

Важная информация для проекта:

Домена сайта: <https://exchangesumo.com/>

Версия php: 7.2

Версия WordPress: 5.2.4

Что надо сделать:

Надо сделать для WORDPRESS плагин рейтинга для оценки статей, обменов и прочего материала на на нашем сайте в виде звезд с микроразметкой для поисковиков. С его помощью чтобы можно было проголосовать и узнать среднюю оценку пользователей.

Требования к плагину:

1. Создает обработку шорткода для добавления HTML в контент страницы или через `do_shortcode()` там где посчитаем нужным, а добавления JS/CSS как отдельные файлы .
2. Привязки результатов рейтинга к уникальному урл страницы.

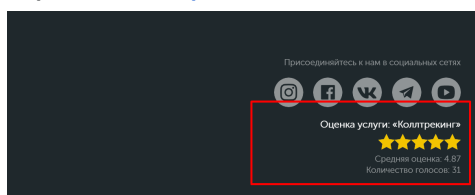
особенности нашей WP что на одной странице привязаны разные урл-адреса.

Например страницы имеют адреса

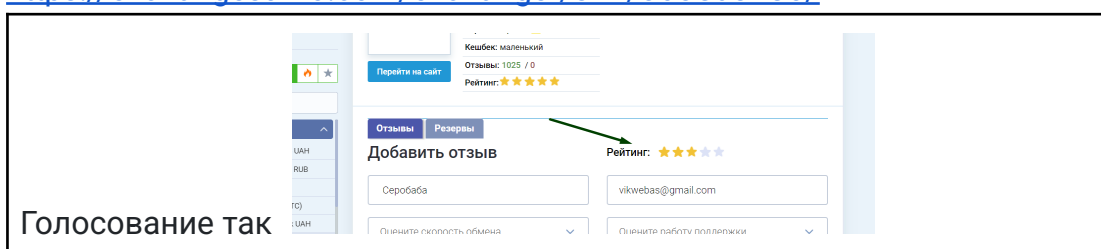
<https://exchangesumo.com/obmen/MONOBUAH-ETH/> или

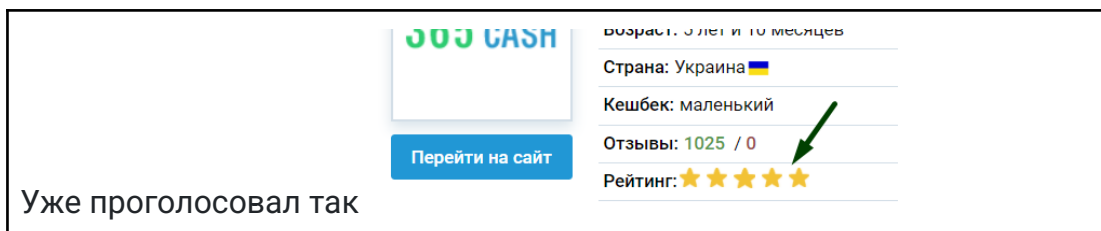
<https://exchangesumo.com/obmen/P24UAH-EXMBTC/> , но в системе WP это всего одна запись в таблице posts с id=100

3. Учесть чтобы один IP не голосовал дважды (и даже с разных устройств)
4. пример того как должен работать виджет можно посмотреть на странице <https://www.calltouch.ru/product/calltracking/>

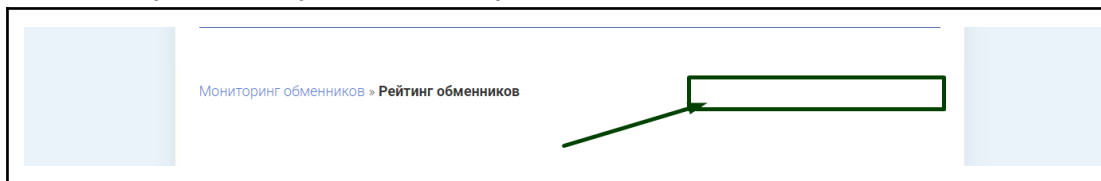


5. Готовое html для решения нужно взять на странице <https://exchangesumo.com/exchanger/817/365CashCo/>





6. Вставить рейтинг нужно вниз страницы:



7. Надо исключить показ плагином рейтингов на страницах обменников (post_type=exchanger) (/exchanger/.../.../)

Дополнительные требования к скриптам:

8. Скрипт создается как плагин. Вставляется как шорткод.
9. Тема одна и все что касается HTML/CSS/JS для плагина, адаптированное под задачу, размещается внутри плагина .

JS и CSS как отдельные файлы (в html виджете только самое необходимое для уменьшения тела страницы) . Все js и css потом объединятся в один файл и хранятся на CDN (посмотрите как сейчас)

10. Учесть что база загружена основной задачей сервиса, поэтому не грузить базу расчетами итогов “на лету” , а считать итоги сразу при голосовании чтобы потом использовать уже готовые расчеты
11. Не использовать для сбора данные admin-ajax , а создавать свой обработчик настраивая его только на обработку конкретных запросов
12. Придерживаться “условной” модели MVC при создании плагина

Где главный скрипт плагина нечто типа контроллера с экшенами, работа с базой данных через модели плагина , html выносится в отдельные шаблоны , js и css в папке плагина /assets/ (пример ожидаемой структуры <http://prntscr.com/xtaemi>) Подробный пример дадим выбранному исполнителю в личку.

13. Уходить от универсальности решения. В приоритете скорость обработки и выдачи html.
14. Приветствуется минимализм в подключениях файлов, вызовах БД и других функций,...
15. На хуки вешать только самое необходимое .
16. namespace приветствуется, но учесть что WP сам подключает основной файл плагина после активации, “городить” для этого плагина autoload не надо (см. п.п. выше про “минимализм”)

17. В активацию написать нужно создание таблиц для плагина.
Деактивацию удалениям таблиц и следов плагина писать не нужно.
18. Тут может быть что-то еще ... если возникнут дополнительные вопросы..

Ожидается что это будет готовый плагин который активировали и он сделал все что нужно для этого сам , запустился и работает в нашей среде. На страницах появился рейтинг.

Если возникнут дополнительные вопросы - пишите , можно прямо в этом файле как комментарии добавлять.