

Примечания:

Салатовый - требуется обратить внимание программисту и/или требует обсуждения с программистом.

Желтым - требуется обратить внимание дизайнеру.

Сиреневым - требуются решения от нас.

ТЗ для сервиса парсинга данных с маркетплейса Amazon

1. Общее описание. Проект представляет из себя SaaS, основанный на парсинге данных с сайта Amazon.com и выводе данных в интерактивные таблицы с их последующей фильтрацией и сортировкой; также предусмотрен форум и личный кабинет, за которым будет закрепляться оплаченный тариф (уровень доступа) и сообщения на форуме. Цель - предоставить возможность анализа данных с Амазона (парсер) на основе готовых алгоритмов. Сервис является почти полной копией аналога Amzscout.com.
(для проверки работы того или иного функционала у конкурентов - установите расширение для Chrome через сайт <https://amzscout.net/> (введите email, далее нажмите "Try it for free" и скачайте с Chrome маркетплейса; затем зайдите на Amazon.com, введите в поиск любое название товара - например "game" и нажмите на иконку расширения в панели расширений браузера Chrome)
- 1.1. Предполагается, что часть проекта будет работать, как HTML парсер (для сбора больших данных и аналитики), а часть - как парсер через официальное API Amazon.com (в тех частях, где нужны более свежие/"быстрые" и обновленные данные).
 - 1.1.1. Например, для выявления самых минимальных значений по продажам за какой-то период - понадобятся данные по всем товарам за этот период, что проблематично получить через API (в API лимит на 30 запросов в секунду), но можно получить благодаря парсингу
 - 1.1.1.1. Парсер - желательно использовать Scrapy
 - 1.1.1.1.1. Также требуется конфигурирование прокси и
 - 1.1.1.1.2. anti-blacklisting
 - 1.1.2. В то же время, если мы хотим получить первую страницу результатов поиска (данные по ним - т.е. кол-во продаж и т.д.) - это проще сделать через API, данные будут более свежими и обновление будет происходить быстрее.
 - 1.1.2.1. Для экономии квоты на API запросы - запросы нужно кешировать
 - 1.1.2.1.1. и обновлять не чаще 1 раза в день
 - 1.1.2.1.1.1. (этот параметр должен быть в конфиге).

1.1.2.1.1.1.1. также в конфиг необходимо
добавить различные критерии
обновления и их частоту
(например:

1.1.2.1.1.1.1.1. как часто и какие данные
мы парсим...

1.1.2.1.1.1.1.2. обновляем ли информацию
при запросе этих данных и
как часто

1.1.3. Общее правило такое: статистику по продажам частично храним на сервере (там, где нужно делать расчеты и сортировку), а информацию по товару (тексты описания, картинки, характеристики и т.д. - все, что занимает большой объем на диске и одновременно может обновляться достаточно часто) - не храним, а подтягиваем каждый раз с Amazon.com

1.1.3.1. При этом информацию о продажах(рейтинге товаров) можно запрашивать пачками, экономя лимиты на запросы. (Цитата из документации API Амазона: *"You can also use the API to retrieve price information, title information, prime detection, review counts, and ratings of Amazon products in bulk, up to 1,000 with each request"*).

2. Эксплуатационное назначение - продажа сервиса по аналитике спроса на товары по модели подписки и рекуррентных платежей. Сервис предназначен для определения уровней продаж на Amazon, таким образом в конечном итоге предназначен для поиска прибыльных ниш/товаров/категорий.

3. Структура сайта клиентской части

3.1. Основное меню:

состоит из:

3.2. Главная (digardo.com/)

3.2.1. Если пользователь залогинен - редирект на страницу Search

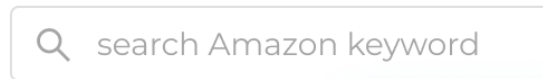
3.2.2. Если пользователь НЕ залогинен - то действуют определенные ограничения по поиску, см. далее (например, можно отображать не более **X первых результатов** и далее

показывать замок + ••• (3 точки) и кнопку для покупки подписки на PRO-версию.

3.3. Логотип (при клике должен вести на главную страницу)

3.4. Search. Основной раздел. Предназначен для поиска и глубокой аналитики продаж/спроса по товарам Amazon.com

3.4.1. Поле для поиска по ключевым фразам. Основной механизм, через который происходит поиск продуктов на амазоне (киллер фича).



- 3.4.1.1. Должна быть возможность искать по синонимам
- 3.4.1.2. Должна быть возможность искать по частично обрезанным ключевым фразам
- 3.4.1.3. Должна быть возможность искать одновременно по нескольким ключевым фразам, разделенным через запятую.

3.4.2. Чекбоксы в поиске/фильтры.



Данный набор чекбоксов предназначен для фильтрации результатов поиска / для поиска по каким-то детальным критериям.

При нажатии на [more criterias](#) V разворачивается полный список всех фильтров. При нажатии на какой либо из фильтров - появляются настройки этого фильтра.

Например, при нажатии Date - открывается настройка фильтра с выбором дат:

<https://i.imgur.com/HyKij1y.png>

Должны присутствовать такие фильтры:

3.4.2.1. Дата (Date). Фильтр предназначен для вывода результатов по конкретным датам (диапазон). Диапазон должен регулироваться с помощью datepicker, см. пример макета:

<https://i.imgur.com/1UJp7KL.png>

3.4.2.1.1. При наведении/клике на "Год" - (2018 в макете-примере) - открывается мини-окошко с выбором года и возможностью скролла.

3.4.2.2. URL. Возможность поиска по конкретным URL или по ASIN

3.4.2.2.1. Через обычное поле быстрого поиска, скриншот макета:

<https://i.imgur.com/OgklnbG.png>

3.4.2.2.2. Через расширенное поле поиска URL, см. примеры макетов:

<https://i.imgur.com/FHpnNk5.png>

<https://i.imgur.com/KloOcDw.png>

- 3.4.2.2.3. Важно то, что поиск осуществляется как при вводе всего URL, так и при вводе только ASIN. Что такое ASIN, см. скриншот:
<https://i.imgur.com/Xw7iwFn.png>
(он является частью URL)
- 3.4.2.2.4. Фильтр URL даёт возможность вывести результаты по конкретным ссылкам. Можно ввести не более 6 ссылок. Если ввести больше, то появляется сообщение с предупреждением (см. макет:
<https://i.imgur.com/u5NdZpv.png>).
- 3.4.2.3. Reviews. Возможность указать диапазон фильтрации по минимальному и максимальному кол-ву отзывов.
- 3.4.2.4. Price. Возможность указать диапазон цен товара. Если диапазон будет, к примеру от 30 до 30 - то будет использоваться цена "30" при выводе.
Скриншот настроек фильтра:
<https://i.imgur.com/GwNnlUS.png>
- 3.4.2.4.1. Также должен отображаться массив данных в виде гистограммы - т.е. кол-во товаров по той или иной цене выраженное барами/столбцами. При наведении на столбец - отображается кол-во товаров с этой ценой. По скриншоту должно быть понятно, о чем идет речь
- 3.4.2.4.1.1. Важно то, что при отображении массивов данных - учитываются предыдущие выбранные фильтры.
- 3.4.2.4.1.1.1. Например, если кроме фильтра Price выбран фильтр Est. Revenue - то массив данных в гистограмме Price уменьшится в соответствии с критериями всех фильтров, суммарно.
- 3.4.2.5. Цена после скидки. То же самое, что и предыдущее.
- 3.4.2.6. **Изменение цены. Выводить только те товары, цена которых изменилась более, чем на -X% или X% (x - можно указать в настройках фильтра).**
- 3.4.2.7. **Категория. Возможность выбора категории и подкатегории товаров (данные берутся с Амазон)**
- 3.4.2.8. Average Monthly Sales - среднее кол-во продаж за месяц. Используется такой же пикер, как и для Price (Цены), пример: <https://i.imgur.com/GwNnlUS.png> (также нужно отображать массив усредненных данных; только здесь мы выбираем не цену, а кол-во продаж).
- 3.4.2.9. **Мин. кол-во продаж на товар** - здесь отображается цифра, которая зафиксирована у любого продавца с наименьшим кол-вом продаж за месяц в целом по поисковому запросу. *Используется такой же пикер, как и для Price, только без диапазона - поскольку "мин. кол-во продаж на товар" - значение единичное. Важно то, что при*

отображении массивов данных - учитываются предыдущие выбранные фильтры.

3.4.2.10. **20% нижний перцентиль медианный уровень продаж** - берем 20% на нижней границе/перцентиль худших зафиксированных результатов у продавцов с наименьшим кол-вом продаж за месяц по данному запросу, затем высчитываем их медиану. Пикер и требования к нему такие же, как и для Price.

3.4.2.11. **MINIMUM Last 12M revenue** - здесь отображается минимальная цифра, которая зафиксирована у любого продавца с доходом (ревеню) за последние 12 месяцев (минимальная). Пикер и требования к нему такие же, как и для Price.

3.4.2.12. **MINIMUM Next 12M revenue** - тоже самое, что и предыдущее, но экстраполированное в будущее 12 месяцев. Пикер и требования к нему такие же, как и для Price.

3.4.2.13. **Sales Change** - Изменение продаж, график; возможность выбрать диапазон.

3.4.2.14. **ANNUAL SALES TREND** - на сколько процентов изменились продажи по всем листингам (с учетом критериев поиска) за 12 месяцев. Пикер и требования к нему такие же, как и для Price (только выбираем % и год; процент выбираем таким же пикером, как для Price, а год - выбираем через такой пикер: <https://i.imgur.com/4nTpyab.png> ; при этом, массив данных при выборе % - отображается только за выбранный год; если прокручивать года, то массивы также изменяются быстро - таким образом пользователь может "просканировать" визуально разные года и то, как это все меняется).

3.4.2.15. Важные примечания:

1. Разница между Sales и Revenue: Sales - кол-во продаж, а Revenue - доход от этих самых продаж (кол-во продаж умноженное на цену продукта).

2. Алгоритм должен максимально точно прогнозировать кол-во продаж и соответствовать значениям AmznScout +/-10% отклонений. Само кол-во продаж не предоставляется Amazon.com, но Амазон в API предоставляет так называемый Amazon Product Ranking и BSR по которому и рассчитывается примерное число продаж (алгоритмы расчета можно прогуглить, это делается достаточно просто). Для определения продаж по продукту используется API Amazon и т.н. Amazon Product Ranking - гуглите, есть алгоритмы, это делается достаточно просто.

3.4.3. Большинство этих же полей из фильтров - должны отображаться в таблице с выводом результатов поиска - см. далее.

3.4.4. Вывод в таблице можно настроить через Config view



<https://i.imgur.com/xVVVQaE.png>

(при нажатии - появляется окно с настройками того, какие данные и столбцы отображать в таблице; пример макета:

<https://i.imgur.com/5PyUHrp.png>

- 3.4.4.1. При перемещении порядка критериев - меняется и порядок в таблице, где 1 - отображается с левой части таблицы.
- 3.4.4.2. При нажатии на "Show blacklisted items" - открывается окно с настройками/списком товаров, которые добавлены в блеклист. Пример макета: <https://i.imgur.com/t7ReLOi.png>
- 3.4.4.2.1. При нажатии Restore - добавленные в блеклист товары восстанавливаются и снова отображаются в основной таблице.
- 3.4.5. **Важно:** если какое-то поле/критерий выбран в фильтре/фильтрах(см. выше) - столбец принудительно включается в таблице, даже, если он отключен в настройках таблицы.
- 3.4.6. При вводе поискового запроса и нажатии на кнопку Search - появляется таблица с результатами отфильтрованного (если выбран фильтр) поиска.
- 3.4.7. Также, после нажатия на Search - появляется кнопка Save results
- 3.4.8. Полученные результаты можно сохранить с помощью кнопки Save results



- 3.4.8.1. При нажатии Save results - происходит автоматическое создание результата поиска с именем в формате, в котором по умолчанию указывается "Last saved #" (без кавычек), где "#" - номер сохраненного результата (результаты, где имя указывается/изменяется вручную - не учитываются в нумерации), нумерация идет с последнего существующего номера. Например, если есть Last saved 3, то следующий создастся Last saved 4. Если после этого удалить Last saved 3, то все равно создастся Last saved 5, т.к. у нас доступен последний - Last saved 4. Если же удалить Last saved 4, то тогда следующим опять будет создан Last saved. При этом, после сохранения результата - курсор и обзор перебрасывается на созданный результат, в котором можно указать имя результата, сохранить его(галочка) или отменить(крестик); пример:

[Last saved name 2](#) ✓ ✕

(см. в Компонентах).

- 3.4.8.1.1. После сохранения - анимация+ переброс курсора на Last saved name xNumber (с возможностью сразу

там же ввести нужное имя), который оказывается новый, в самой левой части.

3.4.8.2. Сохраненные результаты отображаются под полем поиска.


3.4.8.3. Последний сохраненный результат всегда перемещается в левую, крайнюю область

3.4.8.4. Результат можно загрузить, нажав на него

3.4.8.4.1. В этом случае последний использованный результат перемещается в левую, крайнюю область

3.4.8.4.2. Также, в поле поиска (где мы вводим ключевой запрос для поиска) автоматически подставляется то, какой запрос был сохранен и подгружен

3.4.8.4.3. Если текущий результат не сохранен и пользователь пытается подгрузить другой или закрыть страницу - всплывает предупреждение.

3.4.8.5. Результаты также можно удалять (иконка - крестик ) или перематывать/прокручивать[см. иконку там же, рядом - стрелки перематки; при этом также должен работать скролл мышью, когда мы наводим на эту область])

3.4.9. Справа от формы поиска располагается кнопка Discuss, которая открывает чат/мини-форум



<https://i.imgur.com/la02Vch.png>

3.4.9.1. Представляет из себя стандартный чат, который ведёт на форум. Сообщения с чата дублируются на форум и наоборот. При этом на основе подраздела "Today chat"(ежедневный чат) каждый день создаётся тема на форуме с архивом/историей переписки этого чата. Для реализации чата/форума можно использовать готовые, сторонние решения (IPB? В нем также есть плагин чата)

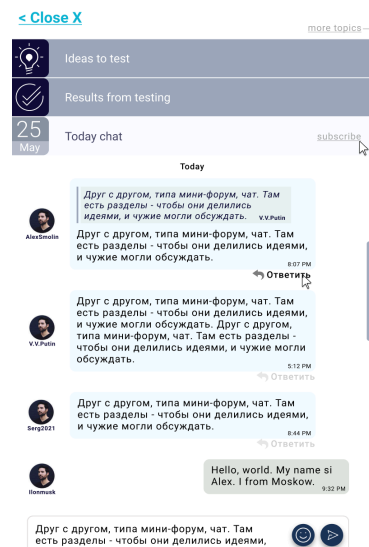
3.4.9.2. Кнопка Discuss - плавающая при вертикальном скролле, но вплоть до границы таблицы, которая (таблица) чуть ниже этой кнопки:

<https://i.imgur.com/iSR8hNC.png>

3.4.9.3. При наведении - фиолетовым. При клике - синим, с эффектом

"разворачивания" чата из кнопки (как в mac OS) или "zoom out"/выдвижение)

3.4.9.4. Пример окна чата/мини-форума:



<https://i.imgur.com/hCZdv7P.png>

(вероятно потребуются WebSocket; необходимо оценить стоимость этой фичи отдельно)

3.4.10. Результаты поиска/вывод результатов начинается с отображения ключевых показателей, которые располагаются чуть выше основной таблицы с результатами



<https://i.imgur.com/XYnryl1.png>

3.4.10.1. Ключевые показатели - это суммирование/резюмирование и усреднение всех результатов из основной таблицы, что расположена ниже и включают в себя Total Trend (общий тренд) в виде графика, Potential Score - общий рейтинг запроса, рассчитываемый по формуле (см. приложение с формулой расчета) и некоторые другие показатели, которые дублируются из общего вида/общей таблицы, но рассчитываются как среднее по всем товарам (в таблице - лишь по каждому товару отдельно), например:



3.4.10.1.1. - Total Trend - общий график продаж по всем товарам по данному запросу (среднее значение кол-ва продаж, т.е.: суммируем все продажи по этому запросу и делим на кол-во позиций в выдаче). Используется 3 графика: средняя(сглаженная), общий тренд (линия) и индикатор в виде уменьшения/увеличения показателей - в форме осциллятора.

3.4.10.1.1.1. Осциллятор показывает только уменьшение/увеличение показателей в процентном соотношении. Каждый бар осциллятора - это промежуток времени

деленное на число баров, отображаемых в текущий момент на экране. По умолчанию берется промежуток 1 год (если не выбраны даты вручную). Например, если у нас сейчас выбран промежуток 1 год, а на экране отображается 100 баров, то каждый бар - это будет 3.5 дня - т.е. в этом случае мы смотрим, как менялись данные каждые 3.5 дня в сравнении с предыдущими барами - увеличивались ли они или уменьшались и на сколько %. Таким образом в результатах поиска также может быть выведен тренд/график показателей за определенный промежуток времени. Еще пример для лучшего понимания: диапазон делится на одинаковое кол-во баров, где каждый бар представляет отрезок времени/кол-во баров. Например, если мы берём промежуток 30 дней, а баров в график вмещается всего 10, то каждый бар будет делить 3-х дневный промежуток.

3.4.10.1.1.2. Средняя же - это 3 таких бара(как в предыдущем пункте), усредненные (но, показанные уже не в процентном соотношении, а в абсолютном).

3.4.10.1.1.3. Линия тренда - это линия, которая проводится от минимального значения на начало периода и до максимального значения на конец периода, проведенная по средней линии (может быть и отрицательной - т.е. иметь направление вниз).

20% low percentile [?]


3.4.10.1.2. **\$995054** - средний уровень ревеню у 20 худших продавцов. Перцентиль считается так: раскладываем числовой ряд, сортируем от наибольшего к меньшему и берем 20% наименьших чисел, которые находятся "внизу" сортировки. Далее вычисляем среднюю этих 20 наименьших %%. **При клике - должна быть возможность выбрать % (например, мы хотим показывать не 20%, а 5%) и показатель (например, мы хотим показывать не ревеню, а 20% самых низких цен и т.д.).**


3.4.10.1.3. **Newbie first 3 month revenue. Медианное значение Ревеню у новичков за первые X месяцев. Здесь учитываются только продавцы, которые начали продажи в указанный период. При клике на показатель - должны быть следующие фильтры**


3.4.10.1.3.1. Выбор, что именно показывать: ревеню или продажи или что-то другое. По умолчанию - ревеню.


3.4.10.1.3.2. Выбор, сколько первых месяцев считать.

3.4.10.1.3.3. Чекбокс - учитывать ли историю за все время или только выбранный период в основном фильтре. Например, если мы выберем учитывать всю историю - то будут искаряться продавцы за все время по данному ключевому запросу (по умолчанию так). Если же выбрать учитывать только изначально выбранный промежуток, скажем с 2020 по 2021, то будут учитываться только продавцы, которые НАЧАЛИ продавать строго в этом промежутке, а все остальные - исключаются.

3.4.10.1.4.  - здесь отображается минимальная цифра, которая зафиксирована у любого продавца с доходом (ревеню) за последние 12 месяцев (минимальная). (здесь учитываются не все продавцы, а именно 1 продавец - с самым меньшим значением)

3.4.10.1.5.  - то, как изменился тренд продаж за последний год по данному запросу. Берем продажи товаров по запросу и делим на кол-во позиций (т.е. вычисляем среднее).

3.4.10.1.6.  Average Monthly Sales - медианное кол-во продаж за месяц по данному запросу в целом. Берем продажи по этому запросу и вычисляем медиану.

3.4.10.1.7.  - средняя цена товара по данному запросу. Берем цены по этому запросу, все суммируем и делим на кол-во позиций - получаем среднее. Для этого показателя - берем только первую страницу (через API) (возможно в API уже есть вычисление средней цены).


3.4.10.1.8. Reviews. Среднее кол-во отзывов по запросу.

3.4.10.1.9. Среднее кол-во остатков по всему запросу. При наведении - показывать все количество остатков.


3.4.10.1.9.1. При этом также кол-во остатков должно отображаться и по каждому продавцу/товару.

3.4.10.1.9.2. Также выборка/диапазоны по остаткам должна быть в фильтрах.

Total Our Score


3.4.10.1.10.  - рейтинг/скоринг, рассчитанный по нашей формуле (см. приложение #1). Идеи, как его рассчитывать: каждому показателю нужно определить вес и далее все веса просто складываются.

PDC Score

3.4.10.1.11.  - т.н. Profit Demand Competition - это рейтинг, который обычно есть у конкурентов - нужно изучать конкурентов и думать, как его считать.
Пример у <https://amzscout.net/>
<https://i.imgur.com/vUI6kSI.png>
<https://i.imgur.com/Sy0zMB9.png>

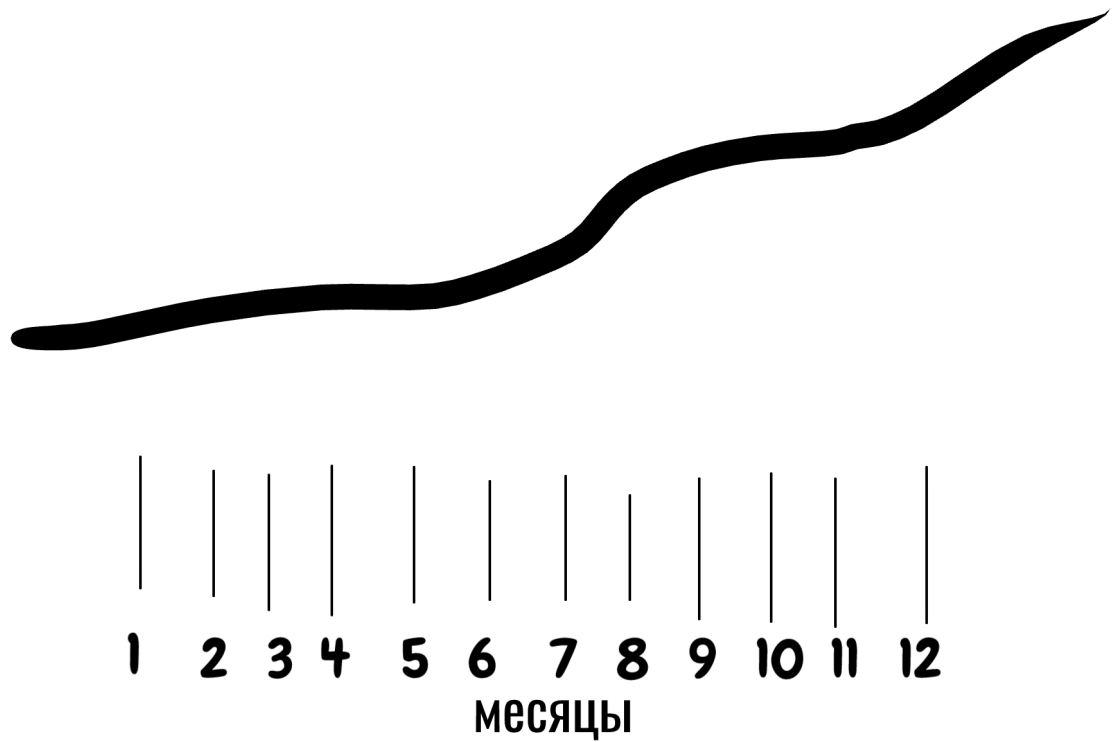
Можно на бесплатном PDC выдавать, а наш скоринг только на платном.

3.4.10.1.12. Visibility score - кол-во запросов этой ниши + данные суммироваться должны с тем, что выдал <https://amzscout.net/> (т.е. данные нужно подтягивать от них + добавлять наши).

3.4.10.1.13.  - переключение между отображением вида:
3.4.10.1.13.1. Отображение в виде таблицы:
<https://i.imgur.com/gUE6kkG.png>
3.4.10.1.13.2. Отображение в виде плиток:
<https://i.imgur.com/kIn0aRW.png>

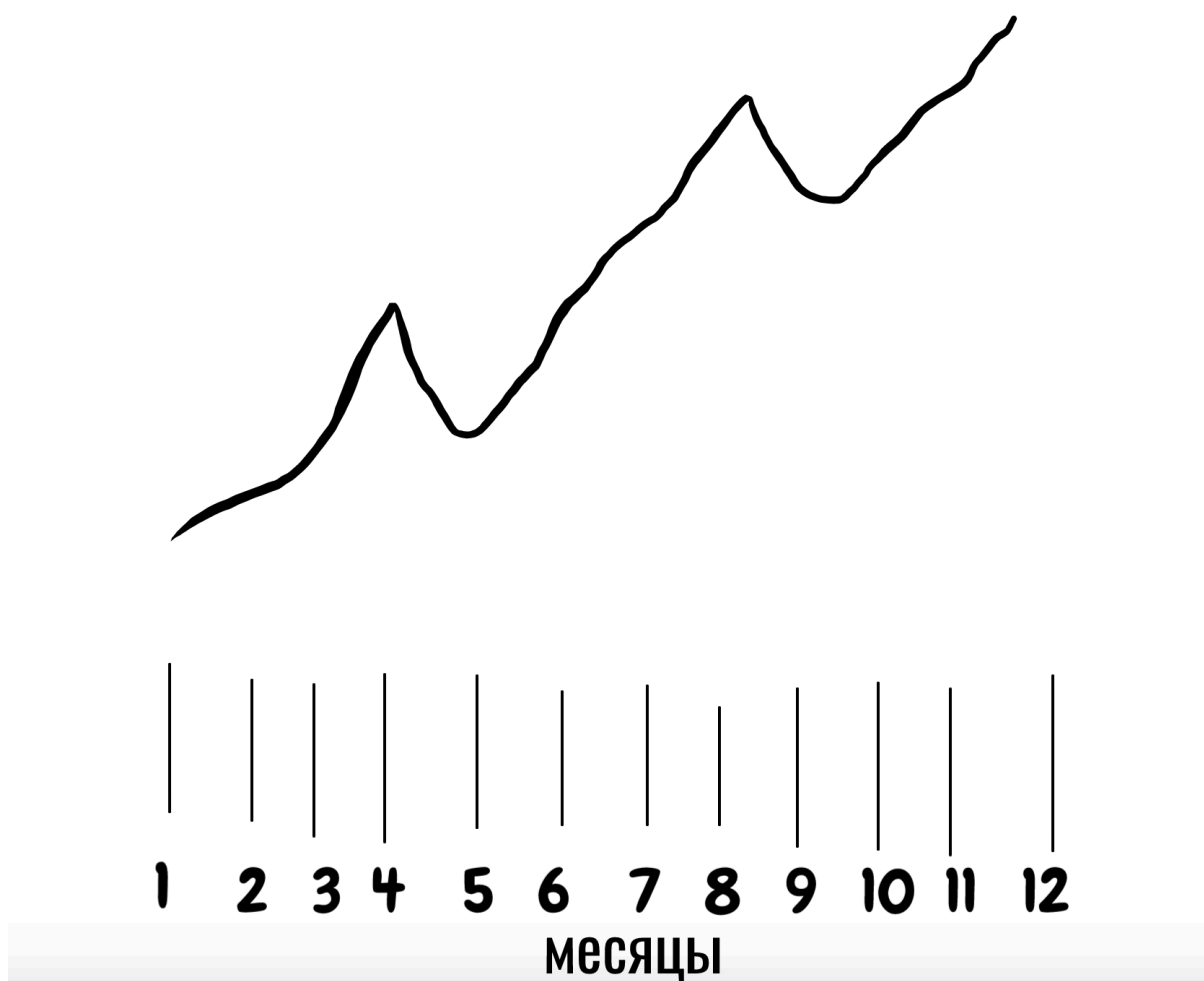
3.4.10.1.14. Ревеню (доход) Худшего месяца (и то, какой это был месяц - например - январь). Не учитываются первые 2-3. Также, если худший месяц попал сразу после начала (первые 3-6 мес.), то тоже не учитывать эти 3-6 мес.

ПРИМЕР:



Здесь худшим месяцем будет только 7-й, хотя по графику видно, что самые низкие значения - это первый-второй месяц


А вот здесь



<https://take.ms/Y2daD>

Минимум будет 9й- месяц, несмотря на то, что 5й - имеет значения меньше/ниже 9-го

(этот показатель не отображен в дизайн-макете, но его включение должно быть заложено в настройках Config view).

3.4.10.1.15.  Config view - при клике на него - всплывает окно с выбором того, какие столбцы и данные отображать в основной таблице. Пример окна с настройками - Подробнее о "Config view" было ранее [здесь](#)

3.4.10.2. Ниже ключевых показателей - отображается основная таблица с результатами:








<https://i.imgur.com/pW7ShlX.png>

3.4.10.3. Таблица состоит из следующих элементов:

3.4.10.3.1. Заголовки/хедер:

★ #	Product photo & name	Basic trends	Brand	Product Score	Sellers	Rank	Price	FBA Fees	Net Margin	Est.Sales ²	Est. Revenue	# of Reviews	▼ RPR ²	Rating	LQS	Available From	Seller
-----	----------------------	--------------	-------	---------------	---------	------	-------	----------	------------	------------------------	--------------	--------------	--------------------	--------	-----	----------------	--------

<https://imgur.com/a/aH3gG09>




- 3.4.10.3.1.1. Хедер таблицы включает функции:
- 3.4.10.3.1.1.1. Сортировка. При нажатии на какое либо название столбца - происходит сортировка по этому столбцу. 
- 3.4.10.3.1.1.2.  - отображается в правой части хедера таблицы. Отвечает за горизонтальную прокрутку/скролл.
- 3.4.10.3.1.1.2.1. Присутствует **только при условии, что в "Config view" выбрано более 29 столбцов** для отображения **или**, если текущие столбцы не влезают в 1 экран при размере экрана 15.4 дюйма. Подробнее о "Config view" было ранее [здесь](#)
- 3.4.10.3.1.1.3. Тултипы. На определенных столбцах присутствуют тултипы, при наведении на которые показываются всплывающие подсказки. Должна быть возможность создавать и редактировать их в админке.
- 3.4.10.3.1.1.3.1. Есть **2 типа тултипов:**
- 3.4.10.3.1.1.3.1.1. **С символом "?" (вопрос)** 
- 3.4.10.3.1.1.3.1.2. С подчеркиванием
- 3.4.10.3.2.
- 3.4.10.3.3. Каждую строку в таблице можно развернуть, нажав на соответствующую иконку: 
(<https://i.imgur.com/ZvHsDja.png>)
тогда график продаж (Basic trends) отображается в увеличенном виде.
- 3.4.10.3.4.  - свернуть строку
- 3.4.10.3.5.  - развернуть все строки разом
- 3.4.10.3.6.  - свернуть все строки разом

Строка

			2 Sets Round & Oval Banneton...		Swice		3	1	#27,759	\$17.99	\$7.52	57%	360	\$6,217	62	\$100	4.6	90	06/11/20	FBA
--	---	---	---------------------------------	---	-------	---	---	---	---------	---------	--------	-----	-----	---------	----	-------	-----	----	----------	-----

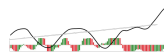
(<https://i.imgur.com/qA1nACx.png> & <https://imgur.com/A07inO6.png>) состоит из следующих элементов:

- 3.4.10.3.7.1.  - добавить в избранное
- 3.4.10.3.7.2.  - удалить из избранного
- 3.4.10.3.7.3. **2** - номер строки

- 3.4.10.3.7.3.1. При добавлении товара в блеклист - номера не сбрасываются. Пример: <https://i.imgur.com/dUSrRys.png>
- 3.4.10.3.7.3.1.1. Таким образом, в блеклисте отображаются номера именно те, которые были в основной таблице.
- 3.4.10.3.7.4.  - добавить товар в блеклист. Появляется при наведении на строку. Пример: <https://streamable.com/4ol1he>
Номера “виртуально” остаются теми же, в зависимости от сортировки и фильтров, но если какой-то из товаров в выводе добавлен в блеклист, то он просто исключается из таблицы, но целостность (integrity/continuity) нумерации сохраняется.
- 3.4.10.3.7.5.  - миниатюра картинки. При наведении - появляется увеличенная миниатюра, пример: <https://streamable.com/po5mm6>
- 3.4.10.3.7.5.1. При клике - подгружается следующая миниатюра из карточки товара.
- 3.4.10.3.7.6.  - появляется напротив товаров, у которых рост продаж был более **30%** за последний месяц.
- 3.4.10.3.7.6.1. При наведении на Hot - появляется тултип. Текст тултипа: “This product is considered HOT because its sales grew by 30% or more for the past month. [View](#) the yearly sales history of the product to check for yourself.” (без кавычек).
Пример:
- 3.4.10.3.7.6.1.1. При нажатии на [View](#) - появляется график продаж.
Пример:
<https://streamable.com/jv6fjk>
- 3.4.10.3.7.7. [2 Sets Round & Oval Banneton...](#) -
название/заголовок товара
- 3.4.10.3.7.7.1. Если название не влезит в ячейку, то оно обрезается, с добавлением “...” (без кавычек) в конце.
- 3.4.10.3.7.7.2. При наведении на название - отображается полный заголовок через ALT-тег.

3.4.10.3.7.7.3. При клике на название - открывается URL, ведущая на товар на Amazon.com

3.4.10.3.7.8.



- мини-график продаж (в уменьшенном виде). При наведении - отображается увеличенная миниатюра(увеличенная миниатюра не должна перекрывать уменьшенную миниатюру - чтобы уменьшенная миниатюра оставалась кликабельной/оставалась возможность на нее кликнуть - см. далее). По умолчанию (в т.ч. в миниатюре) используется 3 графика: средняя(сглаженная), общий тренд (линия) и индикатор в виде уменьшения/увеличения показателей - в форме осциллятора. Период по умолчанию - 1 год (если не выбрано иное в фильтрах).

При клике - откроется полноразмерный график с настройками. Пример макета полноразмерного графика и настроек:

<https://i.imgur.com/jlPPSrQ.png>

3.4.10.3.7.8.1. ТЗ для полноразмерного графика:

1. При наведении курсора на график - отображается дата и значения, при этом линия, к которому отображаются данные - выделяется более жирным/становится толще.
2. Курсор должен “прилипать” к ежемесячным значениям”, но при этом доступны и в промежутке - ежедневные значения, т.е. курсор несмотря на “примагничивание” к ежемесячным (символ треугольника), все же можно разместить внутри месяца и узнать дневные значения.
3. График Reviews/Отзывы - не должен выходить на слишком высокие значения относительно всего графика, в представленном графике-примере - это красная линия, где максимальное значение - это уже 5 звезд и выше - не может быть.
4. Все значения в полноразмерном графике:
 - a. Revenue
 - b. 7 Day Revenue Moving Average
 - c. Revenue Trend Line
 - d. Reviews
 - e. Worst seller Revenue {{то же, что и **MINIMUM Last 12M revenue**}}
 - f. 20% Lower Percentile Median Revenue
 - g. How many FAQ's [Вопросы и ответы покупателя (сколько)]
 - h. How many sizes/models [Какие и сколько размеров/вариантов]
 - i. Rating [Рейтинг, Звезд].
 - j. Sales (кол-во продаж) (если товар дешевый, то)
 - k. Кол-во продавцов по запросу
 - l. Кол-во продавцов по товару

- m. Сколько платных объявлений (узнать, можно ли выводить отдельно по продавцам).
- n. Рекламный бюджет, расходы (средние расходы на 1 объявление)
- o. Сколько переходов по рекламе
- p. Выхлоп/выручка с платной рекламы (если есть) (продажи именно с этой рекламы, исключая другие источники)
- q. Валовая прибыль с платной рекламы (если есть) (продажи именно с этой рекламы минус расходы на рекламу)
- r. Кол-во добавленных в избранное
- s. Показатель сезонности
- t. Кол-во возвратов, в динамике
- u. По качеству листинга
- v. Средняя цена за переход по платной рекламе

5. При этом последние используемые значения - поднимаются вверх.

- a. К примеру, если мы промотаем ниже "Price" (см. пример: <https://i.imgur.com/RyA3sGT.png>) - то там будут другие графики, и выбрав - он при следующем открытии будет наверху, если только перед закрытием графика он не был "отключен" (снята галочка)

3.4.10.3.7.8.2. Цвет чекбоксов соотносится с цветом линии графика

3.4.10.3.7.8.2.1. При наведении на чекбокс - подсвечивается график.

3.4.10.3.7.8.3. В графе "Activities" - отображаются вся активность связанная с этим запросом, включая: // Здесь вопрос к программисту - насколько сложно это реализовать и сколько будет стоить отдельно, именно эта фишка //

3.4.10.3.7.8.3.1. добавление новых листингов

3.4.10.3.7.8.3.2. резкие изменения продаж (более, чем на 3% за день)

3.4.10.3.7.8.3.3. Резкое изменение цены от 1% за 7 дней

3.4.10.3.7.8.3.4. Увеличение и уменьшение кол-ва заказов на более, чем 1% за 7 дней

3.4.10.3.7.8.3.5. Добавление и удаление новых продавцов

3.4.10.3.7.8.3.6. Добавление новых и удаление брендов

3.4.10.3.7.8.3.7. Добавления в избранное

- 3.4.10.3.7.8.3.8. Добавление новых и удаление товаров
- 3.4.10.3.7.8.3.9. Резкое изменение выручки - больше 1% за 7 дней

- 3.4.10.3.7.8.4. При клике на год - открывается пикер выбора дат.
- 3.4.10.3.7.8.5. При нажатии на "make histo" график превращается в гистограмму:
<https://i.imgur.com/X92WPIB.png>
- 3.4.10.3.7.8.6. При выборе чекбокса - выбранный график перемещается наверх.
- 3.4.10.3.7.8.7. Всего может быть до 8 открытых графиков, включая гистограмму. При выборе больше - появляется ошибка/предупреждение с модальным окном "Может быть добавлено не более 8 графиков. При добавлении больше - самый старый будет удален.", с кнопками "ОК" и "Отмена". При выборе ОК - удаляется самый старый график.
- 3.4.10.3.7.9. Swice - название бренда. При клике - открывается страница бренда на Amazon.com
- 3.4.10.3.7.10. 3 - Product Score for PL - индикатор, показывающий score для собственного брендинга. Алгоритм расчета см. в приложении #2.

* Смотреть тренд, растет ли он (динамика роста продаж, кол-во продаж). С Каждым новым днем чтобы продавалось больше, чем в предыдущие дни.

* Длительность тренда. Если меньше полугода, то плюс. (товары живут 1.5 года в РФ).

* Кол-во имеющихся брендов. Если много ноунеймов, то плюс.

* Смотреть у новичков, увеличивается ли прибыль после нанесения бренда или выгоднее продавать ноунейм.

*

- 3.4.10.3.7.10.1. При нажатии открывается окно с более детальной информацией:
<https://i.imgur.com/kOT2JGP.png>

- 3.4.10.3.7.11. **1** - кол-во продавцов точно такого же товара (точная копия).
- 3.4.10.3.7.12. **#27,759** - **внутренний ранг/рейтинг Amazon (позиция в рейтинге). Именно по нему делаются все остальные расчеты, включая кол-во продаж и доход.**
- 3.4.10.3.7.13. **\$17.99** - Price - цена продукта. При нажатии - открывается график с временным рядом (история) цены этого продукта.
- 3.4.10.3.7.14. **\$7.52** - FBA Fees - сумма, которую Amazon берет за обработку 1 заказа, если он продан через FBA. (данный показатель доступен в Amazon-калькуляторе - присутствует в API). При клике - открывается калькулятор прибыльности. Пример макета: <https://i.imgur.com/ukkwjnz.png>
Пример работы калькулятора: <https://streamable.com/96yjue>
(для проверки работы у конкурентов - установите расширение для Chrome через сайт <https://amzscout.net/> (введите email, далее нажмите Try it for free и скачайте с Chrome маркетплейса; затем зайдите на Amazon.com, введите в поиск любое название товара - например "game" и нажмите на иконку расширения в панели расширений браузера Chrome)
- 3.4.10.3.7.15. **57%** - Net Margin - чистая маржа. Рассчитывается, как разница между Price и FBA Fees, выраженная в % (разница между ними в %).
- 3.4.10.3.7.16. **360** - Est.Sales - количество продаваемых единиц товара за месяц. Значение основано на ранге (позиции в рейтинге) продукта. При нажатии - открывается график с временным рядом (история) цены этого продукта.
- 3.4.10.3.7.17. **\$6,217** - Est. Revenue - выручка/доход по этой позиции за месяц. Вычисляется умножением Цены(Price) на Кол-во продаж(Est.Sales).
- 3.4.10.3.7.18. **62** - # of Reviews - кол-во отзывов на данный продукт.

- 3.4.10.3.7.19. **\$100** - RPR(Revenue Per Review) - Это соотношение того, сколько дохода получает продавец на один отзыв (общая выручка делится на кол-во отзывов).
- 3.4.10.3.7.20. **4.6** - Rating - Рейтинг товара (кол-во звезд/оценка товара):
<https://i.imgur.com/fPCrYyh.png>
- 3.4.10.3.7.21. **69** - LQS(Listing Quality Score) - показывает качество листинга. При клике - появляется окно с пояснением:
<https://i.imgur.com/jqFbuqM.png>
- 3.4.10.3.7.21.1. В левой колонке расположены значения, которые изменяются в зависимости от листинга (товара/продукта). Т.е. эти значения берутся непосредственно из карточки товара на Amazon.com.
Скриншот:
<https://i.imgur.com/3vKFbIV.png>
// Здесь вопрос к программисту - насколько сложно это реализовать и сколько будет стоить отдельно, именно эта фишка //
- 3.4.10.3.7.21.1.1. Name length - длина заголовка товара/листинга (кол-во символов). Пример:
<https://i.imgur.com/oT2r3fA.jpeg>
Если значение меньше рекомендуемого - отображается зеленым шрифтом. Больше - красным.
- 3.4.10.3.7.21.1.2. Brand length - кол-во символов в названии бренда товара:
<https://i.imgur.com/gHK8l5y.jpg>
Если значение меньше рекомендуемого - отображается зеленым шрифтом. Больше - красным.
- 3.4.10.3.7.21.1.3. Bullets count - кол-во "буллетов" в списке к описанию товара.
Скриншот:
<https://i.imgur.com/BznGBGm.jpg>
Если значение больше или равно рекомендуемому -

- отображается зеленым шрифтом. Меньше - красным.
- 3.4.10.3.7.21.1.4. Rating - Рейтинг товара (кол-во звезд/оценка товара):
<https://i.imgur.com/fPCrYyh.png>
 Если значение больше или равно рекомендуемому - отображается зеленым шрифтом. Меньше - красным.
- 3.4.10.3.7.21.1.5. Reviews - кол-во отзывов на данный продукт:
<https://i.imgur.com/73bZfkF.png>
 Если значение больше или равно рекомендуемому - отображается зеленым шрифтом. Меньше - красным.
- 3.4.10.3.7.21.1.6. Description length - длина описания. Если значение больше или равно рекомендуемому - отображается зеленым шрифтом. Меньше - красным.
- 3.4.10.3.7.21.1.7. Seller type - тип продавца:
<https://i.imgur.com/4knaAQm.png>
 Если рекомендуемые значения - FBA(FullfilledByAmazon) - зеленым. Если не соответствует рекомендуемыми значениям т.е. - Amazon или MCH - красным.
- 3.4.10.3.7.21.1.8. Image count - кол-во картинок к товару. Если значение больше или равно рекомендуемому - отображается зеленым шрифтом. Меньше - красным.
- 3.4.10.3.7.21.2. Алгоритм расчета качества листинга нужно придумать. Пример:
<https://www.tool4seller.com/images/scor e/how-to-use-img2.png>
 (см. сайт <https://tool4seller.com/> - его нужно посмотреть, в качестве примера)

В целом расчет примерно такой: на каждое "соответствие" рекомендуемыми значениям - 12.5 баллов. Всего 100 баллов и 8 значений. Допустим, если выполнено

5 условий - умножаем их на $12.5 = 62$ балла (округляем до меньшего). Но формулу нужно дорабатывать, поскольку у тех же конкурентов, если условие к примеру по отзывам - рейтинг 4 минимум - выполнено на "4", то будет одна цифра баллов качества листинга. А если 5, то будет больше баллов, что логично. Т.е. можно добавить какой-то множитель на "превышение" рекомендуемых значений, главное, чтобы этот множитель лимитировался и не получилось так, что у нас не будет выполнено других условий, а только будет длиннее описание (намного больше рекомендуемого) и за это начисляется 100 баллов...

3.4.10.3.7.21.3. Фильтр по качеству листинга должен быть также в общих фильтрах.

3.4.10.3.7.21.3.1.

3.4.10.3.7.21.3.2.

3.4.10.3.7.21.4. В правой колонке - не меняющиеся значения.

Скриншот:

<https://i.imgur.com/3vKFbIV.png>

3.4.10.3.7.21.5.

3.4.10.3.7.21.5.1.

3.4.10.3.7.22. 06/11/20 - дата создания листинга.

3.4.10.3.7.23. FBA^с - тип продавца. Может быть: FBA, MCH, AMZ

3.4.10.3.7.24.

3.4.10.3.7.25.

3.4.10.3.8.

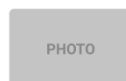
3.4.10.3.9.

3.4.10.4. Развернутая строка:

<https://i.imgur.com/P3dPomV.png>

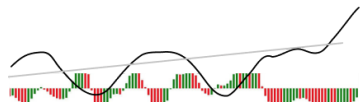
состоит в основном из тех же элементов, что и свернутая, но:

3.4.10.4.1. Превью товара/фото - в увеличенном виде:




3.4.10.4.1.1. (название товара - наоборот сокращается, но при наведении на него - показывается в виде "бегущей строки")

3.4.10.4.2. График продаж - в увеличенном виде:



- состоит из двух элементов:

3.4.10.4.2.1.  - собственно сам график продаж.

3.4.10.4.2.2.  - линия направления тренда, усредненное направление продаж (общее направление графика).


3.4.10.4.2.3.


3.4.10.4.2.4. Добавить при наведении какую-то кнопку, которая бы открывала выбор того, по каким именно показателям (выручка, продажи, кол-во отзывов, изменения цены и т.д.) - показывать тренды в миниатюрах. При применении настройки - во всех миниатюрах будут отображаться именно эти показатели, с этими данными.

3.4.10.4.3. Вспомогательная панель:

<https://i.imgur.com/79aDjrw.png>

3.4.10.4.3.1.  Add to Tracker - Add to tracker - добавить


3.4.10.4.3.2.  Profit Calculator - калькулятор прибыльности. Пример макета был [ранее](#)

3.4.10.4.3.3.  Product History - история продаж. При клике открывается график.

3.4.10.4.3.4.  Get keywords - список ключевых запросов.


Пример макета:

<https://i.imgur.com/wWkGBIf.png>

3.4.10.4.3.4.1.  Export to CSV - экспортировать доступный (тот, что отображается пользователю) список ключевых слов в CSV таблицу. Пример таблицы по этим же ключевым словам, что и в предыдущем пункте:
<https://drive.google.com/file/d/1HaZblGfgKamjcfoXZY0W6BWlhMC0q5Hp/view?usp=sharing>

3.4.10.4.3.4.2. Важно! В экспортированном списке отображаются именно ключевые слова, а не запросы. Чтобы увидеть больше примеров - можно зарегистрироваться у конкурента - <https://amzscout.net/> (либо напишите заказчику - он предоставит).

3.4.10.4.3.4.2.1. При этом слова должны быть отсортированы по частоте употребления(что видно из примера выше, в таблице).


3.4.10.4.3.4.3.  - закрыть окно со списком ключевых запросов.


3.4.10.4.3.4.4.  - сам ключевой запрос.


3.4.10.4.3.4.4.1. При наведении на него - меняется цвет, как при наведении на гиперссылку.

Пример: 

3.4.10.4.3.4.4.2. При клике на запрос - открывается сайт Амазон, где уже вбит данный ключевой запрос в поиск. Пример ссылки:
<https://www.amazon.com/s?k=motorcycle+gps>

3.4.10.4.3.5.  Find on AliBaba - найти продукт на Алибаба. Открывает поиск на Alibaba.com по картинке товара.

3.4.10.4.3.6.  Copy ASIN - скопировать ASIN продукта


3.4.10.4.3.7.  Full view - открывает отдельную страницу по товару, содержащую сразу всю информацию о конкретном товаре (см. пример макета
<https://i.imgur.com/PFEOaom.png>)

3.4.10.4.4.

3.5. **TOP** - ТОП. Состоит из аналогичной, как в Search таблице, но в упрощенном виде:
<https://i.imgur.com/7Lvup2l.png>

Его цель - показать примерно те же данные, но выводить их в немного другом виде, более понятном для другой категории пользователей (кому не нравятся гибкие, но сложные настройки - а нравится простота и скорость доступа к информации). Здесь отображается список позиций уже отсортированных по кол-ву продаж за определенный диапазон времени. Также есть возможность сделать сортировку по другим критериям, что отображено на макете.

3.5.1.  - быстрый выбор временного промежутка

3.5.2.  - открывает фильтр по дате: <https://i.imgur.com/hyhzXgS.png>

3.5.3.

 product/brand name

ПОИСК

- поле для фильтрации

по ключевым словам/бренду.

3.5.4.

Основная таблица:

<https://i.imgur.com/XIIQgmB.png>

Почти копия по содержанию, как таблица со страницы Search, только в упрощенном виде:

3.5.4.1. Названия столбцов:

Фото Название

Наличие

Отзывы

Оценка

Цена

Прирост продаж в %

Прирост продаж

Продажи %

Дата добавления

Тренд

FBA ☒

FBM ☐

3.5.4.2.



- сортировка.

3.5.4.3.



- миниатюра фото товара. При наведении - появляется увеличенная миниатюра, пример:

<https://streamable.com/po5mm6>

3.5.4.3.1.

При клике - подгружается следующая миниатюра из карточки товара.

3.5.4.4.

9 Inch Proofing Basket, WERTIOO Bread Proofing Basket + Bread Lame + Dough Scraper + Linen Liner Cloth for Professional & Home Bakers

- название товара

3.5.4.4.1.

Если название не влезает в ячейку, то оно обрезается, с добавлением "..." (без кавычек) в конце.

3.5.4.4.2.

При клике на название - открывается URL, ведущая на товар на Amazon.com

3.5.4.5.

6580

- Наличие - кол-во товара у продавца.

3.5.4.6.

1785

- Отзывы - кол-во отзывов на товар.

3.5.4.7.

5.0

- Rating - Рейтинг товара (кол-во звезд/оценка товара):

<https://i.imgur.com/fPCrYyh.png>

3.5.4.8.

\$199.95

- Price - цена продукта. При нажатии - открывается график с временным рядом (история) цены этого продукта.

3.5.4.9.

+500%

- прирост продаж за выбранный промежуток времени, в %.

3.5.4.10.

+25400

- прирост продаж за выбранный промежуток времени, в количественном выражении.

3.5.4.11.

90255

- количество продаж товара за месяц (сколько было продано единиц). Значение основано на ранге (позиции в рейтинге) продукта. При нажатии - открывается график с временным рядом (история) цены этого продукта.

3.5.4.12.

7/17/2021

- дата добавления товара в листинг.


3.5.4.13.



- тренд продаж за выбранный промежуток времени.

При наведении - миниатюра увеличивается. При клике - открывается полноэкранный график с выбором того же

периода и тех же показателей с “х” крестиком (красным) - чтобы была возможность сбросить все фильтры + было более очевидно, что выбраны именно эти фильтры и данные показываются именно по этим критериям.

3.5.4.14.  - тип продавца. Может быть: FBA, MCH/FBM

FBA - это Амазон сам хранит и упаковывает товар. FBM - продавец сам хранит и упаковывает, просто размещается/продает через Амазон. У вайлдберрис и других маркетплейсов - тоже такое же разделение есть, поэтому мы здесь также разделили.

3.5.5.

3.6. Выделенная страница/окно, содержащая всю информацию о товаре. Открывается при клике на “Full view” в более детальном меню к товару: <https://i.imgur.com/PFEOaom.png>

Страница состоит из нескольких экранов:

3.6.1. 1 экран: <https://i.imgur.com/PT8zKtc.png>

3.6.1.1. Заголовок/название товара:

<https://i.imgur.com/o2EQjmW.png>

(то же, что было и в таблице, описание [здесь](#))

3.6.1.2. Фото-превью товара: <https://i.imgur.com/uTdPzqc.png>

Состоит из двух частей:

3.6.1.2.1. Само превью: <https://i.imgur.com/ofCBJHL.png>


3.6.1.2.2. Быстрый выбор из всех превью:

<https://i.imgur.com/wk1c5XI.png>

(при наведении - сразу идет переключение на это превью).


3.6.1.3. <https://i.imgur.com/12qt0or.png> - текст описания товара.

Берется из: <https://i.imgur.com/A1PfhcL.jpg>

3.6.1.3.1.  - разворачивает более полное описание товара.

3.6.1.4. **Price: \$33.89** - цена товара. То же, что и [здесь](#)

3.6.1.5. **Electronics › Computers & Accessories › Computer Components** - путь до категории товара.

3.6.1.6.  - кнопка, которая ведет на товар на самом Amazon.com (т.е. это обычная ссылка).


3.6.1.6.1. Ссылка должна подставляться реферальная, привязана к нашему аккаунту Amazon.

3.6.1.7. Вспомогательная панель. Полная копия Вспомогательной панели из основной таблицы (см. [здесь](#))


3.6.1.8.  - полная копия фичи из основной таблицы (см. [здесь](#))

Our Score

1000

3.6.1.9.  - полная копия фичи из основной таблицы (см. [здесь](#))

3.6.1.10.  - лайк/дизлайк товара

- 3.6.1.11.  Discuss - кнопка Discuss - относится к фиче чата/мини-форума со страницы Search (см. [здесь](#)), работает по тому же принципу, за исключением:
- 3.6.1.11.1. При нажатии Discuss - происходит автоматический скролл ниже, в секцию комментариев; и когда пользователь после оставляет хотя бы 1 коммент./пост - создается топик с этим товаром, и он виден на форуме, в последних постах. Также эта инфа добавляется в Activities (см. [здесь](#)) - в полноразмерном графике.
- 3.6.2. 2 экран: <https://i.imgur.com/REaaRhj.png> - плитки с отображением сразу всех показателей по товару (все те же показатели, что в основной таблице на странице Search, см. [здесь](#)).
- 3.6.2.1. Чуть выше плиток отображаются элементы управления отображением плиток: <https://i.imgur.com/1H4sXX1.png>
 - 3.6.2.1.1. Config view - открывается окно, как при выборе столбцов в основной таблице на странице Search (см. [здесь](#)).
 - 3.6.2.1.2. Show all (при выборе становится явно активным/"нажатым". Когда выбрано Config view - частично)
 - 3.6.2.2. Плитки можно перетаскивать в удобный вид
 - 3.6.2.2.1. При перетаскивании курсор меняется на "руку": <https://i.imgur.com/yEJzXt4.png>
 - 3.6.2.2.2. При перетаскивании должен быть эффект "фантома", с границами в виде пунктирных линий, см. пример: <https://streamable.com/jz6aip>
 - 3.6.2.2.3. При наведении на перетаскиваемые плитки - курсор должен меняться, пример: <https://i.imgur.com/etJ8DUH.png>
 - 3.6.2.2.4. При перетаскивании плитка должна быть под углом, пример [1, 2]
 - 3.6.2.2.5. Должно быть размытие фона той плитки, которую мы перетаскиваем, пример [1, 2]
 - 3.6.2.2.6. При "значительном" перетаскивании - идет замена одной плитки на другую.
 - 3.6.2.2.7. Все должно быть на чистом HTML5, например: <https://codepen.io/acauamontiel/full/NPBWyM>
 - 3.6.2.3. Должна быть возможность регулировать, какие показатели отображать.
 - 3.6.2.4. Вид должен сохраняться.
- 3.6.3. 2 экран (продолжение):
- 3.6.3.1. <https://i.imgur.com/aGxe3Ib.png> - тот же полноразмерный, что и на странице Search в основной таблице (см. [здесь](#)), но в более увеличенном виде

- 3.6.3.1.1. и Activities отображается поверх графика, пример:
<https://i.imgur.com/ygOH86V.png>
- 3.6.4. 3 экран: <https://i.imgur.com/7PYf2Kr.png> - комментарии к товару. Тот же функционал, что и при Discuss (см. [здесь](#), а также и [здесь](#))
- 3.7. Pricing. Раздел Тарифы. Здесь пользователь может видеть свой тариф и может его изменить, выбрав из нескольких основных тарифов. Тарифная сетка. Текст, картинки и ссылки на кнопках, ведущие к оформлению заказа.
- 3.8. About. Просто статичный текст и картинки.
- 3.9. ЛК. Профиль и т.д. .

Раздел Профиль - здесь пользователь может:

1. Изменить имя пользователя отображаемое на форуме

2. Изменить аватар (он отображается на форуме и в верхнем меню).

3. Пароль

4. Email

5. Добавить/изменить авторизацию через соц. Сети

6. Если изначально настроена авторизация только через соц сети - должна быть возможность включить авторизацию по паролю и email.

7. Управление подписками и уведомлениями (см макеты и приложение к ТЗ). Здесь нужны макеты по аналогии с Data camp - см док маркетинг lead magnet + нужно придумать интерфейс управления подписками через мессенджеры и соцсети.

4. Админ-часть.

4.1. Тарифы.

4.1.1. Биллинг

4.2. Пользователи

4.2.1. их тарифы

4.2.2. имя...

4.2.3. email...

4.2.4. возможность зайти в их аккаунт

4.3. Лимиты

4.3.1. лимиты

4.3.1.1. возможность настроить лимиты и поведение системы при их достижении (например, троттлинг времени загрузки и/или обновления[как часто обновлять])...

4.3.2. пользователи с высокой и пограничной нагрузкой

4.3.3. индивидуальные лимиты

4.4. Тултипы. Приветствуются готовые решения (библиотеки или даже CMS [для вордпресс есть плагин редактора тултипов, который практически полностью удовлетворяет требованиям]; например, на уровне CMS

можно было бы управлять содержимым тултипов, а на уровне движка сервиса просто вставлять эти тултипы в нужные места).

5. Общие требования

- Обязательная визуальная поддержка действий пользователя – «интерактив» (визуальное отображение активных, пассивных элементов [ссылок/кнопок/полей] при обычном состоянии, наведении и клике - в отдельности). При этом, если в верстке/макетах не представлены варианты дизайна активных/пассивных элементов, то используется алгоритм: если элемент/фон светлый - при наведении элемент делая на X% темнее, если элемент/фон темный, то элемент делая на Y% светлее. Также при наведении добавляется легкая тень, а при нажатии - тень исчезает. При нажатии элемент становится еще темнее, чем при наведении. Если элемент на светлом фоне итак слишком темный (определяется, как все, что темнее определенного порога. 1. Переводим цвет HTML -> RGB -> HSL 2. Получаем: H - цветовой тон (например, красный, зелёный или сине-голубой), S - насыщенность, L - светлота (именно светлота, а не яркость) от 0 до 1. Пример: $\text{var light} = (\text{this.r} * 0.8 + \text{this.g} + \text{this.b} * 0.2) / 510 * 100;$. Вот еще примеры реализации: <https://phpclub.ru/talk/threads/Цвет-тёмный-или-светлый-Определение-яркости-цвета-по-его-XXXXXX-значению.40961/>
Если элемент на темном фоне - то тень не добавляется. Также должна быть возможность каким-то тегом помечать элементы, где делается исключение для теней при наведении - актуально для сложных меню, например: <https://take.ms/fkbnH>
На этих кнопках (скорее даже вкладках) нет смысла отображать тень.
- Мета-теги и контент сайта на этапе изготовления сайта настроены
- На любые некорректные действия посетителя, связанные с вводом неверных данных, незаполнением обязательных полей ввода в формах и прочие, которые могут быть обработаны системой, генерируются соответствующие сообщения об ошибках на русском языке, в рамках общего дизайна сайта.
- Должны быть адекватные ограничения на кол-во и тип символов во всех полях, при этом разные, в зависимости от контекста: для имени и email ограничение естественно сильнее, чем для пароля.
Имя: 30 символов
Email: 50 символов
Пароль: 50 символов
- Пустые или выходящие за лимиты по кол-ву/типу символов поля, должны обрабатываться в JS на стороне юзера; невозможно отправить форму с пустыми или выходящие за

лимиты по кол-ву/типу символов полями (усложнение брутфорса и DDoS-атак).

- Основная валюта сайта - \$
- Валюта учета - \$
- SEO-friendly URL. SEF ссылки.
- Интеграция с платежной системой и рекуррентные платежи

6. Конкретные требования

- 6.1. Бэкенд проекта разрабатывается на Express.js/FastAPI/Django (на усмотрение разработчика), а фронтенд - на ReactJS/Angular(также на усмотрение). Часть функционала можно разрабатывать средствами Wordpress (headless CMS), а также плагинами. Приветствуются дополняющие расширения/аддоны, но их добавление должно быть согласовано с заказчиком (это должны быть популярные, поддерживаемые разработчиками решения с высоким рейтингом).
- 6.2. Запрещено модифицировать код ядра WP.
- 6.3. Все модификации и добавление нового функционала должны реализовываться через создание новых, кастомных плагинов и через взаимосвязь новых написанных плагинов с предустановленными (для реализации нового функционала из ТЗ возможно использование также дополнительных сторонних готовых плагинов, но без вмешательства в те плагины, список которых приведен в данном документе). Запрещено вносить какие-либо изменения в код плагинов других авторов, в том числе и в их аддоны/расширения. Данное требование необходимо для исключения массовых проблем при обновлении плагинов и их аддонов в будущем.
- 6.4. Там, где это возможно - необходимо по максимуму использовать REST API от woocommerce и других плагинов, REST API самого вордпресса, а также вебхуки. В остальных случаях, под недостающий функционал пишутся свои плагины/находятся готовые решения (эти решения должны быть согласованы, см. выше).
- 6.5. В рамках работ должен быть выпущен TLS сертификат Let's Encrypt и настроен на автоперевыпуск. Let's Encrypt — центр сертификации, предоставляющий бесплатные криптографические сертификаты X.509 для TLS-шифрования. Процесс выдачи сертификатов в Let's Encrypt полностью автоматизирован. Есть API:
<https://letsencrypt.org/ru/docs/acme-protocol-updates/>
Так как данный сертификат рассчитан на срок не более 90 дней, то возможно необходима соответствующая задача cron, чтобы сертификаты обновлялись в автоматическом режиме.
- 6.6. Цвет фона админки по умолчанию F9FAFC
- 6.7. Для упрощения разработки допускается "прятать" лишние поля плагинов через CSS или другим "костылем", не меняя код бэкенда.
- 6.8. Мультиязычность. Сайт должен поддерживать мультиязычность, первоначально русскую и английскую версию с возможностью добавления новых языков. Плагины, как и кастомная админка — должны иметь возможность перевода через WPML, который устанавливается на

~~сервере заказчика~~

~~Перевод осуществляется силами Заказчика через визуальную админку и не должен требовать специальных знаний и навыков.~~

7. ТЗ на верстку.

Верстка и наполнение сайта информационными и графическими материалами.

Исполнитель разрабатывает сайт на основании утвержденной дизайн-концепции, “резиновой” (респонсив) верстки, обеспечивающей корректную работу сайта на всех компьютерных и мобильных устройствах.

Заказчик предоставляет все необходимые Исполнителю текстовые и графические материалы в удобном, для последующей обработки, виде, а также комментарии, касающиеся их содержания, объема, оформления и размещения.

Исходные материалы в виде текстов и изображений для информационного наполнения сайта предоставляются Заказчиком.

Требования к оформлению Сайта.

Необходимо использовать в дизайне графические решения, паттерны, цветовые схемы, разработанные в рамках фирменного стиля.

Запрещается использование в дизайне сайта Flash-элементов. Необходимо использовать анимационные эффекты на основе технологий JS, JQ, HTML5 и CSS3.

Шрифты, используемые для оформления текстовых материалов сайта, не должны противоречить разработанному фирменному стилю. В случае отсутствия необходимых шрифтов на компьютере пользователя необходимо предусмотреть использование стандартных групп шрифтов браузеров (Arial/Helvetica, Times New Roman, Courier) таким образом, чтобы замена шрифтов из соответствующей группы не приводила к визуальному искажению текста.

7.1. Десктопная версия

7.1.1. Базовые требования

- 7.1.1.1. Верстка темы WordPress, желательно на основе шаблона WP фреймворка Genesis.
- 7.1.1.2. **Платформы:** Windows, MacOS.
- 7.1.1.3. Соответствие рекомендациям W3C.
- 7.1.1.4. Код должен быть семантически корректным.
- 7.1.1.5. **Стандарты:** HTML5/CSS3. HTML - должен проходить валидацию. CSS - не обязательно. Для определения поддержки HTML5 используется [Modernizr](#).
- 7.1.1.6. Документ должен корректно отображаться в большинстве современных браузеров.
- 7.1.1.7. Верстка должна быть «резиновая» (responsive), то есть тянется по всей ширине экрана.

- 7.1.1.8. Оптимизация (максимальное качество) под разрешение экрана 1366×768, и хорошее отображение при любых других разрешениях.
- 7.1.1.9. Приемлемое отображение и доступность всех функциональных элементов при отключении пользователем JavaScript и/или картинок, а также при увеличении/уменьшении шрифта (в том числе не блокировать изменение шрифта).
- 7.1.1.10. Семейство шрифтов: Helvetica neue CYR (предпочтительно), Helvetica neue, Montserrat, Helvetica Now, Roboto, Open Sans. Можно указать и другие популярные шрифты с засечками.
- 7.1.1.11. Верстка должна быть идентична дизайну в наиболее распространенных браузерах (Chrome, Firefox, Safari, Internet Explorer & Edge, Opera). Для проверки используется BrowserStack
- 7.1.1.12. Использование хаков крайне нежелательно, но допустимо (в разумных количествах).
- 7.1.1.13. Все стили должны быть прописаны во внешних файлах.
- 7.1.1.14. Структуру меню, цвета и стиль ссылок делать в соответствии с ТЗ на сайт и рекомендациями к дизайну
- 7.1.1.15. Полностью блочная верстка структуры
- 7.1.1.16. Использование «распорок» недопустимо.
- 7.1.1.17. Иерархию кода нужно выделять табуляцией и пустыми строками.
- 7.1.1.18. Все элементы должны быть закрытыми. Теги, не имеющие закрывающего тега, завершаются «/». Например: `
`, ``. Тег `<base>` закрывается `</base>`.
- 7.1.1.19. Недопустим «перехлест» тегов. Например:
``
- 7.1.1.20. Недопустимо «разрывать» теги:
``
- 7.1.1.21. Из повторяющихся однообразных участков в блоках достаточно сверстать только первый и размножить его для проверки корректного заполнения контейнера.
- 7.1.1.22. Если присутствует текст, нужно проверить на корректное поведение большого его объема.
- 7.1.1.23. Значения атрибутов нужно заключать в кавычки.
- 7.1.1.24. Формат кавычек, в тегах: `"..."` (двойные), в русском тексте: «...» (ёлочка), в скриптах: `'...'` (одиночные). Допустимо экранирование.
- 7.1.1.25. Ссылкам нужно прописывать атрибуты `href="/"` и `title="/"`.
- 7.1.1.26. Всем изображениям нужно прописывать атрибут `alt`. Если изображение не является ключевым, например -

элемент навигации, значение атрибута нужно делать пустым.

- 7.1.1.27. Формат имён:
причастиеПрилагательныеСуществительное, англ. язык (используйте словарь), без транслита. Слова выделяются заглавными буквами, первый символ всегда строчная буква. Пример: shortNews, mainMenu.
- 7.1.1.28. Областям с фоновым изображением нужно устанавливать соответствующий цвет фона близкий по тону бэкграунда, чтобы информация на переднем плане была различима без подгруженного изображения.
- 7.1.1.29. Изображения размещаются в директории img/. Порядок формирования имён указан выше.
- 7.1.1.30. При необходимости разместить скрипты или описания стилей, делать это нужно или подключаемыми файлами, или используя CDATA:

```
<script type="text/javascript">  
<![CDATA[  
... unescaped script content ...  
]]>  
</script>
```

7.1.2. CSS Стили

- 7.1.2.1. Всё оформление нужно выносить в описания стилей. Использование атрибута style нежелательно. Использование атрибутов vspace, align, cellpadding, cellspacing, border, width, height и им подобных недопустимо.
- 7.1.2.2. Таблицы стилей размещаются в директории css/. Основные таблицы стилей именуются css/init.css, css/web.css, css/print.css.
- 7.1.2.3. Обязательно использование обнуляющих CSS.
- 7.1.2.4. Селекторы сортируются по алфавиту без участия точек и решеток, группы селекторов разделяются пустыми строками. Описание селектора { ... } отделяется табуляцией и размещается в одной строке. Свойства, по возможности, нужно также располагать по алфавиту.
- 7.1.2.5. Имена селекторов должны быть зависимыми от содержимого блока, а не от оформления, формы или относительного расположения. Например, неправильно: redParrot, greenSearchBox, bottomPart; правильно: bullet, quickSearch, footer.
- 7.1.2.6. Иерархия блоков и элементов блока должна поддерживаться иерархией селекторов. Пример ниже:

```
<div class="news">  
<div class="date">01.01.1980</div>  
<h1>Новость.</h1>
```

```
<div class="body">Текст новости.</div>
</div>
```

```
.news { ... }
.news .body { ... }
.news .date { ... }
.news h1 { ... }
```

7.1.2.7. Нежелательно «клонирование имён», например для news: newsDate, newsHead, newsBody.

7.1.2.8. Назначение тегу нескольких классов предпочтительнее «обёрток» и клонирования имён. Пример:

```
.person { ... }
.gone { color:#aaa; }
.present { color:#222; }
<div class="person gone">...</div>
<div class="person present">...</div>
```

7.1.2.9. Предпочтительно группировать значения свойств, например: margin:10px 5px 0px 5px; вместо margin-bottom:5px; margin-left:5px; margin-top:10px;.

7.1.2.10. У любого размерного значения, в том числе равного нулю, указание единиц измерения обязательно.

7.1.2.11. Предпочтительны относительные единицы измерения.

7.1.2.12. Последнее свойство в описании класса должно завершаться точкой с запятой.

7.1.2.13. Порядок модификаторов для ссылок: link, visited, hover, active (love-hate).

8. Требования к нагрузоустойчивости и быстродействию

8.1. Все запросы должны [кэшироваться](#). Не должно быть ни одного некешируемого запроса (кроме некоторых исключений: если предполагается использование WordPress, то в WooCommerce, где многие страницы некешируемые - корзина, оплата, профиль и т.д. - в этом случае используются вспомогательные плагины, работающие с AJAX запросами). При запросе в функционале аналитики - нет необходимости доставлять данные в реальном времени - достаточно обновлений минимум раз в несколько часов(и то, только при запросе пользователя) или даже дней.

8.2. Браузерное (клиентское) кэширование

8.2.1. Truly static content (global CSS styles, logos, etc.) - **access plus 1 year**

8.2.2. Everything else - **access plus 1 week**

8.2.3. Использование бесплатного CDN Cloudflare

9. Требования к надежности сайта

9.1. Сайт должен предусматривать базовую защиту от основных видов атак: межсайтового скриптинга (XSS), SQL-инъекций, CSRF-уязвимостей.

9.2. Если предполагается использование WordPress, то необходимо соблюдение стандартных требований безопасности WordPress, см.

отдельный документ:

<https://docs.google.com/document/d/1djEKIMTGLdsqoNDao5jXTdOSaNc0kll6UTddklBKVQg/edit>

10. **Тестирование и корректировка тестовой версии сайта.**

Тестирование сайта производится совместными силами Заказчика и Исполнителя. Заказчик имеет право привлекать к тестированию и баг-репорту третьих лиц (тестируемых). Баг-репорт (сообщения об ошибках) осуществляется в системе баг-трекере Jira с помощью bugdigger, marker.io или аналогичных инструментов (bug report chrome extension).

Срок выполнения работ **по тестированию**: 1 неделя.