**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 6](#_Toc106292941)

[ГЛАВА 1. АНАЛИТЕЧСКАЯ ЧАСТЬ 8](#_Toc106292942)

[*1.1 Цель разработки* 8](#_Toc106292943)

[*1.2 Обзор средств программирования* 8](#_Toc106292944)

[*1.3 Описания языка программирования* 10](#_Toc106292945)

[ГЛАВА 2. ПРАКТИЧЕСКАЯ ЧАСТЬ 16](#_Toc106292946)

[*2.1 Постановка задачи* 16](#_Toc106292947)

[*2.2 Описание алгоритм* 17](#_Toc106292948)

[*2.3 Текст программы с описанием* 18](#_Toc106292949)

[2.4 Описание процесса отладки программы 38](#_Toc106292950)

[2.5 Оценка результатов решения задачи 38](#_Toc106292951)

[ЗАКЛЮЧЕНИЕ 40](#_Toc106292952)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 41](#_Toc106292953)

[ПРИЛОЖЕНИЯ 42](#_Toc106292954)

## ВВЕДЕНИЕ

Один из самых важных аспектов нашего здоровья - это правильное питание. В наше время многие люди сталкиваются с проблемами, связанными с неправильным питанием, такими как ожирение, диабет, сердечные заболевания и т.д.

В связи с тем, что питание играет такую важную роль в нашей жизни, можно предположить, что знание о правильном питании должно быть распространено как можно шире. Но для освоения этих знаний нужно затратить большое количество времени и сил.

За частую у людей не хватает свободного времени для получения необходимых знаний и навыков в этой сфере. Именно поэтому появились множество организаций предоставляющие сбалансированные рационы питания на весь день.

Одной из таких организаций является кухня BaskayTime. Заявки для получения рационов осуществляются по телефону или же через социальную сеть Инстаграмм, Что не является самым удобным методом проведения подобных операций.

Таким образом целью дипломной работы является разработка Web – Сайта с удобным и понятным интерфейсом для того что бы пользователь мог в пару кликов самостоятельно составить для себя рацион на предстоящий день. Либо оставить заявку с просьбой перезвонить для подробной консультации.

# ГЛАВА 1. АНАЛИТЕЧСКАЯ ЧАСТЬ

## *1.1 Цель разработки*

Разработать Web-сайт «Правильного питания» “BaskayTime” на фреймворках Vue.js[8] и Laravel[9].

Целью этого Web-сайта является повышение удобства составления подходящего рациона.

## *1.2 Обзор средств программирования*

Для проектирования и написания мобильного приложения были отобраны несколько редакторов кода: SublimeText 4, Visual Studio Code и WebStorm.

Редактор кода отбирался по следующим критериям:

* Простой и функциональный редактор кода с подсветкой синтаксиса, авто-форматированием и авто-отступами для поддерживаемых языков.
* Навигация в коде и файлах.
* Помощь при написании кода, включающая в себя автодополнение, автоимпорт, шаблоны кода, генерация, подсветка и многое другое.
* Большое количество инспекций кода.
* Глубокая кастомизация интерфейса вплоть до курсора мыши действия которого можно поменять в настройках.
* Встроенная командная строка, для более удобного запуска сборки клиентской части Web-сайта.

Всем критериям указанным выше полностью соответствовал Visual Studio Code (VS Code)[2]

* Visual Studio Code (VS Code) — это [редактор исходного кода](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B4%D0%B0%D0%BA%D1%82%D0%BE%D1%80_%D0%B8%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BA%D0%BE%D0%B4%D0%B0), разработанный [Microsoft](https://ru.wikipedia.org/wiki/Microsoft) для [Windows](https://ru.wikipedia.org/wiki/Windows), [Linux](https://ru.wikipedia.org/wiki/Linux) и [macOS](https://ru.wikipedia.org/wiki/MacOS). Позиционируется как «лёгкий» редактор кода для [кроссплатформенной](https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%BE%D1%81%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D1%81%D1%82%D1%8C) разработки [веб](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5)- и [облачных](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D0%BB%D0%B0%D1%87%D0%BD%D1%8B%D0%B5_%D0%B2%D1%8B%D1%87%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F) приложений. Включает в себя [отладчик](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BB%D0%B0%D0%B4%D1%87%D0%B8%D0%BA), [подсветку синтаксиса](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%B4%D1%81%D0%B2%D0%B5%D1%82%D0%BA%D0%B0_%D1%81%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81%D0%B0) и средства для [рефакторинга](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D1%84%D0%B0%D0%BA%D1%82%D0%BE%D1%80%D0%B8%D0%BD%D0%B3). Имеет широкие возможности для [кастомизации](https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D1%81%D1%82%D0%BE%D0%BC%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F): пользовательские темы, [сочетания клавиш](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%87%D0%B5%D1%82%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BA%D0%BB%D0%B0%D0%B2%D0%B8%D1%88) и [файлы конфигурации](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB_%D0%BA%D0%BE%D0%BD%D1%84%D0%B8%D0%B3%D1%83%D1%80%D0%B0%D1%86%D0%B8%D0%B8). Распространяется бесплатно, разрабатывается как программное обеспечение с [открытым исходным кодом](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BA%D1%80%D1%8B%D1%82%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5).

Visual Studio Code выполняет инспекцию код параллельная написанию кода, авто-дополнение, в том числе основываясь на информации, полученной во время исполнения кода, навигацию по коду.

Для разработки Web-сайта, были выбраны, следующие Фреймворки и Библиотеки: JavaScript/VueJs, PHP/Laravel .

* **Vue.js[8]** (также *Vue*; [/vjuː/](https://ru.wikipedia.org/wiki/%D0%A2%D0%B0%D0%B1%D0%BB%D0%B8%D1%86%D0%B0_%D0%9C%D0%A4%D0%90_%D0%B4%D0%BB%D1%8F_%D0%B0%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%BE%D0%B3%D0%BE_%D1%8F%D0%B7%D1%8B%D0%BA%D0%B0)) — [JavaScript](https://ru.wikipedia.org/wiki/JavaScript)-[фреймворк](https://ru.wikipedia.org/wiki/%D0%A4%D1%80%D0%B5%D0%B9%D0%BC%D0%B2%D0%BE%D1%80%D0%BA) с [открытым исходным кодом](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BA%D1%80%D1%8B%D1%82%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) для создания [пользовательских интерфейсов](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F). Легко интегрируется в проекты с использованием других [JavaScript-библиотек](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_JavaScript). Может функционировать как [веб-фреймворк](https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D1%80%D0%BA%D0%B0%D1%81_%D0%B2%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B9) для разработки [одностраничных приложений](https://ru.wikipedia.org/wiki/%D0%9E%D0%B4%D0%BD%D0%BE%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%87%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5) в [реактивном](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B0%D0%BA%D1%82%D0%B8%D0%B2%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) стиле. Может функционировать как [веб-фреймворк](https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D1%80%D0%BA%D0%B0%D1%81_%D0%B2%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B9) для разработки [одностраничных приложений](https://ru.wikipedia.org/wiki/%D0%9E%D0%B4%D0%BD%D0%BE%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%87%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5) в [реактивном](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B0%D0%BA%D1%82%D0%B8%D0%B2%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) стиле.
* Разработчики называют Vue.js прогрессивным и постепенно адаптируемым по сравнению с другими [веб-фреймворками](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%84%D1%80%D0%B5%D0%B9%D0%BC%D0%B2%D0%BE%D1%80%D0%BA).
* Это позволяет разработчику настроить структуру приложения в соответствии с собственными требованиями. Разработчики считают Vue.js более простым в освоении, чем [AngularJS](https://ru.wikipedia.org/wiki/AngularJS), поскольку [API](https://ru.wikipedia.org/wiki/API) построен намного проще в освоении. В Vue.js можно использовать только знания [JavaScript](https://ru.wikipedia.org/wiki/JavaScript) и HTML. Возможно применение Typescript. У Vue.js есть собственная официальная достаточно богатая документация на многих языках, выложенная на vuejs.org, которая может послужить примером в объяснении проектирования и разработки в браузере. В Vue.js реализуется шаблон MVVM, Vue.js предлагает возможность привязки данных на Javascript, так что вывод и ввод данных сопрягаются непосредственно с источником данных. Таким образом, режим ручного определения данных (например, через jQuery) из [HTML](https://ru.wikipedia.org/wiki/HTML)-[DOM](https://ru.wikipedia.org/wiki/DOM) не нужен. При этом нет необходимости в никаких дополнительных аннотациях, как в [Knockout.js](https://ru.wikipedia.org/wiki/Knockout.js), объявленные в Vue-Element обычные переменные JavaScript включаются в качестве реактивных элементов.
* **Реактивность** - означает, что представление в модели MVC изменяется по мере изменения модели. В Vue разработчики просто привязывают представление к соответствующей модели, и Vue автоматически наблюдает за изменениями в модели и перерисовывает представление. Эта функция делает управление состоянием Vue довольно простым и интуитивно понятным

## *1.3 Описания языка программирования*

**JavaScript(JS) –** этомультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Является реализацией спецификации ECMAScript (стандарт ECMA-262).

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

Основные архитектурные черты: динамическая типизация, слабая типизация, автоматическое управление памятью, прототипное программирование, функции как объекты первого класса.

*Преимущества языка*

* Востребованный язык программирования. Если верить статистике, Javascript входит в топ-3 языков программирования в мире.
* Простота и рациональность применения. Несложные задачи можно решать быстро, порой достаточно 5 минут. Для непростых в интернете полно вариантов решения, остается лишь выбрать подходящий и использовать его.
* Легкость в освоении. Сначала код, возможно, покажется очень трудным, однако это ощущение быстро пройдет. Кроме того, сильно мотивирует визуальное отображение действий.

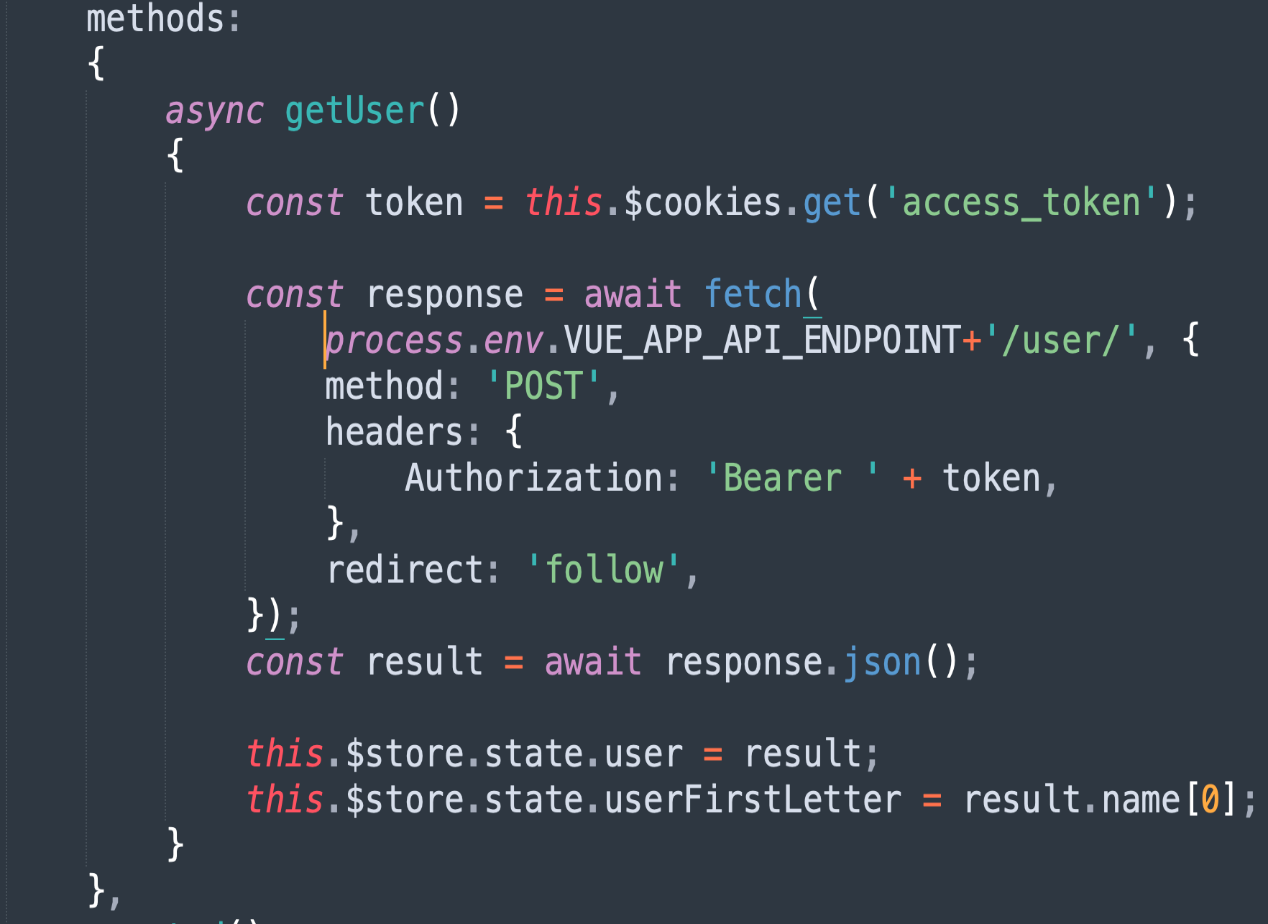


Рис 1.1 Пример JavaScript кода

* На данный VueJs момент поддерживается создателем Эваном Ю и остальными активными членами основной команды из различных компаний, таких как [Netlify](https://ru.wikipedia.org/w/index.php?title=Netlify&action=edit&redlink=1), [Netguru](https://ru.wikipedia.org/w/index.php?title=Netguru&action=edit&redlink=1), [Baidu](https://ru.wikipedia.org/wiki/Baidu), [Livestorm](https://ru.wikipedia.org/w/index.php?title=Livestorm&action=edit&redlink=1).

В качестве основного дополнения для VueJs используется библиотека [VueRouter](https://reactnavigation.org/).

Для серверной части использовался язык программирования PHP.

**PHP** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *PHP: Hypertext Preprocessor* — «PHP: [препроцессор](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D1%80) [гипертекста](https://ru.wikipedia.org/wiki/%D0%93%D0%B8%D0%BF%D0%B5%D1%80%D1%82%D0%B5%D0%BA%D1%81%D1%82)»; первоначально PHP/FI (Personal Home Page / Form Interpreter), а позже названный *Personal Home Page Tools*[[11]](https://ru.wikipedia.org/wiki/PHP#cite_note-off_history-11) — «Инструменты для создания персональных веб-страниц») — [C-подобный](https://ru.wikipedia.org/wiki/C-%D0%BF%D0%BE%D0%B4%D0%BE%D0%B1%D0%BD%D1%8B%D0%B9_%D1%81%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81) [скриптовый язык](https://ru.wikipedia.org/wiki/%D0%A1%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством [хостинг-провайдеров](https://ru.wikipedia.org/wiki/%D0%A5%D0%BE%D1%81%D1%82%D0%B8%D0%BD%D0%B3) и является одним из лидеров среди языков, применяющихся для создания [динамических веб-сайтов](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%BD%D0%B0%D0%BC%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D1%81%D0%B0%D0%B9%D1%82).

Язык и его интерпретатор ([Zend Engine](https://ru.wikipedia.org/wiki/Zend_Engine)) разрабатываются группой энтузиастов в рамках проекта с [открытым кодом](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BA%D1%80%D1%8B%D1%82%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5). Проект распространяется под [собственной лицензией](https://ru.wikipedia.org/wiki/%D0%9B%D0%B8%D1%86%D0%B5%D0%BD%D0%B7%D0%B8%D1%8F_PHP), несовместимой с [GNU GPL](https://ru.wikipedia.org/wiki/GNU_GPL).

В области веб-программирования, в частности серверной части, PHP — один из популярных [сценарных языков](https://ru.wikipedia.org/wiki/%D0%A1%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) (наряду с [JSP](https://ru.wikipedia.org/wiki/JSP), [Perl](https://ru.wikipedia.org/wiki/Perl) и языками, используемыми в [ASP.NET](https://ru.wikipedia.org/wiki/ASP.NET)).

Популярность в области построения [веб-сайтов](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B0%D0%B9%D1%82) определяется наличием большого набора встроенных средств и дополнительных модулей для разработки веб-приложений[[15]](https://ru.wikipedia.org/wiki/PHP#cite_note-15). Основные из них:

Автоматическое извлечение [POST](https://ru.wikipedia.org/wiki/HTTP#POST)- и [GET](https://ru.wikipedia.org/wiki/HTTP#GET)-параметров, а также переменных окружения веб-сервера в предопределённые массивы;

Взаимодействие с большим количеством различных систем управления базами данных через дополнительные модули ([MySQL](https://ru.wikipedia.org/wiki/MySQL), [MySQLi](https://ru.wikipedia.org/wiki/MySQLi), [SQLite](https://ru.wikipedia.org/wiki/SQLite), [PostgreSQL](https://ru.wikipedia.org/wiki/PostgreSQL), [Oracle Database](https://ru.wikipedia.org/wiki/Oracle_Database) (OCI8), [Microsoft SQL Server](https://ru.wikipedia.org/wiki/Microsoft_SQL_Server), [Sybase](https://ru.wikipedia.org/wiki/Sybase), [ODBC](https://ru.wikipedia.org/wiki/ODBC), [mSQL](https://ru.wikipedia.org/wiki/MSQL), [IBM DB2](https://ru.wikipedia.org/wiki/IBM_DB2), [Cloudscape](https://ru.wikipedia.org/wiki/Cloudscape) и [Apache Derby](https://ru.wikipedia.org/wiki/Apache_Derby), [Informix](https://ru.wikipedia.org/wiki/Informix), [Ovrimos SQL](https://ru.wikipedia.org/w/index.php?title=Ovrimos_SQL&action=edit&redlink=1), [Lotus Notes](https://ru.wikipedia.org/wiki/Lotus_Notes), [DB++](https://ru.wikipedia.org/w/index.php?title=DB%2B%2B&action=edit&redlink=1), [DBM](https://ru.wikipedia.org/wiki/DBM), [dBase](https://ru.wikipedia.org/wiki/DBase), [DBX](https://ru.wikipedia.org/w/index.php?title=DBX_(%D0%A1%D0%A3%D0%91%D0%94)&action=edit&redlink=1), [FrontBase](https://ru.wikipedia.org/w/index.php?title=FrontBase&action=edit&redlink=1), [FilePro](https://ru.wikipedia.org/w/index.php?title=FilePro&action=edit&redlink=1), [Ingres II](https://ru.wikipedia.org/w/index.php?title=Ingres_II&action=edit&redlink=1), [SESAM](https://ru.wikipedia.org/w/index.php?title=SESAM&action=edit&redlink=1), [Firebird](https://ru.wikipedia.org/wiki/Firebird) и [InterBase](https://ru.wikipedia.org/wiki/InterBase), [Paradox File Access](https://ru.wikipedia.org/w/index.php?title=Paradox_File_Access&action=edit&redlink=1), [MaxDB](https://ru.wikipedia.org/w/index.php?title=MaxDB&action=edit&redlink=1), [интерфейс](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81) [PDO](https://ru.wikipedia.org/wiki/PDO), [Redis](https://ru.wikipedia.org/wiki/Redis));

* Автоматизированная отправка [HTTP-заголовков](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%B3%D0%BE%D0%BB%D0%BE%D0%B2%D0%BA%D0%B8_HTTP);
* Работа с HTTP-авторизацией;
* Работа с [cookies](https://ru.wikipedia.org/wiki/Cookies) и сессиями;
* Работа с локальными и удалёнными файлами, [сокетами](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D0%BA%D0%B5%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D1%8B%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81));
* Обработка файлов, загружаемых на сервер;
* Работа с [XForms](https://ru.wikipedia.org/wiki/XForms).
* Создание и работа с [API](https://ru.wikipedia.org/wiki/API)
* Создание приложение с [графическим интерфейсом пользователя](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F) при использовании Фреймворка [Qt Designer](https://ru.wikipedia.org/wiki/Qt_Designer)
* Создание консольных приложений

Язык автоматически поддерживает [HTTP Cookies](https://ru.wikipedia.org/wiki/HTTP-Cookies) в соответствии со стандартами Netscape. Это позволяет проводить установку и чтение небольших сегментов данных на стороне клиента. Работа с Cookies организована посредством сеансов (сессий). У сессий есть срок действия (после его истечения данные удаляются), в сессиях можно хранить и редактировать разные типы данных, в том числе сериализованные — пропущенные через serialize (процесс происходит автоматически), PHP-объекты.

Среди сайтов, использующих PHP — [Facebook](https://ru.wikipedia.org/wiki/Facebook), [Wikipedia](https://ru.wikipedia.org/wiki/Wikipedia), [Yahoo!](https://ru.wikipedia.org/wiki/Yahoo!), [Baidu](https://ru.wikipedia.org/wiki/Baidu).

Среди фреймворков PHP выбор пал на Larave.

**Laravel** — бесплатный веб-[фреймворк](https://ru.wikipedia.org/wiki/%D0%A4%D1%80%D0%B5%D0%B9%D0%BC%D0%B2%D0%BE%D1%80%D0%BA) с [открытым кодом](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BA%D1%80%D1%8B%D1%82%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4), предназначенный для разработки с использованием архитектурной модели [MVC](https://ru.wikipedia.org/wiki/Model-View-Controller) ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Model View Controller* — модель-представление-контроллер). Laravel выпущен под [лицензией MIT](https://ru.wikipedia.org/wiki/%D0%9B%D0%B8%D1%86%D0%B5%D0%BD%D0%B7%D0%B8%D1%8F_MIT).

Исходный код проекта размещается на [GitHub](https://ru.wikipedia.org/wiki/GitHub). В результате опроса *sitepoint.com* в декабре 2013 года о самых популярных [PHP](https://ru.wikipedia.org/wiki/PHP)-фреймворках Laravel занял место самого многообещающего проекта на 2014 год.

В 2015 году в результате опроса *sitepoint.com* по использованию [PHP](https://ru.wikipedia.org/wiki/PHP)-фреймворков среди программистов занял первое место в номинациях:

* Фреймворк корпоративного уровня
* Фреймворк для личных проектов
* Кладезь антипаттернов

**Преимущества Laravel по сравнению с другими фреймворками**.

Изучив ключевые особенности технологии и опробовав их на практике, мы можем перечислить основные преимущества Laravel

* Высокая производительность - веская причина выбрать Laravel заключается в том, что он расширяет возможности создаваемых приложений. Данный фреймворк помогает разрабатывать высокопроизводительные приложения, даже если определенные функции оказывают повышенную нагрузку на веб-сервис. Вы можете интегрировать такие инструменты, как [Memcached](https://memcached.org/) и [Redis](https://redis.io/) в фреймворк Laravel и легко разрабатывать веб-приложения; они также помогают повысить производительность веб-сервисов. Не случайно Laravel признан самой высокоэффективной платформой современности.
* Интегрированная система авторизации - Фреймворк Laravel позволяет создавать многофункциональные приложения быстрее и проще. Например, Laravel позволяет без особых усилий встроить в приложение надежную и защищенную систему аутентификации пользователей.
* Открытый исходный код и мощное сообщество - Технологии с открытым исходным кодом всегда хороши, поскольку они позволяют создавать потрясающие приложения без необходимостиц платить за лицензию. Это отличный опыт для разработчиков, которые также могут сделать собственный вклад в развитие технологии – в отличие от платных продуктов. Кроме того, наличие [мощного и дружественного сообщества](https://laravel.io/) означает поддержку, которую вы получаете во время проектирования и разработки приложений – особенно если вы сталкиваетесь с серьезной проблемой. Советы опытных разработчиков помогут сделать ваш проект более гибким и масштабируемым.
* Миграция баз данных. Еще одним преимуществом фреймворка Laravel является миграция баз данных. Эта процедура упрощает процесс создания и управления базами данных, и разработчик может отменить изменения, внесенные в таблицы. Программисты могут легко обмениваться базами данных с другими разработчиками, что делает совместную работу более согласованной, организованной и простой. Функция сопоставления обеспечивает простое обслуживание баз данных без необходимости их повторного создания.
* Объектно-ориентированные библиотеки Laravel предлагает предустановленные объектно-ориентированные библиотеки. Среди них, к примеру, есть библиотека аутентификации, в которую включено множество полезных функций. Все библиотеки просты в использовании. Фреймворк также предоставляет дополнительные модули, которые позволяют разработчикам делать приложения более отзывчивыми и структурированными. Некоторые модули по функциональности равнозначны готовым приложениям.
* Простота тестирования Опции тестирования фреймворка имеют первостепенное значение, поскольку тесты помогают разработчикам обеспечить наилучший пользовательский опыт. Laravel, будучи передовым фреймворком, предлагает модульное тестирование, которое упрощает проверку и отладку каждого модуля и компонента в отдельности. Модульное тестирование поможет вам найти все неполадки в проекте, оно гарантирует отсутствие ошибок в готовом приложении и обеспечит первоклассный пользовательский опыт при использовании созданного вами сервиса.
* Локализация Поддержка многоязычного интерфейса в приложении необходима во многих случаях. Локализация приложения позволяет вывести бизнес на международный уровень. Упрощенная система локализации – одно из веских преимуществ Laravel.

# ГЛАВА 2. ПРАКТИЧЕСКАЯ ЧАСТЬ

## *2.1 Постановка задачи*

Разработать Web-сайт на фреймворках Vue.js и Laravel для людей желающих избавиться от лишнего веса и поправить здоровье. Web-сайт должен содержать информацию по следующим разделам:

1. Главная – на этой странице расположен баннер – призыва с кнопкой отсылающей на отдел в котором пользователь может составить для себя рацион питания. Слайдер с предыдущими клиентами кухни с отзывами о ней. После идёт блок с полезной информацией. Затем для удобства расположена секция с часто задаваемыми вопросами и ответами на них.
2. Войти/Зарегистрироваться – стандартная форма для регистрации или входа пользователя в систему.
3. Марафоны – На странице “Марафоны” расположены все проходившие раннее марафоны, с подробной информацией о победителе и разыгранных призах. Но помимо этого есть информация о текущем марафоне и возможность для каждого посетителя сайта поучаствовать в нем путем заполнения несложной формы и выполнением всех условий.

*Функциональные требования к системе*

1. Возможность использовать доступный функционал Web-сайта.
2. Отзывчивая верстка для разных устройств, в том числе и для планшетов с минимальной шириной экрана 800px.
3. Реализация структуры Web-сайта.
4. Наличие админ – панели для удобного отслеживания всез заказов и заявок, а так же удобного редактирования меню
5. Интерфейс.

*Программные требования*

* Компилятор клиентской части Web-сайта Vue CLI.
* Версия Node JS не меньше 14 версии
* Объем оперативной памяти не меньше 2гб.
* Наличие любого браузера на устройстве, на котором ведется разработка

Для реализации программы был выбран язык программирования и фреймворки

* + JavaScript
  + Vue JS
  + PHP
  + Node.js
  + Laravel

## *2.2 Описание алгоритм*

Для того чтобы приложение работало ему нужно подключение к интернету.

Не загруженный интерфейс, чтобы любой человек мог спокойно ориентироваться по приложению.

Простой и понятный функционал.

Не броский дизайн, который будет приятный и в тоже время, не имеющий лишние элементы.

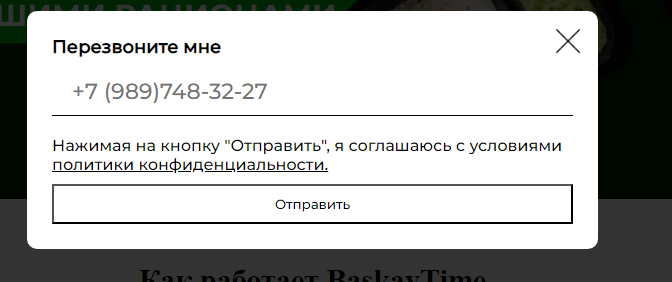
## *2.3 Текст программы с описанием*

При открытии Web-сайта, появляется главная страница с плавающей навигацией по сайту (шапка)



Рисунок 2.1 Шапка сайта

При нажатии на текст “Перезвонить мне” открывается окно с вводом номера телефона для последующего звонка со стороны администрации сайта



Сразу после этого расположен баннер с кнопкой отсылающей на отдел составления заказа

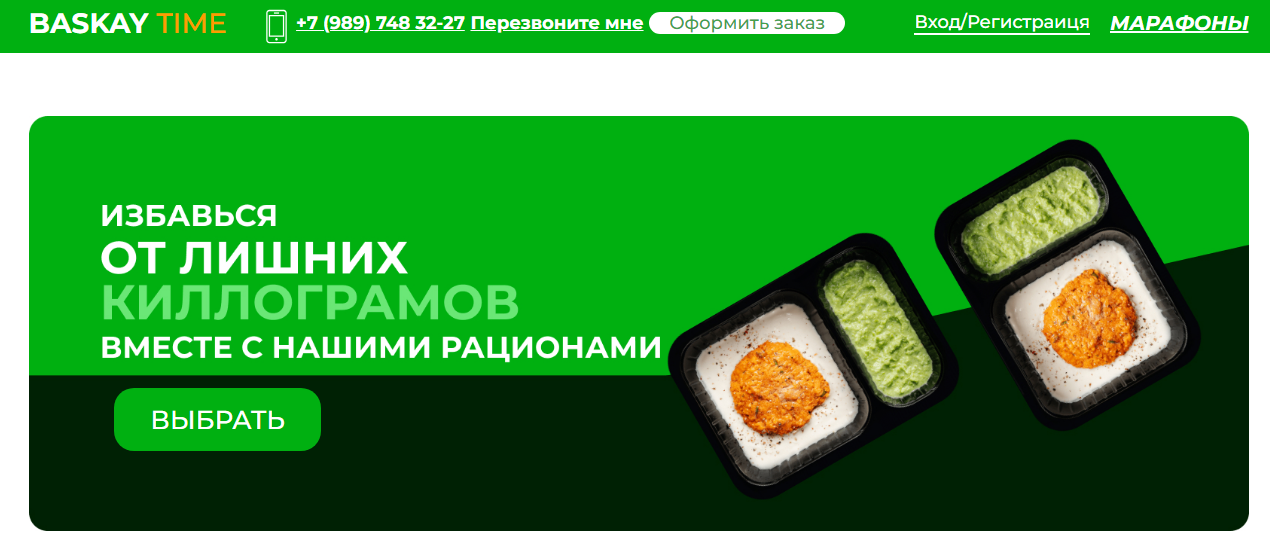


Рисунок 2.2 Главная страница с баннером

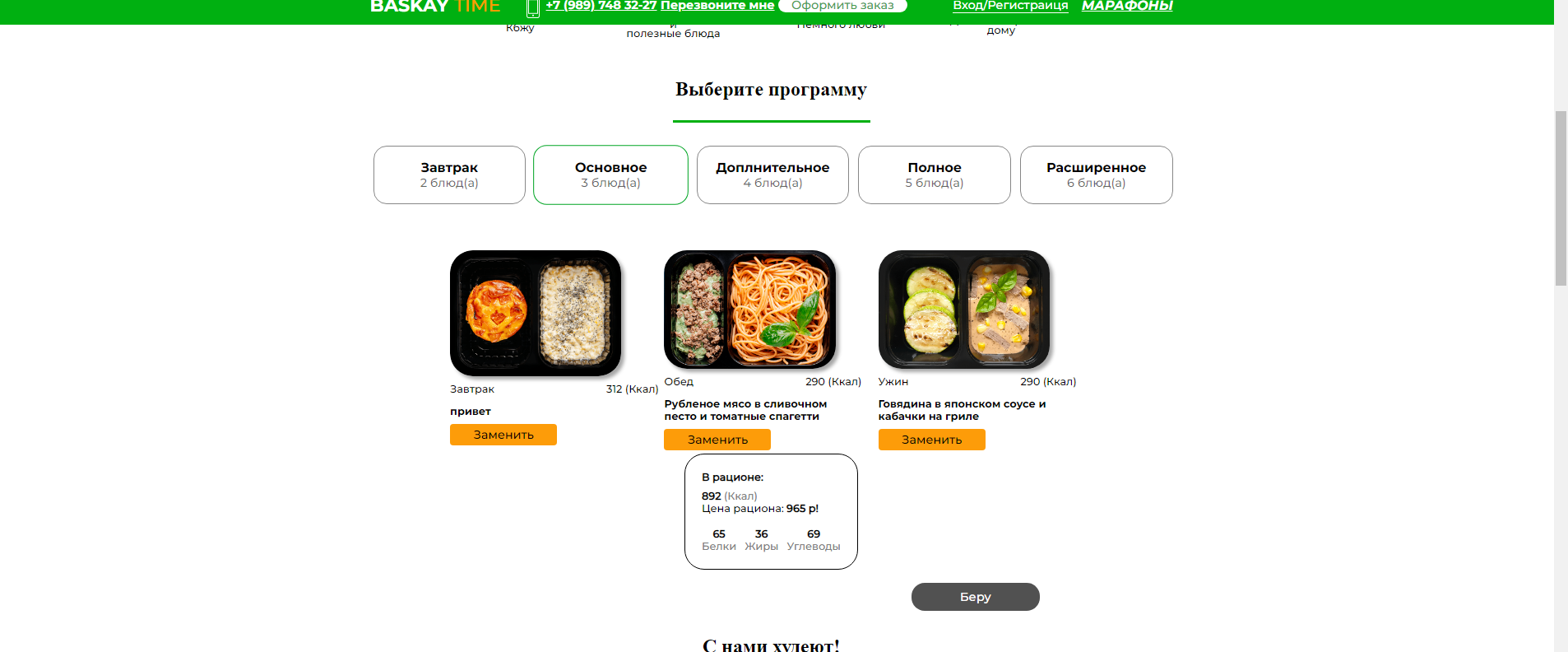


Рисунок 2.3 Выбор программы

Составление заказа зависит от цели пользователя. Хочет ли он создать дефицит калорий и сбросить лишний вес. Или же ему просто нужно экономить время затрачиваемое на приготовление блюд.

Можно выбрать только завтрак который включает в себя два легких набора пищи для начала дня.

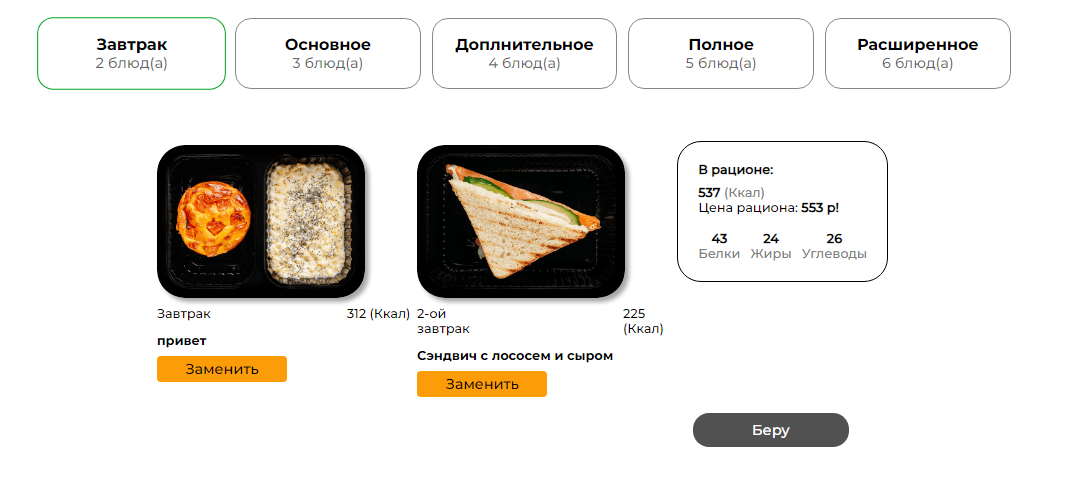


Рисунок 2.4 Программа – “Завтрак”

Или же выбрать программу с рационом на весь день

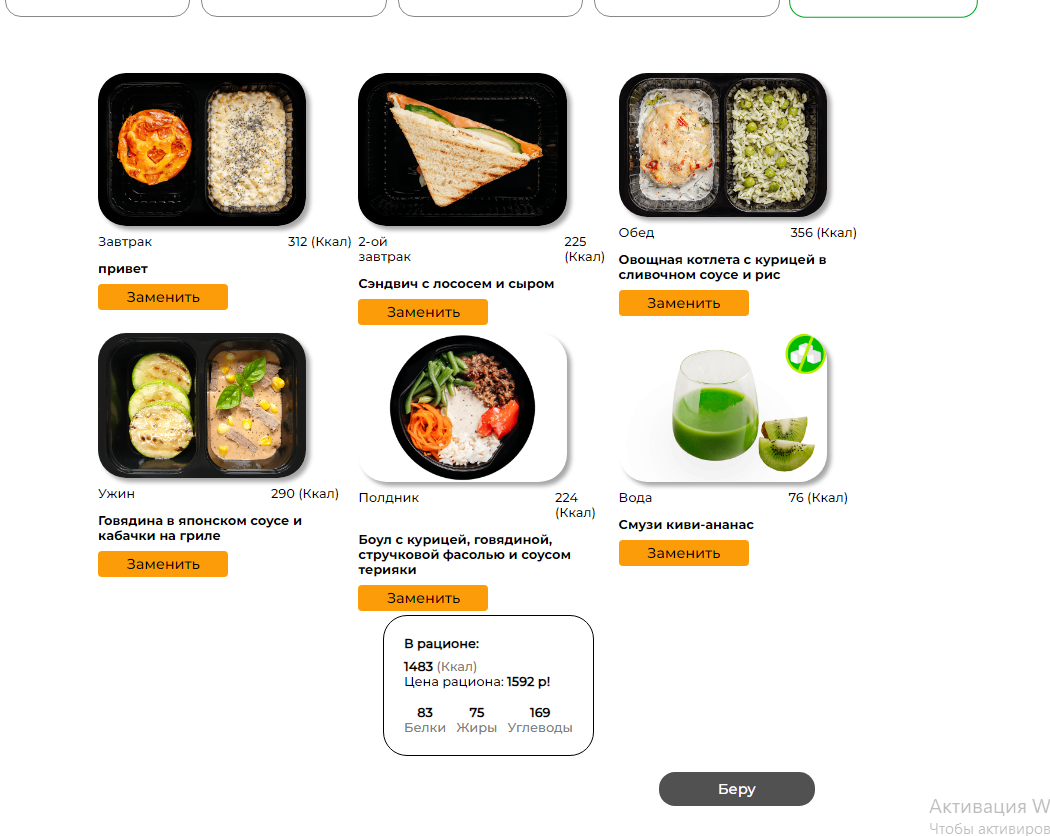


Рисунок 2.4 Программа – “Расширенное”

Умный калькулятор подсчитает все калории и укажет цену.

При нажатии на кнопку заменить открывается окно с заменой выбранного блюда.

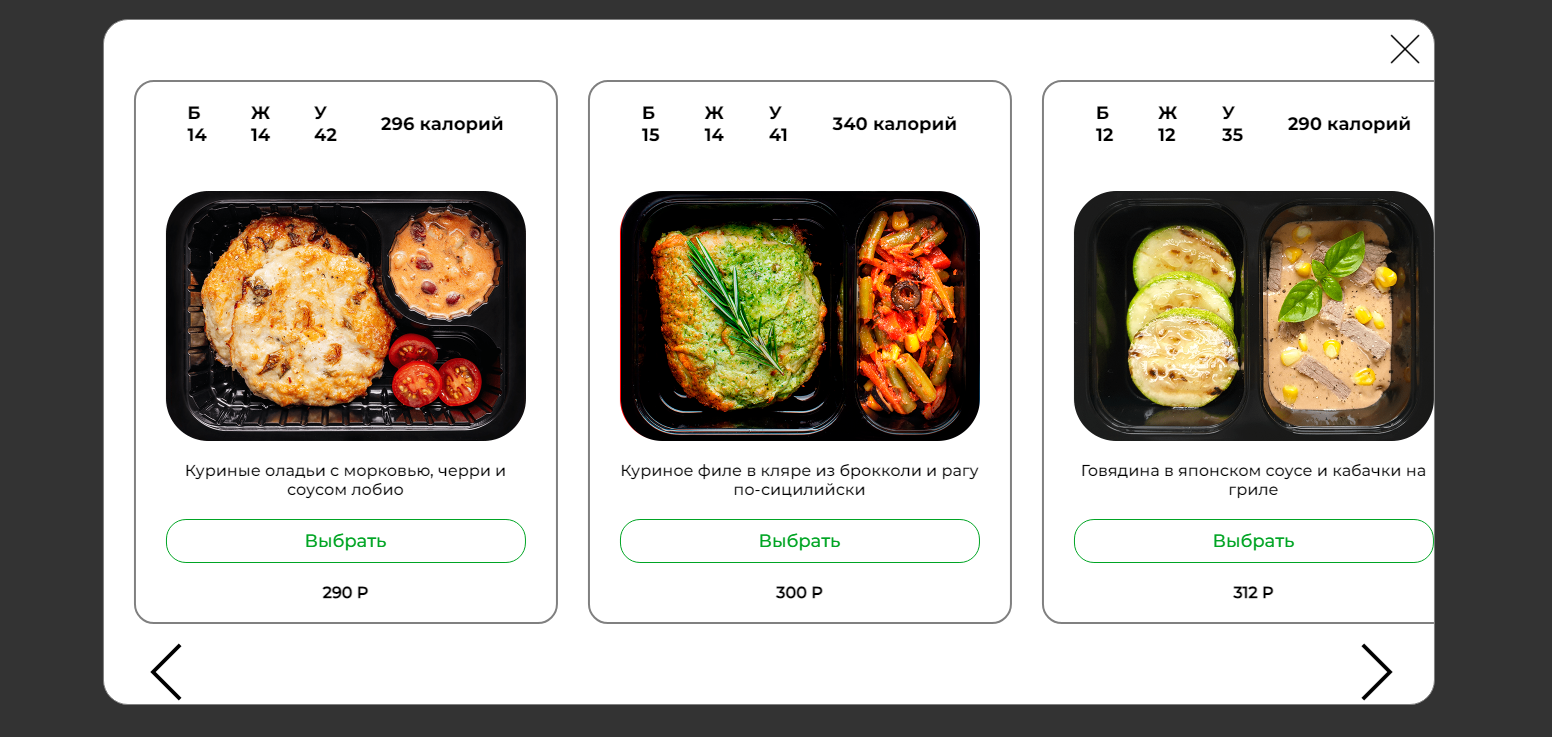


Рисунок 2.5 Замена блюда.

После нажатия на кнопку “Беру” открывается форма для уточнения деталей заказа

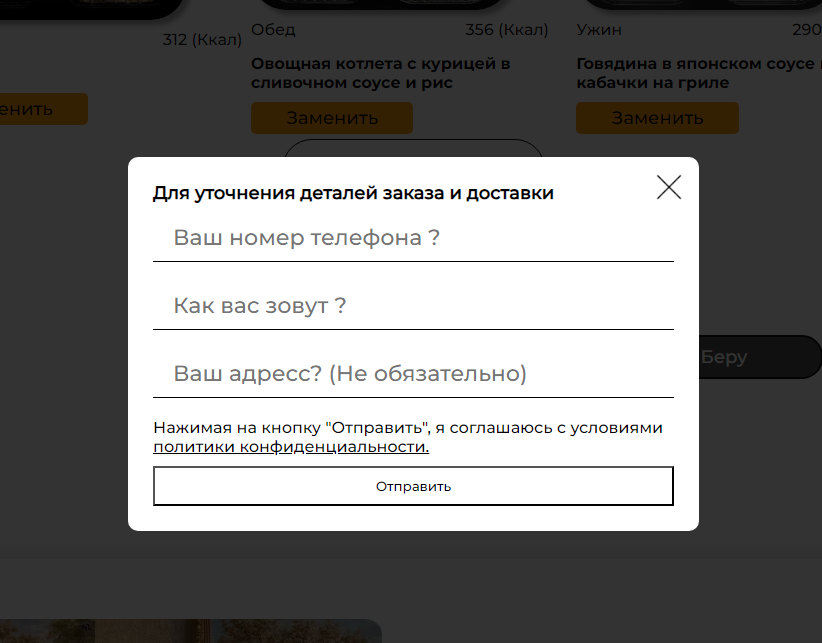


Рисунок 2.6 Форма заказа.

После заполнения формы и нажатии кнопки отравить появляется окно с уведомлением о том что заказ принят.

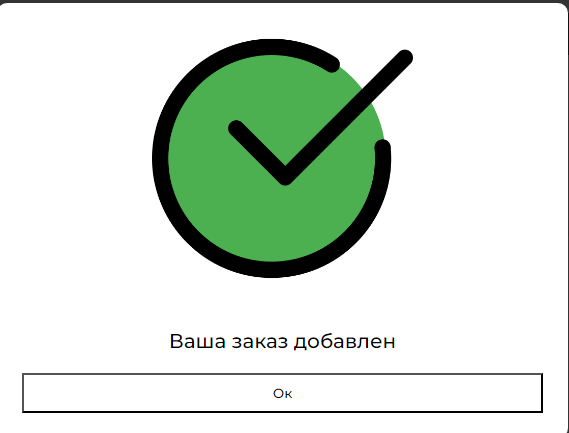
\

Рисунок 2.7 Окно уведомления.

Весь процесс отображения блюд на клиентской части выглядит так

menu() {

if (this.currentCategory === 'Завтрак') {

this.currentArr = [

this.breakfasts[this.randomIndex(this.breakfasts)],

this.secondBreakfasts[this.randomIndex(this.secondBreakfasts)]

]

}

if (this.currentCategory === 'Основное') {

this.currentArr = [

this.breakfasts[this.randomIndex(this.breakfasts)],

this.lunchs[this.randomIndex(this.lunchs)],

this.dinners[this.randomIndex(this.dinners)]

]

}

if (this.currentCategory === 'Доплнительное') {

this.currentArr = [

this.breakfasts[this.randomIndex(this.breakfasts)],

this.secondBreakfasts[this.randomIndex(this.secondBreakfasts)],

this.lunchs[this.randomIndex(this.lunchs)],

this.dinners[this.randomIndex(this.dinners)]

]

}

if (this.currentCategory === 'Полное') {

this.currentArr = [

this.breakfasts[this.randomIndex(this.breakfasts)],

this.secondBreakfasts[this.randomIndex(this.secondBreakfasts)],

this.lunchs[this.randomIndex(this.lunchs)],

this.snacks[this.randomIndex(this.snacks)],

this.dinners[this.randomIndex(this.dinners)]

]

}

if (this.currentCategory === 'Расширенное') {

this.currentArr = [

this.breakfasts[this.randomIndex(this.breakfasts)],

this.secondBreakfasts[this.randomIndex(this.secondBreakfasts)],

this.lunchs[this.randomIndex(this.lunchs)],

this.dinners[this.randomIndex(this.dinners)],

this.snacks[this.randomIndex(this.snacks)],

this.waters[this.randomIndex(this.waters)]

]

}

},

А вот так серверная

<?php

namespace App\Http\Controllers;

use App\Models\Product;

use Illuminate\Http\Request;

class ProductController extends Controller

{

public function all()

{

$products = Product::all();

return $products;

}

public function get(Request $req)

{

$products = Product::where('category\_id', $req->input('id'))->get();

return $products;

}

При нажатии кнопки “Заменить” срабатывает функция updateProduct(), которая на входе принимает объект с данными продукта на котором была нажата кнопка. После чего отрабатывается код замены

updateProduct(product) {

this.updateArr = [];

let index = this.currentArr.findIndex(el => el.category\_id === product.category\_id);

this.currentArr.splice(index, 1, product);

this.currentUpdateProduct = 0;

},ъ

Затем идёт слайдер клиентов с отзывами добившихся хороших результатов вместе с рационами BaskayTime

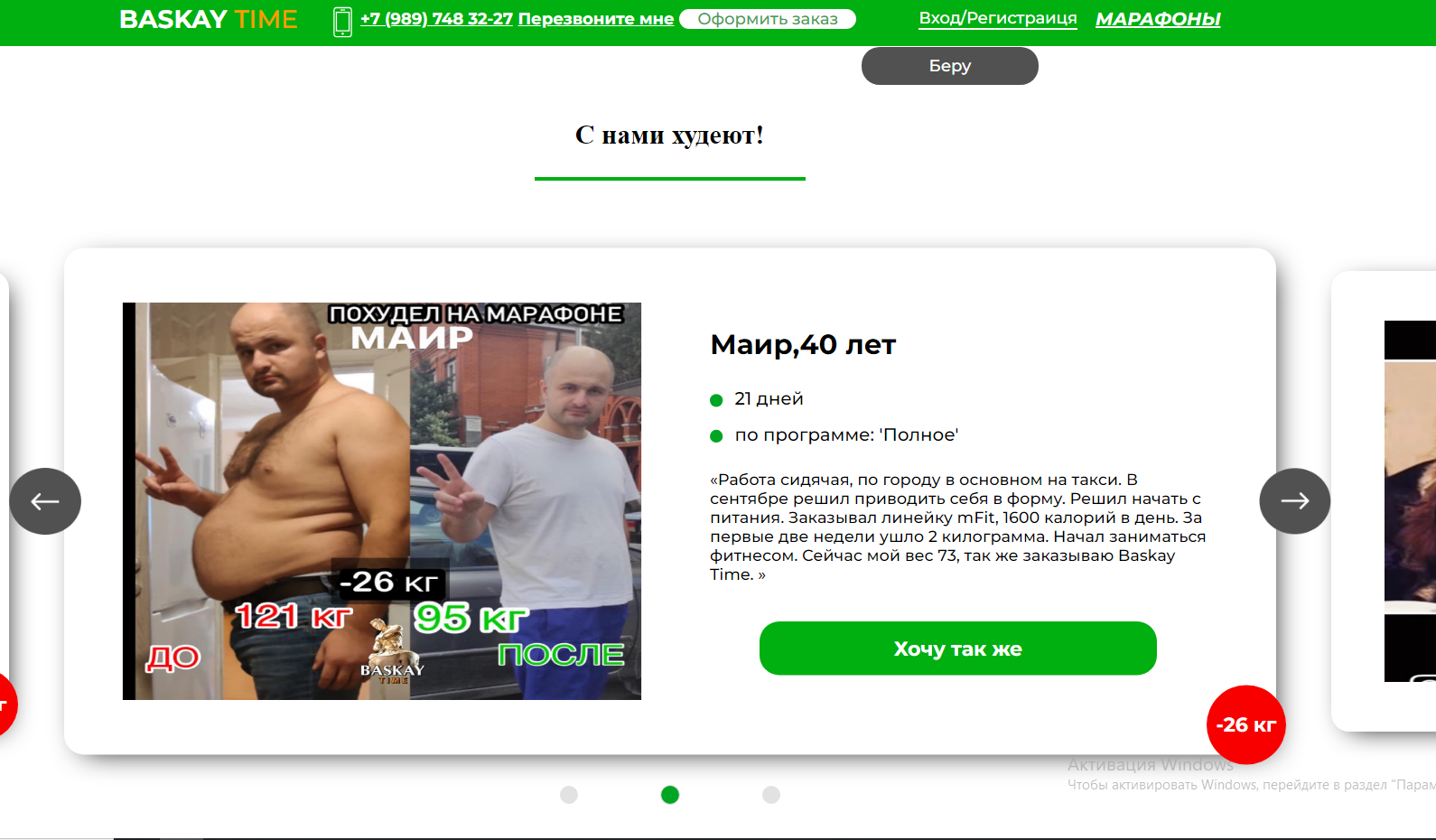


Рисунок 2.8 Слайдер с клиентами.

Затем расположен блок с призывом и перечислением пользы избавления от лишнего веса

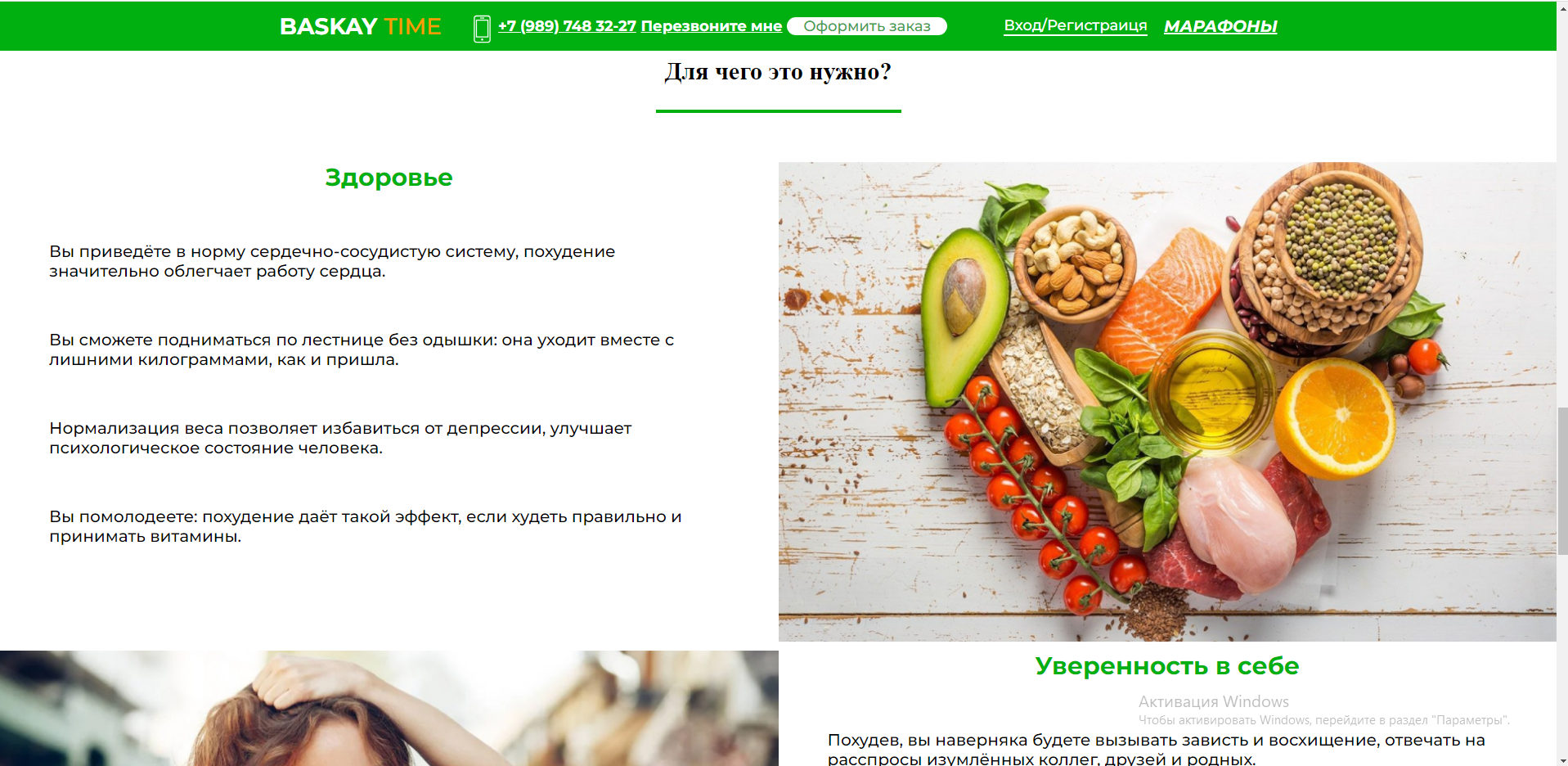


Рисунок 2.9 Выгода.

И в конце раздел с часто задаваемыми вопросами и ответами на них.

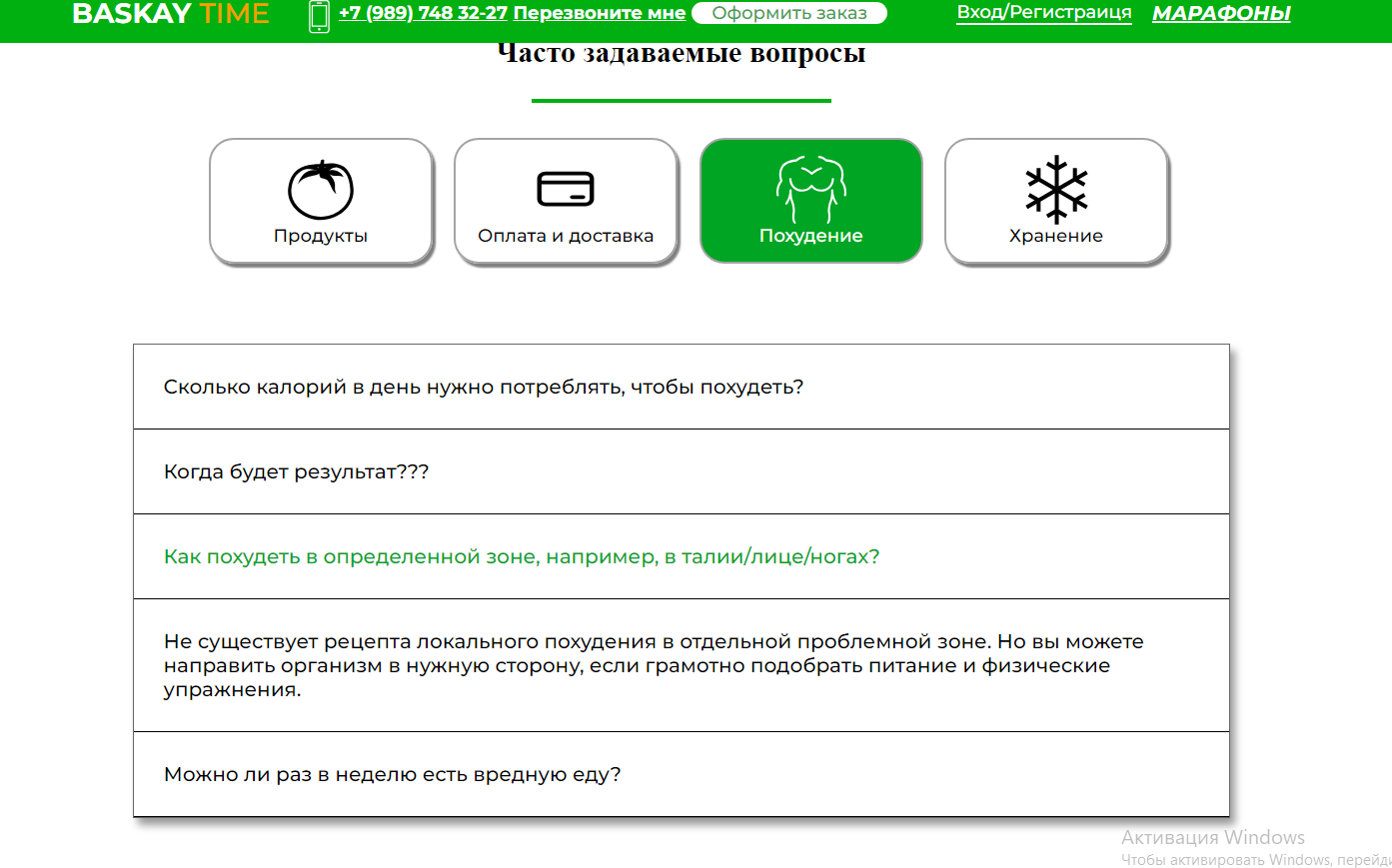


Рисунок 2.9 FAQ.

При нажатии на кнопку Вход/ Регистрация открывается окно входа в аккаунт. Так же есть возможность создать новый.

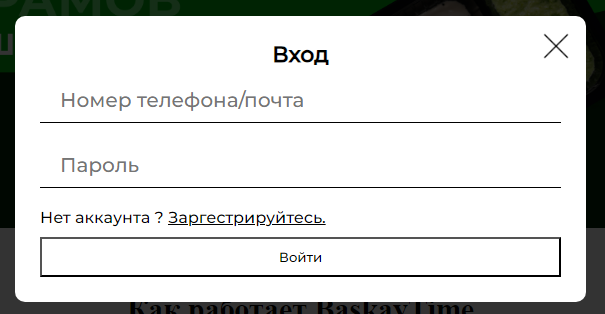


Рисунок 2.10 Вход.

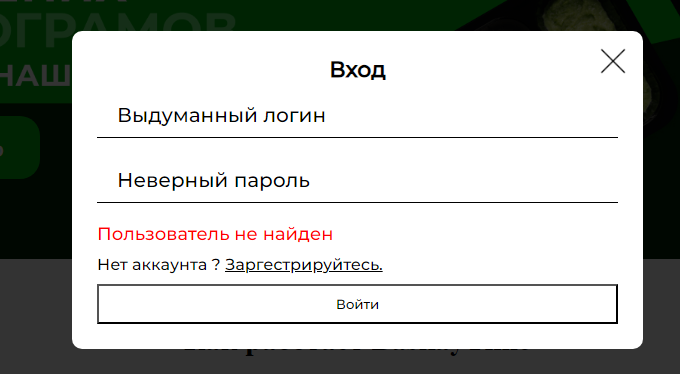


Рисунок 2.11 Ошибка входа.

Валидцаия осуществляется при помощи серверной части

public function login( Request $req ) {

$user = User::where( 'email', $req->input('email') )->where( 'password', $req->input('password') )->first();

if( !$user ) return response()->json([

'message' => 'Пользователь не найден'

], 401);

return $user;

}

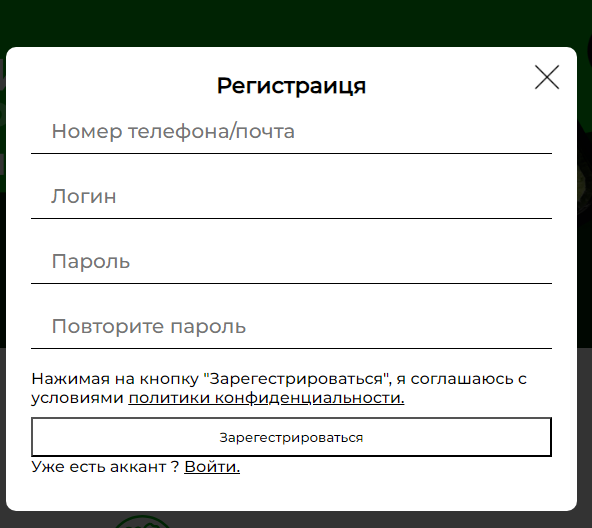


Рисунок 2.12 Регистрация.

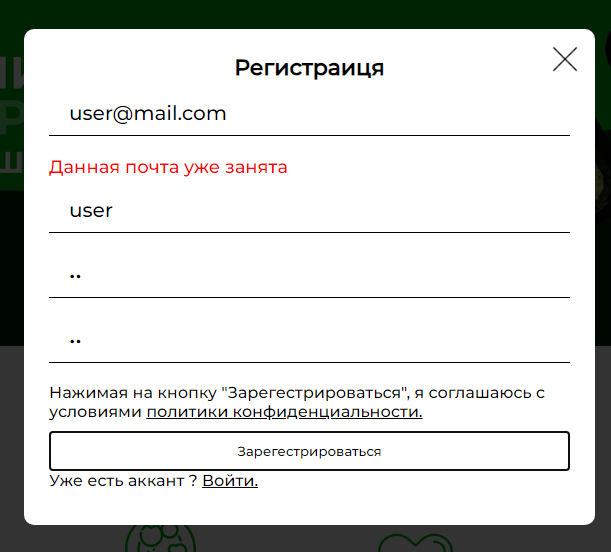


Рисунок 2.13 Почта занята.

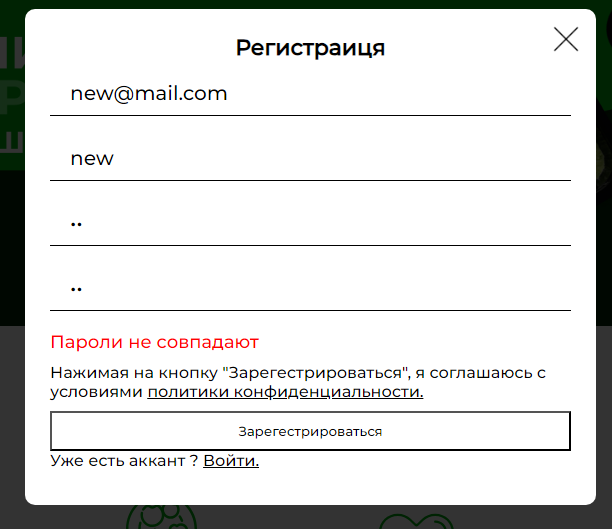


Рисунок 2.14 Пароли не совпадают.

Валидция регистрации через сервер

public function registration( Request $req ) {

$validator = Validator::make( $req->all(), [

'email' => 'unique:users,email',

], [

'email.unique' => 'Данная почта уже занята',

]);

if( $validator->fails() ) return response()->json([

'message' => $validator->errors()

], 401);

$user = User::create( $req->all() );

return $user;

}

}

При нажатии на кнопку “Марафоны” откроется новая страница с уже проведёнными ранее марафонами



Рисунок 2.15 Марафоны.

После нажатия на кнопку подробнее открывается детальная информация о проведённом марафоне.

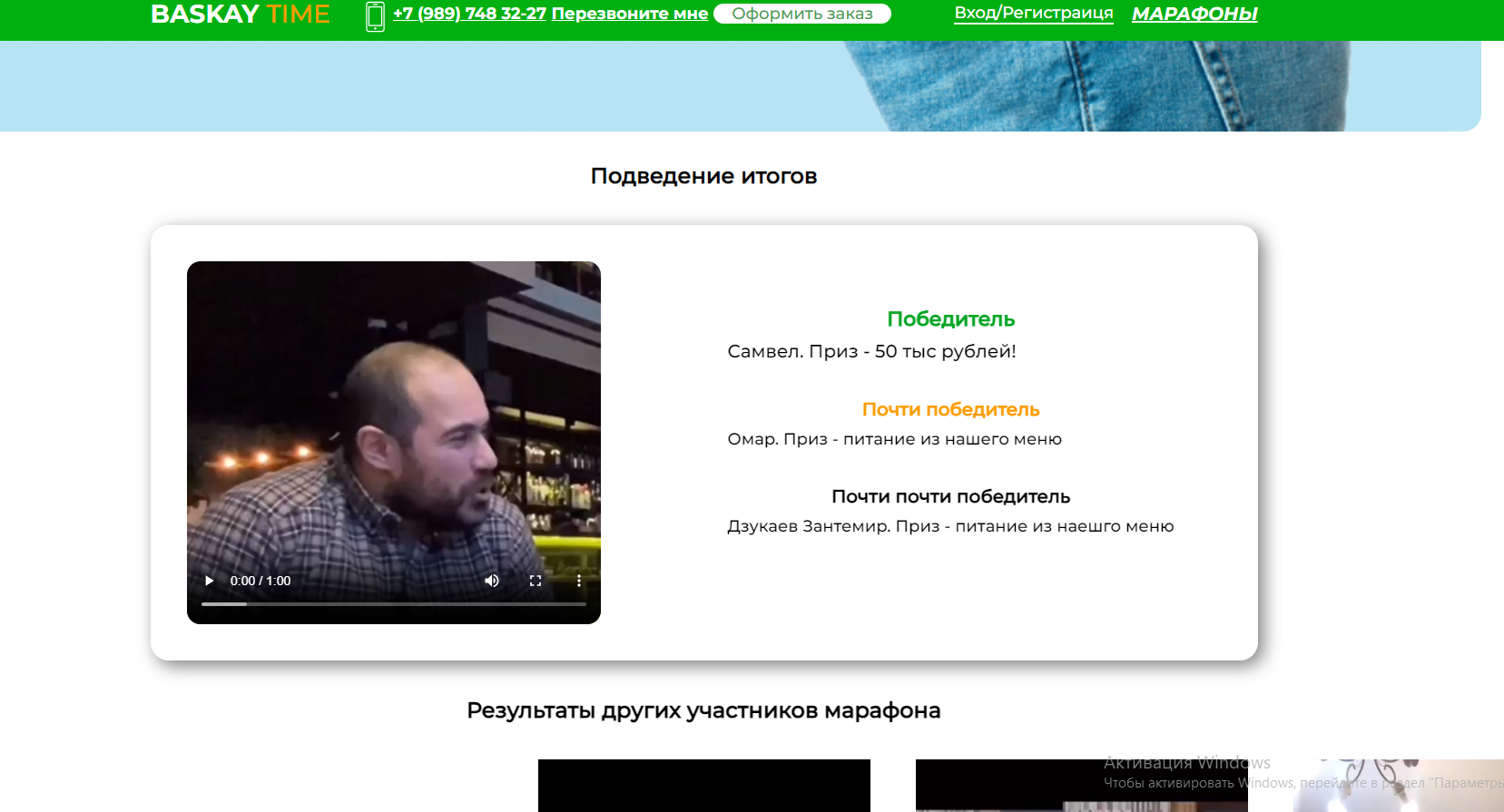


Рисунок 2.16 Детальная марафона.

На этой же странице есть информация о текущем марафоне с возможностью принять участие.

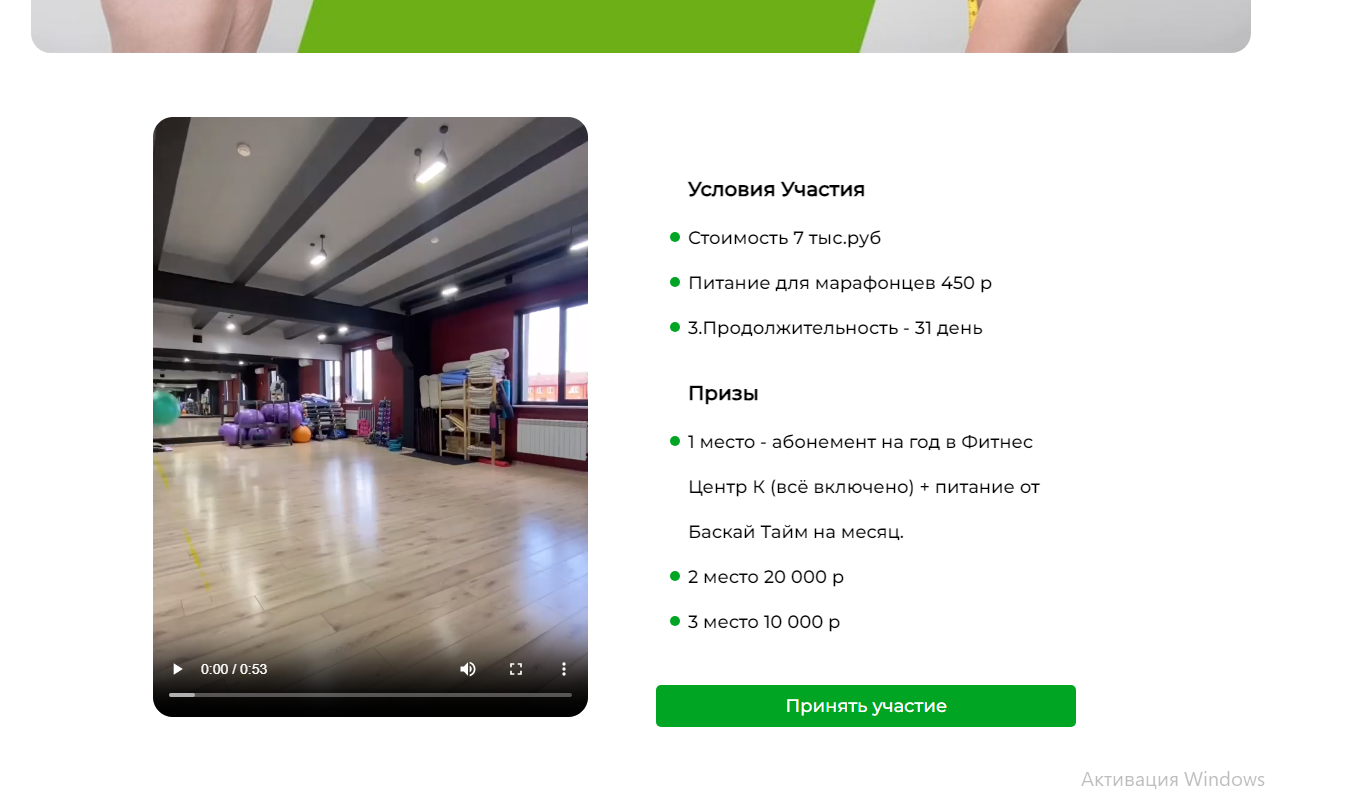


Рисунок 2.17 Текущий марафон.

Для того что бы принять участие нужно нажать на соответствующую кнопку, после чего заполнить несложную форму.

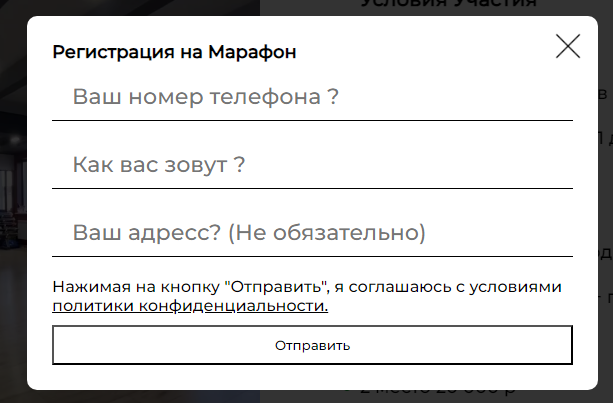


Рисунок 2.17 Регистрация на марафон.

Так же в работе есть панель администратора. Для того что бы войти достаточно в окне Входа в поле почты ввести [admin@mail.com](mailto:admin@mail.com), и в поле пароль ввести admin. После чего в шапке сайта появится админ панель



При нажатие на которую открывается страница с редактированием меню

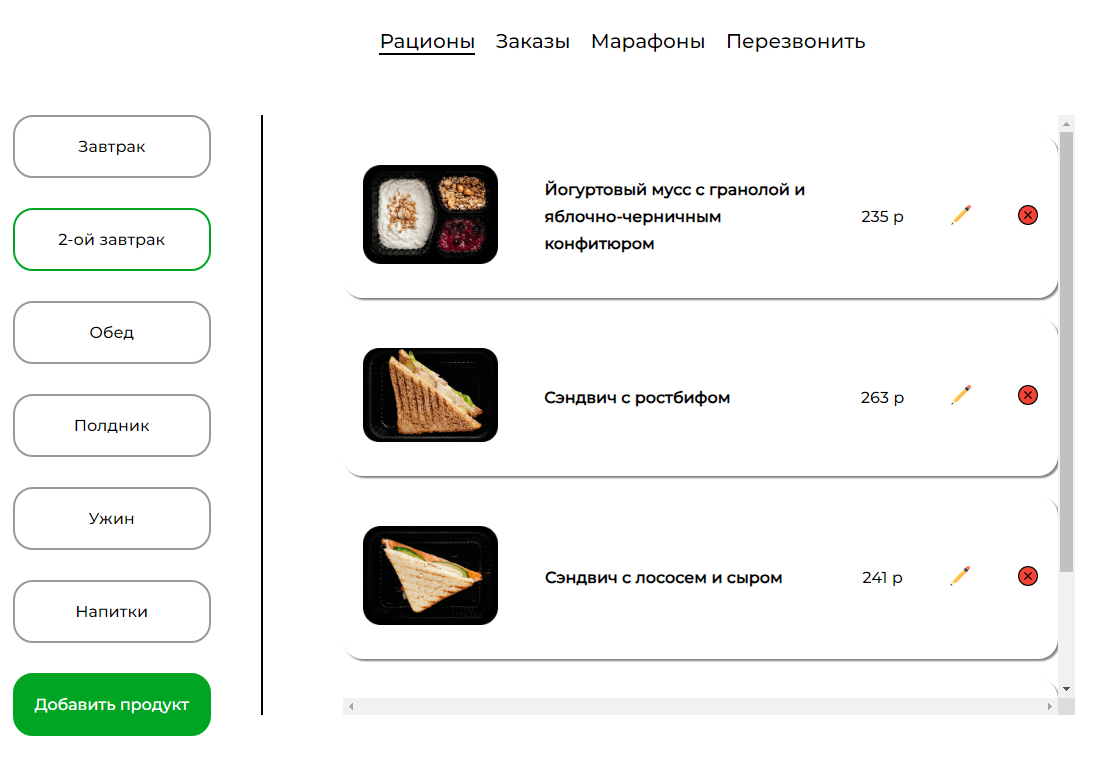


Рисунок 2.18 Редактирование меню.

На этой вкладке можно удалить блюдо, редактировать его или же добавить новое. При нажатии на соответствующие кнопки срабатывает функция, которая делает запрос на сервер, который обрабатывает данные. Так выглядит клиентская часть на примере создание продукта

postProduct() {

post('http://pup/public/api/products/create', this.creatingProduct)

this.creatingProduct = null;

this.render();

this.currentPopUp = 7;

},

А это уже код обработки сервера

public function create(Request $req)

{

Product::create($req->all());

}

По такому же принципу работает редактирование и удаление.

Клиентская часть:

postChanges() {

post('http://pup/public/api/products/update', this.editingProduct);

this.editingProduct = null;

this.render()

this.currentPopUp = 8;

},

removeProduct(product) {

post('http://pup/public/api/products/delete', product);

this.render()

this.currentPopUp = 9;

},

Сервер:

public function update(Request $req)

{

$product = Product::where('id', $req->input('id'))->first();

$product->update($req->all());

}

public function delete(Request $req)

{

$product = Product::where('id', $req->input('id'))->first();

$product->delete();

}

Так же в админ панель отслеживает заявки на марафон, просьбы перезвонить и конечно же заказы, совершённые клиентами.

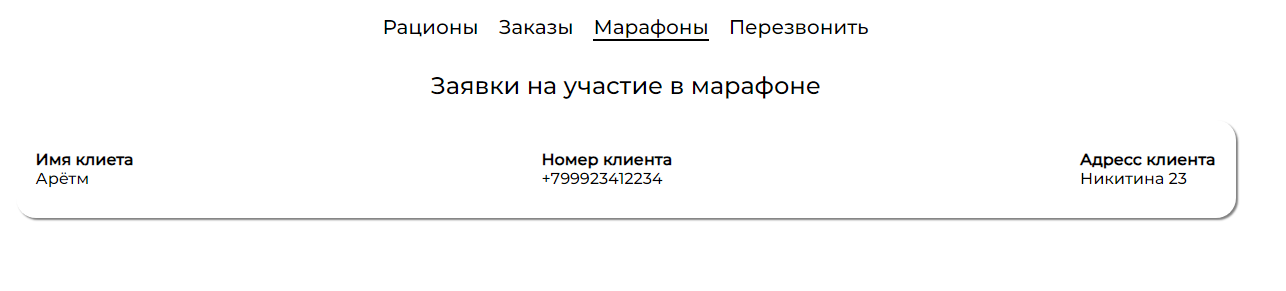


Рисунок 2.19 Панель администратора - марафоны.



Рисунок 2.20 Панель администратора - заказы.

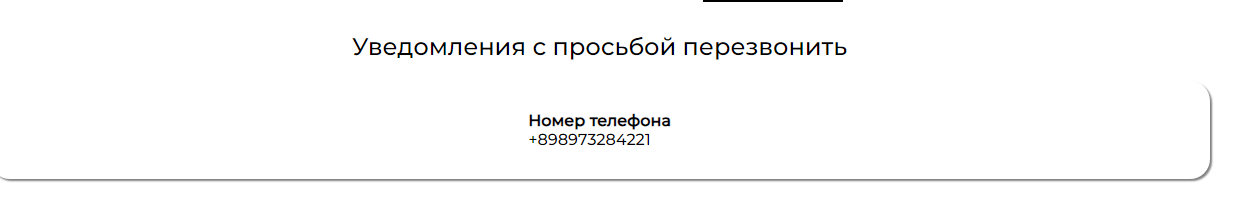


Рисунок 2.21 Панель администратора –перезвонить

## 2.4 Описание процесса отладки программы

Web-сайт был запущен на следующих браузерах: Fire Fox, Safari, Google Chrome, Opera, Yandex Browser, Microsoft Edge. На всех браузерах таблица стилей работает корректно, верстка отображается без проблем. Во время загрузки Web-сайта, все ссылки отрабатываю корректно (без ошибок).

## 2.5 Оценка результатов решения задачи

Был разработан Web-сайт. Данная разработка полностью удовлетворяет поставленной задачи.

При проектировании были учтены все возможные области работы, а также специфика предметной области и области данных.

Программа проста в освоении, так как наследует знакомые пользователю приемы и методы работы, стандартный вид окна и интерфейс.

Результатом тестирования, последующей отладки и корректировки программы является готовый продукт.

Разрабатываемая программа отвечает всем требованиям при условии её реализации на тех технических средствах, которые могут обеспечить эти характеристики.

## ЗАКЛЮЧЕНИЕ

Данная дипломная работа «Сайт правильного питания » разработан на языке JavaScript и фреймворках Vue.js и Laravel.

Разработанный сервис предназначен как для обычных пользователей, которые хотят сбросить лишние килограммы и привести тело в тонус.

Созданный сайт отвечает всем требованиям, предъявляемым к подобного рода программному продукту. В предоставленной разработке реализованы все важные функции, необходимые для работы.

В данной дипломной работе были рассмотрены следующие вопросы:

* создание дизайна сайта в приятном цветовом исполнении;
* внедрение CSS-свойств для более удобного восприятия кода и красивого отображения страниц в браузере;
* создание графических элементов сайта;
* вёрстка дизайна сайта на шаблон.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. JavaScript. Подробное руководство - <https://g.co/kgs/k6QA2Q>
2. JavaScript: сильные стороны - <https://g.co/kgs/AHV75L>
3. Learning Node: Moving to the Server-Side - https://g.co/kgs/TkRgLV
4. Node.js в действии Авторы: Кантелон Майк, Хартер Марк, Райлих Натан, Головайчук TJ.
5. Node.js. Путеводитель по технологии - https://g.co/kgs/FuEUbx
6. Документация Node.js - https://nodejs.org/en/docs/
7. Разработка веб-приложений с помощью Node.js, MongoDB и Angular. Исчерпывающее руководство по использованию стека MEAN | Дейли Брэд, Дейли Брендан
8. Секреты JavaScript ниндзя - <https://g.co/kgs/uJx85n>
9. Джон Дакетт. “HTML и CSS. Разработка и дизайн веб-сайтов” Read more
10. Дэвид Скляр "Изучаем PHP 7 .«Руководство по созданию интерактивных веб-сайтов» 2017.
11. Vue JS Библиотека JavaScript для построения пользовательских интерфейсов, Документация - <https://vuejs.org/>
12. Д.Н. Колисниченко - PHP и MySQL. Разработка Web-приложений 2013

## ПРИЛОЖЕНИЯ

**Приложение1**

<?php

use App\Http\Controllers\CategoryController;

use App\Http\Controllers\ProductController;

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\UserController;

Route::post('/registration', [UserController::class, 'registration']);

Route::post('/login', [UserController::class, 'login']);

Route::post('/products/all', [ProductController::class, 'all']);

Route::post('/products/get', [ProductController::class, 'get']);

Route::post('/products/update', [ProductController::class, 'update']);

Route::post('/products/delete', [ProductController::class, 'delete']);

Route::post('/products/create', [ProductController::class, 'create']);

Route::post('/categories/all', [CategoryController::class, 'all'])

<?php

namespace App\Http\Controllers;

use App\Models\Category;

use Illuminate\Http\Request;

class CategoryController extends Controller

{

public function all()

{

return Category::all();

}

}

<?php

namespace App\Http\Controllers;

use App\Models\Product;

use Illuminate\Http\Request;

class ProductController extends Controller

{

public function all()

{

$products = Product::all();

return $products;

}

public function get(Request $req)

{

$products = Product::where('category\_id', $req->input('id'))->get();

return $products;

}

public function update(Request $req)

{

$product = Product::where('id', $req->input('id'))->first();

$product->update($req->all());

}

public function delete(Request $req)

{

$product = Product::where('id', $req->input('id'))->first();

$product->delete();

}

public function create(Request $req)

{

Product::create($req->all());

}

}

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

// Models

use App\Models\User;

use Illuminate\Support\Facades\Validator;

class UserController extends Controller

{

public function login( Request $req ) {

$user = User::where( 'email', $req->input('email') )->where( 'password', $req->input('password') )->first();

if( !$user ) return response()->json([

'message' => 'Пользователь не найден'

], 401);

return $user;

}

public function registration( Request $req ) {

$validator = Validator::make( $req->all(), [

'email' => 'unique:users,email',

], [

'email.unique' => 'Данная почта уже занята',

]);

if( $validator->fails() ) return response()->json([

'message' => $validator->errors()

], 401);

$user = User::create( $req->all() );

return $user;

}

}

**Приложение2**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="stylesheet" href="assets/style.css">

<title>Document</title>

</head>

<body>

<div class="wrapper" id="app">

<header :class="{'open' : burgerMenu===true}" class="header">

<div class="heder-content container">

<button v-if="!isAuth || user.role==='user'" @click="burgerMenu = !burgerMenu" class="burger\_\_menu-btn">

<span></span><span></span><span></span>

</button>

<a @click="currentPage='home'" href="#" class="header-content\_\_logo">Baskay <span>Time</span> </a>

<ul v-if="!isAuth || user.role==='user'" class="header-content\_\_menu">

<a class="header-content\_\_menu-tel" href="tel:+79897483227">+7 (989) 748 32-27</a>

<a @click="currentPopUp=1" href="#">Перезвоните мне</a>

<a href="#menu"><button>Оформить заказ</button></a>

</ul>

<span v-if="user">{{user.email}}</span>

<div>

<a v-if="!isAuth" @click="currentPopUp = 3" class="haeder\_\_autorization"

href="#">Вход/Регистраиця</a>

<a v-if="isAdmin" class="haeder\_\_admin-panel" @click="currentPage='admin'" href="#">Панель

администратора</a>

<a v-if="isAuth" class="haeder\_\_admin-panel" @click="currentPage='home';user= null" href="#">

Выход</a>

<a @click="currentPage='maraphon'" href="#" class="header-content\_\_link">марафоны</a>

</div>

</div>

</header>

<main class="main">

<div v-if="currentPage==='confid'" class="confid container">

1. Обрабатываемые данные <br>

1.1. Мы не осуществляем сбор ваших персональных данных с использованием Сайта<br>

1.2. Все данные, собираемые на Сайте, предоставляются и принимаются в обезличенной форме (далее –

«Обезличенные данные»).<br>

1.3. Обезличенные данные включают следующие сведения, которые не позволяют вас идентифицировать:<br>

1.3.1. Информацию, которую вы предоставляете о себе самостоятельно с использованием онлайн-форм и

программных модулей Сайта, включая имя и номер телефона и/или адрес электронной почты.<br>

1.3.2. Данные, которые передаются в обезличенном виде в автоматическом режиме в зависимости от настроек

используемого вами программного обеспечения.<br>

1.4. Администрация вправе устанавливать требования к составу Обезличенных данных Пользователя, которые

собираются использованием Сайта.<br>

1.5. Если определенная информация не помечена как обязательная, ее предоставление или раскрытие

осуществляется Пользователем на свое усмотрение. Одновременно вы даете информированное согласие на

доступ неограниченного круга лиц к таким данным. Указанные данные становится общедоступными с момента

предоставления и/или раскрытия в иной форме.<br>

1.6. Администрация не осуществляет проверку достоверности предоставляемых данных и наличия у

Пользователя необходимого согласия на их обработку в соответствии с настоящей Политикой, полагая, что

Пользователь действует добросовестно, осмотрительно и прилагает все необходимые усилия к поддержанию

такой информации в актуальном состоянии и получению всех необходимых согласий на ее использование.<br>

1.7. Вы осознаете и принимаете возможность использования на Сайте программного обеспечения третьих лиц,

в результате чего такие лица могут получать и передавать указанные в п.1.3 данные в обезличенном виде.

Пример! К указанному программному обеспечению третьих лиц относятся системы сбора статистики посещений

Google Analytics и Яндекс.Метрика.<br>

1.8. Состав и условия сбора обезличенных данных с использованием программного обеспечения третьих

лицопределяются непосредственно их правообладателями и могут включать

данные браузера (тип, версия, cookie); данные устройства и место его положения; данные операционной

системы (тип, версия, разрешение экрана); данные запроса (время, источник перехода, IP-адрес).<br>

1.9. Администрация не несет ответственность за порядок использования Обезличенных данных Пользователя

третьими лицами.<br>

2. Цели обработки данных

2.1. Администрация использует данные в следующих целях:

2.1.1. Обработка поступающих запросов и связи с Пользователем;

3. Требования к защите данных

2.1.2. Информационное обслуживание, включая рассылку рекламно-информационных материалов;

2.1.3. Проведение маркетинговых, статистических и иных исследований;

2.1.4. Таргетирование рекламных материалов на Сайте.

3.1. Администрация осуществляет хранение данных и обеспечивает их охрану от несанкционированного доступа

и распространения в соответствии с внутренними правилами и регламентами.

3.2. В отношении полученных данных сохраняется конфиденциальность, за исключением случаев, когда они

сделаны Пользователем общедоступными, а также когда используемые на Сайте технологии и программное

обеспечение третьих лиц либо настройки используемого Пользователем программного обеспечения

предусматривают открытый обмен с данными лицами и/или иными участниками и пользователями сети

Интернет.<br>

3.3. В целях повышения качества работы Администрация вправе хранить лог-файлы о действиях, совершенных

Пользователем в рамках использования Сайта в течение 1 (Одного) года.<br>

4. Передача данных<br>

4.1. Администрация вправе передать данные третьим лицам в следующих случаях:

Пользователь выразил свое согласие на такие действия, включая случаи применения Пользователем настроек

используемого программного обеспечения, не ограничивающих предоставление определенной информации;

Передача необходима в рамках использования Пользователем функциональных возможностей Сайта;

Передача требуется в соответствии с целями обработки данных;<br>

В связи с передачей Сайта во владение, пользование или собственность такого третьего лица;

По запросу суда или иного уполномоченного государственного органа в рамках установленной

законодательством процедуры;<br>

Для защиты прав и законных интересов Администрации в связи с допущенными Пользователем нарушениями.

5. Изменение Политики конфиденциальности<br>

5.1. Настоящая Политика может быть изменена или прекращена Администрацией в одностороннем порядке без

предварительного уведомления Пользователя. Новая редакция Политики вступает в силу с момента ее

размещения на Сайте, если иное не предусмотрено новой редакцией Политики.

</div>

<div v-if="currentPage==='admin'" class="admin-panel container">

<div class="admin-navigation">

<span :class="{'active': adminPage==='menu'}" @click="adminPage = 'menu'">Рационы</span>

<span :class="{'active': adminPage==='orders'}" @click="adminPage = 'orders'">Заказы</span>

<span :class="{'active': adminPage==='adminMar'}" @click="adminPage = 'adminMar'">Марафоны</span>

<span :class="{'active': adminPage==='perez'}" @click="adminPage = 'perez'">Перезвонить</span>

</div>

<div v-if="adminPage ==='orders'" class="admin-orders container">

<div v-if="orders.length === 0" class="empty\_\_orders">

<span>Здесь пока нет заказов</span>

</div>

<div v-else class="admin-orders\_\_container">

<h2 style="text-align: center; font-weight: normal; margin-top: 30px;">Заказы</h2>

<div v-for="order in orders" :key=" order.id" class="admin-menu\_\_product">

<span> <b>Имя клиета</b> <br>{{order.client}}</span>

<span><b>Номер клиента</b> <br> {{order.phone}}</span>

<span><b>Адресс клиента</b> <br> {{order.address}}</span>

<div class="admin-product\_\_img2-container">

<div v-for="product in order.product" :key=" product.id" class="admin-product\_\_img2">

<img :title="product.name" :src="'./assets/images/'+ product.img " alt="">

</div>

</div>

</div>

</div>

</div>

<div v-if="adminPage ==='perez'" class="admin-orders container">

<div v-if="adminNumbers.length === 0" class="empty\_\_orders">

<span>Здесь пока нет заявок</span>

</div>

<div v-else class="admin-orders\_\_container">

<h2 style="text-align: center; font-weight: normal; margin-top: 30px;">Уведомления с просьбой

перезвонить </h2>

<div style="justify-content: center;" v-for="num in adminNumbers" :key=" num.id"

class="admin-menu\_\_product">

<span> <b>Номер телефона</b> <br>{{num.num}}</span>

</div>

</div>

</div>

<div v-if="adminPage ==='adminMar'" class="admin-orders container">

<div v-if="adminMaraphons.length === 0" class="empty\_\_orders">

<span>Здесь пока нет участиников</span>

</div>

<div v-else class="admin-orders\_\_container">

<h2 style="text-align: center; font-weight: normal; margin-top: 30px;">Заявки на участие в

марафоне</h2>

<div v-for="marr in adminMaraphons" :key=" marr.id" class="admin-menu\_\_product">

<span> <b>Имя клиета</b> <br>{{marr.client}}</span>

<span><b>Номер клиента</b> <br> {{marr.phone}}</span>

<span><b>Адресс клиента</b> <br> {{marr.address}}</span>

</div>

</div>

</div>

<div v-if="adminPage==='menu'" class=" admin-menu">

<div class="admin-menu\_\_categories">

<span v-for="category in adminCategories" :key="category.id"

:class="{'active': currentAdminCategory === category.name }"

@click="currentAdminCategory=category.name"

class="admin-menu\_\_category">{{category.name}}</span>

<span @click="createProduct" class="admin-menu\_\_category admin-menu\_\_category--add">Добавить

продукт</span>

</div>

<div class="admin-menu\_\_products">

<div v-for="product in adminMenu" :key="product.id" class="admin-menu\_\_product">

<div class="admin-product\_\_img">

<img :src="'./assets/images/'+ product.img " alt="">

</div>

<span class="admin-product\_\_name">

{{product.name}}</span>

<span class="admin-product\_\_price"> {{product.price}} р</span>

<div title="Редактировать" class="admin-product\_\_update"><img @click="editProduct(product)"

src="./assets/images/edit.png "></div>

<div title="Удалить" class="admin-product\_\_delete"><img @click="removeProduct(product)"

src="./assets/images/remove.png">

</div>

</div>

</div>

</div>

</div>

<div v-if=" currentPage==='maraphon'" class=" maraphons container">

<div v-if="currentMaraphon" class="current-marphon">

<h2>Текущий марафон</h2>

<div v-for="mar in maraphons" :key="mar.id" class="maraphon\_\_preview">

<img v-if="mar.current===true" :src="'./assets/images/' + mar.vrewiew" alt="">

</div>

<div class="maraphon\_\_conditions">

<div class="maraphon\_\_conditions-video">

<video controls :src="'./assets/videos/' + currentMaraphon.video "></video>

</div>

<div class="maraphon\_\_conditions-rules">

<div class="maraphon\_\_conditions-container">

<span>Условия Участия</span>

<p v-for="el in splitCondition" :key="el">

{{el}}

</p>

</div>

<div class="maraphon\_\_conditions-container">

<span>Призы</span>

<p v-for="el in splitGift" :key="el">

{{el}}

</p>

</div>

<button @click="currentPopUp=10"

class="maraphon\_\_preview-btn maraphon\_\_preview-btn--first ">Принять

участие</button>

</div>

<button @click="currentPopUp=2"

class="maraphon\_\_preview-btn maraphon\_\_preview-btn--second">Принять

участие</button>

</div>

</div>

<div class="other\_\_maraphons">

<h2>Предидущие марафоны</h2>

<div v-for="el in otherMaraphons" class="maraphon\_\_preview">

<img :src="'./assets/images/' + el.vrewiew" alt="">

<button @click="goDetail(el)" class=" maraphon\_\_preview--orange">Подробнее</button>

</div>

</div>

</div>

<div v-if="currentPage==='home'" class="home">

<section class="intro container">

<img class="intro-img" src="./assets/images/Intro.jpg" alt="">

<a href="#menu"><button class="intro-btn"> ВЫБРАТЬ</button></a>

</section>

<section id="menu" class=" profits container">

<h2 class="section-caption">Как работает BaskayTime</h2>

<div class="profits-line">

<div class="profits-line\_\_item">

<div class="profits-line\_\_item-img">

<img src="./assets/images/profit-calc.svg" alt="">

</div>

<span>Расчитаем <br>

Кбжу</span>

</div>

<div class="profits-line\_\_item">

<div class="profits-line\_\_item-img">

<img src="./assets/images/profit-vegetabble.svg" alt="">

</div>

<span>Приготовим вкусыне <br>

и <br>

полезные блюда</span>

</div>

<div class="profits-line\_\_item">

<div class="profits-line\_\_item-img">

<img src="./assets/images/profit-heart.svg" alt="">

</div>

<span>Немного любви </span>

</div>

<div class="profits-line\_\_item">

<div class="profits-line\_\_item-img">

<img src="./assets/images/profit-track.svg" alt="">

</div>

<span>Доставим прямо к <br>

дому

</span>

</div>

</div>

</section>

<section class="programs container">

<h2 class=" section-caption">Выберите программу</h2>

<div class="categories\_\_line">

<div :class="{active :currentCategory == category.name }"

@click="currentCategory = category.name, menu" v-for="category in categories"

:key="category.id" class="category">

<span class="category-name">{{category.name}}</span> <br>

<span class="category-dishes">{{category.dishes}} блюд(а)</span>

</div>

</div>

<div class="current-program">

<div class="product-line">

<product v-for="product in currentArr" :key="product" :product="product"

:array="categoriesProducts" @editprod="changeProduct" v-if="currentArr.length">

</product>

<div v-if="currentArr.length" v-for="element in totalCalories" :key="element.id"

class="total-calories">

<h2>В рационе: </h2>

<span> <b>{{element.calories}}</b> (Ккал)</span>

<p>Цена рациона: <b>{{totalSum}} р!</b> </p>

<div class="total-elements">

<span> <b>{{element.proteins}}</b> <br> Белки</span>

<span> <b>{{element.fats}}</b> <br> Жиры</span>

<span> <b>{{element.carbohdrate}}</b> <br> Углеводы</span>

</div>

</div>

</div>

<button v-if="currentArr.length" @click="currentPopUp = 2" class="current-program\_\_button">

Беру</button>

<div v-if="updateArr!=0" class="update-product\_\_line-container">

<div class="update-product\_\_line">

<img @click="updateArr=0;currentUpdateProduct=0" src="./assets/images/krest.png" alt=""

class="update-product\_\_krestik">

<img @click="prevProduct" src="./assets/images/left\_arr.png" alt=""

class="update-product\_\_arrow update-product\_\_arrow--left">

<img @click="nextProduct" src="./assets/images/left\_arr.png" alt=""

class="update-product\_\_arrow update-product\_\_arrow--right">

<div :style="{'margin-left': '-' + (currentUpdateProduct \* 106.5 + '%') }"

class="update-product\_\_line-slider">

<div v-for="product in updateArr" :key="product.id" class="update-product">

<div class="product-elements">

<span>Б <br>

{{product.proteins}}

</span>

<span>Ж <br>

{{product.fats}}

</span>

<span>У <br>

{{product.carbohdrate}}

</span>

<span class="update-product-calories">{{product.calories}}

калорий</span>

</div>

<img class="update-product-img" :src="'./assets/images/' + product.img" alt="">

<span class="update-product-name">{{product.name}}</span>

<button @click="updateProduct(product)"

class="update-product-btn">Выбрать</button> <span

style="font-weight: bold;">{{product. price}}

Р</span>

</div>

</div>

</div>

</div>

</div>

</section>

<section class="result container">

<h2 class="section-caption">C нами худеют!</h2>

<div class="slider-wrapper">

<div :style="{'margin-left': '-' + (currentSlide \* 110 + '%') }" class="slider">

<div :class="{active : currentSlide == client.id }" v-for="client in clients"

:key="client.id" class="slide">

<div class="slide-img\_\_contaienr">

<img :src="'./assets/images/' + client.img" alt="" class="slide-img">

</div>

<div class="slide-content">

<span class="slide-content\_\_name">{{client.name}},{{client.howOld}} лет</span>

<ul class="slide-content\_\_menu">

<li class="slide-content\_\_list">{{client.programmCount}} дней</li>

<li class="slide-content\_\_list"> по программе: {{client.programmName}}</li>

</ul>

<span class="slide-content\_\_comment">

{{client.comment}}

</span>

<a href="#menu"><button class="slide-content\_\_button"> Хочу так же</button></a>

<div class="slide-content\_\_result">

-{{client.result}} кг

</div>

</div>

</div>

</div>

<img :class="{disable : currentSlide == 0}" @click="prevSlide()" src="./assets/images/Arrow.png"

alt="" class="slider-btn prev-btn">

<img :class="{disable : clients.length - 1 == currentSlide }" @click="nextSlide()"

src="./assets/images/Arrow1.png" alt="" class="slider-btn next-btn">

<div class="pagination-line">

<span v-for="pag in clients" :key="pag.id" @click="currentSlide = pag.id"

:class="{active : pag.id == currentSlide}" class="pagination"></span>

</div>

</div>

</section>

<section class="advice">

<h2 class="section-caption">Для чего это нужно?</h2>

<div class="advice-items">

<div class="advice-item">

<div class="advice-item\_\_content">

<h3 class="advice-item\_\_caption">Здоровье</h3>

<span>Вы приведёте в норму сердечно-сосудистую систему,

похудение значительно облегчает работу сердца.</span>

<span>Вы сможете подниматься по лестнице без одышки: она уходит вместе с лишними

килограммами,

как и пришла.</span>

<span>Нормализация веса позволяет избавиться от депрессии, улучшает психологическое

состояние

человека.</span>

<span>Вы помолодеете: похудение даёт такой эффект, если худеть правильно и принимать

витамины.</span>

</div>

<div class="advice-item\_\_img-container ">

<img src="./assets/images/bigheart.jpg" alt="" class="advice-item\_\_img">

</div>

</div>

<div class="advice-item">

<div class="advice-item\_\_img-container advice-item\_\_img-container--order">

<img src="./assets/images/girl.jpg" alt="" class="advice-item\_\_img">

</div>

<div class="advice-item\_\_content">

<h3 class="advice-item\_\_caption">Уверенность в себе</h3>

<span>Похудев, вы наверняка будете вызывать зависть и восхищение, отвечать на

расспросы

изумлённых коллег, друзей и родных.</span>

<span>Вы покажете пример другим, и, возможно, этим измените чью-то жизнь к

лучшему.</span>

<span>Похудение облегчит процесс ходьбы: вы станете легче, без усилий будете

проходить

дальние дистанции.</span>

<span>Похудение позволяет поднять самооценку, вы станете больше себя ценить, что

повлияет на

всю вашу жизнь. Для многих похудение — это начало перемен в карьере, в личной

жизни.</span>

</div>

</div>

</div>

</section>

<section class="faq container">

<h2 class="section-caption">Часто задаваемые вопросы</h2>

<div class="faq-button-line">

<div :class="{'active': currentFaq == tem.id}" v-for="tem in tems" :key="tem.id"

@click="currentFaq = tem.id" class="faq-button\_\_section">

<img v-if="currentFaq !== tem.id" :src="'./assets/images/' + tem.img"

class="faq-button\_\_section-img"></img>

<img v-else :src="'./assets/images/' + tem.img1" class="faq-button\_\_section-img"></img>

<span class="faq-button\_\_section-title">{{tem.title}}</span>

</div>

</div>

<div class="faq-table">

<div @click="currentQestion = faq.id" v-for="faq in filtredFaq" :key="faq.id"

class="faq-question">

<span :class="{'active': currentQestion == faq.id}" class="question">{{faq.question}}</span>

<span v-show="currentQestion == faq.id " class="answer">{{faq.answer}}</span>

</div>

</div>

</section>

</div>

<div v-if="currentPage==='detail'" class="detail-maraphon">

<div class="detail-maraphon-preview">

<img :src="'./assets/images/' + ditailMaraphon.vrewiew" alt="">

</div>

<h2 class="maraphon-itog\_\_title">Подвeдение итогов</h2>

<div class="container">

<div class="maraphon-itog ">

<div class="maraphon-itog\_\_video">

<video controls :src="'./assets/videos/'+ ditailMaraphon.video"></video>

</div>

<div class="maraphon-itog\_\_winners">

<span class="itog\_\_winners-winner"> <span>Победитель</span> {{ditailMaraphon.winner}}</span>

<span class="itog\_\_winners-second"> <span>Почти победитель </span>

{{ditailMaraphon.secondPlace}}</span>

<span class="itog\_\_winners-third"> <span>Почти почти

победитель</span>{{ditailMaraphon.thirdPlace}}</span>

</div>

</div>

<h2 class="maraphon-itog\_\_title">Результаты других участников марафона</h2>

<div class="video-slider\_\_wrapper ">

<div :style="{'margin-left': '-' + (currentVideo \* 113.6 + '%') }" class="video-slider">

<div v-for="video in resultVideos" :key="video.id" class="video-slide">

<video controls :src="'./assets/videos/' + video.name"></video>

</div>

</div>

<img :class="{disable: currentVideo == 0 }" class="video-slider\_\_btn video-slider\_\_btn--prev"

@click="prevVideo" src="/assets/images/left\_arr.png" alt="">

<img :class="{disable: resultVideos.length - 1 == currentVideo }"

class="video-slider\_\_btn video-slider\_\_btn--next" @click="nextVideo"

src="/assets/images/left\_arr.png" alt="">

</div>

</div>

</div>

<div v-if=" currentPopUp===1" class="pop-up-container">

<form @submit.prevent="sendNumber" class="pop-up">

<h2> Перезвоните мне</h2>

<div class="pop-up\_\_input-container">

<input placeholder="+7 (989)748-32-27" v-model="number.num" required minlength="11"

class="pop-up\_\_input" type="tel">

</div>

<span class="pop-up\_\_confid">Нажимая на кнопку "Отправить", я

соглашаюсь

с условиями <a @click="confid" href="#">политики конфиденциальности.</a> </span>

<input type="submit" class="pop-up\_\_button" value="Отправить">

<img @click="currentPopUp=0;number = '';" src="./assets/images/krest.png" alt=""

class="pop-up\_\_close">

</form>

</div>

<div v-if="currentPopUp===2" class="pop-up-container">

<form @submit.prevent="addToOrders" class="pop-up">

<h2>Для уточнения деталей заказа и доставки</h2>

<div class="pop-up\_\_input-container">

<input required placeholder="Ваш номер телефона ?" v-model="currentOrder.phone"

class="pop-up\_\_input" type="text">

</div>

<div class="pop-up\_\_input-container">

<input required placeholder="Как вас зовут ?" v-model="currentOrder.client"

class="pop-up\_\_input" type="text">

</div>

<div class="pop-up\_\_input-container">

<input placeholder="Ваш адресс? (Не обязательно)" v-model="currentOrder.address"

class="pop-up\_\_input" type="text">

</div>

<span class="pop-up\_\_confid">Нажимая на кнопку "Отправить", я соглашаюсь

с условиями <a @click="confid" href="#">политики конфиденциальности.</a> </span>

<input type="submit" value="Отправить" class="pop-up\_\_button">

<img @click="currentPopUp=0;currentOrder.phone='';currentOrder.client='';currentOrder.address='';"

src="./assets/images/krest.png" alt="" class="pop-up\_\_close">

</form>

</div>

<div v-if="currentPopUp===10" class="pop-up-container">

<form @submit.prevent="addToAdminMar" class="pop-up">

<h2>Регистрация на Марафон </h2>

<div class="pop-up\_\_input-container">

<input required placeholder="Ваш номер телефона ?" v-model="currentMar.phone"

class="pop-up\_\_input" type="text">

</div>

<div class="pop-up\_\_input-container">

<input required placeholder="Как вас зовут ?" v-model="currentMar.client" class="pop-up\_\_input"

type="text">

</div>

<div class="pop-up\_\_input-container">

<input placeholder="Ваш адресс? (Не обязательно)" v-model="currentMar.address"

class="pop-up\_\_input" type="text">

</div>

<span class="pop-up\_\_confid">Нажимая на кнопку "Отправить", я соглашаюсь

с условиями <a @click="confid" href="#">политики конфиденциальности.</a> </span>

<input type="submit" value="Отправить" class="pop-up\_\_button">

<img @click="currentPopUp=0;currentMar.phone='';currentMar.client='';currentMar.address='';"

src="./assets/images/krest.png" alt="" class="pop-up\_\_close">

</form>

</div>

<div v-if="currentPopUp===3" class="pop-up-container">

<form @submit.prevent="logIn" class="pop-up">

<h2 style="text-align: center;font-size: 22px;">Вход</h2>

<div class="pop-up\_\_input-container">

<input v-model="formLog.email" style="font-size: 20px" required

placeholder="Номер телефона/почта" class="pop-up\_\_input" type="text">

</div>

<div class="pop-up\_\_input-container">

<input v-model="formLog.password" style="font-size: 20px" placeholder="Пароль"

required="password" class="pop-up\_\_input" type="password">

</div>

<span class="error" v-if="errors.length !== 0 ">{{errors}}</span>

<span class="pop-up\_\_confid"> Нет аккаунта ? <a href="#"

@click="currentPopUp=4">Заргестрируйтесь.</a> </span>

<input type="submit" value="Войти" class="pop-up\_\_button">

<img @click="currentPopUp=0;formLog.password='';formLog.email='';" src="./assets/images/krest.png"

alt="" class="pop-up\_\_close">

</form>

</div>

<div v-if="currentPopUp===4" class="pop-up-container">

<form @submit.prevent="registration" class="pop-up">

<h2 style="text-align: center;font-size: 22px;">Регистраиця</h2>

<div class="pop-up\_\_input-container">

<input required v-model="formReg.email" style="font-size: 20px"

placeholder="Номер телефона/почта" class="pop-up\_\_input" type="text">

</div>

<span class="error" v-if="errors.length !== 0 && errors?.email ">{{errors.email[0]}}</span>

<div class="pop-up\_\_input-container">

<input required v-model="formReg.login" style="font-size: 20px" placeholder="Логин"

class="pop-up\_\_input" type="text">

</div>

<div class="pop-up\_\_input-container">

<input required v-model="formReg.password" style="font-size: 20px" placeholder="Пароль"

class="pop-up\_\_input" type="password">

</div>

<div class="pop-up\_\_input-container">

<input required v-model="formReg.dbPassword" style="font-size: 20px"

placeholder="Повторите пароль" class="pop-up\_\_input" type="password">

</div>

<span class="error" v-if="errors?.password ">{{errors.password[0]}}</span>

<span class="pop-up\_\_confid">Нажимая на кнопку "Зарегестрироваться", я соглашаюсь

с условиями <a @click="confid" href="#">политики конфиденциальности.</a> </span>

<input type="submit" value="Зарегестрироваться" class="pop-up\_\_button">

<span class="pop-up\_\_confid"> Уже есть аккант ? <a href="#" @click="currentPopUp=3">Войти.</a>

</span>

<img @click="currentPopUp=0;formReg.email = ''; formReg.password = ''; formReg.login = ''; formReg.dbPassword = ''; "

src="./assets/images/krest.png" alt="" class="pop-up\_\_close">

</form>

</div>

<div v-if="currentPopUp===5" class=" pop-up-container">

<div class="pop-up pop-up-5">

<img class="galka" src="./assets/images/galka.png" alt="">

<span class="ok-text">Ваша заявка приянта</span>

<button @click="currentPopUp=0" type="button" class="pop-up\_\_button"> Ок</button>

</div>

</div>

<div v-if="currentPopUp===6" class=" pop-up-container">

<div class="pop-up pop-up-5">

<img class="galka" src="./assets/images/galka.png" alt="">

<span class="ok-text">Ваша заказ добавлен </span>

<button @click="currentPopUp=0" type="button" class="pop-up\_\_button"> Ок</button>

</div>

</div>

<div v-if="currentPopUp===7" class=" pop-up-container">

<div class="pop-up pop-up-5">

<img class="galka" src="./assets/images/galka.png" alt="">

<span class="ok-text">Продукт успешно добавлен</span>

<button @click="currentPopUp=0" type="button" class="pop-up\_\_button"> Ок</button>

</div>

</div>

<div v-if="currentPopUp===8" class=" pop-up-container">

<div class="pop-up pop-up-5">

<img class="galka" src="./assets/images/galka.png" alt="">

<span class="ok-text">Продукт успешно изменён </span>

<button @click="currentPopUp=0" type="button" class="pop-up\_\_button"> Ок</button>

</div>

</div>

<div v-if="currentPopUp===9" class=" pop-up-container">

<div class="pop-up pop-up-5">

<img class="galka" src="./assets/images/galka.png" alt="">

<span class="ok-text">Продукт успешно удалён </span>

<button @click="currentPopUp=0" type="button" class="pop-up\_\_button"> Ок</button>

</div>

</div>

<div v-if=" editingProduct" class="pop-up-container">

<div class="pop-up">

<h2 style="text-align: center;font-size: 22px;">Редактирование продукта</h2>

<div>

<div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Название

продукта:</label>

<input v-model="editingProduct.name" style="font-size: 20px" class="pop-up\_\_input"

type="text">

</div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Фото

продукта:</label>

<input v-model="editingProduct.img" style="font-size: 20px"

placeholder="Введите название картинки" class="pop-up\_\_input" type="text">

</div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Цена

продукта:</label>

<input v-model="editingProduct.price" style="font-size: 20px" class="pop-up\_\_input"

type="text">

</div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Калории

продукта:</label>

<input v-model="editingProduct.calories" style="font-size: 20px" class="pop-up\_\_input"

type="text">

</div>

</div>

<div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Белки

продукта:</label>

<input v-model="editingProduct.proteins" style="font-size: 20px" class="pop-up\_\_input"

type="text">

</div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Жиры

продукта:</label>

<input v-model="editingProduct.fats" style="font-size: 20px" class="pop-up\_\_input"

type="text">

</div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Углеводы

продукта:</label>

<input v-model="editingProduct.carbohdrate" style="font-size: 20px"

class="pop-up\_\_input" type="text">

</div>

</div>

</div>

<button @click="postChanges" class="pop-up\_\_button">Редактировать</button>

<img @click="editingProduct=null" src="./assets/images/krest.png" alt="" class="pop-up\_\_close">

</div>

</div>

<div v-if="creatingProduct" class="pop-up-container">

<div class="pop-up">

<h2 style="text-align: center;font-size: 22px;">Добавление продукта</h2>

<div>

<div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Название

продукта:</label>

<input v-model="creatingProduct.name" style="font-size: 20px" class="pop-up\_\_input"

type="text">

</div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Фото

продукта:</label>

<input v-model="creatingProduct.img" style="font-size: 20px"

placeholder="Введите название изображения" class="pop-up\_\_input" type="text">

</div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Категоря

продукта:</label>

<input v-model="creatingProduct.category\_id" style="font-size: 20px"

class="pop-up\_\_input" type="text">

</div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Цена

продукта:</label>

<input v-model="creatingProduct.price" style="font-size: 20px" class="pop-up\_\_input"

type="text">

</div>

</div>

<div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Калории

продукта:</label>

<input v-model="creatingProduct.calories" style="font-size: 20px" class="pop-up\_\_input"

type="text">

</div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Белки

продукта:</label>

<input v-model="creatingProduct.proteins" style="font-size: 20px" class="pop-up\_\_input"

type="text">

</div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Углеводы

продукта:</label>

<input v-model="creatingProduct.carbohdrate" style="font-size: 20px"

class="pop-up\_\_input" type="text">

</div>

<div class="pop-up\_\_input-container">

<label style="font-size: 20px;margin-bottom: 10px;font-weight: bold;" for="">Жиры

продукта:</label>

<input v-model="creatingProduct.fats" style="font-size: 20px" class="pop-up\_\_input"

type="text">

</div>

</div>

</div>

<button @click="postProduct" class="pop-up\_\_button">Добавить</button>

<img @click="creatingProduct=null" src="./assets/images/krest.png" alt="" class="pop-up\_\_close">

</div>

</div>

</main>

<div class="container">

<footer class="footer ">

<span>2023 © Все права защищены</span>

<div class="footer\_\_icons">

<a target="\_blank" href="https://web.telegram.org/k/#681550712"> <img

src="./assets/images/telegram 1.svg" alt="" class="footer\_\_icon"></a>

<a target="\_blank" href="https://instagram.com/baskay\_time?igshid=MzRlODBiNWFlZA=="><img

src="./assets/images/instagram 1.svg" alt="" class="footer\_\_icon"></a>

</div>

</footer>

</div>

</div>

<script src="js/vue.global.js"></script>

<script src="js/helper.js"></script>

<script type="module" src="js/main.js"></script>

</body>

</html>

@font-face {

font-family: MontB;

src: url(./fonts/Montserrat-Bold.ttf);

}

@font-face {

font-family: MontBI;

src: url(./fonts/Montserrat-BoldItalic.ttf);

}

@font-face {

font-family: MontM;

src: url(./fonts/Montserrat-Medium.ttf);

}

@font-face {

font-family: MontI;

src: url(./fonts/Montserrat-Italic.ttf);

}

\* {

padding: 0;

margin: 0;

box-sizing: border-box;

font-family: MontM;

}

img {

display: inline-block;

}

a {

text-decoration: none;

color: inherit;

}

li {

list-style: none;

}

body {

height: 1000px;

}

.container {

max-width: 1220px;

margin: 0 auto;

}

/\* -----------------------------------HEADER---------------------------------- \*/

.wrapper {

max-width: 100%;

overflow: hidden;

}

.header {

background-color: #00b011;

min-height: 60px;

display: flex;

align-items: center;

position: fixed;

width: 100%;

z-index: 99;

top: 0;

left: 0;

}

.burger\_\_menu-btn {

width: 40px;

height: 40px;

background-color: inherit;

border: none;

position: relative;

display: none;

z-index: 10;

}

.burger\_\_menu-btn span {

position: absolute;

width: 30px;

height: 3px;

background-color: white;

left: 5px;

transition: transform 0.5s, opacity 0.5s;

}

.burger\_\_menu-btn span:nth-child(1) {

transform: translateY(-8px);

}

.burger\_\_menu-btn span:nth-child(2) {

transform: translateY(8px);

}

.haeder\_\_autorization {

font-size: 18px;

color: white;

right: 3%;

border-bottom: 2px solid white;

}

.haeder\_\_admin-panel {

font-size: 18px;

}

.heder-content div {

display: flex;

align-items: center;

gap: 20px;

}

.heder-content {

width: 100%;

display: flex;

justify-content: space-between;

align-items: center;

color: white;

}

.header-content\_\_logo {

font-family: MontB;

font-size: 28px;

text-transform: uppercase;

}

.header-content\_\_logo span {

color: #fd9c00;

}

.header-content\_\_menu {

width: 45%;

display: flex;

justify-content: space-between;

align-items: center;

font-size: 18px;

}

.header-content\_\_menu a {

text-decoration: underline;

font-family: MontB;

cursor: pointer;

}

.header-content\_\_menu a:hover {

text-shadow: 5px 5px 12px rgb(120, 120, 120);

}

.header-content\_\_menu-tel {

font-family: MontI;

cursor: pointer;

position: relative;

text-decoration: none;

}

.header-content\_\_menu-tel::before {

content: url("./images/header-phone.svg");

position: absolute;

top: -6px;

left: -40px;

}

.header-content\_\_menu-tel:hover {

text-shadow: 5px 5px 12px rgb(120, 120, 120);

}

.header-content\_\_menu button {

background-color: white;

border-radius: 20px;

border: none;

padding: 0 20px;

color: #44a04d;

font-size: 18px;

cursor: pointer;

}

.header-content\_\_menu button:hover {

box-shadow: 5px 5px 12px rgb(120, 120, 120);

}

.header-content\_\_link {

text-transform: uppercase;

text-decoration: underline;

font-family: MontBI;

font-size: 20px;

}

.header-content\_\_link:hover {

text-shadow: 5px 5px 12px rgb(120, 120, 120);

}

/\* -----------------------------------intro------------------------------------ \*/

.main {

padding-top: 60px;

}

.intro {

position: relative;

margin-top: 63px;

}

.intro-img {

width: 100%;

}

.intro-btn {

position: absolute;

bottom: 20%;

left: 7%;

width: 17%;

height: 15%;

border-radius: 20px;

border: none;

color: white;

font-size: 26px;

background-color: #00b011;

cursor: pointer;

}

/\* -----------------------------------profit------------------------------------ \*/

.profits {

margin-top: 60px;

}

.section-caption {

font-family: MonstB;

font-size: 30px;

margin: 35px 0;

text-align: center;

}

.section-caption::after {

content: "";

margin: 30px auto;

display: block;

height: 4px;

width: 300px;

background: #00b011;

}

.profits-line {

display: flex;

justify-content: space-between;

align-items: center;

width: 70%;

margin: 0 auto;

}

.profits-line\_\_item {

text-align: center;

}

.profits-line\_\_item span {

margin-top: 10px;

text-align: center;

display: block;

line-height: 0.9;

}

/\* ---------------------------------------programs----------------------------------- \*/

.programs {

margin-top: 60px;

}

.categories\_\_line {

display: flex;

justify-content: space-between;

}

.category {

width: 19%;

margin-left: 5px;

border: 2px solid rgb(0, 0, 0, 0.47);

border-radius: 20px;

cursor: pointer;

transition: all 0.2s;

text-align: center;

padding: 20px 0;

}

.category.active {

border: 2px solid #00a524;

transform: scale(1.02);

}

.category:hover {

transition: all 0.2s;

transform: scale(1.02);

}

.category-name {

font-family: MontB;

font-size: 20px;

}

.category-dishes {

font-family: MontM;

font-size: 18px;

opacity: 0.6;

}

.current-program {

width: 80%;

margin: 0 auto;

margin-top: 60px;

}

.program-caption {

text-align: center;

opacity: 60%;

font-size: 18px;

}

.product-line {

display: flex;

justify-content: center;

align-items: start;

flex-wrap: wrap;

margin-top: 60px;

}

.product {

flex: 0 0 33.333%;

}

.product div {

display: flex;

gap: 170px;

margin-top: 10px;

}

.product-img {

width: 80%;

}

.product-img img {

width: 100%;

height: 100%;

object-fit: cover;

object-position: center center;

border-radius: 35px;

box-shadow: 5px 5px 5px rgb(167, 167, 167);

}

.product-name {

display: block;

margin-top: 15px;

color: black;

font-family: MontB;

width: 90%;

}

.product-btn {

width: 50%;

background-color: #fd9c09;

margin-top: 10px;

border: none;

border-radius: 5px;

font-size: 18px;

cursor: pointer;

transition: 0.5s;

padding: 5px 0px;

}

.product-btn:hover {

transition: 0.5s;

transform: scale(1.01);

}

.total-calories {

border: 1px solid black;

width: 27%;

padding: 25px;

margin-top: 5px;

border-radius: 30px;

}

.total-calories h2 {

font-size: 16px;

margin-bottom: 10px;

}

.total-calories > span {

color: gray;

}

.total-elements {

margin-top: 20px;

justify-content: space-between;

display: flex;

text-align: center;

}

.total-elements > span {

color: gray;

}

.total-calories b {

color: black;

}

.current-program\_\_button {

background-color: rgb(81, 81, 81);

color: white;

position: relative;

width: 20%;

border: none;

border-radius: 20px;

padding: 10px;

font-size: 18px;

margin-top: 10px;

cursor: pointer;

margin-top: 20px;

float: right;

right: 80px;

}

.current-program\_\_button:hover {

transform: scale(1.01);

}

.update-product\_\_line {

background-color: white;

border-radius: 25px;

border: 1px solid gray;

position: fixed;

left: 50%;

top: 50%;

z-index: 10;

transform: translate(-50%, -50%);

width: 70%;

display: flex;

align-items: center;

overflow: hidden;

padding: 60px 70px 50px 30px;

}

.update-product\_\_line-container::before {

content: "";

width: 100%;

height: 100%;

position: fixed;

background-color: #000;

opacity: 0.8;

left: 0;

top: 0;

z-index: 9;

}

.update-product\_\_line-slider {

display: flex;

gap: 30px;

width: 100%;

transition: 0.5s;

padding-bottom: 30px;

}

.update-product {

flex: 0 0 33%;

border: 2px solid gray;

border-radius: 20px;

display: flex;

align-items: center;

flex-direction: column;

justify-content: center;

padding: 20px 30px;

}

.product-elements span {

font-family: MontB;

}

.product-elements {

display: flex;

width: 100%;

align-items: center;

margin: 0 auto;

justify-content: space-around;

font-size: 18px;

margin-bottom: 25px;

}

.update-product-img {

margin: 20px 0;

}

.update-product-name {

display: block;

text-align: center;

color: black;

}

.update-product-btn {

display: block;

margin: 20px 0px;

width: 100%;

padding: 10px 15px;

border: 1px solid #00a524;

background-color: inherit;

font-size: 18px;

border-radius: 20px;

cursor: pointer;

color: #00a524;

transition: 0.5s;

}

.update-product-btn:hover {

transform: scale(1.01);

transition: 0.5s;

}

.update-product-calories {

font-family: MontB;

}

.update-product\_\_krestik {

position: absolute;

right: 5px;

top: 5px;

cursor: pointer;

}

.update-product\_\_arrow {

position: absolute;

cursor: pointer;

bottom: 0px;

}

.update-product\_\_arrow--right {

right: 25px;

transform: rotate(180deg);

}

/\* ------------------------------results------------------------- \*/

.result {

margin-top: 100px;

}

.slider-wrapper {

width: 100%;

margin: 0 auto;

position: relative;

margin-top: 100px;

}

.slider {

width: 100%;

transition: 0.5s;

display: flex;

}

.slide {

margin-right: 10%;

transition: 0.5s;

flex: 0 0 100%;

display: flex;

box-shadow: 7px 7px 30px -11px black;

border-radius: 20px;

padding: 40px 30px;

}

.slide.active {

transform: scale(1.1);

}

.slide-img\_\_contaienr {

display: flex;

justify-content: center;

width: 50%;

align-items: center;

}

.slide-img {

width: 90%;

height: 400px;

}

.slide-content {

padding: 40px;

width: 50%;

position: relative;

}

.slide-content\_\_name {

font-size: 28px;

font-family: MontB;

}

.slide-content\_\_menu {

line-height: 2;

margin-top: 20px;

}

.slide-content\_\_list {

font-size: 18px;

position: relative;

padding-left: 25px;

}

.slide-content\_\_list::before {

content: "";

position: absolute;

left: 0;

top: 13px;

border-radius: 50%;

width: 13px;

height: 13px;

background-color: #00a524;

display: inline;

}

.slide-content\_\_comment {

display: flex;

width: 100%;

align-items: center;

justify-content: center;

height: 150px;

}

.slide-content\_\_button {

width: 80%;

margin-left: 50%;

transform: translateX(-50%);

background-color: #00b010;

border: none;

border-radius: 20px;

padding: 15px 0px;

cursor: pointer;

margin-top: 20px;

color: white;

font-size: 20px;

font-family: MontB;

}

.slide-content\_\_result {

position: absolute;

right: -40px;

bottom: -50px;

background-color: #f90000;

width: 80px;

height: 80px;

border-radius: 50%;

color: white;

font-size: 20px;

font-family: MontB;

display: flex;

align-items: center;

justify-content: center;

}

.slider-btn {

position: absolute;

top: 50%;

transform: translateY(-50%);

cursor: pointer;

width: 80px;

}

.prev-btn {

left: -10%;

}

.slider-btn.disable {

opacity: 60%;

}

.next-btn {

right: -10%;

transform: translateY(-50%);

}

.pagination-line {

position: absolute;

bottom: -80px;

left: 50%;

transform: translateX(-50%);

width: 20%;

display: flex;

justify-content: space-between;

}

.pagination {

width: 20px;

height: 20px;

border: 1px solid rgb(226, 226, 226);

border-radius: 50%;

transition: all 0, 5s;

background-color: rgb(226, 226, 226);

cursor: pointer;

}

.pagination.active {

background-color: #00a524;

transform: scale(1.1);

transition: all 0, 5s;

}

/\* ----------------------------------------advice---------------------------------- \*/

.advice {

margin-top: 150px;

}

.advice-items {

margin-top: 60px;

}

.advice-item {

display: flex;

margin-top: 10px;

}

.advice-item\_\_content {

flex-direction: column;

width: 50%;

font-size: 20px;

}

.advice-item\_\_caption {

color: #00b011;

font-size: 30px;

font-family: MontB;

text-align: center;

}

.advice-item\_\_content span {

display: block;

padding: 0px 60px;

margin-top: 60px;

}

.advice-item\_\_img-container {

display: flex;

justify-content: center;

align-items: center;

width: 50%;

}

.advice-item\_\_img {

width: 100%;

}

/\* ---------------------------------------faq---------------------- \*/

.faq {

margin-top: 80px;

}

.faq-button-line {

margin-top: 50px;

display: flex;

justify-content: space-between;

width: 80%;

margin: 0 auto;

}

.faq-button\_\_section {

border: 2px solid #a4a4a4;

text-align: center;

flex: 0 0 23%;

margin-left: 15px;

box-shadow: 3px 4px 2px 0px gray;

border-radius: 25px;

cursor: pointer;

padding: 15px 0px;

display: flex;

flex-direction: column;

justify-content: center;

align-items: center;

}

.faq-button\_\_section.active {

background-color: #00a524;

color: white;

box-shadow: none;

}

.faq-button\_\_section:hover {

transform: scale(1.01);

}

.faq-button\_\_section-img {

display: block;

}

.faq-button\_\_section-img {

width: 70px;

height: 70px;

}

.faq-button\_\_section.active > .faq-button\_\_section-img path {

fill: white;

}

.faq-button\_\_section-img path {

fill: white;

}

.faq-button\_\_section-title {

font-size: 18px;

}

.faq-table {

width: 90%;

margin: 0 auto;

border: 1px solid rgb(0, 0, 0, 60%);

margin-top: 80px;

box-shadow: 5px 7px 8px 0px rgb(0, 0, 0, 40%);

}

.question {

cursor: pointer;

}

.faq-question span {

padding: 30px;

display: block;

border-bottom: 1px solid #000;

font-size: 20px;

}

.question.active {

color: #00a524;

}

.answer {

transition: all 0.5s;

font-size: 18px;

}

.footer {

margin-top: 80px;

padding: 50px 0;

border-top: 3px solid #bebebe;

display: flex;

justify-content: space-around;

font-size: 22px;

}

.footer\_\_icons {

display: flex;

justify-content: space-between;

width: 23%;

cursor: pointer;

}

.footer\_\_icon {

width: 40%;

}

/\* ------------------------------------pop-up------------------------------- \*/

.pop-up-container {

position: fixed;

left: 0;

top: 0;

z-index: 10;

inset: 0;

display: flex;

justify-content: center;

align-items: center;

}

.pop-up-container::before {

content: "";

inset: 0;

z-index: 10;

background-color: #000;

position: fixed;

opacity: 80%;

}

.pop-up {

position: relative;

background-color: white;

z-index: 10;

padding: 25px;

width: 30%;

border-radius: 10px;

}

.pop-up > div {

display: flex;

gap: 30px;

}

.pop-up h2 {

font-family: MontM;

font-size: 18px;

margin-bottom: 10px;

text-align: start;

}

.pop-up\_\_input-container {

width: 100%;

}

.pop-up\_\_input {

font-size: 22px;

border: none;

border-bottom: 1px solid #000;

width: 100%;

outline: none;

padding: 10px 20px;

margin-bottom: 20px;

}

.pop-up\_\_confid {

font-size: 16px;

display: block;

width: 100%;

margin-bottom: 10px;

}

.pop-up\_\_close {

position: absolute;

right: 10px;

top: 10px;

width: 40px;

cursor: pointer;

}

.pop-up\_\_button {

width: 100%;

padding: 10px;

background-color: inherit;

cursor: pointer;

}

.pop-up\_\_confid a {

text-decoration: underline solid;

}

/\* --------------------maraphon------------ \*/

.current-marphon {

margin-top: 60px;

}

.other\_\_maraphons {

margin-top: 40px;

}

.maraphons h2 {

text-align: center;

margin-bottom: 30px;

}

.maraphon\_\_preview {

width: 100%;

position: relative;

margin-bottom: 60px;

}

.maraphon\_\_preview img {

width: 100%;

border-radius: 20px;

max-height: 500px;

}

.maraphon\_\_preview button {

position: absolute;

right: 10%;

bottom: 8%;

width: 20%;

padding: 10px 0px;

border: none;

cursor: pointer;

outline: none;

font-size: 18px;

border-radius: 5px;

}

.maraphon\_\_preview-btn {

background-color: #00a524;

color: white;

}

.maraphon\_\_preview-btn--second {

display: none;

}

.maraphon\_\_preview--orange {

background-color: #fd9c00;

color: black;

}

.maraphon\_\_conditions {

display: flex;

width: 80%;

margin: 0 auto;

margin-top: 40px;

}

.maraphon\_\_conditions-video video {

width: 100%;

object-fit: cover;

height: 100%;

object-position: center center;

border-radius: 20px;

}

.maraphon\_\_conditions-video {

width: 100%;

height: 600px;

border-radius: 20px;

}

.maraphon\_\_conditions-rules {

padding: 30px 40px;

margin-left: 40px;

}

.maraphon\_\_conditions-rules span {

font-weight: bold;

font-size: 20px;

padding-left: 20px;

display: block;

margin-bottom: 15px;

}

.maraphon\_\_conditions-rules p {

font-size: 18px;

line-height: 2.5;

padding-left: 20px;

position: relative;

}

.maraphon\_\_conditions-rules p::before {

content: "";

position: absolute;

left: 2px;

top: 16px;

width: 10px;

height: 10px;

border-radius: 50%;

background-color: #00a524;

}

.maraphon\_\_conditions-container {

margin-top: 30px;

}

.maraphon\_\_conditions button {

position: relative;

right: 10%;

bottom: 0%;

width: 100%;

padding: 10px 0px;

border: none;

cursor: pointer;

outline: none;

font-size: 18px;

border-radius: 5px;

margin-top: 40px;

margin-left: 30px;

}

.detail-maraphon-preview {

width: 90%;

margin: 0 auto;

margin-top: 100px;

}

.detail-maraphon-preview img {

width: 100%;

max-height: 600px;

border-radius: 20px;

object-fit: cover;

object-position: center center;

}

.maraphon-itog\_\_title {

text-align: center;

margin-top: 30px;

}

.maraphon-itog {

display: flex;

margin: 40px auto;

padding: 40px;

border-radius: 20px;

box-shadow: 6px 6px 20px gray;

}

.maraphon-itog\_\_winners {

padding: 40px 40px;

margin-left: 100px;

}

.maraphon-itog\_\_winners span {

display: block;

}

.maraphon-itog\_\_video {

width: 40%;

height: 400px;

}

.maraphon-itog\_\_video video {

border-radius: 15px;

width: 100%;

height: 100%;

object-fit: cover;

object-position: center center;

}

.itog\_\_winners-winner {

margin-bottom: 40px;

font-size: 20px;

}

.itog\_\_winners-winner span {

color: #00a524;

margin: 10px 0;

font-size: 22px;

display: block;

text-align: center;

font-weight: bold;

}

.itog\_\_winners-second {

font-size: 18px;

margin-bottom: 40px;

}

.itog\_\_winners-second span {

margin: 10px 0;

color: #fd9c00;

font-size: 20px;

display: block;

text-align: center;

font-weight: bold;

}

.itog\_\_winners-third {

font-size: 18px;

}

.itog\_\_winners-third span {

margin: 10px 0;

font-size: 20px;

display: block;

text-align: center;

font-weight: bold;

}

.video-slider\_\_wrapper {

width: 30%;

margin: 0 auto;

margin-top: 40px;

position: relative;

}

.video-slider {

display: flex;

width: 100%;

gap: 50px;

transition: 0.5s;

}

.video-slide {

flex: 0 0 100%;

}

.video-slide video {

width: 100%;

height: 100%;

object-fit: cover;

object-position: center center;

}

.video-slider\_\_btn--prev.disable {

opacity: 40%;

cursor: default;

}

.video-slider\_\_btn--next.disable {

opacity: 40%;

cursor: default;

}

.video-slider\_\_btn--prev {

left: -10%;

}

.video-slider\_\_btn--next {

right: -10%;

transform: rotate(180deg);

}

.video-slider\_\_btn {

position: absolute;

bottom: -70px;

cursor: pointer;

}

.error {

display: block;

font-size: 18px;

color: red;

margin-bottom: 10px;

}

/\* ---------------admin------------------ \*/

.empty\_\_orders {

width: 100%;

display: flex;

justify-content: center;

align-items: center;

font-size: 24px;

margin-top: 100px;

}

.admin-navigation {

margin-top: 30px;

font-size: 20px;

display: flex;

gap: 20px;

justify-content: center;

}

.admin-navigation span {

cursor: pointer;

}

.admin-navigation span.active {

transition: all 0.1s;

border-bottom: 2px solid black;

}

.admin-menu {

display: flex;

margin-top: 60px;

max-height: 600px;

}

.admin-menu\_\_categories {

display: flex;

flex-direction: column;

gap: 30px;

border-right: 2px solid black;

padding-right: 50px;

}

.admin-menu\_\_category {

border-radius: 20px;

border: 2px solid rgb(154, 154, 154);

text-align: center;

padding: 20px 20px;

cursor: pointer;

transition: all 0.1s;

}

.admin-menu\_\_category--add {

color: white;

background-color: #00a524;

border-color: #00a524;

}

.admin-menu\_\_category:hover {

transform: scale(1.01);

transition: all 0.1s;

}

.admin-menu\_\_category.active {

transition: all 0.1s;

border: 2px solid #00a524;

}

.admin-menu\_\_products {

margin-left: 80px;

width: 60%;

overflow: scroll;

}

.admin-menu\_\_product {

box-shadow: 2px 2px 2px gray;

border-radius: 20px;

padding: 30px 20px;

margin-top: 20px;

display: flex;

align-items: center;

width: 100%;

justify-content: space-between;

}

.admin-product\_\_img img {

width: 100%;

height: 100%;

object-fit: cover;

object-position: center center;

}

.admin-product\_\_img {

width: 15%;

}

.admin-product\_\_img2-container {

width: 60%;

display: flex;

gap: 2px;

}

.admin-product\_\_img2 img {

width: 80%;

height: 100%;

object-fit: cover;

object-position: center center;

}

.admin-product\_\_img {

width: 20%;

}

.galka {

width: 50%;

margin-bottom: 40px;

}

.ok-text {

margin-bottom: 20px;

font-size: 20px;

}

.pop-up-5 {

display: flex;

flex-direction: column;

align-items: center;

}

.admin-product\_\_name {

font-weight: bold;

flex: 0 0 40%;

line-height: 1.7;

}

.admin-product\_\_delete img {

width: 100%;

height: 100%;

object-fit: cover;

object-position: center center;

}

.admin-product\_\_update img {

width: 100%;

height: 100%;

object-fit: cover;

object-position: center center;

}

.admin-product\_\_update,

.admin-product\_\_delete {

cursor: pointer;

width: 3%;

}

/\* --------------breakpoints-header--------- \*/

@media (min-width: 1800px) {

}

@media (max-width: 1700px) {

.advice-item\_\_content span {

margin-top: 40px;

font-size: 19px;

}

}

@media (max-width: 1500px) {

.slider-wrapper {

width: 80%;

}

.prev-btn {

left: -5%;

}

.next-btn {

right: -5%;

}

.advice-item\_\_content span {

margin-top: 20px;

font-size: 17px;

}

.pop-up {

width: 50%;

}

}

@media (max-width: 1350px) {

.update-product\_\_krestik {

top: 20px;

}

.update-product\_\_arrow {

display: none;

}

.update-product\_\_line-slider {

align-items: center;

gap: 10;

overflow: scroll;

position: static;

width: 100%;

padding: 0px;

}

.update-product\_\_line {

padding: 25px;

}

.update-product {

align-self: normal;

flex: 0 0 40%;

}

.product-elements {

font-size: 16px;

margin-bottom: 5px;

}

.update-product-img {

width: 50%;

margin: 5px 0;

}

.update-product-btn {

font-size: 16px;

margin: 5px 0px;

padding: 4px;

}

}

@media (max-width: 1300px) {

}

@media (max-width: 1200px) {

.maraphon\_\_conditions-video {

width: 100%;

}

.header-content\_\_menu {

font-size: 16px;

}

.header-content\_\_menu-tel {

font-size: 16px;

}

.header-content\_\_link {

font-size: 18px;

}

.haeder\_\_autorization {

font-size: 16px;

}

.header-content\_\_menu button {

font-size: 16px;

}

.header-content\_\_logo {

font-size: 20px;

}

.container {

padding: 0px 10px;

}

.advice-item\_\_content span {

margin-top: 19px;

font-size: 16px;

}

.advice-item\_\_caption {

font-size: 20px;

}

.section-caption {

font-size: 26px;

margin: 20px 0;

}

.section-caption::after {

content: "";

margin: 20px auto;

display: block;

height: 3px;

width: 300px;

background: #00b011;

}

}

@media (max-width: 1100px) {

.detail-maraphon-preview {

margin: 5px 5px;

width: 100%;

}

.maraphon\_\_preview button {

font-size: 16px;

}

.maraphon\_\_preview-btn--first {

display: none;

}

.maraphon\_\_preview-btn--second {

width: 30%;

display: block;

}

.maraphon\_\_conditions button {

width: 100%;

margin-left: 0px;

left: 50%;

transform: translateX(-50%);

margin-top: 20px;

}

.maraphon\_\_conditions {

display: block;

}

.maraphon\_\_conditions-rules {

display: flex;

gap: 30px;

}

.maraphon\_\_conditions-rules p {

line-height: 2;

font-size: 16px;

}

.maraphon\_\_conditions-rules p::before {

top: 9px;

}

.maraphon\_\_conditions-rules span {

text-align: center;

font-size: 18px;

}

.header-content\_\_menu {

font-size: 14px;

}

.header-content\_\_menu-tel {

font-size: 14px;

}

.header-content\_\_link {

font-size: 16px;

}

.haeder\_\_autorization {

font-size: 14px;

}

.header-content\_\_menu button {

font-size: 14px;

}

.header-content\_\_logo {

font-size: 20px;

}

.container {

padding: 0px 10px;

}

.slide-content\_\_name {

font-size: 20px;

}

.slide-content\_\_list {

font-size: 16px;

}

.slide-content\_\_comment {

font-size: 14px;

}

.slide-img\_\_contaienr {

width: 50%;

}

.slide-content\_\_button {

font-size: 16px;

width: 100%;

padding: 10px 0px;

}

.slider-btn {

width: 8%;

}

.slide-content\_\_result {

width: 20%;

}

.advice-item\_\_content span {

margin-top: 14px;

font-size: 14px;

}

.advice-item\_\_caption {

font-size: 18px;

}

.profits-line\_\_item span {

margin-top: 10px;

text-align: center;

display: block;

font-size: 14px;

line-height: 1;

}

.profits-line\_\_item-img img {

width: 50px;

object-fit: cover;

object-position: center center;

}

.profits-line\_\_item-img {

display: flex;

justify-content: center;

width: 100%;

}

.category-name {

font-size: 16px;

}

.category-dishes {

font-size: 14px;

}

.total-elements span {

font-size: 14px;

}

.faq-button\_\_section-img {

width: 30%;

}

.faq-button\_\_section-title {

font-size: 16px;

}

.faq-question span {

font-size: 16px;

}

.faq-table {

margin-top: 40px;

}

.footer {

font-size: 18px;

padding-top: 30px;

margin-top: 100px;

}

.footer\_\_icon {

width: 35%;

}

.faq-button\_\_section {

padding: 0px;

}

.pagination {

width: 15px;

height: 15px;

}

.pop-up {

width: 80%;

}

}

@media (max-width: 1000px) {

.haeder\_\_admin-panel {

font-size: 16px;

}

.admin-product\_\_update,

.admin-product\_\_delete {

width: 5%;

}

.admin-product\_\_name {

font-size: 12px;

line-height: 1.2;

}

.admin-product\_\_img {

width: 19%;

}

.admin-menu\_\_products {

margin-left: 20px;

padding-right: 30px;

}

.maraphon-itog\_\_winners span {

text-align: center;

}

.maraphon-itog {

display: block;

}

.maraphon-itog\_\_video {

width: 100%;

}

.slide {

display: block;

}

.slide-img\_\_contaienr {

width: 100%;

}

.slide-content {

width: 100%;

}

.slide-content\_\_comment {

width: 100%;

}

.slide-content\_\_result {

width: 12%;

}

.intro-btn {

font-size: 20px;

}

.section-caption {

font-size: 22px;

}

.section-caption::after {

content: "";

margin: 15px auto;

display: block;

height: 2px;

width: 200px;

background: #00b011;

}

.total-calories {

width: 30%;

}

.advice-item {

flex-direction: column;

align-items: center;

}

.advice-item\_\_img-container {

width: 90%;

}

.advice-item\_\_content {

width: 100%;

margin-bottom: 10px;

}

.advice-item\_\_img-container--order {

order: 2;

}

.burger\_\_menu-btn {

display: block;

}

.header-content\_\_menu {

position: absolute;

width: 300px;

height: 200px;

left: -300px;

top: -200px;

transition: transform 0.5s;

background-color: #00b026;

border-radius: 0px 0px 150px 0px;

}

.header-content\_\_menu {

display: block;

padding: 63px 20px 20px 20px;

}

.header-content\_\_menu a {

display: block;

line-height: 2;

font-size: 18px;

}

.header-content\_\_menu-tel::before {

content: "";

}

.header.open .header-content\_\_menu {

transform: translate(100%, 100%);

}

.header.open .burger\_\_menu-btn span:nth-child(1) {

transform: translate(0) rotate(45deg);

}

.header.open .burger\_\_menu-btn span:nth-child(2) {

opacity: 0;

}

.header.open .burger\_\_menu-btn span:nth-child(3) {

transform: translate(0) rotate(-45deg);

}

}

import product from "./product.js";

import { post } from "./helper.js";

const App = {

data() {

return {

categories: [

{ id: 0, name: 'Завтрак', dishes: 2 },

{ id: 1, name: 'Основное', dishes: 3 },

{ id: 2, name: 'Доплнительное', dishes: 4 },

{ id: 3, name: 'Полное', dishes: 5 },

{ id: 4, name: 'Расширенное', dishes: 6 },

],

categoriesProducts: [],

adminCategories: [

{ id: 0, name: 'Завтрак', },

{ id: 1, name: '2-ой завтрак', },

{ id: 2, name: 'Обед', },

{ id: 3, name: 'Полдник', },

{ id: 4, name: 'Ужин', },

{ id: 5, name: 'Напитки', },

],

currentCategory: 'Основное',

breakfasts: [],

secondBreakfasts: [],

lunchs: [],

snacks: [],

dinners: [],

waters: [],

clients: [

{ id: 0, name: 'Сослан', howOld: 60, img: "client(1).jpg", programmName: " 'Полное' ", programmCount: 22, comment: "«Похудел на 15 кг без спорта всего за месяц. Физических нагрузок хватало в огороде»", result: 15, },

{ id: 1, name: 'Маир', howOld: 40, img: "client(2).jpg", programmName: "'Полное'", programmCount: 21, comment: "«Работа сидячая, по городу в основном на такси. В сентябре решил приводить себя в форму. Решил начать с питания. Заказывал линейку mFit, 1600 калорий в день. За первые две недели ушло 2 килограмма. Начал заниматься фитнесом. Сейчас мой вес 73, так же заказываю Baskay Time. »", result: 26, },

{ id: 2, name: 'Анжела', howOld: 37, img: "client(3).jpg", programmName: "'Основное'", programmCount: 30, comment: "«Мне очень нравится ваша еда! У меня никогда не было каких-либо очень привередливых вкусов в еде, плюс когда я решила встать на «путь истинный» = похудения, я понимала, что, мне не так важно, насколько это будет вкусно, как насколько это будет эффективно! Чтобы я не была голодной и при этом худела»", result: 18, },

],

tems: [

{ id: 0, img: 'tomato.svg', img1: 'tomato-white.svg', title: 'Продукты' },

{ id: 1, img: 'card.svg', img1: 'card-white.svg', title: 'Оплата и доставка' },

{ id: 2, img: 'body.svg', img1: 'body-white.svg', title: 'Похудение' },

{ id: 3, img: 'sneg.svg', img1: 'sneg-white.svg', title: 'Хранение' },

],

faq: [

{ id: 0, categoryId: 1, question: 'Как я могу оплатить свой заказ?', answer: 'Оплатить заказ можно несколькими способами: банковской картой или наличными курьеру в день первой доставки питания.Если вы меняете блюда, воспользуйтесь безналичной оплатой. Оплата подписки происходит до 19:00 дня списания, иначе автоматически включается заморозка.' },

{ id: 1, categoryId: 0, question: 'Используете ли вы сахар или сахарозаменитель. Если да, то какой?', answer: 'В меню сахар заменен на натуральный сахарозаменитель Fitparad 7. Подробный состав вы всегда можете уточнить у наших специалистов call-центра.' },

{ id: 2, categoryId: 0, question: 'Какие продукты вы используете для готовки?', answer: 'Мы используем диетические сорта мяса, например куриное филе, филе индейки и говядину. Молочная продукция оптимальна по жирности — до 5% ' },

{ id: 3, categoryId: 2, question: 'Сколько калорий в день нужно потреблять, чтобы похудеть?', answer: 'Чаще всего наши клиенты выбирают рационы на 800 ккал, 1000 ккал или 1200 ккал. Но все индивидуально. В интернете вы найдете множество методов, с помощью которых легко определить энергетическую ценность вашего меню. Самые популярные формулы: Харриса — Бенедикта и Миффлина — Сан Жеора. Вам нужно больше 1200 ккал для комфортного похудения? Присмотритесь к линейке Основное на 1400 ккал. Хорошие новости: о кухонных весах можно забыть навсегда. Мы все посчитали, чтобы вы ели и худели, не занимая голову лишней информацией.' },

{ id: 4, categoryId: 3, question: 'Какой срок хранения вашей продукции?', answer: 'Контейнеры с едой необходимо хранить в холодильнике, соблюдая температурный режим. Мы используем новейшую технологию газомодифицированной среды с герметичной запайкой и температурным контролем, которая позволяет сохранять свежесть блюд несколько дней, при условии хранения в холодильнике.На контейнерах указана дата производства.Мы рекомендуем соблюдать последовательность приемов пищи и не хранить наши блюда дольше определенного срока во избежание их порчи.' },

{ id: 5, categoryId: 3, question: 'Можно ли разогревать еду в микроволновке?', answer: 'Контейнеры предназначены для разогрева в микроволновой печи. Просто снимите защитную пленку и поместите контейнер на пару минут в микроволновку.' },

{ id: 6, categoryId: 2, question: 'Когда будет результат???', answer: 'Сразу. За неделю наши клиенты теряют от 500 г до 2,5 кг. Сколько сбросите вы? Зависит от множества факторов: точки старта, состояния здоровья, физической формы, привычек и цели.' },

{ id: 7, categoryId: 1, question: 'Как осуществляется доставка?', answer: 'Мы привозим еду каждые два дня, в любые 2-х часовые интервалы: утром — с 6:00 до 13:00, вечером — с 18:00 до 23:59. Выбирайте, когда вам удобнее! Наши курьеры звонят, чтобы подтвердить доставку: — с 12:00 до 16:00, если вы ждете вечернюю доставку' },

{ id: 8, categoryId: 1, question: 'Как получить доставку день в день?', answer: 'Если вы закажете еду до 15:00, то мы привезем ваш заказ уже вечером. Как правило, количество коробок с едой ограничено.' },

{ id: 9, categoryId: 2, question: 'Как похудеть в определенной зоне, например, в талии/лице/ногах?', answer: 'Не существует рецепта локального похудения в отдельной проблемной зоне. Но вы можете направить организм в нужную сторону, если грамотно подобрать питание и физические упражнения.' },

{ id: 10, categoryId: 2, question: 'Можно ли раз в неделю есть вредную еду?', answer: 'По словам врача, даже во время диеты раз в неделю нужно устраивать читмил — то есть выделять день, когда есть можно все, даже самую калорийную пищу. С помощью читмила можно поддержать основной обмен веществ, который снижается у тех, кто вводит в свою жизнь дефицит калорий или другие ограничения.' },

{ id: 11, categoryId: 3, question: 'Нужно ли готовить BaskayTime после получения?', answer: 'Все, что вам нужно будет сделать - это поставить коробочку с едой на пару минут в микроволновую печь и обед будет готов.' },

{ id: 12, categoryId: 3, question: 'Смогу ли я питаться BaskayTime в разъездах или носить обеды с собой на работу?', answer: 'Наши контейнеры герметично упакованы и компактны. Обед от BaskayTime легко брать с собой и удобно есть даже сидя в пробке. Важно помнить, что в случае длительной поездки еду лучше перевозить в холодильной сумке.' },

{ id: 13, categoryId: 4, question: 'Можно ли есть BaskayTime, не разогревая?', answer: 'Все зависит от ваших вкусовых предпочтений. Вы можете съесть томатный суп и десерты без разогрева. Горячие блюда все же будут вкуснее и полезнее, если их предварительно разогреть.' },

],

maraphons: [

{ id: '0', current: true, vrewiew: 'firstPrewiev.jpg', video: 'firstVideo.mp4', condition: 'Стоимость 7 тыс.руб ,Питание для марафонцев 450 р,3.Продолжительность - 31 день', gift: '1 место - абонемент на год в Фитнес Центр К (всё включено) + питание от Баскай Тайм на месяц., 2 место 20 000 р, 3 место 10 000 р ' },

{ id: '1', current: false, vrewiew: 'secondPrewiev.jpg', video: 'winner(1).mp4', winner: 'Самвел. Приз - 50 тыс рублей!', secondPlace: ' Омар. Приз - питание из нашего меню', thirdPlace: 'Дзукаев Зантемир. Приз - питание из наешго меню', },

{ id: '2', current: false, vrewiew: 'thirdPrewiev.jpg', video: 'winner(2).mp4', winner: 'Фатима. Приз - 96 тыс рублей', secondPlace: 'Анджела.Приз 30 000 р ', thirdPlace: 'Тимур. Приз - 20 000 р', },

],

resultVideos: [

{ id: 0, name: 'result(1).mp4' },

{ id: 1, name: 'result(2).mp4' },

{ id: 2, name: 'result(3).mp4' },

{ id: 3, name: 'result(4).mp4' },

{ id: 4, name: 'result(5).mp4' },

{ id: 5, name: 'result(6).mp4' },

],

orders: [],

currentOrder: {},

ditailMaraphon: 'dsa',

currentMaraphon: null,

currentPage: 'home',

currentQestion: null,

currentFaq: 1,

currentSlide: 0,

currentArr: [],

updateArr: [],

currentUpdateProduct: 0,

currentPopUp: 0,

currentVideo: 0,

currentUser: null,

adminPage: 'menu',

currentAdminCategory: 'Завтрак',

editingProduct: null,

burgerMenu: false,

creatingProduct: null,

formLog: {

email: null,

password: null,

},

formReg: {

email: null,

password: null,

login: null,

dbPassword: null,

},

user: null,

errors: [],

emptyError: false,

currentMar: {},

adminMaraphons: [],

adminNumbers: [],

number: {},

}

},

methods: {

addToAdminMar() {

this.adminMaraphons.push({ ...this.currentMar })

this.currentPopUp = 5

console.log(this.adminMaraphons, this.currentMar);

this.currentMar.client = '';

this.currentMar.address = '';

this.currentMar.phone = '';

},

async render() {

this.breakfasts = await post('http://pup/public/api/products/get', { id: 1 })

this.secondBreakfasts = await post('http://pup/public/api/products/get', { id: 2 })

this.lunchs = await post('http://pup/public/api/products/get', { id: 3 })

this.snacks = await post('http://pup/public/api/products/get', { id: 4 })

this.dinners = await post('http://pup/public/api/products/get', { id: 5 })

this.waters = await post('http://pup/public/api/products/get', { id: 6 })

},

addToOrders() {

this.orders.push({ ...this.currentOrder, product: [...this.currentArr] })

this.currentOrder.phone = '';

this.currentOrder.address = '';

this.currentOrder.client = '';

this.currentPopUp = 6;

},

postProduct() {

post('http://pup/public/api/products/create', this.creatingProduct)

this.creatingProduct = null;

this.render();

this.currentPopUp = 7;

},

createProduct() {

this.creatingProduct = {};

},

confid() {

this.currentPage = 'confid';

this.currentPopUp = 0;

},

editProduct(product) {

this.editingProduct = product;

},

postChanges() {

post('http://pup/public/api/products/update', this.editingProduct);

this.editingProduct = null;

this.render()

this.currentPopUp = 8;

},

removeProduct(product) {

post('http://pup/public/api/products/delete', product);

this.render()

this.currentPopUp = 9;

},

goDetail(el) {

this.currentPage = 'detail'

this.ditailMaraphon = el

console.log(this.ditailMaraphon);

},

changeProduct(product) {

this.updateArr = [];

if (product.category\_id == '1') {

this.breakfasts.forEach(element => {

this.updateArr.push(element)

});

}

if (product.category\_id == '2') {

this.secondBreakfasts.forEach(element => {

this.updateArr.push(element)

});

}

if (product.category\_id == '3') {

this.lunchs.forEach(element => {

this.updateArr.push(element)

});

}

if (product.category\_id == '4') {

this.snacks.forEach(element => {

this.updateArr.push(element)

});

}

if (product.category\_id == '5') {

this.dinners.forEach(element => {

this.updateArr.push(element)

});

}

if (product.category\_id == '6') {

this.waters.forEach(element => {

this.updateArr.push(element)

});

}

},

updateProduct(product) {

this.updateArr = [];

let index = this.currentArr.findIndex(el => el.category\_id === product.category\_id);

this.currentArr.splice(index, 1, product);

this.currentUpdateProduct = 0;

},

prevSlide() {

if (this.currentSlide == 0) this.currentSlide = 0;

else this.currentSlide--;

},

nextSlide() {

if (this.clients.length - 1 <= this.currentSlide) this.currentSlide = this.currentSlide;

else this.currentSlide++;

},

prevVideo() {

if (this.currentVideo == 0) this.currentVideo = 0;

else this.currentVideo--;

},

nextVideo() {

if (this.resultVideos.length - 1 <= this.currentVideo) this.currentVideo = this.currentVideo;

else this.currentVideo++;

},

prevProduct() {

if (this.currentUpdateProduct == 0) this.currentUpdateProduct = 0;

else this.currentUpdateProduct--;

},

nextProduct() {

if (Math.ceil(this.updateArr.length / 3) === this.currentUpdateProduct + 1) this.currentUpdateProduct = this.currentUpdateProduct;

else this.currentUpdateProduct++;

console.log(Math.floor(this.updateArr.length / 3));

console.log(this.currentUpdateProduct);

},

randomIndex(array) {

const index = Math.floor(Math.random() \* (array.length - 1))

return index

},

sendNumber() {

this.adminNumbers.push(this.number)

this.number = '';

this.currentPopUp = 5;

console.log(this.adminNumbers);

},

async logIn() {

const user = await post('http://pup/public/api/login', this.formLog)

if (user.message) {

this.errors = user.message;

return

}

this.user = user

this.currentPopUp = 0;

this.formLog.email = ''

this.formLog.password = ''

this.errors = []

},

async registration() {

if (this.formReg.password !== this.formReg.dbPassword) {

this.errors['password'] = ['Пароли не совпадают']

console.log(this.errors);

return

}

const user = await post('http://pup/public/api/registration', this.formReg)

if (user.message) {

this.errors = user.message;

return

}

this.user = user

this.currentPopUp = 0;

this.errors = []

this.formReg.email = '';

this.formReg.password = '';

this.formReg.login = '';

this.formReg.dbPassword = '';

},

},

computed: {

isAuth() {

return this.user

},

isAdmin() {

return this.user?.role === 'admin'

},

adminMenu() {

let adminMenu = [];

if (this.currentAdminCategory === 'Завтрак') {

this.breakfasts.forEach(el => {

adminMenu.push(el)

})

}

if (this.currentAdminCategory === '2-ой завтрак') {

this.secondBreakfasts.forEach(el => {

adminMenu.push(el)

})

}

if (this.currentAdminCategory === 'Обед') {

this.lunchs.forEach(el => {

adminMenu.push(el)

})

}

if (this.currentAdminCategory === 'Полдник') {

this.snacks.forEach(el => {

adminMenu.push(el)

})

}

if (this.currentAdminCategory === 'Ужин') {

this.dinners.forEach(el => {

adminMenu.push(el)

})

}

if (this.currentAdminCategory === 'Напитки') {

this.waters.forEach(el => {

adminMenu.push(el)

})

}

return adminMenu;

},

otherMaraphons() {

return this.maraphons.filter(el => !el.current)

},

splitCondition() {

return this.currentMaraphon.condition.split(',')

},

splitGift() {

return this.currentMaraphon.gift.split(',')

},

filtredFaq() {

let filtredFaq = this.faq.filter(item => item.categoryId == this.currentFaq);

return filtredFaq;

},

menu() {

if (this.currentCategory === 'Завтрак') {

this.currentArr = [

this.breakfasts[this.randomIndex(this.breakfasts)],

this.secondBreakfasts[this.randomIndex(this.secondBreakfasts)]

]

}

if (this.currentCategory === 'Основное') {

this.currentArr = [

this.breakfasts[this.randomIndex(this.breakfasts)],

this.lunchs[this.randomIndex(this.lunchs)],

this.dinners[this.randomIndex(this.dinners)]

]

}

if (this.currentCategory === 'Доплнительное') {

this.currentArr = [

this.breakfasts[this.randomIndex(this.breakfasts)],

this.secondBreakfasts[this.randomIndex(this.secondBreakfasts)],

this.lunchs[this.randomIndex(this.lunchs)],

this.dinners[this.randomIndex(this.dinners)]

]

}

if (this.currentCategory === 'Полное') {

this.currentArr = [

this.breakfasts[this.randomIndex(this.breakfasts)],

this.secondBreakfasts[this.randomIndex(this.secondBreakfasts)],

this.lunchs[this.randomIndex(this.lunchs)],

this.snacks[this.randomIndex(this.snacks)],

this.dinners[this.randomIndex(this.dinners)]

]

}

if (this.currentCategory === 'Расширенное') {

this.currentArr = [

this.breakfasts[this.randomIndex(this.breakfasts)],

this.secondBreakfasts[this.randomIndex(this.secondBreakfasts)],

this.lunchs[this.randomIndex(this.lunchs)],

this.dinners[this.randomIndex(this.dinners)],

this.snacks[this.randomIndex(this.snacks)],

this.waters[this.randomIndex(this.waters)]

]

}

},

totalCalories() {

let calories = 0;

let proteins = 0;

let fats = 0;

let carbohdrate = 0;

let totalCalories = [];

if (this.currentArr.length) {

this.currentArr.forEach(el => {

proteins += el.proteins;

calories = calories + el.calories;

fats += el.fats;

carbohdrate += el.carbohdrate;

})

totalCalories.push({ 'calories': calories, 'proteins': proteins, 'fats': fats, 'carbohdrate': carbohdrate })

return totalCalories

}

return []

},

totalSum() {

let totalSum = 0;

this.currentArr.forEach(el => {

totalSum = totalSum + el.price

})

console.log(totalSum);

return totalSum;

},

},

components: {

product

},

async created() {

this.categoriesProducts = await post('http://pup/public/api/categories/all', {})

this.breakfasts = await post('http://pup/public/api/products/get', { id: 1 })

this.secondBreakfasts = await post('http://pup/public/api/products/get', { id: 2 })

this.lunchs = await post('http://pup/public/api/products/get', { id: 3 })

this.snacks = await post('http://pup/public/api/products/get', { id: 4 })

this.dinners = await post('http://pup/public/api/products/get', { id: 5 })

this.waters = await post('http://pup/public/api/products/get', { id: 6 })

this.menu

this.totalCalories

this.currentMaraphon = this.maraphons.find(el => el.current)

this.adminMenu;

},

}

Vue.createApp(App).mount('#app')

.