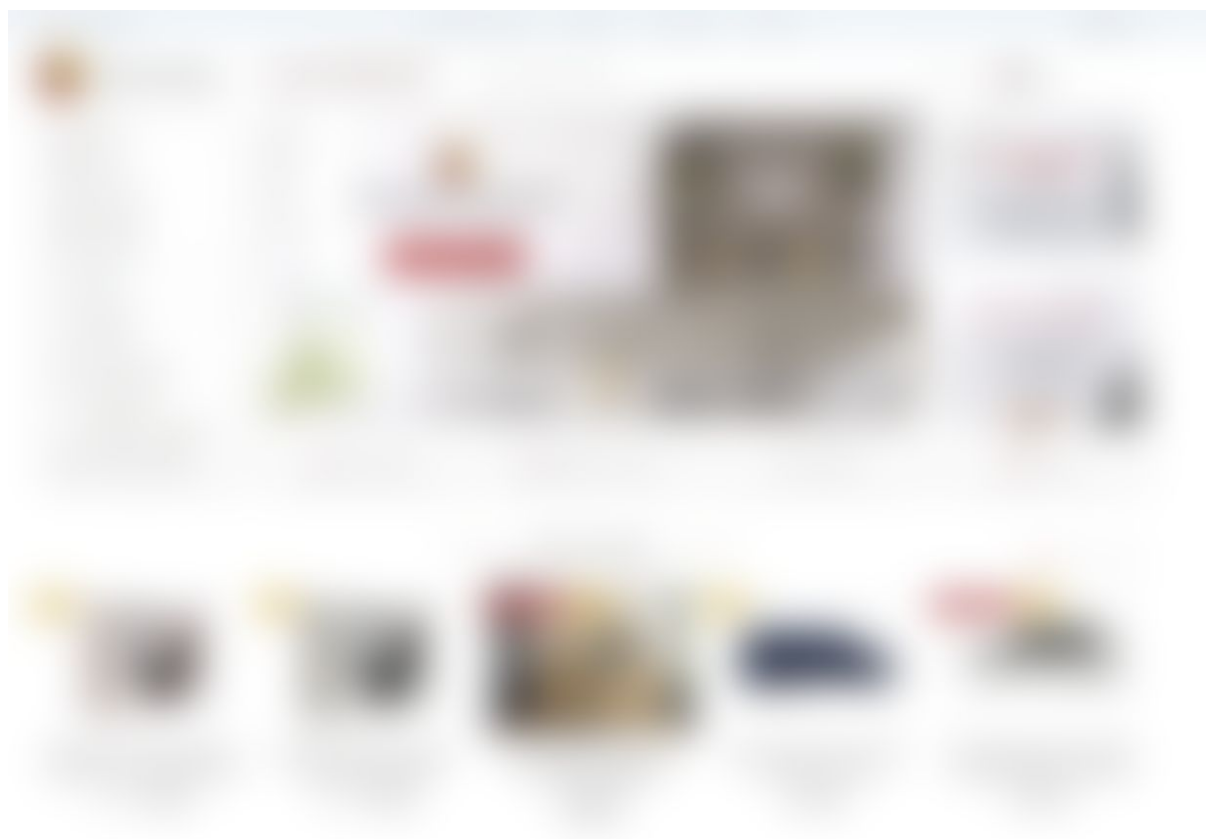


Технический аудит сайта по продаже мебели



Содержание

Введение	2
1. Возможности Битрикс по сжатию CSS, JS	2
2. Автокеширование компонентов	3
3. Технология АвтоКомпозит	3
4. Кеширование при установленном фильтре	4
5. Фасетный индекс	5
6. Таблица пользователей корзины	6
7. Товары в разделах каталога	7
8. Скрипт фильтрации городов	8
9. Технология CDN	9
10. Проверка использования компонентов	9
11. Проверка оптимальности настройки сервера	10

Введение

Целью технического аудита является выявление проблем и ошибок, влияющих на скорость загрузки страниц сайта.

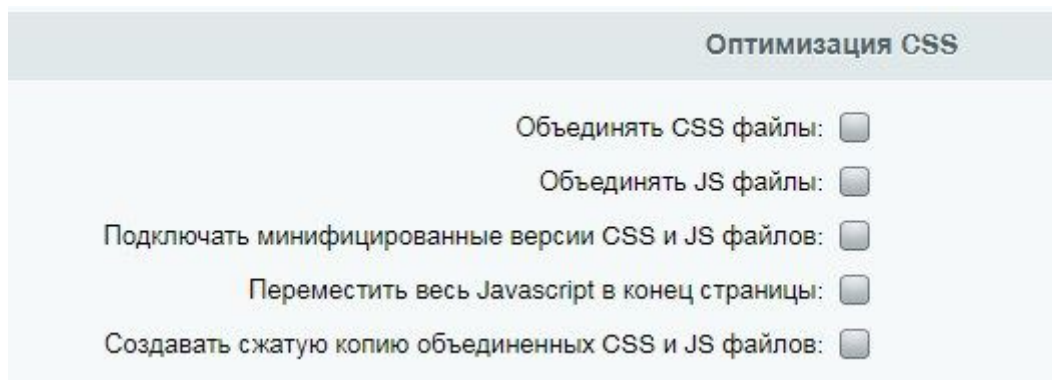
В ходе аудита были проведены следующие работы:

- Проверка соответствия выполнения требований CMS 1С-Битрикс, предъявляемых к серверу сайта;
- Проверка количества запросов к базе данных при загрузке страницы;
- Проверка оптимальности исполнения скриптов.

В ходе технического аудита сайта [REDACTED] было выявлено:

1. Возможности Битрикс по сжатию CSS, JS

Не используются возможности Битрикс по сжатию CSS, JS, перемещению кода вниз страницы.



The screenshot shows the 'Оптимизация CSS' (CSS Optimization) settings in Bitrix. It contains five checkboxes, all of which are currently unchecked:

- Объединять CSS файлы:
- Объединять JS файлы:
- Подключать минифицированные версии CSS и JS файлов:
- Переместить весь Javascript в конец страницы:
- Создавать сжатую копию объединенных CSS и JS файлов:

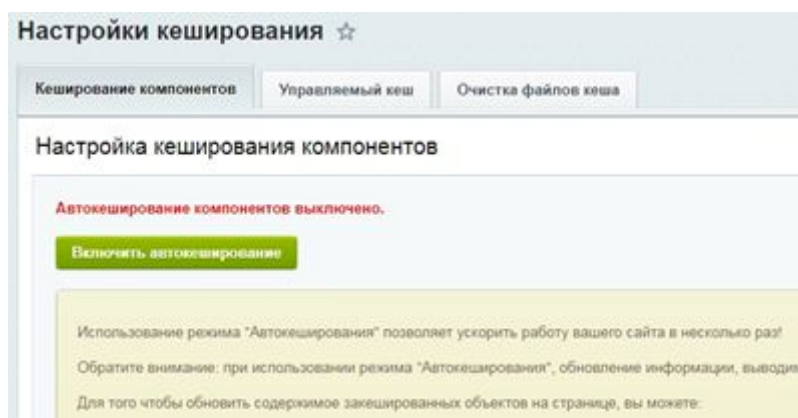
Для включения необходимо активировать функцию в настройках главного модуля. Включение сжатия может привести к ошибкам на сайте.

Рабочий стол – Настройки – Настройки продукта – Настройки модулей – Главный модуль.

[REDACTED]

2. Автокеширование компонентов

Отключено автокеширование компонентов.



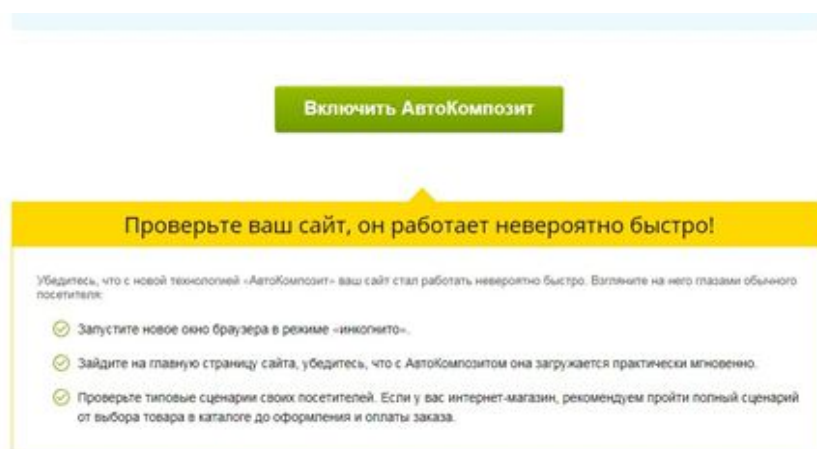
Использование режима «Автокеширования» позволяет ускорить работу сайта. Стоит уточнить у разработчиков о причинах его отключения, обычно этот режим рекомендуется использовать всегда.

Рабочий стол – Настройки – Настройки продукта – Автокеширование.



3. Технология АвтоКомпозит

Отключена технология АвтоКомпозит.



Использование технологии «АвтоКомпозит» позволяет ускорить работу сайта. Рекомендуется использовать всегда, но у разработчиков могли быть причины его отключить. Очень часто при наличии интеграции с 1С «АвтоКомпозит» приводит к случаям, когда цены на сайте не обновляются после выгрузки из 1С, требуя сбросить кеш сайта вручную.

Рабочий стол – Настройки – Настройки продукта – Композитный сайт – Настройки.

[https://studia-jl.ru/ru/settings/product/composite-site/settings](#)

4. Кеширование при установленном фильтре

Отключено кеширование при установленном фильтре. Фильтр установлен в разделах каталога [https://studia-jl.ru/ru/catalog/section](#).

На странице [https://studia-jl.ru/ru/catalog/section](#) в URL передается параметр «TYPE». В данном случае TYPE=gostinaya. Исходя из этого параметра, собирается предустановленный фильтр, который передаётся в компонент для вывода товаров (catalog.section). В компоненте catalog.section установлен параметр «Кешировать» при установленном фильтре – «Нет». Поэтому на страницах вида

[https://studia-jl.ru/ru/catalog/section?TYPE=gostinaya](#) catalog.section всегда отрабатывает без кеша, в отличие, например, от страниц разделов

[https://studia-jl.ru/ru/catalog/section?TYPE=gostinaya&filter=1](#) где результат работы компонента кешируется, и при повторной отдаче страницы делается меньше запросов в БД.

В использовании параметра «Кешировать при установленном фильтре» обычно применяется две стратегии:



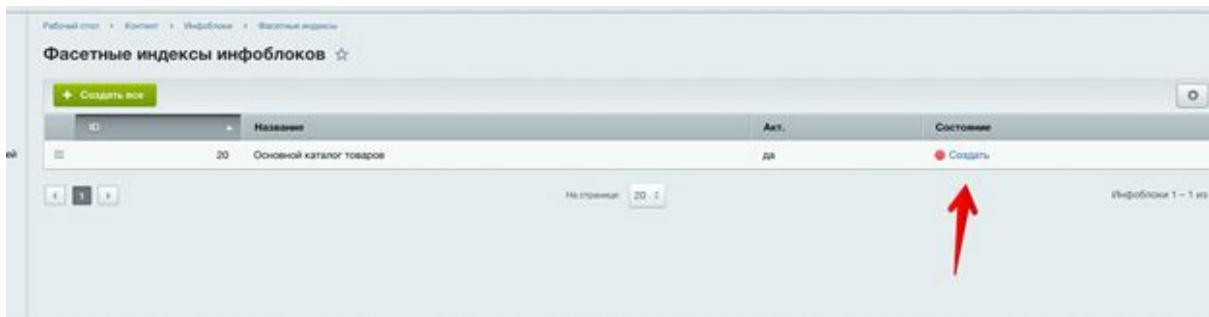
- Если на странице установлен «умный фильтр», то параметр «Кешировать при установленном фильтре» приравнивают к значению «Нет». Кеширование в данном случае не очень эффективно, так как количество комбинаций свойств, которые может выбрать пользователь, большое, а вероятность того, что другой пользователь выберет такую же комбинацию и попадёт в ранее сгенерированный кеш — маленькая.
- Если на странице установлен «фиксированный» фильтр (одинаковый на каждом хите – такой, на который пользователь своим выбором повлиять не может), например, когда товары на странице фильтруются по свойству «Новинка», параметр «Кешировать при установленном фильтре» приравнивают к значению «Да».

На странице [https://studija-jal.ru/catalog?filters=type:1](#) есть возможность использования умного фильтра, к тому же на странице устанавливается свой фильтр, исходя из параметра «TYPE» в URL. В компонент передаётся комбинация этих фильтров: предустановленного и выбранного пользователем. В этом случае параметр «Кешировать при установленном фильтре» в компоненте catalog.section лучше устанавливать динамически. Если параметры в умном фильтре не выбраны (в наличии только «предустановленный» фильтр), выставляем отметку «Да» на условие «Кешировать при установленном фильтре». В этом случае фильтр на странице [https://studija-jal.ru/catalog?filters=type:1](#) будет кешироваться.

Если параметры в умном фильтре выбраны, например, на странице, выставляем «Кешировать при установленном фильтре» — «Нет».

5. Фасетный индекс

На странице [https://studija-jal.ru/catalog?filters=type:1](#) установлен умный фильтр. Даже если самой фильтрации в данный момент не происходит, умному фильтру необходимо построить набор свойств для показа. Это будет происходить быстрее, если создан «Фасетный индекс». Сейчас «Фасетный индекс» не валиден.



Пересоздать его можно здесь:

Рабочий стол – Контент – Инфоблоки – Фасетные индексы.

6. Таблица пользователей корзины

Количество записей в таблице `b_sale_fuser` составляет почти 122 000 000 (!), а общий вес таблицы – 8.3 Гб.



Сейчас в таблицу добавляется порядка 67 000 записей каждые несколько минут. Увеличение этой таблицы – довольно распространённая проблема в Битрикс, но обычно она увеличивается всего лишь до 100-200 000 записей.

Чаще всего это вызвано тем, что пользователи корзины создаются даже на тех хитах, когда добавление в корзину не происходит. Это происходит при вызове функции `CSaleBasket::GetBasketUserID()` без параметров:

(см. комментарий).

Сейчас есть две проблемы: огромный размер таблицы и регулярное добавление новых строк в больших объёмах. Возможно, первая проблема уже возникала ранее, ситуацию можно решить с помощью инструмента, который уже установлен на сайте:



Настройки — Инспектор данных — Проверка b_sale_fuser.

Непонятно, с чем на данный момент связана вторая проблема. Раньше такая ситуация могла возникнуть в связи с использованием стандартного компонента «малой корзины», возможно, и сейчас дело именно в нём.

Есть вероятность, что увеличение таблицы частично связано с тем, что не отработывает агент, который её очищает (см. пункт «Ошибка в проверке сайта»), но вряд ли столь глобальные изменения произошли за несколько дней.

Решение этих проблем должно снизить нагрузку на БД: за счёт уменьшения размера таблиц и за счёт устранения непрекращающейся записи в БД.

7. Товары в разделах каталога

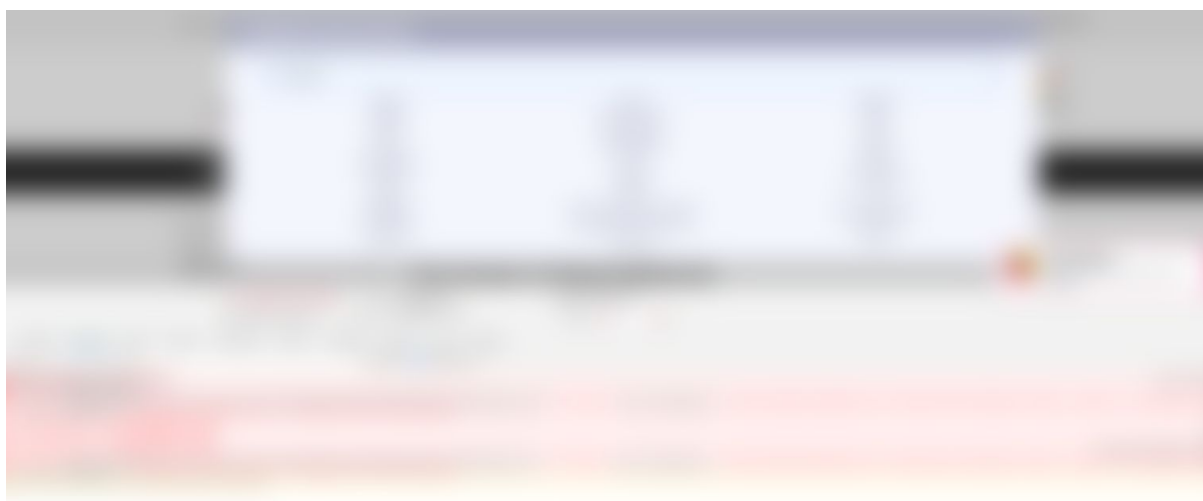
В разделах каталога выводится слишком много товаров.

Раздел [https://www.1c-bitrix.ru/...](#)

Чем больше товаров выводится на странице, тем больше времени требуется для загрузки этой страницы. Если по умолчанию выводить 18 товаров, а не 72, то сократится количество обращений к базе данных и количество фотографий, загружаемых на странице, что приведет к сокращению времени загрузки.



8. Скрипт фильтрации городов



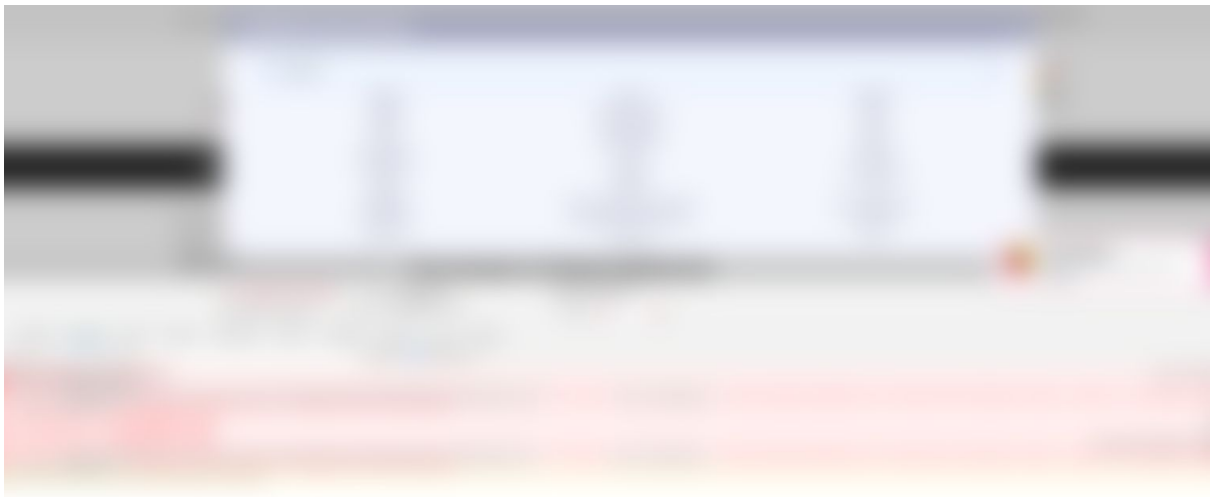
Скрипт Uncaught ReferenceError: \$ is not defined.

Ошибка выводится в скрипте, который отвечает за фильтрацию городов по мере печати в окне с регионами. Сейчас эта функция не работает. Чтобы исправить эту ошибку, нужно поменять местами порядок подключения скриптов: сделать так, чтобы jQuery подключался до скрипта с ошибкой.

9. Технология CDN

На сайте подключена технология CDN. Эта технология позволяет отдавать пользователю файлы с серверов, расположенных в одном с ним регионе.

На этом же скриншоте с ошибкой фильтрации городов видно, что не удалось загрузить шрифт с CDN Битрикса.



Эта ошибка оказывает влияние на скорость загрузки страницы. Для её устранения рекомендуем обратиться в Битрикс.

10. Проверка использования компонентов

Причины медленной загрузки сайта из-за использования нестандартных компонентов не выявлены.

Практически все компоненты сайта стандартные, кастомизирован только умный фильтр. Кроме стандартных компонентов используются модули маркетплейса. Модуль интеграции с 1С стандартный.

11. Проверка оптимальности настройки сервера

Проверка оптимальности настройки сервера показала, что обнаружена только одна ошибка.

Ошибка!

Не настроен запуск `cron_events.php` на `cron`, последний агент отработал больше суток назад

Прямого влияния на работу сайта она не оказывает, но если используются агенты для очистки базы, нужно устранить ошибку.

Общая работа сайта		
Наличие необходимых модулей php	✓ Все необходимые модули установлены	🔍
Обязательные параметры PHP	✓ Настройки правильные	🔍
Модули веб-сервера	✓ Конфликтов не выявлено	🔍
Значения переменных сервера	✓ Корректные	🔍
Сохранение сессии	✓ Успешно	🔍
Параметры настройки UTF (encoding и константа BX_UTF)	✓ Правильные. Сайт работает в UTF кодировке	🔍
Служебные скрипты в корне сайта	✓ Отсутствуют	🔍
Работа с cookie	✓ Успешно	🔍
Выполнение агентов на cron	⚠ Ошибка! Не настроен запуск <code>cron_events.php</code> на <code>cron</code> , последний агент отработал больше суток назад.	🔍