помогаем бизнесу
продавать онлайн

ООО «Редлайн»
г. Томск, ул. Карла Маркса 17а, 4 этаж
Тел.: 8 (3822) 99-41-30
info@redlg.ru
https://redlg.ru

Требования к back-end

Общие требования

- Разработка ведется в кодировке UTF-8
- В Git не должно находиться лишних файлов и папок
    - node_modules
    - bitrix
    - upload
    - vendor
- Разработка ведётся в папке local
- Категорически запрещается модифицировать код ядра Bitrix
- Подробнее о правилах работы с Git:
    - Каждая задача - отдельный коммит
    - Коммит не должен приводить к неработоспособности проекта
    - Проект должен содержать dev и master ветки

Требования к настройке 1С-Битрикс

1. Внешние файлы сервисов и скриптов подключены без указания протокола
    a. <script src="//sitename.com"></script>
2. Ссылки на внешние сайты открываются в новой вкладке.
3. Закрыты от индексации служебные разделы /bitrix/*; /vendor/* и /.git/*,/upload/*,/local/*,/ajax/* и папку с включаемыми областями /includes/

Red line

помогаем бизнесу
продавать онлайн

ООО «Редлайн»
г. Томск, ул. Карла Маркса 17а, 4 этаж
Тел.: 8 (3822) 99-41-30
info@redlg.ru
https://redlg.ru

Общие требования к работе функциональных блоков

1. Не используются прямые запросы к БД, если необходимую логику можно реализовать через api. Исключение: запрос через api работает в несколько раз медленней
2. Поддержка стандартного SEO битрикса должна работать в компонентах news, catalog
3. Для динамически подгружаемого контента предусмотрен прелоадер(если есть в верстке или дизайне)
4. В комплексных компонентах используется стандартная страница 404 ошибки (реализуется для согласованного набора инфоблоков)
5. Результаты поиска имеют корректные ссылки, нет 404 ошибок при переходе.
6. Не используются собственные правила в urlrewrite для создания динамических страниц, если этот функционал можно реализовать при помощи комплексных компонентов. .
7. Фильтры элементов в каталогах привязаны к URL(GET параметры) (если прописано в тз)

Архитектурные требования

1. Все дополнительные классы и библиотеки должны находиться в /local/php_interface/ и подключаться через /local/php_interface/init.php
    a. Классы
        i. /local/php_interface/class/
    b. Библиотеки
        i. /local/php_interface/lib/
        ii. Если библиотека есть в composer, она ставится через него
2. Если для функционала требуется дополнительный визуальный интерфейс в админ. панели, то функционал реализуется как модуль для битрикс.
3. ООП-методология
4. При написании импорта/экспорта данных обязательно должны создаваться логи (как об ошибках, так и о успешном завершении)
5. Для хранения плоских данных, не связанных с другими сущностями (например размеры одежды, цвета, типы деталей и т.д.) используются highload-инфоблоки
6. Вся повторяющаяся логика должна быть вынесена в методы классов собственного модуля.
7. В файлах template.php шаблонов компонентов не должна происходить модификация данных или получение данных из базы данных, только вывод.
8. Работа с данными производится в component.php и result_modifier.php
9. Включаемые области проекта размещены в непосредственно в папке страницы, где она используется /{PAGE}/include/, если область подключается в хедере

ООО «Редлайн»
г. Томск, ул. Карла Маркса 17а, 4 этаж
Тел.: 8 (3822) 99-41-30
info@redlg.ru
https://redlg.ru

или футере, то она должна быть размещена в папке с шаблоном - /local/templates/{TEMPLATE}/include/

Работа с внешними модулями и библиотеками

Конечный набор обязательных требований из данного раздела определяется тим-лидом команды.

1. Внешние библиотеки должны быть подключены через composer и находиться в папке /local/php_interface/
2. Используем модуль миграции https://github.com/izica/bitrix-migrations, все изменения делаем через него.
3. Приветствуется(опционально) при написании собственных компонентов использовать модуль bbc - http://bbc.bitrix.expert/docs/v1/
4. Настройки проекта задаются через модуль “Дополнительные настройки” - https://marketplace.1c-bitrix.ru/solutions/grain.customsettings/

Red line

помогаем бизнесу
продавать онлайн

ООО «Редлайн»
г. Томск, ул. Карла Маркса 17а, 4 этаж
Тел.: 8 (3822) 99-41-30
info@redlg.ru
https://redlg.ru

Требования к качеству кода

1. Классы и методы классов документированы в соответствии с PHPDoc
2. Именование
    a. База данных
        i. таблица
            1. snake_case
        ii. поле
            1. snake_case
    b. Классы
        i. UpperCamelCase
    c. Функции
        i. lowerCamelCase
    d. Переменные
        i. Массив
            1. $arUsers
        ii. Объект
            1. $obUser
        iii. Число
            1. $nUserId
        iv. Строка
            1. $sUserName
        v. Boolean
            1. $isActive
            2. $bActive
3. В качестве стандарта PHP используется PSR-2: http://www.php-fig.org/psr/psr-2/
4. В коде должна использоваться табуляция - 1 таб = 4 пробела.
5. Категорически избегать вывода ошибок посетителям сайта.
    a. Для дебага кода, рекомендуется использовать https://packagist.org/packages/izica/php-browser-log
6. Не использовать массив $_REQUEST. Пользоваться $_POST и $_GET.
7. В коде не должны использоваться глобальные переменные (Исключение: стандартные переменные $APPLICATION, $USER, переменная для фильтрации в компонентах)
8. Стараться придерживаться принципа DRY.
9. В метод передается не более 4 входных параметров, иначе они должны быть сгруппированы
10. Метод должен решать не более чем одну задачу, иначе он должен быть декомпозирован

ООО «Редлайн»
г. Томск, ул. Карла Маркса 17а, 4 этаж
Тел.: 8 (3822) 99-41-30
info@redlg.ru
https://redlg.ru

Производительность

1. Алгоритмы оптимизированы: Нет повторных запросов одних и тех же данных, нет запросов к БД в цикле, выбираются только необходимые данные и т.д
2. При получении данных по ajax, стараться минимизировать размер данных(использовать json)

помогаем бизнесу
продавать онлайн

ООО «Редлайн»
г. Томск, ул. Карла Маркса 17а, 4 этаж
Тел.: 8 (3822) 99-41-30
info@redlg.ru
https://redlg.ru

Безопасность

1. При загрузке файлов (например, прикрепление файлов в отзывах) выполняются необходимые проверки на стороне сервера (тип, размер и т.д.)
2. PHP-скрипты должны учитывать права доступа (защита от direct access)

Frontend

1. При необходимости изменения верстки, менять её с помощью инструментов приложенных к верстке(сборщики)