***Пользовательская часть***

Запуск скрипта через REST API пройзводится посредством отправки POST запроса на относительный адрес “/calc/” c передачей в формате json следующих параметров:

1. intParam – целое число
2. arParams – массив целых чисел

***Пример вызова:***

curl\_setopt\_array($curl, **array**(

  CURLOPT\_URL => 'http://exbico/public/calc',

  CURLOPT\_RETURNTRANSFER => **true**,

  CURLOPT\_ENCODING => '',

  CURLOPT\_MAXREDIRS => 10,

  CURLOPT\_TIMEOUT => 0,

  CURLOPT\_FOLLOWLOCATION => **true**,

  CURLOPT\_HTTP\_VERSION => CURL\_HTTP\_VERSION\_1\_1,

  CURLOPT\_CUSTOMREQUEST => 'POST',

  CURLOPT\_POSTFIELDS =>'{

    "intParam" : 2,

    "arParams" : [4, 4, 4, 4, 2, 4, 4]

}',

  CURLOPT\_HTTPHEADER => **array**(

    'Content-Type: application/json'

  ),

));

$response = curl\_exec($curl);

curl\_close($curl);

**echo** $response;

***Техническая часть***

Используется фреймворк Laravel.

Пути:

**Контроллеры:** */app/Http/Controllers/*

**Модели:** */app/*

**Файлы миграций:** /*database/migrations/*

**Сидеры:** */database/seeds/*

**Шаблоны:** */resources/views/*

**Файл класса CSRF верификации:** */app/Http/Middleware/VerifyCsrfToken.php*

Обращение по относительному адресу */calc/* обрабатывается методом *store* контроллера *CalcController* c передачей в качестве входных параметров запроса и объекта класса Calc одноимённой модели. Логика контроллера обеспечивает запись в таблицу *“calcs”* базы данных:

параметров запроса - числа **intParam** (в *task\_int*) и массива **arParams** (сериализованным массивом в *task\_array*), а так же ответа в виде индекса элемента массива, следующего за разделителем (в *answer*) и id пользователя (на текущей стадии выполнения в базу заносится пользователь с ID = 1 – статический параметр в *user\_id*).

Ответ определяется методом *getTaskAnswer* класса *Calc*, вызванным из контекста объекта модели, переданного в контроллер.

Ответ выводится в шаблоне *calc.blade.php* в виде целого числа.

В целях тестирования именованный роут *“calc”* вынесен в исключения в файле *VerifyCsrfToken.php*

Для хранения данных о пользователях предназначена таблица *users*, добавлен тестовый пользователь *Sergey*. Таблицы созданы посредством миграций, пользователь добавлен через соответствующий *Seeder*.