**ТЗ. два варианта реализации MVP - сервиса поиска подарка**

**По оценке нужны цифры, сколько будет стоить разработка (бекэнда и частично фронта «без дизайна») варианта I и варианта II. И сколько времени ориентировочно уйдет на каждый.**

Функционал делаем двумя версиями: первая только с самым важным, но с учетом возможности реализации второй. И вторая, со всем описанным в ТЗ (выделено серым цветом).  
Но возможно второй версии не будет, если первая не покажет положительных результатов.

Функциональные требования к MVP (что хотим получить в итоге)

1. Цель функционала - клон Я.Подарков, но только ссылки будут на конкретные магазины. И другой список товаров. А в дальнейшем, если сервис будет востребован, вместо ссылки с "товара" сразу на магазин, будет промежуточная страница-агрегатор. На этой странице будет свой заголовок (h1), meta-поля, возможно текст описания и список магазинов.

2. Функциональные требования по фронту. Форма подбора как у Маркета, только вместо ссылок на товары будут ссылки на сторонние магазины. По интерфейсу блоки будут точно такие же, только графически дизайн отличается.

3. Функциональные требования по бэк-енду. Возможность указывать произвольные поля (изначально заполняем парсингом), листинг всех "товаров", быстрое и наглядное редактирование полей товаров. А также возможность в дальнейшем указать товару несколько ссылок.  
И ещё: возможность результатам поиска создать страницу с собственным URL и заполнить поля: meta, h1, уникальное описание. Или хотя бы возможность реализовать это в дальнейшем, для первой версии не обязательно.

4. Сценарии взаимодействия юзера с фронтом. Их 2:  
4.1. Юзер попадает на главную страницу. Пользуется блоком подбора, смотрит-выбирает результаты в листинге, кликает на 1 или несколько товаров. Уходит изучать подробнее или покупать понравившийся.  
4.2. Юзер попадает уже на страницу подбора, где в блоке выбраны некоторые параметры. Этот сценарий основной. Здесь юзер может или смотреть/выбирать товары, или/и дополнительно уточнять параметры в блоке подбора. А дальше уйдет покупать или изучать подробнее выбранное.

5. Сценарий взаимодействия контентщика с бэк-ендом. Изначально контещник идет в листинг спарсенных результатов с Маркета и меняет каждому товару урл на нужный. В дальнейшем контентщик будет заполнять и менять 5-10 полей каждому товару, включая параметры подбора.  
Далее, в последующих версиях сервиса, оптимизатор + контентщик будут совестно создавать и заполнять страницы подбора (те самые, с собственным урлом).

**Вариант I – парсинг с** [**http://market.yandex.ru/gifts.xml**](http://market.yandex.ru/gifts.xml)

*Описан в большей степени не вариант реаилизации, а возможность спарсить данные и реализовать их в нужном фронте. По архитектуре, реализации и коду разработчик делает на свое усмотрение, но с теми же возможностями что описаны ниже:*

**Описание: парсинг всех данных и логики работы с Яндекс.Подарков. Должна быть админка с возможностью редактирования данных подарков.**

**Реализация: парсиг товаров и логики работы**[**http://market.yandex.ru/gifts.xml**](http://market.yandex.ru/gifts.xml)**в БД:**

1. <http://market.yandex.ru/gifts-result.xml>?  
Парсим в БД весь ассортимент товаров (цена, фото, название). Всего 731 шт.

\*отдельно обратить внимание на фото (оно должно парсится в определенный каталог в будущем используемый на сайте) и + в БД должен записываться адрес фото, по которому его можно найти в нашем каталоге.

2. Выясняем - get запросы каждого из параметров чекбоксов, радио кнопок, всплывающего списка.  
Найти их просто, переменные логически названы + отображаются в форме, при ручном форматировании get запроса. Парсим названия товаров по get запросам - по одному (например <http://market.yandex.ru/gifts-result.xml?gender=ForMan>)

Названия товаров связываем с нашей БД (п.1), добавляя аналогичные запросу поля. Например если товар показывается по радиокнопке любимому значит нужно сделать поле - *Любимому* со значением 1, если не показывается то значение 0.

\*Один и тот же товар может иметь несколько значений единицы в полях, что логично. Прим. - *Любимому-1, Босу-1, Папе-0… и т.д.*

3. Аналогично с ползунками \*(кроме цены она уже есть в п.1). У них будет следующее отличие. Товар будет показываться при значении 1 - когда у него романтичность будет стоять 5.  
Поэтому делаем следующее. Парсим по романичность 1 - всем выбранным ставим 1ку, парсим роамантичность 2 - всем вышедшим ставим двойку, те кто не вышел, значит у них и есть 1ка.

4. Далее необходимо создать числовые поля со значением от 0 до 600 (хотя возможно и меньше). 16 шт. под каждый из вариантов радио кнопки Кому.  
Далее вытаскиваем все товары по радиокнопке. В зависимости от позиции ставим цифру в подходящее поле. Если по радиокнопке Дедушке - товар .... распологается на 10 позиции, ставим в поле цифру - 10 , если этот же товар в поле Босс распологается на 120 позиции ставим в соответствующее поле цифру 120. По умолчанию товары имеют в поле цифру 0.  
По логике у всех товаров должно быть заполнено хотя бы одно поле.

5. Текстовое поле – которое в будущем будут заполнять контенщики (подставляя наши url).

**Реализация: админки:**

В админке должно быть 2 страницы.

1. Список всех спарсенных товаров по названию (берется с БД).
2. Возможность редактирования всех полей в ручную \*(или проще только 1го поля по умолчанию пустого – п. 5. Поле под ссылку товара).

**Реализация: страниц:**

**Состоит из 2х страниц.**

1. **Главной**: аналог <http://market.yandex.ru/gifts.xml> (только форма подбора, рецепты и прочее не нужно.)
2. **Результаты поиска**: аналог <http://market.yandex.ru/gifts-result.xml?day=BirthdayPresent&gender=ForWoman&for=ForLover&age=NoImportance&Price_min=0&Price_max=400000&Game=on&Romantic=1&Original=1&Useful=1&ForSoul=4&Funny=1&Chic=1&Smart=1&Miracle=1> (форма, блок товаров, текстовый блок количество товаров, текстовые блоки meta, пагинация.)

**1. Главная.**

Форма реализуется по аналогии с <http://market.yandex.ru/gifts.xml> активность каждого параметра подставляется под значения, которые мы вытащили при парсинге в БД.

Выбранные параметры должны добавлять get параметры урлу. И после нажатия кнопки Подобрать, переходить по сформированному адресу.

Например в форме мы выбрали День рождения, Женщине, Любимой и Сувениры

Url получается

<http://market.yandex.ru/gifts-result.xml?day=BirthdayPresent&gender=ForWoman&for=ForLover&age=NoImportance&Price_min=0&Price_max=400000&Souvenir=on&Original=1&Romantic=1>

*\*&Price\_min=0&Price\_max=400000&Souvenir=on&Original=1&Romantic=1* – эти параметры мы не указывали, но маркет их подставляет. Можно делать также можно просто оставить пустыми, и так понятно что по умолчанию будет 0 или 1(на ползунках). На усмотрение разработчика.

\*как обязательное условие (называние переменных нужно другое уникальное, ни как у маркета, для отличия)

\*как обязательное условие – урлы должны формироваться динамично и работать даже без заполнения формы (только по урлу выдавать нужный товар).

**2. Результаты поиска.**

Результаты поиска должны формировать запрос взяв get переменные из url и сделав выборку к БД. Вытащив товары.

Сортировка осуществляется по 2м ключевым факторам. Сначала:

**А** – по соответствию выбранных параметров (отсекая которые не были выбраны (радиокнопки) в форме или не подпадают под ее значение. По умолчанию, некоторые поля (чекбоксы) например рубрики Категории, если не были выбраны то считаются активными).

**Б** – по значениям БД (описанным в п. 4). Т.к. поле кому можно выбрать только одно (радиокнопкой) то и сортировка не будет пересекаться.

**Блоки на странице:**

Блок вывода товаров – по аналогии с Яндекс.Подарками – только картинки выводим из нашего каталога (обсуждалось в парсинге в БД. П.1). А урл товара вставляем из БД п.5.

Текстовый блок количество товаров – показывает сколько товаров выбрано по count.

Текстовые блоки meta – Нужно генерировать своими «средствами» без парсинга (title и h1). Данные брать из фильтра, для первой версии пойдет вид (Подарки на день рождение: для любимой: 40-50 лет). То есть подставлять активированные пользователем (или урлом) фильтры. Каждый фильтр будет иметь свой человеческий вид, немного отличающийся от названия в фильтре. Полученный результат выдается в h1 и title.

*Название фильтров:* текст подстановки для генерации

* *Повод:* Подарок на День Рождение\на юбилей\на день свадьбы\на новоселье\в благодарность\-
* *Кому:* для любимой\коллеге\босса\подруги\девочки\мамы\бабушки\другу\мальчику\папе\дедушке\себе
* *Возраст:* (градация возраста)

В последующей версии нужен более гибкий вариант.

Пагинация – По аналогии с Маркетом. Если сложно реализовать, то можно без нее \*(максимум по фильтру выходит 335 товаров, но это совсем редкость. Обычно меньше 60).

Если пагинация реализуется, то нужно в конце title добавлять «Страница - %n».

Форма подбора – просто вставить ту, которая была реализована с главной стр. Только изменяется дизайн.

\*Верстку нужно на 1м этапе взять с я.подарков. Конкретно css стили, картинки, html верстку. А уже далее при смене дизайна изменить вид блоков – отдельно!

**Вариант II – MVP версия сервиса**

Нужна уже готовая CMS в которой будет реализовано 2 ключевые возможности (по сути нужно тоже самое что и описывалось в варианте №1, только со своим ручным заполнением, картинки, текстовые поля, параметры значений под форму). На первом этапе нужно будет спарсить данные «товаров» из Яндекс.Подарков и реализовать аналогичный подбор.

В CMS должны быть:

1. Ключевое! Возможность создать через админку товарам любые поля (текстовые, числовые, булевые). И возможность далее удобно работать по заполнению этих товаров.
2. Ключевое! Возможность создание формы подбора товаров по заданным параметрам формы (которые мы берем заполненные с п.1). Любо легкая реализация этой формы разработчиком через средства предоставляемые CMS.
3. Возможность каждому варианту фильтрации задавать отдельную страницу для индексации (ЧПУ урл вместо набора get-параметров) и задавать SEO настройки (meta, тексты, работа со страницами).
4. Известная CMS или фреймворк (первое предпочтительнее). По которой уже есть стартовый набор базовых дополнений, а также разбирающиеся в движке разработчики.

\*По работе CMS нужно будет 3 типа страниц (в первом варианте было 2). 3й тип это карточка товара. Со всеми страницами должно быть гибкое управление под нужды описанные в первом варианте.