REST API Веб приложения

Описание

Готовое решение будет представлять собой веб приложение с программным интерфейсом, отвечающим архитектурному стилю REST и обладающим следующим функционалом: приём, обработка и отправка HTTP-запросов; составление запросов к реляционной базе данных и осуществление транзакций, сохранение данных; сериализация и десериализация данных в формате JSON.

Само приложение, для которого делается API, является сервисом для помощи людям с заболеваниями артерий нижних конечностей, которое позволяет с помощью рекомендаций врачей, имеющих доступ к паттернам ходьбы своих пациентов, назначать правильное лечение.

Наименование

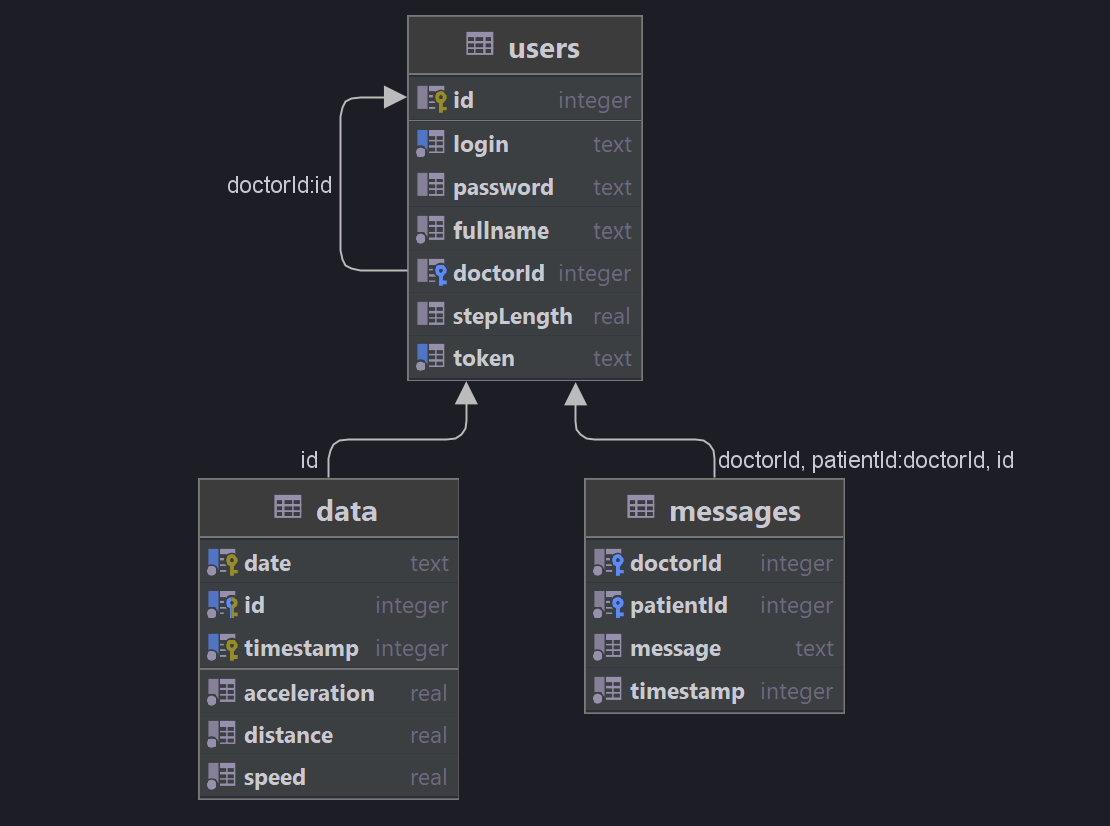
API Медицинского приложения

Предметная область

Медицинские сервисы, работа с пользователями, мобильная разработка, веб-разработка, REST API.

Данные

Диаграмма:



Создание таблиц:

*CREATE TABLE users (  
id INTEGER PRIMARY KEY,  
login TEXT UNIQUE NOT NULL,   
password TEXT,   
fullname TEXT NOT NULL,   
doctorId INTEGER,   
stepLength REAL,  
token TEXT NOT NULL,   
FOREIGN KEY (doctorId) REFERENCES users (id)  
);  
CREATE TABLE messages (  
doctorId INTEGER NOT NULL,   
patientId INTEGER NOT NULL,   
message TEXT NOT NULL,   
timestamp INTEGER NOT NULL,   
FOREIGN KEY (doctorId) REFERENCES users (id),  
FOREIGN KEY (patientId) REFERENCES users (id)  
);  
CREATE TABLE data (  
date TEXT NOT NULL,   
id TEXT NOT NULL,  
timestamp INTEGER NOT NULL,   
acceleration REAL NOT NULL,   
distance REAL NOT NULL,   
speed REAL NOT NULL,   
PRIMARY KEY (date, id, timestamp),   
FOREIGN KEY (id) REFERENCES users (id)  
)*

**Users:**

Отношение, отвечающее за представление пользователя (врача или пациента) в базе данных.

Атрибуты:

Id – id пользователя

login – уникальный логин пользователя,

password – пароль пользователя,

fullname – ФИО пользователя,

doctorId – хранит id врача, если пользователь является пациентом этого врача, либо NULL, если пользователь сам является врачом,

stepLength – длина шага пациента (NULL, если пользователь является врачом).

**Messages:**

Отношение, отвечающее за представление сообщений, отправляемых врачом пациенту, в базе данных.

Атрибуты:

doctorId – id врача, отправившего сообщения,

patientId – id пациента, которому было отправлено сообщение,

message – текст сообщения,

timestamp – время отправки сообщения (Unix time)

**Data:**

Отношение, отвечающее за представление медицинских данных пациентов по дням.

Атрибуты:

date – дата измерения в формате dd.mm.yyyy,

id – id пациента, чьи данные хранятся в записи,

timestamp – время замера данных (замер ведётся по пятиминутным интервалам),

acceleration – среднее ускорение за 5 минут,

distance – пройденное за 5 минут расстояние,

speed – средняя скорость за 5 минут.

Особенности

Использование ORM: на данном этапе не требуется.

Prepared запросы: есть.

Работа с NULL значениями: есть.

Транзакции: нет, кроме автоматически созданных самой СУБД.

Обработка ошибок в процессе работы с источником данных: есть.

Работа с BLOB: на данном этапе не требуется.

Доступ к метаинформации: на данном этапе не требуется.

Функционал

Пользовательские роли:

* Пациент
* Врач

Возможности пациента:

регистрация, авторизация, возможность добавить длину своего шага, возможность выбрать одного своего врача, возможность просматривать только свои медицинские данные, возможность отсылать свои медицинские данные на сервер, возможность просматривать сообщения своего врача.

Возможности врача:

регистрация, авторизация, возможность иметь нескольких пациентов, возможность просматривать медицинские данные только своих пациентов, возможность отсылать сообщения только своим пациентам

Технологии разработки

Общая бизнес-логика

Так как API является лишь интерфейсом для обмена запросами, то вся логика приложения будет построена на том, что программа принимает соответствующий HTTP-запрос, обрабатывает его, по надобности подключаясь к источнику данных, затем формирует ответный запрос и отправляет его клиенту.

Язык программирования

Для написания приложения используется язык программирования Python вместе с веб-фреймворком Flask.

СУБД

Для тестовой версии программы в качестве СУБД используется SQLite.