

# Anwendungsszenarien

## Materialtausch

### Inhalt

|   |   |
|---|---|
| 1. Dateitypen .....                       | 1 |
| 1.1. Software .....                       | 1 |
| 1.2. Skripte & Arbeitsblätter .....       | 2 |
| 1.3. Präsentationen .....                 | 2 |
| 2. Szenarien für den Materialtausch ..... | 2 |
| 2.1. Fortbildner untereinander .....      | 2 |
| 2.2. Fortbildner - Lehrer .....           | 2 |
| 2.3. Lehrer - Lehrer .....                | 3 |

Für den Informatikunterricht gibt es eine Vielzahl von Fortbildungsmaterialien, die zum Teil von den Fortbildnern entwickelte Software oder Programmieranwendungen für den Unterricht enthalten. Diese werden fortlaufend weiterentwickelt und Fehler bereinigt.

GIT bietet die Möglichkeit, eine professionelle Versionsverwaltung dafür zu realisieren. Alle Lehrenden haben nun jederzeit die Möglichkeit, sich über neue Versionen zu informieren und die neuste Version abzurufen.

Lehrende soll auf strukturierte Weise Projekte aus der Lehrerfortbildung weiterentwickeln können und so zu der Weiterentwicklung der Unterrichtsmaterialien beitragen können.

Auch eigene, erprobte Projekte sollen von jedem Lehrenden anderen Lehrenden angeboten und getauscht werden können.

All dies kann mit GIT realisiert werden.

## 1. Dateitypen

GIT arbeitet textbasiert. Daher bieten sich textbasierte Dateiformate an, um eine effektive Versionskontrolle gewährleisten zu können.

### 1.1. Software

Bei Softwareprojekten ist dies kein Problem, da diese ohnehin alle textbasiert sind.

## 1.2. Skripte & Arbeitsblätter

Als Format für Skripte sollte in Zukunft ASCII-Doc verwendet werden. ASCII-Doc bietet die notwendigen Formatierungsbefehle für die üblichen Textgestaltung und entwickelt sich zu einen Standard weiter. Markdown wäre eine Alternative, die aber aufgrund vieler verschiedener Markdown-Dialekte problematischer ist. Außerdem werden Dokumente im ASCII-Doc Format auf Git-Camp BW automatisch gerendert und formatiert angezeigt.

## 1.3. Präsentationen

Präsentationen können durch reveal.js einfach im HTML-Format erstellt werden. Dadurch ist eine Versionskontrolle mit GIT möglich. Die Darstellung kann auf jedem Endgerät erfolgen, ist responsive-fähig. Die Präsentation kann auch direkt auf der Webseite von Git-Camp BW angezeigt werden. Ohne Download steht immer die neuste Version bereit.

# 2. Szenarien für den Materialtausch

## 2.1. Fortbildner untereinander

Für jedes Projekt gibt es einen oder mehrere **hauptverantwortliche Fachberatende**. In einem Development-Branch entwickeln diese die Projekte und arbeiten Änderungswünsche ein.

Sie besitzen Lese- und Schreibrechte auf den Repositorys, die auf dem **Git Camp Fortbildner** angelegt werden. Alle anderen FBUs können Repositorys forken und Verbesserungen einbauen. Die Hauptverantwortlichen entscheiden, ob diese übernommen werden.

Die Hauptverantwortlichen arbeiten größere Änderungen in separaten Branches ein und bitten andere FBU um die Begutachtung der Änderungen. Ggf. wird der Branch mit dem Hauptbranch gemerged. Kleinere Bugfixes erfolgen auf dem Hauptbranch.

Davon abgeleitet wird ein Branch zum Veröffentlichen, der dann mit Versionsnummern versehen wird. Jede zu veröffentliche Version wird in ein zweites Repository kopiert, das einen Deployment-Branch enthält und auf dem **GitCamp Lehrerfortbildung** verfügbar gemacht wird, so dass alle Lehrpersonen darauf zugreifen können. Dazu müssen die Repositorys so eingestellt werden, dass die GRUPPE XXX lesenden Zugriff auf das Repository hat.

Alle Arbeitsversionen während des Development-Prozesse bleiben daher den Lehrenden verborgen. Daher sind Urheberprobleme von Zwischenversionen ein geringeres Problem.

## 2.2. Fortbildner - Lehrer

Die Materialien aus Lehrerfortbildungen (insbesondere Softwareprojekte) werden von den Fachberatern im **GitCamp Lehrerfortbildung** bereit gestellt. Das Repository XXX enthält eine Übersicht über alle bereitgestellten Projekte mit der Zuordnung zu den möglichen Einsatzbereichen. Links verweisen auf die Repositorys mit den dazugehörigen Materialien.

Jede Lehrperson sollte diese Repositorys

1. clonen, wenn sie die Projekte nutzen möchte und ihren SuS zur Verfügung stellen möchte. Dazu fügt sie dem Projekt ein weiteres Remote-Verzeichnis hinzu, das im **schuleigenen GitCamp** liegt und pushed dort das Projekt.
2. forken, wenn sie vor hat, das Projekt zu verändern (Verbesserungen einzuprogrammieren) und diese Veränderungen der Allgemeinheit zur Verfügung zu stellen. Nachdem die Veränderungen eingefügt wurden, stellt die Lehrperson einen Pull-Request für das ursprüngliche Repository mit einer aussagekräftigen Beschreibung, welche Änderungen eingefügt wurden. Der betreuende Fachberater entscheidet, ob die Änderungen in das Originalprojekt übernommen werden soll.

**Fehler melden:** Finden Lehrpersonen Fehler (Schreibfehler, inhaltliche Fehler, Bugs) in den Materialien können sie Issues im normalen Projekt anmelden, so dass der betreuende Fachberater diese korrigieren kann. Auch Änderungswünsche können auf diesem Wege angezeigt werden.

## 2.3. Lehrer - Lehrer

Jede Lehrperson hat jederzeit die Möglichkeit eigene Repositorys im **GitCamp Lehrerfortbildung** anzulegen. Sie kann einzelnen anderen Lehrpersonen Leserechte oder allen Lehrpersonen (GRUPPE XXX) auf diese Repositorys einräumen, so dass diese die Repositorys für den eigenen Unterricht nutzen sollen.

Soll alle Lehrpersonen Zugriff erhalten, müssen die Fachberater informiert werden und gebeten werden, einen Link auf der Übersichtsseite mit allen Projekten zu erstellen.