

# Практическое задание

## «Разработка сайтов электронной торговли»

[Структура проекта и исходные данные](#)

[Структура директорий](#)

[Исходные данные](#)

[Предлагаемые к использованию инструменты](#)

[Функциональные требования](#)

[Общие](#)

[Навигация](#)

[Футер](#)

[Формы](#)

[Список товаров \(list view\)](#)

[Блок с описанием товара](#)

[Фильтр](#)

[Постраничная навигация](#)

[Единичный товар \(item view\)](#)

[Галерея изображений](#)

[Техническое описание](#)

[Пользовательские отзывы](#)

[Корзина и процесс заказа \(shopping cart, checkout process\)](#)

[Виджет корзины](#)

[Корзина](#)

[Страница заказа](#)

[Routing](#)

[Persistence](#)

[Описание основных сценариев с сохраненными данными](#)

[Корзина](#)

[Страница оформления заказа](#)

## Структура проекта и исходные данные

Макет разрабатываемого сайта скачать можно [тут](#).

## Структура директорий

Рекомендованная структура директорий для практического задания выглядит так:

```
index.html
images
|— layout/
```

```
└─ content/
css
└─ market.css
js/
└─ market.js
```

## Исходные данные

Для выполнения практического задания были подготовлены исходные данные, описывающие различные группы товаров (моб. телефоны, планшеты, фотоаппараты, плееры, телевизоры). Данные представлены в формате JSON, для каждой группы есть описания 150 единиц товара.

Каждая сущность имеет название, краткое описание, диапазон цен, пользовательский рейтинг, ссылки на изображение продукта, список свойств составляющих описание технических характеристик.

## Пример:

```
{
  "itemTitle": "Apple iPhone 5s (16GB)",
  "itemThumbImgUrl": "http://goo.gl/Savc8d",
  "shortDescr": "Apple iOS, экран 4\" IPS, процессор 1300 МГц (2 ядр.), аккумулятор 1560 мАч",
  "rating": 4.5,
  "price": {
    "max": "3200",
    "min": "790"
  },
  "techInfo": {
    "thumbs": [
      {
        "thumbURL": "http://goo.gl/0HfwQG",
        "fullSizeURL": "http://goo.gl/wzOUwp"
      },
      ...
    ],
    "birthday": {
      "title": "Дата выхода на рынок",
      "value": "2013 г."
    },
    "band": {
      "title": "Стандарт связи",
      "value": "GSM (2G), EDGE (2.9G), HSPA+ (3.75G), LTE (4G)"
    },
    "mobile_type": {
      "title": "Тип",
      "value": "смартфон"
    },
    ...
  }
}
```

Скачать исходные данные можно по след ссылкам:

- [телефоны](#)
- [планшеты](#)
- [фотоаппараты](#)
- [плееры](#)
- [телевизоры](#)

## Предлагаемые к использованию инструменты

В силу того, что задание довольно объемное мы решили разрешить к использованию следующие библиотеки (их использование *необязательно*, но позволит сэкономить время)

- [underscore.js](#) для темплейтинга и работы с массивами
- роутинг модуль [Backbone.Router](#) (?)
- css препроцессор [Sass/Less](#) (?)

## Функциональные требования

### Общие

Все элементы страницы должны структурно соответствовать [макету](#), стилизация элементов — на усмотрение разработчика.

Все действия на на странице не должны приводить к ее перезагрузке, кроме случаев, где переход на отдельную страницу указан как требование.

### Навигация

Основная навигация должна содержать ссылки, при переходе на которые будет осуществлено отображение товаров различных [групп](#) а так же [виджет корзины](#) покупок.

Панель навигации при выходе из зоны видимости страницы должна закрепляться в верхней части страницы, оставаясь, таким образом, постоянно на виду у пользователя.

### Футер

Навигационные ссылки должны быть продублированы в средней части футера, кроме того в нем должна содержаться подписная форма новостей. При клике по кнопке «подписаться» на сервер должен быть отправлен асинхронный запрос, результат его должен быть выведен в диалоговом окне (можно просто алерт, бонусом будет сделать красивый лайтбокс).

## Формы

Все поля форм на страницах проекта должны иметь валидации и маски ввода, визуально сообщать пользователю о том, что он ошибся при заполнении, если это имело место быть, и делать невозможной отправку формы в этом случае.

## Список товаров (list view)

Контентная часть страницы представляет из себя 2х колоночный макет, в левой части которого расположены [блоки с кратким описанием продуктов](#), в правой — сайдбар с [фильтром](#). Под контейнером продуктовых блоков должна располагаться [постраничная навигация](#).

### Блок с описанием товара

Блоки должны быть расположены по 3-4 штуки в строку, несколько строк друг под другом. Каждый блок должен содержать изображение товара, его название, цену, кнопки «сравнить» и «в корзину» (опционально можно добавить пользовательский рейтинг).

Клик по изображению продукта, его цене или названию должен привести к переходу на [страницу детального просмотра](#) этого товара. Клик по рейтингу товара, если он есть в блоке — к [фильтрации](#) товаров, отображены должны быть только товары с таким же рейтингом.

Клик по кнопке «в корзину», как ни странно, должен привести к добавлению товара в корзину, что должно визуально отразиться на блоке с этим товаром и счетчике товаров в [виджете корзины](#), а также добавлению данного товара в виджет и пересчету итоговой суммы.

### Фильтр

Фильтр представляет из себя форму, обязательными полями которой являются диапазон цен и пользовательский рейтинг. Остальные поля формируются по усмотрению разработчика (таким образом, чтобы их было не менее 10) из технического описания конкретной модели продукта.

Булевы поля должны отображаться как чекбоксы, пользовательский рейтинг как выпадающий список, цена, объемы памяти и другие свойства, которые можно описать интервалами должны представлять из себя пары текстовых полей («от» и «до») (бонусом будет сделать ползунок).

Заданные значения фильтров должны [сохраняться при перезагрузке страницы](#).

### Постраничная навигация

Для каждой группы товаров есть возможность реализовать постраничную навигацию, на одной странице не стоит размещать более 15-20 товаров. Активная страница должна быть визуально выделена, кроме того значение должно [сохраняться в хэше страницы](#).

## Единичный товар (item view)

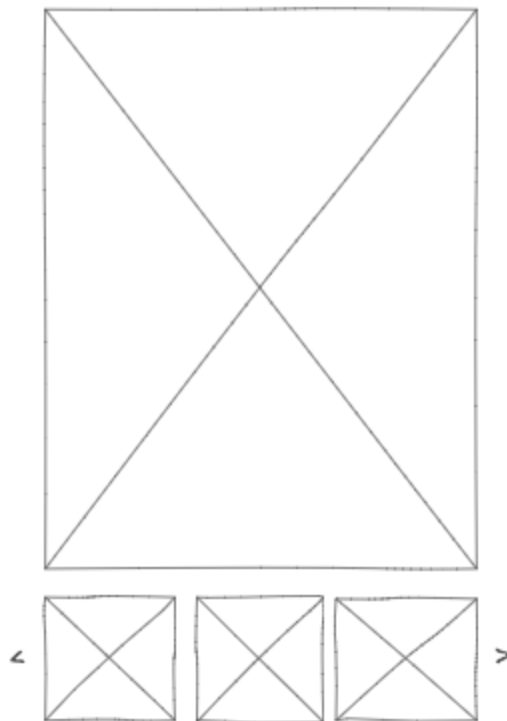
Контентная часть страницы представляет из себя 2х колоночный макет, в левой части которой расположен сайдбар с [галереей изображений](#) продукта; в правой — детальная информация о продукте (название, цена, пользовательский рейтинг, подробное описание, техническое описание, пользовательские отзывы, кнопка «в корзину»).

Под кнопкой «В корзину» будет находится блок с двумя вкладками: [Техническое описание](#) и [пользовательские отзывы](#). По-умолчанию активна вкладка «тех. описание».

Действия кнопок «сравнить» и «в корзину» соответствуют описанию из [секции list view](#).

### Галерея изображений

Галерея состоит из двух частей — большое изображение и лента из трех уменьшенных картинок под ним. По умолчанию должна быть выбрана первая картинка из ленты. Выделить ее стилистически. Она же будет отображена и в качестве большой картинки. Если для данного продукта существует более трех изображений, то необходимо наличие навигационных стрелок вправо и влево. Когда пользователь долистывает вправо (влево) до конца, правая (левая) стрелка становится неактивной.



*галерея изображений продукта*

## Техническое описание

Техническое описание должно из себя представлять таблицу из 2х столбцов: в левом перечислены технические характеристики, в правом — значения этих характеристик.

## Пользовательские отзывы

Панель пользовательских отзывов состоит из самих отзывов и формы добавления отзыва под ними.

Каждый отзыв должен содержать имя автора, рейтинг и комментарий.

Форма добавления отзыва состоит из двух текстовых полей (оба они обязательны для заполнения, отметить это визуально) и виджета выставления рейтинга. Виджет работает следующим образом: пользователь кликает на звезду, все стоящие перед ней и она включительно становятся закрашенными, а ее порядковый номер будет означать цифру рейтинга от 1 до 5. При нажатии на кнопку «Опубликовать отзыв», отправляется AJAX запрос на сервер, и под уже имеющимися отзывами снизу добавляется ваш.

Техническое описание	Отзывы
<div><div>Саня Корейко</div><div>☆☆☆☆☆</div><div>&gt;Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</div></div> <div><div>Шура Балаганов</div><div>☆☆☆☆☆</div><div>&gt;Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</div></div> <div><div>Зоя Синицкая</div><div>☆☆☆☆☆</div><div>&gt;Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</div></div> <div><div>Оставьте свой отзыв</div><div><div>Имя</div><div>Текст отзыва</div><div>☆☆☆☆☆</div><div>Опубликовать отзыв</div></div></div>	

## Корзина и процесс заказа (shopping cart, checkout process)

### Виджет корзины

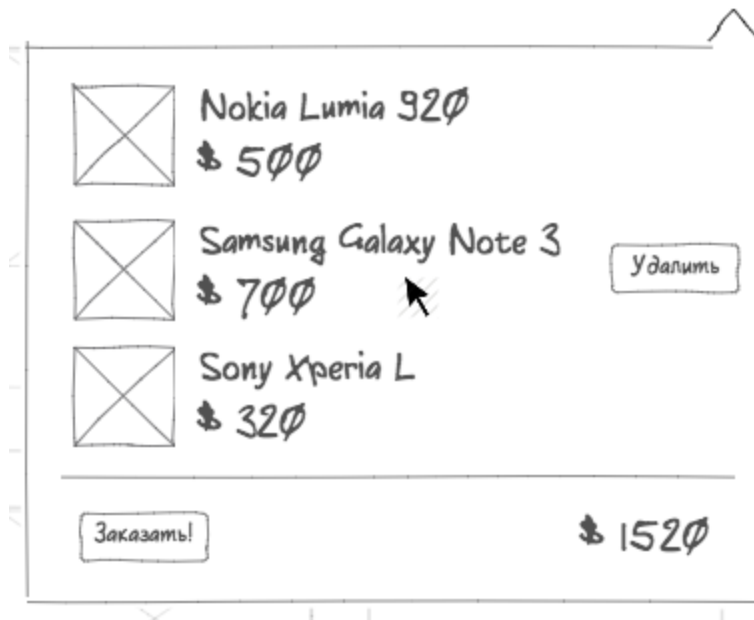
У виджета корзины есть два состояния: свернутое (ссылка «корзина» и счетчик товаров в панели навигации) и активное (выпадающий список с выбранными товарами).

По умолчанию находится в свернутом положении. Счетчик товаров должен быть скрыт, если в корзине на данный момент не находится ни одного товара.



свернутое состояние виджета корзины

При наведении курсором мыши на счетчик товаров или ссылку «корзина», появляется блок со списком выбранных товаров. Под списком отображена суммарная стоимость всех товаров и кнопка «Заказать», при нажатии на которую, переходим к странице «Корзина».



активное состояние виджета корзины

Каждый из элементов списка товаров в корзине должен состоять из миниатюрного изображения, наименования и цены продукта. При наведении на область элемента в списке товаров, появляется кнопка «удалить», нажатие на которую приведет к удалению данного товара из корзины, пересчету итоговой суммы и обновлению счетчика товаров в верхней части виджета.

Чтобы вернуть виджет в состояние по умолчанию (свернутый вид), нужно кликнуть в любое место на странице.

## Корзина

В корзине, аналогично виджету, размещен список товаров с названием и ценой. Кроме того, для каждого товара имеется поле, отражающее количество единиц этого товара в корзине, сумма для каждого товара, с учетом его кол-ва, итоговая сумма для всех товаров в корзине. Под списком должны быть кнопки «купить что-нить еще» и «я хочу заплатить». Первая кнопка вернет пользователя в каталог, вторая — осуществит переход на страницу оформления заказа.

По наведению курсора мыши на элемент с информацией о продукте появится кнопка «удалить» действие которой описано в секции [виджет корзины](#).

## Страница заказа

Страница оформления заказа состоит из трех разделов. Первый раздел: «Способ доставки». Все поля обязательны для заполнения в случае, если пользователь выберет выбор доставки товаров курьером. Эти поля становятся необязательными, если выбирается самовывоз. Второй раздел: «Способ оплаты» — аналогично первому разделу.

Три раздела реализованы по принципу аккордеона. Пользователь должен иметь возможность возвращаться на предыдущие этапы. Но при этом, переход к заполнению следующего раздела невозможен, если в предыдущих разделах упала ошибка валидации.

Переход от одного этапа оформления заказа к другому осуществляется в строгом порядке. Введенные данные должны сохраняться и быть доступны при перезагрузке страницы. Также, если пользователь ушел с нашего ресурса, не закончив оформление заказа, а потом вернулся, — он должен быть сразу направлен на страницу оформления заказа и текущим этап оформления заказа должен быть тот, на котором данный процесс был прерван. (реализована данная функциональность должна быть с учетом требований и правил, описанных в разделе [persistence](#))

В разделе «Обзор заказа» должен быть приведен список купленных пользователем товаров. В зависимости от того, какой способ доставки он выбрал, ему нужно показывать либо просто поздравление, либо карту Яндекса | Гугла, с проставленным на ней маркером, а также контактные телефоны.

## Routing

Следующие пути должны быть реализованы:

`#/phones/x`



```
#/tablets/x
#/laptops/x
#/accessories/x
```

где x – номер страницы, если выбран вид «List View»

либо

```
#/phones/lumia920
#/tablets/...
#/laptops/...
#/accessories/...
```

если выбран вид «Item View»

как вариант: #!/

## Persistence

Все данные должны быть сохранены в localStorage в виде JSON объекта, преобразованного в строку с ключом `bsu-market`. «Sensitive» пользовательские данные сохранять нельзя (в данном случае платежная информация).

Общая структура сохраняемого объекта:

```
{
  cartItems : [
    {
      id : "apple_iphone5s16",
      qnt : 1
    },
    ...
  ],
  orderInfo : {
    takeAway : false,
    userData : {
      city      : "Minsk",
      street    : "Timiryazeva",
      building  : 68,
      flat      : 901,
      phone     : "+375-29-388-00-44"
    }
  }
}
```

## Описание основных сценариев с сохраненными данными

### Корзина

В массиве `cartItems` должны содержаться данные обо всех товарах, добавленных пользователем в корзину. Если пользователь покидает сайт (перезагружает страницу), а затем возвращается обратно, у него в корзине по-прежнему должны отображаться товары, которые он добавил туда до того, как покинул сайт.

Если пользователь удаляет товар из корзины или успешно завершает оформление заказа, то соответствующие товары должны быть удалены из хранилища.

### Страница оформления заказа

В хранилище необходимо сохранить такие пользовательские как: адрес и телефон, значение `takeAway` изначально должно быть установлено в `null` и пока оно таким остается при перезагрузке страницы пользователь будет оставаться на первом этапе оформления заказа, а данные, которые он успел ввести, должны быть автоматически заполнены. В момент, когда пользователь нажал на одну из кнопок «Везите, уже везите!» или «Хочу быстрее, сам заберу!» то `takeAway` принимает булево значение, а при перезагрузке страницы пользователь теперь будет находиться на втором этапе оформления заказа.

Пользовательские данные пользователя должны сохраняться и после оформления заказа, но значение `takeAway` должно сбрасываться в `null`.