Детали реализации

# Инструменты

### Дизайн

1. Adobe Photoshop
2. InVision

### Frontend

1. HTML + Django templates
2. Sass -> CSS
3. JS + jQuery
4. Gulp

### Backend

1. Python + Django
2. sqlLite

### Взаимодействие внутри команды

1. Slack
2. Github

### Визуальный стиль

Metro

Шрифты:

1. Segoe UI (Windows)
2. Open Sans( Остальные платформы)

## Codestyle

#### Общие

Если переменные не упакованы в модули сначала называется сфера или объект применения, затем модификаторы.

color-accent, font-color-main (в данном случае шрифт – сфера применения, цвет-объект, основной-его модификатор)

main-color, main-color-font (основной – модификатор цвета, он не имеет значения основная палитра цветов или цвета модуля основной, второй пример абсолютно нелогичен иерархически)

Для разделения слов в имении переменной использовать нижнее подчеркивание либо тире, в зависимости от возможностей языка

number\_big, number-big

Названия конструкторов и классов пишутся с большой буквы, всего остального – с маленькой

Class(), Constructor(), variable, array, instance, function( arg1, arg2 )

В функциях при наличии 1 аргумента он не отбивается от скобок пробелом, при большем количестве аргументов или выражений переменные от логических знаков отбиваются пробелом с 2 сторон, от запятых справа, от скобок с внутренней стороны

function(arg), function( arg1, arg2, arg3 ), if(true), if( a == b ), while( x+y > 0 )

Название функции и массива не отбивается от скобок пробелом

function(arg), array[1][2]

Все переменные объявляются в начале области видимости, до кода

global\_var1

global\_var2

If( true ):

local\_var 1

local\_var2

//code

Для отступов использовать Tab

#### JS

Фигурные скобки отбиваются пробелом от других слов, открывающая фигурная скобка пишется на 1 строке с выражением \ объявлением функции, закрывающая пишется на новой строке

while(true) {

//code

}

#### Python

Двоеточие не отбивается пробелами

#### SASS

БЭМ для классов etc.

## Бренд

В самом приложении чувствуется научность, он выражается слегка витиевато, сложными предложениями, но не используя чрезмерно научной лексики.

У приложения нет талисмана.

Соблюдается иерархия, приложения не пытается указывать юзеру и не прерывает его работу, хотя всегда неподалеку готовое помочь и/объяснить что-то новое.