

Уральский государственный экономический университет

ОСНОВЫ WEB ПРОГРАММИРОВАНИЯ

02.03.03 -Математическое обеспечение и администрирование информационных систем,
направленность (профиль) -разработка и администрирование информационных систем

<http://vikchas.ru>

Лекция 5. «Bootstrap и создание адаптивных web - сайтов и web-приложений»

Часовских Виктор Петрович
доктор технических наук, профессор кафедры
ШИиКМ, ФГБОУ ВО «Уральский государственный
экономический университет

Екатеринбург 2026

Адаптивная веб-разработка

Адаптивная веб-разработка — это подход к созданию сайтов и веб-приложений, при котором интерфейс корректно отображается на разных устройствах:

- компьютерах;
- ноутбуках;
- планшетах;
- смартфонах;
- устройствах с разным разрешением экрана.

Главная цель адаптивного дизайна — сделать сайт удобным для пользователя независимо от размера экрана.

Проблема разных устройств

Один и тот же сайт может открываться на экране шириной:

- 320 px — смартфон;
- 768 px — планшет;
- 1366 px — ноутбук;
- 1920 px и выше — монитор.

Если сайт не адаптирован, пользователь может столкнуться с проблемами:

- текст слишком мелкий;
- элементы выходят за границы экрана;
- появляются горизонтальные полосы прокрутки;
- кнопки сложно нажимать;
- меню неудобно использовать.

Основные принципы адаптивного дизайна

Адаптивная верстка строится на нескольких принципах.

1. Гибкая сетка

Элементы страницы задаются не только в фиксированных пикселях, но и в относительных единицах:

CSS

```
width: 50%;
```

```
max-width: 1200px;
```

Это позволяет блокам изменять размер в зависимости от ширины экрана.

2. Гибкие изображения

Изображения должны масштабироваться внутри родительского блока:

CSS

```
img {  
    max-width: 100%;  
    height: auto;  
}
```

Так картинка не выйдет за пределы экрана.

3. Media Queries

Media queries — это CSS-правила, которые применяются только при определённых условиях, например при заданной ширине экрана.

Пример:

CSS

```
@media (max-width: 768px) {  
    .menu {  
        display: none;  
    }  
}
```

Это правило скрывает меню на экранах шириной меньше 768 px.

4. Mobile First

Mobile First — подход, при котором сначала создаётся версия сайта для мобильных устройств, а затем добавляются стили для более крупных экранов.

Пример:

CSS

```
.card {  
  width: 100%;  
}
```

```
@media (min-width: 768px) {  
  .card {  
    width: 50%;  
  }  
}
```

```
@media (min-width: 1200px) {  
  .card {  
    width: 25%;  
  }  
}
```

Bootstrap

Bootstrap — это популярный HTML, CSS и JavaScript-фреймворк для быстрой разработки адаптивных сайтов и веб-приложений.

Он содержит готовые:

- CSS-классы;
- сеточную систему;
- компоненты интерфейса;
- JavaScript-компоненты;
- утилиты для отступов, цветов, размеров и позиционирования.

Bootstrap позволяет быстрее создавать современные интерфейсы без написания большого количества CSS вручную.

Преимущества Bootstrap

1. Быстрая разработка

Можно использовать готовые компоненты:

- кнопки;
- формы;
- карточки;
- навигационные панели;
- модальные окна;
- выпадающие списки.

2. Адаптивность по умолчанию

Bootstrap изначально ориентирован на адаптивную верстку и подход Mobile First.

3. Единый стиль интерфейса

Компоненты Bootstrap имеют согласованный внешний вид.

4. Кроссбраузерность

Bootstrap учитывает особенности популярных браузеров.

5. Большое сообщество и документация

Bootstrap имеет подробную документацию и множество примеров.

Подключение Bootstrap

Bootstrap можно подключить несколькими способами.

1. Подключение через CDN (Content Delivery Network, «сеть доставки контента») — это географически распределённая сеть серверов, которая ускоряет отдачу статического контента пользователям за счёт того, что данные приходят не с одного центрального сервера, а с ближайшего к пользователю узла.

Самый простой способ — подключить Bootstrap через CDN.

Пример базового HTML-документа:

HTML

```
<!DOCTYPE html>
```

```
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Bootstrap пример</title>

  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>

  <h1 class="text-center mt-5">Привет, Bootstrap!</h1>

  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js">
</script>
</body>
</html>
```

Важный тег viewport

HTML

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Этот тег необходим для корректной адаптивности на мобильных устройствах.

Контейнеры Bootstrap

Контейнеры используются для ограничения ширины содержимого и выравнивания его по центру.

1. .container

Класс .container создаёт адаптивный контейнер фиксированной максимальной ширины.

HTML

```
<div class="container">  
  <h1>Контент страницы</h1>  
</div>
```

2. .container-fluid

Класс .container-fluid занимает всю ширину экрана.

HTML

```
<div class="container-fluid">  
  <h1>Контент на всю ширину</h1>  
</div>
```

3. .container-{breakpoint}

Контейнер может быть резиновым до определённой ширины экрана.

Пример:

HTML

```
<div class="container-md">  
  <h1>Контейнер container-md</h1>  
</div>
```

Сеточная система Bootstrap

Одна из главных возможностей Bootstrap — **grid system**, или сеточная система.

Сетка Bootstrap состоит из:

- контейнеров;
- строк;
- колонок.

Основная структура:

HTML

```
<div class="container">  
  <div class="row">
```

```
<div class="col">Колонка 1</div>
<div class="col">Колонка 2</div>
<div class="col">Колонка 3</div>
</div>
</div>
```

1. 12-колоночная сетка

Bootstrap использует сетку из 12 условных колонок.

Например:

HTML

```
<div class="row">
  <div class="col-4">4 колонки</div>
  <div class="col-8">8 колонок</div>
</div>
```

Сумма колонок в строке обычно равна 12.

Другой пример:

HTML

```
<div class="row">
  <div class="col-6">Половина ширины</div>
  <div class="col-6">Половина ширины</div>
</div>
```

Breakpoints в Bootstrap

Breakpoint — контрольная ширина экрана, при которой меняется поведение макета.

В Bootstrap 5 используются следующие breakpoints:

Префикс	Размер	Минимальная ширина
col-	xs	< 576 px

Префикс	Размер	Минимальная ширина
col-sm-	sm	≥ 576 px
col-md-	md	≥ 768 px
col-lg-	lg	≥ 992 px
col-xl-	xl	≥ 1200 px
col-xxl-	xxl	≥ 1400 px

Пример адаптивных колонок:

HTML

```
<div class="row">
  <div class="col-12 col-md-6 col-lg-4">Блок 1</div>
  <div class="col-12 col-md-6 col-lg-4">Блок 2</div>
  <div class="col-12 col-md-6 col-lg-4">Блок 3</div>
</div>
```

Что произойдёт:

- на смартфоне каждый блок занимает 12 колонок;
- на планшете — 6 колонок;
- на большом экране — 4 колонки.

Отступы и spacing utilities

Bootstrap предоставляет классы для управления внешними и внутренними отступами.

1. Формат классов

{property}{sides}-{size}

Где property:

- m — margin;
- p — padding.

sides:

- t — top;
- b — bottom;
- s — start;
- e — end;
- x — left/right;
- y — top/bottom;
- без указания стороны — все стороны.

size:

- 0 — 0;
- 1 — маленький;
- 2;
- 3;
- 4;
- 5 — большой;
- auto.

Примеры:

HTML

```
<div class="mt-3">Отступ сверху</div>
```

```
<div class="mb-4">Отступ снизу</div>
```

```
<div class="p-3">Внутренний отступ</div>
```

```
<div class="mx-auto">Центрирование по горизонтали</div>
```

Цвета и текст

Bootstrap содержит готовые классы для цветов.

1. Цвет текста

HTML

```
<p class="text-primary">Основной цвет</p>
```

<p class=" -success">Успех</p>

<p class=" -danger">Ошибка</p>

<p class=" -muted">Приглушённый текст</p>

2. Цвет фона

HTML

<div class="bg-primary -white">Синий фон</div>

<div class="bg-success -white">Зелёный фон</div>

<div class="bg-warning">Жёлтый фон</div>

3. Выравнивание текста

HTML

<p class=" -start">Слева</p>

<p class=" -center">По центру</p>

<p class=" -end">Справа</p>

Кнопки Bootstrap

Кнопки создаются с помощью класса .btn и модификаторов.

HTML

<button class="btn btn-primary">Основная кнопка</button>

<button class="btn btn-secondary">Вторичная кнопка</button>

<button class="btn btn-success">Сохранить</button>

<button class="btn btn-danger">Удалить</button>

Размеры кнопок:

HTML

<button class="btn btn-primary btn-lg">Большая</button>

<button class="btn btn-primary btn-sm">Маленькая</button>

Контурные кнопки:

<button class="btn btn-outline-primary">Подробнее</button>

```
<button class="btn btn-outline-danger">Удалить</button>
```

Формы Bootstrap

Bootstrap значительно упрощает оформление форм.

Пример формы авторизации:

```
<form>
```

```
<div class="mb-3">
```

```
<label for="email" class="form-label">Email</label>
```

```
<input type="email" class="form-control" id="email" placeholder="Введите email">
```

```
</div>
```

```
<div class="mb-3">
```

```
<label for="password" class="form-label">Пароль</label>
```

```
<input type="password" class="form-control" id="password" placeholder="Введите пароль">
```

```
</div>
```

```
<button type="submit" class="btn btn-primary">Войти</button>
```

```
</form>
```

Основные классы:

- .form-control — для текстовых полей;
- .form-label — для подписей;
- .form-select — для выпадающих списков;
- .form-check — для checkbox и radio.

Карточки Bootstrap

Карточка — универсальный компонент для отображения информации.

```

<div class="card" style="width: 18rem;">
  
  <div class="card-body">
    <h5 class="card-title">Заголовок карточки</h5>
    <p class="card- ">Описание карточки.</p>
    <a href="#" class="btn btn-primary">Подробнее</a>
  </div>
</div>

```

Карточки часто используются для:

- товаров;
- новостей;
- профилей пользователей;
- статей;
- услуг.

Навигационная панель

Bootstrap позволяет быстро создать адаптивное меню.

```

<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container">
    <a class="navbar-brand" href="#">Мой сайт</a>

    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-
bs-target="#navbarMenu">
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarMenu">
      <ul class="navbar-nav ms-auto">

```

```
<li class="nav-item">
  <a class="nav-link active" href="#">Главная</a>
</li>
<li class="nav-item">
  <a class="nav-link" href="#">О нас</a>
</li>
<li class="nav-item">
  <a class="nav-link" href="#">Контакты</a>
</li>
</ul>
</div>
</div>
</nav>
```

Класс `navbar-expand-lg` означает, что меню будет раскрыто на больших экранах, а на меньших превратится в кнопку.

JavaScript-компоненты Bootstrap

Некоторые компоненты Bootstrap требуют JavaScript.

К ним относятся:

- `modal` — модальные окна;
- `dropdown` — выпадающие меню;
- `carousel` — слайдер;
- `collapse` — сворачиваемые блоки;
- `tooltip` — всплывающие подсказки;
- `offcanvas` — боковая панель.

Если подключить файл:

```
<script  
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js">  
</script>
```

то большинство компонентов будут работать без дополнительного JavaScript-кода.

Модальное окно

Пример модального окна:

```
<button type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#exampleModal">
```

Открыть окно

```
</button>
```

```
<div class="modal fade" id="exampleModal" tabindex="-1">
```

```
<div class="modal-dialog">
```

```
<div class="modal-content">
```

```
<div class="modal-header">
```

```
<h5 class="modal-title">Заголовок окна</h5>
```

```
<button type="button" class="btn-close" data-bs-dismiss="modal"></button>
```

```
</div>
```

```
<div class="modal-body">
```

Содержимое модального окна.

```
</div>
```

```
<div class="modal-footer">
```

```
<button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Закрыть</button>
```

```
<button type="button" class="btn btn-primary">Сохранить</button>
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

Пример адаптивной страницы на Bootstrap

Ниже пример простой адаптивной страницы с меню, карточками и формой.

```
<!DOCTYPE html>
```

```
<html lang="ru">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Адаптивный сайт Bootstrap</title>
```

```
<link
```

```
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet">
```

```
</head>
```

```
<body>
```

```
<nav class="navbar navbar-expand-lg bg-dark navbar-dark">
```

```
<div class="container">
```

```
<a class="navbar-brand" href="#">WebCourse</a>
```

```
<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-
bs-target="#menu">
```

```
<span class="navbar-toggler-icon"></span>
```

```
</button>
```

```
<div class="collapse navbar-collapse" id="menu">
```

```
<ul class="navbar-nav ms-auto">
  <li class="nav-item"><a class="nav-link active" href="#">Главная</a></li>
  <li class="nav-item"><a class="nav-link" href="#">Курсы</a></li>
  <li class="nav-item"><a class="nav-link" href="#">Контакты</a></li>
</ul>
</div>
</div>
</nav>
```

```
<header class="bg-primary -white -center py-5">
  <div class="container">
    <h1>Основы WEB программирования</h1>
    <p class="lead">Bootstrap и адаптивная разработка</p>
    <a href="#" class="btn btn-light btn-lg">Начать обучение</a>
  </div>
</header>
```

```
<main class="container my-5">
```

```
<section class="mb-5">
  <h2 class="-center mb-4">Темы лекции</h2>

  <div class="row g-4">
    <div class="col-12 col-md-6 col-lg-4">
      <div class="card h-100">
        <div class="card-body">
          <h5 class="card-title">Адаптивность</h5>
          <p class="card- ">Принципы создания сайтов для разных
устройств.</p>
```

[Подробнее](#)

Bootstrap Grid

Использование контейнеров, строк и колонок.

[Подробнее](#)

Компоненты

Кнопки, формы, карточки, меню и модальные окна.

[Подробнее](#)

```
<h2 class="mb-4">Форма обратной связи</h2>
```

```
<form>
```

```
<div class="mb-3">
```

```
<label for="name" class="form-label">Имя</label>
```

```
<input type="text" class="form-control" id="name" placeholder="Введите  
имя">
```

```
</div>
```

```
<div class="mb-3">
```

```
<label for="email" class="form-label">Email</label>
```

```
<input type="email" class="form-control" id="email" placeholder="Введите  
email">
```

```
</div>
```

```
<div class="mb-3">
```

```
<label for="message" class="form-label">Сообщение</label>
```

```
<div class="form-control" id="message" rows="4"></div>
```

```
</div>
```

```
<button type="submit" class="btn btn-success">Отправить</button>
```

```
</form>
```

```
</section>
```

```
</main>
```

```
<footer class="bg-dark text-white text-center py-3">
```

```
<div class="container">
```

```
&copy; 2026 WebCourse
```

```
</div>

</footer>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js">
</script>

</body>

</html>
```

Bootstrap Icons

Bootstrap также имеет отдельную библиотеку иконок — **Bootstrap Icons**.

Подключение:

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-
icons@1.11.3/font/bootstrap-icons.css">
```

Пример использования:

```
<i class="bi bi-house"></i>

<i class="bi bi-envelope"></i>

<i class="bi bi-github"></i>
```

Когда стоит использовать Bootstrap

Bootstrap хорошо подходит для:

- учебных проектов;
- административных панелей;
- прототипов;
- корпоративных сайтов;
- лендингов;
- веб-приложений с типовыми интерфейсами.

Bootstrap особенно полезен, когда нужно быстро получить аккуратный и адаптивный интерфейс.

Недостатки Bootstrap

Несмотря на преимущества, Bootstrap имеет и ограничения:

- сайты могут выглядеть однотипно;
- при неправильном использовании HTML может быть перегружен классами;
- не всегда удобно создавать уникальный дизайн;
- часть CSS может быть неиспользуемой;
- для глубокого изменения темы нужно разбираться в переменных и Sass.

Рекомендации по использованию Bootstrap

1. Не использовать Bootstrap как замену знаниям HTML и CSS.
2. Понимать, как работает сетка.
3. Использовать документацию.
4. Не перегружать страницу лишними компонентами.
5. Подключать JavaScript Bootstrap только если нужны интерактивные компоненты.
6. Для уникального дизайна добавлять собственный CSS.
7. Соблюдать семантику HTML.
8. Проверять сайт на разных размерах экрана.

Практическое задание

Создать адаптивную страницу сайта учебного курса с использованием Bootstrap.

Страница должна содержать:

1. адаптивную навигационную панель;

2. главный блок с заголовком и кнопкой;
3. секцию с тремя карточками;
4. форму обратной связи;
5. футер.

Требования

- Использовать Bootstrap 5.
- Использовать сетку Bootstrap.
- Карточки должны располагаться:
 - ✓ по одной в строке на смартфоне;
 - ✓ по две в строке на планшете;
 - ✓ по три в строке на большом экране.
- Форма должна содержать поля:
 - ✓ имя;
 - ✓ email;
 - ✓ сообщение;
 - ✓ кнопку отправки.
- Страница должна корректно отображаться на мобильных устройствах.

Контрольные вопросы КТ 2

1. Что такое адаптивный веб-дизайн?
2. Для чего используется тег viewport?
3. Что такое Bootstrap?
4. Какие преимущества даёт Bootstrap?
5. Что такое контейнер в Bootstrap?
6. Чем отличается .container от .container-fluid?
7. Из скольких колонок состоит сетка Bootstrap?
8. Для чего используется класс .row?
9. Что означает класс col-md-6?

10. Какие breakpoints используются в Bootstrap 5?
11. Что такое Mobile First?
12. Как создать кнопку в Bootstrap?
13. Какие классы используются для оформления форм?
14. Какие компоненты Bootstrap требуют JavaScript?
15. Для чего используется класс navbar-expand-lg?
16. Как создать модальное окно в Bootstrap?
17. Какие недостатки есть у Bootstrap?
18. Когда Bootstrap использовать целесообразно?

Bootstrap — это удобный инструмент для быстрой разработки адаптивных сайтов и веб-приложений. Однако важно понимать, что он не заменяет знания HTML, CSS и JavaScript, а лишь помогает эффективнее применять их на практике.

Благодарю за внимание!

