**Маршруты**

**AuthRegistration (регистрация нового пользователя)**

post('/auth/registration') – регистрация нового пользователя, если флаг доступ к системе - accessToSystem = TRUE, то в этом случае должен быть объект profile (ниже), и у профиля должна быть выбрана роль, если FALSE то profile должен быть null. На вход передается json с полями пользователя, выбранной специализацией и профилем в случае если есть доступ к системе вида:

|  |  |
| --- | --- |
| **Если есть доступ к системе** | **Если нет доступа к системе** |
| {          "lastname": "Иванова",          "firstname": "Елена",          "patronymic": "",          "phone": "+7(999)777-77-77",          "snils": "777777777",          "inn": "777777777777",          "description": "Директор ",          "birthday": "1975-05-25",          "accessToSystem": **true**,          "showInSchedule": **false**,          "published": **true**,          "specialization": {              "id": 1,              "name": "Педиатр",              "description": "",              "published": **true**          },          "profile": {              "username": "director@gmail.com",              "password": "qwerty",              "role": {                  "id": 2,                  "name": "ROLE\_DIRECTOR",                  "runame": "Пользователь"              }          }  } | {          "lastname": "Иванова",          "firstname": "Елена",          "patronymic": "",          "phone": "+7(999)777-77-77",          "snils": "777777777",          "inn": "777777777777",          "description": "Директор ",          "birthday": "1975-05-25",          "accessToSystem": **false**,          "showInSchedule": **false**,          "published": **true**,          "specialization": {              "id": 1,              "name": "Педиатр",              "description": "",              "published": **true**          }  } |

post('/auth/login') – аутентификация пользователя, на вход передается json c логином и паролем вида:

{

    "username": "ivanov\_ii@gmail.com",

    "password": "qwerty"

}

В случае аутентификации будет получен ответ с json, в котором будет jwt-токен для данного пользователя и информация о пользователе, вида:

{

    "doctor": {

        "id": 3,

        "lastname": "",

        "firstname": "Директор",

        "patronymic": "",

        "phone": "+7(999)777-77-77",

        "snils": "777777777",

        "inn": "777777777777",

        "description": "",

        "birthday": "1975-05-25",

        "accessToSystem": **true**,

        "showInSchedule": **false**,

        "published": **true**,

        "createdAt": "2024-12-05T13:34:28.74463",

        "updatedAt": "2024-12-05T13:34:28.74463",

        "specialization": {

            "id": 1,

            "name": "Педиатр",

            "description": "",

            "published": **true**

        },

        "profile": {

            "id": 2,

            "username": "director@gmail.com",

            "password": "$2a$10$5vKFjpZpKBLpmQD4lB9apeQf6Q77o/Zx1PBChryCddqaKkgHi0fW6",

            "createdAt": "2024-12-05T13:34:28.812139",

            "updatedAt": "2024-12-05T13:34:28.812139",

            "role": {

                "id": 4,

                "name": "ROLE\_DIRECTOR",

                "runame": "Директор"

            }

        }

    },

    "jwt-token": ""

}

**Doctor (врач, администратор, директор)**

get('/doctors?published=true&show\_in\_schedule=true'') – возвращает всех пользователей, в запросе могут быть параметры запроса флаг для врача показывается в документах или нет published (true или false) и флаг показывать ли в расписании show\_in\_schedule (true или false), тогда возвращает всех пользователей с учетом этого, пользователей возвращает в виде массива в json в следующем виде:

[

    {

        "id": 2,

        "lastname": "Иванов",

        "firstname": "Иван",

        "patronymic": "Иванович",

        "phone": "+7(999)888-88-88",

        "snils": "888888888",

        "inn": "888888888888",

        "description": "",

        "birthday": "1975-05-25",

        "accessToSystem": **true**,

        "showInSchedule": **true**,

        "published": **true**,

        "createdAt": "2024-12-04T22:12:37.659923",

        "updatedAt": "2024-12-05T17:06:25.898282",

        "specialization": {

            "id": 1,

            "name": "Педиатр",

            "description": "",

            "published": **true**

        },

        "profile": {

            "id": 4,

            "username": "ivanov\_ii@gmail.com",

  "password": "$2a$10$9YaXGYHoQ0yarn9KxkR1uuoVWp08oUD64QBcnEzwZ.ofgWRjnqfRy",

            "createdAt": "2024-12-05T16:06:59.865464",

            "updatedAt": "2024-12-05T17:06:26.004459",

            "role": {

                "id": 1,

                "name": "ROLE\_USER",

                "runame": "Пользователь"

            }

        }

    }

]

get('/users/:id') – возвращает одного пользователя по id в формате json следующего вида:

{

    "id": 2,

    "lastname": "Иванов",

    "firstname": "Иван",

    "patronymic": "Иванович",

    "phone": "+7(999)888-88-88",

    "snils": "888888888",

    "inn": "888888888888",

    "description": "",

    "birthday": "1975-05-25",

    "accessToSystem": **true**,

    "showInSchedule": **true**,

    "published": **true**,

    "createdAt": "2024-12-04T22:12:37.659923",

    "updatedAt": "2024-12-05T17:06:25.898282",

    "specialization": {

        "id": 1,

        "name": "Педиатр",

        "description": "",

        "published": **true**

    },

    "profile": {

        "id": 4,

        "username": "ivanov\_ii@gmail.com",

        "password": "$2a$10$9YaXGYHoQ0yarn9KxkR1uuoVWp08oUD64QBcnEzwZ.ofgWRjnqfRy",

        "createdAt": "2024-12-05T16:06:59.865464",

        "updatedAt": "2024-12-05T17:06:26.004459",

        "role": {

            "id": 1,

            "name": "ROLE\_USER",

            "runame": "Пользователь"

        }

    }

}

patch('/users/:id') - изменить одного пользователя по id, id прописывается в URL плюс json такой же по формату выше

**Patient (пациент)**

.post('/patients') – создание нового пациента, у пациента может быть указано несколько родителей, при этом родители могут быть новые или уже существующие в базе данных, если создается новый родитель, то указывается без id, если уже существующий то c id из базы данных, на вход передается json вида:

{

    "lastname": "Степаненко",

    "firstname": "Сергей",

    "patronymic": "Петрович",

    "description": "",

    "phone": "+78888885555",

    "email": "sergey@gmail.com",

    "policy": "444444555",

    "snils": "444444555",

    "medicalInsuranceCompany": "ООО Полюс",

    "inn": "444444444555",

    "disability": "отсутствует",

    "allergy": "отсутствует",

    "birthday": "17.06.2019",

    "published": **true**,

    "placeStudy": {

        "name": "Детский сад №47"

    },

    "document": {

        "name": "123456",

        "typeDoc": {

            "id": 2,

            "name": "свидетельство о рождении"

        }

    },

    "address": {

        "city": "Томск",

        "district": "",

        "village": "",

        "street": "Ленина",

        "house": "56",

        "apartment": "48",

        "postcode": "123456",

        "region": {

            "id": 1,

            "code": 70,

            "name": "Томская область"

        }

    },

    "blood": {

        "id": 1,

        "name": "O(I) Rh+ — первая положительная"

    },

    "gender": {

        "id": 1,

        "name": "мужской"

    },

    "parents": [

        {

            "lastname": "Степаненко",

            "firstname": "Петр",

            "patronymic": "Иванович",

            "parentNow": **true**,

            "powerOfAttorney": "отсутствует",

            "phone": "+79998888887",

            "email": "petr27@gmail.com",

            "policy": "333333337",

            "snils": "333333337",

            "medicalInsuranceCompany": "ООО Полюс",

            "inn": "333333333337",

            "job": "ООО Компания",

            "birthday": "27.09.1992",

            "published": **true**,

            "document": {

                "name": "6901 333333",

                "typeDoc": {

                    "id": 1,

                    "name": "паспорт"

                }

            },

            "address": {

                "city": "Томск",

                "district": "",

                "village": "",

                "street": "Ленина",

                "house": "56",

                "apartment": "48",

                "postcode": "123456",

                "region": {

                    "id": 1,

                    "code": 70,

                    "name": "Томская область"

                }

            },

            "typeRelationshipWithPatient": {

                "id": 2,

                "name": "отец"

            },

            "education": {

                "id": 1,

                "name": "высшее"

            },

            "blood": {

                "id": 1,

                "name": "O(I) Rh+ — первая положительная"

            },

            "typeEmployment": {

                "id": 1,

                "name": "работает"

            },

            "gender": {

                "id": 1,

                "name": "мужской"

            }

        }

    ]

}

.get('/ patients') – возвращает весь список пациентов в формате json

.get('/patients/:id') – возвращает одного пациента по id в формате json (формат такой же как и при создании выше, только с id)

.patch('/patients/:id') – изменяет существующего пациента по id, на вход передается json с измененными данными пациента ( формат аналогично созданию выше), при изменении пациента может быть добавлен новый родитель (и)

.get(‘/patients/search ?text\_search=88888’) – поиск пациента по имени, фамилии и номеру телефона, текст поиска указывается в виде параметра запроса с именем text\_search, его наличие обязательно, иначе будет ошибка, в ответ отправляется список пациентов, удовлетворяющих данному условию

**Calling (вызов врача на дом)**

.post('/callings') – создание нового вызова врача, на вход передается json с вызовом, выбранным уже существующим врачом и новым или существующим пациентом, если пациент новый, то без id, если существующий, то с id, а также с новым или существующим родителем(и), и выбранным прайсом, json следующего вида:

{

    "dateVisit": "10.12.2024",

    "dateCall": "09.12.2025",

    "timeStart": "09-00",

    "timeEnd": "10-00",

    "description": "",

    "doctor": {

        "id": 2,

        "lastname": "Иванов",

        "firstname": "Иван",

        "patronymic": "Иванович",

        "phone": "+7(999)888-88-88",

        "snils": "888888888",

        "inn": "888888888888",

        "description": "",

        "birthday": "1975-05-25",

        "accessToSystem": **true**,

        "showInSchedule": **true**,

        "published": **true**,

        "createdAt": "2024-12-04T22:12:37.659923",

        "updatedAt": "2024-12-05T17:06:25.898282",

        "specialization": {

            "id": 1,

            "name": "Педиатр",

            "description": "",

            "published": **true**

        },

        "profile": {

            "id": 4,

            "username": "ivanov\_ii@gmail.com",

            "password": "$2a$10$9YaXGYHoQ0yarn9KxkR1uuoVWp08oUD64QBcnEzwZ.ofgWRjnqfRy",

            "createdAt": "2024-12-05T16:06:59.865464",

            "updatedAt": "2024-12-05T17:06:26.004459",

            "role": {

                "id": 1,

                "name": "ROLE\_USER",

                "runame": "Пользователь"

            }

        }

    },

    "journal": {

        "diagnosis": "простыл",

        "complaint": "горло болит",

        "anamnesis": "",

        "therapy": "чай с лимоном",

        "recommendation": "побольше радости",

        "published": **true**

    },

    "price": {

        "id": 2,

        "name": "Вызов педиатра на дом, по Томску",

        "cost": 2500,

        "published": **true**

    },

    "patient": {

        "lastname": "Петренко",

        "firstname": "Игнат",

        "patronymic": "Васильевич",

        "description": "",

        "phone": "+78888885551",

        "email": "ignat@gmail.com",

        "policy": "444444551",

        "snils": "444444551",

        "medicalInsuranceCompany": "ООО Полюс",

        "inn": "444444444551",

        "disability": "отсутствует",

        "allergy": "отсутствует",

        "birthday": "17.08.2019",

        "published": **true**,

        "placeStudy": {

            "name": "Детский сад №25"

        },

        "document": {

            "name": "123456",

            "typeDoc": {

                "id": 2,

                "name": "свидетельство о рождении"

            }

        },

        "address": {

            "city": "Томск",

            "district": "",

            "village": "",

            "street": "Ивана Черных",

            "house": "12",

            "apartment": "78",

            "postcode": "123456",

            "region": {

                "id": 1,

                "code": 70,

                "name": "Томская область"

            }

        },

        "blood": {

            "id": 1,

            "name": "O(I) Rh+ — первая положительная"

        },

        "gender": {

            "id": 1,

            "name": "мужской"

        },

        "parents": [

            {

                "lastname": "Петренко",

                "firstname": "Василий",

                "patronymic": "Иванович",

                "parentNow": **true**,

                "powerOfAttorney": "отсутствует",

                "phone": "+79998888882",

                "email": "petrenko@gmail.com",

                "policy": "333333331",

                "snils": "333333331",

                "medicalInsuranceCompany": "ООО Полюс",

                "inn": "333333333331",

                "job": "ООО Компания",

                "birthday": "27.09.1980",

                "published": **true**,

                "document": {

                    "name": "6901 333331",

                    "typeDoc": {

                        "id": 1,

                        "name": "паспорт"

                    }

                },

                "address": {

                    "city": "Томск",

                    "district": "",

                    "village": "",

                    "street": "Ивана Черных",

                    "house": "12",

                    "apartment": "78",

                    "postcode": "123456",

                    "region": {

                        "id": 1,

                        "code": 70,

                        "name": "Томская область"

                    }

                },

                "typeRelationshipWithPatient": {

                    "id": 2,

                    "name": "отец"

                },

                "education": {

                    "id": 1,

                    "name": "высшее"

                },

                "blood": {

                    "id": 1,

                    "name": "O(I) Rh+ — первая положительная"

                },

                "typeEmployment": {

                    "id": 1,

                    "name": "работает"

                },

                "gender": {

                    "id": 1,

                    "name": "мужской"

                }

            }

        ]

    }

}

.get('/callings') – возвращает весь список вызовов врачей в формате json

.get('/callings/:id') – возвращает один вызов врача по id в формате json

.patch('/callings/:id') – изменяет существующий вызов врача по id, на вход передается json с новыми данными (аналогично созданию) только с id

**Contract (договор на медицинские услуги)**

.post('/contracts') – создание нового договора, на вход передается json с выбранным существующим врачом, новым (без id) или существующим пациентом (с id), с новыми (без id) или существующими родителями пациента (с id), с выбранным существующим типом договора, json следующего вида:

{

    "dateStart": "08.12.2024",

    "dateEnd": "07.12.2025",

    "description": "",

    "doctor": {

        "id": 2,

        "lastname": "Иванов",

        "firstname": "Иван",

        "patronymic": "Иванович",

        "phone": "+7(999)888-88-88",

        "snils": "888888888",

        "inn": "888888888888",

        "description": "",

        "birthday": "1975-05-25",

        "accessToSystem": **true**,

        "showInSchedule": **true**,

        "published": **true**,

        "createdAt": "2024-12-04T22:12:37.659923",

        "updatedAt": "2024-12-05T17:06:25.898282",

        "specialization": {

            "id": 1,

            "name": "Педиатр",

            "description": "",

            "published": **true**

        },

        "profile": {

            "id": 4,

            "username": "ivanov\_ii@gmail.com",

            "password": "$2a$10$9YaXGYHoQ0yarn9KxkR1uuoVWp08oUD64QBcnEzwZ.ofgWRjnqfRy",

            "createdAt": "2024-12-05T16:06:59.865464",

            "updatedAt": "2024-12-05T17:06:26.004459",

            "role": {

                "id": 1,

                "name": "ROLE\_USER",

                "runame": "Пользователь"

            }

        }

    },

    "patient": {

        "lastname": "Степаненко",

        "firstname": "Сергей",

        "patronymic": "Петрович",

        "description": "",

        "phone": "+78888885555",

        "email": "sergey@gmail.com",

        "policy": "444444555",

        "snils": "444444555",

        "medicalInsuranceCompany": "ООО Полюс",

        "inn": "444444444555",

        "disability": "отсутствует",

        "allergy": "отсутствует",

        "birthday": "17.06.2019",

        "published": **true**,

        "placeStudy": {

            "name": "Детский сад №47"

        },

        "document": {

            "name": "123456",

            "typeDoc": {

                "id": 2,

                "name": "свидетельство о рождении"

            }

        },

        "address": {

            "city": "Томск",

            "district": "",

            "village": "",

            "street": "Ленина",

            "house": "56",

            "apartment": "48",

            "postcode": "123456",

            "region": {

                "id": 1,

                "code": 70,

                "name": "Томская область"

            }

        },

        "blood": {

            "id": 1,

            "name": "O(I) Rh+ — первая положительная"

        },

        "gender": {

            "id": 1,

            "name": "мужской"

        },

        "parents": [

            {

                "lastname": "Степаненко",

                "firstname": "Петр",

                "patronymic": "Иванович",

                "parentNow": **true**,

                "powerOfAttorney": "отсутствует",

                "phone": "+79998888887",

                "email": "petr27@gmail.com",

                "policy": "333333337",

                "snils": "333333337",

                "medicalInsuranceCompany": "ООО Полюс",

                "inn": "333333333337",

                "job": "ООО Компания",

                "birthday": "27.09.1992",

                "published": **true**,

                "document": {

                    "name": "6901 333333",

                    "typeDoc": {

                        "id": 1,

                        "name": "паспорт"

                    }

                },

                "address": {

                    "city": "Томск",

                    "district": "",

                    "village": "",

                    "street": "Ленина",

                    "house": "56",

                    "apartment": "48",

                    "postcode": "123456",

                    "region": {

                        "id": 1,

                        "code": 70,

                        "name": "Томская область"

                    }

                },

                "typeRelationshipWithPatient": {

                    "id": 2,

                    "name": "отец"

                },

                "education": {

                    "id": 1,

                    "name": "высшее"

                },

                "blood": {

                    "id": 1,

                    "name": "O(I) Rh+ — первая положительная"

                },

                "typeEmployment": {

                    "id": 1,

                    "name": "работает"

                },

                "gender": {

                    "id": 1,

                    "name": "мужской"

                }

            }

        ]

    },

    "typeContract": {

        "id": 1,

        "name": "Малыш Хоум Стандарт ( от 0 до 2-х лет)",

        "cost": 9000

    }

}

.get('/contracts') – возвращает весь список договоров в формате json

.get('/contracts/:id') – возвращает один договор по id в формате json (аналогично созданию выше)

.patch('/contracts/:id') – изменяет существующий договор по id, на вход передается json с новыми данными (аналогично созданию) только с id

**Price (прайс услуг)**

.post('/prices') – создание нового прайса, на вход передается json с названием услуги, ценой и флагом отображения на фронтенде следующего вида:

{

    "name": "Вызов педиатра НА ДОМ, по Томску",

    "cost": 2500,

    "published": **true**

}

.get('/prices') – возвращает весь список услуг с ценами в формате json:

    {

        "id": 1,

        "name": "Прием (осмотр, консультация) врача-педиатра",

        "cost": 1800,

        "published": **true**,

        "createdAt": "2024-12-03T22:01:46.633918",

        "updatedAt": "2024-12-05T21:38:09.762742"

    },

    {

        "id": 2,

        "name": "Вызов педиатра НА ДОМ, по Томску",

        "cost": 2500,

        "published": **true**,

        "createdAt": "2024-12-05T21:39:33.700991",

        "updatedAt": "2024-12-05T21:39:33.700991"

    }

.get('/prices/:id') – возвращает один прайс на услугу по id в формате json

.patch('/prices/:id') – изменяет существующий прайс по id, на вход передается json с новым названием, ценой и флагом отображения на фронте (аналогично созданию)

**Region (регион: Томская область и др.)**

.post('/regions') – создание нового региона, на вход передается json с названием региона и кода региона вида:

{

    "code": 70,

    "name": "Томская область"

}

.get('/ regions') – возвращает весь список регионов в формате json

.get('/ regions/:id') – возвращает один регион по id в формате json

.patch('/ regions/:id') – изменяет существующий регион по id, на вход передается json с новым названием региона и кодом региона (аналогично созданию)

**Role (роль пользователя: пользователь, администратор и т.д.)**

get('/roles') - возвращает весь список ролей в формате json

добавление, изменение и удаление ролей отдельными методами в контроллере не предусмотрено, только через бэк

**Specialization (специализация врача: педиатр, лор, хирург и т.д.)**

.post('/specializations') - создание новой специализации, на вход передается json с названием специализации, описанием и published вида:

    {

        "name": "Невролог",

        "description": "",

        "published": true

    }

.get('/specializations?published=true') - возвращает весь список специализаций в формате json

.get('/specializations/:id') - возвращает одну специализацию по id в формате json

.patch('/specializations/:id') *-* изменяет существующую специализацию по id, на вход передается json с новым название, описанием и published (аналогично созданию)

**TypeDoc (тип документа: паспорт, св-во о рождении и др)**

.post('/typesDocs') – создание нового типа документа, на вход передается json с названием вида:

{

    "name": "свидетельство о рождении"

}

.get('/typesDocs') – возвращает весь список типов документов в формате json

.get('/typesDocs'/:id') – возвращает один тип документа по id в формате json

.patch('/typesDocs'/:id') – изменяет существующий тип документа по id, на вход передается json с новым названием

**Blood (группа крови)**

.post('/bloods') – создание новой группы крови, на вход передается json с названием группы крови вида:

    {

        "name": "O(I) Rh+ — первая положительная"

    }

.get('/bloods') – возвращает весь список групп крови в формате json

.get('/bloods/:id') – возвращает одну группу крови по id в формате json

.patch('/bloods/:id') – изменяет существующую группу крови по id, на вход передается json с новым названием группы крови

**TypeRelationshipWithPatient (тип отношения родителя с пациентом: мать, отец, бабушка и т.д.)**

.post('/typesRelationships') – создание нового типа отношения, на вход передается json с названием вида:

{

    "name": "мать"

}

.get('/typesRelationships') – возвращает весь список типов отношений в формате json

.get('/typesRelationships/:id') – возвращает один тип отношений по id в формате json

.patch('/typesRelationships/:id') – изменяет существующий тип отношений по id, на вход передается json с новым названием

**TypeEmployment (тип занятости родителя: работает, учится и т.д.)**

.post('/typesEmployment') – создание нового типа занятости родителя, на вход передается json с названием и ценой вида:

{

    "name": "работает"

}

.get('/typesEmployment') – возвращает весь список типов занятости родителей в формате json

.get('/typesEmployment/:id') – возвращает один тип занятости родителя по id в формате json

.patch('/typesEmployment/:id') – изменяет существующий тип занятости родителя по id, на вход передается json с новым названием

**Gender (гендер: женский, мужской, не выбран)**

.post('/genders') – создание нового гендера, на вход передается json с названием гендера вида:

    {

        "name": "мужской"

    }

.get('/genders') – возвращает весь список гендеров в формате json

.get('/genders/:id') – возвращает один гендер по id в формате json

.patch('/genders/:id') – изменяет существующий гендер по id, на вход передается json с новым названием гендера

**Education (образование родителей: высшее, среднее и т.д.)**

.post('/educations') – создание нового типа образования, на вход передается json с названием типа образования вида:

    {

        "name": "высшее"

    }

.get('/educations') – возвращает весь список типов образований в формате json

.get('/educations /:id') – возвращает один тип образования по id в формате json

.patch('/educations /:id') – изменяет существующий тип образования по id, на вход передается json с новым названием образования

**TypeContract (тип договора оказания услуг)**

.post('/typesContracts') – создание нового типа договора, на вход передается json с названием типа договора и ценой вида:

{

    "name": "Здоровый ребенок Педиатр рядом (c 2-х лет)",

    "cost": 5500

}

.get('/typesContracts') – возвращает весь список типов договоров в формате json

.get('/typesContracts/:id') – возвращает один тип договора по id в формате json

.patch('/typesContracts/:id') – изменяет существующий тип договора по id, на вход передается json с новым названием образования

**PlaceStudy (место учебы пациента)**

.get('/placesStudy') – возвращает весь список мест учебы пациентов в формате json

Создание места учебы и апдейт происходит совместно с созданием или апдейтом пациентов, отдельных методов в контроллере не предусмотрено.

Удаление не предусмотрено.