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 «Железнодорожного вокзала»

**Исполнитель**

студент(ка) 1 курса 6 группы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д. С. Савицкий

подпись, дата

**Руководитель**

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

должность, ученая 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

Содержание

[Введение 4](#_Toc199432481)

[1 Постановка задачи 5](#_Toc199432482)

[1.1 Обзор аналогичных решений 5](#_Toc199432483)

[1.2 Техническое задание 7](#_Toc199432484)

[1.3. Выбор средств реализации программного продукта 8](#_Toc199432485)

[1.4. Вывод 8](#_Toc199432486)

[2. Проектирование страниц веб-сайта 9](#_Toc199432487)

[2.1. Выбор способа верстки 9](#_Toc199432488)

[2.2. Выбор стилевого оформления 9](#_Toc199432489)

[2.3. Выбор шрифтового оформления 9](#_Toc199432490)

[2.4. Разработка логотипа 9](#_Toc199432491)

[2.5. Разработка пользовательских элементов 10](#_Toc199432492)

[2.6. Разработка спецэффектов 10](#_Toc199432493)

[2.7. Выводы 11](#_Toc199432494)

[3. Реализация структуры веб-сайта 12](#_Toc199432495)

[3.1. Структура HTML-документа 12](#_Toc199432496)

[3.2. Добавление таблиц стилей SCSS и CSS 12](#_Toc199432497)

[3.3. Использование стандартов XML (SVG) 14](#_Toc199432498)

[3.4 Управление элементами DOM 14](#_Toc199432499)

[3.4. Выводы 16](#_Toc199432500)

[4. Тестирование веб-сайта 17](#_Toc199432501)

[4.1. Адаптивный дизайн веб-сайта 17](#_Toc199432502)

[4.2. Кроссбраузерность веб-сайта 18](#_Toc199432503)

[4.3. Руководство пользователя 20](#_Toc199432504)

[4.4. Выводы 20](#_Toc199432505)

[Заключение 21](#_Toc199432506)

[Список использованных источников 22](#_Toc199432507)

Введение

С развитием цифровых технологий интернет стал неотъемлемой частью жизни современного человека. Сегодня веб-сайты играют важнейшую роль в укреплении позиций компаний на рынке, привлекая новых клиентов и повышая узнаваемость бренда. Эффективный веб-сайт — это инструмент, который позволяет пользователю легко находить необходимую информацию, продукцию или веб-сайта требует тщательного подхода к разработке дизайна, функционала и адаптивности ресурса для различных устройств. Именно эти аспекты определяют успех веб-ресурса и его способность удовлетворить потребности целевой аудитории.

Целью данного проекта является разработка веб-сайта для железнодорожного вокзала, реализующего просмотр онлайн-табло, покупку и заказ билетов, просмотр возможных маршрутов, получение актуальной информации о вокзале.

Для выполнения проекта необходимо решить следующие задачи:

• изучить технологии и инструменты разработки веб-сайтов;

• разработать уникальный дизайн и макет страниц сайта;

• построить структуру и функциональность ресурса;

• наполнить сайт контентом, соответствующим тематике;

• оптимизировать веб-сайт для работы на разных устройствах;

• протестировать сайт на предмет функциональности и удобства использования.

Целевая аудитория проекта — это пассажиры, заказывающие билеты и туристы, интересующиеся возможными маршрутами.

1 Постановка задачи

* 1. Обзор аналогичных решений

Перед разработкой собственной системы следует исследовать уже готовые программные продукты и выявить, что по мнению других разработчиков является наиболее важным для удобства пользователя. Подчерпнуть достоинства для использования в своем проекте и выявить недостатки чтобы в дальнейшем их избежать. Что поможет в создании удобного для пользователей сайта и поднимет конкурентоспособность в сравнении с другими сайтами. Обеспечит схожесть для привычности пользователю, но с сохранением индивидуальности. Такой подход позволит создать эффективный и удобный в использовании проект

1.1.1. Аналог «Расписание поездов БЖД»

Веб-сайт выглядит стильно, цветовая гамма довольно приглушённая. Сайт предоставляет широкий перечень услуг по поиску и покупке железнодорожных билетов. Главная страница представлена на рисунке 1.1.

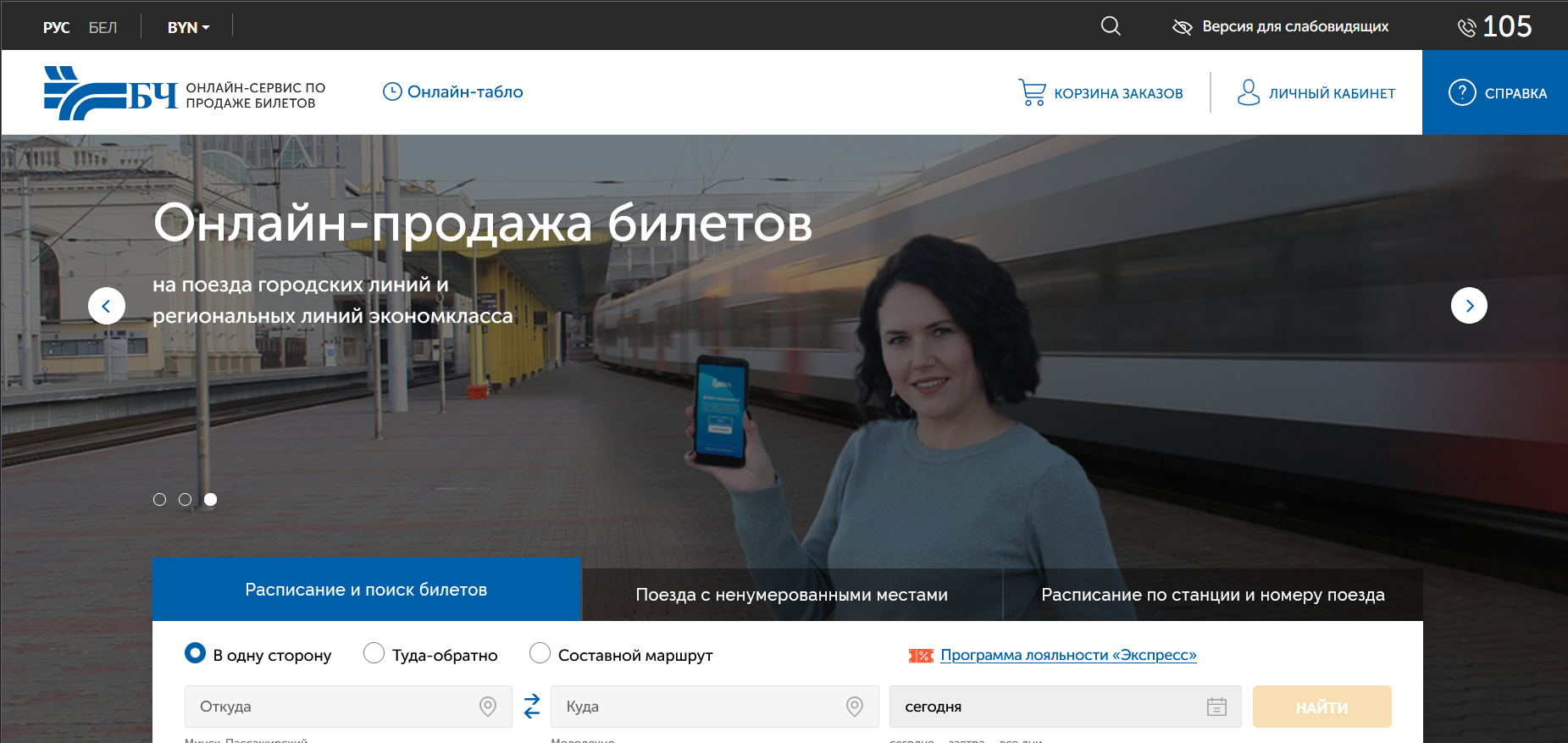


Рисунок 1.1 Главная страница аналога «Расписание поездов БЖД»

Есть возможность авторизации для облегчения последующего заказа билетов. Можно выбрать один из нескольких языков, имеется версия для слабовидящих, что свидетельствует о клиентоориентированности.

На веб-сайте предоставляется актуальная информация о возможных маршрутах, изменениях в следовании железнодорожных поездов. Есть раздел с новостями, где пользователи могут узнать про все интересующие темы. Также сайт содержит раздел онлайн табло, где удобно размещена вся необходимая информация о пребывающих и убывающих поездах, можно выбрать интересующую станцию.

На сайте реализована удобная наглядная система выбора билетов, чтобы пользователю было легко найти нужное предложение в желаемой ценовой категории. Пользователь имеет возможность выбрать место, узнать его расположение в вагоне, в зависимости от предоставленного состава, увидеть направление следования вагона, которое удобно показано стрелкой. Также можно получить информацию о том, какие места ещё свободны на текущей станции. На рисунке 1.2 показана страница выбора желаемых билетов.

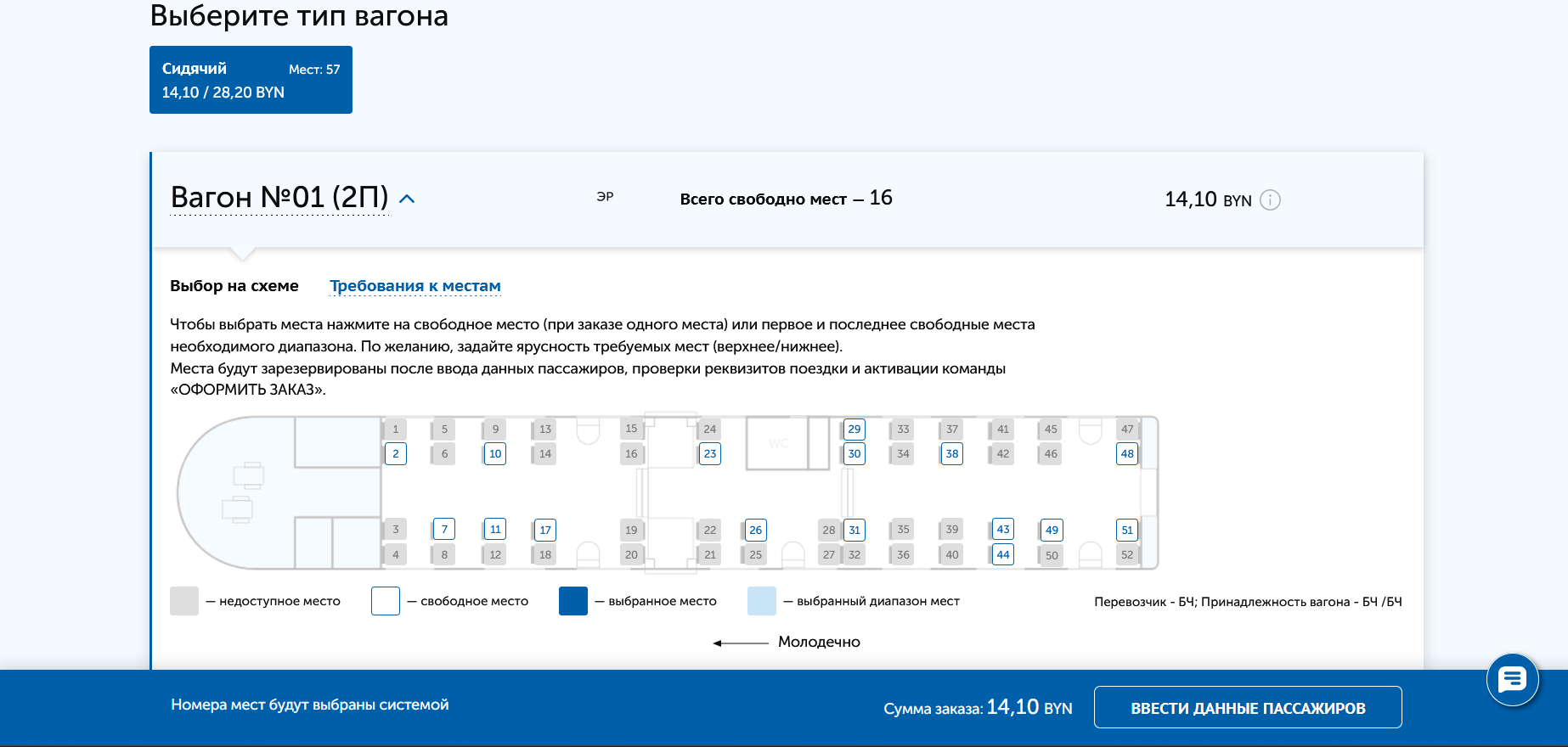


Рисунок 1.2 Страница выбора билетов аналога «Расписание поездов БЖД»

1.1.2 Аналог «РЖД»

Сайт имеет необычный дизайн, интуитивно понятный пользователю. Выбрана интересная цветовая гамма, отличающая веб-сайт от других аналогов. Реализован удобный подбор билетов по месту назначения, есть версия для слабовидящих и реализация выбора разных языков. Но сайт в некоторых местах слишком яркий, что может отталкивать некоторых потенциальных клиентов. На рисунке 1.3 представлена главная страница веб-сайта.

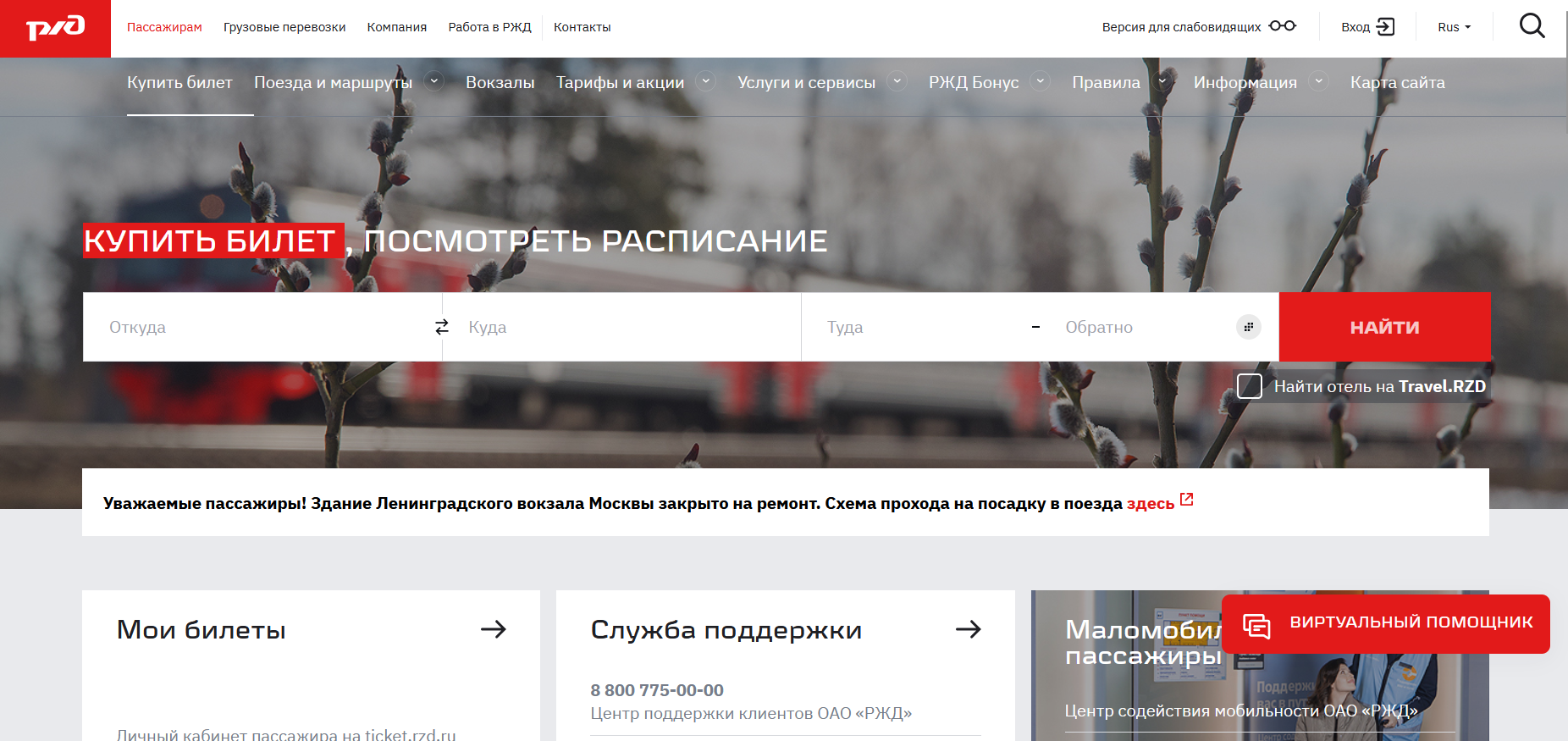


Рисунок 1.3 Главная страница аналога «РЖД»

Есть раздел «Карта сайта», где удобно размещён перечень всех операций и разделов, предоставляемых на веб-сайте. Данный раздел является удобным и лаконичным решением, но в некоторых случаях он неактуален, так как веб сайт и так является интуитивно понятным и не нуждается в дальнейшем сопровождении. На рисунке 1.4 изображена карта веб-сайта. Все разделы разделены в иерархичном порядке и разделены по категориям. Все цвета на данной веб-странице приглушены, поэтому по ней удобно ориентироваться. Но яркая красная кнопка в правом углу слишком яркая, поэтому она привлекает излишнее внимание.

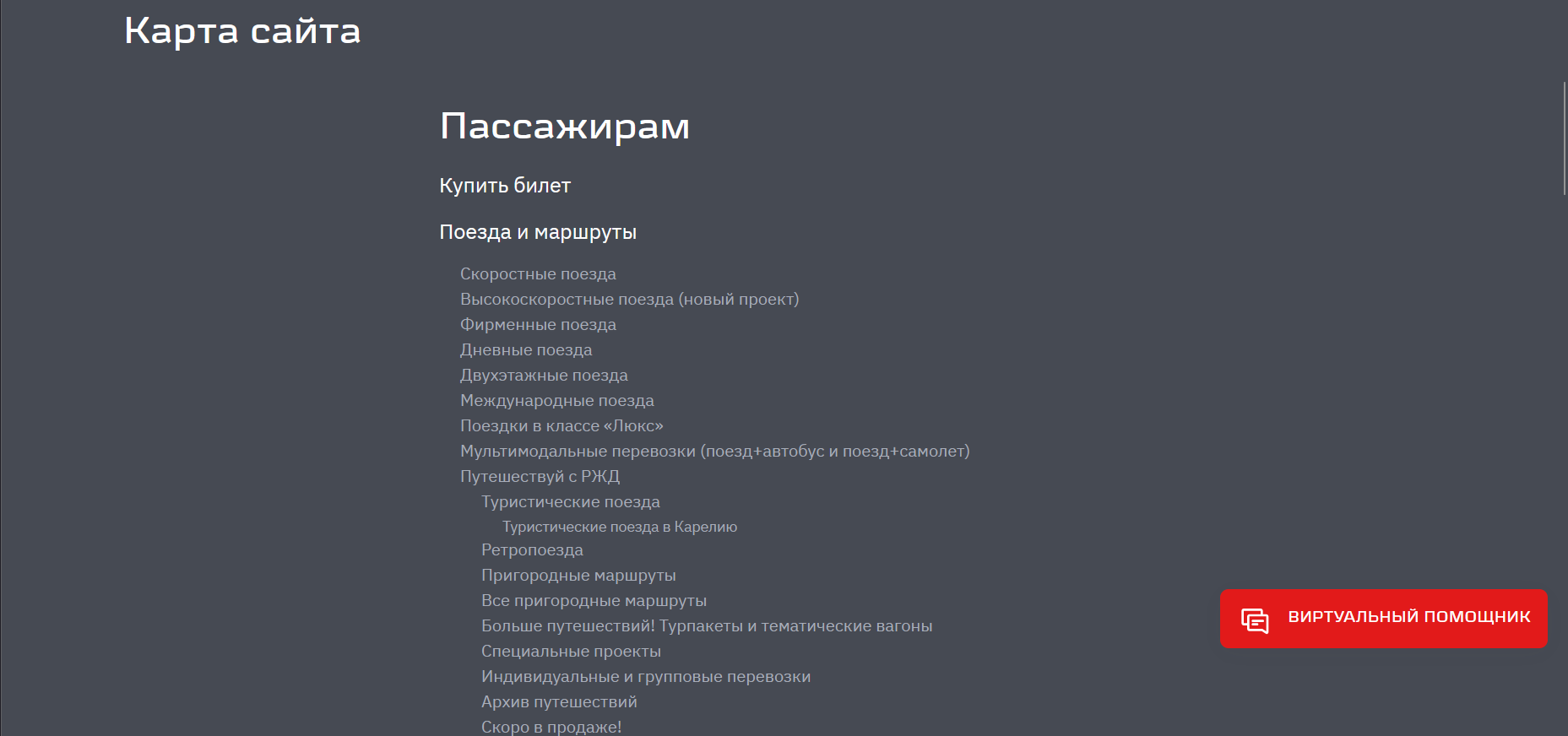


Рисунок 1.4 Окно раздела «Карта сайта» аналога «РЖД»

1.2 Техническое задание

Данная курсовая работа является проектом по созданию веб-сайта для железнодорожного вокзала «SpeedWagon.by». На сайте будет размещена информация о показаниях к посещению комплекса, каталог возможных для заказа комнат, узнать подробности предоставляемых условий, показаны фото для ознакомления с территорией санатория.

Целями сайта являются:

-ознакомить пользователя с возможными показаниями для оздоровления в комплексе;

-дать возможность пользователю посмотреть территорию санаторно-оздоровительного комплекса

-ознакомить пользователя с доступными комнатами для проживания;

-дать возможность подробно узнать о предоставляемых условиях в комнатах

Веб-сайт предназначен и нацелен на тех пользователей, которые хотят оздоровиться и хорошо провести время в санаторно-оздоровительном комплексе.

Исполнитель обязуется выполнить минимальное наполнение сайта, которое позволит начать эксплуатацию сайта заказчиком. В обязанности исполнителя входит: создание макета и разработка дизайна с помощью FIGMA, разработка веб-сайта с использованием HTML, создание внешнего офaормления с помощью CSS и SASS, разработка контента веб-сайта на XML и создание адаптивной верстки сайта для всех типов устройств.

Данный программный продукт реализуется с помощью: HTML, CSS, SASS, JavaScript, XML.

При разработке будут использоваться приложения: FIGMA, Visual Studio Code.

Сайт размещен в репозитории GitHub, по следующей ссылке <https://github.com/Lozuno/KYAR_KursProject>

1.3. Выбор средств реализации программного продукта

Курсовая работа выполняется в редакторе кода Visual Studio Code.

Для разработки веб-страниц применялись следующие технологии: HTML5 для структурирования контента, CSS3 и его препроцессор SCSS для стилизации, а также JavaScript для реализации интерактивных элементов. HTML5 (HyperText Markup Language) служит основой для семантической структуры веб-страниц

Визуальное оформление реализовано с использованием современных возможностей CSS3, включая медиа-запросы, обеспечивающие адаптивное отображение на устройствах с различными разрешениями экранов. Препроцессор SCSS был задействован для упрощения работы со стилями и поддержания чистоты кода.

Для добавления интерактивности и динамического поведения элементов применялся JavaScript. В проекте также использованы векторные изображения в формате SVG, что позволяет сохранять качество графики на любых разрешениях.

На этапе проектирования интерфейса использовалось приложение Figma, позволяющее создать детализированные макеты. Финальная реализация объединила семантическую разметку HTML5, современные стили CSS3/SCSS и функциональность JavaScript для создания полноценного адаптивного веб-интерфейса.

1.4. Вывод

Постановка задачи включает анализ существующих сайтов железнодорожных вокзалов, а также определение технических требований. Рассмотрение сайта Расписание поездов БЖД и РЖД.

В рамках технического задания определён ряд требований к функционалу: интуитивно понятная навигация меню и доступность контактной информации, эффективное и простое бронирование мест, адаптивный дизайн для всех типов устройств начиная с десктоп версий заканчивая мобильными устройствами.

Для реализации проекта были выбраны современные веб-технологии, гарантирующие надежную работу и комфортное взаимодействие с пользователями.

Такой технологический стек позволяет создать удобный и функциональный веб-ресурс, отвечающий всем требованиям современного пользователя. Сай предоставляет быстрый доступ к информации, возможность онлайн-бронирования билетов и стабильную работу в любом браузере на различных устройствах, что улучшает пользовательский опыт и удовлетворяет ключевые потребности посетителей.

2. Проектирование страниц веб-сайта

2.1. Выбор способа верстки

При создании веб-страниц использованы технологии flexbox и CSS Grid для гибкого позиционирования элементов, а также медиазапросы для адаптивного отображения на разных устройствах. Эти подходы позволяют легко управлять расположением компонентов при изменении размеров экрана, обеспечивая удобный и интуитивно понятный интерфейс. Все разработанные макеты изначально учитывают требования адаптивного дизайна.

2.2. Выбор стилевого оформления

Курсовой проект выполнен в стиле современного минимализма. Простой дизайн c яркими элементами делает его запоминающимся и акцентирует внимание пользователя на важных элементах.

Основными цветами на веб-страницах выбраны светлый голубовато-белый (#f8f9fd) и королевский пурпур (#4a11a8). Для акцентов на веб-сайте использован радикальный красный (#ff3366). ). Для оформления подвала и бургер-меню веб-сайта выбран кобальтово-синий (#191d3a). Такое цветовое сочетание позволяет сделать яркий и необычный веб-проект.

Изображение выглядит как белый, дизайн

Контент, сгенерированный ИИ, может содержать ошибки. Изображение выглядит как фиолетовый, Сирень, Фиолетовый, синий

Контент, сгенерированный ИИ, может содержать ошибки. Изображение выглядит как красный, розовый

Контент, сгенерированный ИИ, может содержать ошибки. ![Изображение выглядит как синий, Цвет электрик, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.](data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAGkAAABfCAYAAADroemKAAAAAXNSR0IArs4c6QAAAARnQU1BAACxjwv8YQUAAAAJcEhZcwAADsMAAA7DAcdvqGQAAAD7SURBVHhe7dExEYAADMDAggsGjODfFyyIaO7+lwjIcd3PO6x2/mUxkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAkwJMCjApwKQAk9ab+QCaPgIts13iWAAAAABJRU5ErkJggg==)

#f8f9fd #4a11a8 #ff3366 #191d3a

Рисунок 2.1 – Выбранные цвета для оформления веб-страницы

2.3. Выбор шрифтового оформления

Для курсового проекта был выбран минималистичный шрифт Montserrat в соответствии с рисунком 2.2. Размер шрифта адаптируется под ширину экрана.

Изображение выглядит как Шрифт, белый, текст, типография

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2.2 – Демонстрация шрифта Montserrat

Montserrat — геометрический гротеск с мягкими округлыми формами, минимальным контрастом и высокой читаемостью. Открытые пропорции и крупные строчные буквы делают его универсальным для заголовков и текста. Montserrat отличается высокой читаемостью, универсальностью и эстетичной простотой, что делает его популярным выбором для веб-дизайна, брендинга и полиграфии. Доступный в различных начертаниях. Шрифт широко используется в коммерческих и личных проектах, сохраняя при этом элегантность и функциональность.

2.4. Разработка логотипа

Логотип веб-сайта представлен на рисунке 2.3. Разработано в приложении Adobe Illustrator, является минималистичным и. Выбор данного логотипа оправдан. Логотип железнодорожного вокзала оформлен в минималистичном и современном стиле. Осной фон выполнен в насыщенном фиолетовом цвете, создавая контраст и придавая логотипу яркость и запоминаемость.

Изображение выглядит как Шрифт, Графика, Цвет электрик, графический дизайн

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2.3 – Логотип веб-сайта

Выбор данного логотипа оправдан. Логотип железнодорожного вокзала оформлен в минималистичном и современном стиле. Осной фон выполнен в насыщенном фиолетовом цвете, создавая визуальный контраст и придавая логотипу яркость и запоминаемость. Выбор цветов подобран на цветовом круге.

Текстовое оформление включает две части: «SpeedWagon.by». Слово «SpeedWagon» выполнено белым шрифтом, а домен «.by» выделен красным цветом, что акцентирует внимание на национальной принадлежности бренда. Такой выбор цветов делает логотип легко читаемым и визуально привлекательным.

Дизайн логотипа прост и лаконичен, что делает его удобным для использования на различных носителях, будь то вывески, билеты или цифровые платформы. Контраст цветов и четкие линии создают ощущение надежности и скорости — качеств, ассоциируемых с железнодорожными перевозками.

2.5. Разработка пользовательских элементов

В проекте представлены такие элементы пользовательского интерфейса как шапка, подвал, бургер-меню. Шапка сайта представлена на рисунке 2.4. Данный раздел значительно упрощает навигацию на сайте, позволяя быстро переходить на нужный раздел.



Рисунок 2.4 – Шапка сайта

В шапке мобильной версии реализовано бургер-меню со всеми разделами веб-сайта. Шапка мобильной версии представлена на рисунке 2.5

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2.5 – Бургер-меню для мобильной версии

Подвал сайта – блок сайта, размещённый на рисунке 2.6, в котором размещены ссылки на все важные ресурсы. Пользователь с лёгкостью может перейти на интересующий раздел и узнать всю необходимую информацию.

Изображение выглядит как текст, снимок экрана, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2.6 – Подвал веб-сайта

2.6. Разработка спецэффектов

Эффекты при наведении курсора на карточку маршрута представлены на рисунке 2.7. Иконки маршрутов немного поднимаются, а полоска, находящаяся за буквами на сером фоне, увеличивается в длине. Такое решение позволяет сделать интерфейс динамичным, увидеть наглядно выбранный пользователем маршрут.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Контент, сгенерированный ИИ, может содержать ошибки. Изображение выглядит как текст, снимок экрана, Шрифт, Бренд

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2.7 – Карточки маршрутов

На рисунке 2.8 представлены эффекты при выборе маршрута на странице покупки билета. На веб странице появляется новое диалоговое окно, которое показывает выбранный пользователем маршрут следования, время и дату поездки Они делают веб-сайт отзывчивым и клиентоориентированным. Так пользователь может легко сохранить все свои данные и не забыть их

Изображение выглядит как текст, снимок экрана, программное обеспечение, мультимедиа

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2.8 – Отправка сообщения после заказа билета

2.7. Выводы

На текущем этапе разработки уже созданы логотип, прототипы и макеты веб-сайта, что существенно ускорило процесс его верстки. Были проработаны ключевые элементы страниц, такие как навигационное меню, логотип, основной контент, расположение изображений и подвал веб-сайта, а также определен стиль оформления, включая цветовую гамму, шрифты и анимации, придающие сайту динамичность и визуальную привлекательность.

Макеты обеспечили четкое представление структуры сайта, позволив выстроить логическую и визуальную иерархию контента, создать удобную навигацию и интуитивно понятный пользовательский путь. Особое внимание уделялось стилевому оформлению: была подобрана сбалансированная цветовая палитра, соответствующая целям проекта и ожиданиям аудитории. Интерфейсные элементы, такие как кнопки, формы, иконки и меню, были адаптированы для удобства взаимодействия, а динамические эффекты добавили сайту интерактивности.

3. Реализация структуры веб-сайта

3.1. Структура HTML-документа

Проект реализован с четкой взаимосвязью между страницами, что обеспечивает удобную навигацию для пользователей и упрощает процесс разработки. Такая архитектура позволяет быстро перемещаться между разделами, а также легко вносить изменения и расширять функционал.

HTML-документ создан в соответствии со стандартами HTML5, что гарантирует его корректное отображение в современных браузерах. В листинге 3.1 представлено содержимое тега head.

|  |
| --- |
| <!DOCTYPE html>  <html lang="ru">      <head>          <meta charset="UTF-8" />          <meta name="viewport" content="width=device-width, initial-scale=1.0" />          <meta              name="description              content="SpeedWagon.by - железнодорожные перевозки по Беларуси. Современные поезда, удобное расписание, выгодные цены."          />          <meta              name="keywords"              content="железная дорога, билеты, поезд, Беларусь, SpeedWagon, ж/д билеты, расписание поездов"          />          <meta name="theme-color" content="#0A192F" />          <title>SpeedWagon.by - Железнодорожные перевозки</title>          <link rel="stylesheet" href="css/index.css" />          <script src="js/main.js" defer=""></script>      </head> |

Рисунок 3.1 – Декларация и тег head

Тег <!DOCTYPE html> является обязательным объявлением в начале HTML-документа. Он указывает браузеру, что страница соответствует стандарту HTML5 и должна обрабатываться в строгом режиме отображения. Это гарантирует корректную интерпретацию кода во всех современных браузерах.

В разделе <head> содержатся важные метаданные документа. Метатег <meta charset="UTF-8"> определяет кодировку UTF-8 для правильного отображения текста. Параметры <meta name="viewport"> настраивают адаптивное отображение на мобильных устройствах. Тег <title> устанавливает заголовок страницы, который отображается во вкладке браузера.

Для подключения стилей используются теги <link>. Такой подход соответствует принципам разделения структуры и оформления, что является общепринятой практикой в современной веб-разработке. Отдельное хранение CSS в внешних файлах упрощает поддержку и модификацию дизайна.

3.2. Добавление таблиц стилей SCSS и CSS

В процессе создания веб-сайта применялся исключительно внешний метод подключения CSS-стилей. Данное решение обеспечивает централизованное управление оформлением, облегчая процесс модификации дизайна и повышая быстродействие сайта благодаря кешированию стилевых файлов.

При создании стилей применялся препроцессор SCSS, который предоставил следующие функциональные возможности: использование вложенных правил, переменных для управления цветовой палитрой и миксинов для повторяющихся фрагментов вёрстки. Конкретный пример реализации SCSS-структуры приведён в листинге 3.2.

|  |
| --- |
| <head>      …      <link *rel*="stylesheet" *href*="css/index.css" />  </head> |

Листинг 3.2 – Пример внешнего подключения стилевого файла

В процессе разработки стилей для веб-страницы основное внимание было уделено использованию классовых селекторов. Такой подход обеспечивает прозрачную структуру кода, повышает его читаемость и обеспечивает гибкость в управлении визуальным представлением элементов. Это позволяет создавать модульный и легко масштабируемый код, что значительно упрощает дальнейшее сопровождение и развитие проекта.

Пример реализации стилей на SCSS представлен в листинге 3.3, соответствующий CSS-код показан в листинге 3.4. Подобная организация стилевых правил обеспечивает простоту внесения изменений и добавления новых стилевых решений при сохранении четкой структуры и порядка в кодовой базе.

|  |
| --- |
| .footer-column {      p {          color: rgba(255, 255, 255, 0.7);          margin-bottom: 20px;          line-height: 1.6; }      h3 {          font-size: 1.2rem;          font-weight: 700;          margin-bottom: 25px;          position: relative;          &::after {              content: "";              position: absolute;              bottom: -10px;              left: 0;              width: 40px;              height: 3px;              background-color: var(--secondary);              border-radius: 2px;          } }} |

Листинг 3.3 – Пример таблицы стилей SCSS

3.3. Использование стандартов XML (SVG)

Векторная графика, включая логотипы и иконки, чаще сохраняется в формате SVG. Данный формат обеспечивает высокое качество изображения при любом масштабировании, поскольку основан на математических расчетах, а не на пикселях.

Кроме того, SVG отличается компактным размером файлов, что особенно важно для оптимизации загрузки веб-страниц. Еще одним преимуществом этого формата является возможность создания анимации с использованием CSS и JavaScript, что расширяет его функциональность в веб-разработке. Пример использования svg показан в листинге 3.3

|  |
| --- |
| <a *href*="#" *class*="app-button">              <svg  *xmlns*="http://www.w3.org/2000/svg"  *width*="20"  *height*="20"  *viewBox*="0 0 24 24"  *fill*="none"  *stroke*="currentColor"  *stroke-width*="2"  *stroke-linecap*="round"  *stroke-linejoin*="round"                >                <path *d*="M3 9l9-7 9 7v11a2 2 0 0 1-2 2H5a2 2 0 0 1-2-2z"                ></path>                <polyline  *points*="9 22 9 12 15 12 15 22"                ></polyline>                </svg>                Google Play  </a> |

Листинг 3.4 – Пример использования SVG на главной странице

Для хранения информации о железнодорожных маршрутах было принято решение использовать XML. Пример использования описан в листинге 3.5. Полный листингXML представлен в приложении Д.

|  |
| --- |
| <?xml *version*="1.0" *encoding*="UTF-8"?> |

Листинг 3.5 – Пример использования стандартов XML

Данные о маршрутах удобно хранятся отдельно в XML-документе. Данная строка описывает версию документа и кодировку для обеспечения корректного отображения веб-страницы в браузере.

Данный XML-документ представляет структурированную базу данных железнодорожных маршрутов, содержащую информацию о 12 направлениях между крупными городами Беларуси, включая Минск, Брест, Гомель, Гродно, Витебск и Могилев. Каждый маршрут детально описан с указанием пунктов отправления и назначения, времени в пути, расстояния, стоимости проезда, номера и типа поезда, а также расписания отправления и прибытия. Дополнительно для каждого рейса приведен перечень доступных услуг (Wi-Fi, питание, розетки, перевозка велосипедов) и подробная информация о классах обслуживания с указанием количества свободных мест и тарифов. Документ включает как прямые маршруты из Минска, так и обратные направления, а также межрегиональные сообщения, что делает его полноценной основой для систем автоматизированного бронирования железнодорожных билетов или интеграции в транспортные приложения.

XML-формат удобен для хранения данных, так как XML не зависит от программного обеспечения или операционной системы, что делает его переносимым между платформами. Также XML-формат позволяет создавать собственные теги и атрибуты, что дает возможность расширять его для конкретных нужд проекта.

3.4 Управление элементами DOM

Для интерактивности и удобства работы с интерфейсом был использован JavaScript. На всех страницах имеется механизм управления экраном загрузки веб-страницы, который плавно скрывает его и плавно восстанавливает стандартное поведение страницы. Механизм предоставлен в листинге 3.6.

|  |
| --- |
| function handleLoadingScreen() {          const loadingScreen = document.querySelector(".loading-screen");          if (loadingScreen) {              setTimeout(() => {                  loadingScreen.classList.add("hidden");                  setTimeout(() => {                      document.body.style.overflow = "";                  }, 500);              }, 800)  ;          }      } |

Листинг 3.6 – Обработка экрана загрузки

Сначала находится элемент с классом «loading-screen». Если такой элемент существует, через 800 миллисекунд к нему добавляется класс «hidden», который скрывает экран загрузки. Затем, после дополнительной задержки в 500 миллисекунд, снимается ограничение прокрутки страницы (если оно было установлено ранее). В результате экран загрузки плавно исчезает, и страница становится полностью доступной для пользователя.

При наведении курсора на карточки с маршрутами происходят изменения состояния линий внутри карточек. Данные изменения показаны в листинге 3.7.

Этот код создает интерактивный эффект для карточек маршрутов на сайте. Когда пользователь наводит курсор на карточку, декоративная линия внутри нее плавно расширяется с 70% до 90% ширины и становится более яркой (непрозрачность увеличивается с 0.4 до 0.7). При убирании курсора линия возвращается к исходным размерам и прозрачности. Такой подход обеспечивает визуальную обратную связь, делая интерфейс более отзывчивым и привлекательным для пользователей. Эффект применяется ко всем карточкам маршрутов на странице.

|  |
| --- |
| function setupRouteCardEffects() {          const routeCards = document.querySelectorAll(".route-card");          const bookmarks = document.querySelectorAll(".route-bookmark");          routeCards.forEach((card) => {              card.addEventListener("mouseenter", () => {                  const decorativeLine = card.querySelector(                      ".route-decorative-line");                  if (decorativeLine) {                      decorativeLine.style.width = "90%";                      decorativeLine.style.opacity = "0.7";} });              card.addEventListener("mouseleave", () => {                  const decorativeLine = card.querySelector(                      ".route-decorative-line");                  if (decorativeLine) {                      decorativeLine.style.width = "70%";                      decorativeLine.style.opacity = "0.4";                  } }); }); |

Листинг 3.7 – Эффект для карточки

На выбора маршрутов организована функция поиска и сортировки маршрутов. Код данного решения представлен в листинге 3.8.

В начале функция проверяет наличие необходимых элементов на странице - поля ввода поиска (searchInput) и контейнера для отображения результатов (routesList). Затем она получает поисковый запрос, приведенный к нижнему регистру и очищенный от лишних пробелов.

Особенность работы функции заключается в ее способности обрабатывать маршруты, загруженные из XML-файла. Если массив allRoutes содержит данные, функция фильтрует маршруты, сравнивая поисковый запрос с названием маршрута в формате "ГородОтправления - ГородНазначения". Для этого маршрут преобразуется в строку, которая также приводится к нижнему регистру для обеспечения регистронезависимого поиска. В результате формируется массив filteredRoutes, содержащий только маршруты, которые соответствуют критериям поиска.

|  |
| --- |
| *function* searchRoutes() {          if (!searchInput || !routesList) return;  *const* searchTerm = searchInput.value.toLowerCase().trim();  *let* filteredRoutes;  *// Если маршруты загружены из XML*          if (allRoutes.length > 0) {              filteredRoutes = allRoutes.filter((*route*) => {  *const* routeTitle = `${route.from} - ${route.to}`.toLowerCase();                  return routeTitle.includes(searchTerm);              }  ); |

Листинг 3.8 – Сортировка маршрутов по запросу

3.4. Выводы

На данном этапе была создана структура веб-сайта, включающая главную информацию о вокзале, актуальных маршрутах, списком возможных билетов к заказу, и страницу с бронированием билетов.

В ходе разработки проекта были успешно реализованы и внедрены ключевые функциональные элементы веб-сайта. Среди них - система навигации, интерфейс бронирования билетов, а также различные компоненты пользовательского интерфейса и визуального оформления. На этапе верстки особое внимание уделялось продуманному размещению контентных блоков и их логической структуре, а также тщательной стилизации с помощью CSS для достижения привлекательного и гармоничного внешнего вида страниц.

В результате была создана полнофункциональная веб-платформа, полностью соответствующая требованиям технического задания и первоначальному прототипу. После завершающего этапа тестирования и устранения возможных недочетов сайт будет полностью готов к эксплуатации конечными пользователями.

4. Тестирование веб-сайта

4.1. Адаптивный дизайн веб-сайта

Для создания адаптивного веб-интерфейса были применены современные технологии CSS, flexbox, grid-верстку и медиазапросы. Такой подход обеспечивает корректное отображение сайта на устройствах с разными разрешениями экрана - от мобильных телефонов до настольных компьютеров, что подтверждается скриншотами на рисунках 4.1–4.4. Полные макеты представлены в Приложении Б.

Изображение выглядит как текст, транспортное средство, Наземный транспорт, общественный транспорт

Контент, сгенерированный ИИ, может содержать ошибки. Изображение выглядит как текст, снимок экрана, Шрифт, графический дизайн

Контент, сгенерированный ИИ, может содержать ошибки.

а б

а ─ десктопная версия, б ─ мобильная

Рисунок 4.1 – Главная страница веб-сайта

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Контент, сгенерированный ИИ, может содержать ошибки. Изображение выглядит как текст, снимок экрана, Шрифт, Бренд

Контент, сгенерированный ИИ, может содержать ошибки.

a б

а ─ десктопная версия, б ─ мобильная

Рисунок 4.2 – Страница поиска железнодорожных билетов

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

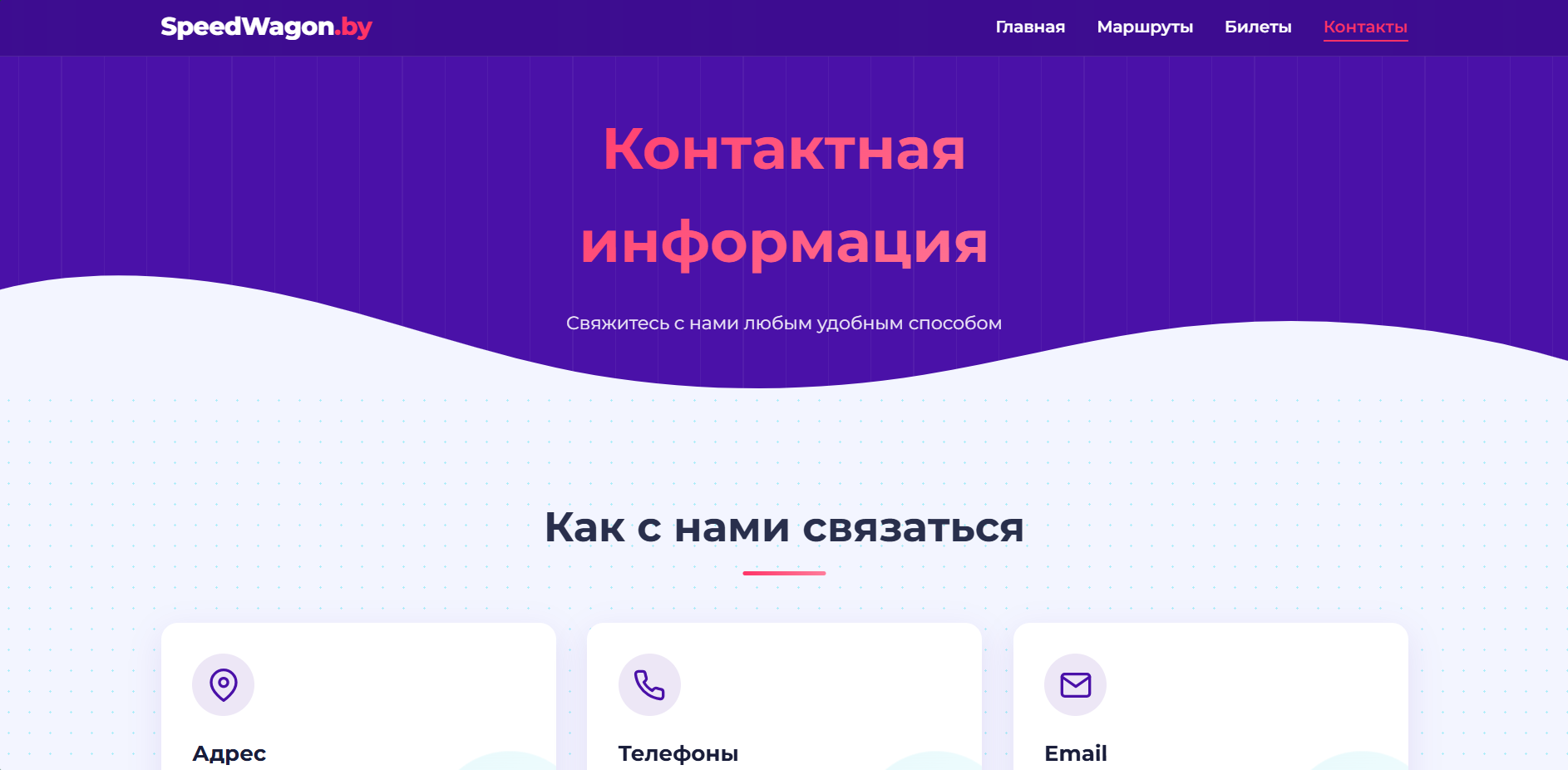
Контент, сгенерированный ИИ, может содержать ошибки. Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Контент, сгенерированный ИИ, может содержать ошибки.

a б

а ─ десктопная версия, б ─ мобильная

Рисунок 4.3 – Страница поиска и покупки билетов

 Изображение выглядит как текст, снимок экрана, Шрифт, Бренд

Контент, сгенерированный ИИ, может содержать ошибки.

a б

а ─ десктопная версия, б ─ мобильная

Рисунок 4.4 – Страница с контактной информацией

4.2. Кроссбраузерность веб-сайта

Кроссбраузерность веб-сайта означает его корректную работу и отображение в разных браузерах, включая Google Chrome, Mozilla Firefox, Safari, Microsoft Edge и другие. Это важный аспект веб-разработки, так как пользователи выбирают браузеры в зависимости от личных предпочтений, устройства, операционной системы.

Поддержка кроссбраузерности решает несколько ключевых задач:

Максимальный охват аудитории – поскольку пользователи заходят на сайт из разных браузеров, совместимость с ними позволяет привлечь больше посетителей и повысить их удовлетворённость.

Комфортный пользовательский опыт – единообразное отображение и стабильная работа сайта в любом браузере улучшают взаимодействие с контентом, укрепляя доверие к ресурсу. Снижение отказов – если сайт работает некорректно в каких-то браузерах, пользователи могут быстро покинуть его. Кроссбраузерность минимизирует такие риски, удерживая аудиторию.

Таким образом, обеспечение совместимости с разными браузерами делает веб-ресурс более универсальным, удобным и эффективным, независимо от платформы или устройства пользователя.

На рисунке 4.5 показано, как сайт выглядит в браузере «Microsoft Edge», а на  4.5 – отображение в «Google Chrome».

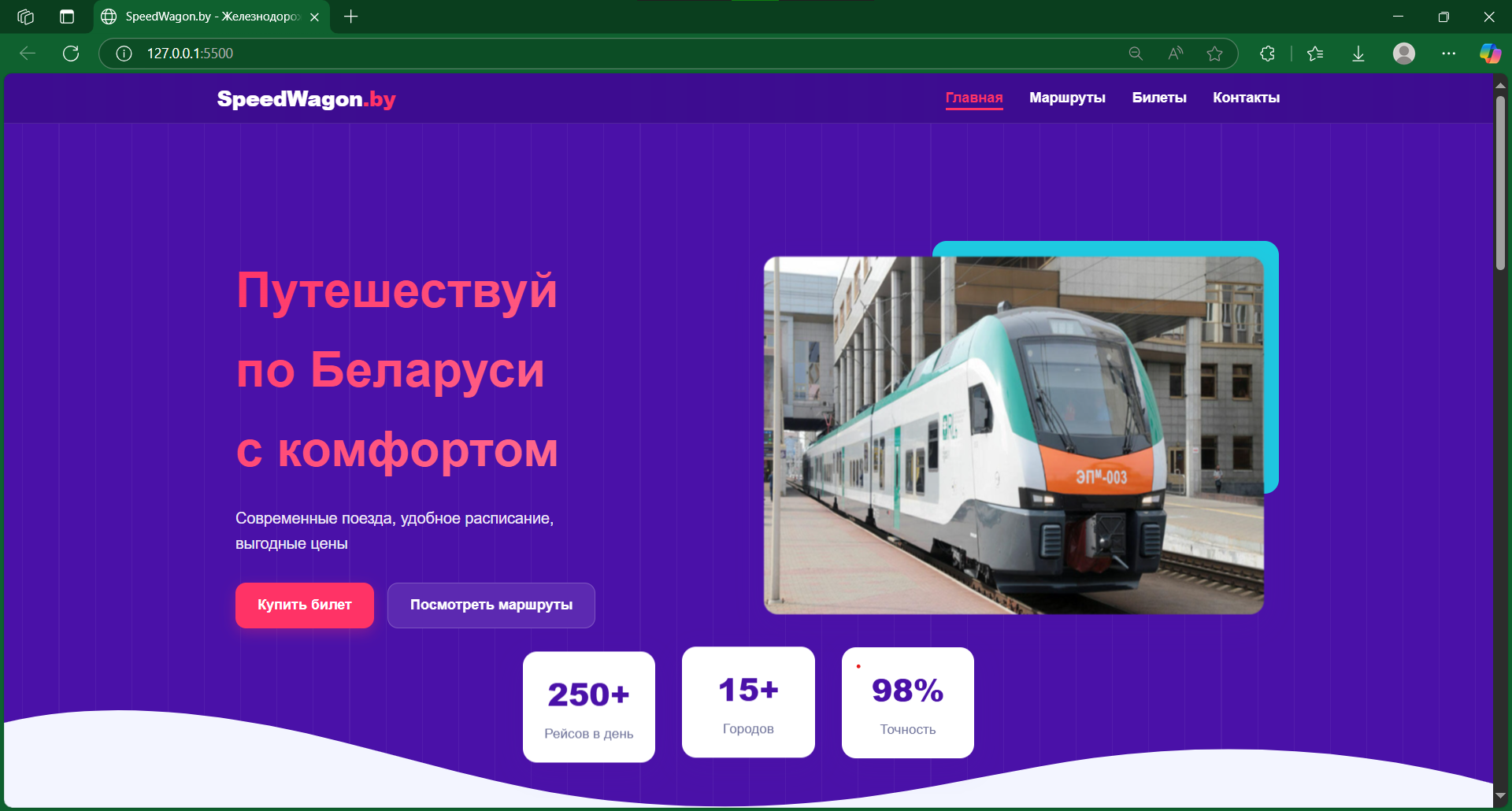


Рисунок 4.5 - Отображение страницы в браузере «Microsoft Edge»

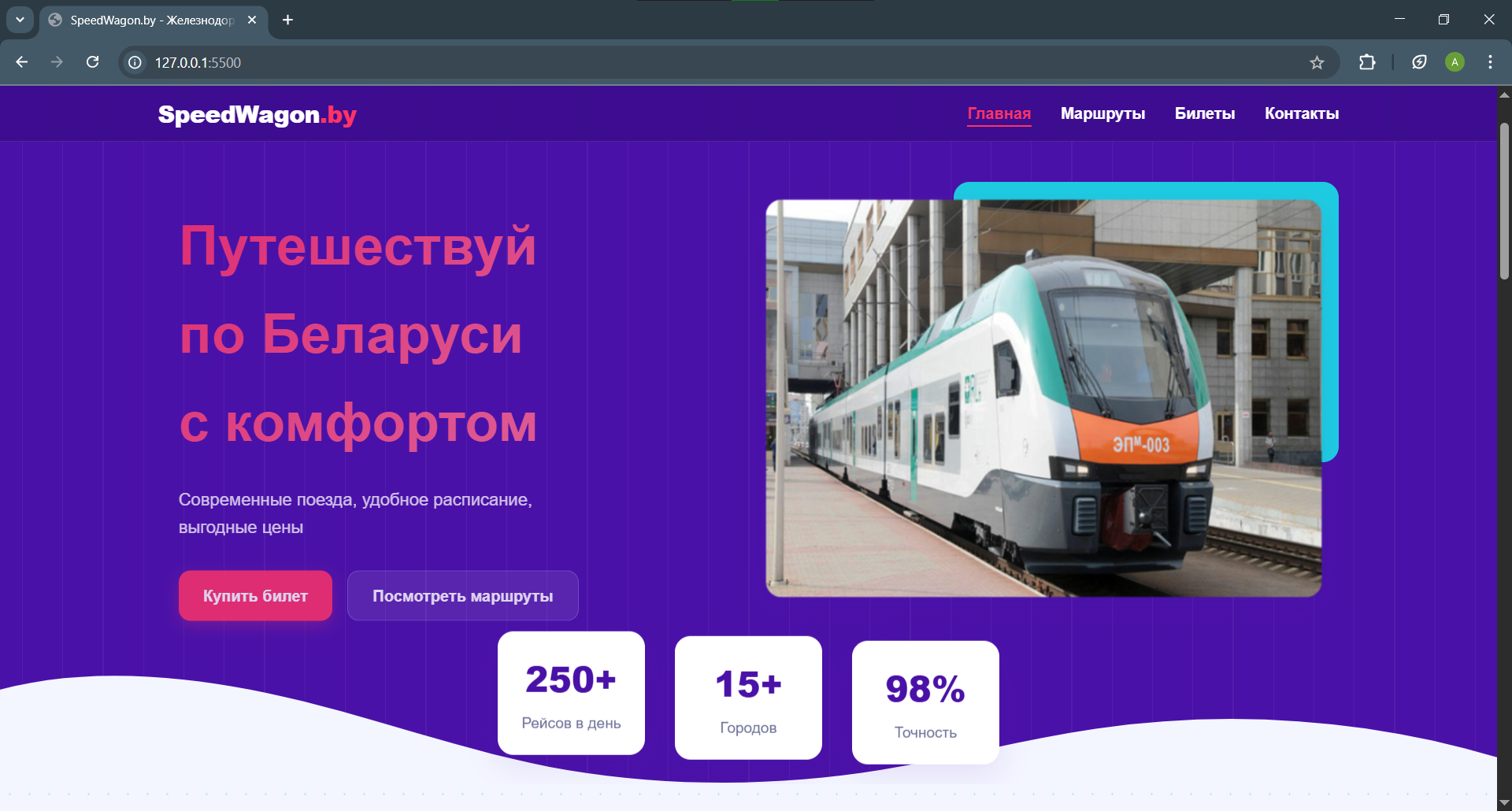


Рисунок 4.6 - Отображение страницы в браузере «Google Chrome»

4.3. Руководство пользователя

Главная страница является точкой входа на веб-сайт. Вверху страницы располагается горизонтальное меню в десктопной версии. В десктопной версии гамбургер-меню содержит ссылки на все остальные страницы. На главной странице располагаются карточки с популярными маршрутами, кликнув на которые автоматически происходит переход на страницу покупки билетов. Также на главной странице находится реклама мобильного приложения, секция с отзывами и подвал веб-сайта с полезными ссылками на интернет-ресурсы.

Для того, чтобы перейти на другую страницу нужно нажать на определенный пункт в навигационном меню. Доступ можно осуществить с любой страницы. С любой веб-страницы можно перейти к разделу заказа билетов, что является ключевой функции для веб-сайта данной тематики.

Структура веб-сайта представлена на рисунке 4.7.

Изображение выглядит как текст, снимок экрана, Прямоугольник, диаграмма

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 4.7 - Структура веб-сайта

4.4 Валидность

Валидность вёрстки означает полное соответствие HTML-кода официальным стандартам, разработанным консорциумом W3C. Соблюдение этих норм играет ключевую роль в обеспечении качества веб-ресурса, делая его более доступным для пользователей и улучшая показатели поисковой оптимизации. Проверка валидности помогает выявить критические недостатки, которые могут негативно сказаться на работе сайта и его восприятии посетителями.

В процессе тестирования часто обнаруживаются различные проблемы вёрстки. Речь идёт о распространённых ошибках, неправильная вложенность тегов, отсутствие обязательных атрибутов или использование устаревших элементов HTML. Особое внимание следует уделять уникальности идентификаторов, корректности объявления типа документа, а также согласованности стилевых правил.

Дополнительные сложности могут возникнуть при реализации адаптивного дизайна. Не менее важны вопросы доступности контента для пользователей с ограниченными возможностями и правильная структура заголовков. Все эти аспекты требуют тщательной проверки, так как напрямую влияют на юзабилити и техническую исправность веб-страницы.

Валидация веб-сайта показана на рисунке 4.8.

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 4.8 – проверка веб-сайта на валидность

4.5. Выводы

Разработанный веб-ресурс успешно прошел комплексную проверку работоспособности в современных браузерах, включая Chrome и Edge, а также на устройствах с различными форматами экранов - от компактных мобильных устройств до полноразмерных ноутбуков. Проведенные испытания доказали стабильность работы всех системных компонентов и правильность визуального отображения интерфейсных элементов.

С целью оптимизации пользовательского опыта была создана исчерпывающая инструкция, содержащая практические рекомендации по работе с навигационной системой и функциональными возможностями платформы. Данное руководство значительно упрощает процесс адаптации новых пользователей, делая их взаимодействие с системой максимально комфортным и понятным.

Архитектура веб-портала построена с учетом принципов логичности и простоты, что обеспечивает оперативный доступ к необходимой информации. Продуманная система навигации способствует эффективной работе с ресурсом, позволяя пользователям концентрироваться на содержательной части без необходимости преодоления сложностей интерфейса.

Заключение

Официальный сайт железнодорожного вокзала: современный цифровой ресурс для пассажиров

В рамках проекта был разработан и внедрен удобный веб-портал, обеспечивающий пассажиров актуальной информацией и упрощающий процесс планирования поездок. Сайт прошел комплексное тестирование на кроссбраузерную совместимость, адаптивность и валидность кода, что подтверждает его надежность и удобство использования.

Ключевые этапы разработки:

Анализ современных технологий и инструментов для создания высокопроизводительного веб-ресурса.

Разработка интуитивно понятного интерфейса и прототипа страниц с учетом потребностей пассажиров.

Создание четкой структуры сайта для быстрого доступа к расписанию, тарифам и сервисам вокзала.

Наполнение ресурса актуальной информацией, включая онлайн-табло, правила проезда и дополнительные услуги.

Всестороннее тестирование функционала, включая проверку на разных устройствах и браузерах.

Подготовка руководства для пользователей с инструкциями по работе с основными разделами сайта.

В результате курсового проекта разработан полнофункциональный веб-сайт железнодорожного вокзала, который обеспечивает удобный доступ к информации, упрощает процесс бронирования и способствует повышению качества обслуживания пассажиров. Все поставленные задачи выполнены в полном объеме, требования к проекту соблюдены.

Проект доступен в открытом репозитории на GitHub.

Список использованных источников

1.Российская железная дорога [Электронный ресурс] – Режим доступа: <https://www.rzd.ru> – Дата обращения 28.04.2025

2. Белорусская железная дорога [Электронный ресурс] – Режим доступа: [Расписание поездов БЖД | Купить билеты на поезд](https://pass.rw.by/ru/) - Дата обращения: 28.04.2025

3.Бесплатные иконки для графического дизайна, UX, социальных сетей и мобильных приложений [Электронный ресурс] – Режим доступа: <https://www.svgrepo.com> – Дата обращения: 28.04.2025.

4.Документация по JavaScript/HTML [Электронный ресурс]. – Режим доступа: <https://developer.mozilla.org> – Дата доступа 28.04.2025.

5.W3C – Служба проверки разметки [Электронный ресурс] – Режим доступа: <https://validator.w3.org> – Дата обращения: 01.05.2025.

6. Репозиторий проекта на GitHub[Электронный ресурс ] – Режим доступа: <https://github.com/Lozuno/KYAR_KursProject>