

# Bootstrap Grid / Проектирование

---

**WEB  
COURSE  
ORT DNIPRO**

---

**ORT**[DNIPRO.ORG/WEB](https://ORTDNIPRO.ORG/WEB)

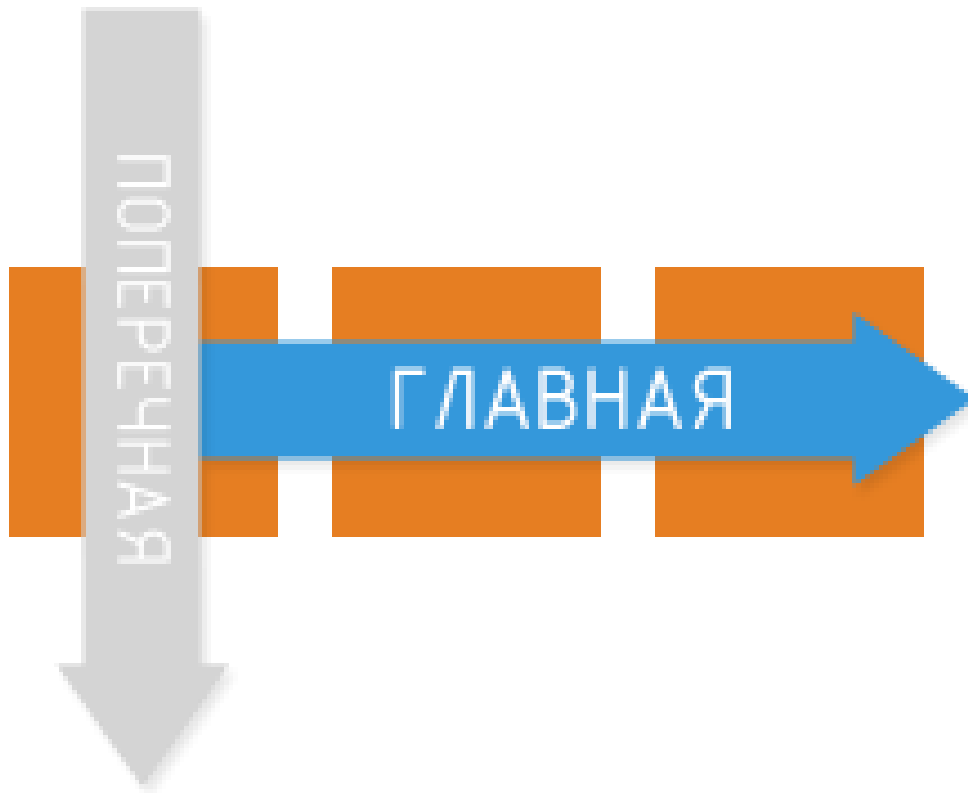
**Важно!**

**Bootstrap Grid  $\neq$  CSS Grid**

**\* это никак не связанные технологии**

# 1. Вспомним о Flexbox

# Flexbox – управление размещением элементов




Идея **flexbox**'а проста: все элементы располагаются в ряд, для этого ряда есть главная и поперечная оси. Как будут элементы выравниваться по осям задаётся при помощи свойств **justify-content** и **align-items** (для главной и поперечной осей соответственно).

Включается режим **flexbox** при помощи свойства **display: flex**, а перенос элементов разрешается при помощи **flex-wrap: wrap**.

При помощи свойства **flex-direction** возможно указать направление размещения строка или колонка (**row**, **row-reverse**, **column**, **column-reverse**).

## 2. Вспомним проблемы **Flexbox**

# Как добиться подобного расположения компонентов?



**Sebastien Saunier**  
CTO at Le Wagon

Get in touch

3,536  
contributions

758  
followers

119  
repositories

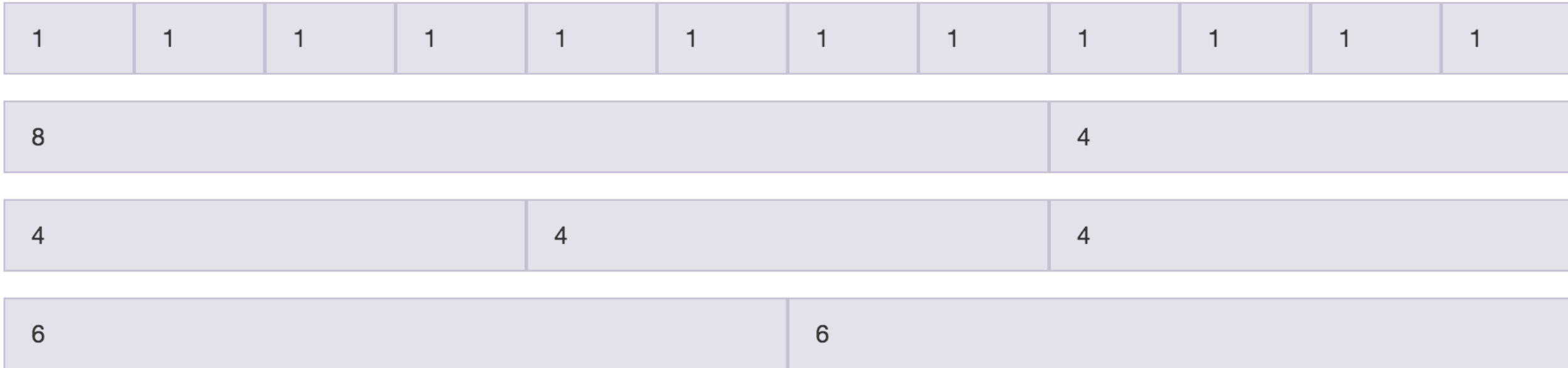
**More about Sebastien**

I am a full-stack developer with 9 years of experience working in startups. Scratching your own itch? Working on a startup project? Lots of ideas but frustrated that you can't code them? Confused with agencies' or freelances' quotes? No clue when a developer is talking? Time to learn to code with Le Wagon with a 9-week intensive and immersive bootcamp!

# 3. Bootstrap Grid\*

\* в Bootstrap 5 построен на базе Flexbox

# Bootstrap Grid 5 – многоколоночное размещение элементов



<https://getbootstrap.com/docs/5.0/layout/grid/>

Одна из опор **Bootstrap – Grid** («сетка») размещения элементов, позволяющая размещать элементы в несколько столбцов, размер (и количество) которых будет адаптироваться под размеры экрана.



# Немного практики с Bootstrap Grid



Шаблон с разметкой в репозитории занятия:

[./src/demo-template](#)

## 4. Классы `.row.row-cols-N`

# Классы `.row.row-cols-N`

```
19 <main class='p-3 row row-cols-2 row-cols-lg-4'>
20   <div>A</div>
21   <div>B</div>
22   <div>C</div>
23   <div>D</div>
24   <div>E</div>
25   <div>F</div>
26   <div>G</div>
27   <div>H</div>
28   <div>I</div>
29   <div>J</div>
30 </main>
```

Реализуют схему: «*Все по N-штук в ряд*». Класс `.row` задаётся для **родительского элемента** (который станет **flex-контейнером**), и для него же задаётся класс задающий количество колонок в нём (**`row-cols-2`, `row-cols-5`** и т.п. **до 6 максимум**). Возможно применение адаптивных классов **`row-cols-md-*`** (например: **`row-cols-md-3`, `row-cols-lg-4`** и т.п.).

<https://getbootstrap.com/docs/5.0/layout/grid/#row-columns>

5. Классы **.row** > **.col-N**

# Классы `.row` > `.col-N`

```
19 <main class='p-3 row'>
20   <div class='col-3'>A</div>
21   <div class='col-6'>B</div>
22   <div class='col-4'>C</div>
23   <div class='col-7'>D</div>
24   <div class='col-2'>E</div>
25   <div class='col-1'>F</div>
26   <div class='col-9'>G</div>
27   <div class='col-10'>H</div>
28   <div class='col-1'>I</div>
29   <div class='col-1'>J</div>
30 </main>
```

Реализуют схему: «Пространство делится на 12 равных колонок, каждый блок занимает часть этих колонок». Класс `.row` задаётся для **родительского элемента** (который станет **flex-контейнером**). Классы **col-N** (например `col-4`, `col-7`, `col-12` и т.п.) задаются для **дочерних элементов** (по сути **для ячеек** в этой «сетке»). Возможно применение адаптивных классов `col-*` (например: `col-md-3`, `col-lg-7` и т.п.).

# 6. Адаптивные классы

## Bootstrap Grid

# Адаптивность по версии Bootstrap Grid 5

1		X-Small	<i>None</i>	0–576px
2		Small	<i>sm</i>	≥576px
3		Medium	<i>md</i>	≥768px
4		Large	<i>lg</i>	≥992px
5		Extra large	<i>xl</i>	≥1200px
6		Extra extra large	<i>xxl</i>	≥1400px

**Bootstrap Grid** – адаптивный инструмент. В зависимости от категории устройства размеры и количество элементов в ряду может меняться.

<https://getbootstrap.com/docs/5.2/layout/grid/>

# 7. Проектирование



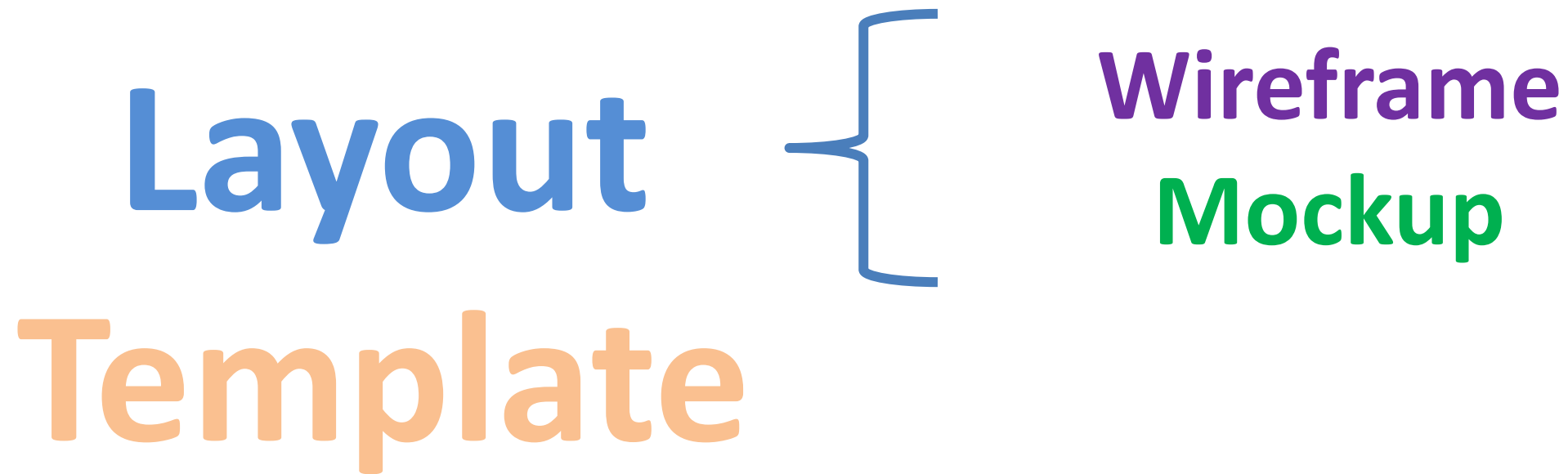
**Раньше терминология была простая**

# Layout Template

**Layout** – макет страницы («картинка»)

**Template** – сверстанные страницы

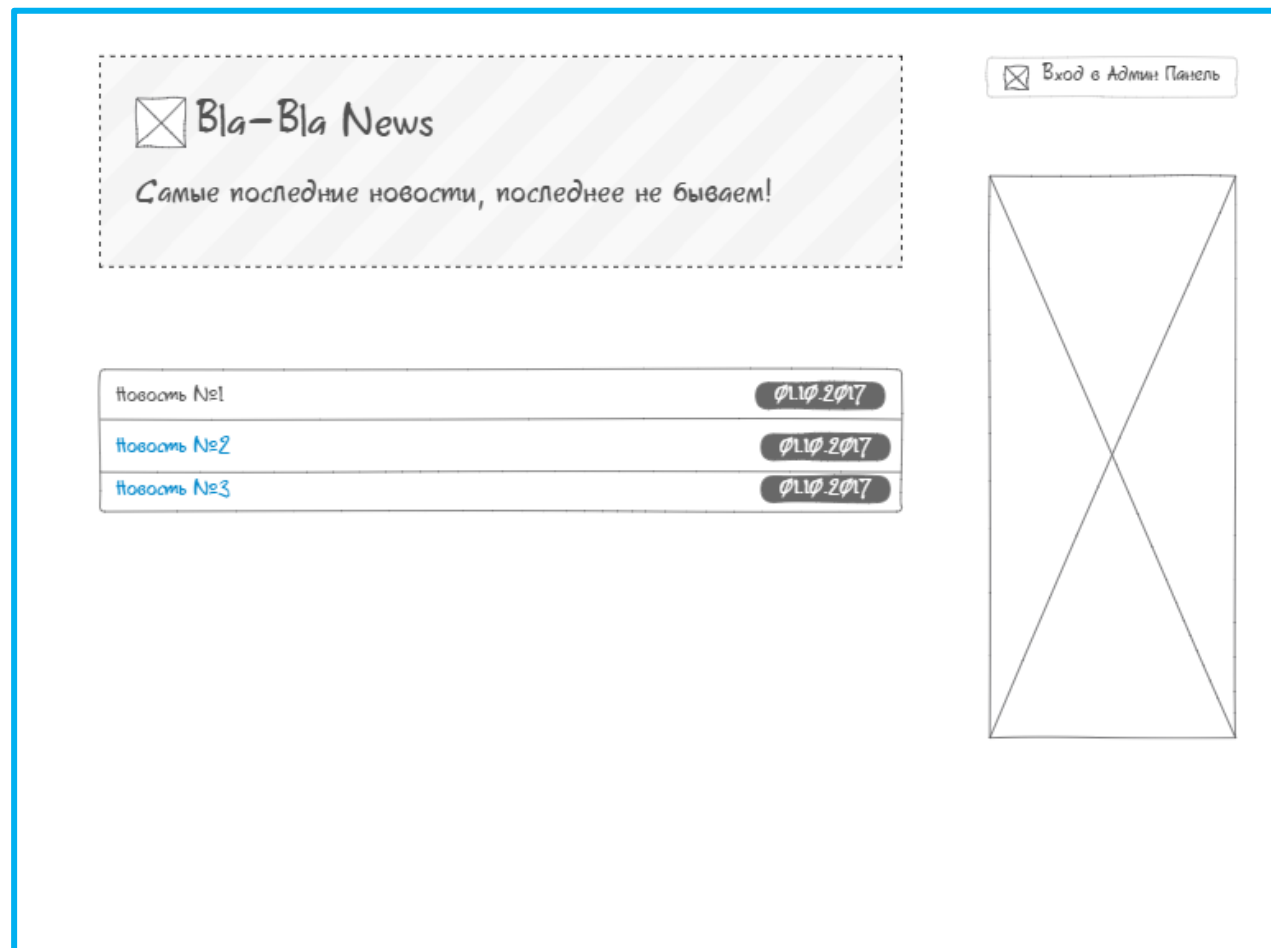
Потом терминология усложнилась



**Wireframe** – низкодетализированный макет страницы.

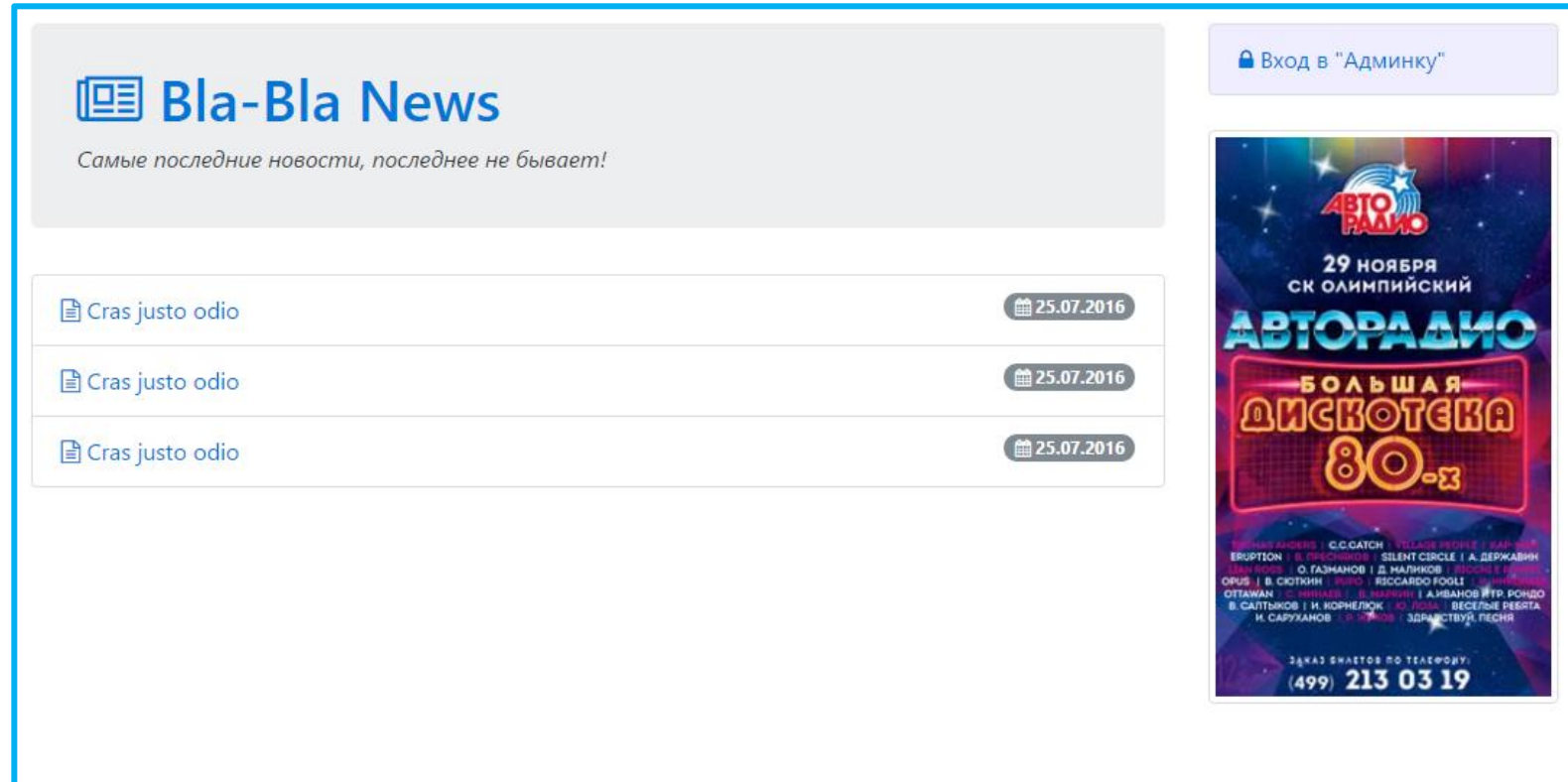
**Mockup** – высокодетализированный макет страницы

# Wireframe – «проволочный каркас»



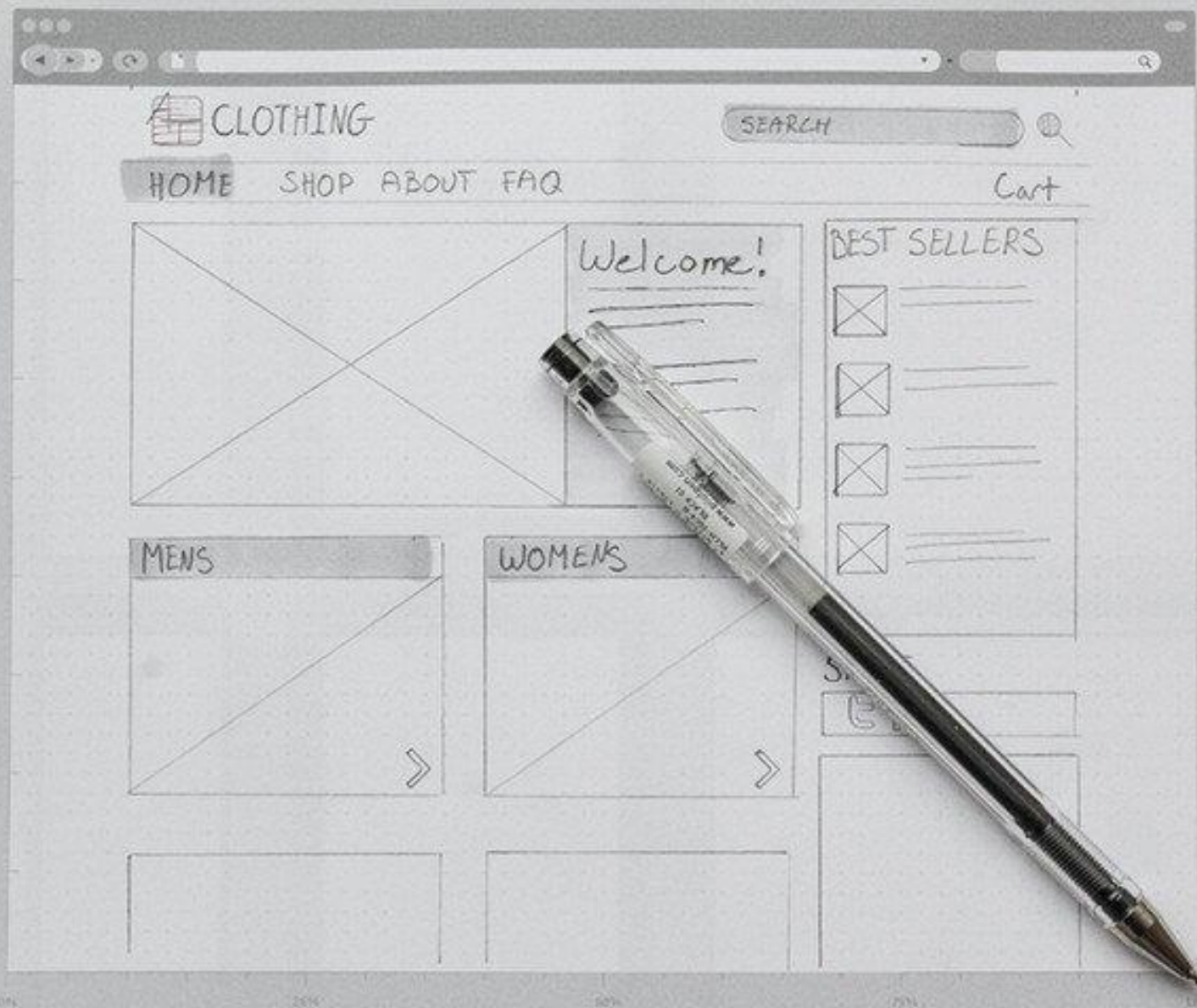
***Wireframe*** – упрощенный макет страницы, схематическое изображение страницы, как правило низкой детализации.

# Москир – «макет страницы»



***Москир** – макет страницы, изображение (картинка) того как страница будет выглядеть в реальности.*

# 8. Инструменты проектирования и прототипирования



# Инструменты проектирования

<https://wireframe.cc/>

wireframe|cc

<https://ninjamock.com/>



<https://moqups.com/>

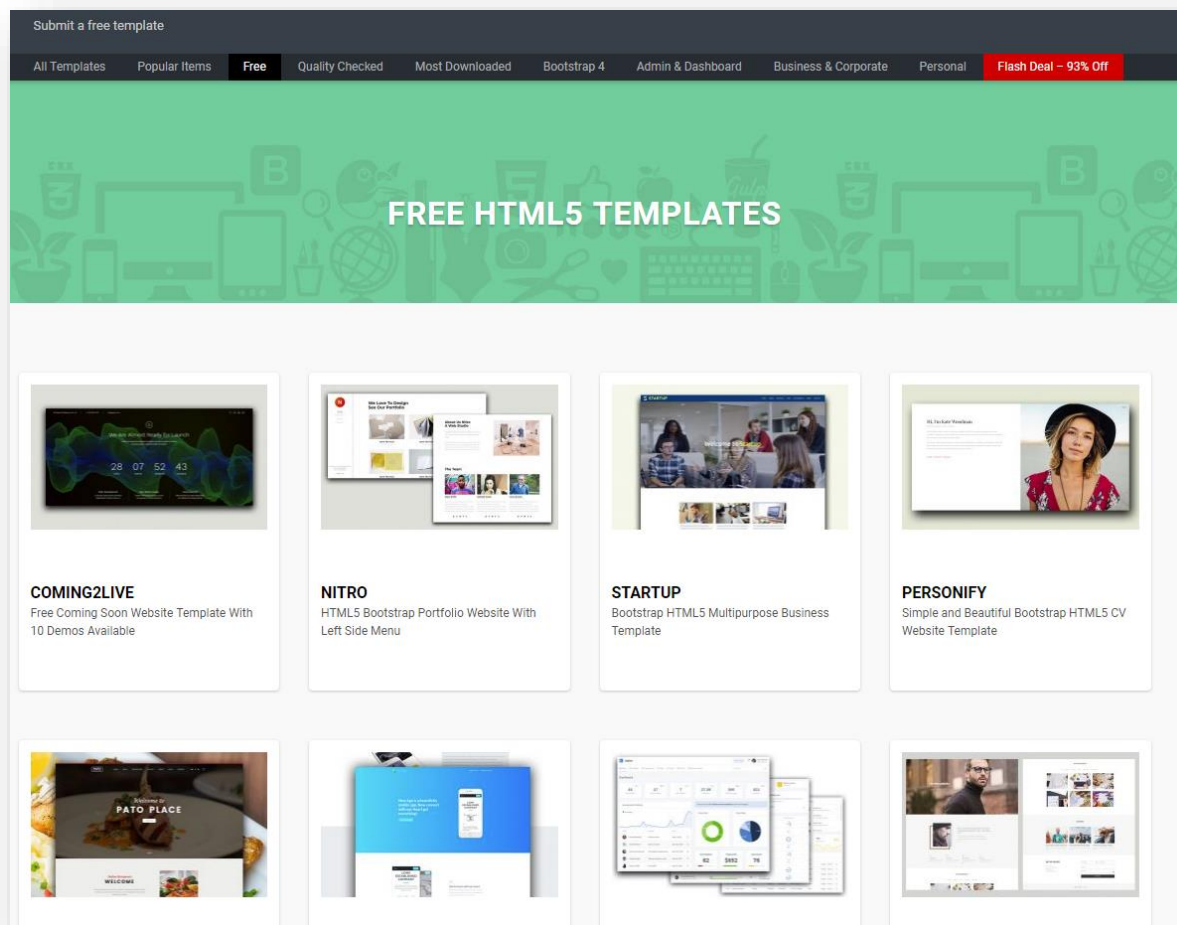
moqups



# 9. Готовые шаблоны



# Theme Wagon – сервис готовых шаблонов

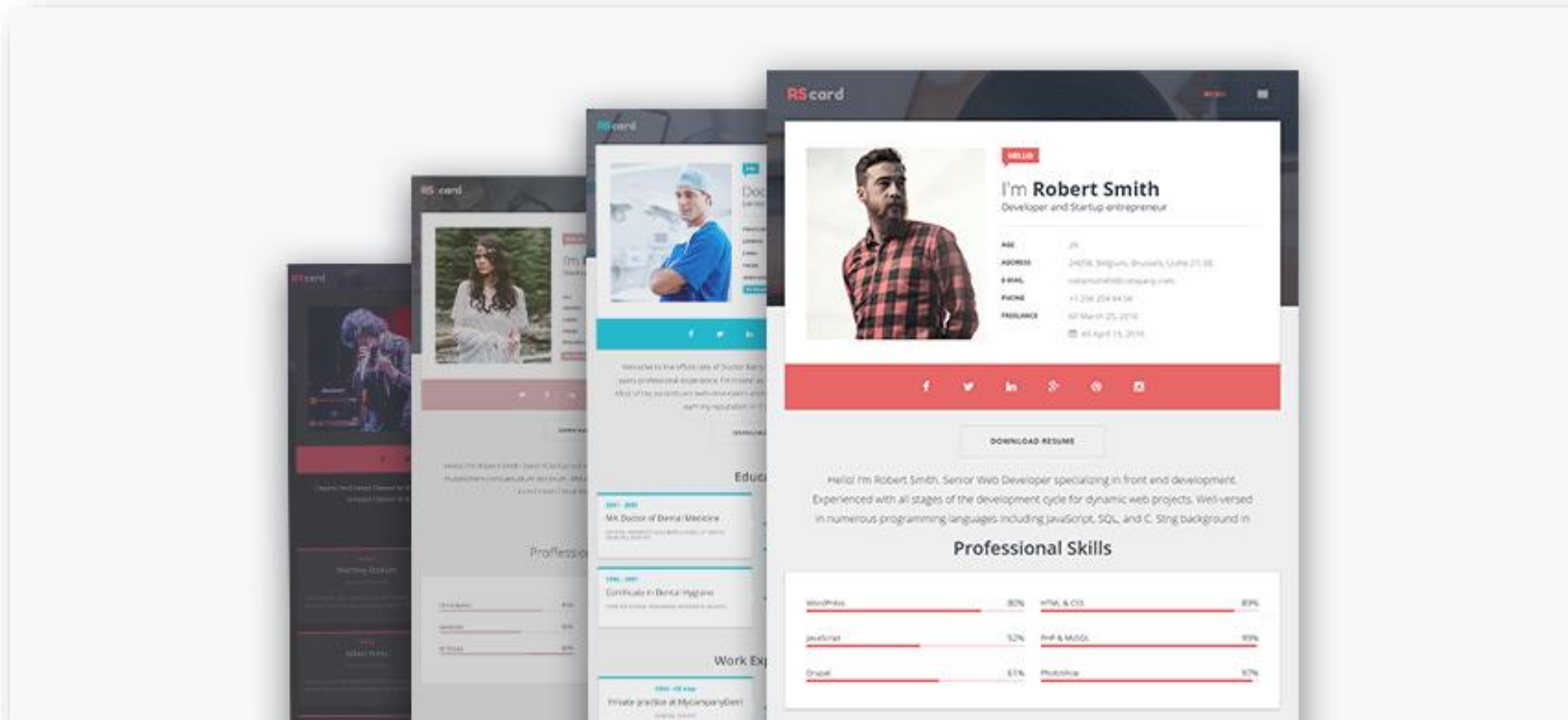


«Theme Wagon» - один из ресурсов которые предлагают верстальщикам **готовые шаблоны** (темы) которые могут быть использованы в качестве основы для разработки своих проектов. По большей части шаблоны бесплатны для использования.

<https://themewagon.com/>

# Домашнее задание

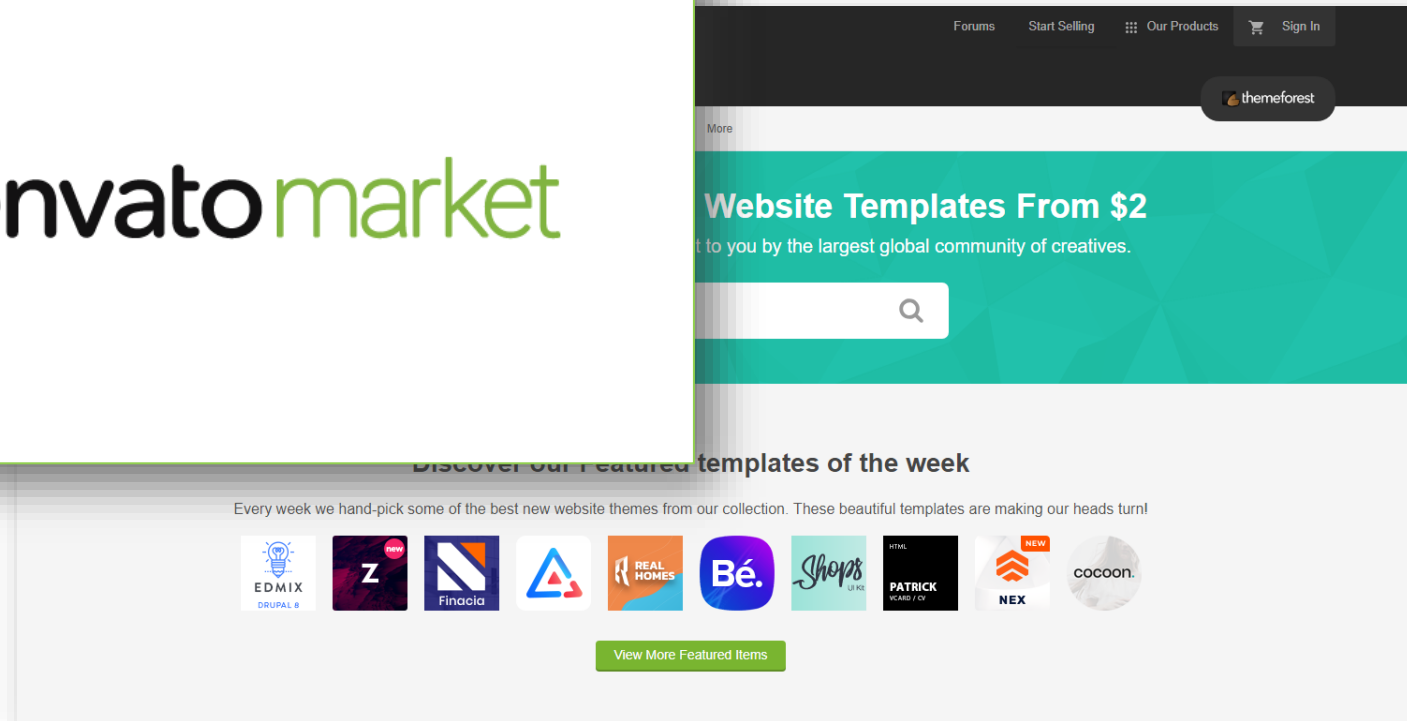
# Обновляем резюме



Пора освежить резюме...!

**Будет полезным**

# Envato Market (ex. ThemeForest)



«Envato Market» - крупнейший магазин **платных** готовых шаблонов.

<https://themeforest.net/>

**К следующему  
занятию...**

# Пора двигаться дальше...



Предварительные знания – лучший помощник в обучении, поэтому к следующему занятию жду, что **Вы обязательно посмотрите ролик...**

[https://youtu.be/SW\\_UCzFO7X0?t=167](https://youtu.be/SW_UCzFO7X0?t=167)