

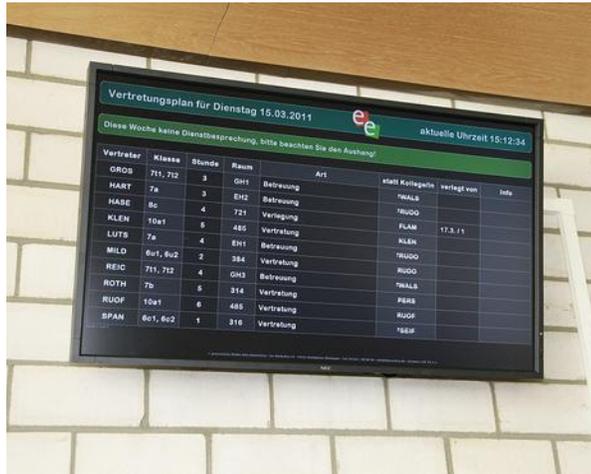
TU Dresden – Sachgebiet 7.5 Web und Video

Plone Digital Signage für die Info-Monitore an der TU Dresden [Update]

Sebastian Schietzold

11.03.2020

Digital Signage



Quellen (von links oben nach rechts unten):
 1 Media Arts (Wikipedia)
 2 Wesley Fryer (Flickr.com), CC BY 2.0
 3, 4 Till Carlos (Flickr.com), CC BY 2.0
 5 Antti Leppänen (Wikipedia), CC BY 4.0

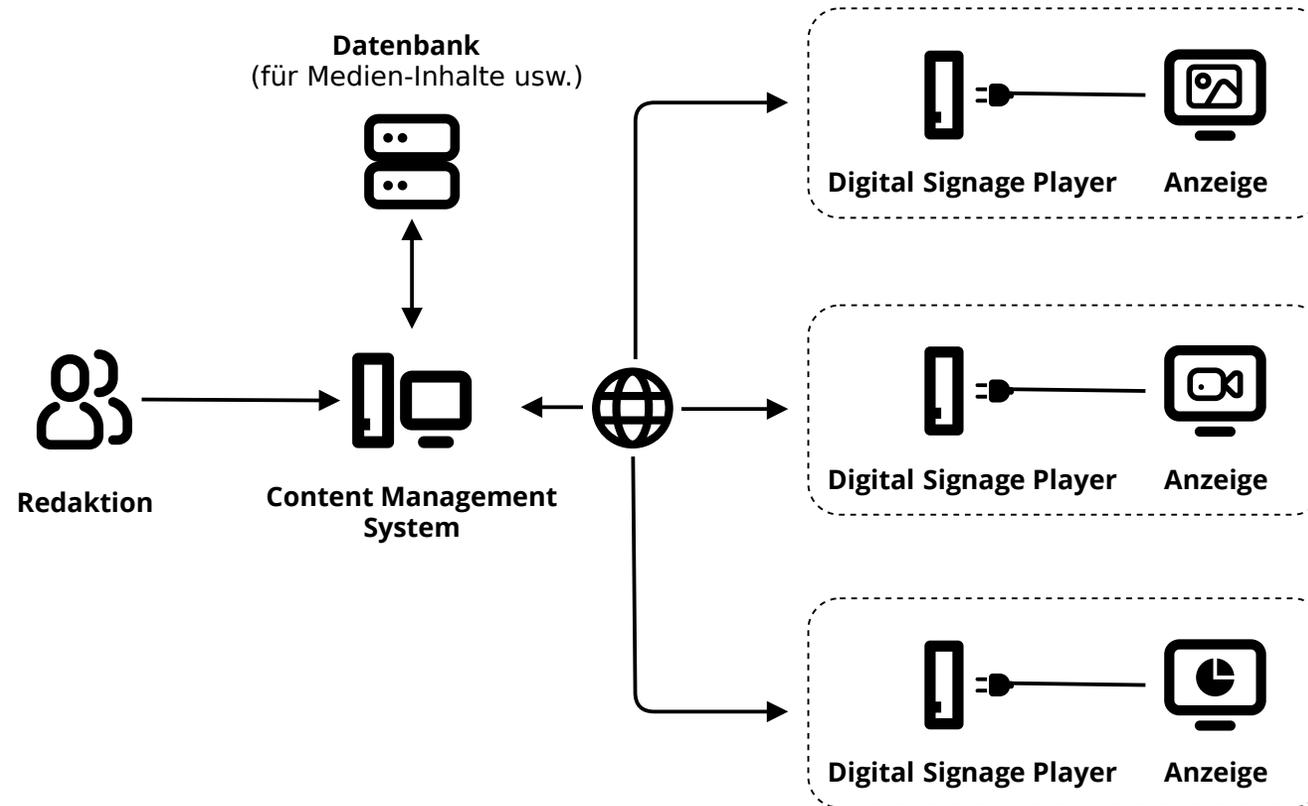
Digital Signage

Warum?

- Einstellen von ortsbezogenen und zeitlich gesteuerten Inhalten
→ Veranstaltungsankündigungen, Raumänderungen, Warnungen
- Fernsteuerbarkeit
→ zentrales Einstellen von (zentralen) Informationen
- Kostenvorteile gegenüber vergleichbaren Print-Kampagnen
- Werbewirksamkeit

Quelle: „Digital Signage. Die globale Studie. Chancen und Risiken.“, POPAI 2008

Digital Signage Komponenten



Digital Signage

Integration in die Plone-Website

- Bestehendes Rechtesystem
- Bestehender Content
- Gewohnte Oberfläche für Redakteure
- Leichte Zuordnung von Bildschirmen (jeder bekommt eine URL)

Digital Signage an der TU Dresden

Digital Signage an der TU Dresden

Problem & Zielstellung

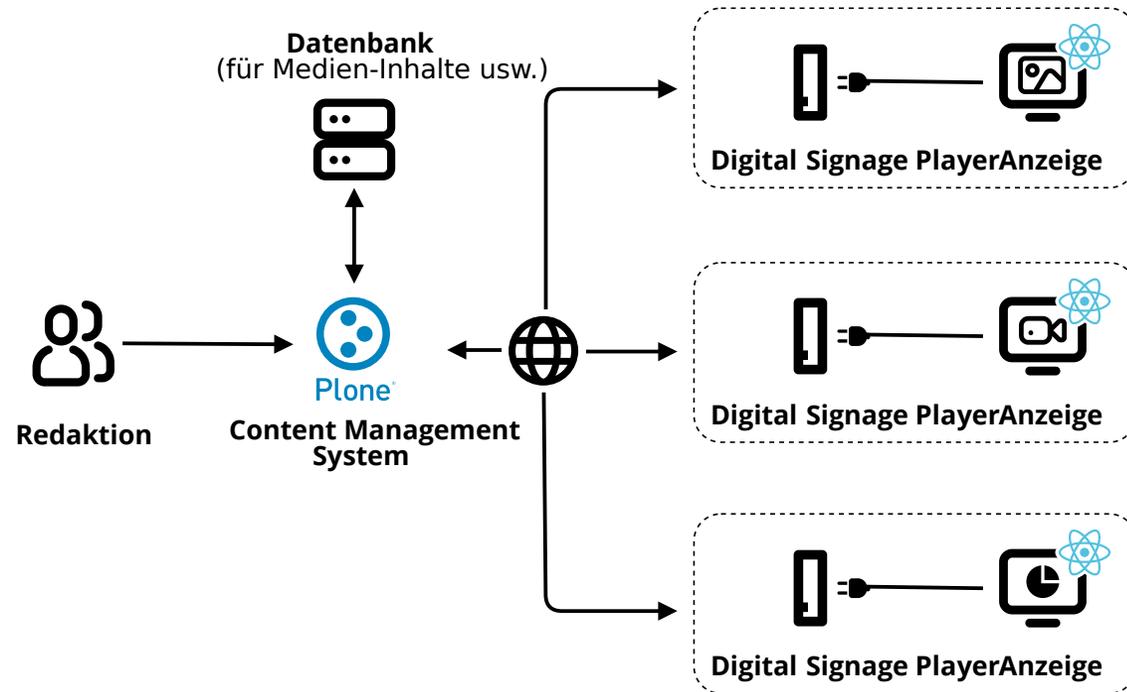
Ist:

Keine Monitore oder „individuelle“, „eigenverantwortliche“ Lösungen (Mensa, Uni-Bibliothek, Fakultäten ...)

Soll:

- Campus-weite, einheitliche Ausstattung von Gebäuden mit Monitoren
- Jeweils ein Monitor für zentrale Informationen + ein Monitor für gebäudespezifische Informationen
- Rechtemanagement mit Rollenkonzept
- Templates
- Ablaufsteuerung

Digital Signage und Plone



Strategie:

- Bereitstellung der Inhalte durch existierende Plone-Installation
- Monitor-Inhaltstyp physischer Monitor
- Endpoints via plone.rest
- Konfiguration im Abspielgerät:
 - Browser ruft „seine“ URL auf
 - React Web App (Player) startet
 - Player holt regelmäßig Folien ab und spielt diese gemäß Ablaufplan ab

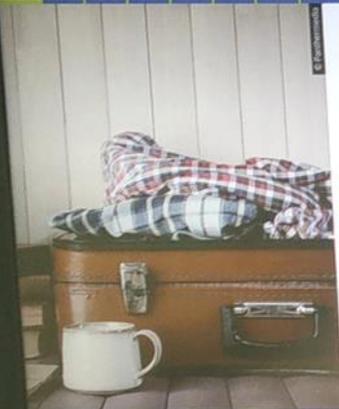
Entwicklung von tud.addons.monitor

tud.addons.monitor

Teaser

 TECHNISCHE UNIVERSITÄT DRESDEN

Woche / Week 4
Dienstag / Tuesday, 21.1.2020 12:14



Umzugsbeihilfe auf der Kippe

Studentenwerk nimmt keine Anträge mehr an

Die Stadt Dresden plant, die Umzugsbeihilfe für Studierende einzustellen. Aktuell werden deshalb vom Studentenwerk keine Anträge auf Umzugsbeihilfe mehr angenommen. Die endgültige Entscheidung trifft der Stadtrat am 30. Januar.

Bisher bekamen Studierende, die ihren Hauptwohnsitz erstmals nach Dresden verlegt hatten, finanzielle Unterstützung.

Hermann-Krone-Bau
Nöthnitzer Str. 61, 01187 Dresden

KRO

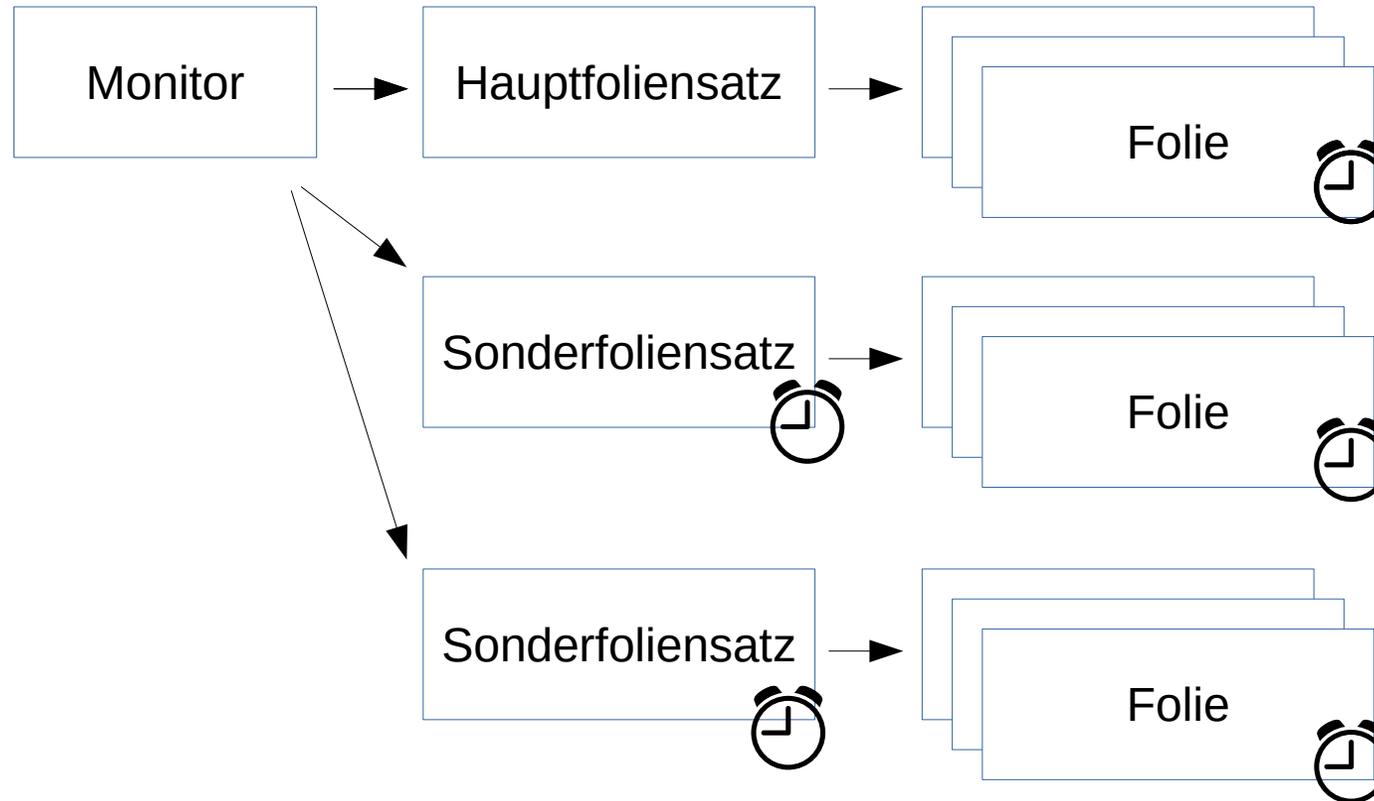


Einschreibung für das WS 2019/2020

Ab 01. Juni 2019 beginnt der Einschreibungszeitraum für das Wintersemester 2019/2020 an der TU Dresden. Die Bewerbung und Einschreibung an der TU Dresden erfolgt im selma-Portal über die Einstiegsseite tu-dresden.de/bewerbung. Bei der Online-Einschreibung werden Sie mit vielen Hilfen sicher durch das Bewerbungsportal geführt. Am Ende erhalten Sie eine Bewerbungsnummer und werden aufgefordert, sich ein Dokument auszudrucken. Darin sind Informationen enthalten, welche Unterlagen das Immatrikulationsamt für Ihre Immatrikulation benötigt.

tud.addons.monitor

Monitor, Foliensatz, Folie



tud.addons.monitor

Folieneditor

- WYSIWYG (gleiche Styles wie in der Enddarstellung)
- Feste Vorlagen
- Eingeschränkte Möglichkeiten

Haupttext *

Deutsch (Hauptsprache)

</> Quellcode Layout Format **B** *I* x₂ x²

Titel

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed ac lorem nisl. Etiam pretium purus eget lacus porta pharetra. Ut iaculis lacinia erat nec convallis. Praesent vitae ipsum pharetra purus tristique tempus. Vivamus quis tellus erat.

Verstecke Meldungen

tud.addons.monitor

Zeitsteuerung

- Optional für Folien
- Fakultativ für Sonderfoliensätze
- Angelehnt an Terminlogik
- Wiederholungen jeder Art möglich

Standard Zeitplanung

Zeitplanung aktivieren

Startzeitpunkt der Anzeige *
Die Folie wird erst ab dem eingestellten Zeitpunkt angezeigt (ggf. abhängig von Wiederholungsregeln). Die Folie muss veröffentlicht sein.

11.03.2020  10  00 

Endzeitpunkt der Anzeige *
Die Folie wird nach dem eingestellten Zeitpunkt nicht mehr angezeigt (ggf. abhängig von Wiederholungsregeln).

11.03.2020  11  00 

Wiederholung

[HINZUFÜGEN](#)

Keine Wiederholung festgelegt.

tud.addons.monitor

Schaltzentrale

- Timeline zeigt Foliensätze und Folien
- Durchwandern der Zeitachse oder Springen zu bestimmtem Zeitpunkt mit Vorschau
- Einfärbung der zu einem bestimmten Zeitpunkt gültigen Foliensätze
- Verwendung von JS-Lib „vis-timeline“

CAMPUS

11.03.2020  16  45 

JETZT **- 1 H** **- 15 MIN** **+ 15 MIN** **+ 1 H**



Vollbild-Vorschau



tud.addons.monitor

Folienarten

- Textfolie (kein Bild, ein Bild, zwei Bilder)
- Bildfolie
- Videofolie (Youtube, Vimeo, Magma)
- DVB-Folie (ÖPNV)

tud.addons.monitor

Rest-Schnittstelle

- Abstrahiert Zeitsteuerungen und Foliensätze weg
- Liefert zu einem Zeitpunkt X eine gültige Playlist
- Endpoints:
 1. Monitor-Informationen (Bezeichnung, Standort, ...)
 2. Playlist (Playlit-Hash, ggf. Folien-UIDs und Modified-Daten)
 3. Folie(n) (liefert Folie(n)-Inhalte und -Metadaten)

tud.addons.monitor

Player

- React Web App als Resource folder ausgeliefert
- Logik an sich relativ „dumm“: Playlist holen, Folien darstellen
- Anzeige von Kopf- und Fußzeile
- Ablaufsteuerung:
 - regelmäßiges Abfragen der Playlist
 - neue Inhalte werden geladen und gecached
 - Jede Folie wird eine bestimmte Zeit angezeigt
 - Playlist wird ständig wiederholt

Demo

tud.addons.monitor

Stolperfallen

- Darstellung auf verschiedenen Auflösungen:
 - Grundsätzlich: Festlegung auf eine Auflösung und Skalierung
 - Skalierung mit Hilfe der Basis-Fontsize (und relativen Angaben „rem“) tut nicht!
 - Lösung: CSS transform (scale) und Event-Listener of window resize
- Scaling von IFrames:
 - Die festgelegte Auflösung kann für IFrame-Content ungünstig sein
 - CSS transform tut seinen Job → Verkettung von scales
- Editor: Echtes WYSIWYG fast unmöglich
 - Rendering von Schrift unterscheidet sich zwischen Systemen → unterschiedliche Umbrüche
 - Vorgabe: Gleicher Browser zum editieren wie für die Darstellung
 - Annähernd gleiche Ausgabe durch CSS Scaling

tud.addons.monitor

Stolperfallen

- React Player:
 - IFrames sagen nicht, wann sie fertig geladen haben → müssen durchgängig im DOM behalten werden
- Berechnung aktueller Playlist
 - Berechnen von zeitlichen Überlagerungen mit Wiederholungen ist nicht easy
 - Leider keine einfach einzusetzende „fertige“ Lösung bekannt
- Rest-API
 - Abholen von mehreren (hunderten von) Folien via GET → im Zweifel zig UUIDs vs. Zeichenbegrenzung der URI (2048 Zeichen?)
 - Momentan: „GET“ Request mit POST-Payload
- Hoffentlich keine Stolperfalle: Pull statt push

tud.addons.monitor

Wo bekomme ich das?

- Keine Veröffentlichung in Aussicht
- Eng verzahnt mit anderen Paketen und Addons
 - Insbesondere auch mit CKEditor und eigenen Addons
- Sorry!

Fragen?

Digital Signage und Plone

Vielen Dank
für die Aufmerksamkeit!