

## Лабораторная работа 1: Вёрстка сайта средствами HTML, CSS и Bootstrap

**Студент:** Дмитриев Андрей Иванович

**Группа:** К3439

**Вариант:** Платформа для образовательных курсов и управления учебным процессом

---

### 1. Описание проекта

В рамках лабораторной работы разработано веб-приложение для управления образовательными курсами — **IT School**. Приложение представляет собой полнофункциональную систему управления обучением с поддержкой различных ролей пользователей.

#### Основной функционал:

- Система аутентификации и авторизации пользователей
  - Личный кабинет с отображением персональной информации
  - Просмотр и управление курсами
  - Система расписания занятий
  - Управление домашними заданиями
  - Система рейтингов и баллов (монеты)
  - Интернет-магазин для обмена баллов
  - Система уведомлений
- 

### 2. Реализованные страницы

#### 2.1 Страница входа (LoginPage.jsx)

##### Функционал:

- Форма авторизации с полями логин/пароль
- Переключение видимости пароля
- Валидация введенных данных
- Обработка ошибок аутентификации
- Ссылка на восстановление пароля

##### Технологии:

- HTML5 семантическая разметка
- CSS3 (flexbox для выравнивания)
- React hooks (useState, useEffect)
- Адаптивный дизайн

##### Ключевые элементы:



---

## 2.3 Личный кабинет (ProfilePage.jsx)

### Функционал:

- Отображение полной информации о пользователе
- Редактирование профиля
- Загрузка аватара
- Изменение пароля
- История активности

### Формы данных:

- ФИО (Фамилия, Имя, Отчество)
- Email и номер телефона
- Дата рождения
- Роль в системе (студент/преподаватель/администратор)

---

## 2.4 Страница курсов студента (StudentCoursesPage.jsx)

### Функционал:

- Список доступных курсов с фильтрацией
- Карточки курсов с изображениями
- Прогресс-бары выполнения курса
- Информация о возрастной категории
- Разделение на "Мои курсы" и "Доступные курсы"

### Карточка курса включает:

- Изображение курса
- Название и описание
- Progress bar (процент выполнения)
- Количество уроков
- Возрастная категория
- Кнопка перехода к курсу

### CSS Grid для карточек:

```
.courses-grid {  
  display: grid;
```

```
grid-template-columns: repeat(auto-fill, minmax(300px, 1fr));
gap: 20px;
padding: 20px;
}
```

---

## 2.5 Страница детального просмотра курса (StudentCoursePage.jsx)

### Функционал:

- Список всех уроков курса
  - Статус прохождения каждого урока
  - Материалы к урокам
  - Домашние задания
  - Прогресс по курсу
- 

## 2.6 Страница расписания (SchedulePage.jsx)

### Функционал:

- Календарное представление занятий
- Фильтрация по курсам, преподавателям, группам
- Просмотр деталей занятия
- Переключение между видами (месяц/неделя/день)

**Используемая библиотека:** FullCalendar React

---

## 2.7 Страница рейтинга (RatingPage.jsx)

### Функционал:

- Таблица топ студентов
  - Сортировка по баллам
  - Фильтрация по группам/курсам
  - Отображение позиции текущего пользователя
- 

## 2.8 Страница магазина (ShopPage.jsx)

### Функционал:

- Каталог товаров для обмена на баллы
- Карточки товаров с фото и описанием

- Модальное окно покупки
- История покупок
- Фильтрация товаров

---

### 3. Использование HTML5 и семантической разметки

#### Семантические теги:

- `<header>` — шапка страницы с навигацией
- `<nav>` — навигационное меню
- `<main>` — основной контент страницы
- `<section>` — логические разделы контента
- `<article>` — независимые статьи (новости, карточки курсов)
- `<aside>` — боковая панель (Sidebar)
- `<footer>` — подвал страницы
- `<form>` — формы входа, регистрации, редактирования

#### Атрибуты доступности:

- `aria-label` — метки для скрин-ридеров
- `role` — роли элементов
- `tabindex` — порядок навигации с клавиатуры
- `alt` — альтернативный текст для изображений

---

### 4. Использование CSS и CSS Grid/Flexbox

#### 4.1 CSS Flexbox

Используется для:

- Выравнивание элементов в навигации
- Центрирование модальных окон
- Расположение элементов форм
- Адаптивные кнопки и контролы

#### Примеры:

```
.login-container {  
  display: flex;  
  justify-content: center;  
  align-items: center;
```

```
min-height: 100vh;  
}
```

```
.topbar {  
  display: flex;  
  justify-content: space-between;  
  align-items: center;  
  padding: 15px 30px;  
}
```

## 4.2 CSS Grid

Используется для:

- Сетка карточек курсов
- Layout главной страницы
- Таблицы данных
- Галереи изображений

### Примеры:

```
.courses-grid {  
  display: grid;  
  grid-template-columns: repeat(auto-fill, minmax(300px, 1fr));  
  gap: 20px;  
}
```

```
.points-grid {  
  display: grid;  
  grid-template-columns: 1fr 1fr;  
  gap: 16px;  
}
```

## 4.3 Адаптивность

Media queries для различных устройств:

```
@media (max-width: 768px) {  
  .sidebar {  
    transform: translateX(-100%);
```

```

}

.courses-grid {
  grid-template-columns: 1fr;
}

}

@media (max-width: 480px) {
  .topbar {
    flex-direction: column;
  }
}

```

---

## 5. JavaScript функционал

### 5.1 Модальные окна

- Открытие/закрытие модальных окон новостей
- Модальные окна событий
- Модальные окна покупки товаров
- Модальные окна редактирования профиля

#### Пример реализации:

```
const [modallItem, setModallItem] = useState(null);
```

```
const openModal = (item) => {
  setModallItem(item);
};
```

```
const closeModal = () => {
  setModallItem(null);
};
```

### 5.2 Интерактивные формы

- Валидация полей
- Динамическое отображение ошибок

- Переключение видимости пароля
- Автозаполнение

### 5.3 Динамическое обновление контента

- Обновление списка курсов при изменении фильтров
  - Обновление расписания
  - Live-поиск по курсам/студентам
  - Обновление счетчика баллов
- 

## 6. Компонентная архитектура

### Переиспользуемые компоненты:

1. **Sidebar** — боковое меню навигации
  2. **TopBar/SmartTopBar** — верхняя панель
  3. **CourseCard** — карточка курса
  4. **CourseImage** — изображение курса с placeholder
  5. **CourseProgressBar** — прогресс-бар курса
  6. **Toast** — уведомления
  7. **Modal** — базовые модальные окна
  8. **SearchableSelect** — выпадающий список с поиском
- 

## 7. Изоляция стилей

Каждая страница и компонент имеет свой CSS файл:

- HomePage.css
- LoginPage.css
- CourseCard.css
- Schedule.css
- ProfilePage.css

Использование БЭМ-подобной методологии именования классов:

```
.course-card {}
```

```
.course-card__image {}
```

```
.course-card__title {}
```

```
.course-card--disabled {}
```

---



## 8. Адаптивный дизайн

### Breakpoints:

- Desktop: > 1024px
- Tablet: 768px - 1024px
- Mobile: < 768px

### Адаптивные изменения:

- Скрытие Sidebar на мобильных устройствах
- Трансформация Grid в одну колонку
- Изменение размеров шрифтов
- Упрощение навигации
- Мобильное меню (гамбургер)

**Файл:** MobileImageFixes.css

```
@media (max-width: 768px) {  
  .course-card-image {  
    height: 180px;  
    object-fit: cover;  
  }  
}
```

---

## 9. Используемые технологии

### Frontend:

- **React 19.1.0** — библиотека для построения UI
- **React Router DOM 7.6.0** — маршрутизация
- **Axios 1.9.0** — HTTP клиент
- **FullCalendar 6.1.17** — компонент календаря
- **React Toastify 11.0.5** — уведомления

### Инструменты:

- **CRACO** — настройка конфигурации
- **CSS Minimizer** — оптимизация CSS

### Стилизация:

- CSS3 (Custom Properties, Flexbox, Grid)
- Адаптивный дизайн

- Анимации и переходы

---

## 10. Структура проекта

front\_it\_school/

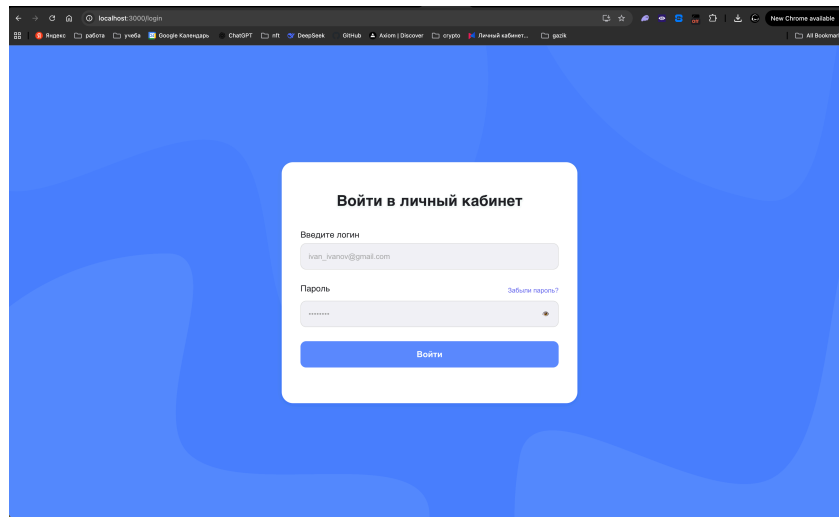
```
├── public/
│   ├── index.html
│   ├── manifest.json
│   └── robots.txt
├── src/
│   ├── api/
│   │   └── axiosInstance.js
│   ├── components/
│   │   ├── Sidebar.jsx
│   │   ├── TopBar.jsx
│   │   ├── CourseCard.jsx
│   │   ├── Schedule.jsx
│   │   └── ... (другие компоненты)
│   ├── contexts/
│   │   ├── AuthContext.js
│   │   └── NotificationsContext.js
│   ├── pages/
│   │   ├── HomePage.jsx
│   │   ├── LoginPage.jsx
│   │   ├── ProfilePage.jsx
│   │   ├── StudentCoursesPage.jsx
│   │   └── ... (другие страницы)
│   ├── services/
│   │   ├── courseService.js
│   │   ├── userService.js
│   │   └── ... (другие сервисы)
│   └── styles/
│       └── HomePage.css
```

- | | | — LoginPage.css
- | | | — ... (другие стили)
- | | — App.js
- | | — index.js
- | — package.json
- | — craco.config.js

---

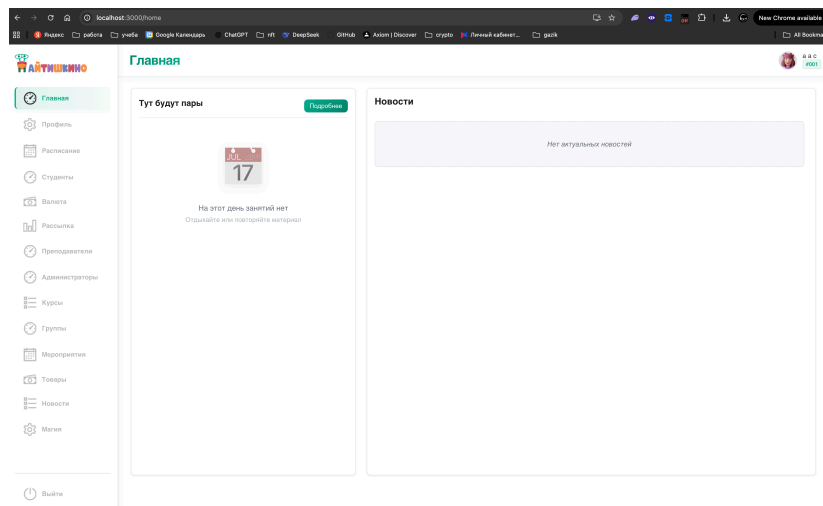
## 11. Скриншоты

### 11.1 Страница входа



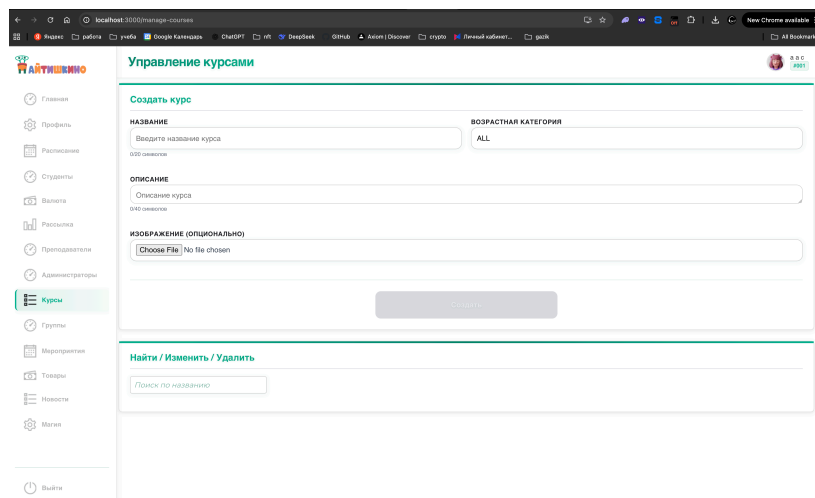
*Форма авторизации с валидацией и обработкой ошибок*

### 11.2 Главная страница



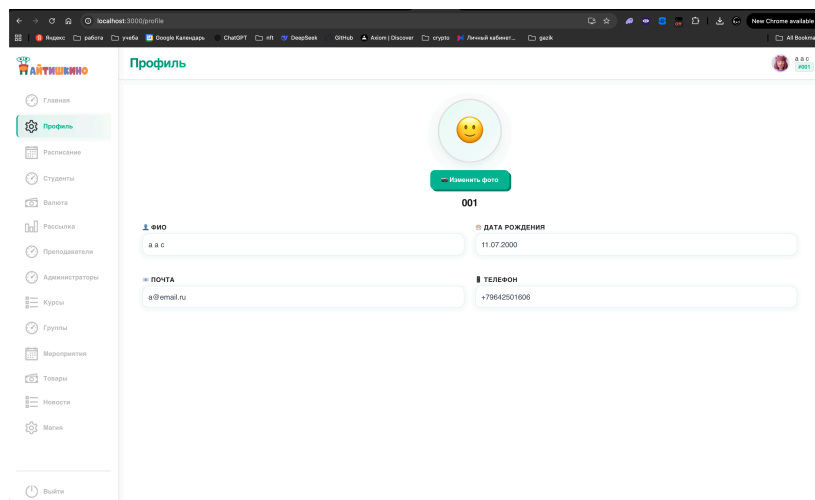
*Главная страница с расписанием, новостями и рейтингом*

### 11.3 Страница курсов



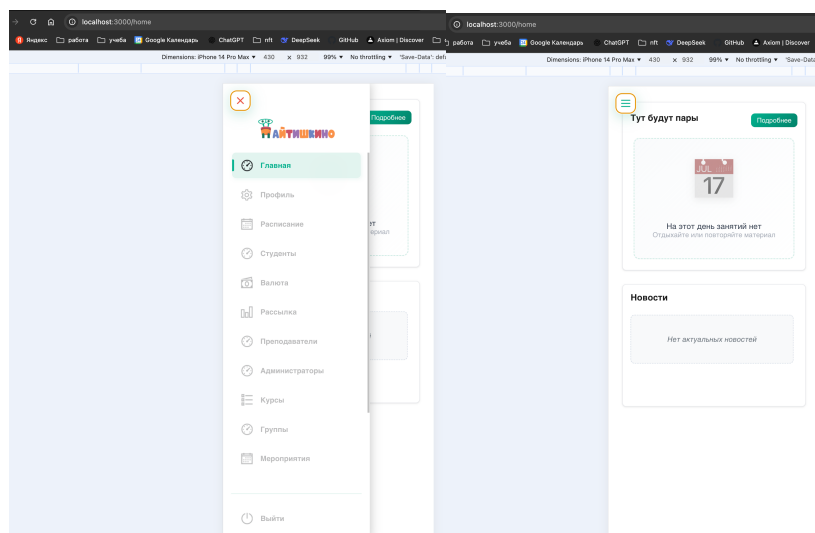
*Каталог курсов с прогресс-барами и фильтрацией*

## 11.4 Личный кабинет



*Личный кабинет пользователя*

## 11.5 Адаптивная версия



*Мобильная версия приложения*

---

## 12. Выводы

В ходе выполнения лабораторной работы были получены следующие навыки:

1. **HTML5 и семантическая вёрстка** — использование современных семантических тегов для улучшения структуры документа
2. **CSS3** — применение Flexbox и Grid для создания адаптивных layouts
3. **Адаптивный дизайн** — создание интерфейса, корректно отображающегося на различных устройствах
4. **JavaScript** — реализация интерактивных элементов (модальные окна, формы, валидация)
5. **Компонентный подход** — разделение приложения на переиспользуемые компоненты
6. **React** — использование современной библиотеки для построения пользовательских интерфейсов

**Дата выполнения:** 21.01.2026