

Dokumentation Mobile Uni-App

Version: Phase 1.5

Autor: Florian Ickelsheimer (florian.ickelsheimer@student.unisg.ch)

Achtung! Diese Dokumentation ist als Ergänzung zu der von Modulabs offiziellen [Kurogo Documentation](#) zu sehen.

Inhaltsverzeichnis

- [Dokumentation Mobile Uni-App](#)
- [Inhaltsverzeichnis](#)
- [1. Modifizierte Dateien am Kurogo-Framework](#)
- [2. Schnittstellenbeschreibungen](#)
 - [2.1 Vorhandene Module](#)
 - [2.2 Weitere Standard-Module](#)
 - [2.3 Liste von Kurogo v1.3 unterstützten Schnittstellen](#)
- [3. Beschreibung eigener Module](#)
 - [3.1 Classes](#)
 - [3.2 Directory](#)
 - [3.3 Maps](#)
 - [3.4 Transport](#)
 - [3.5 Mensa](#)
 - [3.6 Emergency](#)
 - [3.7 Alumni](#)
 - [3.8 Mail](#)
 - [3.9 Login](#)
- [4. Weiteres](#)
 - [4.1 Module](#)
 - [Admin](#)
 - [Stats](#)
 - [News](#)
 - [Links](#)
 - [4.3 Beispielcode und -funktionen](#)
 - [4.4 Google Analytics](#)
 - [4.5 Add2Homescreen](#)
 - [4.5 Zugriffsrechte auf Server](#)
- [5. Unterstützte Geräte](#)
 - [5.1 Bekannte Issues](#)
 - [5.2 Neue Funktionalität](#)
- [6. Systemvoraussetzungen](#)

1. Modifizierte Dateien am Kurogo-Framework

Grundsätzlich müssen keine Änderungen am Kurogo-Framework vorgenommen werden. Änderungen können im eigenen Site-Folder gemacht werden. Sollten Kurogo-libraries (/lib-Ordner) überschrieben werden müssen, dann kann in einem optionalen /lib-Ordner unter dem /site/eigenesite-Ordner angelegt werden (siehe [Dokumentation Source Tour](#)). *Hinweis:*

Aktuell wurde dies noch nicht ausführlich getestet oder angewendet. Bitte immer Kurogo-Dokumentation lesen.

Falls doch Änderungen (vielleicht auch provisorische) vorgenommen werden, sind Ordner und Dateien sowie geänderte Zeilen nachfolgend aufgelistet:

- \lib
 - ...

Anmerkung: Bevor Änderungen vorgenommen werden, lib-Dateien genau anschauen. Ab und zu werden Methoden angeboten, die Kurogo selber nicht verwendet, aber durchaus nützlich sein können.

2. Schnittstellenbeschreibungen

2.1 Vorhandene Module

Modulname	Aktuelle Schnittstelle	Vorschlag für bessere Anbindung
News	RSS (Standard)	
Transit (Transport)	XML-Parser/DOM (fahrplan.search.ch)	Offene API von SBB oder Google (JSON)
Classes (Kalender)	Aktuelle Kurse: XML-Parser/DOM http://www1.unisg.ch/org/hsg/infosystem.nsf/LecturesToday?openagent Öffentliches Angebot: XML-Parser/DOM (http://www1.unisg.ch/org/hsg/zevents.nsf/unisgintegration?openagent) Aktuelle Events (Login): Externer Link Termin finden: Externer Link Wichtige Daten: Lokal gespeichertes Bild	
Directory (Verzeichnis)	XML-Parser/DOM (http://www1.unisg.ch/www/telefon.nsf/)	LDAP-Verzeichnis oder Active Directory
Map	Lokal gespeicherte Google Earth-Files Lokal gespeicherte CampusMap (http://www.unisg.ch/~media/Internet/Content/Dateien/UeberUns/AnfahrtCampusplan/HSG-Campusplan-Mai-2011.ashx?fl=de) und Raumpläne	
Emergency (Notfallinfos)	Lokal gespeicherte Bilder, lokal gespeicherte INI mit Notfallnummern unter /data	
Alumni	Externer Link	---
Mail	Externer Link (nur über Login)	Login-Daten müssen mitgeschickt werden, falls man eingeloggt ist
About	Lokal gespeicherte INI	---
Login	Lokal gespeicherte passwd-Datei	AAI-Schnittstelle OAuth, LDAP, Active Directory (für weitere Nutzung Google, Twitter und Facebook)

Mensa	Externer Link auf Mensapläne (PDF)	XML-Schnittstelle
-------	------------------------------------	-------------------

2.2 Weitere Standard-Module

Modulname	Aktuelle Schnittstelle	Vorschlag für bessere Anbindung
Events	---	iCalendar (ICS), Google
Video	---	JSON (YouTube, Brightcove, Vimeo)
Admissions	---	XML/HTMLParser
Download	---	Externe Links (ini-Files)

2.3 Liste von Kurogo v1.3 unterstützten Schnittstellen

<http://modolabs.com/kurogo/guide/libs.html>

- DOM (HTML und XML)
- ICS (Kalender)
- JSON (Web-Services, insbesondere Google Services)
- Passthroug (einfache Strings)
- RSS

3. Beschreibung eigener Module

3.1 Classes

Beschreibung: Ursprünglich als reine Kursansicht gedacht, mittlerweile für aktuelle Veranstaltungen, öffentliche Veranstaltung und wichtige Termine.

Erbt von: WebModule

Funktionen:

1. index (Index):
 - a. Array aus module.ini erzeugen
 - b. Navlist-Array erstellen
2. current (Aktuelle Veranstaltungen):
 - a. DOM-Element aus XML erzeugen
 - b. Durchsuchen aller div-Elemente
 - c. Setzen von Variablen nach Reihenfolge: Zeit, Raum, Titel, Dozent, Stufe
 - d. Ergebnis-Array erstellen
3. public (Öffentliche Veranstaltungen):
 - a. Heutiges Datum bestimmen und URL erzeugen
 - b. DOM-Element aus HTML erzeugen
 - c. Events durchsuchen
 - d. GermanContent durchsuchen
 - e. Setzen von Variablen nach Tags: Subject, DateFrom, TimeFrom, TimeUntil, Room, Speaker
 - f. Ergebnis-Array erstellen
4. Aktuelle Termine: Externer Link über module.ini
5. Doodle: Externer Link über module.ini
6. dates (Weitere Termine):
 - a. Einbindung eines Liedes aktuell über dates.tpl

3.2 Directory

Beschreibung: Personenverzeichnis

Erbt von: WebModule

Funktionen:

1. index (Index):
 - a. Array aus module.ini erzeugen
 - b. Navlist-Array erstellen
2. results (A...Z)
 - a. Informationen aus Breadcrumb-URL holen, Titel neu setzen und URL erzeugen
 - b. DOM-Element aus XML erzeugen
 - c. Durchsuchen des a-Elements
 - d. Navlist-Array erzeugen
3. detail (Detailseite/Name):
 - a. Informationen aus Breadcrumb-URL holen
 - b. DOM-Element aus XML erzeugen
 - c. Informationen aus Tabellen-Elemente suchen (tr, td)
 - d. Variablen setzen

3.3 Maps

Beschreibung: Maps, Campus Map, Raumpläne, Link zu Google,

Link zu Kurogo Dokumentation: <http://modolabs.com/kurogo/guide/modulemap.html?highlight=zoom>

Erbt von: MapWebModule (derzeit komplette PHP und alle tpl-Dateien übernommen, ungebrauchte können gelöscht werden, typisches Überladen der PHP nur schwer möglich, da private Methoden nicht vererbt werden)

Funktionen:

1. Verschiedene Maps in entsprechende KMZ-Datei (Google Earth) unter /data abgelegt
2. Info-Feld als Etagen (komma-getrennt ohne Leerzeichen) genutzt
3. Campus-Map und Raumpläne werden als Bild im entsprechende /themes-Ordner abgelegt
4. Links-Tab ist vordefiniert von Kurogo
5. initializeForPage()
 - a. floor (Raumpläne):
 - i. Informationen aus Breadcrumb-URL holen, Titel neu setzen
 - ii. Ordnerpfad aus module.ini holen
 - iii. Bilddatei holen und an floor.tpl übergeben
6. generateTabForKey():
 - a. für jedes TabKey gibt es einzelnen Case; TabKeys in page-detail.ini
 - b. eigene Tabs:
 - i. rooms:
 1. String mit getFields() aus Info-Bereich eines Ortes holen (in Google Earth)
 2. String in Array umwandeln (komma-getrennt)
 3. Navlist-Array erzeugen
 4. Return-Wert, ob Floors vorhanden sind (falls nicht, wird das Tab nicht angezeigt)
 5. detail.tpl anpassen (neuer Block {if in_array...}, Achtung: Nomenklatur muss stimmen, somit keine Umlaute/ Sonderzeichen möglich)
 - ii. campus:
 1. Return-Wert: aktuell immer true, somit wird immer die Campus-Map als Tab angezeigt
 2. detail.tpl anpassen (neuer Block {if in_array...}, Achtung: Nomenklatur muss stimmen, somit keine Umlaute/ Sonderzeichen möglich)

3.4 Transport

Beschreibung: Reiseauskunft, Busfahrplan, Taxis

Erbt von: WebModule

Funktionen:

1. index:
 - a. Array aus module.ini erzeugen
 - b. Navlist-Array erstellen

2. fromto (Reiseauskunft):
 - a. fromto:
 - i. Array aus module.ini erzeugen
 - ii. Navlist-Array erstellen, setzen von Variablen Richtung und String „Universität St. Gallen“
 - b. form (Formular zur Dateneingabe):
 - i. form:
 1. Arrival-Array (Ankunft/Abfahrt) aus module.ini holen
 2. Informationen aus Breadcrumb-URL holen und Titel setzen
 3. Vordefinierte Felder an form.tpl übergeben
 4. Formular-Felder in form.tpl definieren und Daten per Post übergeben
 - ii. transport (Ergebnisse von fahrplan.search.ch):
 1. String-Konstanten festlegen (besser: in module.ini speichern)
 2. Daten aus Post holen
 3. Eingabefehler überprüfen
 4. URL-erzugen (aus Basis-URL, Von/Nach, Post-Eingaben) und eventuell erweitern (Arrival-Modus: Ankunft/Abfahrt)
 5. DOM-Element aus HTML (fahrplan.search.ch) erzeugen
 6. Tabellen nach Elementen durchsuchen, Attribut class =
 - a. oev_tripoverview (Abfahrts- und Endzeit)
 - b. oev_duration (Dauer)
 - c. oev_types (Umstiege und Fusswege berechnen)
 - d. oev_detail (Genaue Details; Link auf Seite)
 7. Ergebnis-Array erstellen (Links verweisen auf Detailseite auf fahrplan.search.ch)
3. busschedule (Busfahrplan):
 - a. URLs festlegen (besser: in module.ini speichern)
 - b. getBusContentFromDom() für alle Bushaltestellen
 - i. DOM-Element aus HTML erzeugen
 - ii. Tabellen nach Elementen durchsuchen, Attribut class =
 1. tt1 (Busnummer)
 2. tt3 (Zeit)
 3. tt4 (Richtung)
4. taxis (Taxirufnummern):
 - a. Array aus module.ini erzeugen
 - b. Navlist-Array erstellen

3.5 Mensa

Beschreibung: Mensapläne

Erbt von: WebModule

Funktionen:

1. index (Mensapläne)
 - a. URL aus module.ini holen (<http://www.unisg.ch/~media/Internet/Content/Dateien/UeberUns/Services/Gastronomie/>)
 - b. Heutiges Datum erzeugen
 - c. Mensapläne der nächsten 4 Wochen als PDF auf unisg-Webseite
 - d. Achtung: Neujahr-Problem und Integer-Workaroud

- e. Navlist-Array erzeugen

3.6 Emergency

Beschreibung: Maps, Campus Map, Raumpläne, Link zu Google

Link zu Kurogo Dokumentation:

Erbt von: EmergencyWebModule

Funktionen:

1. index (Link zu Sicherheitsinfos und Notfallnummern)
 - a. index (Notfallnummern)
 - i. Überladen der initializeForPage()-Methode, um Kontakt-Array aus /data/emergency/hsg.ini zu holen
 - ii. Überladen von contactNavListItem()
 1. Verwendung von getPhone() (direkter Wert aus .ini) anstatt von getDiablePhone() (angepasst auf US-Telefonnummern)
 - b. fire
 - i. Bild-URL im fire.tpl, Bild befindet sich im /themes-Ordner
 - c. security
 - i. Bild-URL im security.tpl, Bild befindet sich im /themes-Ordner

3.7 Alumni

Beschreibung: Anbindung der externen URL von Alumni

Erbt von: WebModule (Überprüfung, ob von UrlWebModule erben sollte)

Funktionen:

- Externe URL in module.ini

3.8 Mail

Beschreibung: Anbindung der externen URL von Mail (LotusNotes Client, Login erforderlich)

Erbt von: WebModule (Überprüfung, ob von UrlWebModule erben sollte)

Funktionen:

- Externe URL in module.ini

3.9 Login

Beschreibung: Erlaubt Login

Erbt von: ---

Funktionen:

- User-Accounts in passwd-Datei abgespeichert
- Gruppen können in groups-Datei abgespeichert werden
- Passwort muss jeweils MD5-verschlüsselt sein

4. Weiteres

4.1 Module

Admin

Link: <http://modolabs.com/kurogo/guide/configuration.html#administration-module>

Admin-Oberfläche zum Ändern sämtlicher Systemeinstellungen durch Zugriff auf ini-Dateien im /site/HSG/config-Ordner.

Stats

Link: <http://modolabs.com/kurogo/guide/modulestats.html?highlight=stats>

Das Stats-Modul dient eigentlich zur Erfassung interner Statistiken, die den Umfang von Google Analytics nicht annähernd erfüllen. Alle Daten werden in DATA_DIR/db.sqlite gespeichert. Da es aktuell Probleme mit dem Zugriff auf die Datenbank gibt, ist die Stats-Funktion aktuell ausgeschaltet (kann im Admin-Bereich angepasst werden).

News

Link: <http://modolabs.com/kurogo/guide/modulenews.html?highlight=news>

News ist ein vorgegebenes Modul von Kurogo, was direkt verwendet werden kann. Nur die Config-Files müssen angepasst werden, zum Beispiel über den Admin-Bereich.

Links

Link: <http://modolabs.com/kurogo/guide/modulenews.html?highlight=links>

Links ist ein vorgegebenes Modul von Kurogo, das direkt verwendet werden kann. Nur die Config-Files mit den Link-URLs müssen angepasst werden, z.B über den Admin-Bereich.

4.3 Beispielcode und -funktionen

getModuleArray(\$foo):

1. Einfache Erzeugung von Navlist- und Result-Array
2. Werte in module.ini des entsprechenden Modul unter dem Abschnitt \$foo speichern
3. Folgende Werte können gesetzt werden
 - a. title (Titel gross)
 - b. subtitle (Untertitel klein)
 - c. url (Ziel-URL, Telefonnummern mit tel:/E-Mail mit mailto:)
 - d. class ('email', 'phone', default Webseiten-URL, zeigt kleines Symbol bei Results an)

buildBreadcrumpUrl():

1. Einfache Erzeugung von generischen Navlist- und Result-Arrays sowie einfache Möglichkeit zur Paramtresierung
2.

```
foreach ($categories as $categorie) {  
    $entries[] = array(  
        'title' => $categorie['title'],  
        'url' => $this->buildBreadcrumbURL('results', array(  
            'letter' => $categorie['title'],  
            'type' => $type
```

```

    ))
  );
}

```

3. URL setzt sich zusammen aus Zielseite und Array mit beliebigen Variablen (Parametern)

getArg():

1. Auslesen der in der Breadcrumb-URL übergebenen Parameter
 - a. `$letter = $this->getArg('letter');`
`$type = $this->getArg('type');`

4.4 Google Analytics

Google Analytics ist eine bereits von Kurogo vorimplementierte Funktion, bei der ausschliesslich der Google-Analytics-ID im in den Config-Files angepasst werden muss, z.B. über das Admin-Module (siehe auch <http://modolabs.com/kurogo/guide/configuration.html#analytics>).

Zusätzlich und unabhängig von Kurogo wurde ein Hinweis-Disclaimer auf jeder Seite in den Footer eingefügt. Daher liegt eine footer.tpl im SITE_DIR/app/common/templates-Ordner, die im Block ... einen file-include der im selber befindlichen disclaimer.tpl enthält. Da dieses file-include innerhalb der Abfrage nach dem Vorhandensein einer Analytics-ID erscheint, verschwindet der Disclaimer, sobald die Analytics-ID aus den Config-Files entfernt wird.

4.5 Add2Homescreen

Add2Homescreen ist eine Funktion, die iOS-User ein kleines Pop-up zeigt, und darauf hinweist, dass die mobile Webseite mithilfe eines App-Like Icons auf den Homescreen gelegt werden kann (siehe <http://cubiq.org/add-to-home-screen>).

Es geht dabei ausschliesslich um den Startbildschirm, als dem Home-Module. Die Funktionalität liegt eigentlich in einer add2homescreen.js, das Aussehen in einer add2homescreen.css. Da eine Einführung in die Kurogo-Struktur umständlich gewesen wäre, liegt der Code nun in einer compliant.js unter /home/javascript bzw. in einer compliant.css unter /home/css. Im der unter /home/templates liegenden header.tpl liegt im Block "Javascript" zusätzlich noch der auszuführende add2homescreen-Aufruf.

Ab Kurogo V1.4 soll voraussichtlich die add2homescreen-Funktionalität ebenfalls vorhanden sein, in dem Fall können die Ordner /home/css und /home/javascript mit den darin liegenden Dateien gelöscht werden. Das gleiche gilt für die header.tpl unter /home/templates, da gleichzeitig der Metatag "charset = UTF-8" in der neuen Version aufgenommen werden soll (Achtung! Anpassung an der in index.tpl, an der Stelle, wo der Header eingebunden wird).

4.5 Zugriffsrechte auf Server

```

775   auf /site/hsg (stats: wichtig auch für das Schreiben in SITE_DIR/data/db.sqlite)
755   ;sonst

```

5. Unterstützte Geräte

OS	Hersteller	Gerätebez.	Status	Anmerkung
iOS 3, iOS 4	Apple	iPod Touch	100%	100% (Achtung, falls Javascript deaktiviert ist, funktionieren Module teilweise nicht)
Android 2.2.1	Samsung	GT-S5830, Ace	99%	Pinch-Zoom ist überall aktiv, sollte im besten Fall nur bei den Floorplans aktiv sein. Bei den Floorplans ist dazu die Auflösung sehr gering → die Raumnummern können nicht mehr abgelesen werden.
Windows Phone 7.10 Mango	Samsung	Omnia 7	99%	 für die Raumübersicht bei der Campusmap werden nicht dargestellt.
Android 2.2.2	HTC	Desire	99%	Pinch-Zoom ist überall aktiv

5.1 Bekannte Issues

Von wem gemeldet	Wo (Modul)	Was (Beschreibung)
TSA	News	RSS-Feed von Alexandria enthält Fehler, die Weiterleitung auf einzelne News-Einträge funktioniert nicht.
TSA	Suche	Suche Funktioniert nicht für Personen.

5.2 Neue Funktionalität

Von wem bearbeitet	Wo (Modul)	Was (Beschreibung)
Daniel Fiechter	Verzeichnis	Direkter Anruf und direkte E-Mail aus Verzeichnis-Modul

6. Systemvoraussetzungen

<http://modolabs.com/kurogo/guide/setup.html>