Task: Implement a sample FileService REST-API & Implementation

Motivation:
As part of the distributed platform that we are currently establishing, we want to be able to provide various (micro-)services on different nodes that interact with each other.
One of those serves a large number of files from a distributed file storage backend (such as HDFS).

Other services use this service as their file backend so that they are able to store various data centrally. Also, the file service is meant to be a distributed service, so other file service nodes could take over responsibilities in case the first node goes down.

While it's certainly possible to access a distributed file storage directly without a service in-between, it would require to use either client libraries, rely on the API stability of a thirdparty and/or would require extending functionality for unsupported features such as mandantory access control.
Therefore serving files from the storage using a REST service seems to be a viable vendor agnostic alternative.

Your Task:
In this task we want you to do a demo-implementation of a small subset of that functionality, namely a part of the REST-API that other services can use to perform file operations.

The task is not necessarily meant to be completed by you, it's more important that we can see how you would work on this task, as it includes designing the API itself, the implementation of said API and possibly stating decisions if needed.

It's possible to document your API design and decisions in code, but you can also provide other documents if you feel this might be better here.

We prepared a DropWizard Maven project that can serve as the base to the implementation part of the task. You can modify it in any way that you feel is needed, e.g. changing names, adding/removing modules etc.
As you are more familiar with Spring Boot, you can also opt for using that instead (and not to use the demo project, although we would prefer that) for the implementation, but in this case please try to stick to their JSR-311 implementation to avoid too much of a vendor lock-in to Spring.

- Develop the REST-API to perform various file operations. Each operation might use the provided backend service (dummy) implementation to perform the task. If you want to add functionality you feel is needed, you are free to do so.
- It's not needed to expose every function of the backend service as API, you also should decide how to expose the functionality, e.g. what data types or formats suit the functionality best.
- Provide some tests that show how you would do unit-tests and client/server integration tests. It's not required to provide a certain test coverage, it's sufficient to show how you would address this requirement.
- You should think about authentication and authorization when designing the API, but for the sake of simplicity of the task,
it would be enough to state how you would address this in a distributed environment or provide a dummy implementation
- Additionally, you can provide a REST client that shows the API usage or alternatively command-line operations.