

EAS - Elektronisches AufrufSystem

Handbuch

Julian Hartig

23. Juli 2018

Inhaltsverzeichnis

1. Einleitung	3
2. Installation	4
2.1. Grundsätzliche Systemvoraussetzungen	4
2.2. Server	4
2.2.1. Tonsignal bei Patientenaufruf	4
2.3. Client	4
2.3.1. Webansichten	4
2.3.2. PVS-Anbindung	5
3. Konfiguration	6
3.1. Neuen Raum anlegen	6
3.2. Raum bearbeiten	7
3.3. Nachricht hinterlegen	7
3.4. Raum leeren	7
4. Wartezimmeransicht	8
4.1. Konfiguration	9
4.1.1. Parameter <code>calls</code>	9
4.1.2. Parameter <code>callby</code>	9
5. Anpassung des Designs	10
A. Lizenzinformationen	11

1. Einleitung

Das Elektronische AufrufSystem ist ein für Arztpraxen und vergleichbare Einrichtungen gedachtes Patientenaufrufsystem. Es bietet eine Aufrufansicht für das Wartezimmer sowie ein Kontrollinterface und auf Wunsch auch Interfaces für einzelne Räume, um die Raumbelegung nach außen zu signalisieren.

Die Implementierung erfolgt in Python, somit ist das System auf allen von Python unterstützten Betriebssystemen lauffähig. Die jeweiligen Ansichten werden als Website geladen, sodass sie auf jedem Endgerät angezeigt werden könne, welches über einen Webbrowser und Netzwerkanbindung verfügt, neben den Praxiscomputern beispielsweise also auch auf einem an einen Raspberry Pi angeschlossenen Monitor im Wartezimmer oder einem Tablet neben der Sprechzimmertür.

Das System kann grundsätzlich an jedes Praxisverwaltungssystem angebunden werden, welches die Ansteuerung von Drittsoftware über die GDT-Schnittstelle unterstützt. Bei Praxisverwaltungssystem, die den Patientenaufruf über eine BDT-Aufrufchnittstelle unterstützen (wie z.B. Medical Office der Fa. Indamed) kann der aktuell in einem Raum befindliche Patient auch über das EAS-Kontrollinterface per Klick im Praxisverwaltungssystem aufgerufen werden. Das für die Kommunikation verwendete, offene WAMP-Protokoll erlaubt außerdem prinzipiell eine noch engere Integration des Praxisverwaltungssystem, wenn vom Hersteller gewünscht.

2. Installation

2.1. Grundsätzliche Systemvoraussetzungen

2.2. Server

Zum Betrieb des Servers ist die Installation des kostenlosen Open-Source Websocket-Routers `crossbar.io` erforderlich. Eine passende Konfiguration findet sich im Repository von EAS. Außerdem werden neben einem lauffähigen Python-Interpreter (v2.7.x) noch die Python-Packages `autobahn` und `peewee` mit ihren Abhängigkeiten benötigt. Diese lassen sich, ebenso wie `crossbar.io` komfortabel über das Python-Paketverwaltungssystem `pip` installieren.

Bezüglich einer Anleitung zum Starten von `crossbar.io` beim Systemstart sei an dieser Stelle auf die Dokumentation von `crossbar.io` verwiesen. Es muss außerdem das Python-Skript `server/easserver.py` aus dem Repository ausgeführt werden. Die EAS-Datenbank wird unter dem Dateinamen `eas.db` im Arbeitsverzeichnis erwartet. Existiert diese Datei dort nicht, so wird sie beim Ausführen von `easserver.py` automatisch erzeugt und die notwendigen Tabellen angelegt.

Der Server ist standardmäßig so konfiguriert, dass er eingehende Anfragen auf Port 8080 erwartet. Soll ein abweichender Port verwendet werden, so muss eine entsprechende Änderung in der Datei `.crossbar/config.json` sowie im Kopfbereich von `easserver.py` erfolgen.

2.2.1. Tonsignal bei Patientenaufruf

Die Wartezimmeransicht unterstützt die Ausgabe eines Tonsignals beim Aufruf eines Patienten. Hierfür muss eine Datei mit dem Namen `bell.mp3` im Verzeichnis `webroot` abgelegt werden, die den gewünschten Ton enthält. Das Internet hält hierzu zahlreiche kostenlos verwendbare Klänge bereit.

2.3. Client

2.3.1. Webansichten

Die Ausgabe der Wartezimmer-, Kontroll- und Raumbelungsansicht setzt nur einen Webbrowser voraus. In diesem muss dann eine der folgenden Adressen aufgerufen werden:

```
http://<server-ip>:<server-port>/wartezimmer.html
```

```
http://<server-ip>:<server-port>/control.html
```

2.3.2. PVS-Anbindung

Für die Anbindung des Praxisverwaltungssystems sind die Skripte im Verzeichnis `clients` vorgesehen.

eas-populate-room.py

Mit diesem Skript kann das Praxisverwaltungssystem einen Patienten aufrufen. Es benötigt das Python-Paket **autobahn** (s.o.).

Das Praxisverwaltungssystem muss hierbei so konfiguriert werden, dass das Skript folgendermaßen aufgerufen wird:

```
$ eas-populate-room.py <server-ip>:<server-port>  
C:\Pfad\zur\Datei.gdt "Name des Zimmers"
```

Als Testkennung ist **CALL** zu übergeben. Soll ein Sprechzimmer freigegeben werden, ohne dass ein neuer Patient aufgerufen werden soll, so ist **LEAVE** zu übergeben. Die in der GDT-Datei übermittelten Patientendaten werden dann ignoriert.

Für das Programm Medical Office bietet sich hier die Konfiguration einzelner Aufträge für jedes vorhandene Sprechzimmer im Datenpflegesystem an. Diese Aufträge können dann mit Schaltern in der Schalterleiste verknüpft werden, sodass der aktuell in Medical Office aufgerufene Patient über EAS in das entsprechende Zimmer bestellt wird.

eas-create-gdt.py

Dieses Skript dient als Hilfsskript, um einen automatischen Patientenaufruf im Praxisverwaltungssystem über eine BDT-Datei zu ermöglichen, sofern dies vom Praxisverwaltungssystem unterstützt wird. Hierzu muss der verwendete Webbrowser so konfiguriert werden, dass er beim Klicken eines **easpatient**-Links aus der Kontrollansicht das Skript wie folgt aufruft:

```
$ eas-create-gdt.py C:\Pfad\zur\Aufrufdatei.bdt  
Patientennummer
```

Die Patientennummer darf hierbei auch noch das vorangestellte **easpatient:** enthalten, dieses wird vom Skript dann automatisch vor der Übergabe der Patientennummer an das Praxisverwaltungssystem entfernt.

Konfigurationsanleitungen für benutzerspezifische Linkprotokolle finden sich auf den Webseiten der Browserhersteller. Für Microsoft Windows ist (auch) eine zentrale Konfiguration über die Registry möglich.

3. Konfiguration

Die Konfiguration von EAS erfolgt webbasiert über das Kontrollinterface:

```
http://<server-ip>:<server-port>/control.html
```

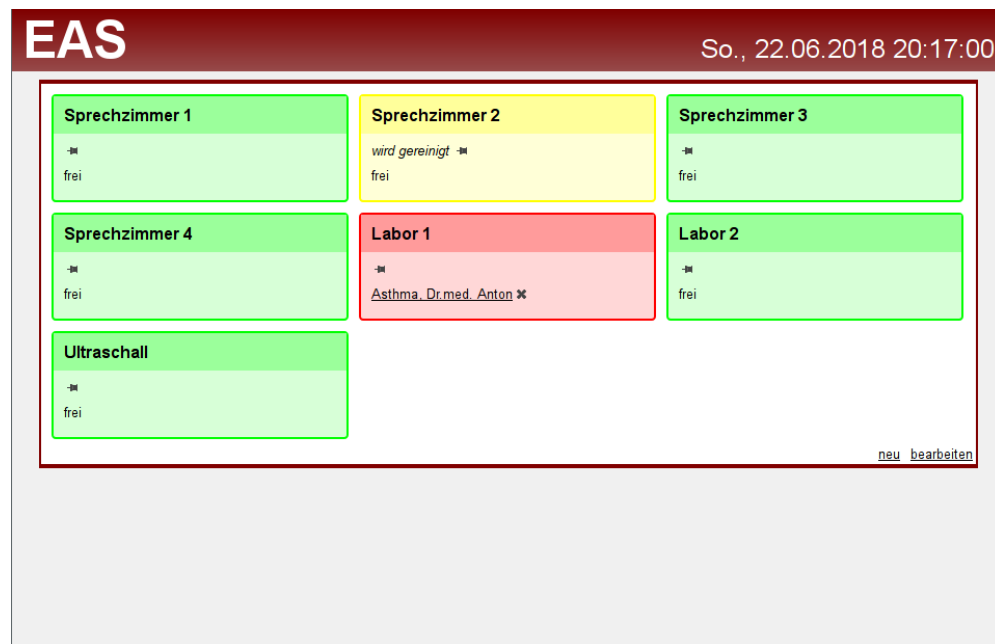


Abbildung 3.1.: Kontrollinterface

3.1. Neuen Raum anlegen

Mit Klick auf den Link „neu“ können neue Räume angelegt werden. Es öffnet sich ein Fenster zur Eingabe des Raumnamens und der Raumpriorität. Die Raumpriorität dient hierbei zur Bestimmung der Anordnung der Räume in der Kontrollansicht. Der Raum mit der niedrigsten Priorität erscheint oben links, der Raum mit der höchsten Priorität unten rechts.

3.2. Raum bearbeiten

Zum Bearbeiten von Räumen muss der Link „bearbeiten“ angeklickt werden. Anschließend werden über jedem Raum ein Eingabefeld für die Priorität sowie ein Button zum Löschen des Raumes eingeblendet. Die eingegebene Priorität wird bei Verlassen des Eingabefeldes oder Druck von <ENTER> automatisch gespeichert.

Der Bearbeitungsmodus kann mit einem erneuten Klick auf den „bearbeiten“-Link verlassen werden.

3.3. Nachricht hinterlegen

Über einen Klick auf die Pinnadel unterhalb des Raumnamens kann eine raumbezogene Nachricht hinterlegt werden. Diese wird dann neben der Pinnadel angezeigt, der Raum verfärbt sich gelb (siehe Raum „Sprechzimmer 2“ Screenshot S. 8).

3.4. Raum leeren

Ist ein Raum durch einen Patienten belegt, so kann der Raum durch Klick auf das neben dem Patientennamen dargestellte Kreuz freigegeben werden. Ein belegter Raum wird in der Kontrollansicht rot dargestellt (siehe Raum „Labor 1“ Screenshot S. 8).

Ein leerer Raum erscheint in der Kontrollansicht grün.

4. Wartezimmeransicht

Die Wartezimmeransicht wird über diese Adresse aufgerufen:

```
http://<server-ip>:<server-port>/wartezimmer.html
```

In der Wartezimmeransicht wird eine konfigurierbare Zahl von Patienten/Aufrufen untereinander angezeigt mit den jeweiligen Räumen, in die die Patienten sich begeben sollen. Der aktuellste Aufruf erscheint jeweils ganz unten in der Liste und blinkt für ca. 20 Sekunden auf. Auf Wunsch kann auch ein Signalton abgespielt werden, wenn ein neuer Aufruf erscheint (siehe Abschnitt 2.2.1).



Abbildung 4.1.: Wartezimmeransicht

Ist die Zahl der darzustellenden Einträge erreicht, so wird bei neuen Einträgen der jeweils älteste Eintrag von der Liste entfernt. Außerdem werden Patienten aus der Liste entfernt, die den entsprechenden Raum verlassen haben. Dies sorgt dafür, dass die Liste so übersichtlich wie möglich bleibt.

4.1. Konfiguration

Die Konfiguration erfolgt über den Query-String der verwendeten URL. Eine beispielhafte URL mit Konfigurationsparametern könnte so aussehen:

```
http://<server-ip>:<server-port>/wartezimmer.html?calls  
=10&callsby=id
```

4.1.1. Parameter calls

Der Parameter `calls` kann mit einer Ganzzahl als Wert verwendet werden, um die Zahl der gleichzeitig angezeigten Aufrufe zu steuern (Standardwert: 8).

4.1.2. Parameter callby

Wird der Parameter `callby` mit dem Wert `id` belegt, so werden Patienten nicht über ihren Namen sondern mit ihrer Patientennummer aufgerufen.

5. Anpassung des Designs

Durch entsprechende Bearbeitung der unter **webroot** befindlichen CSS-Dateien kann das Aussehen der EAS-Ansichten schnell und einfach an das Corporate Design der Praxis angepasst werden.

A. Lizenzinformationen

Die EAS-Software steht unter der GPLv3. Der vollständige Lizenztext kann unter <https://www.gnu.org/licenses/gpl-3.0.de.html> eingesehen werden.