# Internship task - requirements

**CSV Upload - Task**

* Divide the CSV file into multiple parts with around 5-10MB each
* For upload use correct class hierarchy - Control, Service and Parser classes
* Try to use bulk inserts and check the performance difference
* Use proper error handling - if there is one corrupted raw the entire process should not crash, just process the valid lines and return errors

**File Upload/Download API**

Here are the main steps for **file upload**.

* Create a new folder outside the project directory and set the folder path in application properties
* Create a new table in the database for storing documents/files meta-information, do not store the entire file in DB. Here are some key fields: id, type (profile\_photo, author\_photo, book\_cover, path, extension, file\_name, created\_date)
* When file is uploaded save file in the FileSystem, then insert the meta information into storage table and return the storage\_id
* In books or authors table add new column for example book\_cover\_image\_id and store that id only, which should be also Foreign Key to storage table

**For file download**

* Create a new FileController (“/files”) or ResourceController (“/resources”)
* Add one GET method only to request a file by id “/files/{id}”
* In this case we don’t care what type of file this is (author, book or profile), we directly make a call to StorageService and ask for file information
* Storage service will read the file from disk and send the Stream
* From control just write File into response

Check some example here: <https://dzone.com/articles/java-springboot-rest-api-to-uploaddownload-file-on>