

Restful webservice MAGW

Beschreibung:

Schnittstellen Definition des RESTful Webservice für die Kommunikation zwischen Web-Server und Android.

Einsatzzweck:

Verwaltung von Patientendaten.

Kodierung:

Alle Rückgabewerte sind UTF-8 kodiert.

Notation der Rückgabewerte:

Alle Rückgabewerte werden als JSON zurückgegeben.

Verfasser:

Benjamin Oertel

Zu Testzwecken gibt jede Methode zusätzlich noch den Parameter "response" mit zurück, dieser wird nach Abschluss der Entwicklung entfernt, um den Overhead zu minimieren.

1. Login eines Benutzers

Beschreibung:

Mit dieser Methode können Benutzer am Server identifiziert und angemeldet werden.

Eigenschaften:

Methode	POST
URL	http://titania.f4.htw-berlin.de/login/

Request:

Parameter	Typ	Beschreibung
username	String, a-z0-9	Der Benutzername des Benutzers, der am System angemeldet werden soll.
hash	a-z0-9	Das mit Sha1 gehashte Passwort des Benutzers, der am System angemeldet werden soll

Response:

Parameter	Typ	Beschreibung
statuscode	Integer	Der Code, welcher den Erfolg oder den aufgetretenen Fehler eindeutig identifiziert.
statusmessage	String, a-z0-9	Eine Beschreibung des zurückgegebenen Statuscodes.
response	Array	Enthält die userId des Benutzers

Beispiel:

Request		
Body	username	christian
	hash	Sha1(qwertz)
Response		
Body	statuscode	200
	statusmessage	Login successful.
	response:	{ "userId": "42" }

Error codes:

404	User not found.
403	User or password incorrect.

2. Patientendaten abfragen

Mit dieser Methode können die persönlichen Daten eines bestimmten Patienten ausgelesen werden. Der Patient muss zuvor über ein RFID Tag identifiziert werden.

Eigenschaften:

Methode	GET
URL	http://titania.f4.htw-berlin.de/patients/

Request:

Parameter	Typ	Beschreibung
rfid	String, a-z0-9	Die RFID eines bestimmten Patienten.

Response:

Parameter	Typ	Beschreibung
statuscode	Integer	Der Code, welcher den Erfolg oder den aufgetretenen Fehler eindeutig identifiziert.
statusmessage	String, a-z0-9	Eine Beschreibung des zurückgegebenen Statuscodes.
response	Array	Enthält die userId des Benutzers
userId	Integer	Die userId des Benutzers im System
firstname	String, a-z0-9	Der Vorname des Patienten.
lastname	String, a-z0-9	Der Nachname des Patienten.

Beispiel:

Request		
Body	rfid	rfid@1234
Response		
Body	statuscode	200
	statusmessage	Benutzer mit angegebener RFID wurde gefunden.
	response:	{ "userId": "42", "firstname": "Christian", "lastname": "Bunk" }

Error codes:

404	No patient with this RFID found.
-----	----------------------------------

3. Messungen zu Patient auslesen

Mit dieser Methode können alle Messungen einer bestimmten Messungsart über einen definierten Zeitraum zu einem Patienten ausgelesen werden.

Eigenschaften:

Methode	GET
URL	http://titania.f4.htw-berlin.de/measurements

Request:

Parameter	Typ	Beschreibung
limit	(optional) Integer	Die Anzahl der letzten Messungen die zurückgegeben werden soll.
type	String, a-z0-9	Der eindeutige Identifier des Messungstyps. Valide Typen: - temperature
patientId	Integer	Die Id des Patienten in der Datenbank.

Response:

Parameter	Typ	Beschreibung
statusCode	Integer	Der Code, welcher den Erfolg oder den aufgetretenen Fehler eindeutig identifiziert.
statusmessage	String, a-z0-9	Eine Beschreibung des zurückgegebenen Statuscodes.
response	Array	Enthält die userId des Benutzers
userId	Integer	Die userId des Benutzers im System
firstname	String, a-z0-9	Der Vorname des Patienten.
lastname	String, a-z0-9	Der Nachname des Patienten.

Beispiel:

Request		
Body	limit	10
	type	temperature
	patientId	42
Response		
Body	statusCode	200
	statusmessage	Benutzer mit angegebener RFID wurde gefunden.
	response:	<pre>[{ "date" : ":2011-11-28", "doctor" : "Martin Mustermann", "type" : "temperature", "value" : "12", "unit" : "°C" }]</pre>

Error codes:

404	Das Limit muss größer als 0 sein.
-----	-----------------------------------

4. Neue Messung anlegen

Mit dieser Methode kann eine neue Messung zu einem Patienten abgespeichert werden.

Eigenschaften:

Methode	POST
URL	http://titania.f4.htw-berlin.de/measurements

Request:

Parameter	Typ	Beschreibung
type	String, a-z0-9	Der eindeutige Identifier des Messungstyps. Valide Typen: - temperature
value	Double	Der Wert der Messung.
note	String, a-z0-9	Eine Notiz zur Messung
patientId	Integer	Die Id des Patienten in der Datenbank.
userId	Integer	Die Id des aktuellen Benutzers des mobilen Endgerätes.

Response:

Parameter	Typ	Beschreibung
statusCode	Integer	Der Code, welcher den Erfolg oder den aufgetretenen Fehler eindeutig identifiziert.
statusmessage	String, a-z0-9	Eine Beschreibung des zurückgegebenen Statuscodes.
response	Array	Enthält die userId des Benutzers
measurementId	Integer	Die Id der neu angelegten Messung in der Datenbank.

Beispiel:

Request		
Body	type	temperature
	value	5
	note	Normale Körpertemperatur
	patientId	42
	userId	5
Response		
Body	statusCode	200
	statusmessage	Measurement was created successfully.
	response:	{ "measurementId": "8", }

Error codes:

427	Invalid measurement type.
404	Patient not found.
428	Value is required.