



Bachelor's Thesis

Providing a distributed file storage for German Schul-Cloud

**Erstellung einer verteilten Dateiverwaltung für die deutsche
Schul-Cloud**

by

Niklas Kiefer

Potsdam, July 2017

Supervisor

Prof. Dr. Christoph Meinel,
Jan Renz

Internet-Technologies and Systems Group

Disclaimer

I certify that the material contained in this dissertation is my own work and does not contain significant portions of unreferenced or unacknowledged material. I also warrant that the above statement applies to the implementation of the project and all associated documentation.

Hiermit versichere ich, dass diese Arbeit selbständig verfasst wurde und dass keine anderen Quellen und Hilfsmittel als die angegebenen benutzt wurden. Diese Aussage trifft auch für alle Implementierungen und Dokumentationen im Rahmen dieses Projektes zu.

Potsdam, 26. Mai 2017

(Niklas Kiefer)

Inhaltsverzeichnis

1. Einleitung	1
2. Vorbetrachtung	2
3. Konzept	3
4. Implementierung	4
5. Evaluierung	5
6. Zukünftige Arbeit	6
7. Zusammenfassung	7
Literatur	8
A. Appendix	9

1. Einleitung

Die deutsche Schul-Cloud soll “dabei helfen, die digitale Transformation in Schulen zu meistern und den fächerübergreifenden Unterricht mit digitalen Inhalten zu bereichern” [2]. Ein Schwerpunkt ist hierbei das Arbeiten mit verschiedenen Dateiformaten im Unterrichtskontext. Dazu soll die Schul-Cloud eine Möglichkeit schaffen, eigene Dateien zu verwalten und sie unter Lehrern und Schülern zu teilen. Die Bildung einer Dateiablage ist nach neueren Analysen fester Bestandteil von “technisch-organisatorische Kernanforderungen”[1]. für den digitalen Unterricht. Dafür soll keine neue Dateiverwaltungstechnologie entworfen werden, sondern auf bestehende Systeme zurückgeführt werden. Viel mehr soll für den Nutzer der Schul-Cloud möglich sein, ein bestehendes Dateisystem weiter zu nutzen und in die Schul-Cloud zu integrieren. Außerdem soll es möglich sein, Dateien auf verschiedenen Systemen zu lagern, um eine flexible Architektur zu schaffen. Dies ist vor allem dann nötig, wenn große Datenmengen von sehr vielen Schulen, Schülern und Lehrern entstehen. Dann ist es wichtig, eine Lösung in Form von verteilten FileStorage zu finden.

In dieser Bachelor-Arbeit wird eine solche Architektur für ein verteiltes Dateiverwaltungssystem beschrieben und entworfen. Begonnen wird mit einer **Vorbetrachtung** (2), in welcher sich über die bestehende Situation von Dateiverwaltung im Schulunterricht auseinander wird. Danach wird ein **Konzept** (3) für ein verteiltes System beschrieben und anschließend eine mögliche **Implementierung** (4) aufgezeigt. In dieser wird eine Anbindung von FileStorage-Providern im Schul-Cloud Server und Client beschrieben. Diese Anbindungsmöglichkeit wird anschließend **evaluiert** (5). Dann werden mögliche **Anschlusszenarien** (6) aufgezeigt, inwieweit dieses Konzept ausgebaut werden kann. Zum Ende wird die Arbeit **zusammengefasst** (7).

2. **Vorbetrachtung**

3. Konzept

4. Implementierung

5. Evaluierung

6. Zukünftige Arbeit

7. Zusammenfassung

Literatur

- [1] ANDREAS BREITER, BJÖRN ERIC STOLPMANN, ANJA ZEISING: *Szenarien lernförderlicher IT-Infrastrukturen in Schulen.*
- [2] CHRISTOPH MEINEL, JAN RENZ, CATRINA GRELLA NILS KARN CHRISTIANE HAGEDORN: *Die Cloud für Schulen in Deutschland: Konzept und Pilotierung der Schul-Cloud*, March 2017.

A. Appendix

```
1 class AbstractFileStorageStrategy {
2   constructor() {
3     if (new.target === AbstractFileStorageStrategy) {
4       throw new TypeError("Cannot construct AbstractFileStorageStrategy instances directly
5         .");
6     }
7   }
8
9   create() {
10    throw new TypeError("create method has to be implemented.");
11  }
12
13  getFiles() {
14    throw new TypeError("getFiles method has to be implemented.");
15  }
16
17  deleteFile() {
18    throw new TypeError("deleteFile method has to be implemented.");
19  }
20
21  generateSignedUrl() {
22    throw new TypeError("generateSignedUrl method has to be implemented.");
23  }
24
25  createDirectory() {
26    throw new TypeError("createDirectory method has to be implemented.");
27  }
28
29  deleteDirectory() {
30    throw new TypeError("deleteDirectory method has to be implemented.");
31  }
32 }
33 module.exports = AbstractFileStorageStrategy;
```

Listing 1: Abstract file storage strategy