

АДАПТИВНАЯ ВЁРСТКА И ДИНАМИЧЕСКИЕ ЭФФЕКТЫ НА CSS

Современные веб-приложения требуют гибкости в отображении контента на различных устройствах. CSS (Cascading Style Sheets) играет ключевую роль в создании адаптивного дизайна и реализации динамических эффектов.

1. Адаптивная верстка позволяет сделать сайт удобным для использования на мобильных устройствах, планшетах и настольных компьютерах. Основные подходы включают:

- использование медиа-запросов (media queries) для изменения стилей в зависимости от ширины экрана;
- гибкие сетки (Flexbox, CSS Grid), позволяющие легко адаптировать расположение элементов;

- относительные единицы измерения (% , em, rem, vw, vh) ;
- оптимизация изображений через srcset и picture;

2. CSS предоставляет мощные возможности для анимации и интерактивности веб-страниц. Среди наиболее популярных технологий:

- CSS-анимации (@keyframes), позволяющие изменять свойства элементов плавно;
- переходы (transition), обеспечивающие анимированные изменения при наведении или клике;
- фильтры и эффекты (filter, box-shadow, backdrop-filter);
- использование hover, focus, active для создания интерактивных элементов;
- свойство transform (scale, rotate, translate) для анимированных преобразований.

3. Для демонстрации возможностей адаптивной верстки и динамических эффектов можно реализовать следующий код (рисунок 1).

```
.container{
  display: flex;
  justify-content: center;
  align-items: center;
  height: 100vh;
}

.button {
  background: linear-gradient(45deg, #fff6b6b, #f06595);
  padding: 10px 20px;
  border-radius: 8px;
  color: white;
  position: relative;
  transition: transform 0.3s ease-in-out, box-shadow 0.5s ease-in-out;
  box-shadow: 0 0 0 rgba(255, 223, 0, 0); /* Изначально тень невидима */
}

.button:hover {
  transform: scale(1.5);
  box-shadow: 0 0 20px 10px rgba(255, 223, 0, 0.7); /* Добавляем плавное свечение */
}

@media screen and (max-width: 768px) {
  .container {
    display: flex;
    justify-content: center;
    height: 100vh;
  }

  .button {
    margin-top: auto; /* Заставляет кнопку уходить в конец контейнера */
    align-self: center; /* Центрирует кнопку по горизонтали */
    margin-bottom: 40px; /* Отступ от низа */
  }
}
```

Рисунок 1 – Пример кода с адаптивной версткой и динамическими эффектами

В результате получается анимированная кнопка (рисунок 2), которая при наведении на нее увеличивается в размерах и вокруг нее появляется свечение (рисунок 3), а также изменяет свое положение: с центрального по горизонтали и вертикали - на положение внизу экрана, если размер экрана менее 768 пикселей (рисунок 4 (а), и также, при наведении курсора на кнопку, увеличивается и появляется свечение (рисунок 4 (б)).

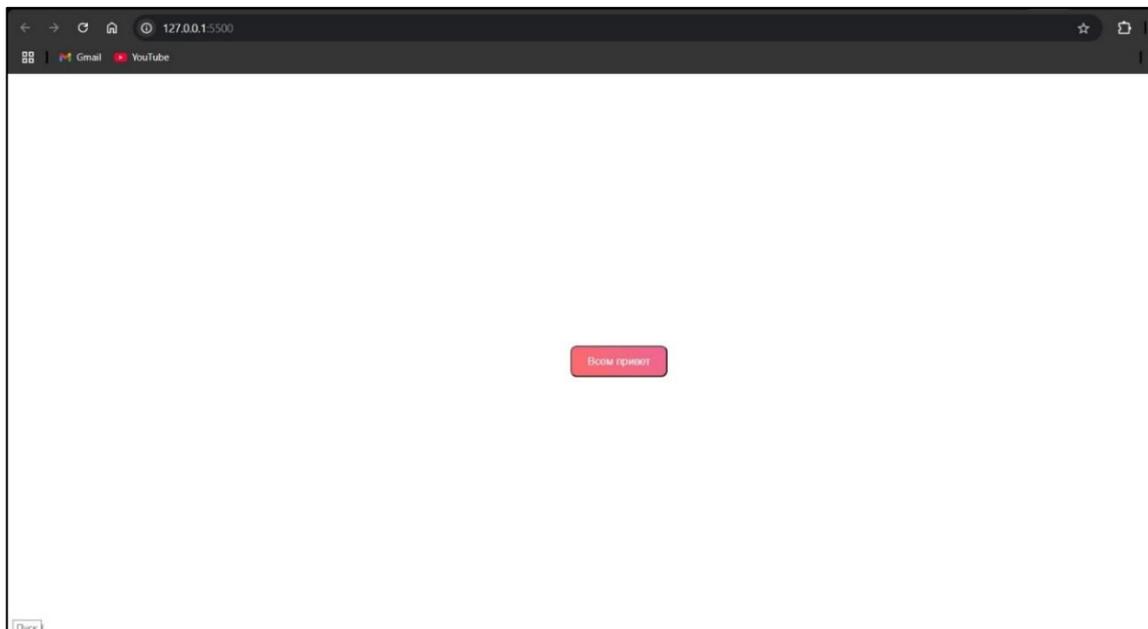
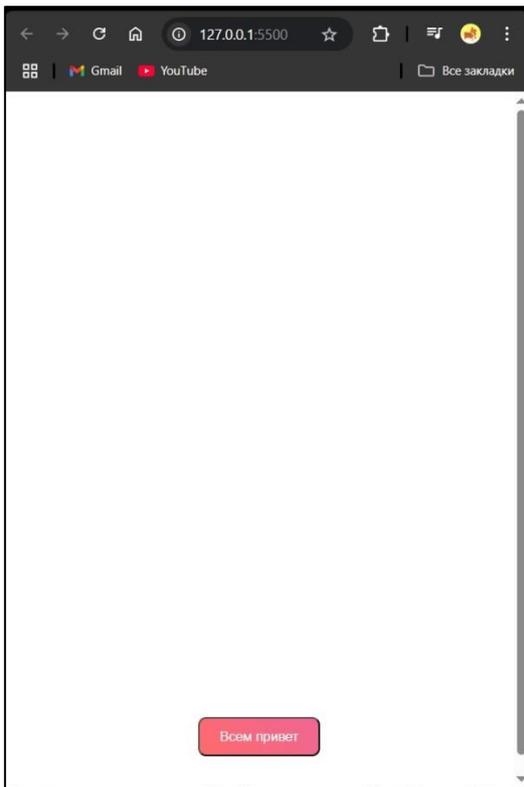


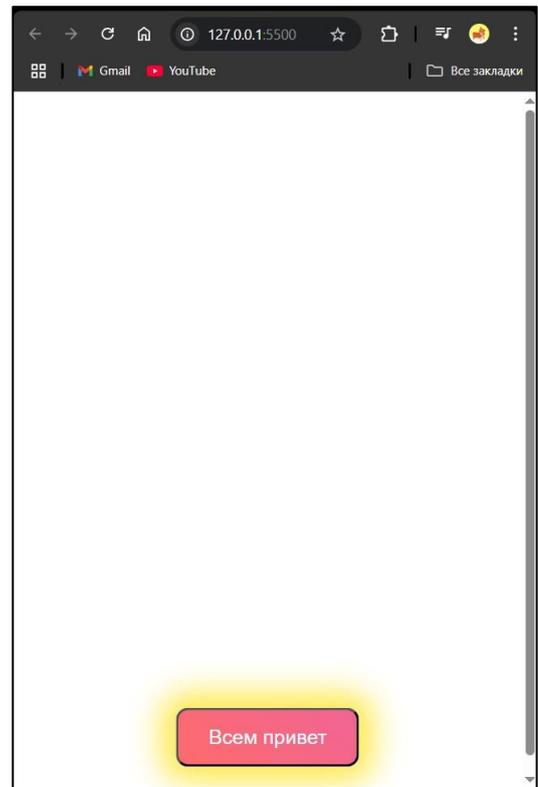
Рисунок 2 – Отображение страницы с размером экрана более 768 px



Рисунок 3 – Отображение страницы с размером экрана более 768 px и курсором наведенным на кнопку



а



б

Рисунок 4 – а – отображение страницы с размером экрана менее 768 px;
б – отображение страницы с размером экрана менее 768 px
и курсором наведенным на кнопку

4. Заключение

Применение адаптивной верстки и динамических эффектов на CSS позволяет улучшить пользовательский опыт и повысить вовлеченность аудитории. Развитие технологий CSS, таких как Flexbox, Grid и анимации, делает веб-страницы более удобными, кроссплатформенными и визуально привлекательными.

Литература

1. Фелке-Морес, Э. CSS: каскадные таблицы стилей для профессионалов / Э. Фелке-Морес. – 2020. – 786 с.
2. Ковер, К. HTML и CSS. Разработка и дизайн веб-сайтов / К. Ковер. – 2018. – 640 с.
3. Мейер, Э. CSS. Подробное руководство / Э. Мейер. – 2019. – 704 с.