

Вёрстка многослойных элементов интерфейса

Выполнен на 51%

Введение

Углублённая теория

Методика вёрстки многослойных элементов

Вёрстка кнопки «Наверх»

Вёрстка «липкой» горизонтальной навигации

Вёрстка «липкого» sidebar

Вёрстка окна с соглашением об использовании Cookie

Вёрстка пользовательской подсказки

Вёрстка компонента Карусель/Слайдер

Вёрстка компонента Выпадающий список

Вёрстка компонента Чат-бот

Вёрстка сложного меню

Вёрстка компонента для расширенного поиска

Вёрстка компонента Tinder card

Вёрстка модального окна

Кейс 1, лёгкий уровень

Кейс 2, лёгкий уровень

Кейс 3, лёгкий уровень

Кейс 4, лёгкий уровень

Кейс 5, средний уровень

Кейс 6, средний уровень

Кейс 7, средний уровень

Кейс 8, сложный уровень

Кейс 9, сложный уровень

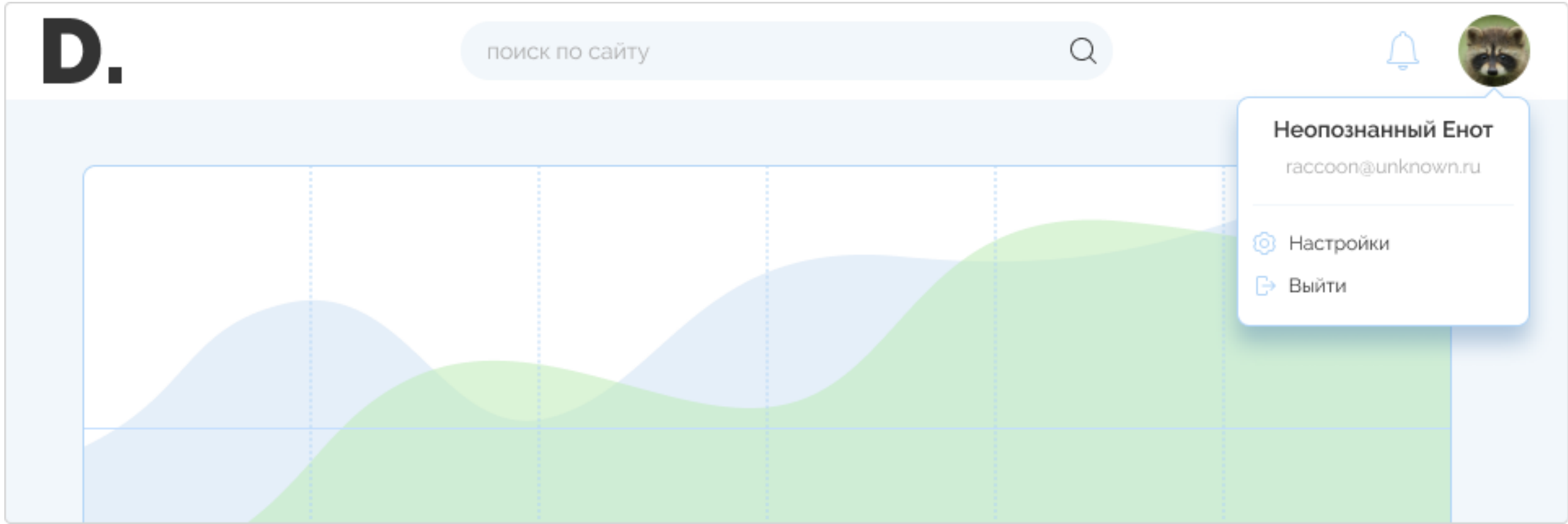
Кейс 10, сложный уровень

Главная / Моё обучение / Вёрстка многослойных элементов интерфейса

Методика вёрстки многослойных элементов

Вёрстка компонента Выпадающий список

Выпадающий список — это ещё один компонент из категории многослойных. Его часто используют в шапке сайта. Это может быть список с выбором языка сайта, пользовательское меню, кнопка с вариантами действий (*Copy -> Copy element, Copy outerHTML, Copy selector, ...*) и другие.



Пользовательское меню в выпадающем списке

Выпадающий список открывается и закрывается при нажатии на кнопку—ссылку или стрелку рядом (этот элемент ещё называют *триггерным*). Реже список открывается при наведении на элемент управления. В библиотеках компонентов этот элемент обычно называют *Dropdown* или *Dropdowns*.

Для реализации компонента обычно используют тег `<div>`, внутри которого лежит *триггерный элемент* и сам *выпадающий список*. Бывает, что поведение компонента определяется добавленными классами. Например, в библиотеке *daisyUI* для выпадающего списка, который открывается при наведении, добавлен класс `dropdown-hover`.

```
<div class="dropdown dropdown-hover">
  <label tabindex="0" class="button">Наведи на меня! И увидишь мой список желаний.</label>
  <ul tabindex="0" class="menu dropdown-content">
    <li>
      <a href="#">Велосипед</a>
    </li>
    <li>
      <a href="#">Обложка для паспорта</a>
    </li>
    <li>
      <a href="#">Туфли</a>
    </li>
  </ul>
</div>
```

Разберёмся со стилями компонента. Во-первых, выпадающий список выводится в отдельном слое, во-вторых, он скрыт по умолчанию. Чтобы разместить список в отдельном слое, используется `position: absolute;`. Скрыть список можно за счёт `opacity: 0;` и `visibility: hidden;`, как это происходит в вышеупомянутой библиотеке. Анимация во время появления списка и его скрытия добавляется с помощью CSS-свойства `transition`.

```
.dropdown .dropdown-content {
  visibility: hidden;
  position: absolute;
  z-index: 50;
  opacity: 0;
  transform-origin: top;
  transition-property: opacity;
  transition-duration: 0.2s;
  transition-timing-function: cubic-bezier(0.4,0,0.2,1);
}

.dropdown.dropdown-hover:~.dropdown-content {
  visibility: visible;
  opacity: 1;
}
```

Для варианта без специального класса `dropdown-hover`, нужно задействовать *JavaScript*-код, в котором меняются значения CSS-свойств `visibility` и `opacity` на противоположные.

Для сравнения, в другой популярной библиотеке компонентов — *bootstrap* реализация следующая. Также есть триггерный элемент и список с выпадающим меню.

```
<div class="dropdown">
  <button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton1" data-bs-toggle="dropdown" aria-expanded="false">Запустить Windows 10</button>
  <ul class="dropdown-menu" aria-labelledby="dropdownMenuButton1">
    <li>
      <a class="dropdown-item" href="#">Обычный запуск</a>
    </li>
    <li>
      <a class="dropdown-item" href="#">Диагностический запуск</a>
    </li>
    <li>
      <a class="dropdown-item" href="#">Выборочный запуск</a>
    </li>
  </ul>
</div>
```

```
.dropdown {
  position: relative;
}

.dropdown-menu {
  position: absolute;
  z-index: 1000;
  display: none;
  min-width: 10rem;
  padding: 0.5rem 0;
  margin: 0;
}
```

При клике на триггерный элемент `dropdown-toggle` добавляется или удаляется класс `show` для элемента с классом `dropdown-menu` и для кнопки-переключателя меню.

```
<div class="dropdown">
  <button class="btn btn-secondary dropdown-toggle show" type="button" id="dropdownMenuButton1" data-bs-toggle="dropdown" aria-expanded="true">Запустить Windows 10</button>
  <ul class="dropdown-menu show" aria-labelledby="dropdownMenuButton1">
    <li>
      <a class="dropdown-item" href="#">Обычный запуск</a>
    </li>
    <li>
      <a class="dropdown-item" href="#">Диагностический запуск</a>
    </li>
    <li>
      <a class="dropdown-item" href="#">Выборочный запуск</a>
    </li>
  </ul>
</div>
```

В стилях для выпадающего списка с классом `show` устанавливается `display: block;`.

```
.dropdown-menu.show { /* Добавляем класс и меню показывается */
  display: block;
}
```

Для читалок в *bootstrap* добавлены атрибуты `aria-labelledby` и `aria-expanded`.

Замечание

Атрибут `aria-expanded` устанавливается для элемента, чтобы указать, развёрнут элемент управления или свернут, а также отображаются или скрыты его дочерние элементы. Подробнее читайте в спецификации *WAI-ARIA*.

Вёрстка компонента Карусель/Слайдер

Вёрстка компонента Чат-бот