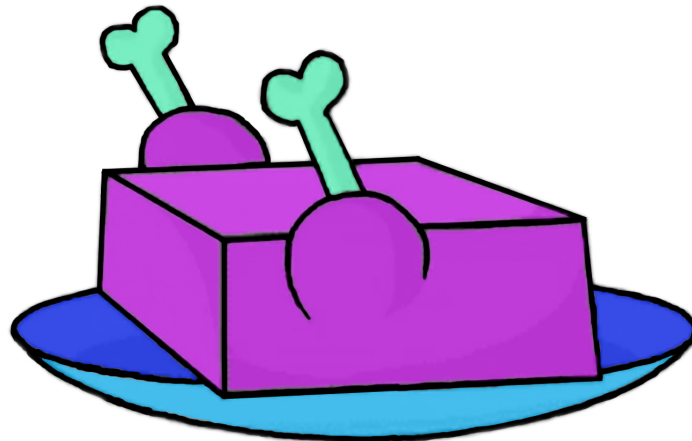


Projekt: kitovu
Anforderungsspezifikation



Florian Bruhin
florian.bruhin@hsr.ch

Méline Sieber
meline.sieber@hsr.ch

Nicolas Ganz
nicolas.ganz@hsr.ch

4. März 2018

Änderungsgeschichte

Datum	Version	Änderung	AutorIn
04.03.2018	1.0	Dokument erstellt, Grundgerüst von Template übernommen	Méline Sieber

1 Einführung

Dieses Dokument beschreibt, was *kitovu* genau ist. Danach veranschaulicht diese Anforderungsspezifikation, für wen der *kitovu*-Client gedacht ist. Das erfolgt anhand von "Use Cases", einerseits in einem kurzen Format (*brief*), andererseits in einer ausführlichen Beschreibung (*fully dressed*). Weitere Anforderungen detaillieren, welche Qualitätsmerkmale und Schnittstellen verwendet werden sowie geltende Randbedingungen.

1.1 Gültigkeitsbereich

Die vorliegende Anforderungsspezifikation ist für das Engineering Projekt im Frühlingssemester 2018 gültig. Falls dem Projekt grössere Veränderungen widerfahren, wird das Dokument dementsprechend angepasst. Umfassende Änderungen werden am Anfang des Dokuments protokolliert.

1.2 Referenzen

Die Anforderungsspezifikation ist eng verbunden mit der Domainanalyse und anderen Dokumenten. Die folgende Tabelle listet die wichtigsten Referenzen auf.

FIXME Verweise zum Projektplan einfügen, zur Domainanalyse u.a. Dokumenten, die wir mit der Anforderungsspezifikation abgeben

Domainanalyse	FIXME
Confluence	https://jira.keltec.ch/wiki
Github-Repository von <i>kitovu</i>	https://github.com/kitovu-bot/kitovu
JIRA	https://jira.keltec.ch/jira
Moodle	https://moodle.hsr.ch
OpenHSR Connect	https://github.com/openhsr/connect
Studentenportal	https://studentenportal.ch/
Switch AAI	https://www.switch.ch/aai/

Beim Logo auf der Titelseite handelt es sich um eine stark überarbeitete Version eines GIFs (https://www.animateit.net/details.php?image_id=8990). Urheber und Copyright waren nicht auffindbar.

2 Allgemeine Beschreibung

2.1 Produktperspektive

2.2 Produktfunktion

Kitovu ist ein Client, der von verschiedenen Plattformen ausgewählte HSR-Unterrichtsmaterialien auf den eigenen Rechner synchronisiert. Er läuft auf allen gängigen Betriebssystemen und funktioniert nicht nur für den HSR-Skripteserver, sondern ist auch erweiterbar für verschiedene Plattformen.

Unser Projekt bindet primär den Skripteserver ein. Der Terminal-basierte Client funktioniert mittels Profilen zu unterschiedlichen Plattformen (Moodle, Skripteserver, Studentenportal). Pro Profil sind Verbindungsdaten und eventuelle Login-Credentials im Client hinterlegt. Die Daten-Synchronisation erfolgt immer nur von Server zu Client, erfolgreiche und misslungene Datentransfers werden protokolliert. Ein rudimentäres GUI dient als Proof-of-Concept.

Pro Profil lässt sich Folgendes definieren:

- welche Ordner/Dateien synchronisiert werden.
- welche Ordner/Dateien von der Synchronisation ausgeschlossen werden.
- wie mit Duplikaten/lokal bestehenden Dateien umgegangen wird.

Kitovu ist ausbaubar und damit modular: Zusätzlich zu den beiden Plattformen (Skripteserver; Moodle oder Studentenportal) können in zukünftigen Projekten beliebig viele Plattformen als separates Plugin bzw. Profil realisiert werden.

Optionale Features:

- Moodle und/oder das Studentenportal. Die Implementation von Moodle oder des Studentenportals ist abhängig von den weiter unten beschriebenen Risiken.
- Komplettes GUI, das der Funktionalität des Terminalprogramms entspricht.

2.3 Benutzercharakteristik

Studentinnen und Studenten verwenden *kitovu*, um ihre Unterrichtsmaterialien auf ihren Rechnern à jour zu halten. Sie verwenden verschiedene Betriebssysteme (Windows, macOS, Linux). Ihre Erfahrung mit der Kommandozeile ist unterschiedlich; manche verwenden sie nie, andere benutzen sie zur Standardinteraktion mit dem Betriebssystem.

Dozentinnen und Dozenten können ebenfalls *kitovu* verwenden, sie sind jedoch nicht die primäre Zielgruppe.

2.4 Einschränkungen

Da die Projekt-Zeitspanne kurz ist, ist die Kernfunktion von *kitovu* ein Client, der über die Kommandozeile bedient wird. Es ist vorerst nur eine rudimentäre grafische Benutzeroberfläche geplant. Falls jedoch genügend Zeit bleibt, baut das Team diese aus. Uns ist bewusst, dass wir damit einen Teil der Studierenden ausschließen, nicht alle können mit der Kommandozeile umgehen. Aufgrund der Zeitbeschränkung müssen wir das in Kauf nehmen, sehen es aber als erste Priorität, die wir implementieren, wenn bis zum Ende des Projekts genügend Zeit bleiben sollte.

Eine weitere Einschränkung sind das Studentenportal und Moodle. Die Gründe dazu beschreibt bereits der Projektplan ausführlich.

2.5 Annahmen

Eine wichtige Voraussetzung sind bereits bestehende Accounts. Wir gehen davon aus, dass die Studierenden bereits einen HSR-Account besitzen und sich damit sowohl per VPN von zu Hause als auch direkt an der HSR mit dem Skripteserver verbinden können. Das Studentenportal¹ verlangt einen separaten Account, der Zugriff auf Moodle² erfolgt via Switch-AAI, der Authentisierung- und Autorisierungsschnittstelle für alle Hochschulen³.

Wir müssen annehmen, dass ein Teil der Studentinnen und Studenten darin vertraut ist, die Kommandozeile zu bedienen. Die Konfiguration von *kitovu* erfolgt über ein einzelnes Konfigurationsfile - wir müssen ebenfalls davon ausgehen, dass die Studierenden damit umgehen können.

2.6 Abhängigkeiten

Kitovu steht und fällt mit der Anbindung an die Plattformen, also den HSR-Skripteserver sowie die optionalen Plattformen Moodle oder Studentenportal.

¹<https://studentenportal.ch/>

²<https://moodle.hsr.ch>

³<https://www.switch.ch/aai/>

3 Use Cases

3.1 Use-Case-Diagramm

3.2 Aktoren und Stakeholder

3.3 Beschreibungen (Brief)

3.3.1 jUse Case Name

3.4 Beschreibungen (Fully Dressed)

3.4.1 Use Case Name

4 Weitere Anforderungen

4.1 Qualitätsmerkmale

4.2 Schnittstellen

4.3 Randbedingungen