is used to count the amount of records found, but the object containing all the records found cannot be "counted" in the way the coder wanted to.

**Reusability:**

This function can easily be reused, since it keeps no remnants of previous data that can possibly corrupt new data. Each use of the function is thus a clean slate, so to speak.

**Portability:**

This function is portable to the extent that mongoose is used as an interface to connect to the database. When that requirement is complied with, the function will be portable (it will accept file data and write it to the connected database).

# Critical Evaluation and Recommendations