**DocGL API description**

**General notes:**

* Each request (except login) has to have the following header: Authorization: Bearer <token>
* If header is not presented or not valid, endpoints return 401
* If input JSON is not valid, endpoints return 400

**POST /api/auth/registration**

Registration endpoint

Status code:

* 200 – OK

Sample request:

{

"firstName": "Rasticek",

"lastName": "Ruscak",

"email": "rastik@ruscak.sk",

"userType": "DOCTOR",

"userName": "docrasto",

"password": "docrasto",

"specialization": "DENTIST",

"phone": "09454546"

}

Sample response:

{

"doctor": {

"id": 5,

"firstName": "Rasticek",

"lastName": "Ruscak",

"email": "rastik@ruscak.sk",

"userName": "docrasto",

"registrationDate": 1493988888893,

"likes": 0,

"specialization": "DENTIST",

"phone": "09454546",

"blocked": false,

"approved": false

},

"token": ""

}

Notes:

* If doctor is not approved he doesn’t receive token
* If username or email is already taken by other user than endpoint response with code 400

**POST /api/auth/login**

Login endpoint

Status code:

* 401 – unauthorized
* 200 – OK

Sample request:

{“userName” : “rastobutton”, “password” : “rastobutton123”, “userType” : “ADMIN”}

Sample response:

{

"admin": {

"id": 1,

"firstName": "rasto",

"lastName": "button",

"email": "rasto@button.sk",

"userName": "rastobutton",

"passwordChanged": false

},

"token": "eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJhZG1pbiIsImlkIjoxLCJqdGkiOiJhYkJGaTQxdjBnNWY1OEFENjFpSWVBIiwiZXhwIjoxNDkzOTg3NTAwfQ.rFAbAaapbh6yfWQ3ctmHUjeARxCIjuL7rQ51k5YQ0RA"

}

**POST /api/auth/logout**

Logout endpoint

Status code:

* 204 – OK

**GET /api/registrations?timePeriod=WEEK/TODAY**

Gets all registrations

Status code:

* 401 – unauthorized
* 200 – OK

Sample response:

[

{

"date": "2017-05-05",

"count": 5

}

]

**GET /api/admins/{id}/profile**

Gets one admin profile

Status code:

* 401 – unauthorized
* 200 – OK

Sample response:

{

"id": 1,

"firstName": "rasto",

"lastName": "button",

"email": "rasto@button.sk",

"userName": "rastobutton",

"passwordChanged": false

}

**PUT /api/admins/{id}/profile**

Updates admin profile

Status code:

* 401 – unauthorized
* 200 – OK

Sample request:

{

"userName" : "rastobutton",

"password" : "rastobutton123",

"email" : "admin"

}

Sample response:

{

"id": 1,

"firstName": "rasto",

"lastName": "button",

"email": "admin",

"userName": "rastobutton",

"passwordChanged": false

}

Notes:

* If any field is empty, the value will not be updated.

**PUT /api/admins/{id}/profile/password**

Updates admins’ password

Status code:

* 401 – unauthorized
* 204 – OK

Sample request:

{

"password" : "rastobutton123",

}

Sample response:

{

"id": 1,

"firstName": "rasto",

"lastName": "button",

"email": "admin",

"userName": "rastobutton",

"passwordChanged": false

}

**GET /api/doctors?limit=x&start=y&sortBy=columnName&way=sortingWay&name=searchedName&spec=searchedSpecialization**

Gets filtered doctors

Status code:

* 401 – unauthorized
* 200 – OK

Sample response:

[

{

"id": 1,

"firstName": "doctor",

"lastName": "who",

"email": "doctor@who.sk",

"userName": "doctorwho",

"registrationDate": "2017-05-04",

"likes": 10,

"specialization": "DENTIST",

"phone": "0949473196",

"blocked": false,

"approved": false

},

{

"id": 2,

"firstName": "Obi-Wan",

"lastName": "Kenobi",

"email": "obi.wan@kenobi.sk",

"userName": "obiwan",

"registrationDate": "2017-05-04",

"likes": 5,

"specialization": "CARDIOLOGIST",

"phone": "0942223896",

"blocked": false,

"approved": false

}

]

Notes:

* If no query params are presented endpoint returns all doctors

**GET /api/doctors/count**

Gets number of all doctors

Status code:

* 401 – unauthorized
* 200 – OK

Sample response:

{

“count” : 10

}

**GET /api/doctors/likes**

Gets number of all likes

Status code:

* 401 – unauthorized
* 200 – OK

Sample response:

{

“count” : 108

}

**GET /api/doctors/specializations**

Gets all specializations

Status code:

* 401 – unauthorized
* 200 – OK

Sample response:

[

"DENTIST",

"CARDIOLOGIST",

"ORTHOPEDIST"

]

**GET /api/doctors/{id}/appointments**

Gets all doctors’ appointments

Status code:

* 401 – unauthorized
* 200 – OK

Sample response:

[

{

"id": 1,

"note": "test",

"patient": {

"id": 2,

"firstName": "Anakin",

"lastName": "Skywalker",

"email": "anakin@skywalker.com",

"userName": "anakin",

"registrationDate": "2017-05-05",

"blocked": false

},

"time": "13:54:59",

"date": "2017-05-05"

}

]

**PUT /api/doctors/{id}/approved**

Approves selected doctor

Status code:

* 401 – unauthorized
* 204 – OK

**PUT /api/doctors/{id}/blocked**

Blocks/unblocks selected doctor

Status code:

* 401 – unauthorized
* 204 – OK

Sample request:

{

"blocked" : true

}

**GET /api/patients?limit=x&start=y&sortBy=columnName&way=sortingWay&name=searchedName**

Gets filtered patients

Status code:

* 401 – unauthorized
* 200 – OK

Sample response:

[

{

"id": 1,

"firstName": "patient",

"lastName": "who",

"email": "patient@who.cz",

"userName": "patientwho",

"registrationDate": "2017-05-05",

"blocked": false

},

{

"id": 2,

"firstName": "Anakin",

"lastName": "Skywalker",

"email": "anakin@skywalker.com",

"userName": "anakin",

"registrationDate": "2017-05-05",

"blocked": false

}

]

Notes:

* If no query params are presented endpoint returns all patients

**GET /api/patients/count**

Gets number of all patients

Status code:

* 401 – unauthorized
* 200 – OK

Sample response:

{

“count” : 10

}

**GET /api/patients/{id}/appointments**

Gets all patients’ appointments

Status code:

* 401 – unauthorized
* 200 – OK

Sample response:

[

{

"id": 1,

"note": "test",

"doctor": {

"id": 1,

"firstName": "doctor",

"lastName": "who",

"email": "doctor@who.sk",

"userName": "doctorwho",

"registrationDate": "2017-05-04",

"likes": 10,

"specialization": "DENTIST",

"phone": "0949473196",

"blocked": true,

"approved": true

},

"time": "13:54:59",

"date": "2017-05-05"

}

]

**PUT /api/patients/{id}/blocked**

Blocks/unblocks selected patient

Status code:

* 401 – unauthorized
* 204 – OK

Sample request:

{

"blocked" : true

}