|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ федеральное государственное учреждение высшего образования «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ» | | | | | | | | | | | | | |
|  | | | Автоматизированные системы управления | | | | | | | | | | |
|  | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | |
| **КУРСОВАЯ РАБОТА(ПРОЕКТ)** | | | | | | | | | | | | | |
| по дисциплине: WEB - технологии | | | | | | | | | | | | | |
| на тему: спроектировать и разработать Web-сайт | | | | | | | | | | | | | |
| Кино и сериалы | | | | | | | | | | | | | |
| Вариант № 21 | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | |
|  |  | Студента | | 1 | курса группы ИС | | | | | | 24а | |  |
|  |  | Направления подготовки: ИСТ | | | | | | | | | | | |
|  |  | Специальности: ИС | | | | | | | | | | | |
|  |  | Шевченко Владислава Евгеньевича | | | | | | | | | | | |
|  |  | (ФИО) | | | | | | | | | | | |
|  |  |  | | | | | | | | | | | |
|  |  | Руководитель: ст. пр. Андриевская Н.К. | | | | | | | | | | | |
|  |  | Национальная шкала | | | | |  | | | | | | |
|  |  |  | | | | |  | | | | | | |
|  |  | Количество баллов: | | | |  | | Оценка: ECTS | | | |  | |
|  |  |  | | | |
|  |  |  | | | |  | |  | | | |  | |
|  |  |  | | | |  | |  | | | |  | |
|  |  | Члены комиссии | | |  | | | |  | Золушкин Ю.А. | | | |
|  |  |  | | | (подпись) | | | |  | (ФИО) | | | |
|  |  |  | | |  | | | |  | Новиков Д. Д. | | | |
|  |  |  | | | (подпись) | | | |  | (ФИО) | | | |
|  |  |  | | |  | | | |  |  | | | |
|  |  |  | | |  | | | |  |  | | | |
|  |  |  | | |  | | | |  |  | | | |

Донецк – 2025

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ федеральное государственное учреждение высшего образования «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ» | | | | | | | | | | |
| Кафедра: Автоматизированные системы управления | | | | | | | | | | |
| Дисциплина: Web-технологии | | | | | | | | | | |
| Специальность: 09.03.02 Информационные системы и технологии в технике и бизнеса | | | | | | | | | | |
| Курс | **1** | Группа | **ИС-24а** | | | Семестр | **2** | |  | |
| **З А Д А Н И Е** | | | | | | | | | | |
| **на курсовую работу(проект) студента** | | | | | | | | | | |
| ШЕВЧЕНКО ВЛАДИСЛАВ ЕВГЕНЬЕВИЧ | | | | | | | | | | |
| (ФИО) | | | | | | | | | | |
| 1. Тема курсового работы | | | | | спроектировать и разработать Web-сайт с | | | | | |
|  | | | | использованием выданного шаблона, который | | | | | | |
| содержит информацию по индивидуальной теме – кино и сериалы | | | | | | | | | | |
|  | | | | | | | | | | |
|  | | | | | | | | | | |
| 2. Сроки сдачи студентом законченной работы | | | | | | | | 26 мая 2025 | |  |
| 3. Выходные данные к проекту | | | | | |  | | | | |
| Созданный на основе индивидуального шаблона веб-сайт, размещенный | | | | | | | | | | |
| в сети интернет с помощью хостинга, и макеты страниц, которые отличны | | | | | | | | | | |
| друг от друга | | | | | | | | | | |
|  | | | | | | | | | | |
| 4. Содержание расчетно-пояснительной записки (перечень вопросов, которые подлежат разработке) | | | | | | | | | | |
| Как создать сайт на языке разметки HTML? Как придавать элементам | | | | | | | | | | |
| необходимый стиль? Как использовать технологии Grid-layout и Flexbox | | | | | | | | | | |
| для размещения контента на странице? Как сделать сайт адаптивным к | | | | | | | | | | |
| разным размерам экранов? | | | | | | | | | | |
|  | | | | | | | | | | |
| 5. Перечень графического материала (с точным указанием обязательных чертежей) | | | | | | | | | | |
| Макет страниц на разных экранах, схема каталога с файлами, изображение | | | | | | | | | | |
| логической структуры сайта | | | | | | | | | | |
|  | | | | | | | | | | |
| 6. Дата выдачи задания | | | | 10.04.2025 | | |  | | | |

**КАЛЕНДАРНЫЙ ПЛАН**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Название этапов курсового работы (проекта) | Период выполнения этапов работы | | Примечание |
| 1 | Выбор типа верстки | 12.04.2025 | | –– |
| 2 | Проектирование структуры сайта | 12.04.2025 – 14.04.2025 | | –– |
| 3 | Проектирование структуры каталогов | 15.04.2025 | | –– |
| 4 | Создание прототипа | 16.04.2025 - 16.04.2025 | | –– |
| 5 | Разработка и наполнение страниц | 17.04.2025 – 19.04.2025 | | –– |
| 6 | Разработка стилей | 19.04.2025 – 25.04.2025 | | –– |
| 7 | Разработка адаптивного шаблона сайта | 26.04.2025 – 28.04.2025 | | –– |
| 8 | Реализация сайта | 29.04.2025 – 29.04.2025 | | –– |
| 9 | Тестирование | 02.05.2025 – 16.05.2025 | | –– |
| 10 | Размещение сайта и продвижение | 19.05.2025 | | –– |
|  |  |  | |  |
| Студент |  |  | Шевченко В. Е. | |
|  | (подпись) |  | (ФИО) | |
| Руководитель |  |  | Андриевская Н. К. | |
|  | (подпись) |  | (ФИО) | |

«\_11\_» \_\_\_\_\_\_\_мая\_\_\_\_\_\_\_ 2025 г.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **РЕФЕРАТ**  Пояснительная записка: 26 с., 2 рис.,  8 приложений, 5 источников.  САЙТ, ПРОТОТИП, ШАБЛОН, СВОЙСТВА, ВЕБ-СЕРВЕР, ВЕБ-СТРАНИЦА, CSS, ОБЪЕКТ, ИДЕНТИФИКАТОР, WEB-ИНТЕРФЕЙС, HTML.  Целью курсовой работы(проекта) является создание сайта по выданному шаблону и индивидуальной теме с помощью языка разметки HTML, таблиц стилей CSS, с последующим размещением созданного веб-сайта в сети Интернет посредством бесплатного хостинга  В результате проектирования был разработан функциональный и информативный веб-сайт о языках программирования с предельно понятным и удобным интерфейсом, который размещен в сети Интернет. | | | | | | | | | | |
|  |  |  |  |  | КР(КП) 09.03.02-25-907-001ПЗ | | | | | |
|  |  |  |  |  |
| Изм. | Лист | № документа | Подпись | Дата |
| Студент | | Шевченко В.Е. |  |  | Пояснительная записка | Лит | | | Лист | Листов |
| Рук. | | Андриевская Н.К. |  |  |  | У |  | 4 | 36 |
| Консульт. | | Золушкин Ю. А. |  |  | ДонНТУ, ФИСТ,  Кафедра АСУ, гр. ИС | | | | |
|  | | Новиков Д.Д. |  |  |
| Н.контр. | | Андриевская Н.К. |  |  |

# СОДЕРЖАНИЕ

[СОДЕРЖАНИЕ 5](#_Toc198860663)

[ВВЕДЕНИЕ 6](#_Toc198860664)

[ОСНОВНАЯ ЧАСТЬ 7](#_Toc198860665)

[1. Постановка задачи проектирования и выбор типа верстки 7](#_Toc198860666)

[2. Проектирование структуры сайта 8](#_Toc198860667)

[3. Проектирование структуры каталогов 10](#_Toc198860668)

[4. Создание прототипа. 14](#_Toc198860669)

[5. Разработка и наполнение HTML-страниц 15](#_Toc198860670)

[6. Разработка стилей для реализации интерфейса Web-сайта 17](#_Toc198860671)

[7. Разработка адаптивного шаблона сайта 22](#_Toc198860672)

[8. Реализация сайта 25](#_Toc198860673)

[9. Тестирование 26](#_Toc198860674)

[10. Размещение сайта и выбор средств для повышения его рейтинга 27](#_Toc198860675)

[ЗАКЛЮЧЕНИЕ 30](#_Toc198860676)

[СПИСОК ЛИТЕРАТУРЫ 31](#_Toc198860677)

[ПРИЛОЖЕНИЕ A. 32](#_Toc198860678)

[ПРИЛОЖЕНИЕ Б. 33](#_Toc198860679)

[ПРИЛОЖЕНИЕ В. 34](#_Toc198860680)

[ПРИЛОЖЕНИЕ Г 35](#_Toc198860681)

# ВВЕДЕНИЕ

В эпоху стремительного развития информационных технологий интернет стал неотъемлемой частью жизни современного человека. Он предоставляет широкий спектр возможностей для получения информации, общения, обучения и развлечений. Особенно популярной областью цифрового пространства является индустрия кино и сериалов. Благодаря стриминговым сервисам и онлайн-платформам доступ к фильмам и сериалам значительно упростился, а объём контента постоянно растёт.

На этом фоне возрастает потребность в удобных и функциональных веб-ресурсах, позволяющих быстро находить информацию о фильмах, жанрах, новинках и рейтингах. Важно, чтобы веб-ресурс был не только информативным, но и визуально понятным, адаптивным и простым в использовании.

Настоящая курсовая работа посвящена созданию информационного web-сайта, посвящённого кино и сериалам. В рамках проекта разрабатывается удобный и современный сайт с каталогом фильмов и телесериалов, включающим описания, жанры, рейтинги и медиаматериалы (статьи). Основная цель — создать ресурс, способный обеспечить доступ к актуальной информации и улучшить пользовательский опыт.

Разработка сайта осуществляется с применением современных технологий веб-программирования, включая HTML, CSS и адаптивную вёрстку. Особое внимание уделяется логике построения структуры сайта, интерфейсу, дизайну и удобству взаимодействия. Заключительным этапом является публикация сайта в интернете и его тестирование в реальных условиях.

Таким образом, в рамках курсовой работы решается практическая задача по разработке актуального веб-ресурса, что позволяет применить полученные знания и навыки в области веб-разработки и внести вклад в развитие цифровой среды в сфере киноиндустрии.

# ОСНОВНАЯ ЧАСТЬ

## Постановка задачи проектирования и выбор типа верстки

Цель проектирования — создание функционального, визуально привлекательного и удобного для пользователя веб-сайта, соответствующего современным требованиям веб-дизайна. В процессе разработки определяется структура страниц, подбирается тип вёрстки и планируются способы адаптации интерфейса под разные устройства.

На основе предложенного шаблона выбирается наиболее подходящий тип вёрстки. Существуют два основных подхода: табличная и блочная. Табличная вёрстка основана на использовании HTML-таблиц для построения макета страницы. Несмотря на простоту реализации, она обладает рядом недостатков: низкой гибкостью, сложностью адаптации под мобильные устройства и несоответствием современным стандартам. В связи с этим такой подход считается устаревшим и на практике практически не применяется.

Блочная вёрстка, напротив, использует каскадные таблицы стилей (CSS) и построение интерфейса на основе логически разделённых блоков. Этот метод обеспечивает гибкость, удобное управление расположением элементов, а также поддерживает адаптивный дизайн. Преимуществом блочной вёрстки является разделение структуры и оформления, что упрощает сопровождение и масштабирование проекта.

В рамках данного проекта применяется именно блочная вёрстка. Основные элементы страниц включают: шапку сайта (header), панель навигации (nav), визуальный блок (например, постер), основное содержимое (main), боковые колонки (aside) и подвал (footer). В зависимости от задач может применяться двух- или трёхколоночная структура. Ширина колонок может быть как фиксированной, так и адаптивной — в зависимости от требований дизайна.

Для построения макета используются современные CSS-технологии: Flexbox и Grid Layout. Они позволяют гибко управлять расположением элементов и создавать адаптивные интерфейсы без использования устаревших решений.

Разработка сайта предусматривает реализацию адаптивного дизайна, обеспечивающего корректное отображение на различных устройствах — от настольных ПК до мобильных телефонов. Для этого применяются медиа-запросы, позволяющие перестраивать сетку элементов в зависимости от разрешения экрана пользователя.

## Проектирование структуры сайта

При проектировании структуры сайта основное внимание было уделено созданию максимально удобной и понятной навигации, обеспечивающей быстрый доступ к необходимой информации без лишних переходов. В качестве приоритета рассматривалось формирование логически выверенной структуры, соответствующей принципам интуитивного восприятия и простоты использования.

При построении логической архитектуры сайта был применён принцип доступности каждого раздела или страницы не более чем в три клика от главной страницы. Такой подход позволяет значительно сократить время поиска информации и способствует улучшению пользовательского опыта. Навигационные элементы размещены таким образом, чтобы быть видимыми сразу после загрузки страницы, что повышает удобство ориентирования и устраняет необходимость дополнительных действий по поиску меню.

Особое внимание было уделено организации двунаправленных внутренних связей между разделами. Это обеспечивает свободное перемещение по сайту без потери контекста и без ограничений в навигации. Кроме того, на каждой странице реализована возможность быстрого возврата на главную страницу, что способствует сохранению ориентации пользователя в структуре ресурса.

Названия разделов подбирались с учётом их ясности и однозначности, что позволяет даже новым пользователям легко ориентироваться в структуре сайта. Такое наименование способствует логичному распределению контента и повышает общее информационное качество ресурса.

С учётом многоуровневой структуры сайта была реализована навигационная карта, обеспечивающая визуализацию текущего положения пользователя и облегчающая переход к нужным разделам. Это особенно актуально при большом объёме размещённой информации.

Таким образом, разработанная система навигации сочетает в себе простоту, логичность и функциональность, что позволяет обеспечить удобство как для новых посетителей, так и для постоянных пользователей веб-ресурса. Подобный подход рассматривается как одно из ключевых условий успешного взаимодействия с сайтом.

Логическая структура веб-сайта показана на рисунке 2.1



Рисунок 2.1 - Логическая структура разрабатываемого веб-сайта

## Проектирование структуры каталогов

На этапе проектирования физической структуры сайта особое внимание уделяется организации файлов и каталогов проекта. Правильно выстроенная и логически структурированная система хранения ресурсов значительно упрощает процесс разработки, облегчает поддержку и модернизацию веб-сайта, а также позволяет сократить время на внесение изменений и поиск нужных компонентов. Несмотря на то, что конечный пользователь не взаимодействует напрямую с внутренним устройством проекта, грамотная физическая структура является основой стабильной и масштабируемой системы.

Организация файловой системы веб-проекта начинается с распределения всех элементов по тематическим папкам, каждая из которых несёт строго определённую функциональную нагрузку. Такой подход позволяет избежать дублирования информации, снизить вероятность ошибок, а также упростить работу как для одного разработчика, так и для команды. Кроме того, наличие чёткой структуры положительно сказывается на скорости ориентации в проекте при его дальнейшем расширении или доработке.

Особое внимание при проектировании структуры уделяется логическому разграничению файлов по категориям: стили, медиафайлы и HTML-страницы. Также в корневой директории размещаются ключевые файлы проекта, такие как главный HTML-документ (index.html), что соответствует общепринятым стандартам веб-разработки.

Таким образом, создание продуманной структуры каталогов является важным этапом проектирования веб-сайта. Она обеспечивает порядок в проекте, упрощает процессы сопровождения и расширения ресурса, а также способствует более высокой эффективности разработки в целом.

├── .gitignore

├── index.html

├── README.md

├── css/

│ ├── about.css

│ ├── adaptive.css

│ ├── main.css

│ ├── show-title.css

│ └── sidebar.css

│ └── titles.css

├── media/

│ ├── background.png

│ ├── contentwrap-background.png

│ ├── footer-background.png

│ ├── logo.png

│ ├── nav-background.png

│ ├── nav-background-hover.png

│ ├── news-poster.jpg

│ ├── pagenav-wrap.png

│ └── wrapper-background.png

│

│ ├── articles/

│ │ ├── frame2\_art1.webp

│ │ ├── frame2\_art2.jpg

│ │ ├── frame2\_art3.webp

│ │ ├── frame3\_art1.jpg

│ │ ├── frame\_art1.webp

│ │ ├── frame\_art2.jpg

│ │ ├── frame\_art3.webp

│ │ ├── poster\_art1.webp

│ │ ├── poster\_art2.webp

│ │ └── poster\_art3.webp

│

│ ├── icon/

│ │ ├── bullet.png

│ │ ├── date.png

│ │ ├── edit.png

│ │ ├── rss.png

│ │ ├── search.gif

│ │ ├── twitter.png

│ │ └── user.png

│

│ └── poster/

│ ├── Kusuriya\_no\_Hitorigoto.webp

│ ├── Squad\_games.webp

│ └── Thunderbolts.webp

├── pages/

│ ├── about.html

│ ├── movies.html

│ ├── serials.html

│ ├── titles.html

│

│ ├── archive/

│ │ ├── december\_2025.html

│ │ ├── march\_2025.html

│ │ └── may\_2025.html

│

│ ├── article/

│ │ ├── art1.html

│ │ ├── art2.html

│ │ └── art3.html

│

│ └── titles/

│ ├── Kusuriya\_no\_Hitorigoto.html

│ ├── Squad\_games.html

│ └── Thunderbolts.html

Рисунок 3.1 – Структура каталога сайта

В корневом каталоге сайта (Рисунок 3.1) находятся основные папки, каждая из которых отвечает за отдельную категорию ресурсов:

**/page/** — содержит в себе все дополнительные HTML страницы, такие как архивные, описывающие тайтлы на сайте (информационная страница о фильме/сериале) и статьи.

**/page/article/** - статьи, опубликованные на сайте. Статьи имеют различную тематику и относятся как к сериалам, так и к фильмам.

**/page/titles/** - страницы описания тайтлов. Тайтлы, которые были упомянуты в статьях, или просто находятся в местном справочнике располагаются в данной директории.

**/page/archive/** - статьи заархивированные, или отсортированные по месяцам и годам на сайте.

**/media/ ­­­­­­**- медиафайлы, размещённые на сайте. В основном там содержатся изображения для интерфейса сайта и постеры/кадры для статьи и тайтлов.

**/media/articles** – изображения для статьи. Кадры из сериала/фильма и обложки.

**/media/icon** – директория, содержащее иконки для элементов сайта.

**/media/poster** – постеры для описания тайтлов.

**/css/** — каталог с файлами таблиц стилей.

Также в корне сайта находится файл index.html, который является главной страницей ресурса. Это соответствует принятой структуре размещения веб-проектов и позволяет браузеру по умолчанию загружать именно его при обращении к сайту.

Выбирая имена к папкам, необходимо было определить для каждой директории роль внутри проекта. Каждый файл назван так, чтобы при просмотре структуры проекта сразу было ясно, к какому содержимому он относится. Такой подход особенно важен при масштабировании проекта или при работе в команде.

Таким образом, выбранная структура каталогов обеспечивает как порядок и чистоту в проекте, так и простоту его масштабирования и сопровождения в будущем.

## Создание прототипа.

На этапе разработки прототипа веб-сайта была реализована блочная структура страниц с применением каскадных таблиц стилей (CSS), что соответствует актуальным тенденциям в сфере веб-разработки. Такой подход обеспечивает гибкость построения макета, упрощает реализацию адаптивного дизайна и позволяет корректно отображать содержимое на устройствах с различными разрешениями экрана.

Проектирование основы сайта стало важным этапом, на котором были выделены ключевые структурные элементы страниц: шапка, основной контентный блок, боковая панель и подвал. Это позволило заранее определить логическое размещение элементов, сформировать интуитивно понятную навигацию и обеспечить удобное восприятие информации. Кроме того, заранее продуманная структура страниц способствовала организации логики переходов и упорядочиванию отображаемого контента.

С целью обеспечения адаптивности интерфейса были подготовлены два макета, ориентированные на разные типы устройств. Прототип главной страницы для настольных компьютеров и планшетов (см. Приложение А) включает все элементы структуры: верхнюю полосу, отступы, заголовок сайта, постер, основной контент, боковую панель и подвал с контактной информацией. Данный вариант ориентирован на экраны с большой шириной и обеспечивает полный доступ ко всем функциям сайта.

Макет для мобильных устройств (см. Приложение Б) представляет собой наиболее компактный вариант. Боковая панель перемещена в нижнюю часть страницы, а элементы меню размещены вертикально, обеспечивая удобное взаимодействие с сайтом на небольших экранах.

Разработка данных прототипов позволила на ранней стадии выявить особенности поведения интерфейса при изменении размеров экрана, а также определить трудности в адаптивной верстке. Это стало важной основой для дальнейшей реализации сайта, направленной на достижение корректного отображения и высокой удобства использования на всех типах устройств.

## Разработка и наполнение HTML-страниц

В процессе разработки сайта особое внимание уделялось оформлению навигации с учётом принципов простоты, удобства и читабельности. Целью было сделать так, чтобы пользователь мог быстро и легко перейти на любую страницу веб-сайта, включая главную.

Для этого на каждой странице сайта реализована верхняя панель навигации с подписями страниц, которые связаны ссылкой на каждый раздел (в том числе и index.html – главная страница). Такое решение позволяет пользователю, попавшему на любую страницу, даже напрямую из поисковой системы, понимать, что он находится в рамках единого сайта, и быстро перейти к другим разделам.

Навигационное меню оформлено в виде горизонтального списка ссылок на основные разделы сайта: Главная, Кино, Сериалы, Тайтлы. Это обеспечивает простоту ориентации и удобство перехода по сайту.

Важным элементом дизайна является адаптивность навигации и структуры страниц. Используются медиазапросы, которые обеспечивают перестройку макета под разные размеры экранов — от больших десктопов до мобильных устройств. Например, при узких экранах меню меняет горизонтальное расположение на вертикальное, а ключевые элементы — боковая панель —перемещается вниз для сохранения удобочитаемости.

**Организация структуры файлов и ссылок**

Материалы сайта разбиты по файлам и папкам согласно логической структуре:

* Главная страница — index.html
* Раздел «Фильмы» — movies.html
* Раздел «Сериалы» — serials.html
* Статьи — отдельные страницы в папке /pages/article/
* Тайтлы — titles.html и в /pages/titles/
* Информация о проекте — about.html

Все ссылки внутри сайта реализованы относительными, что позволяет корректно функционировать при переносе сайта на разные серверы.

**Наполнение страниц**

Наполнение сайта осуществляется на основе созданных HTML-шаблонов, где структура страницы повторяется, а содержимое меняется в зависимости от тематики. В структуре каждой страницы используются:

* Заголовки (<h1>, <h2>, <h3>) для логической иерархии и SEO-оптимизации;
* Параграфы (<p>) для текстового контента;
* Списки (<ul>) для перечислений и улучшения восприятия информации;
* Семантические элементы (<header>, <article>, <nav>, <footer>) для более чёткого описания структуры страницы.

Например, на странице с описанием тайтла используются теги <section>, в которых внутри есть заголовки и таблицы с описанием тайтлов.

**Ключевые метаданные первой страницы:**

* Ключевые слова: кинопоиск, кино, фильмы, сериалы, проект, аниме, доннту, новости, статьи
* Используемый стандарт: HTML5
* Язык сайта: русский (lang="ru")
* Кодировка: UTF-8

Такой подход к созданию сайта обеспечивает удобство использования для посетителей, правильное индексирование поисковыми системами и поддержку адаптивного дизайна, что особенно важно в современном мире с многообразием устройств доступа к Интернету.

## Разработка стилей для реализации интерфейса Web-сайта

В процессе разработки веб-страницы использовалось большое количество CSS-стилей по разным файлам, чтобы добиться как чёткой структуры размещения элементов, так и эстетически привлекательного оформления. Стили применялись для различных целей:

* для управления положением элементов на странице с помощью CSS Grid и Flexbox;
* для оформления текста — отступов, выравнивания, начертания;
* для визуального представления изображений и блоков, включая рамки, скругления и цвета;

Всё это в совокупности позволило создать адаптивный, функциональный и визуально сбалансированный макет, соответствующий структуре, заданной шаблоном. Ниже подробно рассматриваются основные блоки страницы и применённые к ним стили.

Для создания структуры страницы применена современная технология CSS Grid, которая позволяет гибко управлять размещением блоков в виде сетки. Основной контейнер — это элемент, которому задаются параметры деления через следующие свойства:

.content {

    display: grid;

    grid-template-columns: [news] 60% [sidebar] 40%;

}

В этом примере:

* grid-template-columns задаёт две колонки разной ширины в пропорциях;

**Гибкая внутренняя компоновка через Flexbox**

Для выравнивания элементов внутри блоков используется Flexbox. Это позволяет удобно управлять направлением, выравниванием и переносом вложенных элементов.

Пример для карточек в списке тайтлов:

.list-titles {

  margin: 1em;

  display: flex;

  flex-wrap: wrap;

  flex-direction: column;

  gap: 1em;

}

Пример вертикального центрирования и распределения в блоке с описанием:

.left-column {

  display: flex;

  flex-direction: column;

  align-items: center;

  width: 400px;

}

Свойства align-items: center и flex-direction: column позволяют выровнять элементы по вертикали и распределить пространство между ними.

**Текстовое оформление**

Для работы с текстом используются базовые свойства CSS. Например, чтобы задать шрифт по умолчанию для всей страницы, используется универсальный селектор:

\* {     font-family: Arial, Helvetica, sans-serif; }

Отступ первой строки абзаца задаётся с помощью text-indent, что повышает читаемость:

.description-title {

  font-size: 18px;

  margin: 1em;

  text-indent: 1.25cm;

  line-height: .8cm;

  text-align: justify;

}

**Оформление вспомогательных блоков**

Для виджетов в боковой панели применяем следующие стили:

.sidebar .widget h3 {

    background-color: #1c1f23; /\* тёмный основной фон \*/

    background-image: repeating-linear-gradient(

        135deg,             /\* угол диагонали \*/

        rgba(255, 255, 255, 0.168),

        rgba(255, 255, 255, 0.168) 1px,

        transparent 1px,

        transparent 5px

    );

    padding: 0.5em 1em;

    color: white;

    font-weight: bold;

    text-transform: uppercase;

}

Похожие стили применяются к некоторым компонентам боковой панели что придаёт элементам однородный и современный вид:

.sidebar .tags a {

    display: inline-block;

    background-color: #2a2d31;

    color: #bbb;

    padding: 3px 8px;

    margin: 2px;

    font-size: 13px;

    border-radius: 3px;

    text-decoration: none;

    transition: background-color 0.3s ease;

}

**Оформление навигации**

Навигационное меню оформлено с использованием Flexbox, что обеспечивает горизонтальное выравнивание пунктов:

.line-navigation {

    background-color: #a42405;

    background-size: cover;

    display: flex;

    justify-content: flex-start;

    align-items: center;

}

Пункты меню оформляются как прямоугольные блоки с фоном:

.line-navigation a {

    display: flex;

    align-items: center;

    justify-content: center;

    height: 30pt;

    padding: 0 1em;

    margin: auto 1em;

    color: white;

    text-decoration: none;

    font-weight: bold;

    font-size: 10pt;

}

Такое оформление создаёт визуально чёткое меню, где каждый пункт отделён промежутком.

**Контентная область**

Блок .news на главной странице организует основные статьи. Он настроен на вертикальное направление (flex-direction: column) с центрированием:

.news {

    display: flex;

    flex-direction: column;

    margin: 1.2em;

}

**Оформление боковой панели**

Боковая панель (.sidebar) расположена в правой части сетки. Кнопка «подписаться на RSS» оформлена следующим образом:

.widget-button {

    display: flex;

    background-color: #272a31;

    border: #666666 solid 1px;

    padding: .8em;

    font-size: 16px;

    margin: 1em 0;

}

Списки разделов имеют индивидуальные стили:

.sidebar ul {

    list-style: none;

    padding-left: .5em;

}

.sidebar ul li {

    list-style-image: url("/media/icon/bullet.png");

    margin-bottom: 0.5em;

}

**Таблицы**

Для таблиц применяются стили, обеспечивающие слияние границ и оформление углов:

.info-table th {

    color: #777;

    text-align: left;

    vertical-align: top;

    padding: 0.5em 1em 0.5em 0;

    width: 150px;

    font-weight: normal;

}

.description-title {

    font-size: 18px;

    margin: 1em;

    text-indent: 1.25cm;

    line-height: .8cm;

    text-align: justify;

}

**Подвал сайта (footer)**

Подвал страницы выглядит как карточка под всем контентом, выровненный по горизонтали и располагается по центру:

footer {

    background-image: url("/media/footer-background.png");

    background-size: contain;

    margin: 1em 0;

    width: 100%;

}

.text-footer {

    display: flex;

    color: #fff;

    padding: auto 100%;

    height: 50pt;

    text-align: center;

    justify-content: center;

    a {

        color: #fff;

    }

}

Контент подвала разделён на две части: блок с контактами и авторским правом. Всё содержимое окрашено в белый цвет (color: #fff).

**Дополнительные элементы**

Списки тайтлов организованы через flex с направлением – column:

.list-titles {

    margin: 1em;

    display: flex;

    flex-wrap: wrap;

    flex-direction: column;

    gap: 1em;

}

Блок тайтлов также организован через flex и внутри имеет два элемента – постер и информацию о тайтле (название, год, жанр и краткое описание):

.titles {

    display: flex;

    width: 100%;

    height: auto;

    line-height: 25px;

}

.poster-title img {

    width: 10em;

    height: 13em;

    text-align: center;

}

## Разработка адаптивного шаблона сайта

Для современного веб-сайта очень важно, чтобы его интерфейс корректно отображался на разных устройствах — от больших экранов ПК до маленьких экранов телефонов. Для этого применяют адаптивный дизайн: структура, размеры и стили элементов меняются в зависимости от ширины экрана пользователя. Основной инструмент для этого — CSS-медиа-запросы (@media) [3], которые позволяют подгружать и применять разные стили для разных диапазонов разрешений.

Это позволяет, например, на большом экране показывать меню горизонтально, а на маленьком — вертикально; менять размеры картинок и текста для удобства чтения; упрощать или усложнять интерфейс в зависимости от устройства.

* **Размер экран c 1280px**

Для экранов шириной до 1280 пикс. сетка остаётся трёх колоночной, но меняются пропорции некоторых элементов. Это позволяет сохранить структуру, но лучше вписать контент в менее широкое пространство.

@media screen and (min-width: 1280px) and (orientation: landscape) {

    .main-container {

        margin: auto 25em;

    }

    .sidebar {

        grid-template-columns: [col1] 50% [col2] 50%;

        column-gap: .5em;

    }

}

* **Экраны от 980px до 1280px — планшеты**

Для планшетов с экранами до 769px структура сильно не изменяется, но отступы от экрана становятся меньше:

@media screen and (min-width: 980px) and (max-width: 1280px) {

    .main-container {

        margin: auto 5em;

    }

}

* **Экран до 426px — смартфоны**

На узких экранах смартфонов используется одномерная сетка, все блоки идут друг под другом:

@media screen and (max-width: 768px) {

    .main-container { margin: 0em; }

    header { height: 70pt; }

    .search-bar { margin: auto .2em; }

    .line-navigation { flex-direction: column; }

    .line-navigation a {

        width: 100%;

        height: 25pt;

    }

.text-footer { height: 100pt; }

    .main-nav {

        display: flex;

        flex-direction: column;

        height: 100%;

    }

    .content {

        grid-template-columns: 100%;

    }

    .post-image img {

        height: 100%;

    }

}

## Реализация сайта

Процесс наполнения веб-сайта информационным содержанием проводился поэтапно и включал сбор, структурирование и размещение текстовых и графических материалов. Основное внимание уделялось не только качеству представленного контента, но и его соответствию тематике сайта — кино и сериалы. На начальном этапе был сформирован набор необходимых материалов: текстовые описания, иллюстрации, постеры, а также информационные блоки для разделов сайта.

Использование заголовков и параграфов для логической структуры текста подробно рассматривается в [2]. Тексты проходили предварительную обработку: информация разделялась на смысловые части, формировались заголовки различных уровней, параграфы и списки. Такой подход позволял сделать страницы логичными и удобными для восприятия, что особенно важно при большом объёме данных. Пользователю не приходилось тратить много времени на поиск нужной информации, поскольку структура способствовала её быстрому нахождению.

Качественная проверка текстов являлась обязательным этапом. Все материалы обрабатывались в текстовых редакторах с автоматической проверкой орфографии и пунктуации. Это сделало информацию более доступной даже для пользователей, не обладающих техническими знаниями.

Дополнительно проводилась адаптация текстов под веб-формат: использовались короткие абзацы, выделения ключевых фраз, избегались перегруженные конструкции. Такой подход повышает читабельность и снижает утомляемость при просмотре. Также соблюдалась семантическая структура HTML-документов, что положительно влияет на SEO-оптимизацию сайта — важный фактор для повышения его видимости в поисковых системах.

В рамках обучения использовались практические рекомендации по HTML из [4]. Контент подбирался с учётом визуального восприятия: к каждому разделу прилагались тематические изображения и постеры, усиливающие информативность и создающие привлекательный внешний вид страниц.

## Тестирование

В процессе тестирования и доработки веб-сайта основное внимание было сосредоточено на выявлении и устранении возможных ошибок, возникающих как в результате человеческого фактора, так и на этапе интеграции графических и функциональных компонентов. Были тщательно проанализированы как текстовое наполнение, так и технические аспекты верстки. Все обнаруженные недочёты — от орфографических ошибок до несоответствий в разметке и стилях — последовательно устранялись.

Одним из ключевых направлений корректировки стала визуальная доработка интерфейса: выравнивание элементов, корректировка отступов, подбор цветовой палитры текста и фона. Это обеспечило единообразное и гармоничное оформление всех страниц сайта, что способствует комфортному восприятию информации. Стандартизированный подход к отображению заголовков, кнопок, ссылок и навигационных меню позволил добиться целостности дизайна и улучшения пользовательского опыта.

Одновременно с визуальными правками проводилось тестирование работоспособности сайта в различных браузерах — Google Chrome, Mozilla Firefox, Microsoft Edge, Opera и Yandex Browser. Также были задействованы устройства на различных операционных системах, включая Windows и Android. Это помогло выявить различия в отображении и работе отдельных элементов, обусловленные особенностями рендеринга и техническими ограничениями различных платформ.

Особое внимание уделялось скорости загрузки страниц. В случаях, когда производительность сайта оказывалась ниже ожидаемой, предпринимались меры по оптимизации ресурсов: сжатие изображений без потери качества, упрощение CSS-кода и улучшение структуры HTML-документов. Эти шаги позволили снизить нагрузку на браузер и сервер, обеспечив быструю загрузку и стабильную работу сайта.

Для проверки корректности HTML-кода применялся официальный валидатор W3C (validator.w3.org, рисунок 9.1). Результаты показали полное соответствие стандартам: в отчёте отсутствовали ошибки и предупреждения, а были лишь информационные замечания (Info), не влияющие на работоспособность сайта.

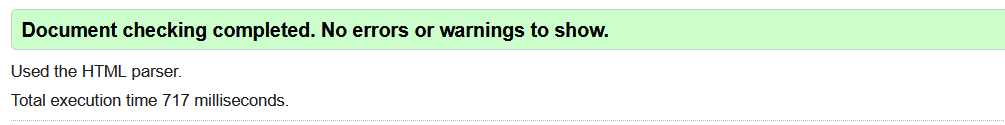


Рисунок 9.1 – Общий результат проверки страниц

## Размещение сайта и выбор средств для повышения его рейтинга

После завершения всех этапов проектирования, разработки, верстки и тестирования сайта DeVin был осуществлён финальный этап — его размещение в сети Интернет. Для публикации проекта был выбран вариант с использованием VPS сервера и программного обеспечения nginx [5], предоставляющий возможность гибкой настройки WEB-сервера через терминал посредством использования утилит на подобии ssh.

Размещение производилось на платной основе, но сервер уже эксплуатировался с другими службами, поэтому целесообразность использования VPS для хостинга проекта была обусловлена экономией на другие хостинги, что особенно удобно для учебных и демонстрационных проектов.

Все файлы проекта — HTML-страницы, CSS-стили, изображения и другие ресурсы — были загружены на сервер через клонирование репозитория с Github. Размещение файлов осуществлялось в директории /usr/share/nginx/html.

После завершения загрузки была выполнена проверка работы сайта DeVin: открытие всех страниц, корректность ссылок, отображение графики и оформление в различных браузерах и на разных устройствах. Также была повторно проверена кроссплатформенность и валидность кода на внешнем сервере.

Дополнительно было необходимо дать WEB-серверу SSL-сертификаты, чтобы обезопасить пользователей от возможных кибер-атак. Получить сертификат от Let’s Encrypt и настроить WEB-сервер на работу в https автоматически возможно через утилиту certbot [5].

Таким образом, сайт DeVin стал доступен онлайн и готов к дальнейшему обновлению и развитию без дополнительных затрат на этапе размещения.

Доступ к сайту по ссылке: https://devinkursach-billvlad.ru/

**Выбор средств для повышения его рейтинга**

С целью продвижения сайта **DeVin** и увеличения его присутствия в результатах поисковых систем был реализован ряд мероприятий, направленных на повышение уровня индексации и улучшение позиции ресурса среди аналогичных веб-площадок.

Одним из первых шагов стало внедрение уникальных метатегов на каждую HTML-страницу. Были добавлены теги <meta name="keywords"> с ключевыми словами, точно отражающими содержание соответствующей страницы, а также теги <meta name="description">, содержащие краткое описание, отображающееся в результатах поисковой выдачи. Дополнительно каждой странице был присвоен уникальный заголовок в теге <title>, соответствующий тематике раздела. Это позволяет улучшить восприятие ресурса поисковыми алгоритмами и привлечь внимание пользователей к содержимому сайта.

Для ускорения процесса индексации сайт был зарегистрирован в поисковых системах **Яндекс**. Подтверждение прав собственности на сайт осуществлялось путём добавления в исходные файлы сайта страницы yandex\_xxx.html:

[root@localhost html]# cat yandex\_639d7df12f3c29cc.html

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

</head>

<body>Verification: 639d7df12f3c29cc</body>

Эти данные необходимы для подтверждения доступа к аналитическим инструментам поисковых систем.

Для улучшения внутренней навигации и оптимального распределения ссылочного веса на сайте реализована система внутренних гиперссылок. Благодаря этому пользователи могут удобно перемещаться между разделами, а поисковые роботы — эффективно индексировать контент.

Также был создан файл **sitemap.xml**, содержащий перечень всех страниц сайта с указанием приоритетности и предполагаемой частоты обновления. Генерация карты сайта производилась с помощью специализированного онлайн-сервиса. Это существенно облегчает работу поисковых систем по выявлению нового или обновлённого контента.

Дополнительно был настроен файл **robots.txt**, содержащий инструкции для поисковых роботов о том, какие страницы подлежат индексации, а какие следует исключить. Такой подход помогает избежать дублирования информации в поисковой выдаче и повышает релевантность проиндексированных данных.

Для мониторинга активности пользователей на главную страницу был добавлен счётчик посещений. Это позволяет отслеживать посещаемость ресурса, анализировать поведение аудитории и принимать обоснованные решения для дальнейшего улучшения проекта.

Таким образом, комплекс выполненных мероприятий обеспечил техническую готовность сайта к эффективному продвижению в сети, что в перспективе должно способствовать росту его популярности и увеличению трафика.

# ЗАКЛЮЧЕНИЕ

В рамках выполнения курсовой работы была реализована полноценная разработка веб-сайта DeVin, посвящённого тематике сериалов и кинофильмов. Проект охватил все ключевые этапы создания сайта: от начального проектирования структуры и дизайна до технической реализации, наполнения содержимым, тестирования и размещения на сервере.

Особое внимание в процессе разработки уделялось удобству навигации, визуальной читаемости интерфейса, корректной работе в разных браузерах, а также адаптивности отображения на различных устройствах. Контент сайта был тщательно структурирован, оформлен с использованием стандартных HTML-элементов — таких как заголовки, списки, параграфы и внутренние ссылки — и приведён к современным требованиям веб-верстки. Также была проведена текстовая и техническая оптимизация с точки зрения грамматики, орфографии и чистоты HTML-кода.

Настройка веб-сервера Nginx на VPS выполнялась согласно официальному руководству [5]. Готовый сайт был размещён на собственном VPS-хостинге. Для улучшения видимости в поисковых системах выполнена базовая SEO-оптимизация: добавлены метатеги keywords и description, установлены валидационные метки от Яндекса и Google, создан и подключён файл sitemap.xml, а также настроен файл robots.txt для управления индексацией страниц.

Процесс разработки способствовал углублению теоретических знаний и приобретению практических навыков в области верстки, проектирования структуры сайта, настройки внутренней навигации и работы с метаинформацией. Кроме того, был получен первый опыт по продвижению веб-ресурсов.

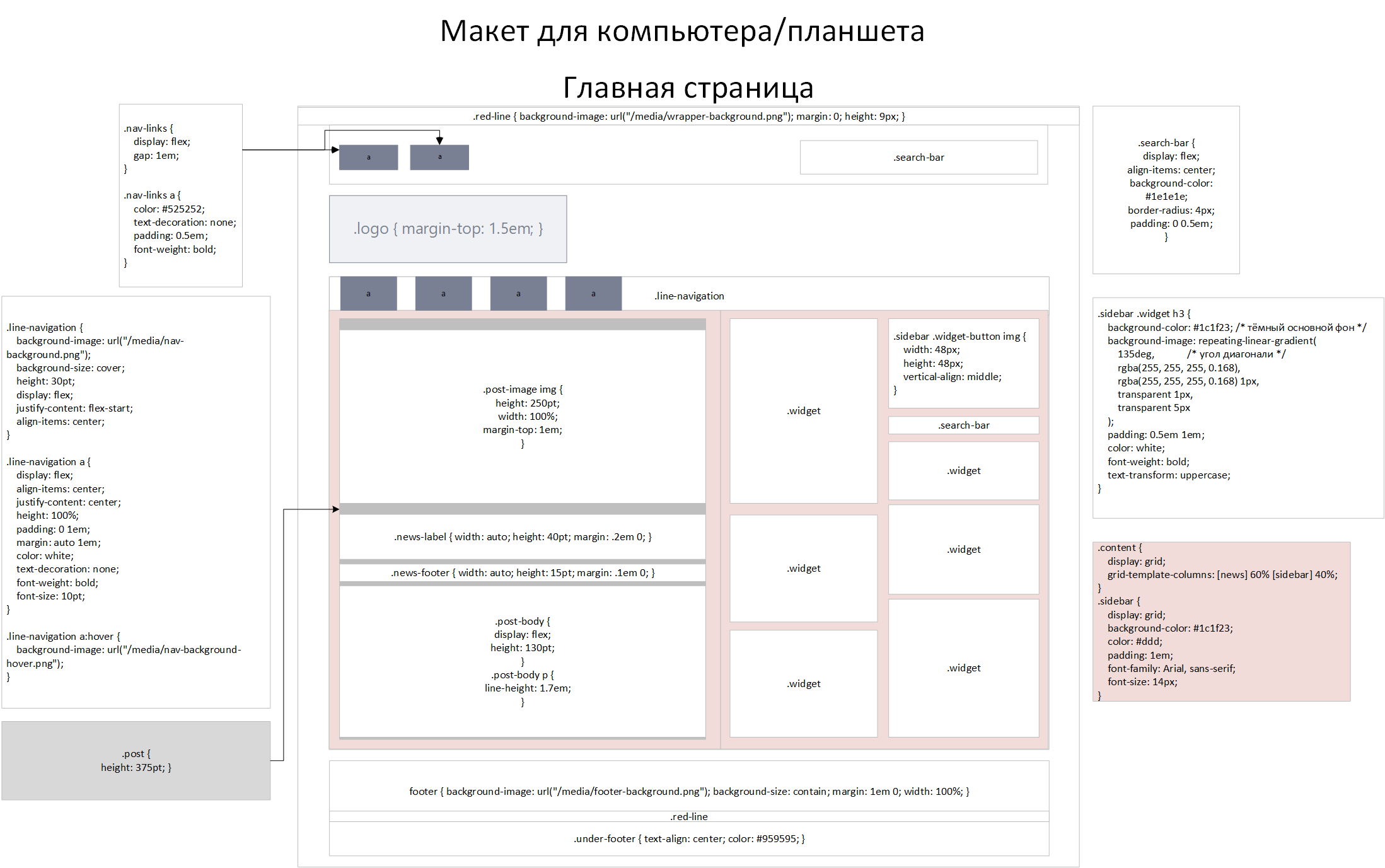
Итогом работы стал полнофункциональный веб-сайт, готовый к использованию и дальнейшему развитию. В перспективе планируется расширение функционала: добавление формы обратной связи, реализации поиска по сайту, интеграции с базой данных, внедрения динамических компонентов и регулярное обновление контента.

# СПИСОК ЛИТЕРАТУРЫ

1. Дронов, В. HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов / Владимир Дронов. - М.: БХВ-Петербург, 2021. - 561 c.
2. Дакетт, Д. HTML и CSS. Разработка и создание веб-сайтов / Д. Дакетт. — М.: Эксмо, 2021. — 716 c.
3. Руководство по HTML5 и CSS3 [Электронный ресурс]. — URL: <https://metanit.com/web/html5/> (дата обращения: 10.05.2025)
4. Учебник HTML и CSS для новичков [Электронный ресурс]. — URL: <https://code.mu/ru/markup/book/prime/> (дата обращения: 12.05.2025)
5. Настройка Nginx на Rocky Linux – URL: https://docs.rockylinux.org/guides/web/nginx-mainline/

# ПРИЛОЖЕНИЕ A.

Макет главной страницы для компьютеров и планшетов



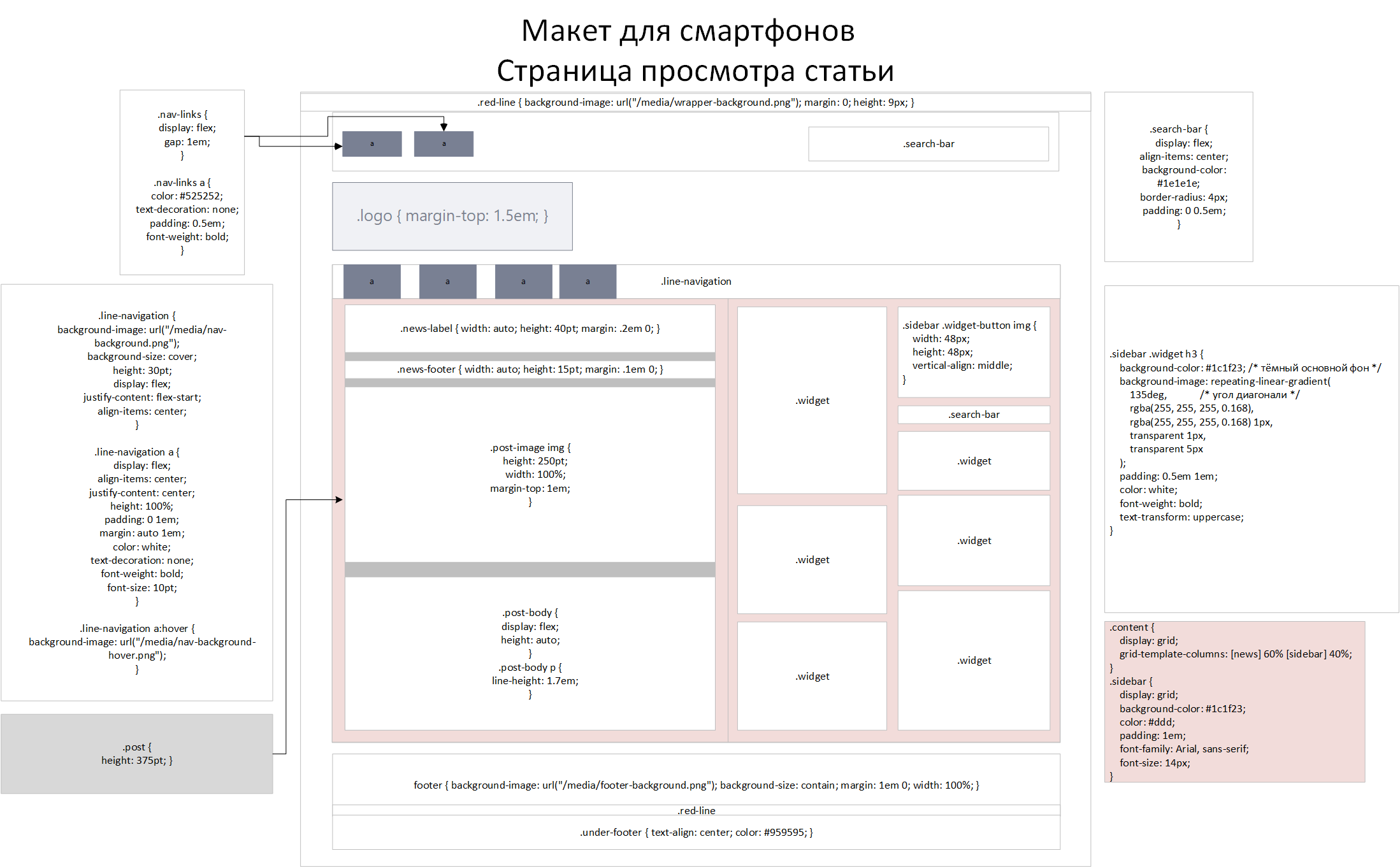
# ПРИЛОЖЕНИЕ Б.

Макет главной страницы для смартфонов



# ПРИЛОЖЕНИЕ В.

Макет страницы просмотра статьей для компьютера/планшетов



# ПРИЛОЖЕНИЕ Г

Макет страницы просмотра статьей для смартфонов



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ФИО студ. | Шевченко | | Владислав | Евгеньевич |
|  | |  |  |  |
| Вид работы | | Макс. балл | Набр. балл | Комментарий |
| Срок сдачи курсовой работы(проекта) | | 5 |  |  |
| Оформление пояснительной записки | | 15 - 20 |  |  |
| Верстка страниц:  Табличная  Блочная | | 20 - 35 |  |  |
| Объем контента (2-4 страницы) | | 10 - 15 |  |  |
| Проверка на валидность верстки | | 5 |  |  |
| Использование JavaScript | | 15 |  |  |
| Кроссплатформенность (IE8/9, Firefox, Chrome) | | 5 |  |  |
| Ключевые слова и метатеги для поиска | | 5 |  |  |
| Размещение сайта на хостинге | | 10 |  |  |
| Регистрация в поисковых системах | | 5 |  |  |
| Всего: | | 120 |  |  |