

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**(ДГТУ)**

Факультет «Информатика и вычислительная техника»

Кафедра «Программное обеспечение вычислительной техники и автоматизированных систем»

|  |  |  |
| --- | --- | --- |
|  | | |
| Зав. кафедрой | | «ПОВТиАС» |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | В.В. Долгов |
| (подпись) | |  |
| «\_\_\_» | \_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г. | |

**ОТЧЕТ**

по производственной технологической практике

на научно-образовательном производственном центре «Визуализация, анализ и защита информации»

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.С. Бережнов

подпись, дата

Обозначение отчета ПП.780000.000 Группа ВПР23

Направление 09.03.04 Программная инженерия

Профиль Программное обеспечение вычислительной техники и автоматизированных систем

Руководитель практики:

от предприятия руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В.Долгов

подпись, дата

от кафедры доцент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.А. Скляренко

подпись, дата

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

дата подпись преподавателя

Ростов-на-Дону  
2021



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**(ДГТУ)**

Факультет «Информатика и вычислительная техника»

Кафедра «Программное обеспечение вычислительной техники и автоматизированных систем»

**ЗАДАНИЕ**

на производственную технологическую практику

на Научно-образовательный производственный центр «Визуализация, анализ и защита информации»

в период с « 21 » июня 2021 г. по « 17 » июля 2021 г.

Обучающийся Бережнов Артем Сергеевич

Обозначение отчета ПП.780000.000 Группа ВПР23

Срок представления отчета на кафедру « 17 » июля 2021 г.

Содержание индивидуального задания:

Разработать сайт-визитку на тему «Визитка начинающего разработчика»

|  |  |  |
| --- | --- | --- |
| Руководитель практики от кафедры | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись, дата | доцент, А.А. Скляренко |
| Задание принял к исполнению | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись, дата | А.С.Бережнов |



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**(ДГТУ)**

Факультет «Информатика и вычислительная техника»

Кафедра «Программное обеспечение вычислительной техники и автоматизированных систем»

|  |  |  |
| --- | --- | --- |
| Зав. кафедрой | | «ПОВТ и АС » |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | В.В. Долгов |
| (подпись) | |  |
| «\_\_\_» | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2021 г. | |

Рабочий график (план) проведения практики

|  |  |  |
| --- | --- | --- |
| № | Мероприятие | Срок выполнения |
| 1 | Прохождение вводного и первичного инструктажа по охране труда на рабочем месте, и инструктажа по пожарной безопасности на объекте | 21.06.2021 |
| 2 | Аналитический обзор | 22.06.2021 – 25.06.2021 |
| 3 | Алгоритмическое конструирование | 26.06.2021 – 1.07.2021 |
| 4 | Программная реализация | 2.07.2021 – 11.07.2021 |
| 5 | Тестирование приложения | 12.07.2021 – 13.07.2021 |
| 6 | Подготовка итогового отчета | 14.07.2021 – 16.07.2021 |
| 7 | Защита итогового отчета | 17.07.2021 |

Руководитель практики:

от предприятия руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_В.В.Долгов\_\_\_\_\_\_\_\_\_\_

должность подпись, дата имя, отчество, фамилия

от кафедры доцент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.А. Скляренко

подпись, дата

Ростов-на-Дону  
2021

**ДНЕВНИК ПРОХОЖДЕНИЯ ПРАКТИКИ**

|  |  |  |  |
| --- | --- | --- | --- |
| Дата | Место работы | Выполняемые работы | Оценка руководителя |

| 1 | 2 | 3 | 4 |
| --- | --- | --- | --- |
| 21.06.2021 | Научно-образовательный производственный центр «Визуализация, анализ и защита информации» | Знакомство с предприятием, прохождение вводного инструктажа |  |
| 21.06.2021 | Научно-образовательный производственный центр «Визуализация, анализ и защита информации» | Ознакомление с территорией предприятия, прохождение первичного инструктажа по ТБ, ПБ |  |
| 22.06.2021 – 25.06.2021 | Научно-образовательный производственный центр «Визуализация, анализ и защита информации» | Анализ предметной области |  |
| 26.06.2021 – 1.07.2021 | Научно-образовательный производственный центр «Визуализация, анализ и защита информации» | Разработка алгоритмов |  |
| 2.07.2021 | Научно-образовательный производственный центр «Визуализация, анализ и защита информации» | Проектирование программного средства (ПС). Выбор средств разработки |  |
| 3.07.2021 – 11.07.2021 | Научно-образовательный производственный центр «Визуализация, анализ и защита информации» | Программная реализация ПС |  |
| 12.07.2021 – 13.07.2021 | Научно-образовательный производственный центр «Визуализация, анализ и защита информации» | Тестирование ПС |  |
| 14.07.2021 – 16.07.2021 | Научно-образовательный производственный центр «Визуализация, анализ и защита информации» | Подготовка итогового отчета |  |
| 17.07.2021 |  | Защита итогового отчета |  |

Руководитель практики:

от предприятия руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_В.В.Долгов\_\_\_\_

должность подпись, дата имя, отчество, фамилия

**ОТЗЫВ - ХАРАКТЕРИСТИКА**

Обучающийся Бережнов Артем Сергеевич

2 курса группы ВПР23

кафедры «Программное обеспечение вычислительной техники и автоматизированных систем»

Вид практики: производственная технологическая

Наименование места практики: Научно-образовательный производственный центр «Визуализация, анализ и защита информации»

Обучающийся выполнил задание программы практики, разработал сайт-визитку.

Дополнительно ознакомился/изучил язык разметки HTML, язык таблиц стилей CSS, язык программирования JavaScript.

Заслуживает оценки \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  | Руководитель практики  от предприятия  руководитель В.В.Долгов  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_2021 г. |

**Содержание**

[Введение 8](#_Toc76985678)

[1. Аналитический обзор предметной области 9](#_Toc76985679)

[1.1 Обзор видов сайтов 9](#_Toc76985680)

[1.2 Обзор основных технологий разработки сайтов 11](#_Toc76985681)

[1.3 Обзор популярных сайтов-визиток 11](#_Toc76985682)

[1.4 Постановка задачи 14](#_Toc76985683)

[1.5 Выводы по главе 15](#_Toc76985684)

[2. Проектирование сайта 16](#_Toc76985685)

[2.1 Описание карты сайта 16](#_Toc76985686)

[2.2 Описание структуры страницы 16](#_Toc76985687)

[2.3 Алгоритм слайдера 17](#_Toc76985688)

[2.4 Вывод 18](#_Toc76985689)

[3. Программное конструирование 19](#_Toc76985690)

[3.1 Выбор средств разработки 19](#_Toc76985691)

[3.1.1 Выбор среды программирования 20](#_Toc76985692)

[3.2 Описание основных страниц сайта 20](#_Toc76985693)

[3.2.1 Страница index.html 21](#_Toc76985694)

[3.3 Описание основных функций 23](#_Toc76985695)

[3.4 Вывод по главе 23](#_Toc76985696)

[4. Тестирование 24](#_Toc76985697)

[4.1 Описание процесса тестирования 24](#_Toc76985698)

[4.2 Выводы по главе 26](#_Toc76985699)

[Заключение 27](#_Toc76985700)

[Перечень использованных информационных ресурсов 28](#_Toc76985701)

[Приложение А](#_Toc76985702) [Техническое задание 29](#_Toc76985703)

[Приложение Б](#_Toc76985704) [Исходный код программного средства 34](#_Toc76985705)

Изм.

Лист

№ докум.

Подпись

Дата

Лист

УП.510000.000

# Введение

В настоящее время Интернет становиться все более развитой средой для осуществления коммуникаций. В связи с глобальным развитием сети Интернет, в программировании все более резко начала выделяться отдельная его отрасль web-программирование.

Сейчас, чтобы привлечь внимание клиентов, покупателей или партнёров, просто необходимо заявить о себе в интернете, путём создания WEB-сайта. Для этих целей как раз и служит web-сайт, содержащий основную информацию об организации, частном лице, компании, товарах или услугах, прайс-листы, контактные данные. Сайты позволяют хранить, передавать, продавать различные типы информации, не отходя от экрана компьютера.

World Wide Web - глобальная компьютерная сеть, на сегодняшний день содержит миллионы сайтов, на которых размещена всевозможная информация. Люди получают доступ к этой информации посредством использования технологий Internet. Для поиска по интернету используют специальные программы - Web-браузеры, которые существенно облегчают путешествие по бескрайним просторам интернета [1].

1. **Аналитический обзор предметной области**

В данной главе рассматривается понятие сайта, основные виды сайтов, подробно описываются основные технологии разработки сайтов, производится обзор популярных сайтов-визиток, определяется цель работы и задачи для её достижения.

* 1. **Обзор видов сайтов**

Сайт — это набор страниц в интернете, позволяющий пользователю получить определённую информацию, либо осуществить действие — например, заказать товар и услугу. Состоит из одной либо нескольких логически связанных между собой страниц, объединенных дизайном, общей тематикой и гиперссылками.

Для создания интернет-ресурсов используются специальные технологии, определяющие как внешний вид, так и внутреннюю структуру. Оформление и наполнение каждой веб-площадки подбирается индивидуально и зависит от выбранной тематики и назначения. Рассмотрим, какие сайты бывают, в чем их отличия и в каких целях они используются.

Сайты могут различаться между собой по многим параметрам, поэтому пока не существует единой классификации. Одну из классификаций можно наблюдать на рисунке 1.1. Эксперты разделяют сайты по наиболее значимым признакам, например, по тематике, структуре, дизайну, способу и цели создания.

Если интернет-ресурс посвящён одной теме, его называют узкотематическим, если же он охватывает широкий круг проблем — общим. В зависимости от ориентированности на получение прибыли различают коммерческие и некоммерческие сайты, а по количеству страниц – одностраничные и многостраничные.



Рисунок 1.1 - Разновидности и категории сайтов

По доступности интернет-площадки делятся на три категории:

* открытые – предоставляющие информацию каждому желающему ее получить;
* полузакрытые – с ограниченным доступом, требующие предварительной регистрации;
* закрытые – с доступом по паролю, открытые для узкого круга лиц (студенческие, военные, служебные и пр.).

Также сайты разделяют по типу загрузки, если он хранится в готовом виде в файловой системе сервера, то он статический. Если же его страницы генерируются в ходе пользовательских запросов на основе загруженных баз данных, то динамический.

В зависимости от круга поставленных задач, выбирается тип сайта и составляется план его внутренней структуры. От способа разработки в большой степени зависит скорость продвижения веб-ресурса в поисковиках [2].

* 1. **Обзор основных технологий разработки сайтов**

Любая разработка сайта состоит из двух частей: программной и визуальной. Над визуальной работает Frontend-разработчик, надо программной Backend-специалист. Для вторых основными инструментами выступают языки программирования PHP, Python, Ruby. Существуют специальные конструкторы по типу WIX и Tilda, упрощающие процедуру создания веб-сайта.

Основные современные технологии создания сайтов:

* HTML, CSS – технологии для создания разметки (HTML) и внешнего вида страницы (язык таблиц стилей CSS);
* PHP, JavaScript – языки программирования, заставляющие «ожить» страницу;
* база данных MySQL – отвечает за хранение текстовой информации;
* Notepad++, Photoshop, Filezilla–специальные инструменты для верстки,создания графики, редактирования файлов на хостинге и других задач;
* CMS – дословно система управления контентом. Нужна для удобного управления содержимым сайта, блога. Обычно используют специальные системы по типу WordPress, Joomla, OpenCart.
  1. **Обзор популярных сайтов-визиток**

В данной подразделе производится обзор популярных сайтов-визиток с их кратким описанием и рассмотрением.

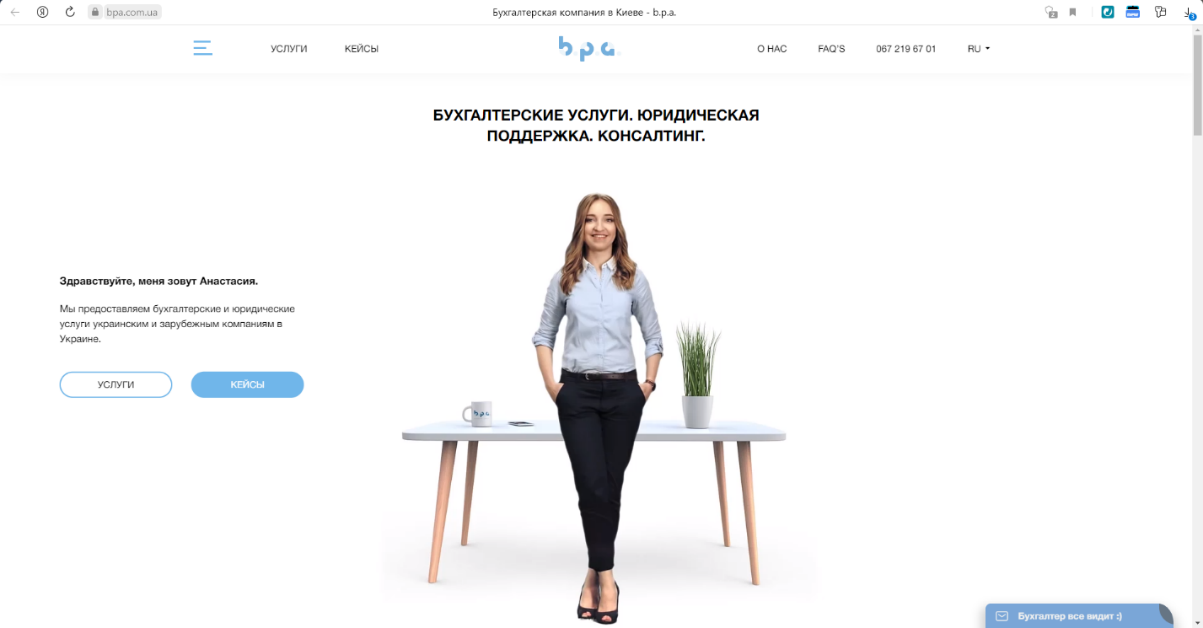
****Пример 1. [www.bpa.com.ua](http://www.bpa.com.ua)

Рисунок 1.2 - Сайт [www.bpa.com.ua](http://www.bpa.com.ua)

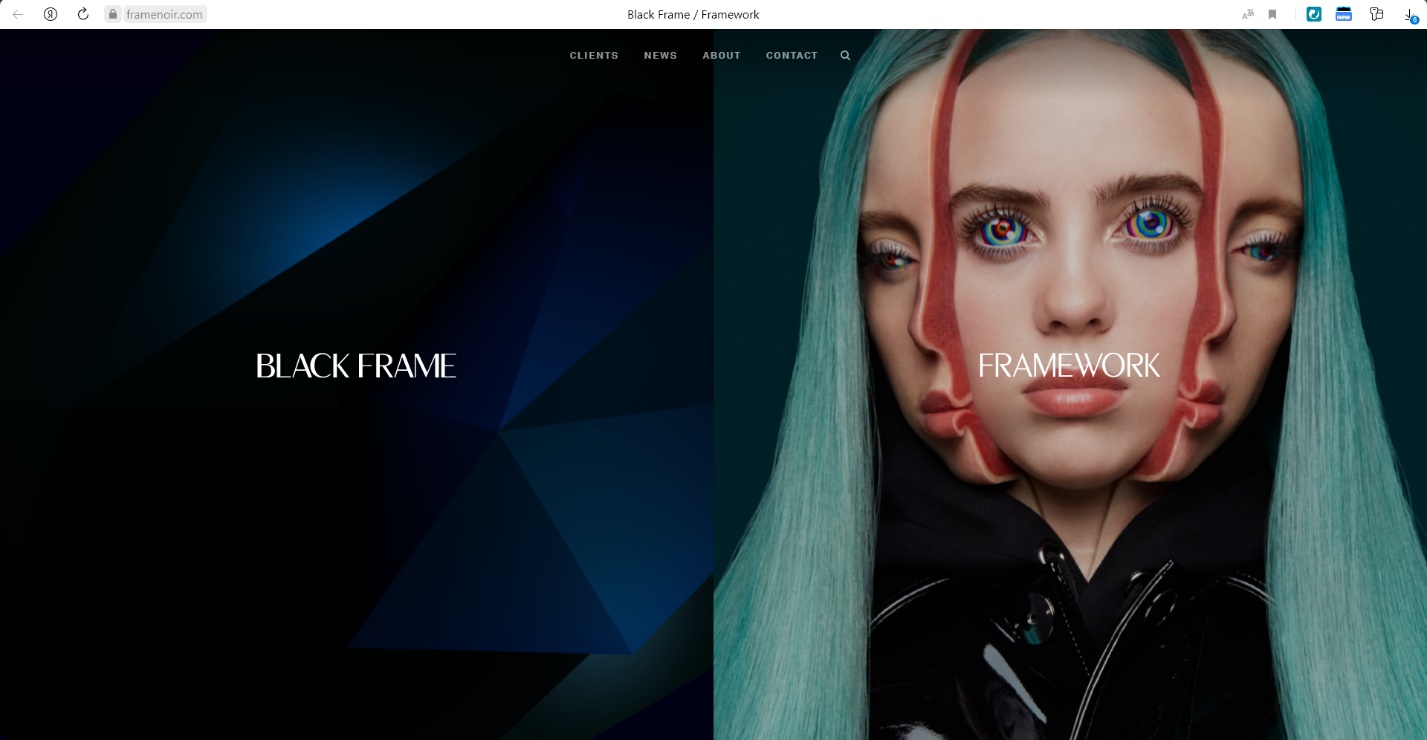
Скромный, лаконичный, но очень полезный сайт-визитка, посвященный деятельности юридической фирмы, который в полной мере раскрывает компанию. Простая навигация, достаточно большой объем инфоконтента с описанием деятельности конторы, очень простой дизайн в совокупности создают впечатления компетентности фирмы. Здесь нет ничего лишнего - только самая важная информация.

Пример 2. [www.evoulve.com](http://www.evoulve.com)

Рисунок 1.3 - Сайт [www.bpa.com.ua](http://www.bpa.com.ua)

Сайт компании, работающей в сфере B2B - разрабатывает приложения для других компаний. Сам ресурс состоит всего из одной страницы и слайдов, где на одном и том же фоне очень кратко описывается деятельность и возможности компании. Как результат, мы видим типичный пример сайта-визитки с интересным дизайном и отличной идеей - познакомить потенциального клиента с компанией, рассказав ее преимущества и ценности.

Пример 3. [www.framenoir.com](http://www.framenoir.com)

Рисунок 1.4 - Сайт [www.framenoir.com](http://www.framenoir.com)

Сайт-визитка, посвященный компании, занимающейся рекламой и брендингом, позволяет близко познакомиться с компанией, выявить ее особенности и преимущества. Сайт состоит всего из нескольких страниц - основных категорий, присущих визитке.

Как и положено данному типу ресурса, здесь наблюдается минимум контента и функционала. Сглаживает все динамичный дизайн, который вызывает приятные впечатления о компании.

* 1. **Постановка задачи**

Проведенный обзор разновидностей и классификации сайтов, основных технологий их разработки, а также рассмотрение некоторых примеров показывает все возможности для разработки сайтов. Таким образом целью производственной практики является разработка сайта-визитки на тему «Визитка начинающего разработчика».

Для достижения поставленной цели необходимо решить следующие задачи:

* выбрать среду и язык реализации;
* разработать текстовое оформление сайта;
* разработать визуальное оформление сайта;
* реализовать алгоритмы для функционирования программных возможностей сайта-визитки;
* провести тестирование программного средства на нескольких контрольных примерах;
* в соответствии с отчетом по тестированию выполнить корректировку ПС;
* разработать сопроводительную документацию: техническое задание, отчет;

Разрабатываемое программное средство будет обладать следующими функциями:

* организация cлайдера;
* cмена темы страницы;
* вызов модульного окна;
* сортировка и поиск товаров.
  1. **Выводы по главе**

В первой главе было рассмотрено понятие сайта, а также разновидности и классификации сайтов. Проведен обзор основных технологий разработки сайтов, приведено несколько примеров сайтов-визиток, поставлена задача.

1. **Проектирование сайта**

В данном разделе рассматривается общая структура сайта, производится описание структуры страница сайта, приведен алгоритм работы слайдера.

* 1. **Описание карты сайта**

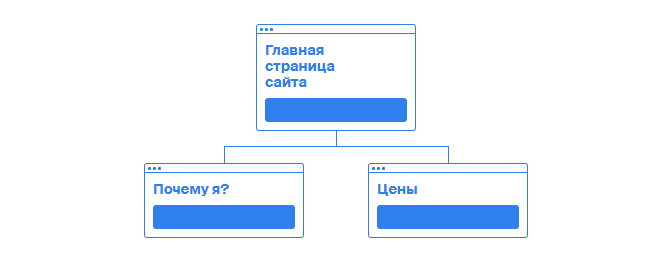
Карта сайта - представляет собой список страниц сайта для поисковых систем или пользователей и аналогична разделу Содержание обычной книги. Используется как элемент навигации, показывает взаимосвязь между страницами сайта [3].

Рисунок 2.1 – Карта сайта

На рисунке 2.1 приведен абстрактный пример карты разрабатываемого сайта, который будет состоять из 3-ех страниц: главной и двух побочных.

* 1. **Описание структуры страницы**

В данном подразделе производится обзор структуры каждой из страниц сайта. Каждая из страниц разрабатываемого сайта будет разделена на три части, а именно:

* Header – представляет из себя верхнюю часть сайта состоящую из переходов на другие страницы;
* Main – основная часть страницы представляющая из себя блок, содержащий какую-либо информацию;
* Footer – нижняя часть листа, содержащая информацию о разработчике или компании.

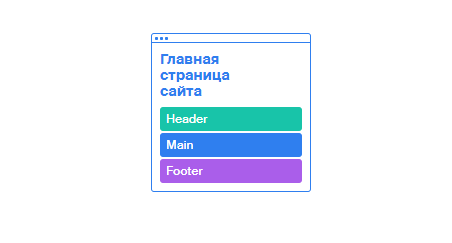
****

Рисунок 2.2 - Структура страницы

* 1. **Алгоритм слайдера**

В разрабатываемом программном средстве используется алгоритм для работы слайдера.

Слайдер – это элемент на сайте, который представляет собой блок определенной ширины с изменяющимся контентом. Это могут быть изображения, видео, текстовые блоки, ссылки.

Составные элементы стандартного слайдера:

* экран (слайд);
* средства навигации;
* маркеры с количеством слайдом.

Можно ещё настроить отображение миниатюр остальных слайдов, таймер со временем до смены слайда и т.д.

В функциях работы слайдера можно указать скорость и задать эффекты перелистывания слайдов, порядок отображения и визуальное оформление слайдов, зациклить их показ по кругу и множество разных настроек.

Чаще всего слайдер создаётся с помощью HTML, CSS, Javascript. Однако уже существует множество готовых решений [4].

Сам алгоритм работы слайдера реализован следующим образом:

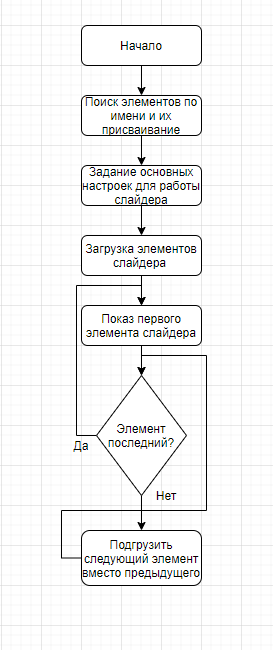
****

Рисунок 2.3 – Блок-схема работы алгоритма слайдера

* 1. **Вывод по главе**

В данной главе была рассмотрена общая структура сайта, было произведено описание структуры страница сайта, приведен алгоритм работы слайдера, который используется для дальнейшей реализации.

1. **Программное конструирование**

На основе выбранных алгоритмов решения поставленной задачи выполнено программное конструирование, предполагающее выбор соответствующих требованиям сред разработки и программирования и определение основных функций, составляющих будущее программное средство.

* 1. **Выбор средств разработки**

Выбор средств обусловлен поставленной задачей. А именно: для разработки текстового оформления сайта использовался язык разметки HTML, для создания стиля сайта использовался язык CSS, для программной части использовался язык программирования JavaScript.

HTML (от [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) HyperText Markup Language — «язык [гипертекстовой](https://ru.wikipedia.org/wiki/%D0%93%D0%B8%D0%BF%D0%B5%D1%80%D1%82%D0%B5%D0%BA%D1%81%D1%82) разметки») — стандартизированный язык разметки документов для просмотра веб-страниц в [браузере](https://ru.wikipedia.org/wiki/%D0%91%D1%80%D0%B0%D1%83%D0%B7%D0%B5%D1%80). Веб-браузеры получают HTML документ от сервера по протоколам [HTTP](https://ru.wikipedia.org/wiki/HTTP)/[HTTPS](https://ru.wikipedia.org/wiki/HTTPS) или открывают с локального диска, далее интерпретируют код в интерфейс, который будет отображаться на экране монитора.

CSS (Cascading Style Sheets) — язык таблиц стилей, который позволяет прикреплять стиль (например, шрифты и цвет) к структурированным документам (например, документам HTML и приложениям XML).

Обычно CSS-стили используются для создания и изменения стиля элементов веб-страниц и пользовательских интерфейсов, написанных на языках HTML и XHTML, но также могут быть применены к любому виду XML-документа, в том числе XML, SVG и XUL.

JavaScript — [мультипарадигменный](https://ru.wikipedia.org/wiki/%D0%9C%D1%83%D0%BB%D1%8C%D1%82%D0%B8%D0%BF%D0%B0%D1%80%D0%B0%D0%B4%D0%B8%D0%B3%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) [язык программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F). Поддерживает [объектно-ориентированный](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5), [императивный](https://ru.wikipedia.org/wiki/%D0%98%D0%BC%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) и [функциональный](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) стили. Является реализацией спецификации [ECMAScript](https://ru.wikipedia.org/wiki/ECMAScript" \o "ECMAScript) (стандарт ECMA-262).

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам [приложений](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0). Наиболее широкое применение находит в [браузерах](https://ru.wikipedia.org/wiki/%D0%91%D1%80%D0%B0%D1%83%D0%B7%D0%B5%D1%80) как язык сценариев для придания [интерактивности](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%B0%D0%BA%D1%82%D0%B8%D0%B2%D0%BD%D0%BE%D1%81%D1%82%D1%8C) [веб-страницам](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0).

Основные архитектурные черты: [динамическая типизация](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%BD%D0%B0%D0%BC%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F), [слабая типизация](https://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D0%B0%D0%B1%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F), [автоматическое управление памятью](https://ru.wikipedia.org/wiki/%D0%A1%D0%B1%D0%BE%D1%80%D0%BA%D0%B0_%D0%BC%D1%83%D1%81%D0%BE%D1%80%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), [прототипное программирование](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D1%82%D0%B8%D0%BF%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5" \o "Объект первого класса), функции как [объекты первого класса](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82_%D0%BF%D0%B5%D1%80%D0%B2%D0%BE%D0%B3%D0%BE_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B0) [5].

* + 1. **Выбор среды программирования**

Для реализации разработанного алгоритма была выбрана среда программирования Visual Studio 2019.

Visual Studio – инструментальная среда разработки, включающая в себя интегрированную среду разработки, редактор исходного кода, встроенный отладчик. Многие другие инструменты возможно получить благодаря подключению плагинов – сторонних расширений. Был выбран по причине наличия интеграции с Unity 3d, поддержки языка программирования С#, наличие возможности написание собственных Unit-тестов. Также это наиболее изученная разработчиком платформа, что позволяет быстрее разрабатывать приложения не тратя время на изучения нового материала [6].

* 1. **Описание основных страниц сайта**

Для реализации страниц сайта были написаны следующие файлы:

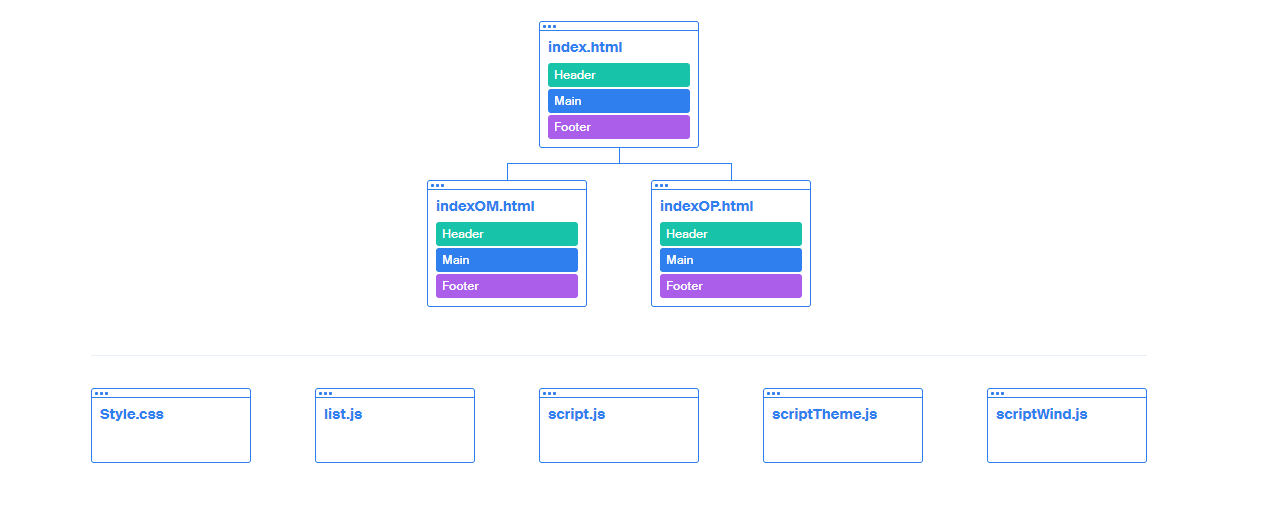
* index.html
* indexOM.html
* indexOP.html
* list.js
* script.js
* scriptTheme.js
* scriptWind.js
* style.css

Рисунок 3.1- Полная карта разрабатываемого сайта

Файлы style.css, list.js, script.js, scriptTheme.js, scriptWind.js подключаются к каждому .html документу они необходимы для реализации стилевой и программной части разрабатываемого сайта, полную карту разрабатываемого сайта можем наблюдать на рисунке 3.1, где в нижней части рисунка отображены документы .css и .js, подключаемые к html- документу.

* + 1. **Страница index.html**

В соответствии с шаблоном был написан Header:

Листинг 3.1- Исходный код Header документа index.html

<header class="header-s">

<a href="./index.html" class="atop">naaZzz</a>

<h1></h1>

<button class="spanTem header-s theme-button" type="button">Изменить тему</button>

<a href="./indexOM.html" class="spanW header-s">Почему я?</a>

<a href="./indexOP.html" class="spanPr header-s">Цены</a>

<div class="dropdown spanlanguage">

<div class="dropbtn">RU</div>

<div class="dropdown-content">

<a href="#">RU</a>

<a href="#">USA</a>

</div>

</div>

<span class="lan"></span>

<span class="line"></span>

<span class="line line2"></span>

<span class="line3"></span>

<span class="line3 line4"></span>

<img class="priceP" src="file:///C:\Users\artem\Desktop\visitka\img\price-tag.svg" />

</header>

В соответствии с шаблоном был написан Main:

Листинг 3.2- Исходный код Main документа index.html

<section class="main">

<h2>Начинающий разработчик</h2>

<p class="pcont">Создаю все своими руками</p>

<button class="button24" onclick="swab()" type="button">Смотреть видео</button>

<div id="overlay">

<div class="popup">

<button class="close" title="Закрыть окно" onclick="swa2b()"></button>

<p class="zag">Видео разработки</p>

<video class="modVid" controls="controls" poster="./img/posterVid.jpg">

<source src="./video/intro\_movie.mp4" type='video/mp4; codecs="avc1.42E01E, mp4a.40.2"'>

</video>

</div>

</div>

<div class="sim-slider poss">

<ul class="sim-slider-list">

<li><img src="http://pvbk.spb.ru/inc/slider/imgs/no-image.gif" alt="screen"></li> <!-- это экран -->

<li class="sim-slider-element"><img src="./img/1.jpg" alt="0"></li>

<li class="sim-slider-element"><img src="./img/2.jpg" alt="1"></li>

<li class="sim-slider-element"><img src="./img/3.jpg" alt="2"></li>

</ul>

<div class="sim-slider-arrow-left"></div>

<div class="sim-slider-arrow-right"></div>

<div class="sim-slider-dots"></div>

</div>

</section>

В соответствии с шаблоном был написан Main:

Листинг 3.2- Исходный код Footer документа index.html

<footer class=" page-footer">

<div class="container">

<span>© naaZzz</span>

<span class="container container1">2021</span>

</div>

</footer>

* 1. **Описание основных функций**

В соответствии с шаблоном были написаны и реализованы функции для работы сайта:

* в файле scriptTheme.js реализована смена темы на сайте;
* в файле list.js реализована сортировка и поиск;
* в файле scriptWind.js реализовано модальное окно;
* в файле script.js реализован алгоритм для функционирования слайдера.
  1. **Вывод по главе**

В данной главе были описаны и продемонстрированы выбранная среда и средства разработки, было произведено точное описание основных страниц сайта, одна из них была разобрана подробно с приведением кода программного средства, а также приведено краткое описание основных функций для работоспособности сайта.

1. **Тестирование**

В соответствии с планом по разработке сайта было проведено тестирование программного средства на выявление ошибок и неточностей в работе.

* 1. **Описание процесса тестирования**

Тестирование производилось в 3 этапа:

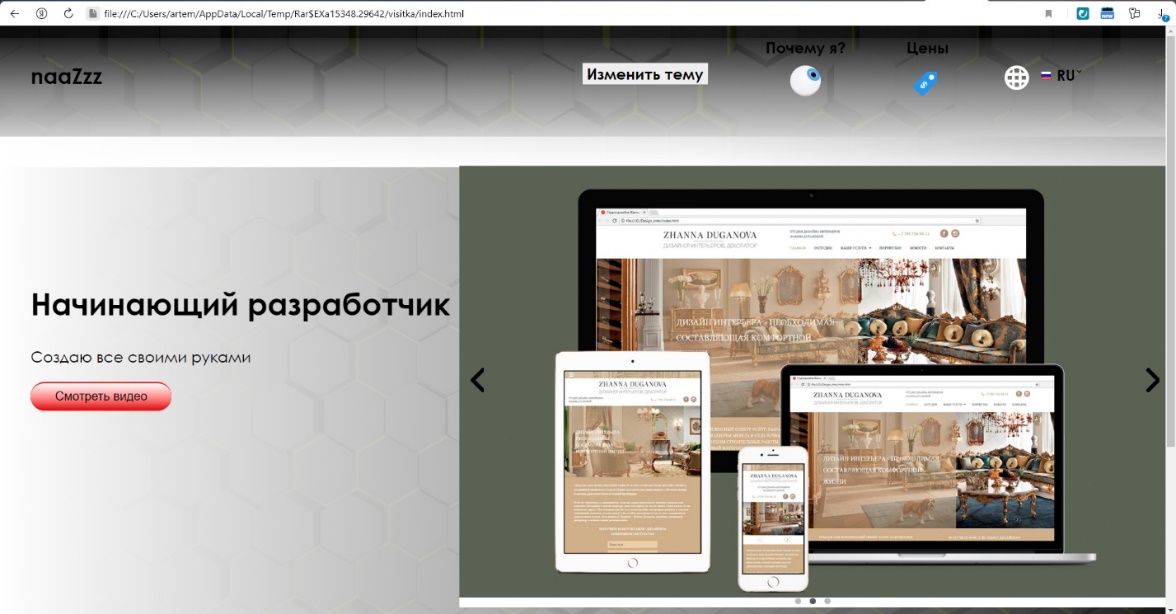
* Тестирование функционала

Рисунок 4.1 - Готовая главная страница сайта с применением темной темы

Были проверены на работоспособность все ссылки и переходы в соответствии с требованиями технического задания, об этом можно убедится на рисунках 4.1,4.2.,4.3. Тестирование производилось сразу на нескольких браузерах, таких как Yandex Browser, Google Chrome, Opera GX.

* Тестирование верстки

Было проверено на соответствие реализация дизайна предоставляемого

разработчиком с шаблоном необходимым для реализации (предоставлено на рисунке 4.2, расположение элементов, цветовые схемы, наличие дизайнерских элементов и кнопок).

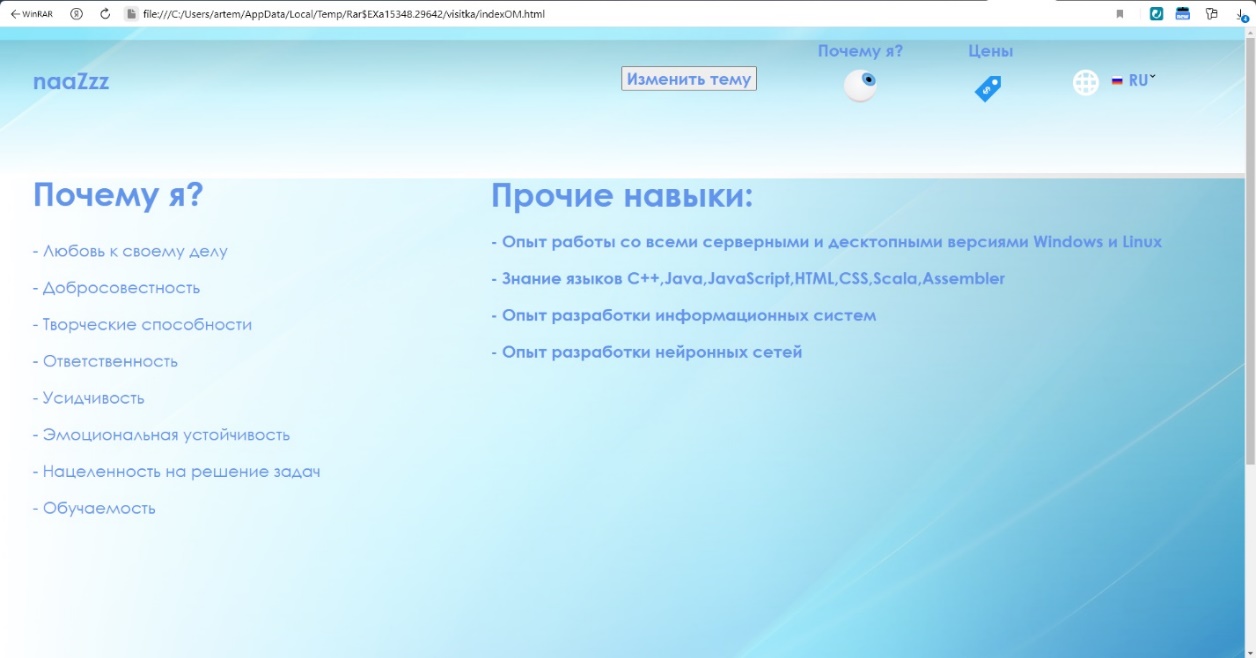
 Была произведена проверка на валидность кода (соответствие общепринятым стандартам).

Рисунок 4.2 Готовая страница файла indexOM.html

* Тестирование юзабилити

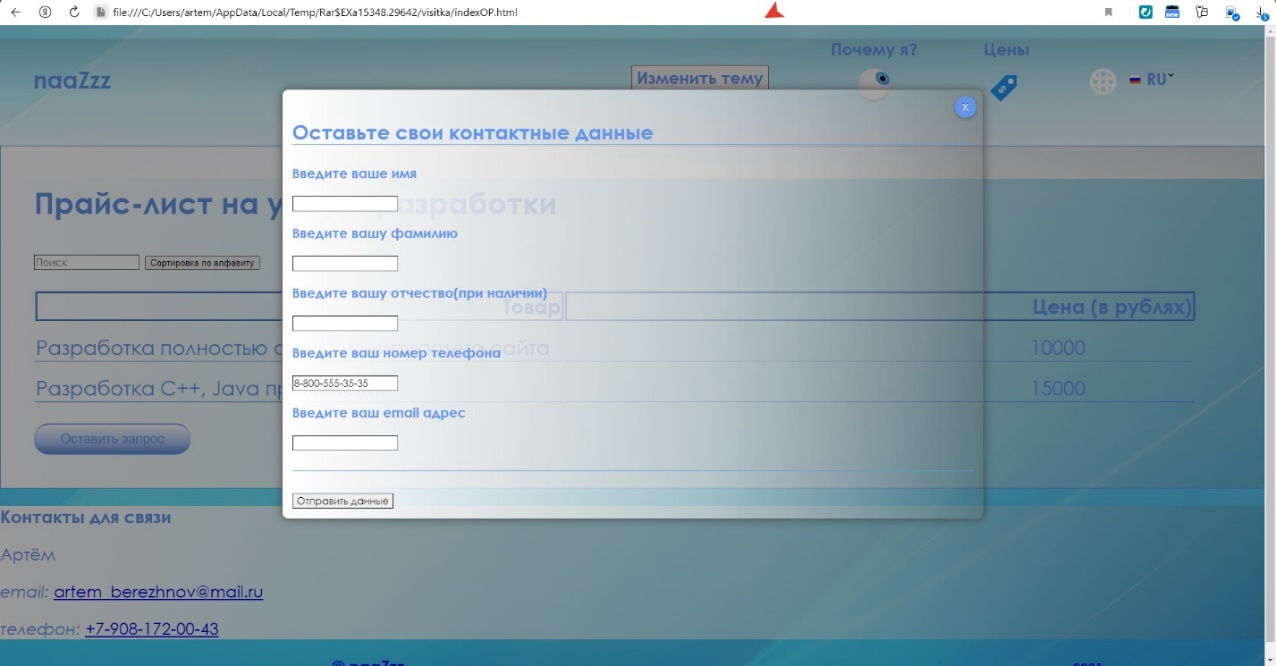
Каждое реализованное на практике решение было проверено на работоспособность, также произведена проверка всех кнопок и всплывающих окон на работоспособность, это можно наблюдать на рисунке 4.3.

Рисунок 4.3 Модальное окно файла indexOP.html

* 1. **Выводы по главе**

При проведении тестирования программного средства не было выявлено неточностей, ошибок или багов. Все страницы сайта работают в правильной форме в соответствии с техническим заданием.

**Заключение**

В рамках данной работы был создан сайт-визитка на тему «Визитка начинающего разработчика».

В ходе выполнения работы был проведен аналитический обзор предметной области, составлен план реализации программного средства, произведено проектирование сайта с описанием структур страниц, произведено программное конструирование, разработаны текстовое и стилевое оформление сайта.

Был построен аналитически обоснованный алгоритм решения поставленной задачи и разработано соответствующее программное средство, конечный исходный код которого можно просмотреть в Приложении Б. Во время написания программного средства была выполнены большая часть поставленных задач и на их основе, была выполнена проверка корректной работоспособности программного средства.

Реализованы алгоритмы:

* алгоритм слайдера;
* алгоритм смены темы;
* алгоритмы поиска и сортировки;
* алгоритм, реализующий модальное окно.

Произведено тестирование программного средства. Получены и закреплены навыки разработки программного средства на языке программирования HTML, CSS, JavaScript в среде Microsoft Visual Studio.

# Перечень использованных информационных ресурсов

1. DENVER [Электронный ресурс] - Официальный сайт. URL: http://www.denwer.ru/ (дата обращения 11.07.2021).
2. AltBlog [Электронный ресурс] - Официальный сайт. URL: https://altblog.ru/ (дата обращения 11.07.2021).
3. Википедия [Электронный ресурс] - Официальный сайт. URL: https:// ru.wikipedia.org/ (дата обращения 11.07.2021).
4. Yandex.вопросы [Электронный ресурс] - Официальный сайт. URL: https:// yandex.ru/q/question/ (дата обращения 11.07.2021).
5. Википедия [Электронный ресурс] - Официальный сайт. URL: https://ru.wikipedia.org/wiki/JavaScript (дата обращения 11.07.2021).
6. Microsoft Visual Studio // Образовательный онлайн-сервис WIKIPEDIA [Электронный ресурс], редактировано 04.06.2020. URL.: <https://ru.wikipedia.org/wiki/Microsoft_Visual_Studio#Visual_Studio_2019> (дата обращения 11.07.2021)

# Приложение А

# Техническое задание

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Доцент каф. «ПОВТиАС»  \_\_\_\_\_\_\_\_\_\_\_/Скляренко А.А  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_2021г. |  |

А.1 Введение

А.1.1 Наименование программного средства

Наименование программного средства – «Сайт-визитка начинающего разработчика».

**А.1.2 Область** применения

Программное средство может применятся в коммерческой сфере для подачи информации в удобной форме, для легкого и быстрого привлечения клиентов в интернет пространстве.

А.2 Основание для разработки

Разработка ведется на основании документа «Учебный план для студентов ВУЗа» направление 09.03.04 «Программная инженерия» кафедры «Программное обеспечение вычислительное техники и автоматизированных систем» факультета «Информатика и вычислительная техника» Донского Государственного Технического Университета.

А.3 Назначение разработки

А.3.1 Функциональное назначение

Функциональное назначение программного средства заключается в предоставлении пользователю всей наглядной информации, установленной по желанию клиента, а также контактных данных.

А.3.2 Эксплуатационное назначение

Эксплуатационное назначение состоит в использовании программного средства на персональном компьютере (ПК) с операционной системой windows.

А.4 Требования к программе

А.4.1 Требования к функциональным характеристикам

Программное средство должно осуществлять следующие функции:

* слайдер;
* смена темы страницы;
* модульное окно;
* сортировка и поиск товаров.

А.4.2 Требования к надежности

Для надежной работы программного средства необходимо, чтобы выполнялись следующие условия:

* бесперебойное питание технического средства, на котором находится продукт;
* регулярная проверка программного средства на наличие вирусов;
* отсутствие шума и спокойная обстановка.

А.4.2.1 Входные данные

В качестве входных данных представляется текстовая и графическая информация.

А.4.2.2 Выходные данные

Выходная информация представляет собой главную страницу сайта, которая в свою очередь отображает деятельность сайта и содержание

А.4.3 Условия эксплуатации

Для стабильного функционирования и оптимальной работы программного продукта необходимо соблюдение всех требований и правил эксплуатации вычислительной техники. Каких-либо требований к пользователю данного приложения нет.

А.4.4 Требование к составу и параметрам технических средств

В состав технических средств должен входить ПК с операционной системой windows 7 и выше, включающая в себя:

* 64-разрядный процессор;
* оперативная память объемом не менее 1 Гбайт.
* Использование современного браузера (YandexBrowser, Google Chrome и т.д.)

Дополнительные требования и ограничения к составу и параметрам технических средств не вводится.

А.4.5 Требования к исходным кодам и языкам программирования

Программное средство должно быть реализовано с применением языка программирования JavaScript, для создания интерфейса должен использоваться язык разметки HTML и язык стиля CSS.

А.4.6 Требования к упаковке и маркировки

К упаковке и маркировки специальных требований не предъявляются.

А.4.7 Требования к транспортировке и хранению

Условия транспортирования, места хранения, условия складирования и сроки хранения в различных условиях должны соответствовать требованиям, предъявляемым к носителям информации, на которых будет содержаться данное программное изделие. Программное средство может храниться на любых цифровых носителях информации (жесткий диск, компакт – диск, флэш накопитель и т. п.).

А.5 Требование к программной документации

Программная документация состоит из следующего:

* титульный лист;
* лист задания на учебную ознакомительную практику;
* пояснительная записка к производственной практике;
* техническое задание по ГОСТ 19.201-78 ЕСПД;
* исходный код программного средства по ГОСТ 19.401-79 ЕСПД.

А.6 Стадии и этапы разработки

Реализация программного средства состояла из следующих этапов:

* постановка задачи (22.06.2021);
* изучение предметной области (23.06.2021 – 25.06.2021);
* алгоритмическое конструирование (26.06.2021 – 1.07.2021);
* программная реализация (2.07.2021 – 11.07.2021);
* тестирование приложения (12.07.2021 – 13.07.2021);
* разработка отчета (14.07.2021 – 16.07.2021).

А.7 Порядок и контроль приемки

Порядок и контроль приемки определяются заведующим кафедрой «ПОВТ и АС», подразумевающие собой демонстрацию показателя владения средствами для разработки программных средств в различных направлениях.

Главным требованием к приемке является наличие корректного работающего программного средства и отчета, предоставленного в печатном виде.

|  |  |
| --- | --- |
| Разработчик технического задания:  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2021 г. | /Бережнов Артем Сергеевич/  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

# Приложение Б

# Исходный код программного средства

Листинг Б.1 – Исходный код script.js

function Sim(sldrId) {

let id = document.getElementById(sldrId);

if (id) {

this.sldrRoot = id

}

else {

this.sldrRoot = document.querySelector('.sim-slider')

};

// Carousel objects

this.sldrList = this.sldrRoot.querySelector('.sim-slider-list');

this.sldrElements = this.sldrList.querySelectorAll('.sim-slider-element');

this.sldrElemFirst = this.sldrList.querySelector('.sim-slider-element');

this.leftArrow = this.sldrRoot.querySelector('div.sim-slider-arrow-left');

this.rightArrow = this.sldrRoot.querySelector('div.sim-slider-arrow-right');

this.indicatorDots = this.sldrRoot.querySelector('div.sim-slider-dots');

// Initialization

this.options = Sim.defaults;

Sim.initialize(this)

};

Sim.defaults = {

// Default options for the carousel

loop: true, // Бесконечное зацикливание слайдера

auto: true, // Автоматическое пролистывание

interval: 5000, // Интервал между пролистыванием элементов (мс)

arrows: true, // Пролистывание стрелками

dots: true // Индикаторные точки

};

Sim.prototype.elemPrev = function (num) {

num = num || 1;

let prevElement = this.currentElement;

this.currentElement -= num;

if (this.currentElement < 0) this.currentElement = this.elemCount - 1;

if (!this.options.loop) {

if (this.currentElement == 0) {

this.leftArrow.style.display = 'none'

};

this.rightArrow.style.display = 'block'

};

this.sldrElements[this.currentElement].style.opacity = '1';

this.sldrElements[prevElement].style.opacity = '0';

if (this.options.dots) {

this.dotOn(prevElement); this.dotOff(this.currentElement)

}

};

Sim.prototype.elemNext = function (num) {

num = num || 1;

let prevElement = this.currentElement;

this.currentElement += num;

if (this.currentElement >= this.elemCount) this.currentElement = 0;

if (!this.options.loop) {

if (this.currentElement == this.elemCount - 1) {

this.rightArrow.style.display = 'none'

};

this.leftArrow.style.display = 'block'

};

this.sldrElements[this.currentElement].style.opacity = '1';

this.sldrElements[prevElement].style.opacity = '0';

if (this.options.dots) {

this.dotOn(prevElement); this.dotOff(this.currentElement)

}

};

Sim.prototype.dotOn = function (num) {

this.indicatorDotsAll[num].style.cssText = 'background-color:#BBB; cursor:pointer;'

};

Sim.prototype.dotOff = function (num) {

this.indicatorDotsAll[num].style.cssText = 'background-color:#556; cursor:default;'

};

Sim.initialize = function (that) {

// Constants

that.elemCount = that.sldrElements.length; // Количество элементов

// Variables

that.currentElement = 0;

let bgTime = getTime();

// Functions

function getTime() {

return new Date().getTime();

};

function setAutoScroll() {

that.autoScroll = setInterval(function () {

let fnTime = getTime();

if (fnTime - bgTime + 10 > that.options.interval) {

bgTime = fnTime; that.elemNext()

}

}, that.options.interval)

};

// Start initialization

if (that.elemCount <= 1) { // Отключить навигацию

that.options.auto = false; that.options.arrows = false; that.options.dots = false;

that.leftArrow.style.display = 'none'; that.rightArrow.style.display = 'none'

};

if (that.elemCount >= 1) { // показать первый элемент

that.sldrElemFirst.style.opacity = '1';

};

if (!that.options.loop) {

that.leftArrow.style.display = 'none'; // отключить левую стрелку

that.options.auto = false; // отключить автопркрутку

}

else if (that.options.auto) { // инициализация автопрокруки

setAutoScroll();

// Остановка прокрутки при наведении мыши на элемент

that.sldrList.addEventListener('mouseenter', function () { clearInterval(that.autoScroll) }, false);

that.sldrList.addEventListener('mouseleave', setAutoScroll, false)

};

if (that.options.arrows) { // инициализация стрелок

that.leftArrow.addEventListener('click', function () {

let fnTime = getTime();

if (fnTime - bgTime > 1000) {

bgTime = fnTime; that.elemPrev()

}

}, false);

that.rightArrow.addEventListener('click', function () {

let fnTime = getTime();

if (fnTime - bgTime > 1000) {

bgTime = fnTime; that.elemNext()

}

}, false)

}

else {

that.leftArrow.style.display = 'none'; that.rightArrow.style.display = 'none'

};

if (that.options.dots) { // инициализация индикаторных точек

let sum = '', diffNum;

for (let i = 0; i < that.elemCount; i++) {

sum += '<span class="sim-dot"></span>'

};

that.indicatorDots.innerHTML = sum;

that.indicatorDotsAll = that.sldrRoot.querySelectorAll('span.sim-dot');

// Назначаем точкам обработчик события 'click'

for (let n = 0; n < that.elemCount; n++) {

that.indicatorDotsAll[n].addEventListener('click', function () {

diffNum = Math.abs(n - that.currentElement);

if (n < that.currentElement) {

bgTime = getTime(); that.elemPrev(diffNum)

}

else if (n > that.currentElement) {

bgTime = getTime(); that.elemNext(diffNum)

}

// Если n == that.currentElement ничего не делаем

}, false)

};

that.dotOff(0); // точка[0] выключена, остальные включены

for (let i = 1; i < that.elemCount; i++) {

that.dotOn(i)

}

}

};

new Sim();

Листинг Б.2 – исходный код scriptWind.js

var b = document.getElementById('overlay');

function swab() {

b.style.visibility = 'visible';

b.style.opacity = '1';

b.style.transition = 'all 0.7s ease-out 0s';

}

function swa2b() {

b.style.visibility = 'hidden';

b.style.opacity = '0';

}

var d = document.getElementById('overlay-1');

function swad() {

d.style.visibility = 'visible';

d.style.opacity = '1';

d.style.transition = 'all 0.7s ease-out 0s';

}

function swa2d() {

d.style.visibility = 'hidden';

d.style.opacity = '0';

}

Листинг Б.3 – исходный код scriptTheme.js

let page = document.querySelector('.page');

let themeButton = document.querySelector('.theme-button');

themeButton.onclick = function () {

page.classList.toggle('light-theme');

page.classList.toggle('dark-theme');

};