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

Факультет Информационных технологий

Кафедра Информационных систем и технологий

Специальность 1-40 01 01 «Программное обеспечение информационных технологий»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА КУРСОВОГО ПРОЕКТА**

по дисциплине «Компьютерные языки разметки»

Тема «Веб-сайт курсов актёрского мастерства»

**Исполнитель**

студентка 1 курса 4 группы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А. М. Жук

подпись, дата

**Руководитель**

ассистент \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 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

Оглавление

[Введение 5](#_Toc102504465)

[1. Обзор технических методов и программных средств разработки 6](#_Toc102504466)

[1.1. Обзор аналоговых решений 6](#_Toc102504467)

[1.2. Техническое задание 7](#_Toc102504468)

[1.3. Выбор средств реализации программного продукта 7](#_Toc102504469)

[1.4. Выводы 7](#_Toc102504470)

[2. Макетирование страницы веб-сайта 8](#_Toc102504471)

[2.1. Выбор способа вёрстки 8](#_Toc102504472)

[2.2. Стилевое оформление 8](#_Toc102504473)

[2.3. Выбор шрифтового оформления 8](#_Toc102504474)

[2.4. Разработка логотипа 8](#_Toc102504475)

[2.5. Разработка пользовательских элементов 9](#_Toc102504476)

[2.6. Разработка спецэффектов 9](#_Toc102504477)

[2.7. Выводы 9](#_Toc102504478)

[3. Разработка структуры веб-сайта c использованием стандартов XML 10](#_Toc102504479)

[3.1. Структура HTML-документа 10](#_Toc102504480)

[3.2 Добавление таблиц стилей Sass и CSS 10](#_Toc102504481)

[3.3. Использование стандартов XML (SVG) 11](#_Toc102504482)

[3.4. Выводы 11](#_Toc102504483)

[4. Тестирование веб-сайта 13](#_Toc102504484)

[4.1. Адаптивный дизайн веб-сайта 13](#_Toc102504485)

[4.2. Кроссбраузерность веб-сайта 14](#_Toc102504486)

[4.4. Выводы 15](#_Toc102504487)

[5. Заключение 16](#_Toc102504488)

[6. Список использованных источников 17](#_Toc102504489)

# Введение

Сайт – обязательная часть продвижения и развития компании, визитка и официальное лицо, который сочетает в себе современный и модный подход к ведению бизнеса. **Также это место, где можно найти подробную информацию о компании,** услугах, выполнении услуг и контактах. Информацию на нём легко и быстро публиковать и изменять, что является преимуществом перед различными видами традиционной рекламы. Это значит, что сайт является ещё и хорошим рекламным инструментом, инструментом привлечения новых клиентов. С помощью каталогов и поисковиков Интернет-пользователь может ознакомится с предлагаемыми услугами и стать потенциальным клиентом. Что и подчёркивает актуальность курсового проекта.

**Цели курсового проекта**:

* Разработка структуры web-сайта для курсов актёрского мастерства с использованием современных средств проектирования и разработки;
* Применение навыков, полученных в результате прохождения курса «Компьютерные языки разметки».

**Задача курсового проекта:**

* Проанализировать существующие языки разметки, инструменты и библиотеки для создания веб-сайта.
* Разработать макет и прототип сайта.
* Разработать структуру веб-сайта.
* Наполнить сайт информацией по теме.
* Протестировать веб-сайт.

**Целевая аудитория:** Данный сайт будет актуален для людей разных возрастов и профессий, которые заинтересованы в информации об курсах актёрского мастерства.

# 1. Обзор технических методов и программных средств разработки

1.1. Обзор аналоговых решений

Курсовой проект – веб-сайт для курсов актерского мастерства. Так как сфера курсов достаточно обширна и сильно развита сегодня, то существует множество аналогичных решений. В этом разделе будут приведены веб-сайты существующих курсов актерского мастерства.

Сайт spbsot.by (рисунок 1.1, а). Сайт имеет яркий и современный дизайн. Отсутствует навигационное меню, что делает данный сайт не удобным в поиске необходимой информации. Также хорошо проработана мобильная версия сайта, но и тут нет навигационного меню (рисунок 1.1, б).

|  |  |
| --- | --- |
|  |  |
| a) | б) |

Рисунок 1.1 – Макет веб-сайта spbsot.by, а) компьютерная версия; б) мобильная версия

Рассмотрим ещё один веб-сайт курсов актёрского мастерства. Данный сайт является визиткой школы актерского мастерства [letta.by](https://letta.by) (рисунок 1.2, а). На всём сайте шрифт не имеет дизайна (рисунок 1.2, б), информация представлена сплошным текстом, без картинок или других вставок – всё это сразу делает сайт непривлекательным для пользователя.

|  |  |
| --- | --- |
|  |  |
| a) | б) |

Рисунок 1.2 – Макет веб-сайта [letta.by](https://letta.by), а) главная станица; б) станица в разделе «О школе»

1.2. Техническое задание

В данном проекте требуется создать многостраничный сайт курсов актёрского мастерства. Сайт должен содержать общую информацию о курсах, такую как информацию о компании, команде преподавателей, предлагаемых курсах, ценах и контактных данных. Основным контентом являются текстовые данные, фотографии.

Главной задачей сайта является предоставление данных о услугах для привлечения внимания потенциальных клиентов. Сайт должен быть удобным, современным, содержать актуальную информацию.

На всех страницах сверху будет располагаться меню, которое содержит ссылки на основную информацию (о нас, наша команда, курсы, цены, контакты).

В мобильной версии (ширина до 425px) страница должна адаптироваться под ширину устройства, меню должно стать выпадающим для удобства просмотра. Также должна быть разработана версия для планшетного устройства (ширина до 768px). Начиная с ширины в 1000px должна открываться версия для персонального компьютера.

Необходимо реализовать макеты страниц для главной страницы, страниц об компании, преподавательского состава, курсов, цен и контактов, включающую форму для заполнения.

**Основные требования**: создание дизайна, проведение анализа существующих решений, создание прототипа, адаптивность и кроссбраузерность сайта.

1.3. Выбор средств реализации программного продукта

**Основными средствами реализации** программного продукта являются: редактор исходного кода Subline Text, графический редактор Figma, компилятор кода Sass.

**Основным языком разработки** является HTML5 с подключенным к нему CSS/ Sass.

**Стэк применяемых технологий**: *git, sass, css, html, figma, photoshop*.

1.4. Выводы

Основным программным продуктом данного курсового проекта является полноценный веб-сайт, основная задача которого – предоставлять полноценный и кратчайший доступ к информации, которая будет размещена на нем в сети Интернет. Данный веб-сайт должен иметь понятную структуру, читабельный текст и соответствующий тематике дизайн.

# 2. Макетирование страницы веб-сайта

2.1. Выбор способа вёрстки

Макет страницы веб-сайта составлен с помощью графический редактор Figma и представлен в приложении 1.

Способом версткиданного сайта является блочная вёрстка, т.е. верстка с помощью прямоугольных блоков (div). Такие блоки на странице можно располагать любым образом, что придаёт сайту эластичность и позволяет реализовать любые идеи программиста и его заказчика. Реализована вёрстка с помощью CSS Grid Layout.

**2.2.** Стилевое **оформление**

При создании макеты были использованы следующие стили: фон для сайта, смесь классического и органического дизайна. Так же в макете осуществилось перекрытие с наложением слоя цвета и применение эффекта двойной экспозиции фотографий, что делает сайт более современным и привлекательным для пользователей. Фоном сайта являются различные изображения, в том числе геометрические фигуры. Каждая страница веб-сайта пронизана одной тематикой, схожими элементами и деталями. Основным цветом является красный и его оттенки: # 890000 (цвет логотипа), # EDC9C9, #BE4C4C, #BE4C4C, #F89D9D, #FF6E6E, #F2D5D5 и др.

**2.3.** Выборшрифтового **оформления**

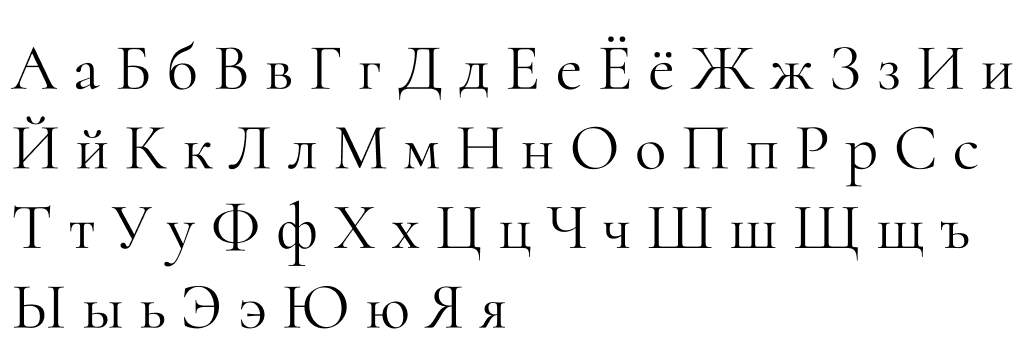
Основным и единственным шрифтом данного веб-сайта является шрифт «Cormorant»: лёгкий, динамический, элегантный, выглядит как прописной.

Рисунок 2.1 – Шрифт «Cormorant»

2.4. Разработка логотипа

Разработка логотипа данного сайта является солнце и луна (использование круглых элементов, что и наблюдается на каждой странице), выдержанные в красных тонах, т.к. именно этот цвет является базовым для данного сайта.

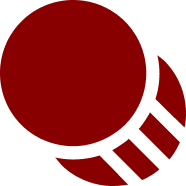


Рисунок 2.2 – Логотип «fACTORio»

2.5. Разработка пользовательских элементов

Для удобного перехода с одной веб-страницы к другой осуществлены ссылки в виде горизонтального меню для компьютеров и планшетов (рисунок 2.3, верхняя часть) и выпадающее меню для мобильных устройств (рисунок 2.3, нижняя часть). Помимо этого, на многих страницах есть кнопки ссылающие на другую страницу (в данном случае на страницу «Контакты»).

|  |
| --- |
|  |
|  |

Рисунок 2.3 – Навигационное меню, верхняя часть – для компьютеров и планшетов; нижняя часть – для мобильных устройств

2.6. Разработка спецэффектов

Будет осуществлять анимация навигационной панели: название страницы, на которой вы находитесь, подчёркнута снизу и приобретает при наведении фон, так же при наведении мыши на названия страниц, они подчёркиваются.

Элементы фона, геометрические фигуры, так же будут анимированы. При наведении курсора элемент меняет прозрачность, угол поворота и/или местоположение на странице. В форме каждая графа при нажатии на неё будет менять цвет фона.

2.7. Выводы

На данном этапе был создан макеты страниц веб-сайта, которые демонстрируют визуальное решение и значительно облегчают верстку сайта. Было продумано расположение элементов на страницах (навигационного меню, логотипа, основного контента каждой страницы, расположение картинок, подвала сайта). Было подобрано стилевое оформление сайта: основная цветовая гамма, шрифтовое оформление, элементы пользовательского интерфейса, а также эффекты и переходы.

# 3. Разработка структуры веб-сайта c использованием стандартов XML

3.1. Структура HTML-документа

Основным элементом при разработке веб-сайта является язык разметки HTML, задача которого – установить структуру программного продукта, предоставив после доступ для стилевых решений путем указания классов, идентификаторов и ключевых имен (тегов).

Главная часть каждой из веб-страниц основана на пути верстки grid, что определяет дальнейшее расположение элементов страницы путем указания классового имени class в стилевом решении проекта. Для оформления ссылок на различные веб-страницы использовался тег a с атрибутом href, который содержит в себе ссылку на ресурс. При необходимости создания списка использовались теги ul, которые состоят из некоторого числа элементов li, каждый из которых содержит в себе некоторую информацию. Для вставки и оформления изображений был использован тег img, атрибут src которого содержит в себе ссылку на изображение для дальнейшего его внедрения. Для оформления основной текстовой информации был применен тег p, задача которого – содержать некоторую информацию, размещение которой требует выделенного пространства. Интеграция SVG-изображений проводилась с использованием тега svg, который инициализировал пространство для разработки изображения, координаты которого указывались с помощью тега path.

3.2 Добавление таблиц стилей Sass и CSS

При разработке веб-сайта ключевое место занимают стилевые решения, которые определяют внешний вид каждого элемента на веб-страницах. Для оформления основной части файла использовался современный язык стилей Sass, инструментарий которого предоставляет расширенный доступ к технологиям написания стилей. Путем компиляции файлов Sass позволяет получать готовые CSS файлы, которые, в свою очередь, могут быть подключены к HTML-страницам. Пример оформления Sass и компиляции его в CSS представлен на Рисунках 3.1 и 3.2.

|  |  |
| --- | --- |
|  |  |
| Рисунок 3.1. Пример Sass-кода | Рисунок 3.2. Пример компиляции кода в CSS |

В качестве примера было представлено стилевое оформление с учетом языка Sass, которое затем было преобразовано автоматически в файл стилей CSS.

3.3. Использование стандартов XML (SVG)

Для соблюдения стандартов оформления веб-сайтов был разработан файл XML (Extensible Markup Language), основной задачей которого является хранение текстовой информации для упрощенного доступа без внедрения в HTML-документ, для которого предназначен данный XML-файл. Его основным достоинством также является возможность создания пользовательских тегов, что упрощает чтение файлов разработчиком, как и создание алгоритмов для получения данных. В проекте был создан файл catalog.xml, информация которого предназначена для веб-страницы prices.html. В нем были использованы следующие теги:

1. PriceList – контейнер, содержащий информацию о трех баннерах, каждый из которых представляет собой массив данных.
2. PriceListColumn – баннер с информацией о курсе.
3. NamePurpose – название курса по его назначению.
4. Weeks – продолжительность курса.
5. Bonus – дополнительные недели, которые будут предоставлены покупателю данного курса.
6. TimesPerWeek – периодичность проведения курса.
7. Lesson – занятие, длительность которого представлена в теге Hour.
8. Hour – продолжительность занятия.
9. Group – размер группы.
10. Price – цена данного курса.

Путем обработки данных тегов можно получить информацию как об имеющихся курсах, так и об их особенностях.

При разработке изображений для веб-сайта были использованы стандарты SVG (Scalable Vector Graphics), которые позволяют производить прямые манипуляции с масштабами веб-страниц без потери качества данных изображений. Основными компонентом веб-сайта, разработка которого велась с использованием стандарта SVG, является логотип, интеграция которого производилась с использованием тега path, который позволяет указывать координаты изображений для дальнейшего их создания. Остальные изображения были подключены с использованием тега img с атрибутом src. Разработка всех SVG-изображений совершалась в Adobe Illustrator.

3.4. Выводы

При разработке структуры веб-сайта были учтены все современные особенности использования и создания стилей. Была применена grid-верстка, которая позволяет равномерно распределять информацию на странице. Разработка стилевых решений велась с использованием языка Sass, который предоставляет широкий выбор различных возможностей алгоритмизации. Применение xml-файлов позволило получать информацию напрямую, без внедрения в HTML-документ. SVG-элементы позволили производить прямые манипуляции с масштабами веб-страниц без потери качества изображений.

# 4. Тестирование веб-сайта

4.1. Адаптивный дизайн веб-сайта

Для просмотра контента веб-сайта на различных устройствах требуется создание дополнительных стилей, которые позволяют отображать страницы и их содержимое в соответствии с тем устройством, на котором пользователь их открывает. Основные устройства: Mobile S (320px), Mobile M (375px), Mobile L (425px), Tablet (768px), Laptop (1024px), Laptop L (1440px).

Для создания адаптивности веб-сайта были использованы следующие инструменты: медиа-запросы (@media), единицы измерения, которые связаны непосредственно с параметрами области просмотра (представляют собой процентные величины от текущей области просмотра браузера), Грид-раскладка (CSS Grid Layout).

На примере страницы «Наша команда» протестируем адаптивность веб-сайта. Тестирование производится в командной строке разработчика Google Chrome. Планшетная версия представлена на рисунке 4.1. По сравнению с версией для персонального компьютера, размер шрифта, изображений, навигационного меню уменьшился, стал более компактным и собранным.

|  |  |
| --- | --- |
|  |  |

Рисунок 4.1 – Страница «Наша команда», область просмотра экрана 768px

Версии веб-страницы для различных мобильных устройств представлены на рисунке 4.2. В данных версиях навигационное меню становится выпадающим, что позволяет не перегружать панель. Все элементы, расположенные в ряд, становятся друг за другом, «оплывают» друг друга.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| а) | б) | с) |

Рисунок 4.2 – Страница «Наша команда», а) область просмотра экрана 425px; б) область просмотра экрана 375px; с) область просмотра экрана 320px

4.2. Кроссбраузерность веб-сайта

При разработке стилевого оформления и структуры веб-сайта необходимо учитывать, как размеры устройства просмотра пользователя, так и программный продукт, который используется для просмотра страниц. Для этого требуется провести тестирование веб-сайта в различных браузерах, результаты которого покажут возможности, достоинства и недостатки стилевого оформления в каждом из рассматриваемых браузеров.

Для тестирования будут использованы такие браузеры, как: Google Chrome, Opera и Microsoft Edge. По результатам исследований, более 90% пользователей сети Интернет могут быть охвачены данными тремя браузерами.

1. Тестирование в Google Chrome.

Для проверки кроссбраузерности в Google Chrome будут рассмотрены две версии главной страницы веб-сайта, такие как мобильная и широкоэкранная, изображения которых представлены на рисунке 4.3.

Результат тестирования в браузере Google Chrome показал полную готовность как основной, так и адаптивной версии веб-сайта к пользованию в данном браузере. Ошибок при пользовании обнаружено не было, стилевые решения сохранены в исходном формате, без искажений.

2. Тестирование в Microsoft Edge.

В качестве альтернативы браузеру Google Chrome будет рассмотрен продукт компании Windows – Microsoft Edge, главной особенностью которого является встроенная поддержка операционной системы Microsoft Windows, начиная с версии 10 и выше. Как и в Google Chrome, в Microsoft Edge будут рассмотрены две версии веб-сайта, результаты проверки которых являются наиболее приоритетными в рамках производимого тестирования. Результаты проверки главной веб-страницы можно наблюдать на рисунке 4.4.

Результат тестирования программного продукта в браузере Microsoft Edge показал готовность к использованию в данной среде пользователем. Ошибок стилевого оформления замечено не было, структура веб-страницы сохранена.

3. Тестирование в браузере Opera.

Последним браузером, принимавшим участие в тестировании кроссбраузерности веб-сайта курсов актерского мастерства, является Opera. Opera – веб-браузер, пакет прикладных программ, разработанных для пользования, сбора и актуализации информации в сети Интернет, разработанный компанией Opera Software в 1994 году. Подготовлен на использования такими операционными системами, как: Microsoft Windows, macOS, Linux, iOS, Android.

В качестве тестируемой веб-страницы будет рассмотрена главная страница веб-сайта, с использованием как широкоэкранной, так и мобильной версий. Представление данных версий можно увидеть на рисунке 4.5.

Проверка веб-сайта в браузере Opera показала отсутствие нарушений при разработке стилевых решений как в широкоэкранной, так и в мобильной версиях программного продукта, что позволяет удостовериться в возможности получения полного доступа клиентом без возникновения ошибок в оформлении веб-страниц.

4.4. Выводы

Тестирование веб-сайта является одним из ключевых моментов процесса разработки программного продукта. Тестирование позволяет выявить ошибки стилевых решений веб-сайта и нарушения структуризации, обработка которых улучшает общую результативность и работоспособность каждой из веб-страниц.

Проверка адаптивности веб-сайта предоставила возможность получения информации о готовности веб-страниц к обзору вне зависимости от используемого устройства и размеров обзорной зоны экрана.

Тестирование кроссбраузерности программного продукта предоставило актуальную информацию о работоспособности веб-сайта в различных пользовательских средах, что является необходимым фактором для полного охвата всех сфер пользовательской деятельности в сети Интернет.

# 5. Заключение

Разработка веб-сайта компании является комплексным решением, состоящим из множества различных пунктов, каждый из которых требует ответственного подхода. Несмотря на простоту внешнего пользования, проект должен следовать современным стандартам разработки, что позволяет создать веб-сайт, полностью соответствующий требованиям пользователя.

В данной курсовой работе был пройден полный путь создания, проектирования и формирования структуры веб-сайта: от постановки основных целей проекта до создания структуры и написания кода. Были учтены ошибки при создании сайтов, уже имеющихся в сети Интернет. Разработка шаблона программного продукта велась в редакторе Figma, который предоставляет возможность быстрой визуализации контента для дальнейшего его проектирования.

Основным методом разработки структуры веб-страницы стал метод Grid, который способствует быстрому и качественному структурированию и разделению информации, представляемой на каждой из страниц веб-сайта.

При создании стилевого оформления веб-страниц был использован язык Sass, основным преимуществом которого является предоставление возможностей алгоритмизации объектов кода. После написания кода на данном языке он был конвертирован в язык разметки CSS, который затем был подключен к веб-страницам.

После создания и оформления кода веб-страниц были проведены тесты кроссбраузерности и адаптивности программного продукта, результаты которых показали готовность веб-сайта к использованию в различных средах, что является необходимым пунктом при разработке. Для упрощения доступа пользователя было разработано руководство пользователя, основной целью которого было предоставление информации о возможностях и пользовательских элементах программного продукта.

В ходе создания веб-сайта и написания пояснительной записки были выполнены все поставленные задачи. Разработанный проект готов к беспрепятственному доступу и использованию в сети Интернет.

# 6. Список использованных источников

1. Международная школа мастерства [Электронный ресурс]. – Режим доступа: https://spbsot.by. – Дата обращения: 02.05.2022.
2. Школа актерского мастерства Letta [Электронный ресурс]. – Режим доступа: https://letta.by. – Дата обращения: 02.05.2022
3. Основы Sass [Электронный ресурс]. – Режим доступа: <https://sass-scss.ru/guide/>. – Дата обращения: 02.05.2022.
4. Введение в кросс-браузерное тестирование [Электронный ресурс]. – Режим доступа: https://developer.mozilla.org/ru/docs/Learn/Tools\_and\_testing/Cross\_browser\_testing/Introduction. – Дата обращения: 02.05.2022.
5. Figma Layouts [Электронный ресурс]. – Режим доступа: <https://www.figma.com/community>. – Дата обращения: 02.05.2022.
6. CSS Selector Reference [Электронный ресурс]. – Режим доступа: https://www.w3schools.com/cssref/css\_selectors.asp. – Дата обращения: 02.05.2022.