

# KHANR-VZ Schnittstelle

**Beschreibung der Schnittstelle zu den Verwaltungssystemen der Krankenhäuser**

11.03.2019



# Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>4</b>
<b>2. Technische Spezifikation</b>	<b>5</b>
<b>3. Sicherheit</b>	<b>6</b>
3.1. Token-Gültigkeit	6
3.2. Ausblick	6
<b>4. Fachliche Schnittstelle</b>	<b>7</b>
4.1. Anlegen eines Arztes	7
4.2. Ändern eines Arztes	7
4.3. Clearing-Ärzte ermitteln	8
<b>5. Beispiele</b>	<b>9</b>
5.1. Anmeldung	9
5.2. Optional: checkToken	11
5.3. Anlegen eines Arztes	13
5.3.1. Standardfall erfolgreich	13
5.4. Anlegen eines Arztes mit Warnmeldungen	15
5.4.1. Korrektur und zweite Meldung	18
5.5. Anlegen eines Arztes – harter Fehler	20

Der Inhalt dieses Dokumentes wird ausschließlich zu Informationszwecken zur Verfügung gestellt. Weder das Ganze noch einzelne Bestandteile dieses Dokuments dürfen ohne schriftliche Genehmigung der innovas GmbH weitergegeben oder in irgendeiner Form reproduziert werden.

# 1. Einleitung

Zur Anbindungen von Systemen für die Personalverwaltung – z.B. von Krankenhausinformationssystemen oder von Personalverwaltungssystemen – wird eine leichtgewichtige, aber dennoch sichere REST-Schnittstelle für die folgenden Use Cases definiert:

- einen Arzteintrag anlegen
- eine zweite Meldung für einen Eintrag durchführen (Im Falle von Warnhinweisen beim Anlegen)
- einen Eintrag aktualisieren
- Clearing-Informationen abrufen

Die Registrierung von Nutzern und die Vergabe von Passwörtern müssen vor Nutzung der REST-Schnittstelle über die Web-Oberfläche erfolgen.

Dieses Dokument beschreibt diese Schnittstelle und erläutert anhand von Beispielen deren Verwendung.

## 2. Technische Spezifikation

Die REST-Schnittstelle verwendet JSON<sup>1</sup> als Notation für die übertragenen Daten.

Die Schnittstelle ist in zwei Teile aufgeteilt. Ein Teil bildet die Authentifizierung ab (benutzer-extern-login-sst.yaml) und der zweite Teil die eigentliche fachliche Schnittstelle zur Verwaltung von Ärzten (arztnummernverzeichnis-extern-sst.yaml). Beide Teile sind formal in der OpenAPI Specification V2.0<sup>2</sup> im yaml-Format spezifiziert. Die Spezifikationsdateien werden zusammen mit diesem Dokument ausgeliefert.

Aus den Spezifikationsdateien lässt sich der REST-Client z.B. mit dem Swagger Codegen<sup>3</sup> für eine Vielzahl von Sprachen generieren. Dieser Weg empfiehlt sich, da dadurch zumindest syntaktische Fehler weitgehend ausgeschlossen werden können.

Aus den Spezifikationsdateien lässt sich mit dem Swagger Codegen zur besseren Lesbarkeit auch eine HTML-Dokumentation generieren.

### 2.1. Anbindung

Das KHANR-VZ wird ab Anfang Juni 2019 unter der Adresse [www.arztnummernverzeichnis.de](http://www.arztnummernverzeichnis.de) betrieben. Ein Testsystem wird unter [test.arztnummernverzeichnis.de](http://test.arztnummernverzeichnis.de) zur Verfügung stehen.

---

<sup>1</sup> <https://www.json.org/json-de.html>

<sup>2</sup> <https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md>

<sup>3</sup> <https://swagger.io/tools/swagger-codegen/>

## 3. Sicherheit

Die Nutzung von https ist verpflichtend. Dies bedeutet, dass der Transportweg SSL-verschlüsselt ist. Der Client muss überprüfen, ob das Zertifikat des Servers gültig ist. Eine Übertragung von KHANR-VZ-bezogenen Daten darf nur in diesem Fall erfolgen.

Die Einrichtung von Nutzern und die Vergabe von Passwörtern erfolgen über die in der Ausschreibung definierte Weboberfläche. Die so erhaltenen Zugangsdaten müssen bei der Anmeldung an der Schnittstelle übergeben werden.

Die Schnittstelle verwendet zur Authentifizierung JSON Web Token<sup>4</sup>.

1. Zunächst muss das Token angefordert werden. Dazu wird der REST-Endpoint `authenticate` (siehe `benutzer-extern-login-sst.yaml`) aufgerufen:
  - Method POST
  - URL: `https://<hostname>/benutzer/api/v0.0.1/authenticate`
  - Body (Content-Type „application/json“ `{"username": "*****", "password": "*****"}`)
2. Als Ergebnis setzt der Aufruf 2 Cookies:
  - XSRF-TOKEN
  - JWT\_TOKEN
3. Um einen weiteren Endpunkt aufzurufen, muss nun das Cookie `JWT_TOKEN` bei jedem folgenden Aufruf übertragen werden.

Die Verwendung einer Authentifizierung mit einer TAN als zweitem Faktor ist in diesem Verwendungskontext nicht vorgesehen. Der Endpunkt `/authenticate/tan` wird daher nicht verwendet.

### 3.1. Token-Gültigkeit

Das auf dem oben genannten Wege ermittelte Token ist 24h lang gültig. Danach muss der Aufrufer ein neues Token ermitteln. Mit dem Endpunkt `/checktoken` besteht die Möglichkeit jederzeit zu prüfen, ob das aktuelle Token noch gültig ist. Nach der Verwendung der Schnittstelle sollte der Benutzer aus Sicherheitsgründen ausgeloggt werden. Dies geschieht unter Verwendung des Endpunktes `/logout`

### 3.2. Ausblick

In einer folgenden Ausbaustufe wird eine tokenbasierte Authentifizierung auf Basis der Telematik-Infrastruktur umgesetzt werden, bei der die Berechtigungsverwaltung über die Primärsysteme des jeweiligen Krankenhauses erfolgt.

---

<sup>4</sup> [https://de.wikipedia.org/wiki/JSON\\_Web\\_Token](https://de.wikipedia.org/wiki/JSON_Web_Token)

## 4. Fachliche Schnittstelle

### 4.1. Anlegen eines Arztes

Zum Anlegen eines Arztes wird der Endpunkt /arzteintrag und die Methode POST verwendet. Ein Arzt wird dabei mit seinen Daten (soweit bekannt auch mit seiner ANR) übermittelt. Dabei wird zunächst iteration=ERSTE\_ITERATION übertragen. Im KHANR-VZ werden die Arztdaten geprüft und je nach Ergebnis der Prüfung wird im Body folgender Status zurückgegeben:

- **ERFOLGREICH:** Der Arzt konnte erfolgreich angelegt werden. Der Arzteintrag ist gefüllt und enthält insbesondere die ANR des Arztes.
- **WARNUNGEN:** Die Attribute müssen noch einmal von einem Sachbearbeiter kontrolliert werden, bevor sie gespeichert werden. Die Warnungen sind gefüllt und optional ist ein Korrekturvorschlag gefüllt. Die Warnungen und der optionale Korrekturvorschlag sollten dem Anwender des Verwaltungssystems angezeigt werden, damit dieser die Daten noch einmal prüfen kann und anschließend die Daten erneut übertragen kann (siehe ZWEITE\_ITERATION unten).
- **CLEARING** Der Arzt wurde im Clearing gespeichert und muss noch von der Geschäftsstelle geprüft werden. Die Clearinghinweise sind gefüllt und enthalten den Grund für die Aussteuerung ins Clearing.

Im Falle des Status WARNUNGEN ist der Korrekturvorschlag nur gefüllt, wenn (1) es nur eine eindeutige Abweichung je Attribut im KHANR-VZ gibt und (2) nur leicht abweichende Daten eingegeben wurden, um zu vermeiden, dass Daten über dieses Verfahren ausgelesen werden können.

Nachdem der Anwender im Falle WARNUNGEN die Daten geprüft, ggf. korrigiert und erneut freigegeben hat, übermittelt das Verwaltungssystem diese erneut über den Endpunkt /arzteintrag mit der iteration=ZWEITE\_ITERATION. Dabei werden die Abweichungsprüfungen deaktiviert und die Daten im System gespeichert.

### 4.2. Ändern eines Arztes

Zum Ändern eines Arztes wird der Endpunkt /arzteintrag/{anr} und die Methode PUT verwendet. Beim Ändern erfolgt keine Prüfung auf Abweichungen im KHANR-VZ. Stattdessen werden die aktualisierten Daten direkt aktualisiert.

Die ANR kann nur geändert werden, wenn die Geschäftsstelle einen Clearinghinweis mit einem ANR-Änderungsvorschlag unterbreitet hat. Die Änderung erfolgt, indem im Parameter anr die alte Nummer übertragen wird und im body die neue Nummer. Wird eine vom Vorschlag der Geschäftsstelle abweichende Nummer übertragen, resultiert dies in einen Fehler.

Beim Ändern eines Arztes werden alle übermittelten Attribute geprüft. Nachdem die Standorte für ein Krankenhaus im Standortverzeichnis veröffentlicht wurden, müssen beim Ändern eines Arztes auch die korrekten Standortnummern übergeben werden.

### **4.3. Clearing-Ärzte ermitteln**

Werden für einen Arzt unterschiedliche Daten zwischen mehreren Krankenhäusern oder zwischen Krankenhaus und Arztnummernvergabe (ANRV) gespeichert oder kann eine ANR beim Speichern eines Arzteintrags nicht direkt vergeben werden, so wird der Arzteintrag im Clearing gespeichert. Je nach Clearingstatus wird das Krankenhaus damit aufgefordert den Arzteintrag zu prüfen und ggf. zu korrigieren. Die Ärzte im Clearing werden über den Endpunkt /clearing ermittelt. Zurückgegeben werden alle Ärzte, die sich derzeit im Clearing befinden. Die Ärzte bei denen der Clearing-Status CLEARING\_KH ist, sollten kurzfristig vom Krankenhaus geklärt und ggf. korrigiert werden. Ärzte in den anderen Clearing-Status werden von der Geschäftsstelle bearbeitet. Hier ist zum aktuellen Zeitpunkt keine Interaktion seitens des Krankenhauses notwendig.

Die Webanwendung ermöglicht zu einem Clearingfall eine Rückfrage an das KH zu stellen. Diese Möglichkeit ist derzeit über die Schnittstelle nicht gegeben.

## 5. Beispiele

### 5.1. Anmeldung

POST `https://<hostname>/benutzer/api-extern/v0.0.1/authenticate`

Der Body wird als `application/json`; `charset=UTF-8` übertragen

```
{
  "username": "admin@stjoseph.de",
  "password": "admin@stjoseph.de"
}
```

Der Server liefert bei erfolgreicher Anmeldung den Statuscode 200 zurück. Zusätzlich wird der folgende Header übermittelt. Das Cookie `JWT_TOKEN` wird für die Authentifizierung bei weiteren Aufrufen der Schnittstelle benötigt und muss immer übertragen werden. Das Token `XSRF-TOKEN` wird für die Authentifizierung nicht benötigt.

Set-Cookie:

```
JWT_TOKEN=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzUxMiJ9.eyJzdWIiOiJhZG1pbkZkdGpvc2VwaC5kZSIsImF1dG8iOiJmF1dGhvcml0eSI6WyJHUK9VUC1OU1FXZi2MTEzMTEwOSIsIkdST1VQLVRFQ0gtSURfMDE1MjZmMzctNWYzNC00YTAwLWFjZDQtNzE2MDEjNWEzNTU0IiwiaWF0Ij0iOTk5YXRlbnBmbGVnZXIiLCJST0xFOX2JlbnV0emVyd2FsdHVuZyJdLCJleHAiOiJlNTIwNDQ2MjN9.dcqUt91XA9qFGQM09MGESibYdvovu2FJjAZmaH3jD47ruujcbR2g7wckdkh5tpIEy2SqrSYxMrvgVL3WW6A6GtiXer8oH0-e4JlozWX-_whtSh-xL_AieW-kn4GREmdQjyeZVKUakJ-iAcwO8ssVxXV2iLiEE9_Gx4ROE0Uc2AF5V7AzjNaZM_zgdyBmyCbNnTPc-ploi0hgWEq0o9tc8RLXw9ep00E084CGpPflQqDjDwJbFnDUi0y8JwtKEYDykTsEs6z6Fm_9ocR-yoh3iWtTRZYqH4G31QY6__T015ImRN7RbSwzmc19Vvv506oBXAOKVvkmKaWvTde1TtYUqPgnZutKWbfg6aaqoX5ZbjgfdqvWRCnC-f17vjhxH1H9fhoAh4s5q8yrFKamNb8B-peqN0Z4rg2UvJuPpOv0e5zmbGvVxyZNTM_VM5pv8IJ_ujshvoXGcUdLgyGQkjWfxNxSvbfpFnkYapMUqZbaahLXTuy06X6zAuQ2f3zJjEDgZYaFP1mgwJewDbxKVpe8h0RqZHO7g1NU6o1_Myykt0a2nqvBXn6PEUR7Hn3ykiSjYkJRy8EfJWQcRaXWePGpVajJFLktZ8uvqFcNbn355PW812bSNZ7USy3tAlSIGv_9HGgfb0EHADZiZjkxIp7Jsu9b4n_rT9WDT48zaNtG4; Max-Age=86399; Expires=Fri, 08-Mar-2019 11:30:22 GMT; Path=/; Secure; HttpOnly
Set-Cookie: XSRF-TOKEN=9eeb3337-e97a-497f-b7b2-e6fad7ecba43; Path=/; Secure
```

Also Body werden Informationen zum Benutzer zurückgemeldet:

```
{
  "benutzerAnmeldung": {
    "benutzer": {
      "username": "admin@stjoseph.de",
      "vorname": "Fritz",

```

## 5. Beispiele

```
"nachname": "Admin",
"email": "admin@stjoseph.de",
"telefonNr": "0123456789",
"telefonnummerZweiFaktorAuth": null,
"fax": null,
"aktivFlag": true,
"gruppe": {
  "fachId": "261111119",
  "name": "St. Joseph Krankenhaus",
  "erlaubteRollen": [
    "anv_kh_admin",
    "anv_kh_datenspflerger",
    "benutzerverwaltung"
  ],
  "untergruppen": [

  ]
},
"rollen": [
  {
    "fachId": "benutzerverwaltung",
    "name": "Benutzerverwaltung",
    "erlaeuterung": "Berechtigt zum Anlegen und Ändern von
Benutzern"
  },
  {
    "fachId": "anv_kh_datenspflerger",
    "name": "KH-Datenpflerger",
    "erlaeuterung": "Mitarbeiter, die berechtigt sind,
Daten im KHANR-VZ zu pflegen"
  },
  {
    "fachId": "anv_kh_admin",
    "name": "KH-Administrator",
    "erlaeuterung": "Ermöglicht die Administrierung eines
Krankenhauses. Das enthält z.B. das Ändern der Geschäftsführung"
  }
],
"gesperrt": false
},
"ablaufAnmeldung": "2019-03-08T12:30:23.000+01:00"
},
"forcedPwdChange": null,
"telefonNr": null,
"tanToken": null
}
```



## 5. Beispiele

```
{
  "benutzer":{
    "username":"admin@stjoseph.de",
    "vorname":"Fritz",
    "nachname":"Admin",
    "email":"admin@stjoseph.de",
    "telefonNr":"0123456789",
    "telefonnummerZweiFaktorAuth":null,
    "fax":null,
    "aktivFlag":true,
    "gruppe":{
      "fachId":"261111119",
      "name":"St. Joseph Krankenhaus",
      "erlaubteRollen":[
        "anv_kh_admin",
        "anv_kh_datenspflerger",
        "benutzerverwaltung"
      ],
      "untergruppen":[
      ]
    ],
    "rollen":[
      {
        "fachId":"benutzerverwaltung",
        "name":"Benutzerverwaltung",
        "erlaeuterung":"Berechtigt zum Anlegen und Ändern von B
enutzern"
      },
      {
        "fachId":"anv_kh_datenspflerger",
        "name":"KH-Datenpflerger",
        "erlaeuterung":"Mitarbeiter, die berechtigt sind, Daten
im KHANR-VZ zu pflegen"
      },
      {
        "fachId":"anv_kh_admin",
        "name":"KH-Administrator",
        "erlaeuterung":"Ermöglicht die Administrierung eines Kr
ankenhauses. Das enthält z.B. das Ändern der Geschäftsführung"
      }
    ],
    "gesperrt":false
  },
  "ablaufAnmeldung":"2019-03-08T12:30:23.000+01:00"
}
```



## 5. Beispiele

```
}  
],  
"standortzugehoerigkeiten": [  
  {  
    "standortnummer": "771234000",  
    "beginn": "2018-05-13"  
  }  
],  
"iteration": "ERSTE_ITERATION"  
}
```

Im Ergebnis wird der vollständige Arzteintrag zurückgeliefert, der insbesondere eine gefüllte anr beinhaltet.

## 5. Beispiele

```
{
  "status": "ERFOLGREICH",
  "arzteintrag": {
    "anr": "1483194",
    "vorname": "Sascha",
    "nachname": "Meyer",
    "geburtsdatum": "1994-02-05",
    "geschlecht": "MAENNLICH",
    "datumExamen": "2018-04-13",
    "datumApprobation": "2018-05-13",
    "drTitel": false,
    "datumErstePromotion": null,
    "qualifikationen": [
      {
        "code": "999",
        "abschlussdatum": "2018-05-13"
      }
    ],
    "standortzugehoerigkeiten": [
      {
        "standortnummer": "771234000",
        "beginn": "2018-05-13",
        "ende": null
      }
    ]
  },
  "korrekturvorschlag": null,
  "warnungen": [
  ],
  "clearinghinweise": [
  ]
}
```

#### 5.4. Anlegen eines Arztes mit Warnmeldungen

```
POST https://<hostname>/arztnummernverzeichnis/api-extern/v0.0.1/arzteintrag
```

Header:



## 5. Beispiele

```
{
  "status": "WARNUNGEN",
  "arzteintrag": null,
  "korrekturvorschlag": {
    "vorname": "Erika",
    "nachname": "Müller",
    "geburtsdatum": "1971-09-18",
    "geschlecht": "WEIBLICH",
    "datumExamen": "1996-04-06",
    "datumApprobation": "1996-06-17",
    "drTitel": true,
    "datumErstePromotion": "1996-09-11"
  },
  "warnungen": [
    {
      "level": "Hinweis",
      "code": "",
      "text": "In der ANRV ist für Nachname folgender Wert
hinterlegt: \"Müller\".",
      "feldliste": [
        "nachname"
      ]
    },
    {
      "level": "Hinweis",
      "code": "",
      "text": "In der ANRV ist für Vorname folgender Wert
hinterlegt: \"Erika\".",
      "feldliste": [
        "vorname"
      ]
    },
    {
      "level": "Hinweis",
      "code": "",
      "text": "Im KHANR-VZ ist für Vorname folgender Wert
enthalten: \"Erika\".",
      "feldliste": [
        "vorname"
      ]
    },
    {
      "level": "Hinweis",
      "code": "",
      "text": "Im KHANR-VZ ist für Nachname folgender Wert
enthalten: \"Müller\".",
      "feldliste": [
```

## 5. Beispiele

```
        "nachname"  
      ]  
    }  
  ],  
  "clearinghinweise": [  
  
  ]  
}
```

Da nur kleinere Abweichungen (vergessener Zweitname, unterschiedliche Schreibweise von „Müller“) zwischen dem gemeldeten und den in der Datenbank enthaltenen Werten vorliegen und diese insbesondere eindeutig sind, wird ein Korrekturvorschlag übermittelt. Bei größeren Abweichungen oder, wenn verschiedene Werte in der Datenbank vorliegen, wird kein Korrekturvorschlag zurückgeliefert. Die Warnungen werden jedoch immer übermittelt.

**5.4.1. Korrektur und zweite Meldung**

Da beim ersten Aufruf Warnmeldungen erzeugt wurden, wurde der Arzteintrag noch nicht gespeichert. Die Warnmeldungen und ggf. der Korrekturvorschlag sollten dem Benutzer angezeigt werden. Die Feldliste an den Warnungen enthält die Namen der Felder (in der Regel nur ein Feld), die für die Warnungen verantwortlich waren. Mit dieser Information können die Warnungen direkt an den Eingabefeldern angezeigt werden. Der Benutzer nimmt die Korrekturen vor und bestätigt diese. Dann ruft das Verwaltungssystem die Schnittstelle erneut als `ZWEITE_ITERATION` auf. Das folgende Beispiel geht davon aus, dass das Krankenhaus den Nachnamen „Mueller“ falsch eingegeben hat und dass „Müller“ korrekt ist, dass jedoch der zweite Vorname korrekt ist.

```
POST https://<hostname>/arztnummernverzeichnis/api-  
extern/v0.0.1/arzteintrag
```

Header:



## 5. Beispiele

Beim Aufruf mit `iteration=ZWEITE_ITERATION` wird der Arzteintrag gespeichert. Als Ergebnis wird der gespeicherte Eintrag zurückgegeben. Dieser enthält jetzt auch die ANR.

```
{
  "status":"ERFOLGREICH",
  "arzteintrag":{
    "anr":"1234567",
    "vorname":"Erika Hanna",
    "nachname":"Müller",
    "geburtsdatum":"1971-09-18",
    "geschlecht":"WEIBLICH",
    "datumExamen":"1996-04-06",
    "datumApprobation":"1996-06-17",
    "drTitel":true,
    "datumErstePromotion":"1996-09-11",
    "qualifikationen":[
      {
        "code":"999",
        "abschlussdatum":"2018-05-13"
      }
    ],
    "standortzugehoerigkeiten":[
      {
        "standortnummer":"771234000",
        "beginn":"2018-05-13",
        "ende":null
      }
    ]
  },
  "korrekturvorschlag":null,
  "warnungen":[
  ],
  "clearinghinweise":[
  ]
}
```

### 5.5. Anlegen eines Arztes – harter Fehler

Beim Anlegen eines Arztes können harte Fehler auftreten, die das Speichern verhindern. Dies können z.B. zu lange Zeichenketten sein oder auch fachliche Fehler, wie dass der Arzt schon angelegt ist. Das folgende Beispiel demonstriert das Verhalten, indem die Schnittstelle zum Anlegen des Arztes mit den gleichen Attributen noch einmal aufgerufen wird.

```
POST https://<hostname>/arztnummernverzeichnis/api-extern/v0.0.1/arzteintrag
```



5. Beispiele

```
}
```

Ergebnis:

```
[  
  {  
    "level": "Fehler",  
    "code": "",  
    "text": "Der Arzt ist bereits vorhanden.",  
    "feldliste": [  
      "anr"  
    ]  
  }  
]
```