

Поиск

Все дополнения 218

Каталог, Поиск, Фильтрация 19

Корзина, Заказы 26

Импорт, Экспорт 9

Платежные системы 25

Пользователи 17

Работа с текстом 13

Работа с фото и файлами 14

Оповещения, Рассылки 14

Утилиты 53

Готовые сайты, темы 11

Другое 20

+ Добавить дополнение

Маркетплейс дополнений для MODX



Маркетплейс готовых дополнений для CMS MODX Revolution от лучших русскоговорящих разработчиков.

Цифры и факты

Дополнений	218
Пользователей	9 194
Загрузок	139 278
Ключей	2 493
Авторов	42

Подпишитесь на рассылку!

Электронная почта

Подписаться

Мы не спамим и отправляем письма только по обновлению Modstore.pro и Modhost.pro не чаще 1 раза в месяц.

Новая версия modstore.pro



Возможность работы на тестовом и "боевом" доменах



Подробная документация



Техническая поддержка напрямую от авторов дополнений

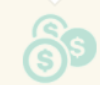


Крупнейший поставщик дополнений для MODX

Схема работы маркетплейса Скрыть

Бесплатное демо дополнений на тестовом тарифе modhost.pro

Возможность задавать вопросы по купленному дополнению в Технической поддержке 1 год.



Платные дополнения



Авторизация в ЛК Modstore



Создание ключа для 1 сайта в ЛК



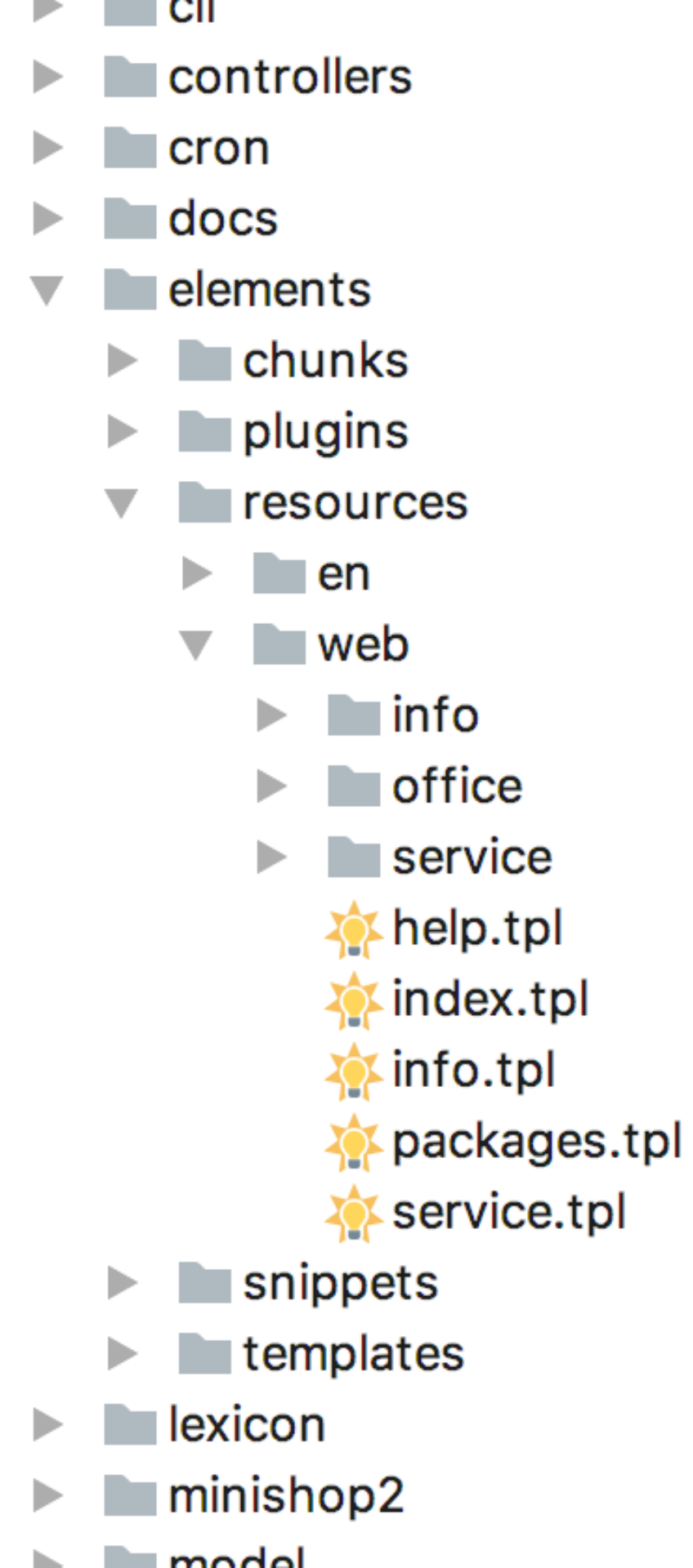
Покупка дополнения для ключа

Основные задачи

- Обновление вёрстки, добавление адаптивности
- Англоязычный раздел магазина
- Улучшение личного кабинета, особенно раздела поддержки и управления ключами
- Рефакторинг кода репозитория, так как он использовал устаревший `modRestServer`
- Перенос администраторского раздела на фронтенд

Для разработки используется свой транспортный пакет **Extras**, который:

- Устанавливает все шаблоны, сниппеты, чанки и ресурсы
- Добавляет\изменяет системные настройки, источник файлов
- Устанавливает свой обработчик заказов miniShop2 и контроллеры личного кабинета
- Копирует и обновляет скрипты, стили и изображения
- Позволяет удобно работать со всем этим из IDE PhpStorm и позволяет использовать контроль версий



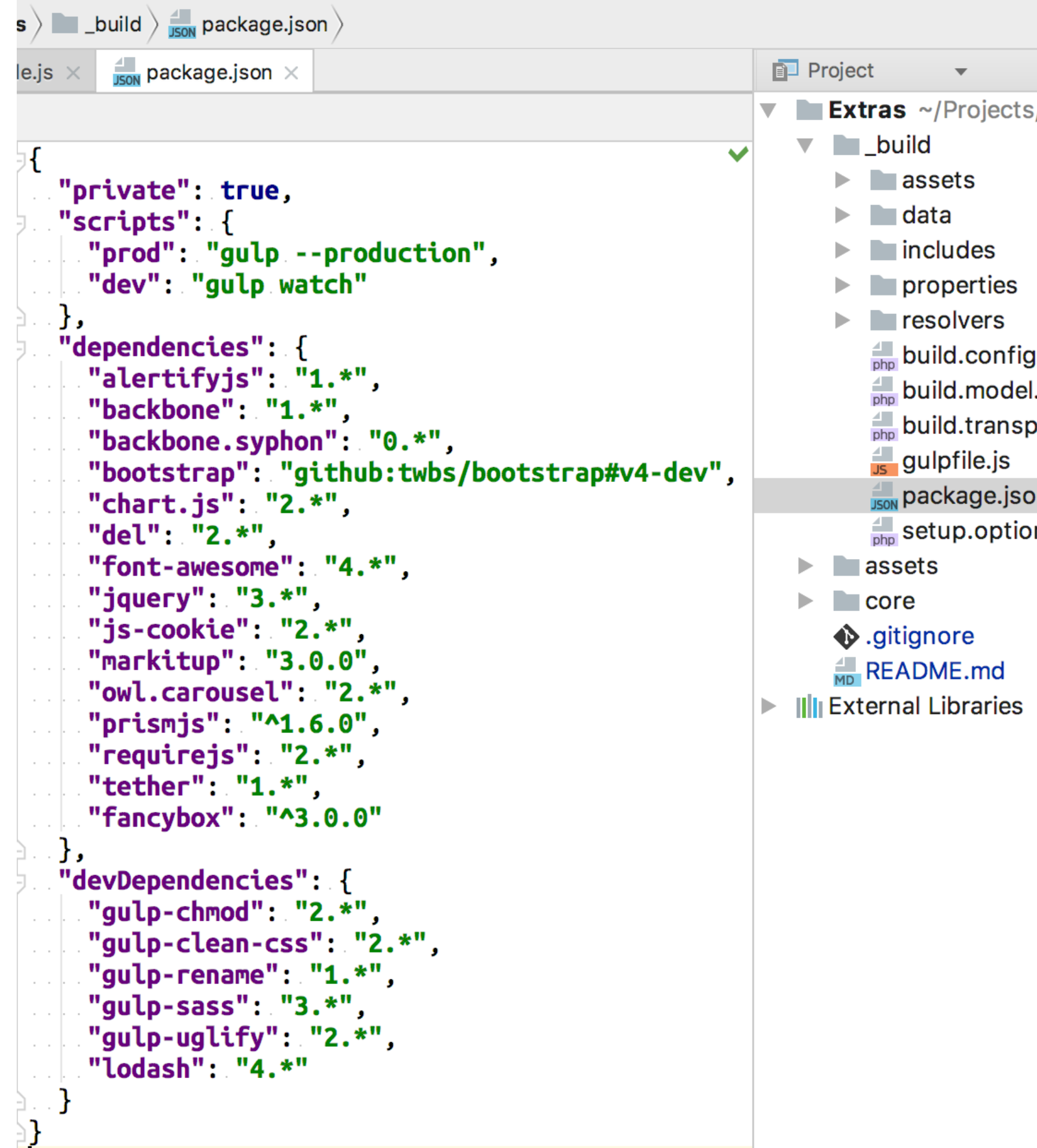
Используемые дополнения

- miniShop2 + платёжные методы для реализации магазина
- Office + HybridAuth для авторизации и редактирования профиля
- mSearch2 для поиска и фильтрации дополнений
- Tickets для раздела поддержки
- pdoTools для поддержки работы всех этих дополнений и шаблонизатора Fenom

Фронтенд

- Полный отказ от MinifyX
- Сборка и минификация файлов через Gulp
- Использование RequireJS для разделения логики скриптов сайта на небольшие модули

```
requirejs(["app/orders"],  
    function(App) {  
        App.Orders.init()  
    }  
);
```



- Скрипты дополнений копируются и минифицируются отдельной командой Gulp, чтобы их можно было подключать как часть основного приложения.
- Самостоятельная инициализация скриптов самими дополнениями **отключена**, никакого js в шапке страницы.
- Когда нужно - эти скрипты вызываются из основного приложения, с предварительным изменением методов.

```
requirejs.config({
  baseUrl: '/assets/components/extras/js/web/',
  urlArgs: 'v=' + document.head.querySelector('meta[name="assets-version"]').content,
  waitSeconds: 30,
  paths: {
    jquery: 'lib/jquery.min',
    bootstrap: 'lib/bootstrap.min',
    tether: 'lib/tether.req',
    backbone: 'lib/backbone.min',
    backbone_syphon: 'lib/backbone.syphon.min',
    underscore: 'lib/underscore.min',
    alertify: 'lib/alertify.min',
    jquery_owl: '/assets/components/extras/js/web/lib/owl.carousel.min',
    cookies: '/assets/components/extras/js/web/lib/js.cookie.min',
    prism: '/assets/components/extras/js/web/lib/prism.min',
    markitup: '/assets/components/extras/js/web/lib/markitup.min',
    chart: '/assets/components/extras/js/web/lib/chart.min',
    fancybox: '/assets/components/extras/js/web/lib/jquery.fancybox.min',
    minishop2: '/assets/components/extras/js/web/lib/minishop2.min',
    msearch2: '/assets/components/extras/js/web/lib/msearch2.min',
    jqui: '/assets/components/extras/js/web/lib/jquery-ui.min',
    pdopage: '/assets/components/extras/js/web/lib/pdopage.min',
  },
  shim: {
    bootstrap: {
      deps: ['jquery', 'tether']
    },
    underscore: {
      exports: '_'
    },
    backbone: {
      deps: ['underscore', 'jquery'],
      exports: 'Backbone'
    },
    alertify: {
      exports: 'alertify'
    },
    minishop2: {
      deps: ['jquery'],
      exports: 'miniShop2'
    },
    msearch2: {
      deps: ['jquery'],
      exports: 'mSearch2'
    },
  },
});
```

Например, инициализация miniShop2:

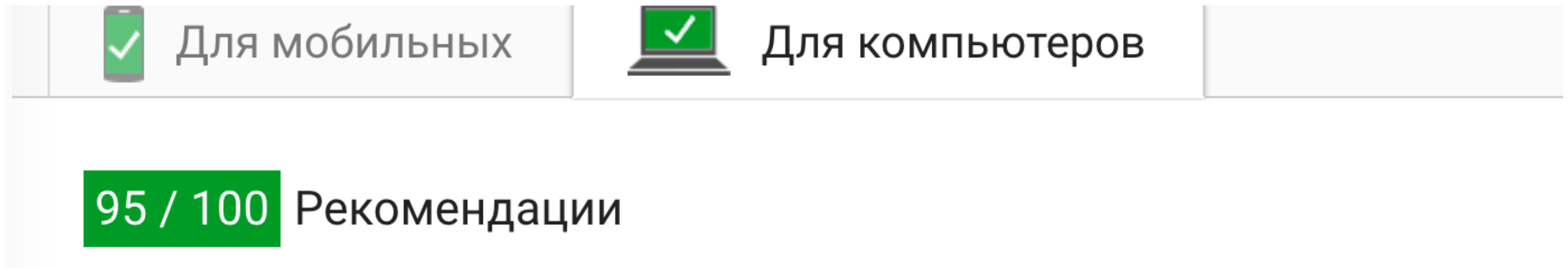
```
initMiniShop2: function () {
    miniShop2.Message = App.utils.Message;

    miniShop2.Cart.status = function (status) {
        var $miniCart = $(miniShop2.Cart.miniCart);
        if (status['total_count'] < 1) {
            $(App.Order.elem).modal('hide');
            $miniCart.removeClass(miniShop2.Cart.miniCartNotEmptyClass);
        }
        else {
            if (status['total_count'] > 0 && !$miniCart.hasClass(miniShop2.Cart.miniCartNotEmptyClass)) {
                $miniCart.addClass(miniShop2.Cart.miniCartNotEmptyClass);
            }
            $(miniShop2.Cart.totalCount).text(status['total_count']);
            if ($(miniShop2.Order.orderCost, miniShop2.Order.order).length) {
                miniShop2.Order.getcost();
            }
        }
    };
};
```

** обновления компонента ничего не сломают*

Таким образом:

- Страницы грузятся максимально быстро



- Лишний код не выполняется
- Полный контроль над javascript логикой работы сторонних дополнений - ведь можно изменить почти любой метод

А вот так эти модули выглядят в коде страницы:

```
<script type="text/javascript">requirejs(["app", "app/office"]);</script>
  <div class="modal fade" id="order">
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header">
          <h4 class="modal-title">Загрузка...</h4>
          <button type="button" class="close" data-dismiss="modal">
            <span aria-hidden="true">&times;</span>
          </button>
        </div>
        <div class="modal-body">
          <i class="fa fa-cog fa-spin"></i>
        </div>
      </div>
    </div>
  </div>
</div>

<script type="text/javascript">requirejs(["app", "app/order"]);</script>
<script type="text/javascript">requirejs(["app", "app/remote-comments"], function(App) {App.RemoteComments()});</script>
<script type="text/javascript">requirejs(["app", "app/index"]);</script>
</body>
</html>
```

* каждый скрипт грузится только один раз



Search

All packages 29

Catalog, Search, Filters 4

Cart, Orders 1

Import, Export 0

Payments systems 0

Users 4

Working with text 3

Working with photos and files 4

Alerts, Mailing 0

Utilities 11

Sites packages, themes 2

Other 2

+ Submit a new package

Marketplace for MODX



Marketplace of extras for MODX Revolution from the best Russian-speaking developers.

Integrated with the Modhost.

Numbers and facts

Extras	218
Users	9 194
Downloads	139 278
Keys	22 413
Authors	42

Subscribe to the newsletter!

Email

Subscribe

We do not spam and send emails with updates of a Modstore.pro and Modhost.pro no more than once per month.

Английская версия



Ability to work with a test and production domains



Detailed documentation



Technical support directly from the extras authors



The biggest supplier of extras for MODX

The marketplace scheme Hide

Free test of selected extras on the test plan of modhost.pro



The paid extras



Authorize in the Modstore



Create a key for a site



Buy extras for created key

The ability to ask questions on purchased extras at technical support for 1 year.

Товары

- 2 контекста, в каждом свои товары miniShop2, связанные с пакетами репозитория через поле *package*.
- Авторы могут редактировать их отдельно в личном кабинете, включая и выключая по желанию.
- Расширенный класс заказа проверяет ключи и покупаемые дополнения при оформлении.

pdoTools

СохранитьКопировать

Товар

Группы ресурсов

Комментарии

Документ

Настройки

Свойства товара

Связи

Категории

Цена:

Старая цена:

0

Дополнение:

pdoTools

yTranslit

YandexMaps

YaCaptcha

xPoller2

xParser

xButtons

Комиссия магазина:

0

Демо-версия:

☐ Новый

☒ Особый

Лексиконы - хорошо, но неудобно

- Нужно отдельно писать их в чанках и словарях
- Нужно следить за внутренней логикой наименования и помнить, какие строки уже есть и как используются
- И главное - нужно чистить кэш после добавления или изменения текста, что делает сложной работу через IDE.

Поэтому было принято решение использовать лексиконы только внутри PHP и Javascript.

Аксессор \$.en

Fenom позволяет добавлять добавлять методы из плагинов на событие pdoToolsOnFenomInit:

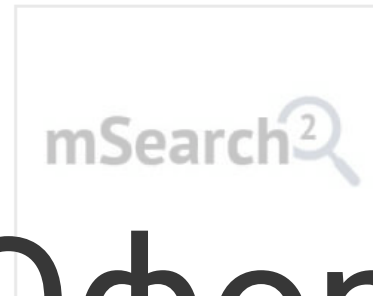
```
$fenom->addAccessorSmart('en', 'en', Fenom::ACCESSOR_PROPERTY);
```

После чего их можно использовать в оформлении:

```
{if $.en}  
    Long english text  
{else}  
    Длинный русский текст  
{/if}
```

Поступила оплата заказа №1704/111

в магазине Modstore.pro

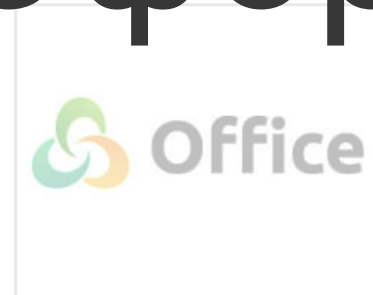


mSearch2

1 990 руб.

Морфологический поиск и фильтрация данных

скидка: 0%



Office

1 990 руб.

Универсальный личный кабинет покупателя

скидка: 0%

Оформление писем

Итого

3 980 руб.

Подпишись на нас



MODSTORE.PRO

- Все письма наследуются от одного шаблона через Fenom
- Для удобной разметки используются CSS классы

```
<style>
  body > table { width: 600px; margin: auto; }
  p { font-size: 16px; line-height: 22px; }
</style>
```

Проблема в том, что Gmail не читает такие стили - он требует оформлять всё inline:

```
<p style="font-size: 16px; line-height: 22px;">
  My paragraph
</p>
```

В packagist есть замечательная библиотека для интеграции стилей inline в подходящие теги - [pelago/emogrifier](#).

Но как заставить её обрабатывать **все** письма сайта?

Оказывается, очень просто:

```
$modx->getService('mail', 'extraMail',  
    MODX_CORE_PATH . 'components/extras/model/'  
);
```

Фокус в том, чтобы сделать это вперёд всех, например плагином на `OnMODXInit`, и тогда `modx::getService()` всегда будет загружать именно наш, расширенный, класс работы с почтой.

Дальше дело техники:

```
class extraMail extends modPHPMailer
{
    public function set($key, $value)
    {
        if ($key == modMail::MAIL_BODY) {
            // Вот она - интеграция стилей в теги
            $emogrifier = new \Pelago\Emogrifier($value);
            $value = $emogrifier->emogrify();
        }

        parent::set($key, $value);
    }
}
```

И теперь мы можем удобно оформлять письма, не прописывая вручную стили в каждый тег.



Поиск

Все дополнения 218

Каталог, Поиск, Фильтрация 19

Корзина, Заказы 26

Импорт, Экспорт 9

Платежные системы 25

Пользователи 17

Работа с текстом 13

Работа с фото и файлами 14

Оповещения, Рассылки 14

Утилиты 53

Готовые сайты, темы 11

Другое 20



Василий Наумкин
bezumkin@ya.ru

Настройки профиля

Ключи Поддержка Покупки Избранное Дополнения Продажи

+ Добавить новое дополнение

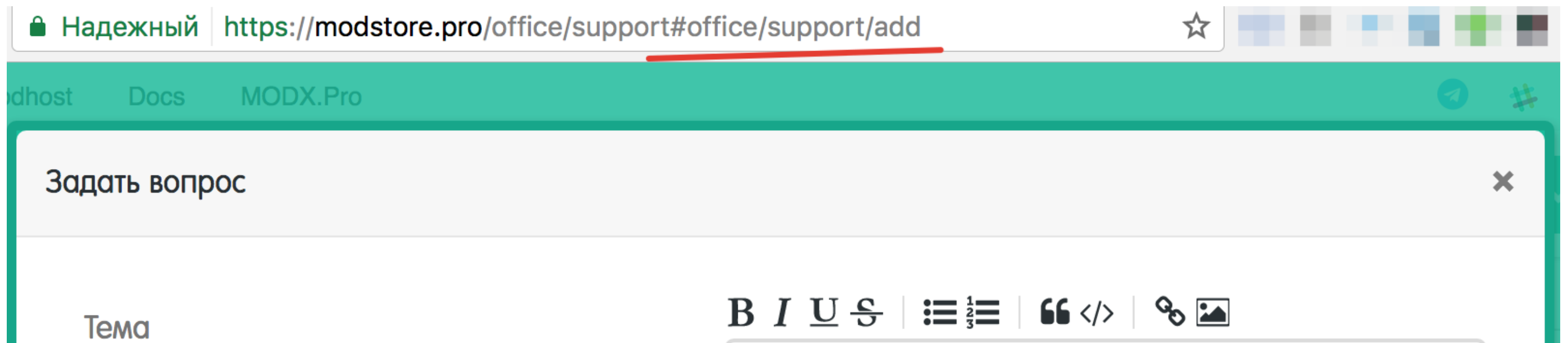
Агентский договор

Поиск

Дополнение	Загрузки	Версия	Стоимость	
HybridAuth	4 571	2.0.4-pl 10 апреля 2017, 12:06	-	
miniShop2	9 614	2.4.10-pl 05 апреля 2017, 14:06	-	
pdoTools	27 545	2.8.4-pl1 02 апреля 2017, 11:52	-	
mSearch2	6 003	1.8.4-pl 24 марта 2017, 05:45	1 990 ₺	
Office	2 571	1.4.5-pl 09 марта 2017, 09:39	1 990 ₺	
Tickets	10 048	1.8.1-pl 01 марта 2017, 14:40	-	

Основные особенности

- Единый коннектор для всех запросов
- Проверка CSRF токена
- Прямые ссылки на все действия



Необычный функционал

- Возможность менять метод оплаты у новых заказов
- Удаление неоплаченных заказов
- Почтовые уведомления авторов о покупке их дополнений
- Избранное и добавление в корзину прямо из него, с выбором ключа
- Тех.поддержка с новой версией MarkItUp и подсветкой синтаксиса Prism

Спасибо за внимание!