МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ   
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных технологий

Кафедра Информационных систем и технологий

Специальность 6-05-0612-01 "Программная инженерия"

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА КУРСОВОГО ПРОЕКТА**

по дисциплине «Компьютерные языки разметки»

Тема «Веб-сайт «журнального издания»

**Исполнитель**

студент 1 курса 7 группы \_\_\_\_\_\_\_\_\_\_\_\_\_\_ А. Д. Шлёг

подпись, дата

**Руководитель**

старший преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е. В. Барковский

должность, учен. степень, ученое звание подпись, дата

Допущен(а) к защите \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

дата, подпись

Курсовой проект защищен с оценкой

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ Е. В. Барковский

подпись дата инициалы и фамилия

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования   
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий   
Кафедра информационных систем и технологий

Утверждаю

И.о. заведующего кафедрой

\_\_\_\_\_\_\_\_\_\_\_ Е.А. Блинова

подпись инициалы и фамилия

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2025 г.

**ЗАДАНИЕ**

**на курсовое проектирование**

по дисциплине«Компьютерные языки разметки»

Курс 1 Группа: 7 Специальность: 6-05-0612-01 Программная инженерия

Студент: Шлёг Александр Дмитриевич

**1. Тема:** Веб-сайтСети кинотеатров **«**КиноГалактика»

**2. Срок выполнения курсового проекта** : с 10 февраля 2025 г. по 05 мая 2025 г.

**3. Технические требования:**

3.1 Прототип веб-сайта должен быть разработан с использованием графических редакторов Figma/Adobe XD/Sketch.

3.2 Для хранения данных должен быть использован XML-формат.

3.3 Разметка содержания сайта должна быть выполнена с применением HTML5 и XML.

3.4 Для описания внешнего вида веб-страниц использовать SCSS и CSS3.

3.5 Веб-сайт должен содержать:

– семантические теги HTML5;

– графические элементы в форме SVG;

– несколько веб-страниц;

– JavaScript для управления элементами DOM.

3.6 Верстка сайта должна быть адаптивной и кроссбраузерной;

3.7 Для тестирования использовать The W3C Markup Validation Service/Git Super Linter

3.8 Проект и пояснения к проекту должны быть размещены на GitHub.

**4. Содержание пояснительной записки:**

1. Титульный лист

2. Задание на курсовое проектирование;

3. Введение;

4. Постановка задачи;

5. Проектирование веб-сайта

6. Реализация структуры веб-сайта

7. Тестирование веб-сайта

8. Заключение

9. Список использованных источников

10. Приложения (полный исходный текст программы разработанного приложения с подробными комментариями)

**5. Форма представления на GitHub выполненного курсового проекта:**

– Теоретическая часть пояснительной записки курсового проекта должна быть представлена в формате MS Word.

– Оформление записки должно быть согласно правилам.

– Листинги представляются в приложении.

#### Календарный план

|  |  |  |
| --- | --- | --- |
| № п/п | Наименование этапов курсового проекта | Срок выполнения этапов проекта |
| 1 | Задание на курсовое проектирование. Титульный лист | 10.02.2025 |
| 2 | Введение. Постановка задачи | 03.03-09.03.2025 |
| 3 | Проектирование веб-сайта | 10.03-23.03.2025 |
| 4 | Реализация структуры веб-сайта на HTML5 и внешнего оформления на SCSS и СSS3 | 24.03–16.04.2025 |
| 5 | Тестирование веб-сайта | 17.04–24.04.2025 |
| 6 | Заключение | 25.04.2025 |
| 7 | Список использованных литературных источников. Приложения | 26.04.2025 |
| 6 | Подготовка и оформление пояснительной записки курсового проекта | 28.04–04.05.2025 |
| 7 | Сдача на допуск к защите курсового проекта | 05.05.2025 |
| 8 | Защита курсового проекта | 19.05-31.05.2025 |

**5. Дата выдачи задания** «10» февраля 2025 г

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.В. Комкова

(подпись)

Задание принял к исполнению \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата и подпись студента)

Содержание

[Введение 2](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229866)

[1 Постановка задачи 4](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229867)

[1.1 Обзор аналогичных решений 4](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229868)

[1.2 Техническое задание 5](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229869)

[1.3 Выбор средств реализации программного продукта 6](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229870)

[1.4 Вывод 6](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229871)

[2 Проектирование страниц веб-сайта 7](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229872)

[2.1 Выбор способа верстки 7](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229873)

[2.2 Выбор стилевого оформления 7](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229874)

[2.3 Выбор шрифтового оформления 7](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229875)

[2.4 Разработка пользовательских элементов 7](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229876)

[2.5 Разработка спецэффектов 8](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229877)

[2.6 Вывод 8](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229878)

[3 Реализация структуры веб-сайта 9](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229879)

[3.1 Структура HTML-документа 9](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229880)

[3.2 Добавление таблиц стилей SCSS и CSS 9](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229881)

[3.3 Использование JSON, SVG, XML 10](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229882)

[3.4 Управление элементами DOM 11](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229883)

[3.5 Вывод 11](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229884)

[4 Тестирование веб-сайта 12](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229885)

[4.1 Адаптивный дизайн веб-сайта 12](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229886)

[4.2 Кроссбраузерность веб-сайта 13](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229887)

[4.4 Вывод 15](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229888)

[Заключение 16](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229889)

[Приложение А 18](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229890)

[Прототипы веб-страниц 18](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229891)

[Приложение Б 22](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229892)

[Макеты страниц 22](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229893)

[Приложение В 26](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229894)

[Листинг HTML-документов 26](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229895)

[Приложение Г 54](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229896)

[Листинг SCSS и CSS 54](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229897)

[Приложение Д 83](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229898)

[Листинг JSON-файлов 83](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229899)

[Приложение Ж 85](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229900)

[Листинг JavaScript 85](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229901)

[Приложение З 88](file:///C:\Users\Александр\Downloads\Telegram%20Desktop\Пояснительная%20записка%20(4).docx#_Toc198229902)

**Введение**

В рамках курсового проекта разработан веб-сайт сети кинотеатров с использованием современных компьютерных языков разметки и стилей: **HTML5**, **SCSS/CSS3**, а также языка программирования **JavaScript**. Основная цель проекта – создание удобного и функционального интерфейса для просмотра афиши кинотеатров, выбора фильмов и покупки билетов онлайн.

**HTML5** использовался для построения семантической структуры веб-страниц, **SCSS/CSS3** – для стилизации и адаптивного дизайна, обеспечивающего корректное отображение на различных устройствах. **JavaScript** добавил интерактивность (например, выбор сеансов и мест в зале), а **JSON** применялся для хранения и обработки данных.

В результате был реализован современный и удобный веб-сайт с адаптивным дизайном, возможностью просмотра афиши, выбора фильма и онлайн-покупки билетов.

### К задачам курсового проекта относится:

1. Разработка дизайна интерфейса с использованием языков стилей (SCSS/CSS3).
2. Создание прототипов веб-страниц на основе HTML5 с акцентом на удобство пользователя.
3. Построение структуры сайта с применением семантической разметки для улучшения доступности.
4. Реализация адаптивного дизайна для корректного отображения на мобильных устройствах, планшетах и ПК.
5. Тестирование кросс-браузерной совместимости и отзывчивости интерфейса.

**1.Постановка задачи**

**1.1Обзор аналогичных решений**

Первым рассматриваемым проектом среди конкурентов является «Bycard». У **сайта** приятное оформлении, но палитра цветов выглядит бедной и недостаточно выразительной.

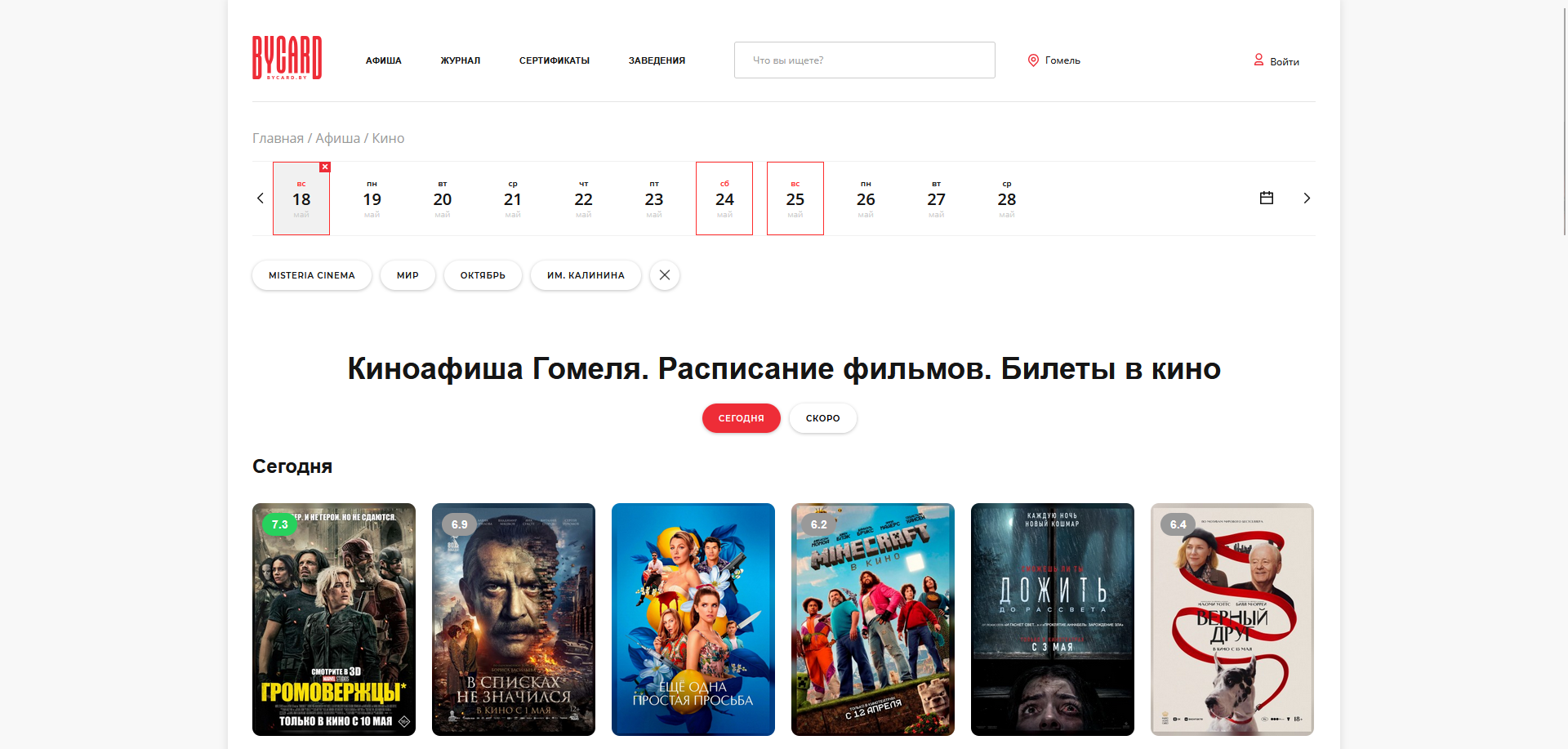


Рисунок 1.1 – Сайт “Bycard”

Футер сайта киноафиши выполнен в лаконичном и удобном стиле, предоставляя пользователям всю необходимую информацию.

В верхней части размещено краткое описание сервиса, подчёркивающее его преимущества — онлайн-покупку билетов, полное расписание сеансов и список кинотеатров. Ниже представлены важные ссылки в виде ярких синих кнопок. Рядом находится поле для ввода электронной почты с кнопкой подписки. Основной блок футера разделён на четыре тематические колонки: «Развлечения», «Заведения», «Важное», а также раздел для бизнеса с доступом в личный кабинет и интеграцией с банками.

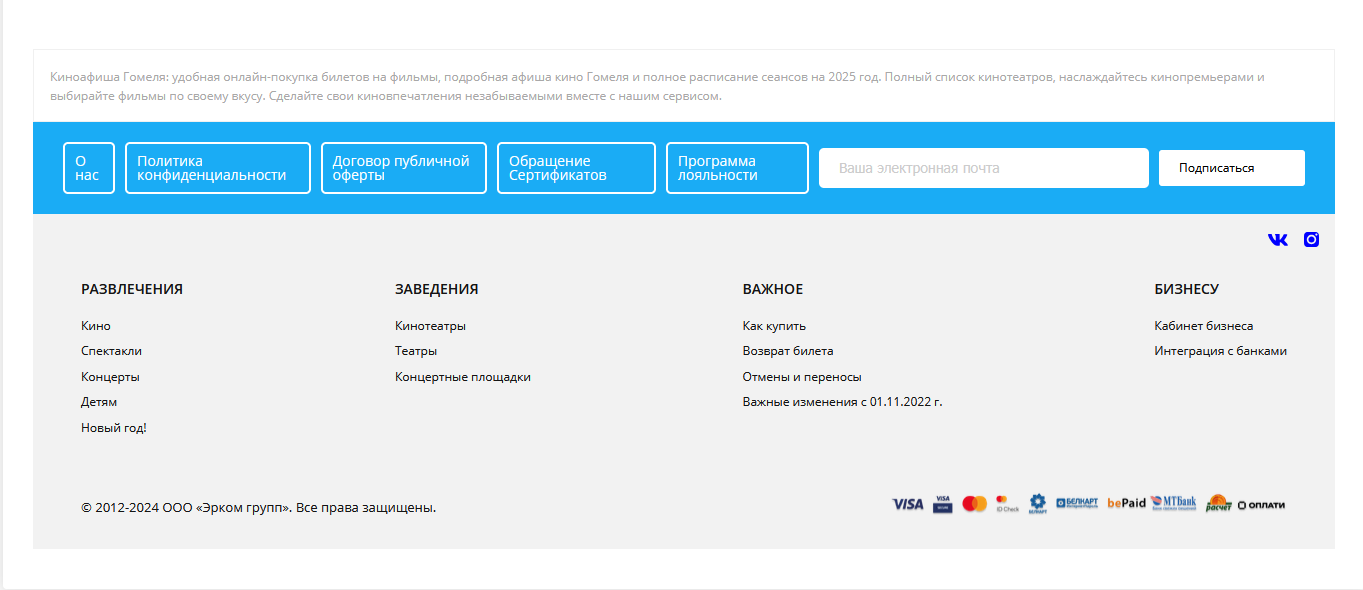


Рисунок 1.2 – Футер Сайта “Bycard”

**1.2. Техническое задание**

**Курсовой проект представляет собой веб-сайт для сети кинотеатров, на котором будет размещена информация о предстоящих премьерах, расписании сеансов, а также реализован функционал онлайн-бронирования билетов. Сайт будет содержать сведения о фильмах: жанр, длительность, возрастные ограничения, описание сюжета, постеры, трейлеры и комментарии пользователей.**

**Цель сайта:** Предоставить пользователю удобный способ получения информации о фильмах и сеансах, а также обеспечить возможность онлайн-покупки билетов в кино.  
Целевая аудитория: широкая – от подростков до пожилых людей, всех, кто интересуется кино и предпочитает бронировать билеты онлайн, не тратя время на поход в кассу. Сайт должен быть доступным, интуитивно понятным и эстетически приятным.

**1.3. Выбор средств реализации программного продукта**

Для реализации программного продукта были выбраны следующие технологии: HTML и CSS/SCSS. Язык разметки HTML используется для создания структуры веб-сайта, а оформление и стилизация реализуются с применением CSS и препроцессора SCSS.  
Для управления элементами DOM и обработки данных используется JavaScript. Данные хранятся в формате JSON.

Разработка курсового проекта осуществляется в среде Visual Studio Code — это мощный редактор кода с широким набором расширений, которые облегчают процесс создания, отладки и тестирования веб-приложений.

**1.4. Постановка задач программного продукта**

В результате выполнения курсового проекта будет разработан веб-сайт для сети кинотеатров. На этом сайте пользователи смогут узнавать расписание сеансов, бронировать билеты онлайн, а также знакомиться с подробной информацией о фильмах — описанием, трейлерами, актёрским составом и рейтингами.

После просмотра фильма пользователи смогут оставить комментарий и поделиться своими впечатлениями, что позволит сформировать пользовательские оценки и рекомендации для других посетителей сайта.

**1.5 Вывод**

В данном разделе пояснительной записки были проанализированы аналогичные решения с выявлением их сильных и слабых сторон, что позволило сформировать представление о дизайне и функционале будущего сайта. Также были рассмотрены инструменты для разработки веб-страницы, включая редактор Visual Studio Code. Определены требования, которым должен соответствовать итоговый продукт. Эти сведения являются основой для успешного выполнения поставленной задачи и создания качественного веб-сайта.

**2. Проектирование страниц веб-сайта**

**2.1. Выбор способа верстки**

Для размещения элементов на странице использовалась Flex-верстка [7], которая предоставляет эффективные инструменты для создания адаптивного и гибкого интерфейса. С её помощью удобно управлять расположением элементов и их поведением при изменении размеров экрана. Flexbox отличается простотой в использовании и высокой настраиваемостью, что делает его оптимальным решением для реализации данной задачи. Технология поддерживается всеми актуальными браузерами. На основе выбранного подхода к вёрстке были разработаны основные компоненты сайта и созданы макеты страниц, представленные в приложении.

**2.2 Выбор стилевого оформления**

При выборе стиля оформления веб-сайта важно придерживаться ключевого принципа — он должен выделяться среди аналогичных ресурсов в интернете. Дизайн сайта является первым, с чем сталкивается пользователь, и именно первое впечатление зачастую определяет, останется ли посетитель или покинет страницу.

При разработке веб-сайта было принято решение использовать минималистичный стиль оформления. Это решение обусловлено рядом факторов. Прежде всего, минимализм способствует более легкому восприятию информации, делая интерфейс сайта удобным и приятным для пользователя. Кроме того, такой подход обеспечивает современный и опрятный внешний вид, который останется актуальным со временем.

Цветовая палитра играет важную роль в формировании общего впечатления о сайте и создании нужного настроения. Оформление должно соответствовать концепции проекта и быть ориентировано на предпочтения целевой аудитории. Поскольку сайт представляет собой онлайн-платформу сети кинотеатров, важно было выбрать стиль, отражающий атмосферу кинопросмотра. В качестве основной цветовой схемы была выбрана комбинация черного и фиолетового цветов — она подчеркивает современность, создает эффект погружения и визуально ассоциируется с кинозалами.



Рисунок 2.1 – Выбранные цвета для сайта

**2.3 Выбор шрифтового оформления**

В курсовом проекте используются такие шрифты, как Serati и Times New Roman. Подключение шрифтов осуществлялось с помощью правила @font-face из локального файла. Шрифт Serati выбран для основного содержимого сайта, так как он хорошо вписывается в минималистичный стиль оформления и гармонирует с остальными элементами интерфейса. Шрифт Times New Roman используется в футере, придавая информационным блокам строгий и классический вид.



Рисунок 2.1 – Выбранные шрифты для сайта

**2.4 Логотип**

Логотип представляет собой символ, отражающий уникальность бренда с помощью лаконичного визуального образа. Основная цель — создать логотип, который будет одновременно простым и запоминающимся. Его разработкой велась в графическом редакторе Adobe Photoshop.

Логотип представлен на рисунке 2.2.



Рисунок 2.2 – Логотип

**2.5 Разработка пользовательских элементов**

В рамках проекта реализован элемент пользовательского интерфейса — форма для заказа билетов в кинотеатр. На рисунке 2.3 представлена данная форма, которая включает поля для выбора кинотеатра, типа билета, даты, времени сеанса, а также конкретного места в зале. В нижней части формы размещена кнопка для отправки введённых данных.

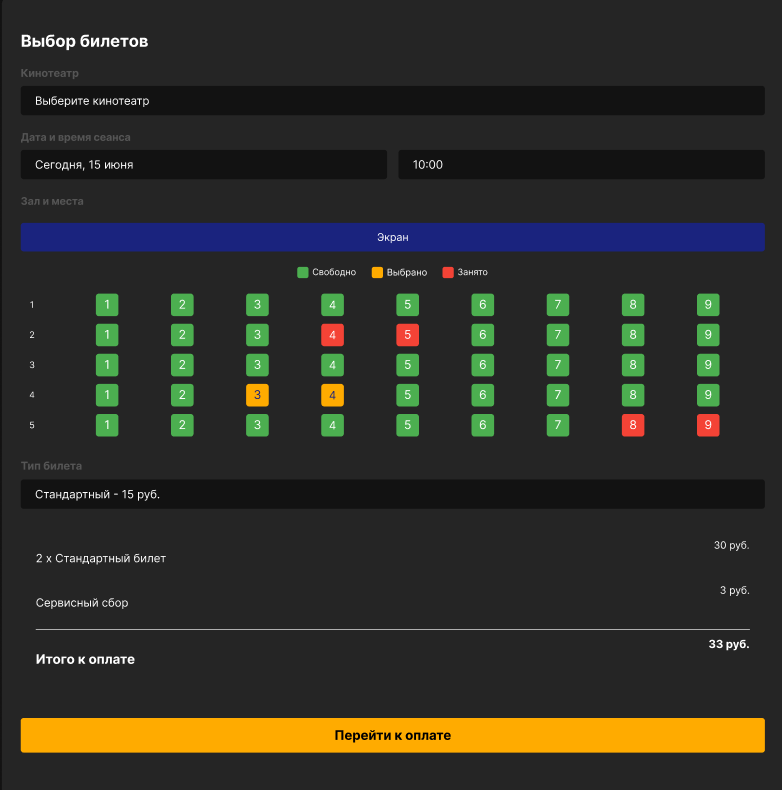


Рисунок 2.3 – форма для выбора билета

**2.6 Разработка спецэффектов**

На страницах веб-сайта реализованы динамические эффекты, которые обеспечивают интерактивность и повышают его привлекательность для пользователей. На главной странице содержатся постеры фильмов, которые меняют свой размер при наведении на них курсора мыши. На рисунке 2.6 показано «до», а на рисунке 2.7 – «после».

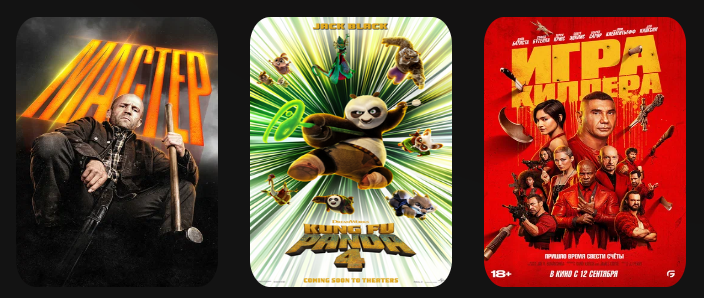


Рисунок 2.6 – Постер до наведения

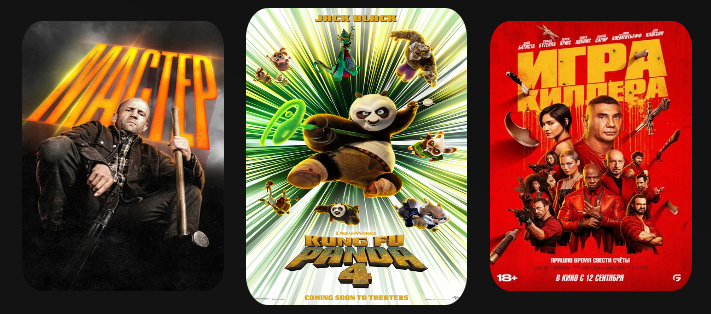


Рисунок 2.7 – Постер после наведения

**2.7 Вывод**

На этом этапе были разработаны прототип сайта, а затем созданы макеты его веб-страниц. Определено стилевое оформление, включая основную цветовую палитру, типографику, элементы пользовательского интерфейса, а также визуальные эффекты и анимации, добавляющие страницам выразительности и динамики

**3. Реализация структуры веб-сайта**

**3.1. Структура HTML-документа**

HTML-документ структурирован в соответствии со стандартом HTML5. В листинге 3.1 показана декларация документа и основное содержимое тега <head>.

<head>

        <meta charset="UTF-8">

        <meta http-equiv="X-UA-Compatible" content="IE=edge">

        <link href="css/global.css" rel="stylesheet" media="all">

        <link href="css/main.css" rel="stylesheet" media="all">

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

        <title> КиноГалактика </title>

</head>

Листинг 3.1 – Декларация и тег head

Структура HTML-документа задаёт основу, или «скелет», для будущего веб-сайта. Верстка выполнена в соответствии со стандартом HTML5.

Тег <head> используется для подключения метаданных и различных вспомогательных элементов, не отображающихся напрямую на странице. Внутри него задаются стили оформления, заголовок страницы и другие служебные данные.

Основной контент размещён внутри тега <body>, где использованы семантические элементы, такие как <header>, <main> и <footer>.

В теге <header> расположено горизонтальное меню, содержащее название компании и ссылки на другие страницы сайта. Также внутри него находится тег <nav>, который активируется при уменьшении ширины экрана. Соответствующий фрагмент кода приведён в приложении В.

В футере размещена информация, включающая раздел помощи по использованию сайта, полезные сведения для посетителей, а также контактные данные компании. Его реализация также представлена в приложении В.

При создании формы использовались стандартные HTML-теги: <form>, <label> и <input>. Для реализации кнопки перехода к форме использованы теги <a> и <button>.

Контент страницы организован с помощью контейнерных блоков <div>, а изображения размещены с использованием тега <img>.

**3.2 Добавление таблиц SCSS и CSS**

Все CSS-файлы размещены в отдельной папке css, а SCSS-файлы — в папке scss. Стили подключаются внешним способом с использованием тега <link>. Пример применения CSS для стилизации шапки сайта приведён в листинге 3.2.

header {

  padding: 10px 0;

  display: flex;

  border-bottom: rgb(36, 34, 34) 2px solid;

  justify-content: space-around;

  flex-wrap: nowrap;

}

header .logo {

  display: flex;

  height: 70px;

  width: 150px;

}

header .nav {

  list-style: none;

  margin: auto;

  width: 30rem;

}

header .nav > li {

  display: inline-block;

}

Листинг 3.2 - Использование CSS для стилизации шапки сайта

Для каждой страницы сайта создан отдельный файл стилей, а общие стили вынесены в файл global.css. Листинг CSS- и SCSS-файлов приведён в приложении Г.

**3.3 Использование JSON, SVG**

В курсовом проекте масштабируемая векторная графика (SVG) использовалась, например, для создания логотипов социальных сетей в футере сайта. На картинке 3.1 показаны логотипы сделанные с помощью SVG.

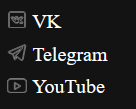


Рисунок 3.1 – Логотипы сделанные с помощью SVG

Комментарии оставленные под фильмами хранятся в формате JSON и помещаются на страницу с помощью JavaScript. Содержимое JSON-файла приведено в приложении Д.

**3.4 Управление элементами DOM**

Загрузка комментариев из JSON-файла была реализована при помощи JavaScript. Содержимое файла MIB.js и rushHour.js приведено в приложении Ж.

На главной странице сайта реализован слайдер для отображения постеров фильмов, выполненный с использованием JavaScript. Слайдер осуществляет ручную прокрутку изображений, что обеспечивает удобный просмотр фильмов пользователем. При открытии сайта с мобильного устройства элементы управления прокруткой скрываются, а переключение между слайдами осуществляется с помощью свайпов, что делает интерфейс более адаптированным под сенсорные экраны и улучшает пользовательский опыт.

**3.5 Вывод**

На этом этапе была успешно выполнена разработка основной структуры веб-сайта, включая главную и остальные страницы, предусмотренные на начальном этапе проекта. Все элементы и блоки сайта были аккуратно интегрированы согласно их назначению и расположению, определённым при создании прототипа.

Также важной частью этапа стала верстка страниц, в ходе которой были установлены блоки контента, их структура и расположение, а также применены CSS-стили для оформления и визуализации сайта.

В результате реализованная структура позволила создать полноценную и работоспособную веб-платформу, которая полностью отвечает требованиям, заложенным в прототипе. Сайт готов к запуску и использованию конечными пользователями.

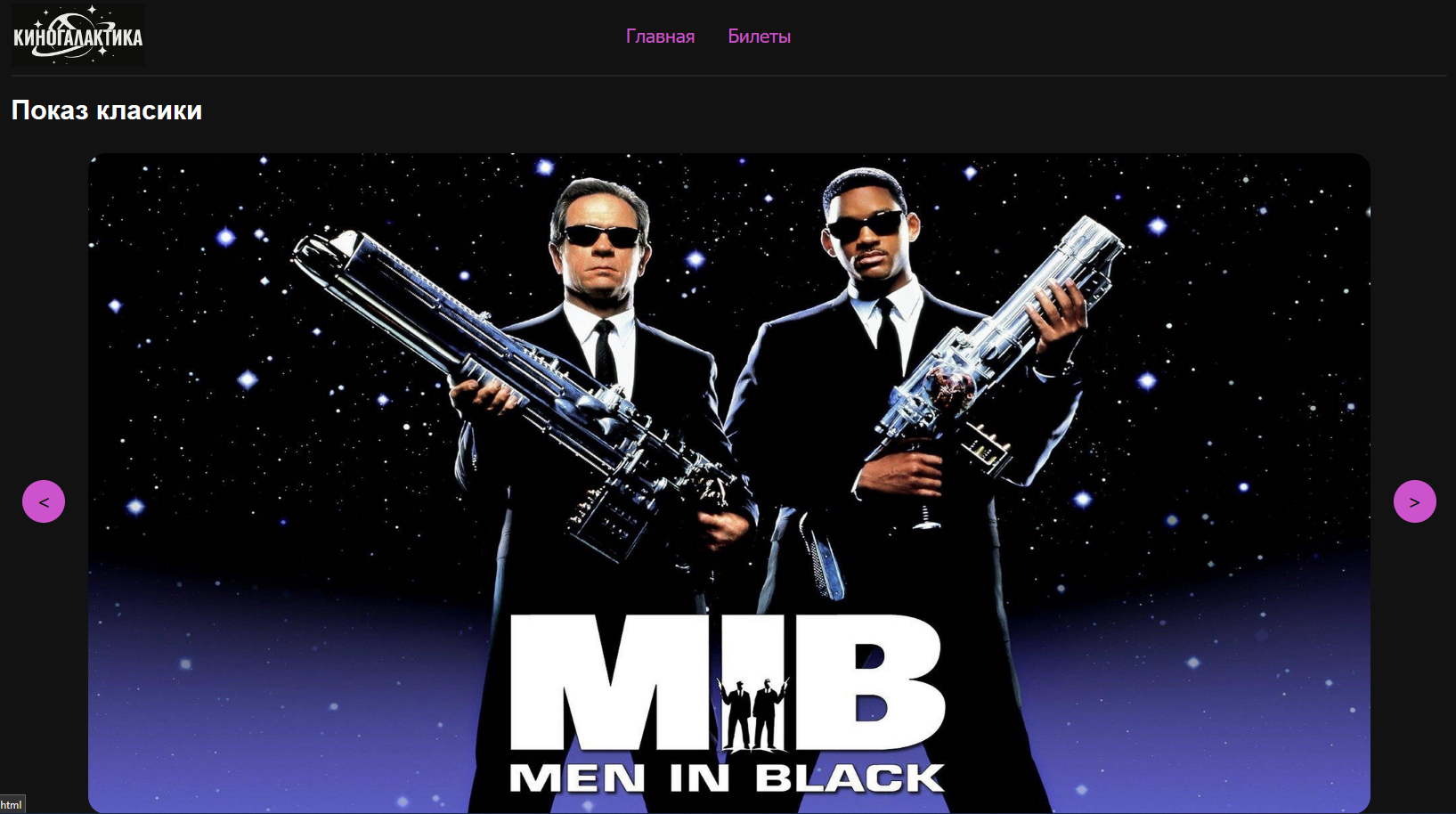
**4. Тестирование веб-сайта**

**4.1 Адаптивный дизайн веб-сайта**

Адаптивность — одно из ключевых требований к современным веб-сайтам. Ресурс должен корректно отображаться на экранах компьютеров, планшетов и смартфонов. Для этого применяется адаптивный дизайн и адаптивная верстка. Такой подход усложняет процесс разработки и требует дополнительных ресурсов, но результат оправдывает усилия: сайт стабильно работает на всех основных устройствах, что обеспечивает максимальный охват аудитории.

Адаптивность означает способность сайта автоматически подстраиваться под разные технические условия и размеры экранов.

В рамках курсового проекта были использованы четыре основных диапазона ширины экрана для медиа-запросов: 1280px 768px и 480px. Эти значения выбраны потому, что структура сайта остаётся читабельной и удобной для восприятия на данных ширинах без необходимости существенных изменений. Фрагмент главной страницы для диапазонов 1920px представлен на рисунке 4.1, а тот же фрагмент для диапазона 425px – 320px — на рисунке 4.2.



**ПРИЛОЖЕНИЕ А**

**Макеты веб-страниц**