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

Факультет Информационных технологий

Кафедра Информационных систем и технологий

Специальность 6-05-0611-01 «Информационные системы и технологии»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА КУРСОВОГО ПРОЕКТА**

по дисциплине «Компьютерные языки разметки»

Тема: «Веб-сайт «Дорамы»

**Исполнитель**

студент 1 курса 3 группы \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Шидловская Я.В

подпись, дата

**Руководитель**

ассистент \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Харланович А.В

должность, учен. степень, ученое звание подпись, дата

Допущена к защите \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

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

дата, подпись

Курсовой проект защищен с оценкой

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ А.В. Харланович

подпись дата инициалы и фамилия

Минск, 2024

**Содержание**

[Введение 4](#_Toc134147544)

[1. Постановка задачи 5](#_Toc134147545)

[1.1. Обзор аналогичных решений 5](#_Toc134147546)

[1.2. Техническое задание 7](#_Toc134147547)

[1.3. Выбор средств реализации программного продукта 8](#_Toc134147548)

[1.4 Вывод 8](#_Toc134147549)

[2. Проектирование страниц веб-сайта 9](#_Toc134147550)

[2.1. Выбор способа вёрстки 9](#_Toc134147551)

[2.2. Выбор стилевого оформления 9](#_Toc134147552)

[2.3. Выбор шрифтового оформления 1](#_Toc134147553)0

[2.4. Разработка логотипа 1](#_Toc134147554)0

[2.5. Разработка пользовательских элементов 10](#_Toc134147555)

[2.6. Разработка спецэффектов 1](#_Toc134147556)1

[2.7. Выводы 12](#_Toc134147557)

[3. Реализация структуры веб-сайта 13](#_Toc134147558)

[3.1. Структура HTML-документа 13](#_Toc134147559)

[3.2. Добавление таблиц стилей SCSS и CSS 1](#_Toc134147560)4

[3.3. Использование стандартов XML (SVG) 1](#_Toc134147561)4

[3.4 Управление элементами DOM 1](#_Toc134147562)5

[3.5. Выводы 1](#_Toc134147563)5

[4. Тестирование веб-сайта 1](#_Toc134147564)6

[4.1. Кроссбраузерность веб-сайта 1](#_Toc134147565)6

[4.2. Адаптивный дизайн 2](#_Toc134147567)0

[4.3 Вывод 2](#_Toc134147568)1

[Заключение 23](#_Toc134147569)

[Список использованных источников 24](#_Toc134147570)

**Введение**

В современном информационном обществе Интернет стал неотъемлемой частью нашей повседневной жизни. Все больше людей обращаются к сети для получения информации, развлечения и общения. Одной из популярных тематик, которая привлекает огромное количество пользователей, являются дорамы – телесериалы из Южной Кореи и других стран Восточной Азии.

Дорамы, также известные как корейские драмы, обладают огромной популярностью во всем мире. Они привлекают зрителей своим уникальным стилем и захватывающими сюжетами, которые часто сочетают в себе элементы романтики, драмы, комедии и мистики. Благодаря интернет-технологиям, дорамы стали доступны для просмотра в любое время и в любом месте.

Целью данной курсовой работы является разработка и создание веб-сайта, посвященного дорамам. Веб-сайт будет предоставлять удобный и информативный интерфейс для поиска и ознакомления с различными дорамами.

Таким образом, разработка веб-сайта для дорам предоставляет широкий спектр возможностей и может быть ценной задачей с несколькими ключевыми направлениями. В ходе работы над проектом были поставлены следующие задачи:

1. Создание прототипа с использованием графического редактора Figma.
2. С помощью XML-формата обеспечиваем хранение данных.
3. Создание эффективного и привлекательного веб-сайта, учитывающего удобство навигации, яркость контента и привлекательность визуального оформления.
4. Тестирование веб-сайта с целью обнаружения и исправления возможных ошибок, а также проверки работы всех функциональных элементов платформы.
5. Добавление интерактивных функций для повышения привлекательности и пользовательского опыта при использовании сайта.

Цели курсового проекта: анализ сайтов похожей тематики, выявление достоинств и недостатков этих сайтов, улучшение навыков верстки и получение опыта разработки проектов, ознакомление с новыми технологиями для создания сайтов, выполнение поставленных задач.

**[1. Постановка задачи](https://docs.google.com/document/d/1HUvb3lcG6yJYhUj8VjKk-OT9mxhwF01-eY6OvWfwnQg/edit" \l "heading=h.30j0zll)**

Цель сайта *Dorama.Jin* – просмотр дорам и трейлеров к ним, отметка просмотренных сериалов и тех, которые в будущем хотите посмотреть, чтение описания фильмов и сериалов. Сайт создан в развлекательных целях, для любителей азиатской культуры.

**[1.1 Обзор аналогичных решений](https://www.readed.me)**

При создании сайта по просмотру сериалов и фильмов важно использовать различные аналитические решения, которые помогают оптимизировать сайт и улучшить его удобство использования. Они позволяют учитывать потребности пользователей и достигать лучших результатов в поисковых системах. Поэтому крайне важно проанализировать сайты конкурентов, на основе анализа будет создан собственный проект.

В качестве аналога был выбран другой сайт по просмотру дорам, а именно: *doramy.club.*

* + 1. **Аналог *doramy.club***

Заходя на сайт мы сразу можем видеть различные дорамы, название сайта, поиск, фильтры, профиль и тему страницы. Это все продемонстрировано на (рисунке 1.1).

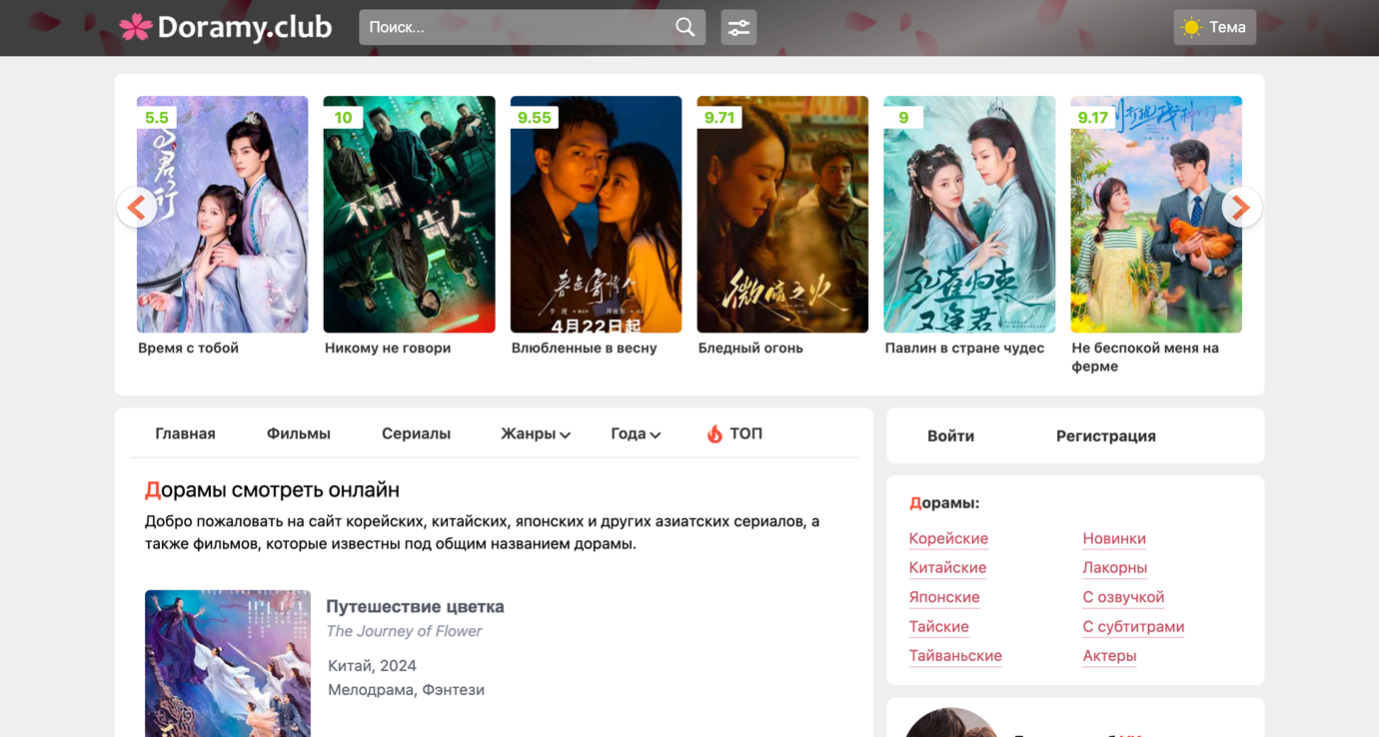


Рис. 1.1 – Главная страница сайта

Редакторы сайта придерживаются одного стиля во всех вкладках страницы, наполняя их максимумом информации необходимой для пользователя.

Это и стало одним из плюсов этого веб-сайта: пользователю дают всю необходимую ему информацию сразу, что делает сайт приятным и легким для пользования.

Если нажать кнопу “Войти” или “Регистрация”, то попадаем на страницы, изображённые на рисунке 1.2 и 1.3 соответственно.

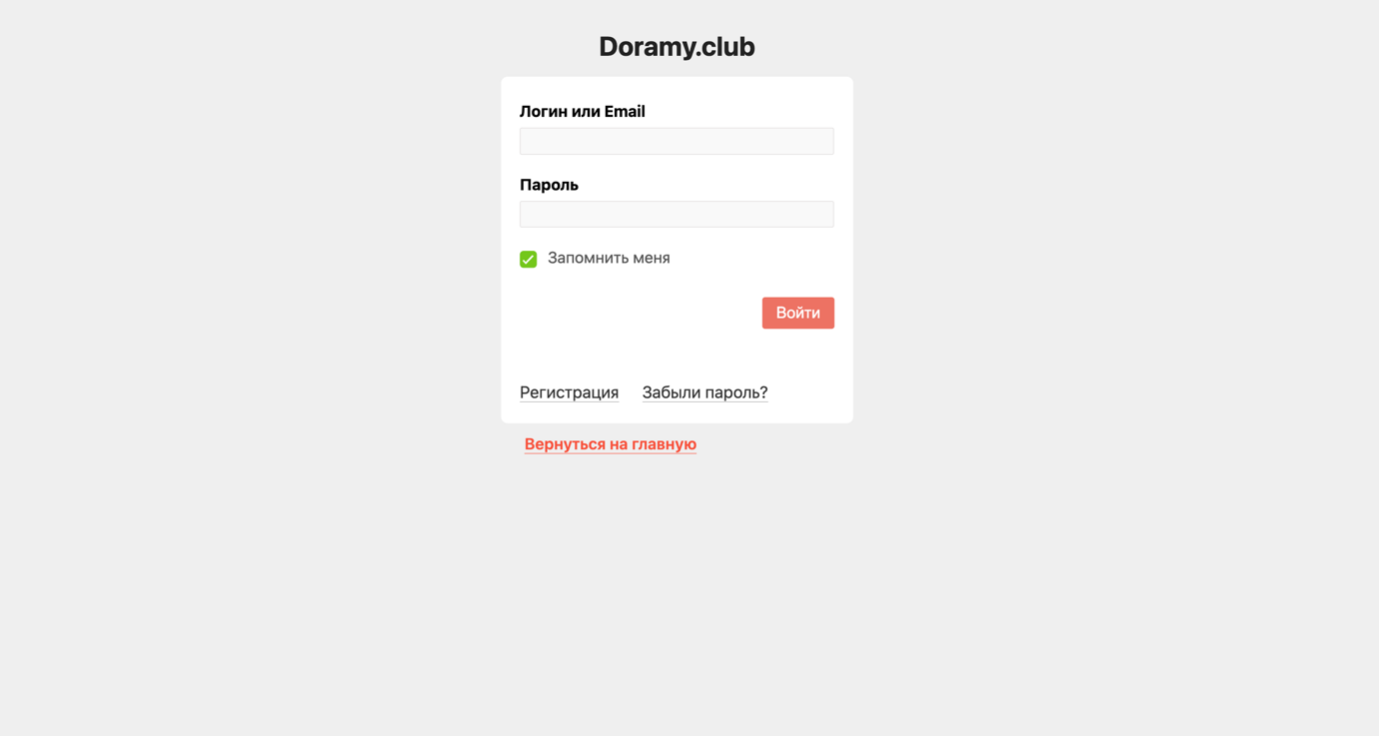


Рис. 1.2 – Вход на свой аккаунт этого сайта

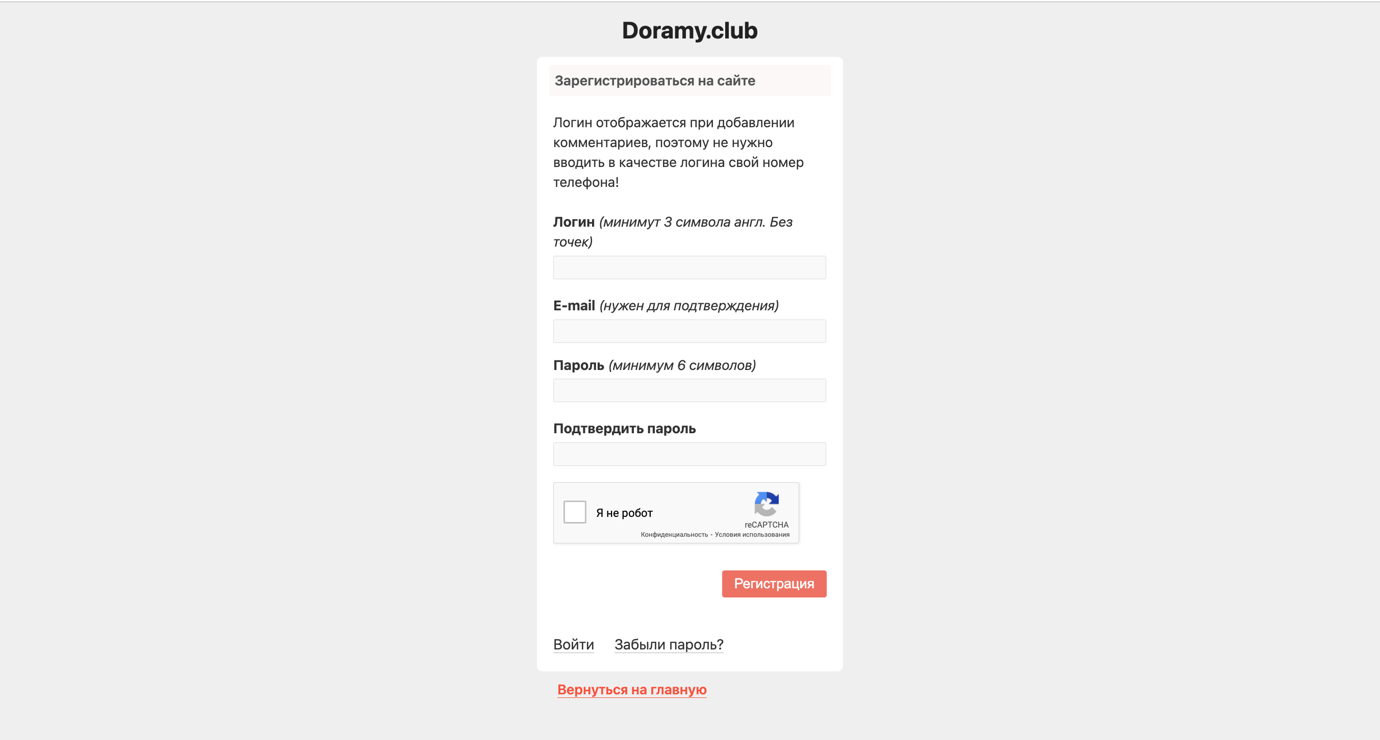


Рис. 1.3 – Регистрация аккаунта на этом сайте

Также в подвале сайта находится ограничение по возрасту, Контакты и Правообладателям (рисунок 1.4).



Рисунок 1.4 - Footer

Основная цветовая палитра сайта нейтральные цвета, такие как белый, серый, черный, а также оттенок красного, что позволяет выделить важные элементы на фоне других. (рисунок 1.5).



Рисунок 1.5 - Цветовая палитра

Разметка сайта выполнена с помощью *HTML*5 и *CSS*/*SCSS*, а также был использован динамический скриптовой язык программирования *JavaScript*. Веб-ресурс является адаптивным и кросс-браузерным, это определяется наличием различным стилей в *CSS*/*SCSS* и префиксами.

* 1. **Техническое задание**

Поставлена задача реализовать веб-сайт на тему «Дорамы». Сайт должен содержать актуальную информацию. С его помощью пользователь сможет быстро искать необходимые дорамы смотреть их онлайн, а так же сортировать их по категориям.

Меню навигации по сайту на всех страницах будет располагаться сверху. В меню  (рисунок 1.6). располагаются такие элементы как:

a.поисковая строка

b. аккаунт

c. главная

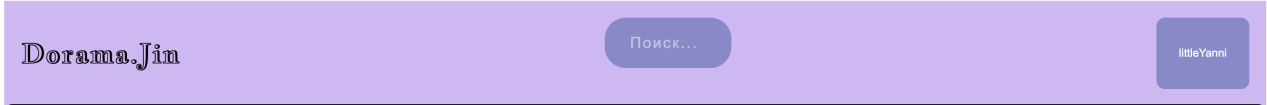


Рисунок 1.6 - Меню навигации

* 1. **Выбор средств реализации программного продукта**

Первым этапом является создание макета страниц веб-сайта. Для этого использовалось приложение для разработки интерфейсов и прототипирования *Figma* (онлайн-сервис для дизайнеров, веб-разработчиков и маркетологов. Он предназначен для создания прототипов сайтов или приложений, иллюстраций и векторной графики. В редакторе можно настроить совместную работу, вносить и обсуждать правки, причём как в браузере, так и через приложение на компьютере). Структура сайта создана с помощью языка разметки *HTML*, дизайн сайта оформлен с помощью *CSS*/*Sass*. *JavaScript* позволит добавить динамичности сайта.

Для написания исходного кода был выбран бесплатный редактор *WebStorm* – это удобное программное обеспечение, предназначенное специально для написания кода, и в частности вёрстки сайта. Для этого в программе предусмотрено множество встроенных функций, ускоряющих процесс написания кода, благодаря всплывающим подсказкам, навигации по написанному коду, автоматическому формированию отступов вложенных строк и встроенной функции автозаполнения. *WebStorm*  имеет отзывчивый и приятный интерфейс и возможность подключения огромного множества различных расширений (плагинов) отлично дополняющих основной функционал.

* 1. **Вывод**

В данном разделе были рассмотрены аналогичные решения, были проанализированы их недостатки и преимущества, для создания в последующем качественного веб-сайта. Были поставлены конкретные задачи для создания интерактивного информационного ресурса. Кроме того, были рассмотрены средства реализации программного продукта, такие как языки разметки, а также редактор кода *WebStorm* . Были проанализированы особенности языков и основные дополнительные средства редактора. Все вышеперечисленные пункты понадобятся для достижения поставленной задачи и для создания в дальнейшем качественного продукта.

1. **Проектирование страниц веб-сайта**
   1. **Выбор способа верстки**

Для того, чтобы сайт был удобным для пользователей различных устройств необходима адаптивность. Для этого будут использованы flex-box и grid вёрстки. Они хорошо подходят для создания таких элементов как горизонтальное меню, футер сайта и так далее. Эта технология позволяет очень легко и гибко расставить элементы в контейнере, распределить доступное пространство между ними и выровнять их даже если они не имеют конкретных размеров.

* 1. **Выбор стилевого оформления**

Выбор цветовой гаммы — важный этап разработки проекта. Цвета должны не только сочетаться между собой, но и отражать тему проекта.

Для данного веб-сайта была выбрана достаточно светлая цветовая палитра (рисунок 2.1), с использованием других цветов для подчёркивания содержимого (рисунок 2.2).



Рисунок 2.1- Цветовая палитра

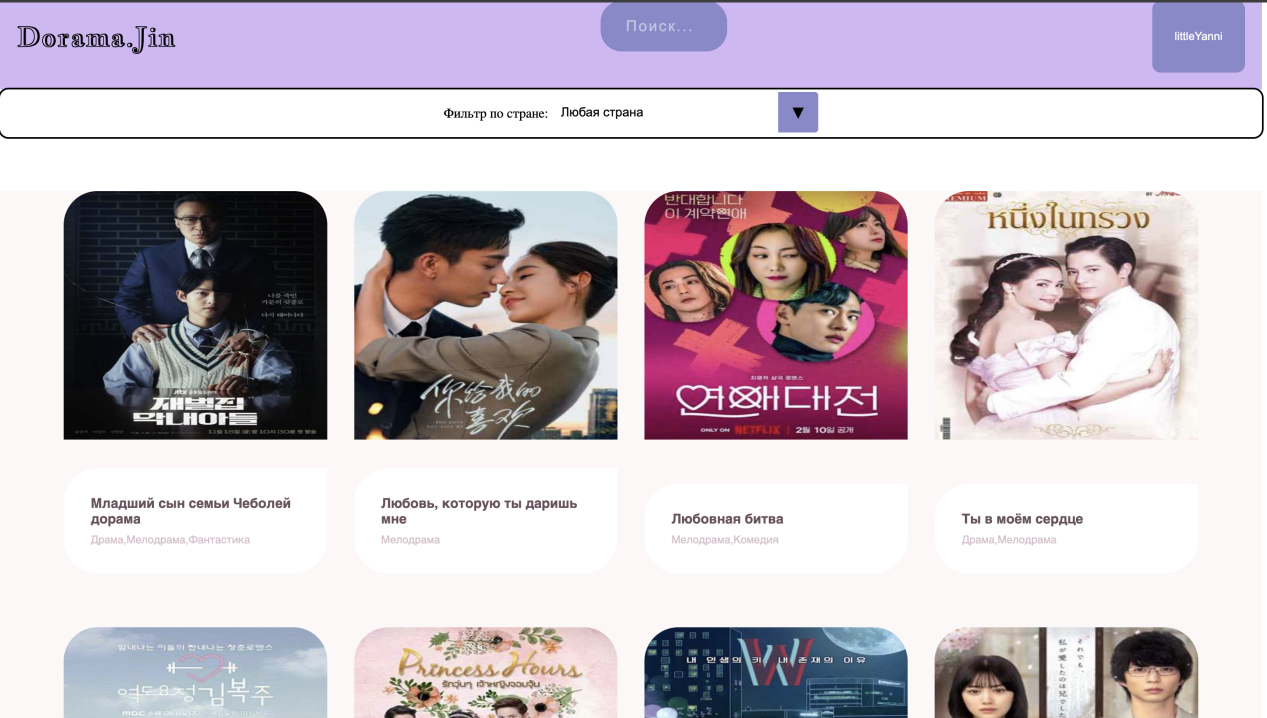


Рисунок 2.2- Титульная страница

**2.3 Выбор шрифтового оформления**

В процессе создания данного макета было выбрано несколько основных шрифтов: «Jacques Francois Shadow», «Noto Sans JP».

Все они между собой хорошо сочетаются, а также отлично подходят под выбранный стиль сайта.

**2.4 Разработка логотипа**

Логотип — это фирменный знак, который люди ассоциируют с конкретным брендом. Хороший логотип должен быть масштабируемым, разборчивым, достаточно простым, запоминающимся и узнаваемым.

Для данной веб-страницы был разработан логотип в подходящем ему стиле (рисунок 2.3)



Рисунок 2.3- Логотип сайта

**2.5 Разработка пользовательских элементов**

В проекте представлены такие элементы пользовательского интерфейса.Навигация – один из наиболее удобных элементов на сайте. Она помогает пользователю быстро перемещаться между страницами и по странице. Навигационное меню представлено на (рисунок 2.4)

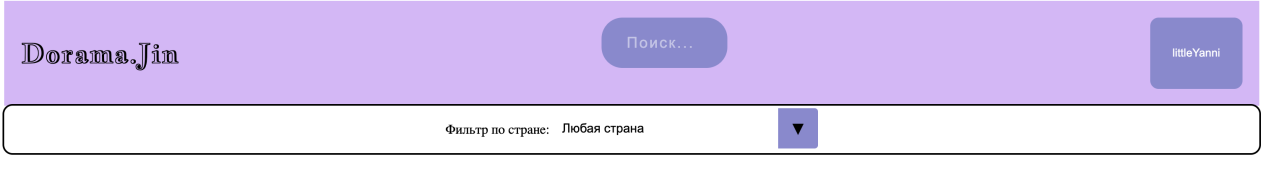


Рисунок 2.4 -  Навигационное меню

Для обратной связи с пользователем, было разработано специальное меню. (рисунок 2.5).

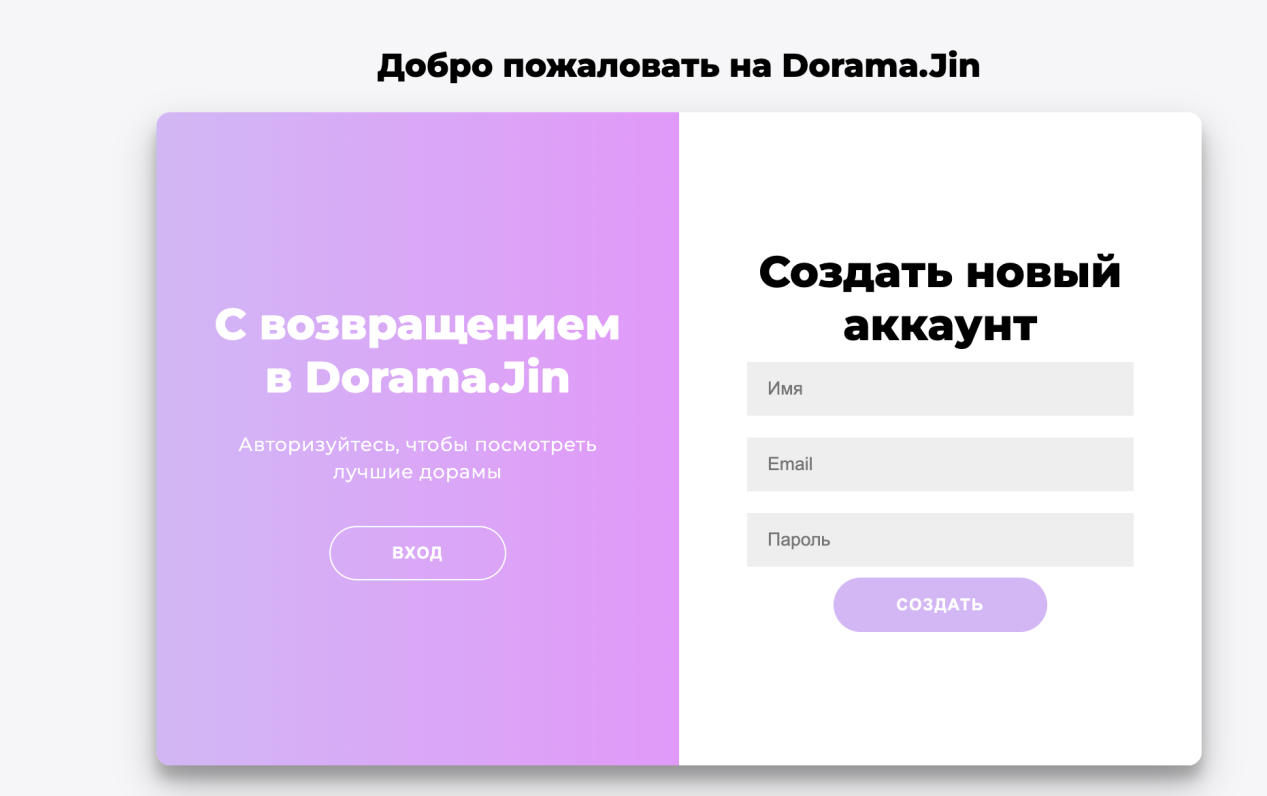


Рисунок 2.4 - Меню регистрации

Данный пользовательский элемент (рисунок 2.5) привлекает внимание посетителя и поможет быстро переместиться к интересующему контенту. Каждый элемент является интерактивным и меняет свой внешний вид наведении на него мыши.

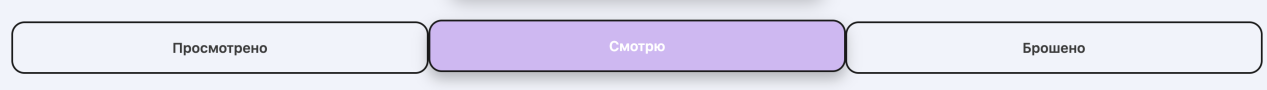
****

Рисунок 2.5 - Страница аккаунта

**2.6 Разработка спецэффектов**

На сайте разработаны некоторые динамические эффекты и анимации, которые позволят придать сайту интерактивность и сделают его более привлекательным для пользователя.

На сайте присутствуют изображения с разделами сайта, которые при наведении курсор мыши выдвигает описание дорамы (рисунок 2.8).

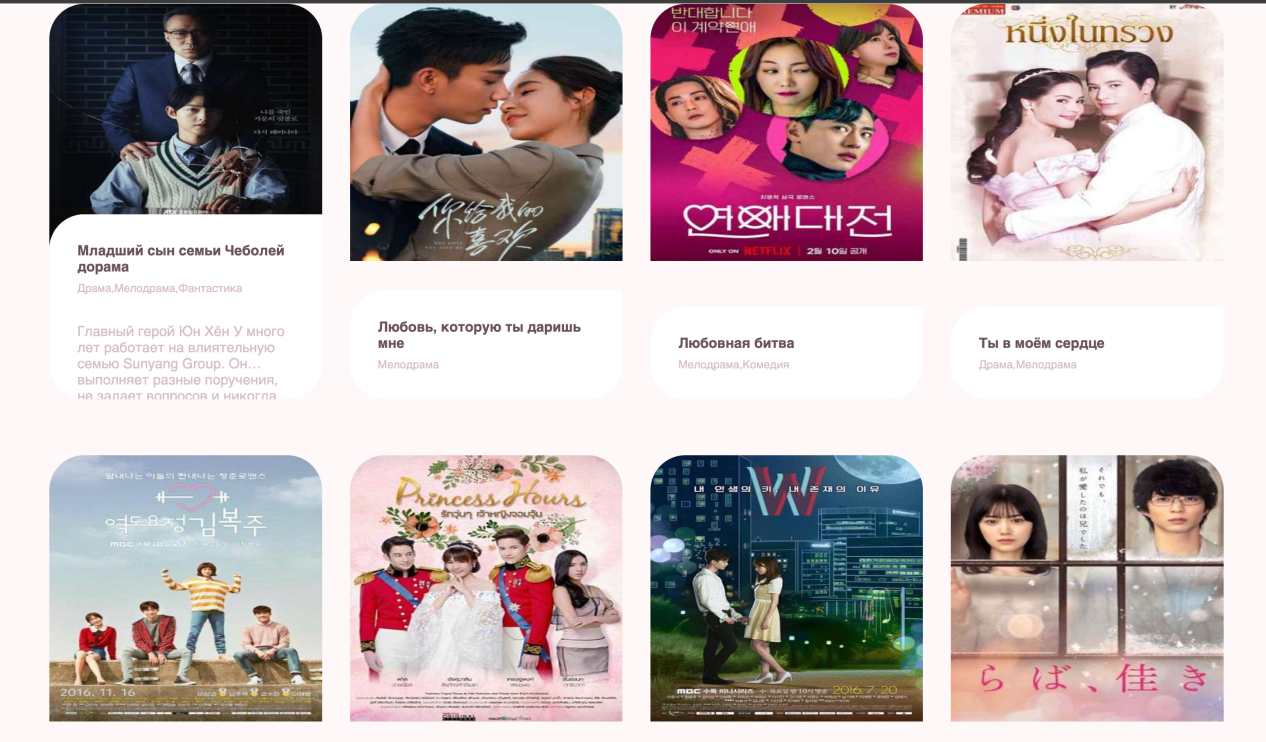


Рисунок 2.8- Главная страница

**2.7. Выводы**

В ходе данной главы была определена структура сайта и составлен макет в Figma с переходами между страницами. Разобрав все подходящие по оформлению аналоги, разработав логотип, пользовательские элементы и спецэффекты, а также подобрав нужную цветовую палитру вместе с шрифтами, был создан макет сайта. Исходя из целей разработать адаптивный сайт был выбран способ верстки.

1. **Реализация структуры веб-сайта**

**3.1 Структура HTML-документа**

Структура HTML-документа определяет базовый «скелет» для будущего веб-сайта. Код выполнен по стандарту HTML5.

HTML-документ состоит из двух частей: **заголовок** (head) и **тело** (body), расположенных в следующем порядке:

<HTML>

<HEAD>

...

</HEAD>

<BODY>

...

</BODY>

</HTML>

В начале документа в теге <head></head> прописана служебная информация, название веб-сайта. Тег <head> предназначен для хранения других элементов, цель которых — помочь браузеру в работе с данными.

Также внутри контейнера <head> находятся метатеги, которые используются для хранения информации, предназначенной для браузеров и поисковых систем. Например, механизмы поисковых систем обращаются к метатегам для получения описания сайта, ключевых слов и других данных.

Содержимое тега <head> не отображается напрямую на веб-странице, за исключением тега <title> устанавливающего заголовок окна веб-страницы.

Далее расположен тег <body></body>, внутри которого размещаются все теги, в которых расположен сайт.

Структура тела документа состоит из семантических тегов header, section и footer. Листинг представлен в приложении 3. Эти теги предназначены для того, чтобы компьютерные программы понимали, какой тип информации заложен в данных тегах.

В теге header содержится логотип сайта, поисковая строка и кнопка аккаунта.(рисунок 3.1)

<header id="baseHeader">

    <h1 id="headerH1">Dorama.Jin</h1>

    <div class="search-box">

        <input type="text" id="input-search" class="input-search" placeholder="Поиск...">

    </div>

    <button id="profileButton" class="button-85" role="button">Логин</button>

</header>

Рисунок 3.1 - Содержание тега <head>

В коде находятся несколько тегов section, каждый из которых представляет собой какую-то основную часть сайта.

**3.2 Добавление таблиц стилей Sass и CSS**

Добавление таблиц стилей CSS позволяет упростить создание стилистического оформления страницы.

Для поиска  использован следующий код CSS(рисунок 3.2)

.input-search {

    height: 50px;

    width: 150px;

    border-style: none;

    padding: 30px;

    font-size: 18px;

    letter-spacing: 2px;

    outline: none;

    border-radius: 25px;

    transition: all .5s ease-in-out;

    background-color: #8989cc;

    padding-right: 40px;

    color: #fff;

}

.input-search::placeholder {

    color: rgba(255, 255, 255, .5);

    font-size: 18px;

    letter-spacing: 2px;

    font-weight: 100;

}

Рисунок 3.1 - CSS код для поиска

## **3.3** Использование стандартов XML (SVG)

XML – язык, который используется для структурированного хранения информации. При разработке данного проекта XML использовался для заполнения информацией содержимое подвала страницы. На (рисунок 3.3)

находится пример информации.

<footer>

    <div class="xml">

        <XML ID="dsoBook">

            <?xml version="1.0"?>

            <PNONE>

                Контактный номер

            </PNONE>

            <NUMBER>

                +375293611196

            </NUMBER>

        </XML>

    </div>

</footer>

Листинг 3.3 –Хранение данных о номере телефона в XML

**3.4 Управление элементами DOM**

JavaScript – это язык программирования, который используют разработчики для создания интерактивных веб-страниц. Функции JavaScript могут улучшить удобство взаимодействия пользователя с веб-сайтом: от обновления ленты новостей в социальных сетях и до отображения анимации и интерактивных карт.

Может подключаться как внутри самого HTML-документа, так и отдельным файлом.

В данном проекте JavaScript был использован для регистрации, сохранения данных в LocalStorage, для автоматического заполнения информации дорам и выводы их, для того чтобы работали кнопки( фильтра и стадии просмотра) и работал поиск.

Представленном ниже коде используется JavaScript для проверки авторизации (рисунок 3.4)

setInterval(() => {

    let currentUser = JSON.parse(localStorage.getItem('currentUser'));

    let profileButton = document.getElementById('profileButton');

    if (currentUser == null) {

        profileButton.text = 'Логин';

        window.location = '../signInAndSignUp/signInAndSignUp.html';

    } else {

        if (!window.location.href.includes('/userProfile/userProfile.html'))

            profileButton.textContent = currentUser['name'];

    }

}, 100);

Рисунок 3.4 - Проверка авторизации

## **3.5 Выводы**

На данном этапе была реализована структура на HTML, созданы таблицы стилей Sass/CSS. Было продемонстрировано использование языка программирования JS. Определено стилевое оформление веб-сайта, созданы анимации, а также веб-сайт был подготовлен к следующему этапу – тестированию.

1. **Тестирование веб-сайта**

**4.1 Кроссбраузерность веб-сайта**

В кроссбраузерное тестирование входит общий вид проекта в других браузерах (сохранились ли шрифты, не съехала ли анимация и так далее).

Для проверки, созданной кроссбраузерности веб-сайта использованы такие современные браузеры, как Google Chrome, Safari, Firefox . Пример корректной работы веб-сайта в таких браузерах как Google Chrome и Safari представлен на рисунках 4.1 - 4.8.

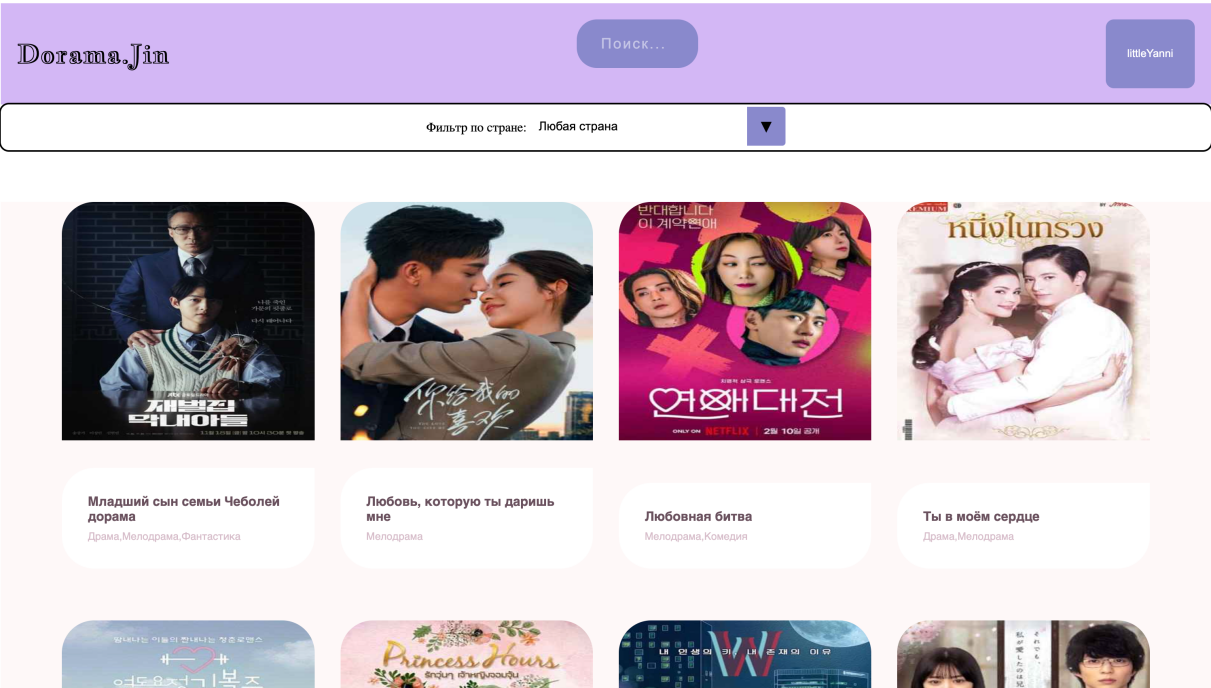


Рисунок 4.1- Google Chrome главная страница

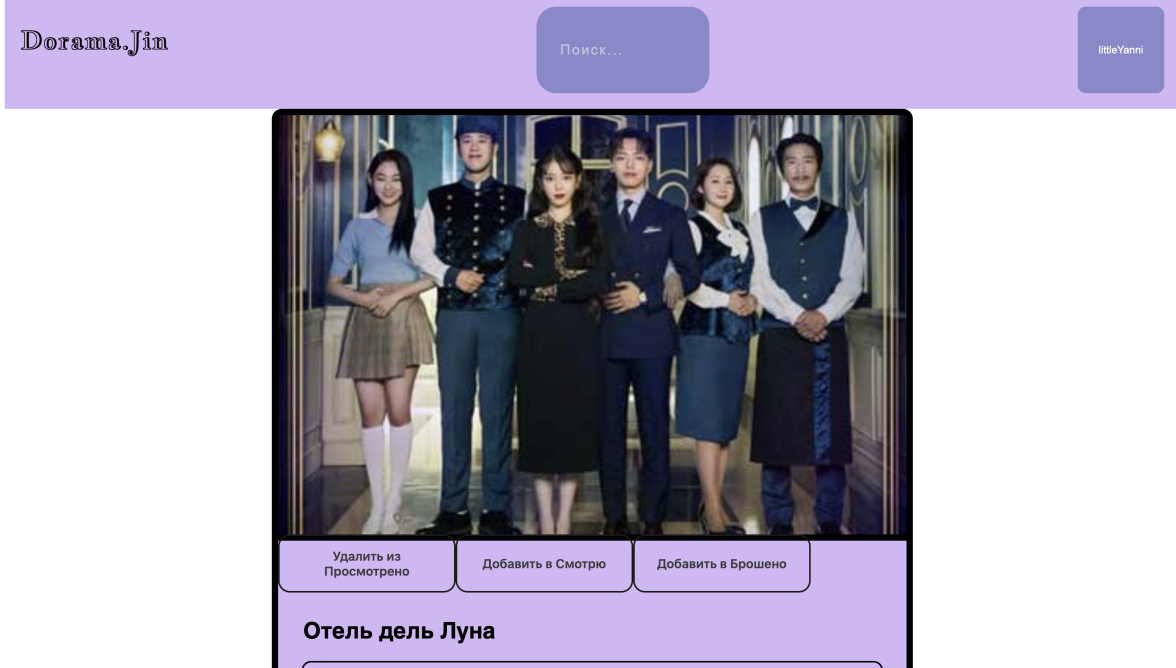


Рисунок 4.2- Google Chrome детальной страница

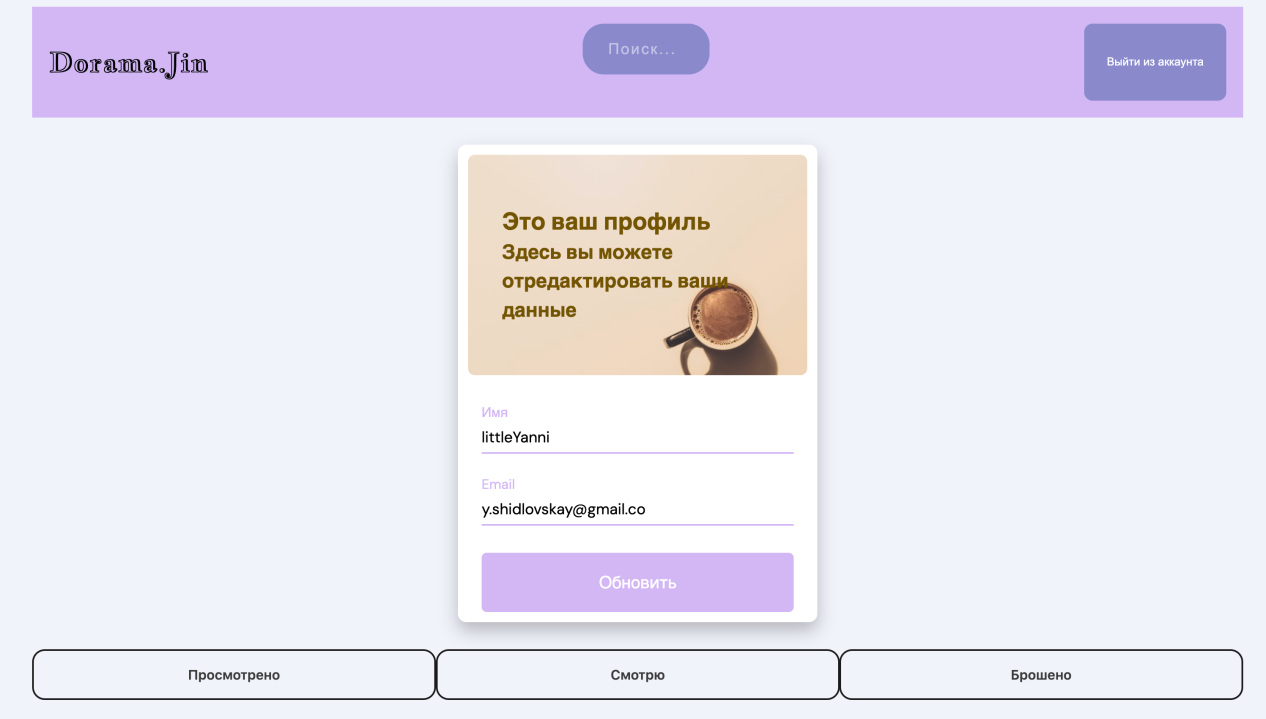


Рисунок 4.3- Google Chrome аккаунт

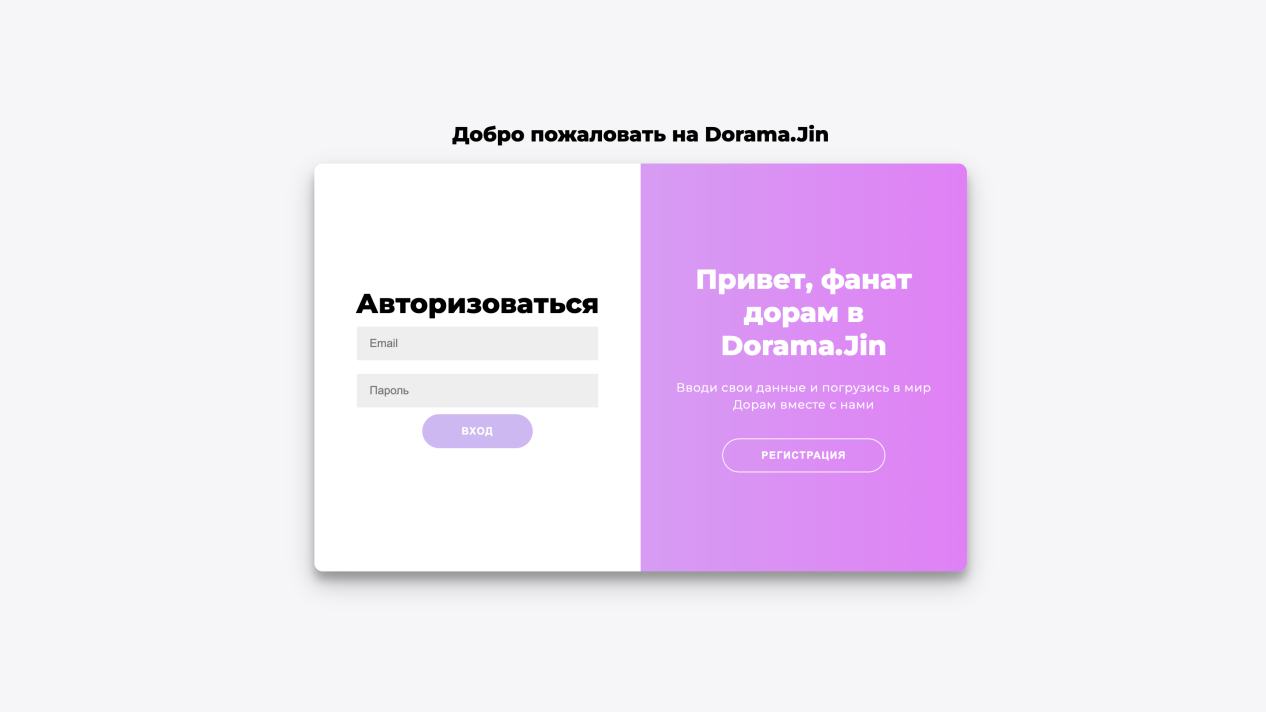


Рисунок 4.4- Google Chrome регистрация

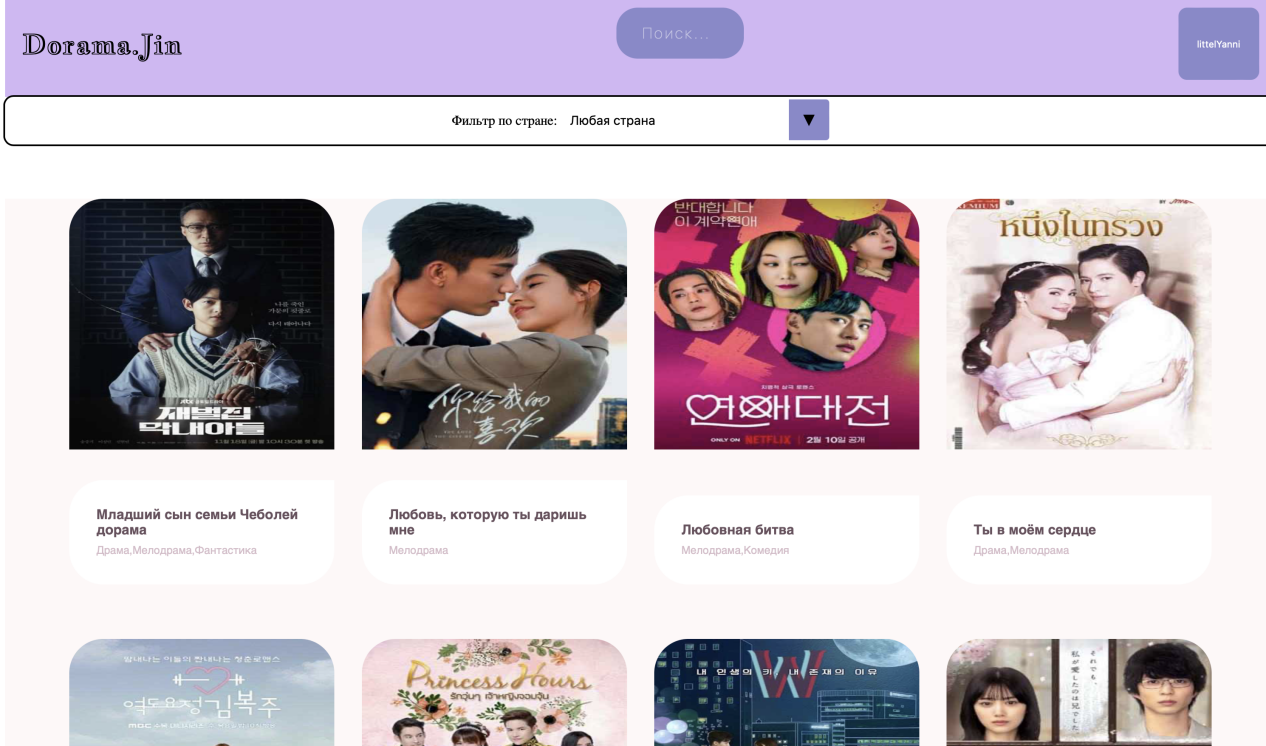


Рисунок 4.5- Safari главная страница

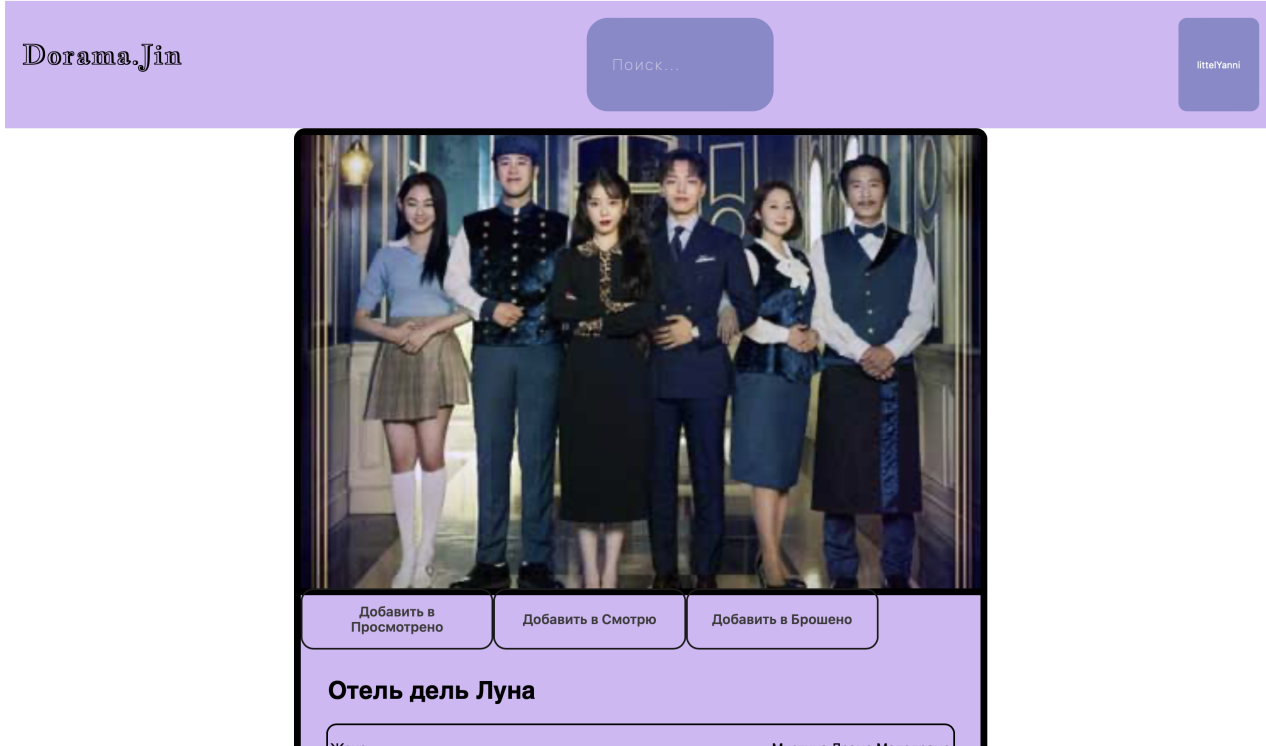


Рисунок 4.6- Safari детальной страница

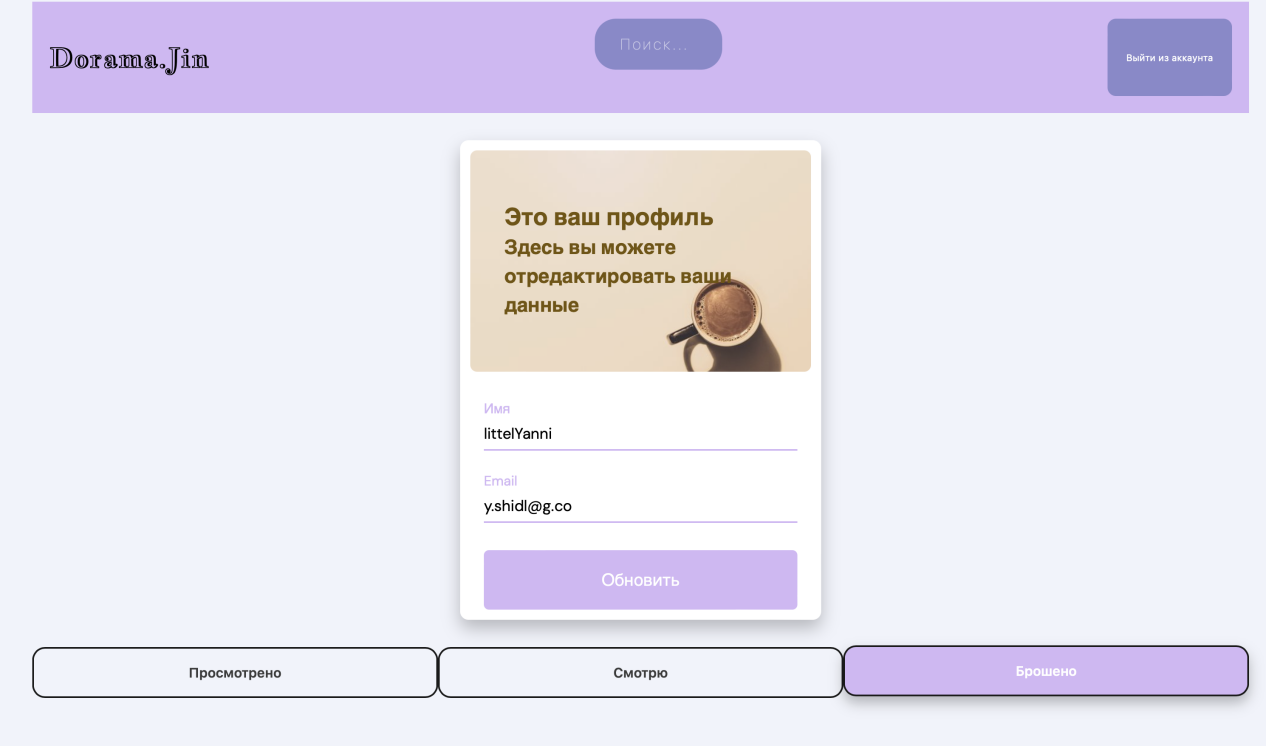


Рисунок 4.7- Safari аккаунт

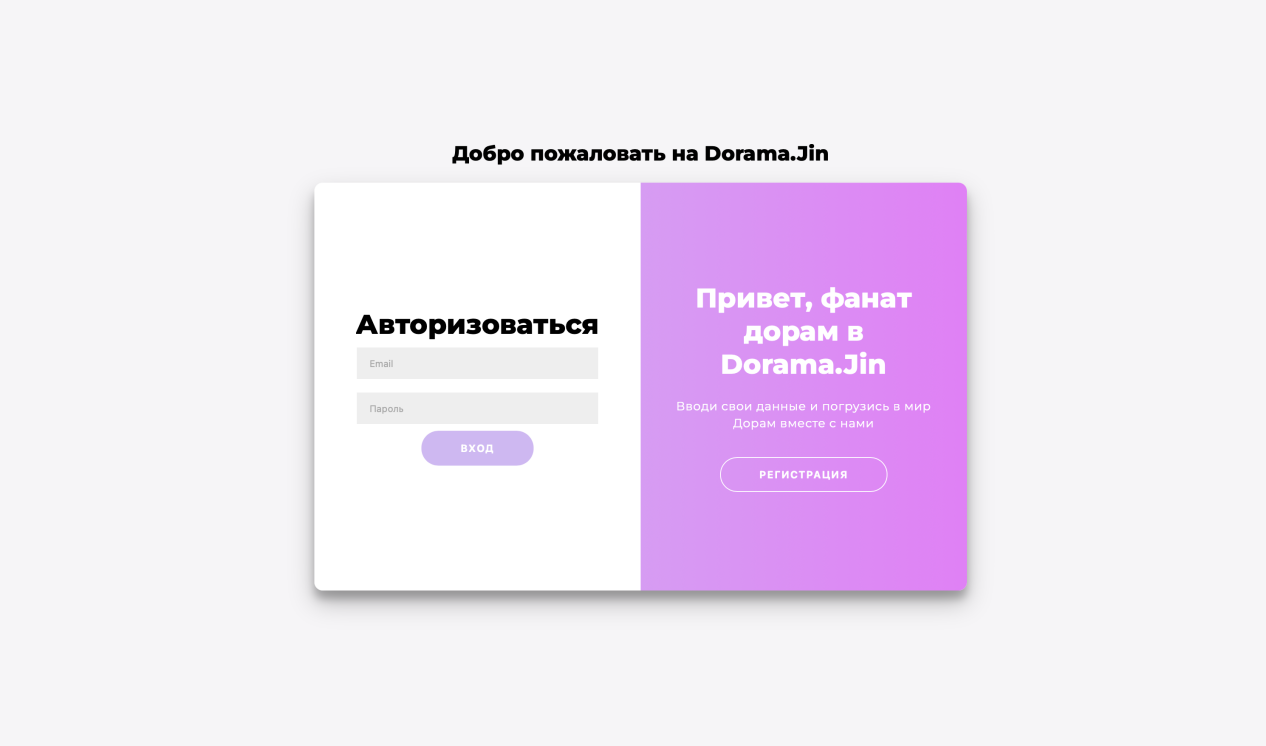


Рисунок 4.8- Safari регистрация

**4.2 Адаптивный дизайн веб-сайта**

Адаптивная вёрстка сайта — это способ обеспечить оптимальное отображение интерфейсов на устройствах с разным разрешением экранов.

Адаптивный дизайн предусматривает, что пользователь будет взаимодействовать с кнопками, текстом, изображениями через разные устройства. Думает об удобстве пользователя в этом случае уже дизайнер.

Для адаптива данного проекта были выбраны медиа-запросы. Адаптив был сделан для телефона (430px – 932px).(рисунок 4.9).

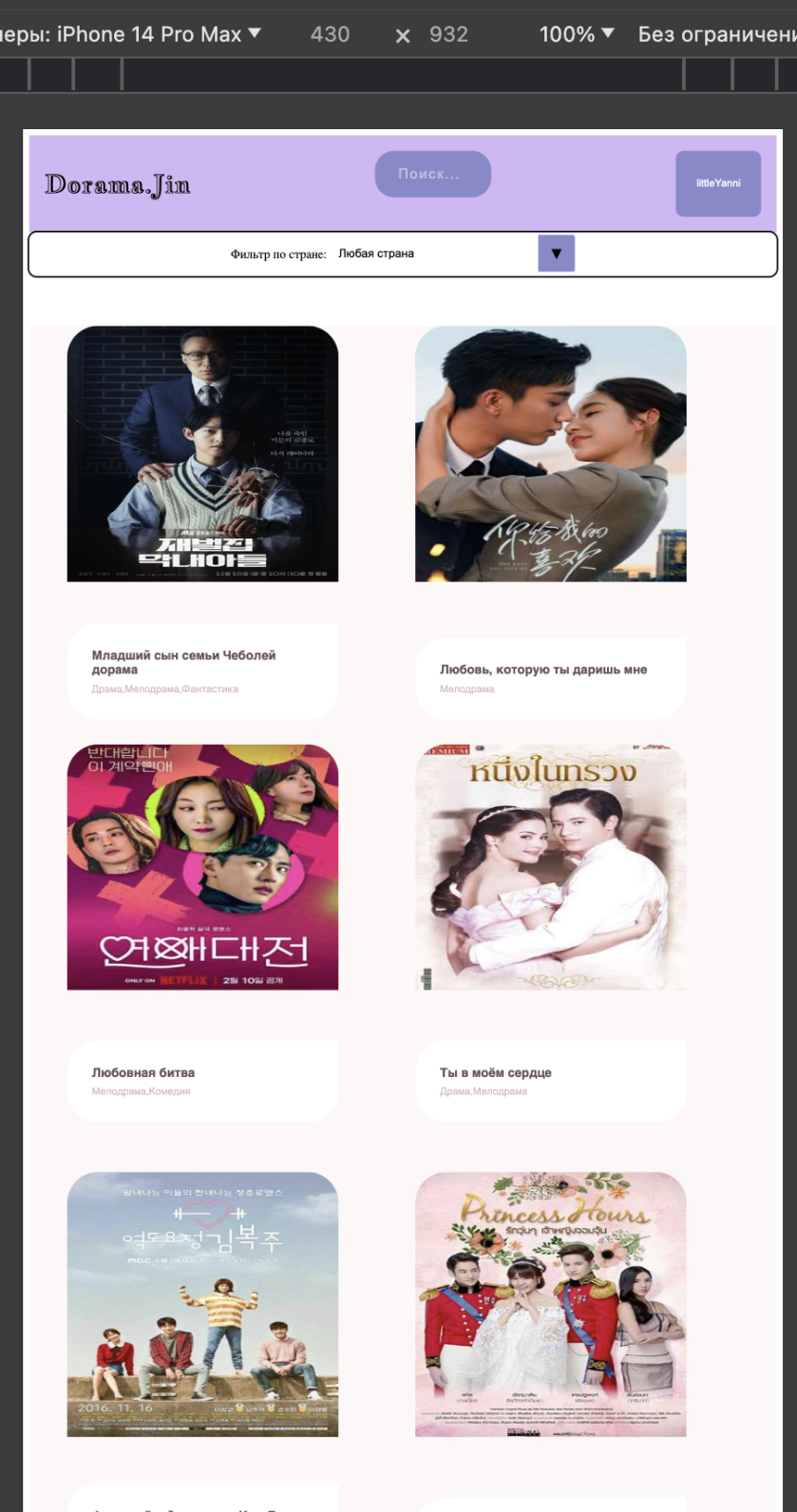


Рисунок 4.9- Адаптивность для телефона

**4.3 Выводы**

В данном разделе была рассмотрена кроссбраузерность веб-сайта. Мы протестировали отображение страниц и исправили все недочеты. После тестирование страница работает корректно в любом браузере. Стоит уделять достаточное внимание тестированию и обеспечению кроссбраузерности веб-сайта. Это поможет предоставить приятный и безпроблемный пользовательский опыт для всех пользователей, независимо от того, какой браузер они используют.

**Заключение**

В ходе выполнения данной курсовой работы был разработан и создан веб-сайт, посвященный дорамам. Целью данного проекта было предоставить пользователям удобный и информативный ресурс, который позволяет найти и ознакомиться с различными дорамами, а также обеспечить автоматическую генерацию контента на страницах и добавление обработчиков для удобства пользователей.

Для создания качественного сайта необходимо уделить внимание не только контенту, но и дизайну, функциональности и удобству использования для пользователей. Важно также обеспечить адаптивность сайта и оптимизацию под различные устройства, чтобы пользователи могли просматривать новости на любом устройстве.

Для реализации макета веб-страницы были исследованы другие сайты спортивной тематики.

Для удобного написания кода веб-страницы была использована среда разработки VS Code, в которой были установлены плагины для облегченного пользования.

Перед написанием кода, были изучены новые стандарты языков разметки и программирования. Были прочтены документации по использованию технологий, которые будут использоваться на этапе написания сайта.

После написания всего веб-сайта следовало его тестирование, в ходе которого были выявлены некоторые недочеты, которые были устранены в скором времени.

Реализация важных элементов страницы либо же скриптов были представлены в приложениях, которые дают общую информацию о проекте.

Если подытожить, то для выполнения всех поставленных задач на этот курсовой проект были использованы как теоретические знания, так и практические навыки разработки веб-сайтов.

Список использованных источников

1 Аналог «Doramy.club» [Электронный ресурс]. – Режим доступа: https://doramy.club/

2 Приложение для разработки макетов и прототипов веб-страниц «Figma» [Электронный ресурс]. – Режим доступа: https://www.figma.com/

3 Справочник по HTML [Электронный ресурс]. – Режим доступа: https://htmlbook.ru/html

4 Документация по SCSS [Электронный ресурс]. – Режим доступа: https://sass-scss.ru/

5 Документация по HTML/CSS/JavaScript [Электронный ресурс]. – Режим доступа: https://developer.mozilla.org/

6 Репозиторий проекта на GitHub [Электронный ресурс]. — Режим доступа: https://github.com/littleYanni/Coursework

7 Киселенко Н. П. HTML. Самое необходимое – Санкт-Петербург: БХВ-Петербург, 2008. – 352 с.