

Стандарты интеграции сайтов

1. Создание нового сайта:

1. Новый сайт создается отдельно, а не из Демо-сайта.
2. Пользователь должен быть перемещен из Демо-сайта в новосозданный.
3. Демо сайт должен быть удален.
4. Все существующие константы однократно должны быть помещены в отдельную группу «Конфигурация».
5. Качество jpg и png должно быть выставлено на максимум, PNG_QUALITY = 0, JPG_QUALITY = 100.
6. Модули «Типограф» и «Услуги раскрутки сайта» должны быть отключены.
7. Индексация сайта на время работ должна быть запрещена.

2. Интеграция:

1. Файлы:

1. Все файлы верстки, кроме файлов самих страниц (html или htm), должны быть перемещены в корневую папку сайта без изменений. В случае, если в верстке содержатся папки с именами «admin» или «modules» их нужно переименовать.
2. Запрещено вносить изменения в файлы верстки. Все изменения вносятся путем обновления верстки верстальщиком или путем копирования исходных файлов с их дальнейшим переименованием в случае, если связаться с верстальщиком нет возможности.

2. Структура:

1. Должно существовать меню с названием «Нет» для пунктов, которые нужно выводить в карте сайта, но не нужно выводить ни в одном меню.
2. Раздел интернет-магазина должен иметь путь «shop», а корзина «shop/cart», если иное не регламентировано техническим заданием.
3. Помимо основных элементов структуры должны быть добавлены:
 1. Поиск, если поддерживается редакцией
 2. Google sitemap с путем «sitemap».
 3. Карта сайта с путем «map».
 4. Страница ошибки 404. Должна быть привязана к сайту.
4. В карте сайта не должно быть ссылок на страницы: Корзины, Карты сайта, Google Sitemap, 404 ошибки, Сайт отключен.

3. Макеты:

1. Всегда существуют 2 макета – «Основной макет» и макет «Главная», вложенный в первый.
2. Шапка должна быть приведена к виду шапки сайта по-умолчанию(демо-сайта), скрипты и стили верстки должны остаться подключены.
3. Строку с вызовом метода «showCss()» необходимо подключать после подключения всех стилей верстки.
4. Все недочеты верстки при интеграции исправляются перекрывающими стилями в CSS основного макета или же с помощью обновления верстки, полученной от верстальщика.
5. Макетов должно быть как можно меньше, но в них не должно содержаться множество условий, привязанных к структуре.
6. Статичные элементы макета – телефон, адрес, социальные иконки и другие должны быть вынесены в «Страницы и документы». В случае, если элемент однострочный и не содержит html-тегов, он должен быть вынесен в

константы.

7. В футере год должен изменяться на диапазон годов автоматически.
8. Все динамические блоки в макетах выводятся с помощью класса «Kad_Template».

4. XSL-шаблоны:

1. Все XSL- шаблоны, отличные от исходных, должны быть скопированы и именованы с префиксом из двух символов, полученным из домена сайта через «_», а также помещены в отдельную группу с названием домена сайта.
2. XSL шаблоны, предназначенные для использования в других XSL шаблонах, содержащие универсальный подшаблон, должны именоваться с дополнительным префиксом «_»
3. Исходные XSL-шаблоны должны оставаться без изменений.
4. Допустимо использование стандартных шаблонов при настройке структуры с ТДС.
5. Для картинок всех динамических элементов должен быть проставлен атрибут «alt», содержащий текст «Картинка для {name}», где «{name}» будет заменяться названием элемента.
6. Все xsl:include должен быть в самом начале XSL шаблона после закрытия тега xsl:stylesheet

5. ТДС:

1. Все ТДС, отличные от исходных, должны быть скопированы и именованы с постфиксом из двух символов, полученным из домена сайта, через пробел.
2. В случае, если требуется вывести статичный документ, структуру или динамический блок, формируемый отдельным XSL-шаблоном, в середине динамической страницы, то делается это путем добавления сущности в ТДС в контроллер показа текста документа и выводом этой сущности в XSL-шаблоне, привязанном к текущей структуре.

6. Страницы и документы:

1. Если не указан текст, то текст документа должен содержать “Текст страницы “[название_страницы]”.
2. Если требуется использовать несколько документов на одной странице, то их необходимо помещать в раздел с названием страницы.
3. Все блоки макета, которые помещены в документы, необходимо размещать в корневом разделе.

7. Константы:

1. Все константы должны быть в верхнем регистре.

8. Недопустимо:

1. Использование сторонних файлов и библиотек, находящихся за пределами папки «modules», а также в корне папки «modules». Такие библиотеки должны быть разработаны строго в соответствии с API HostCMS и размещаться в отдельной папке папки modules.
2. Изменение исходных файлов системы HostCMS, за исключением файла «bootstrap.php», а также конфигурационных файлов «config.php» и файлов находящихся в папке «config», кроме случаев, заранее оговоренных с заказчиком, частично или полностью исключающих дальнейшее обновление системы и техническую поддержку HostCMS.
3. Если требуется отметить доп. свойство для отдельной выборки, например для отображения в списке товаров или на странице сравнения товаров, то делается это с помощью разделов дополнительных свойств или с помощью префиксов для tag_name доп. свойства. Например, “list.property1”. Можно использовать несколько префиксов, например, “list.compare.property1”.

4. Дополнительные свойства списком, например на странице товара, должны выводиться [через property](#), а не через `property_value`.
5. Для любого меню, в том числе и в меню из разделов каталога, активный пункт должен быть подсвечен(выделен жирным), если иное не прописано в ТЗ.
6. Для всех информационных систем и интернет-магазинов, которые выводятся в клиентской части должен быть выбран тип формирования URI - Транслитерация.
7. Вся валидация должна быть реализована не только в клиентской части, но и в серверной.
8. Любой PHP-код должен быть написан и прокомментирован в соответствии со [стандартами оформления кода KAD :: Systems](#)