Маршруты

**AuthUser**

**post('/login', signin); - логин, пароль**

**User**

**post('/users', users.create) - создать юзера, приходит два объекта user и profile**

**get('/users?published=true&show\_in\_schedule=true'', users.findAll) - отдать всех юзеров, могут быть параметры запроса published=true&show\_in\_schedule=true, тогда отдать всех юзеров с учетом этого**

**get('/users'/published', users.findAllPublished) - отдельного ендпоинта не будет**

**get('/users'/doctors', users.findAllsSchedule)**

**get('/users'/:id', users.findOne) - вернуть одного юзера по id**

**put('/users'/:id', users.update) - изменить одного юзера по id, на вход приходит user и profile может быть**

**delete('/users'/:id', users.delete); - делаем при помощи put – меняем published на false**

Patient

**.post('/patients', patients.create) - создать пациента и родителя, приходит один пациент и массив родителей, может быть сколько угодно, создание паспорта, адреса, места учебы**

**.get('/patients?published=true&search= ', patients.findAll) - отдать всех пациентов с учетом параметров запроса**

.get('/patients/published', patients.findAllPublished)

**.get('/patients/:id', patients.findOne) – вернуть одного пациента и массив родителей по ид**

**.patch('/patients/:id', patients.update) - изменить одного пациента по ид, на вход также приходит пациент и массив родителей**

.delete('/patients/:id', patients.delete);

**.get(‘/patients/search’) -**

Parent

get('/parents', parents.findAll)

put('/parents/:id', parents.update);

Calling

**post('/callings', callings.create) - создание вызова, плюс создание или обновление пациента, если есть ид у пациента то обновление пациента и массива родителей, если нет то создание, на вход приходит объект вызов, пациент, массив родителей**

**get('/callings', callings.findAll); - вернуть все вызовы с пациентами и родителями**

**patch('/callings/:id', update); - изменить вызов по ид, пациента, массив родителей**

**get(/calling:id, findOne)**

Contract

**get('/contracts/:id', contracts.findOne) - логика такая же как и у вызовов**

**post('/contracts', contracts.create)**

**get('/contracts', contracts.findAll)**

**patch(/contracts'/:id', contracts.update);**

Price

**.post("/prices", prices.create) - создание прайса**

**.get("/prices", prices.findAll?published=true) - вернуть все прайсы с учетом параметров запроса**

.get("/prices/published", prices.findAllPublished)

**.get("/prices/:id", prices.findOne)**

**.patch("/prices/:id", prices.update)**

.delete("/prices/:id", prices.delete);

**Regions**

**get('/regions', regions.findAll); - для всех таблиц с константами сделать ендпоинты с гет запросами**

**Role**

**get('/roles', roles.findAll); - вернуть все роли**

Specialization

**.post('/specializations', specializations.create) - все просто**

**.get('/specializations?published=true', specializations.findAll)**

.get('/specializations/published', specializations.findAllPublished)

**.get('/specializations/:id', specializations.findOne)**

**.patch('/specializations/:id', specializations.update)**

.delete('/specializations/:id', specializations.delete);

**Type doc**

**get('/typedocs', typedocs.findAll); - вернуть все**

**Blood**

**- вернуть все**

**type\_relationship\_with\_patient**

**- вернуть все**

**type\_employment**

**- вернуть все**

**Gender**

* **Вернуть все**

**Education**

* **Вернуть все**

**type\_contract**

**- вернуть все**

**place\_study**

**- вернуть все**

