

# Техническое задание на разработку интернет-магазина

04/05/2016

## 1 Описание проекта

Рабочее название проекта - lingershop. Проект представляет собой интернет-магазин, в котором желающие клиенты могут покупать нижнее белье.

## 2 Технические уточнения

### 2.1 Термины

- Блок - некий визуальный элемент, выделяющийся либо графически (в виде рамок, очертаний), либо по смыслу (совокупность похожих элементов)
- Компонент - часть содержания, имеющего закрытое визуальное представление. Одна страница сайта может состоять из нескольких компонентов.
- Фронтэнд - для пользователя видимая оболочка веб-страницы
- Бэкэнд - невидимые для пользователя математические алгоритмы
- CMS - все работы происходят на основе системы управления содержанием - CMS 1C Bitrix (1С Битрикс)
- Модуль - является описанием общего функционала, который не может быть классифицирован как привязанный к определенной странице. Он может встречаться на любой странице в любом месте. Модуль может состоять из нескольких компонентов. Также модуль может содержать в себе части логики фронтэнда и бэкэнда.
- Хэдер - верхняя часть сайта, обладающая определенной структурой, которая видна сквозняком на всех или почти всех страницах сайта. Также используется обозначение "шапка".
- Футер - нижняя часть сайта. Функционал аналогичен хэдеру. Также используется обозначение "подвал".
- Администраторская панель ("админка") - часть сайта, скрытая для обычных пользователей, позволяющая редактировать содержимое сайта
- Публичная часть ("паблик") - часть сайта, доступная для просмотра обычным пользователям.

## 2.2 Технические требования к сайту

Сайт располагается на хостинге или выделенном сервере (virtual dedicated server), в зависимости от предпочтений заказчика.

## 3 Архитектура проекта

### 3.1 Общее

Проект изначально должен соблюдать правила объектно ориентированного программирования. Т.е. любой пользователь должен быть объектом класса "пользователь". Это распространяется на все сущности. Все действия системы должны быть методами своих классов. Псевдокод:

```
#Class description

class User < ActiveRecord::Base

  def initialize(name, phone, mail)
    @name = name
    @phone = phone
    @mail = mail
  end

  # Register method
  def register(user_params)
    User.create(user_params)
  end

  # Allow only whitelisted params
  private
    def user_params
      params.require(:user).permit(:name, :phone, :mail)
    end
end
```

Программирование должно осуществляться по подходу TDD - test driven development. Это означает, что перед написанием функционала необходимо написать тест (который заведомо провалится, так как функционала нет). Далее пишется минимальный функционал, который позволяет пройти данный тест.

Псевдокод теста:

```
require 'user'

RSpec.describe User, "#register" do
  context "when user is completely new" do
    it "creates a new entry with valid data" do
      user = User.new
```

```

    user.name = "Vasya_Pupkin"
    user.phone = "8_903_227_88_74"
    user.mail = "vasya@pupkin.ru"
    user.register

    expect(User.count).to change{User.count}.by(1)
  end
it "does_not_create_a_new_entry_with_invalid_data" do
  user = User.new
  user.name = "Vasya_Pupkin"
  user.phone = "j1lkh3i1u2y98"
  user.mail = "someinvalidtext"
  user.register

  expect(User.count).not_to change{User.count}.by(1)
end
end
end

```

Т.е. в данном случае мы тестируем создание нового пользователя с валидными данными (первый случай) и с невалидными данными. В первом случае мы ожидаем увеличение общего кол-ва пользователей на 1, во втором случае мы не ожидаем увеличений.

## 3.2 Модели

Общее описание - есть каталог товаров, товары принадлежат разделам и категориям.

- Раздел
- Категория
- Товар
- Товарное предложение

### 3.2.1 Структура моделей

Товар может принадлежать одному или нескольким разделам и категориям. У товара может быть несколько товарных предложений.

#### Пример

Есть товар "ночнушка прозрачная". Она может относиться как к категории "нижнее белье" так и к категории "спальные вещи". У данной ночнушки есть несколько товарных предложений - конкретный размер и цвет (XXXL может стоить дороже, чем XS)

## 4 Модули

### 4.1 Оплата

Необходимо встроить прием онлайн-оплаты через агрегатор "Робокасса".

### 4.2 Скидки

Необходимо иметь возможность управлять скидками для товаров.

Нужна ли программа лояльности? Например накопительные скидки?

## 5 Страницы

### 5.1 Главная

### 5.2 Список товаров

### 5.3 Карточка товара

Вывод информации в соответствии с шаблоном. На карточке товара перечислены следующие свойства товара:

- Название товара
- Фотографии товара - обязательно с функцией zoom
- Размер одежды
- Цвет одежды
- Цена товара
- Описание товара
- Сопутствующие товары

Также в карточке товара необходимо вывести элементы социального sharing'a.

### 5.4 Корзина (оформление заказа)

В корзине пользователь может просмотреть все свои товары, которые он положил в корзину, отредактировать количество и удалить позиции. При вводе контактных данных для доставки пользователь может оформить заказ выбрав способ оплаты и доставки.

Для оформления заказа достаточно следующих полей от пользователя:

- телефон (обязательно)
- электронная почта (обязательно)

- адрес (страна, город, индекс, улица, дом, квартира) - опционально, может уточнить и менеджер при повторном звонке.

Также при оформлении заказа у клиента должна быть возможность ввести промокод. ИМ должен поддерживать систему промокодов (они должны управляться в админке).

## 5.5 Личный кабинет пользователя

Дефолтно пользователь попадает на страницу с персональными данными. Если у него есть актуальный заказ, то статус данного заказа выводим сверху по центру.

### 5.5.1 Персональные данные

Перечень полей, которые пользователь может изменить. При первичном оформлении заказа (если клиента нет в базе) на почту клиенту приходит пароль, с которым он сможет авторизоваться и в будущем на сайте ИМ. Поля следующие:

- ФИО (отдельные поля)
- Адрес (отдельные поля)
- Электронная почта
- Телефон
- Пароль

### 5.5.2 История покупок

Подстраница с перечнем всех покупок. Табличный вид заказов, состоящий из следующих полей:

- Дата заказа
- Сумма заказа
- Статус доставки
- Статус оплаты
- Перечень товаров (если слишком много, то делаем truncate и показываем полный перечень при наведении мышкой)

### 5.5.3 Вишлист

Перечень товаров, которые нравятся пользователю. Необходимо давать доступ к вишлисту также незарегистрированным пользователям. Суть - что бы клиенты могли делиться своими вишлистами.

Нужен ли действительно функционал?

## 5.6 Типовые страницы

- Контакты - необходимо подгружать из настроек магазина
- Доставка - необходимо подгружать из настроек магазина
- О магазине - текстовая страница
- Соглашение пользователя - текстовая страница
- Конфиденциальность - текстовая страница
- Новости - управляется соответствующим блоком в администраторской панели
- Часто задаваемые вопросы - управляется блоком в админке
- Обратная связь - контактная форма, присылает сообщение на адрес из настроек магазина
-