**Задание на УП ПМ 08. Часть 1.**

Для своей предметной области написать приложение на Django. Выбрать 1 из 3х вариантов реализации:

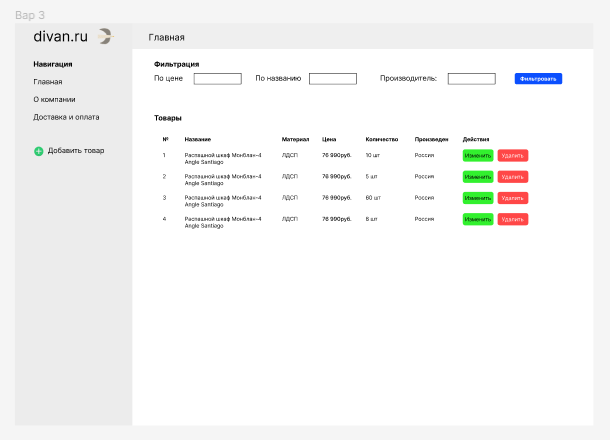
<https://www.figma.com/file/BPl2pdxhG4OUsHMLX3eE7Z/Untitled?type=design&node-id=0-1&mode=design&t=Rj1grEd7z4RHubfA-0>

**Вариант 3.**

Список товаров на главной странице.

+ форма добавления

+ 2 статические страницы



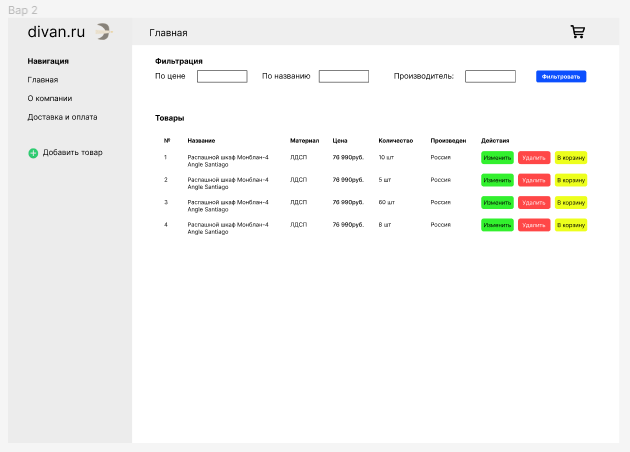
**Вариант 2.**

Список товаров на главной странице.

+ форма добавления

+ 2 статические страницы

+ кнопка добавить в корзину – при добавлении в корзину нужно уменьшать количество товара в 1 единицу. При нажатии на иконку корзины, открывается список набранных товаров. Их можно так же убрать из корзины.



**Вариант 1.**

Список товаров на главной странице. + каждый товар содержит фото.

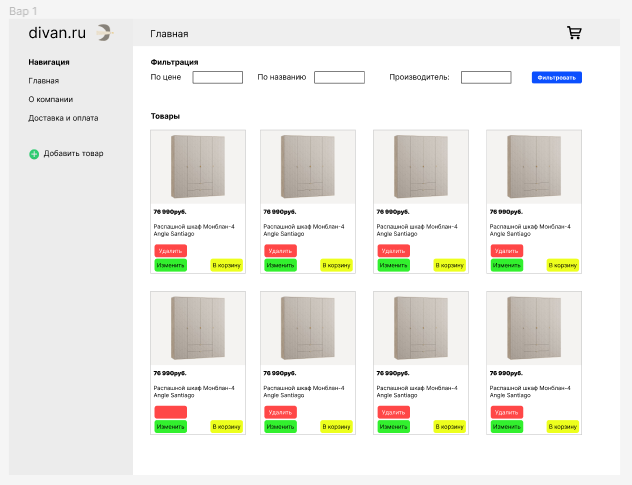
+ форма добавления

+ 2 статические страницы

+ кнопка добавить в корзину – при добавлении в корзину нужно уменьшать количество товара в 1 единицу. При нажатии на иконку корзины, открывается список набранных товаров. Их можно так же убрать из корзины.

**Дополнительно. (название супер-отличника :)**

**Реализовать регистрацию и авторизацию в приложении. Вести учет что купил каждый пользователь у каждого из пользователей в ЛК должны отражаться его покупки.**



Пример файла с моделями:

from django.db import models  
  
  
class Category(models.Model):  
 title = models.CharField('Название', max\_length=50)  
 object = models.Manager()  
  
 def \_\_str\_\_(self):  
 return self.title  
  
 class Meta:  
 verbose\_name = 'Категория'  
 verbose\_name\_plural = 'Категории'  
  
  
class Product(models.Model):  
 title = models.CharField('Название', max\_length=50)  
 descr = models.TextField('Описание')  
 material = models.CharField('Материал', max\_length=50)  
 color = models.CharField('Цвет', max\_length=50)  
 country = models.CharField('Страна', max\_length=50)  
 price = models.IntegerField('Цена', default=0)  
 quantity = models.IntegerField('Количество', default=0)  
 category = models.ForeignKey(Category, on\_delete=models.SET\_NULL, null=True)  
 object = models.Manager()  
  
 def \_\_str\_\_(self):  
 return self.title  
  
 class Meta:  
 verbose\_name = 'Единица продукции'  
 verbose\_name\_plural = 'Продукция'  
  
  
class Basket(models.Model):  
 quantity = models.IntegerField('Количество', default=0)  
 product = models.ForeignKey(Product, on\_delete=models.SET\_DEFAULT, default='Null')  
 object = models.Manager()  
  
 def \_\_str\_\_(self):  
 return str(self.product)  
  
 class Meta:  
 verbose\_name = 'Заказ'  
 verbose\_name\_plural = 'Заказы'