

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего
образования
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА 42

КУРСОВАЯ РАБОТА (ПРОЕКТ)
ЗАЩИЩЕНА С ОЦЕНКОЙ
РУКОВОДИТЕЛЬ

Доцент, к.т.н.		Красильникова О.А.
должность, уч. степень, звание	подпись, дата	инициалы, фамилия

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К КУРСОВОЙ РАБОТЕ

по дисциплине: Web-технологии

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ гр. №	4126	28.05.2024	Степанова Я.И.
		подпись, дата	инициалы, фамилия

Санкт-Петербург 2024

СОДЕРЖАНИЕ

Введение	3
1. Дизайн сайта.....	4
1.1 Модель сайта	4
1.2 Макет сайта	4
1.3 Применение технологии каскадных таблиц стилей.....	8
1.4 Графическое оформление веб-страницы	15
1.5 Создание навигационных панелей для сайта	17
2. Web-сценарии сайта на языке JavaScript	19
2.1 Назначение JavaScript	19
2.2 Разработка JS- скриптов для сайта.....	21
3. Валидация кода и продвижение сайта.	25
3.1 Валидация кода сайта	25
3.2 Способы продвижения сайтов.....	27
3.3 Используемые приемы продвижения сайта	30
Заключение.....	31
ПРИЛОЖЕНИЕ 1	33
ПРИЛОЖЕНИЕ 2	37

Введение

Целью курсовой работы является закрепление навыков создания web-сайтов и использование современных технологий web-дизайна.

Сайт-портфолио посвящен демонстрации моих художественных работ, охватывающих различные стили и техники рисования. Цель создания сайта заключается в представлении моего творчества широкой аудитории, что в будущем может помочь в привлечении потенциальных клиентов и работодателей, а также в установлении профессиональных контактов. Сайт позволяет пользователям просматривать мои рисунки, оценивать разнообразие моих навыков и находить информацию обо мне и способах связи.

Основные функции сайта включают:

- Просмотр галереи портретов.
- Ознакомление с рисунками, выполненными в различных стилях.
- Просмотр моих набросков.
- Получение информации обо мне, моем творческом пути и контактной информации.

1. Дизайн сайта

1.1 Модель сайта

На рисунке 1 представлена схема сайта, где обозначены связи между отдельными страницами.

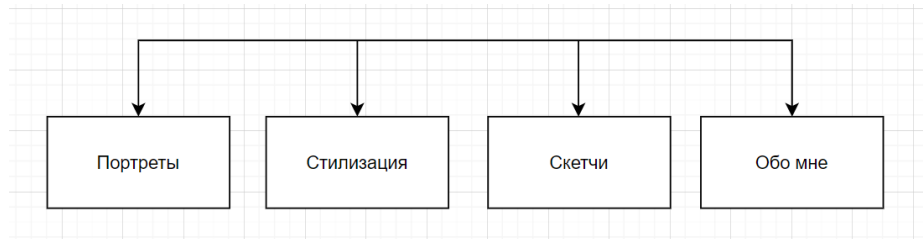


Рисунок 1 – Схема сайта

При загрузке сайта открывается категория Портреты. Далее пользователь может перейти на любую категорию из указанных на рисунке 1.

Главная страница сайта представляет галерею портретов, выполненных в одной технике. Посетители могут увидеть работы в высоком качестве, оценивая детали.

Вторая страница посвящена рисункам в разных стилях. Это позволяет продемонстрировать мою способность работать в различных художественных направлениях.

Третья страница содержит мои наброски, показывая процесс создания рисунков и мою работу с идеями и композициями.

На четвертой странице представлена информация обо мне, моем творческом пути, а также способы связи.

1.2 Макет сайта

Основной целью макета было создание интуитивно понятного интерфейса, который обеспечивает пользователю легкий доступ к информации.

Описание макета

Так как страницы сайта достаточно однотипны, было принято решение сделать макеты только для двух страниц: с портретами и страницы со стилизацией. Это позволило сосредоточиться на ключевых элементах

дизайна и функциональности, которые затем были адаптированы для остальных страниц. На рисунке 2 и 3 представлены первоначальные макет страниц с портретами и стилизацией. В ходе разработки некоторые детали были изменены для улучшения пользовательского опыта.

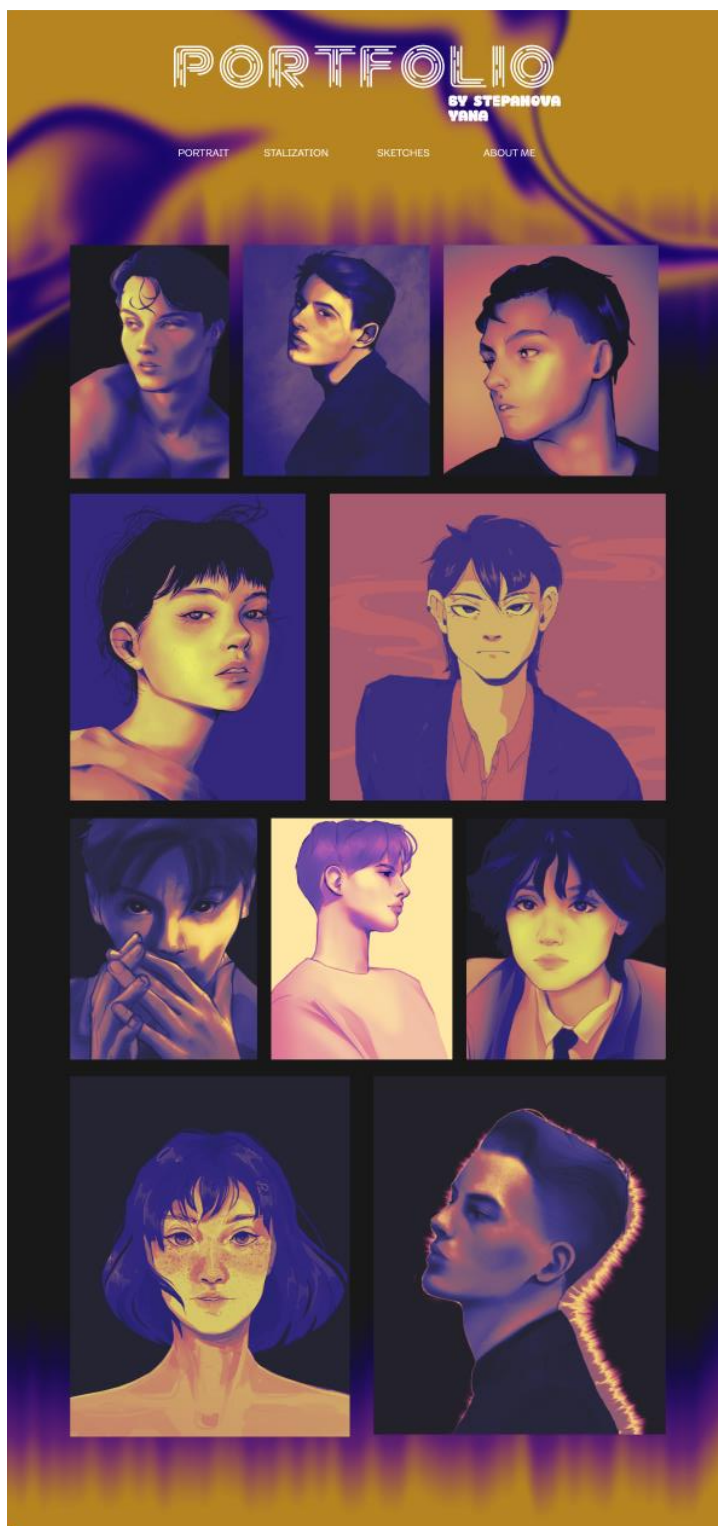


Рисунок 2 – Страница с портретами

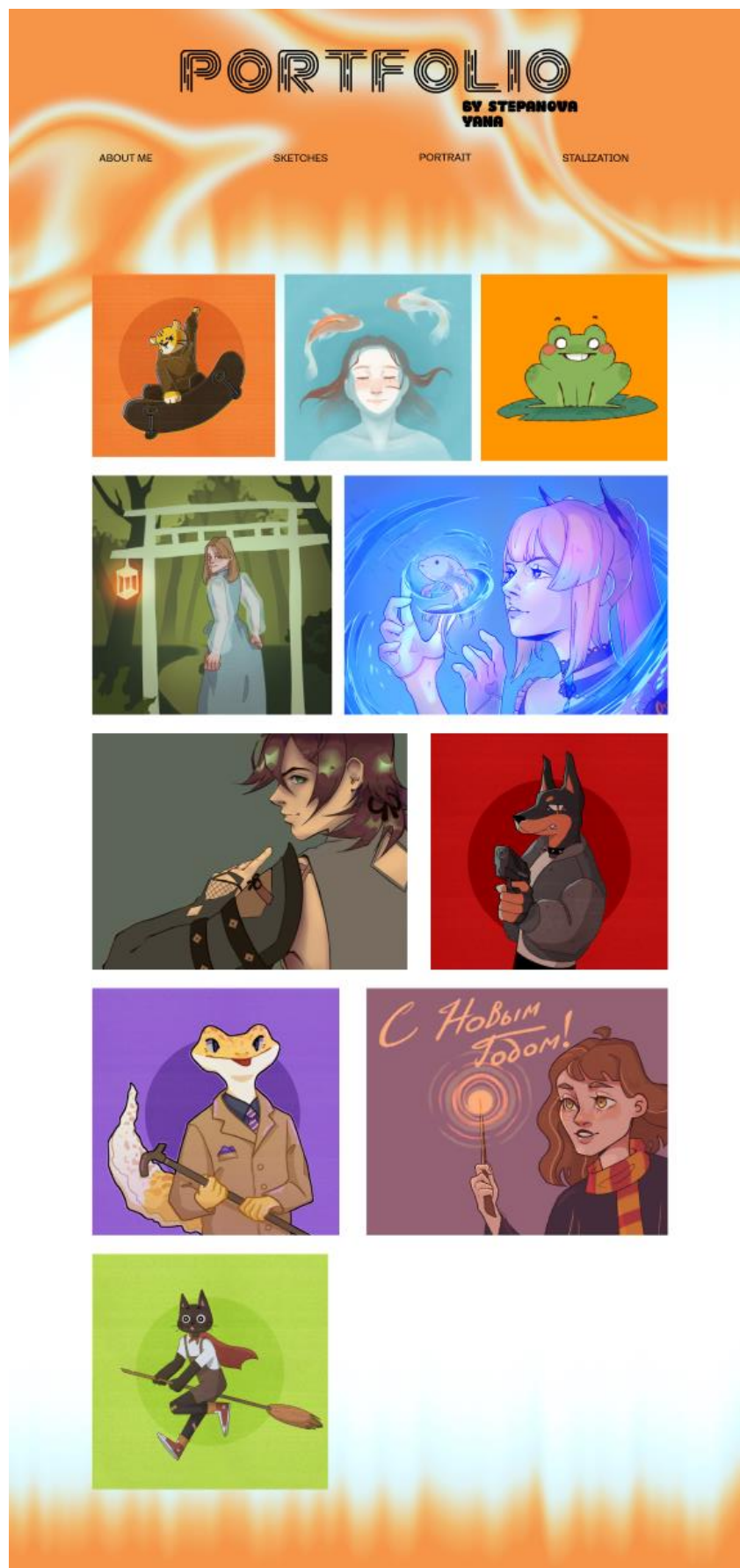


Рисунок 3 – Страница со стилизацией

Технологии и инструменты

Для создания макета использовалось приложение Figma, которое предоставило широкие возможности для прототипирования и визуализации дизайна. Основные преимущества использования Figma включают:

- Прототипирование: Легкость в создании интерактивных прототипов, что позволило предварительно тестировать пользовательские сценарии.
- Удобство использования: Интуитивно понятный интерфейс и широкий набор инструментов для дизайна интерфейсов.

Адаптивная верстка

Адаптивная верстка сайта была реализована с использованием медиа-запросов. Медиа-запросы — это CSS-правила, которые позволяют изменять стили элементов в зависимости от характеристик устройства, таких как ширина экрана, тип устройства и другие параметры.

Процесс создания медиа-запроса начинается с ключевого слова `@media`, после которого указываются одно или несколько условий. Эти условия могут включать тип устройства или конкретные характеристики, такие как ширина экрана. Характеристики записываются в круглых скобках.

Стили, указанные внутри `@media`, будут применяться только если условие выполняется. Вот пример медиа-запросов, которые активируются при ширине экрана менее 800px и 600px:

```
@media (max-width: 800px) {  
  header {  
    font-size: 70%;  
  }  
  
  .gallery-item {  
    max-width: calc(50% - 20px);  
  }  
  nav {  
    text-align: center;  
    margin: 20px 0;  
    font-size: 70%;  
  }  
}
```

```

}

@media (max-width: 600px) {
  header {
    font-size: 60%;
  }
  nav li {
    display: block;
    margin: 10px 0;
  }

  .gallery-item {
    flex: 1 100%;
    max-width: 100%;
  }
}

```

1.3 Применение технологии каскадных таблиц стилей

Основные стили

```

body {
  display: none;
  margin: 0;
  font-family: Dongo;
  color: #ffffff;
  background-color: #181818;
  position: relative;
}

```

В этом фрагменте устанавливаются основные стили для body, включая шрифт, цвет текста, фоновый цвет и относительное позиционирование. Использование display: none скрывает весь контент до тех пор, пока не будет изменено это свойство, что будет в дальнейшем использоваться для анимации.

Фоновые изображения

```

.background-container {
  position: absolute;
  top: 0;
  left: 0;
  width: 100%;
  height: auto;
  z-index: -1;
  pointer-events: none;
}

```



```
}  
  
.background-img {  
  position: absolute;  
  width: 100%;  
  height: auto;  
  opacity: 1;  
}
```

В этом фрагменте создается контейнер для фоновых изображений с абсолютным позиционированием и отрицательным z-index, чтобы они находились позади основного контента. Использование pointer-events: none предотвращает взаимодействие с фоновыми изображениями.

Галерея

```
.gallery {  
  display: flex;  
  flex-wrap: wrap;  
  gap: 40px;  
  justify-content: center;  
  position: relative;  
}  
  
.gallery-item {  
  max-width: calc(30% - 20px);  
  position: relative;  
}  
  
.gallery-item img {  
  width: 100%;  
  height: auto;  
  border-radius: 10px;  
  object-fit: cover;  
  display: block;  
  transition: transform 0.7s ease;  
}
```

Используется flex для создания галереи с равномерным распределением элементов и промежутком между ними. У изображений добавлен плавный переход для эффекта увеличения при наведении, что улучшает интерактивность.

Интерактивность

Интерактивность достигается использованием псевдоклассов и CSS-анимаций. Например, при наведении на элементы галереи (.gallery-item:hover) изображение увеличивается за счет использования свойства `transform: scale(1.2)`, что привлекает внимание пользователя и улучшает взаимодействие с контентом.

```
.gallery-item:hover img {  
  transform: scale(1.2);  
}
```

На рисунке 4 представлен вид элемента в обычном состоянии, а на рисунке 5 представлен вид изображений при наведении.

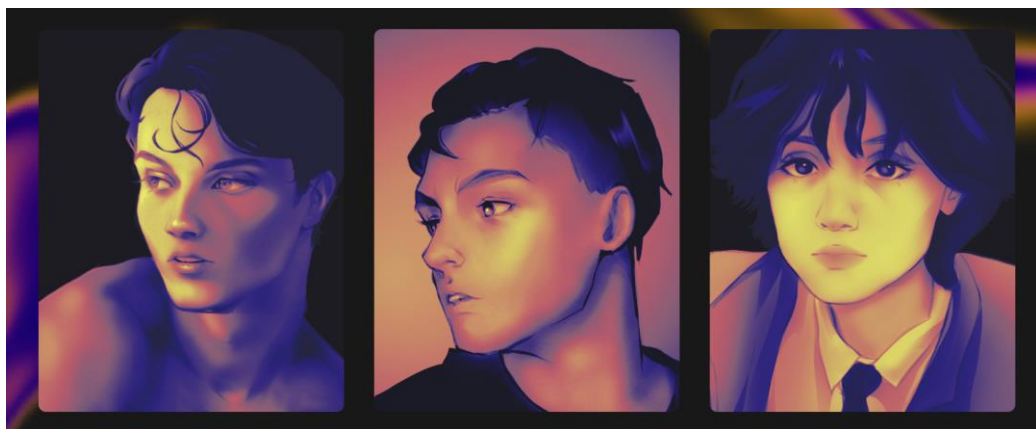


Рисунок 4 – Обычное состояние изображений

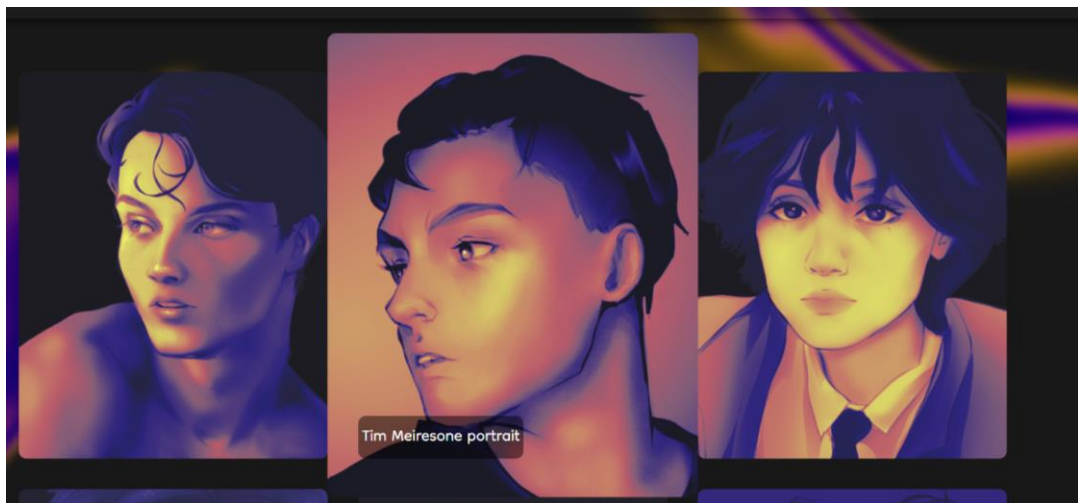


Рисунок 5 – Изображение при наведении

При наведении на категории используется такая же технология.

Реализация адаптивной верстки

На данном сайте применяются различные подходы к дизайну для обеспечения оптимального отображения на различных устройствах:

Для реализации резинового дизайна ширина элементов задана в процентах. Это позволяет элементам адаптироваться к размеру экрана, сохраняя пропорции. В некоторых случаях дополнительно задаются максимальная и минимальная ширина, чтобы элементы не становились слишком маленькими или большими

```
.background-container {  
  width: 100%;  
}  
  
.gallery-item {  
  max-width: calc(30% - 20px);  
}
```

Адаптивный дизайн реализован с помощью медиа-запросов, которые изменяют стили элементов в зависимости от ширины экрана. Это позволяет сайтам адаптироваться к различным размерам экранов и устройствам:

```
@media (max-width: 800px) {  
  header {  
    font-size: 70%;  
  }  
  
  .gallery-item {  
    max-width: calc(50% - 20px);  
  }  
  
  nav {  
    text-align: center;  
    margin: 20px 0;  
    font-size: 70%;  
  }  
}  
  
@media (max-width: 600px) {  
  header {  
    font-size: 60%;  
  }  
  
  nav li {  
    display: block;  
    margin: 10px 0;  
  }  
}
```

```

.gallery-item {
  flex: 1 100%;
  max-width: 100%;
}
}

```

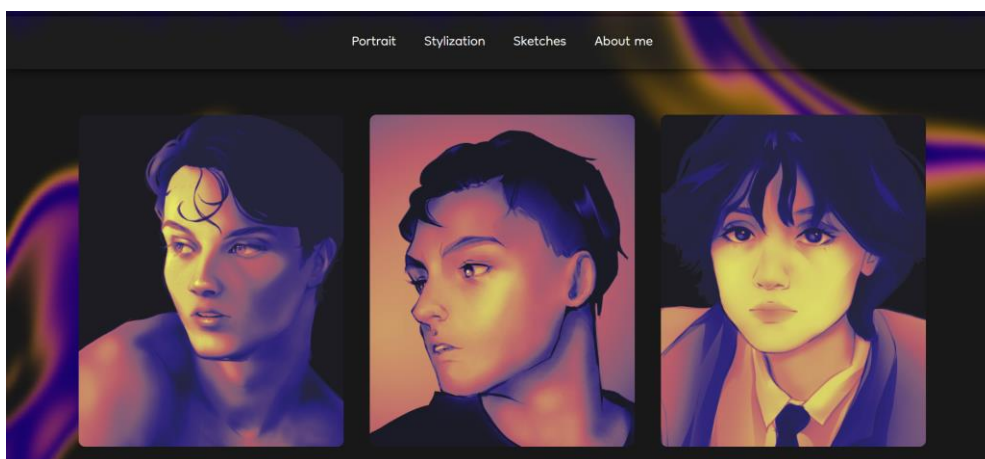


Рисунок 6 – При ширине больше 800px

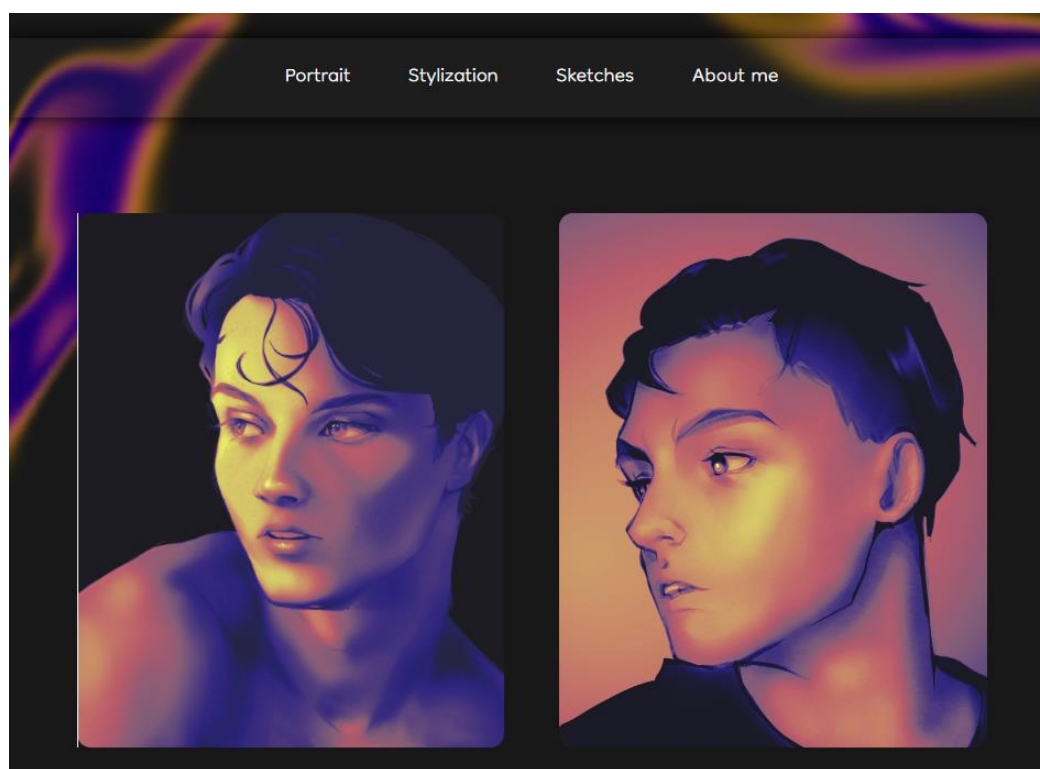


Рисунок 7 – При ширине меньше 800px

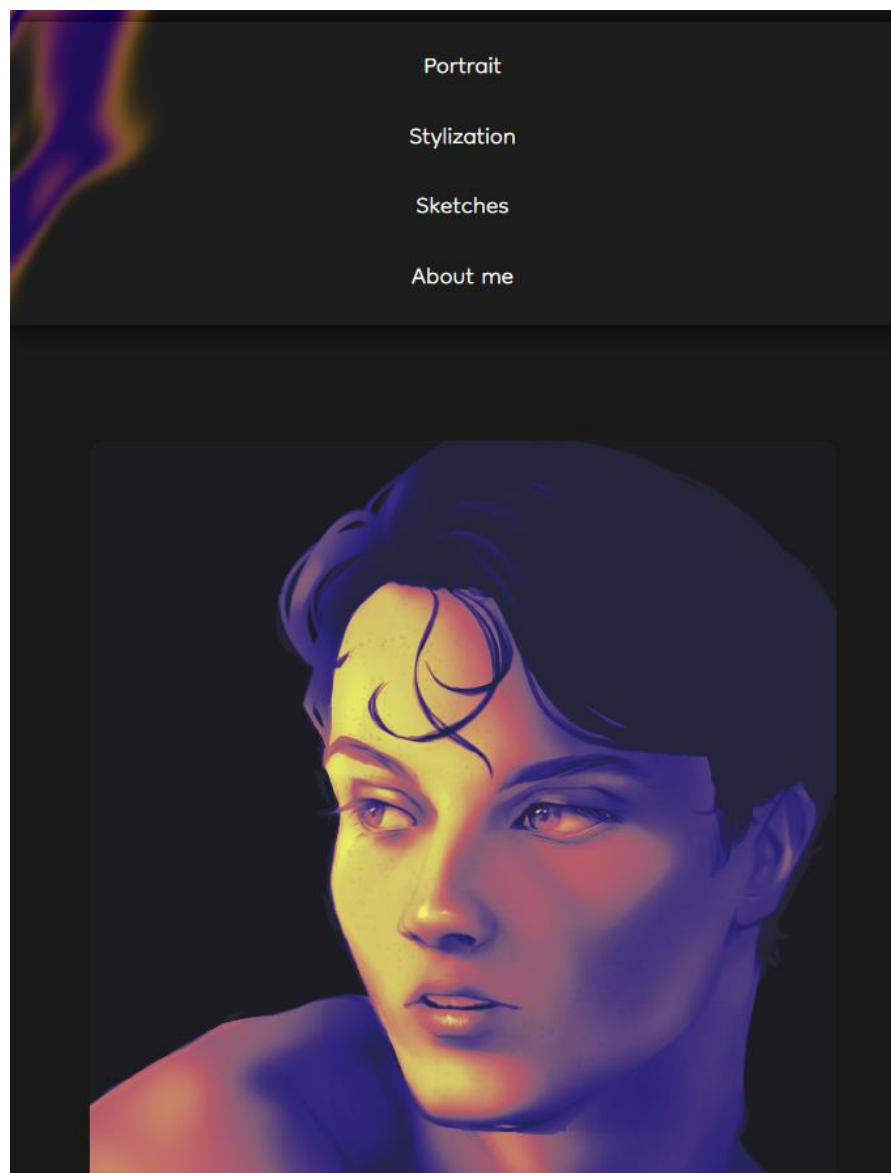


Рисунок 8 – При ширине меньше 600px

Обеспечение кроссбраузерности веб-сайта

Кроссбраузерность – это способность сайта одинаково успешно адаптироваться под разные браузеры, корректно и без сбоев в функционировании отображаться в них.

Для обеспечения кроссбраузерности важно использовать CSS-свойства, поддерживаемые большинством современных браузеров. В данном коде применяются основные свойства CSS, которые поддерживаются большинством браузеров. Однако, для полной кроссбраузерности рекомендуется:

- Использовать префиксы для CSS-свойств, которые могут отличаться в разных браузерах (например, -webkit-, -moz-, -o-).
- Проверять сайт в различных браузерах и их версиях.
- Использовать современные инструменты для тестирования кроссбраузерности, такие как BrowserStack или Sauce Labs.

1.4 Графическое оформление веб-страницы

Графика на сайте используется для различных целей, таких как фоновые изображения, иллюстрации, фотогалерея и оформление элементов страницы. Вот подробное описание использования графики и обоснование выбора форматов и методов оптимизации.

Фоновые изображения использовались для создания визуально привлекательного фона, улучшения общей эстетики сайта и структурирования пространства страницы. Например, на сайте применяются два фоновых изображения (bck1.png и bck3.png), которые размещены в контейнере .background-container. Пример фонового изображения представлен на рисунке 9

```
<div class="background-container">  
    
    
</div>
```



Рисунок 9 – Фоновые изображения

Фотогалерея используется для представления портфолио работ. Это позволяет демонстрировать иллюстрации, предоставляя пользователям возможность просматривать и оценивать работы. Галерея, представленная на странице, включает несколько изображений. На рисунке 10 представлен пример галереи.

```
<section class="gallery">  
  <div class="gallery-item" data-title="Gavin Grabowski portrait"></div>  
  <div class="gallery-item" data-title="Tim Meiresone portrait"></div>
```


...
</section>

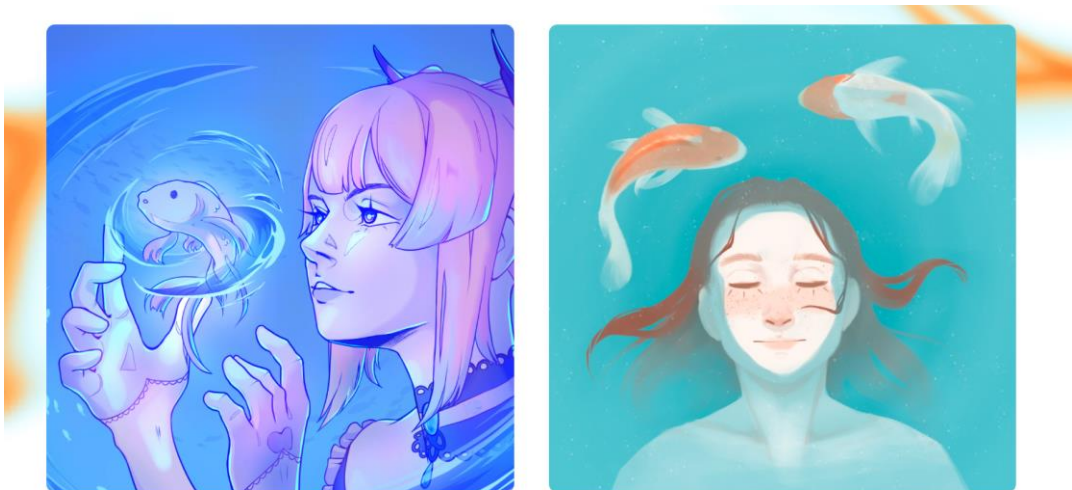


Рисунок 10 - Галерея

Форматы графических изображений были выбраны с учетом их преимуществ и особенностей. Для фоновых изображений и портретов использовались форматы PNG и JPEG. Формат PNG был выбран за его поддержку прозрачности и высокое качество изображения, что важно для фоновых элементов и детализированных иллюстраций. Формат JPEG использовался для больших фотографий, где прозрачность не требуется, но важно сохранить высокое качество при меньшем размере файла.

Оптимизация графики включала регулирование размеров изображений с помощью CSS, чтобы они правильно отображались на разных устройствах и экранах. Изображения были адаптированы под различные размеры экранов с помощью свойств `max-width` и `calc()`:

```
.gallery-item {  
  max-width: calc(30% - 20px);  
  position: relative;  
}  
  
@media (max-width: 800px) {  
  .gallery-item {  
    max-width: calc(50% - 20px);  
  }  
}  
  
@media (max-width: 600px) {
```



```
.gallery-item {  
  flex: 1 100%;  
  max-width: 100%;  
}  
}
```

1.5 Создание навигационных панелей для сайта

Навигация является ключевым элементом любого веб-сайта, так как она обеспечивает пользователям доступ к различным разделам и страницам сайта. Основные требования, предъявляемые к навигации на сайте, включают:

- **Простота и интуитивность:** Навигационная панель должна быть простой в использовании и легко понятной. Пользователи должны сразу видеть, куда они могут перейти и как это сделать.
- **Доступность:** Навигация должна быть доступной на всех устройствах и экранах. Это включает в себя адаптивность навигационной панели для корректного отображения на мобильных устройствах, планшетах и десктопах.
- **Консистентность:** Навигационная структура должна быть последовательной на всех страницах сайта, чтобы пользователи не запутались при переходе между разделами.
- **Функциональность:** Навигационная панель должна работать без сбоев и обеспечивать быстрый доступ к основным разделам сайта.

HTML структура навигационной панели:

```
<nav>  
<ul>  
  <li><a href="index.html">Portrait</a></li>  
  <li><a href="stylization.html">Stylization</a></li>  
  <li><a href="sketches.html">Sketches</a></li>  
  <li><a href="about.html">About me</a></li>  
</ul>  
</nav>
```

В этом фрагменте HTML кода, навигация представлена в виде списка (), где каждый элемент списка () содержит ссылку (<a>) на соответствующую страницу сайта.

CSS код для стилизации навигационной панели:

```
nav {  
  text-align: center;  
  margin: 20px 0;  
  padding: 1px;  
  box-shadow: 0 0 15px rgba(0, 0, 0, 1);  
  background-color: #28282850;  
}
```

```
nav ul {  
  list-style: none;  
  padding: 0;  
}
```

```
nav li {  
  display: inline;  
  margin: 0 20px;  
}
```

```
nav a {  
  text-decoration: none;  
  color: #ffffff;  
  font-size: 2em;  
}
```

```
@media (max-width: 800px) {  
  nav {  
    font-size: 70%;  
  }  
}
```

```
@media (max-width: 600px) {  
  nav li {  
    display: block;  
    margin: 10px 0;  
  }  
}
```

CSS код отвечает за внешний вид и поведение навигационной панели:

Основной стиль:

text-align: center; - центрирует текст и элементы внутри панели.

margin: 20px 0; - задает внешние отступы сверху и снизу.

padding: 1px; - добавляет внутренний отступ для панели.

`box-shadow: 0 0 15px rgba(0, 0, 0, 1);` - добавляет тень для улучшения визуального восприятия.

`background-color: #28282850;` - устанавливает полупрозрачный фоновый цвет.

Стилизация списка:

`list-style: none;` - убирает маркеры списка.

`padding: 0;` - убирает внутренние отступы у списка.

Стилизация элементов списка:

`display: inline;` - располагает элементы списка в одну линию.

`margin: 0 20px;` - добавляет отступы между элементами списка.

Стилизация ссылок:

`text-decoration: none;` - убирает подчеркивание у ссылок.

`color: #ffffff;` - устанавливает белый цвет текста.

`font-size: 2em;` - увеличивает размер шрифта.

Адаптивность:

При ширине экрана менее 800px, размер шрифта в навигационной панели уменьшается до 70%.

При ширине экрана менее 600px, элементы списка располагаются вертикально, а не горизонтально, что достигается с помощью `display: block;` и изменения отступов.

Эти фрагменты кода обеспечивают простую, доступную и интуитивно понятную навигацию, которая адаптируется к различным устройствам и экранам, улучшая пользовательский опыт на сайте.

2. Web-сценарии сайта на языке JavaScript

2.1 Назначение JavaScript

JavaScript является одним из ключевых языков программирования в web-разработке, обеспечивая интерактивность и динамическое поведение веб-страниц. Вот основные цели и функции, для которых используется JavaScript и его библиотеки в web-программировании:

Динамическое обновление контента: JavaScript позволяет обновлять содержимое веб-страницы без необходимости перезагрузки всей страницы.

Манипуляции с DOM (Document Object Model): JavaScript предоставляет возможность изменять структуру, стиль и содержание веб-страницы. Это включает добавление, удаление и изменение элементов и атрибутов HTML.

Обработка событий: JavaScript используется для управления событиями, такими как клики мыши, нажатия клавиш и движения мыши, что позволяет создавать интерактивные пользовательские интерфейсы.

Валидация форм: JavaScript используется для проверки данных, введенных пользователем в форму, до отправки этих данных на сервер. Это помогает обеспечить правильность и полноту введенной информации.

Асинхронные запросы (AJAX): JavaScript позволяет отправлять и получать данные с сервера асинхронно, без перезагрузки страницы. Это делает возможным создание динамических и интерактивных веб-приложений.

Анимации и эффекты: JavaScript и его библиотеки, такие как jQuery, позволяют создавать анимации и визуальные эффекты на веб-страницах, что делает пользовательский интерфейс более привлекательным и интерактивным.

Использование библиотек и фреймворков:

JavaScript поддерживает множество библиотек и фреймворков, таких как React, Angular, Vue.js, которые упрощают разработку сложных веб-приложений, улучшая производительность и упрощая процесс разработки.

React: библиотека для создания пользовательских интерфейсов, позволяет создавать компоненты, управлять состоянием и эффективно обновлять DOM.

Angular: фреймворк для создания веб-приложений, предоставляет средства для разработки SPA (Single Page Applications) с использованием TypeScript.

Vue.js: прогрессивный фреймворк для создания пользовательских интерфейсов, позволяет создавать компоненты и управлять состоянием приложения.

Таким образом, JavaScript и его библиотеки играют важную роль в web-программировании, предоставляя инструменты для создания динамичных, интерактивных и пользовательских интерфейсов, улучшая общий пользовательский опыт и функциональность веб-приложений.

2.2 Разработка JS- скриптов для сайта

Интерактивность веб-сайта обеспечивается с помощью языка JavaScript и библиотеки jQuery. Реализованные сценарии включают в себя плавное появление элементов при загрузке страницы, создание пульсирующего эффекта для заголовков, отображение названий изображений при наведении на них курсора и увеличение текста навигационных ссылок при наведении. Эти сценарии направлены на улучшение пользовательского опыта, делая взаимодействие с сайтом более привлекательным и интуитивно понятным.

Плавное появление элементов и пульсирующий эффект

Для создания плавного появления элементов при загрузке страницы и пульсирующего эффекта для заголовков используется следующий JavaScript код в файле fade.js:

```
$(document).ready(function() {  
    // Плавное появление элементов  
    $('h1fade').fadeIn(4000);  
    $('.pfade').fadeIn(5000);  
    $('body').fadeIn(2000);  
  
    // Пульсирующий эффект для заголовков  
    function pulsate() {  
        $('h1').animate({ opacity: 0.4 }, 600)  
            .animate({ opacity: 1.0 }, 800, pulsate);  
    }  
    pulsate();  
});
```

На рисунке 11 и 12 представлены кадры пульсирующего эффекта заголовка



Рисунок 11 – Кадр пульсации заголовка



Рисунок 12 – Кадр пульсации заголовка

Этот код выполняет следующие функции:

Плавно отображает элементы с классами `h1fade`, `pfade` и `body` при загрузке страницы с различной продолжительностью (4, 5 и 2 секунды соответственно).

Реализует пульсирующий эффект для заголовков (`h1`), изменяя их прозрачность от 1 до 0.4 и обратно, непрерывно повторяя анимацию.

Отображение названий изображений при наведении

Для отображения названий изображений в галерее при наведении курсора используется следующий JavaScript код в файле `gallery.js`:

```
$(document).ready(function(){
    $(".gallery-item").mouseenter(function(){
        var title = $(this).data("title");
        $(this).append("<div class='image-title'>" + title + "</div>");
        $(this).find(".image-title").fadeIn(); // Плавное появление названия
изображения
    }).mouseleave(function(){
        $(this).find(".image-title").fadeOut(); // Плавное исчезновение названия
изображения
    });
});
```

Этот код выполняет следующие функции:

При наведении курсора на элемент с классом `gallery-item` отображает название изображения, взятое из атрибута `data-title`, создавая новый элемент с классом `image-title`.

Использует метод `fadeIn` для плавного появления названия изображения.

При уходе курсора с элемента с классом `gallery-item` использует метод `fadeOut` для плавного исчезновения названия изображения.

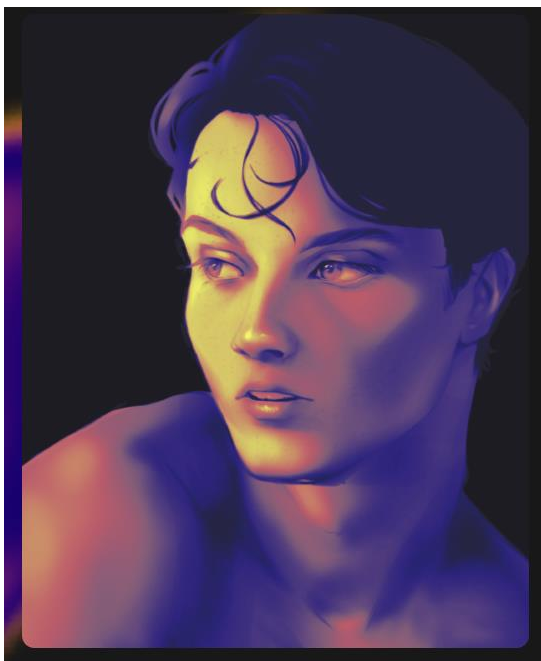


Рисунок 13 – Изображение в обычном состоянии

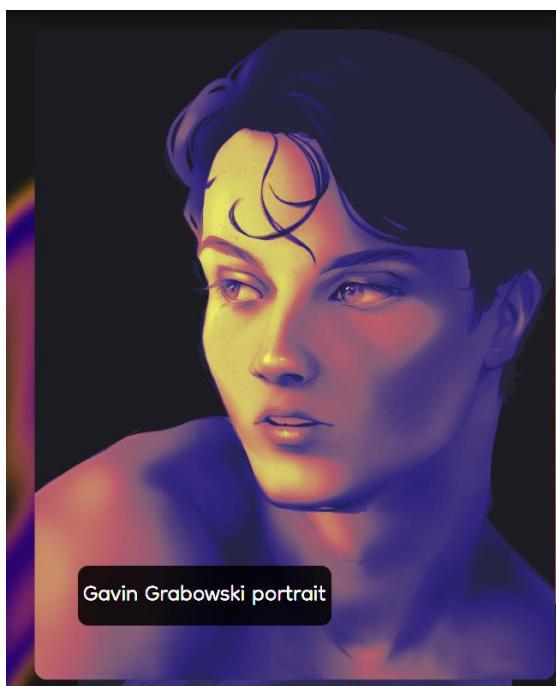


Рисунок 14 – Всплывающий текст при наведении на картинку

Увеличение текста навигационных ссылок при наведении

Для создания эффекта увеличения текста навигационных ссылок при наведении курсора используется следующий JavaScript код в файле linkHover.js:

```
$(document).ready(function(){
  $("nav ul li a").hover(
    function() {
      // Увеличение текста при наведении
      $(this).css("transform", "scale(1.15)");
    },
    function() {
      // Возврат текста к нормальному размеру
      $(this).css("transform", "scale(1)");
    }
  );
});
```

Этот код выполняет следующие функции:

При наведении курсора на ссылку навигации (nav ul li a) изменяет ее размер с помощью CSS-свойства transform: scale(1.15), что увеличивает текст на 15%.

При уходе курсора с ссылки возвращает ее размер к нормальному значению с помощью transform: scale(1).

Эти сценарии, реализованные с использованием JavaScript и jQuery, добавляют интерактивности на веб-сайт, делая его более привлекательным и удобным для пользователей. Плавное появление элементов и анимации создают приятный визуальный опыт, а улучшенная навигация и интерактивные элементы галереи способствуют лучшему взаимодействию с сайтом.

3. Валидация кода и продвижение сайта.

3.1 Валидация кода сайта

Для проверки валидности HTML и CSS кода были использованы следующие онлайн инструменты:

- [W3C CSS Validator](#)
- [W3C HTML Validator](#)

Валидация CSS

При проверке CSS кода с использованием W3C CSS Validator ошибок и предупреждений не было обнаружено. Это свидетельствует о том, что весь написанный CSS код соответствует стандартам и является корректным. Сообщение о результате валидации представлено на рисунке 15.

W3C результаты проверки CSS для styles.css (CSS3 + SVG)

Поздравляем! Ошибок не обнаружено.

Рисунок 15 – Сообщение о результате валидации CSS

Валидация HTML

При проверке HTML кода с использованием W3C HTML Validator было получено одно предупреждение. Сообщение предупреждения представлено на рисунке 16.



Рисунок 16 – Сообщение о результате валидации CSS

Данное предупреждение указывает на отсутствие заголовка в элементе `<section>`. Валидатор рекомендует использовать элементы заголовков `<h2>`-`<h6>` для идентификации секций, либо использовать элемент `<div>` в тех случаях, когда заголовок не требуется.

Для устранения этого предупреждения было принято решение заменить `<section>` на `<div>`, так как заголовок не требовался. После исправления код был заново отправлен на проверку. В результате проверки было получено сообщение об отсутствии ошибок и предупреждений. Это можно увидеть на рисунке 17.

Document checking completed. No errors or warnings to show.

Used the HTML parser.

Total execution time 2 milliseconds.

Рисунок 17 – Повторная валидация кода

3.2 Способы продвижения сайтов

Основная цель оптимизации сайта – повышение его позиции в выдаче поисковых систем и доработка до максимального соответствия запросам пользователей.

SEO-продвижение (Search Engine Optimization)

SEO-продвижение включает в себя различные методы и стратегии, направленные на улучшение видимости сайта в поисковых системах. Основные приемы SEO-продвижения включают:

Ключевые слова: Оптимизация контента с использованием релевантных ключевых слов и фраз, которые потенциальные пользователи могут вводить в поисковые системы.

Качественный контент: Создание уникального, полезного и интересного контента, который привлекает пользователей и удерживает их на сайте.

Внутренняя оптимизация: Оптимизация структуры сайта, включая правильное использование заголовков (H1, H2, H3), метатегов (title, description), атрибутов изображений (alt) и внутренних ссылок.

Мобильная оптимизация: Обеспечение корректного отображения сайта на мобильных устройствах.

Скорость загрузки сайта: Ускорение загрузки страниц для улучшения пользовательского опыта и повышения рейтинга в поисковых системах.

Ссылочная масса: Наращивание качественных внешних ссылок (backlinks) на сайт, что повышает его авторитет и позицию в поисковых результатах.

Контекстная реклама (PPC - Pay-Per-Click)

Контекстная реклама — это форма интернет-рекламы, при которой объявления показываются в контексте результатов поиска или на страницах, связанных с тематикой рекламируемого продукта. Основные платформы для контекстной рекламы включают Google Ads и Яндекс.Директ. Контекстная

реклама позволяет привлекать целевых пользователей на сайт и оплачивается по модели "плата за клик".

SMM (Social Media Marketing)

Маркетинг в социальных сетях включает продвижение сайта через социальные платформы, такие как Facebook, Instagram, Twitter и другие. Основные приемы SMM включают:

Создание и ведение профилей: Регулярное обновление контента на страницах компании в социальных сетях.

Взаимодействие с аудиторией: Ответы на комментарии, участие в обсуждениях, проведение опросов и конкурсов.

Таргетированная реклама: Использование рекламных возможностей социальных платформ для показа объявлений целевой аудитории.

E-mail маркетинг

E-mail маркетинг включает рассылку электронных писем потенциальным и текущим клиентам с целью продвижения продуктов или услуг, информирования о новостях и предложениях. Основные приемы e-mail маркетинга включают:

Персонализация: Настройка писем под конкретного получателя, включая использование имени, учет интересов и предыдущих покупок.

Сегментация: Разделение базы подписчиков на сегменты для более точного таргетинга.

Аналитика: Анализ открываемости и кликабельности писем для оптимизации последующих рассылок.

Медийная реклама

Медийная реклама включает использование графических баннеров, видео и других мультимедийных элементов для продвижения сайта. Она может размещаться на различных платформах, таких как новостные сайты, блоги и видеохостинги. Основные виды медийной рекламы включают баннерную рекламу, видеорекламу и рекламу в мобильных приложениях.

Дополнительные способы продвижения

Influencer маркетинг: Сотрудничество с влиятельными лицами (блогерами, известными личностями) для продвижения продукта или услуги.

Контент-маркетинг: Создание и распространение полезного контента (статьи, видео, подкасты) для привлечения и удержания целевой аудитории.

Партнерские программы: Установление партнерских отношений с другими компаниями для обмена трафиком и увеличения видимости.

Вебинары и онлайн-курсы: Проведение обучающих мероприятий и курсов, которые привлекают пользователей и демонстрируют экспертность компании.

Форумы и сообщества: Участие в тематических форумах и онлайн-сообществах для увеличения узнаваемости бренда и привлечения трафика на сайт.

Каждый из этих методов может быть использован в зависимости от целей, бюджета и специфики бизнеса, обеспечивая комплексное продвижение сайта и увеличение его видимости в интернете.

3.3Использованные приемы продвижения сайта

Для продвижения сайта был использован метод SEO-продвижения.

Описание страницы (Meta Description):

```
<meta name="description" content="The art portfolio of Yana Stepanova. Portraits, stylizations, and sketches. Discover more about the artist and her work.">
```

Этот метатег предоставляет краткое описание содержимого страницы. Описание помогает поисковым системам и пользователям понять, о чем идет речь на странице, и помогает привлечь трафик.

Ключевые слова (Meta Keywords):

```
<meta name="keywords" content="Yana Stepanova, portfolio, portraits, stylization, sketches, art, artist">
```

Метатег ключевых слов содержит список слов или фраз, которые наиболее характерны для содержимого страницы. Хотя некоторые поисковые системы могут игнорировать этот метатег, его использование все еще может быть полезным.

Инструкции для поисковых роботов (Meta Robots):

```
<meta name="robots" content="index, follow">
```

Эти метатеги указывают поисковым роботам, следует ли индексировать содержимое страницы и следовать по ссылкам на ней. В данном случае, "index" означает, что страницу нужно индексировать, а "follow" означает, что поисковые роботы могут следовать по ссылкам на этой странице.

Эти метаданные помогают поисковым системам лучше понять и оценить содержимое вашего сайта, что в свою очередь может улучшить его позиции в результатах поиска и привлечь больше посетителей.

Заключение

В процессе выполнения работы были проделаны следующие шаги: изучение основных принципов создания веб-сайтов, анализ требований к курсовой работе, разработка структуры и дизайна сайта, а также создание HTML-кода с учетом современных стандартов и требований к SEO.

Структура сайта была определена на основе его назначения — представить портфолио художника. Были созданы основные разделы, такие как "Portrait", "Stylization", "Sketches" и "About me", что позволяет посетителям сайта легко найти необходимую информацию.

Дизайн сайта был разработан с учетом современных тенденций и эстетических принципов, что делает его привлекательным и удобным для использования.

Особое внимание было уделено оптимизации сайта для поисковых систем. Были добавлены метаданные, такие как описание страницы, ключевые слова и инструкции для поисковых роботов, что помогает улучшить видимость сайта в результатах поиска.

Для улучшения функциональности и взаимодействия с пользователем на сайте были использованы JavaScript и библиотека jQuery.

JavaScript был применен для создания интерактивных элементов, таких как анимации, эффекты переходов и реакции на действия пользователя. Использование JavaScript позволило значительно улучшить пользовательский опыт, сделав сайт более динамичным и привлекательным.

Библиотека jQuery была использована для упрощения и ускорения процесса разработки. С помощью jQuery были реализованы эффекты плавного появления и исчезания текста, а также другие анимации, что позволило создать более профессиональный и современный вид сайта.

В целом, веб-сайт является функциональным и привлекательным, однако требует дополнительной работы над составлением коллекции работ для достижения более высокого уровня целостности и профессионализма.

Проделанная работа позволяет сделать вывод о том, что поставленные задачи были успешно выполнены, и сайт готов к предполагаемой публикации в Интернете.

Ознакомиться с курсовой работой в подробностях можно по следующей ссылке, содержащей архив проекта:

https://drive.google.com/drive/folders/1FXc0m3V_mY-wewhtYRtlJ1tgSuPbdbL?usp=sharing

ПРИЛОЖЕНИЕ 1

Скриншоты страниц

Главная страница (Портреты)

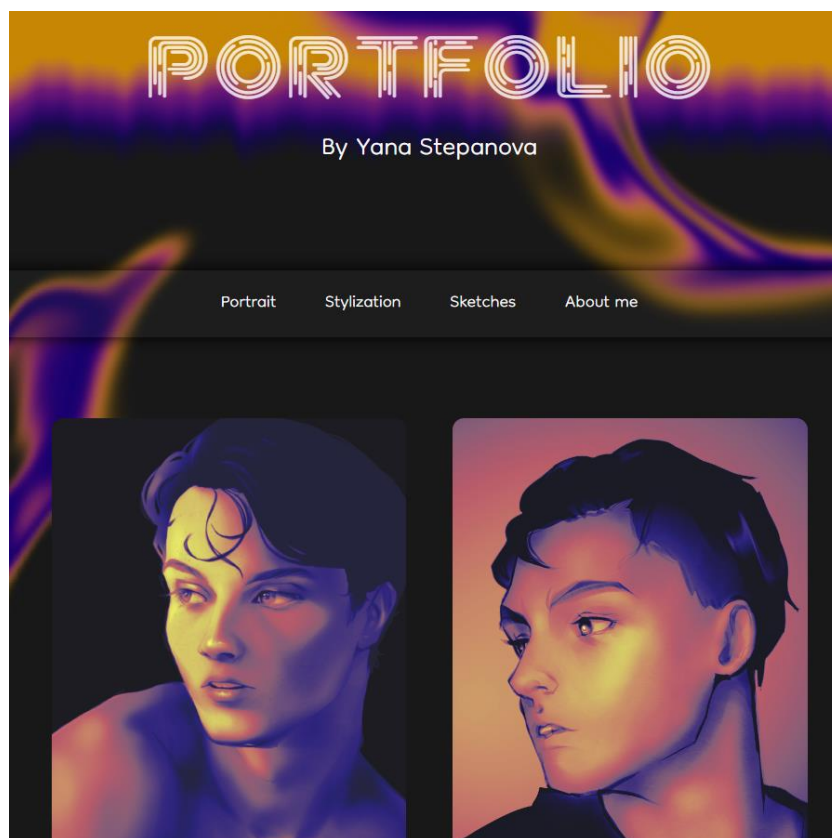


Рисунок 18 – Страница “Портреты”

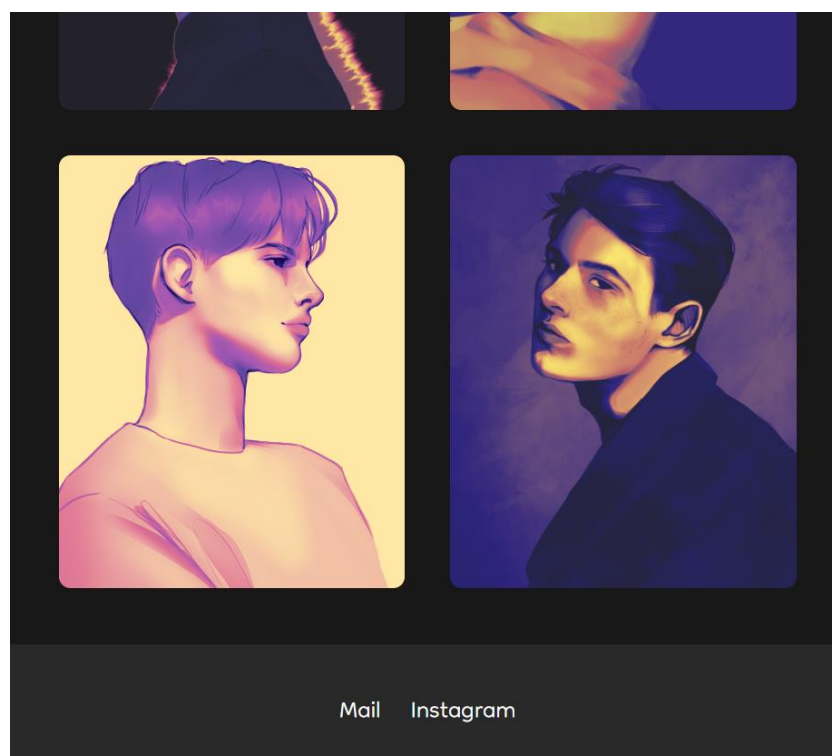


Рисунок 19 – Страница “Портреты”

Страница “Стилизация”

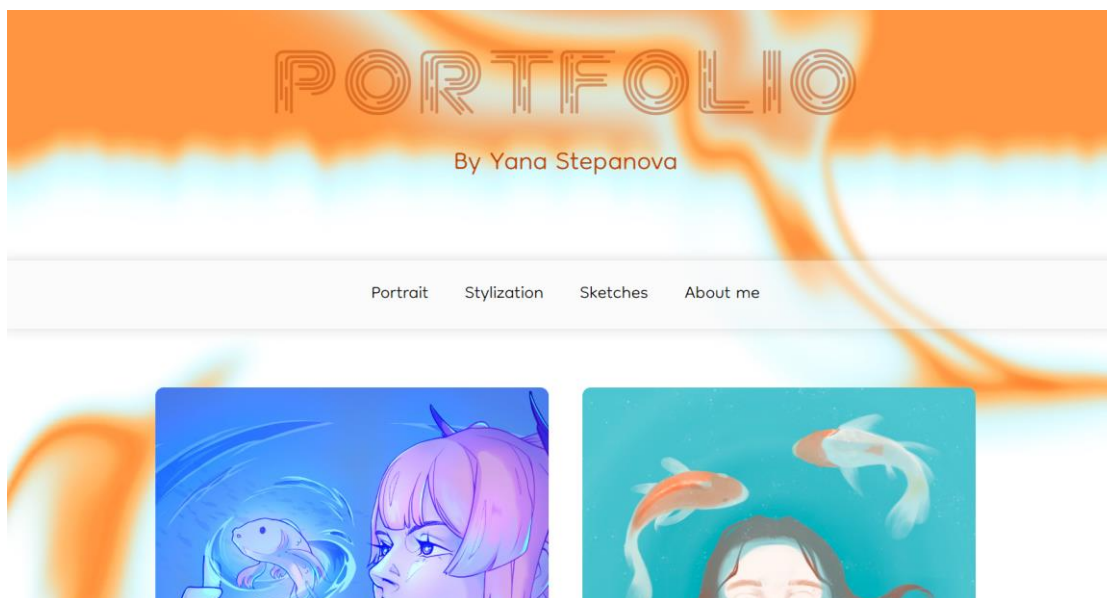
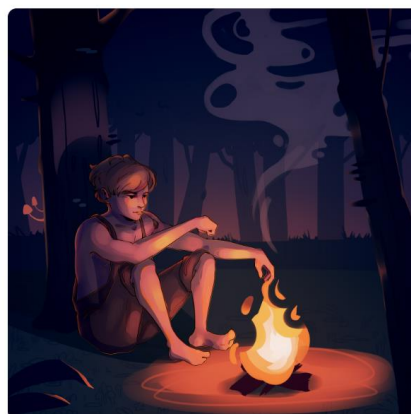
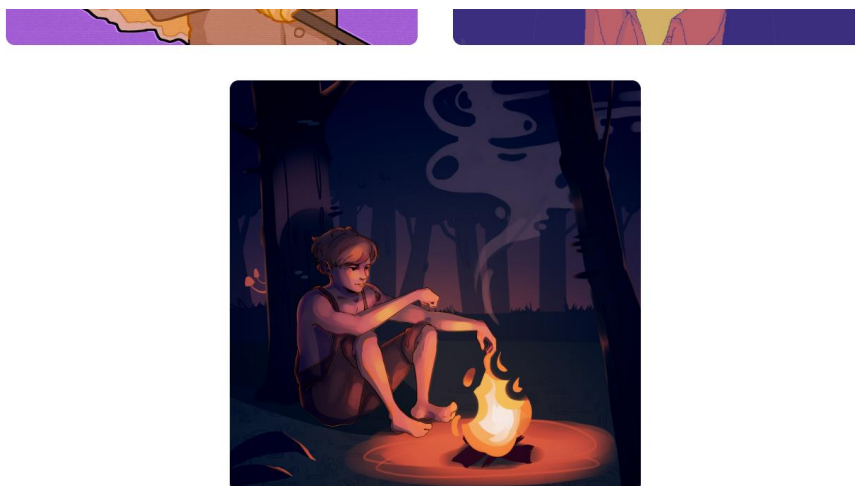


Рисунок 20 - Страница “Стилизация”



[Mail](#) [Instagram](#)

Рисунок 21 - Страница “Стилизация”

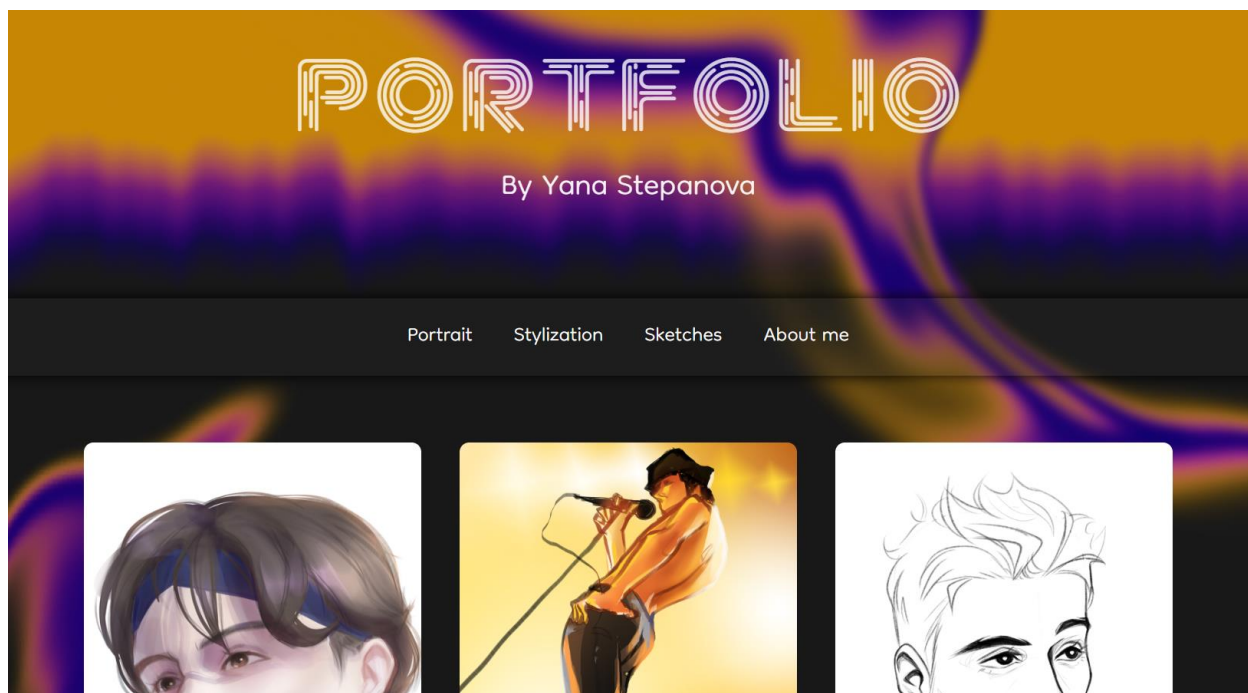


Рисунок 22 - Страница “Наброски”

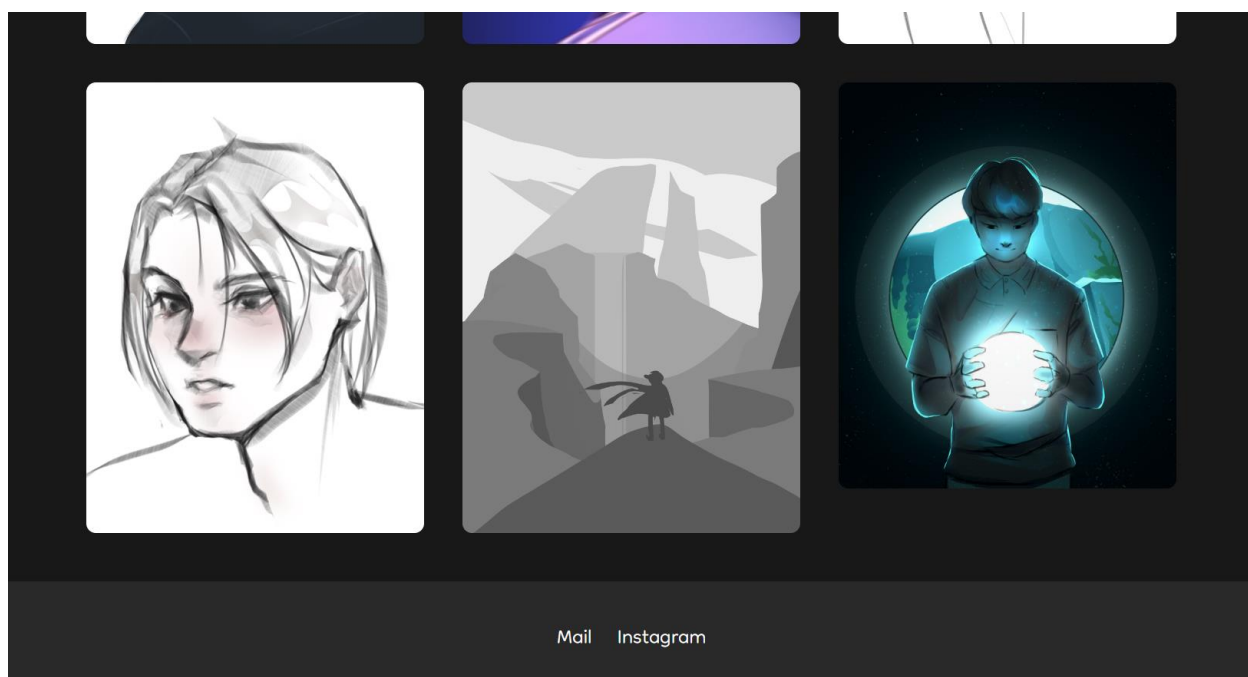


Рисунок 23 - Страница “Наброски”

Страница “О себе”



Рисунок 24 – Страница “О себе”

Рисунок 25 – Страница “О себе”

ПРИЛОЖЕНИЕ 2

Листинг html-кода страницы с портретами

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Art-portfolio - Yana Stepanova</title>
  <link rel="stylesheet" href="styles.css">
  <meta name="description" content="The art portfolio of Yana Stepanova. Portraits,
stylizations, and sketches. Discover more about the artist and her work.">
  <meta name="keywords" content="Yana Stepanova, portfolio, portraits, stylization,
sketches, art, artist">
  <meta name="robots" content="index, follow">
</head>
<body>
  <div class="background-container">
    
    
  </div>

  <header>
    <h1 class="h1fade">PORTFOLIO</h1>
    <p class="pfade">By Yana Stepanova</p>
  </header>

  <nav>
    <ul>
      <li><a href="index.html">Portrait</a></li>
      <li><a href="stylization.html">Stylization</a></li>
      <li><a href="sketches.html">Sketches</a></li>
      <li><a href="about.html">About me</a></li>
    </ul>
  </nav>

  <main>
    <div class="gallery">
      <div class="gallery-item" data-title="Gavin Grabowski portrait"></div>
      <div class="gallery-item" data-title="Tim Meiresone portrait"></div>
      <div class="gallery-item" data-title="Gafarilo portrait"></div>
      <div class="gallery-item" data-title="Yaochi portrait"></div>
      <div class="gallery-item" data-title="Dominik Sadoch portrait"></div>
      <div class="gallery-item" data-title="Alice Vink portrait"></div>
```

```
<div      class="gallery-item"      data-title="Unknown      guy"></div>
<div      class="gallery-item"      data-title="Andreas      Praeg      portrait"></div>
</div>
</main>

<footer>
<ul>
<li><a href='mailto:ja_na00@mail.ru'>Mail</a></li>
<li><a href="https://another-site.com">Instagram</a></li>
</ul>
</footer>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="gallery.js"></script>
<script src="fade.js"></script>
<script src="linkHover.js"></script>
</body>
</html>
```

Листинг html-кода страницы со стилизацией

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Art-portfolio - Yana Stepanova - Stylization</title>
  <link rel="stylesheet" href="stylesStylization.css">
</head>
<body>
  <div class="background-container">
    
    
  </div>

  <header>
    <h1 class="h1fade">PORTFOLIO</h1>
    <p class="pfade">By Yana Stepanova</p>
  </header>

  <nav>
    <ul>
      <li><a href="index.html">Portrait</a></li>
      <li><a href="stylization.html">Stylization</a></li>
      <li><a href="sketches.html">Sketches</a></li>
      <li><a href="about.html">About me</a></li>
    </ul>
  </nav>

  <main>
    <div class="gallery">
      <div class="gallery-item" data-title="Kokomi"></div>
      <div class="gallery-item" data-title="Koi"></div>
      <div class="gallery-item" data-title="Froggy"></div>
      <div class="gallery-item" data-title="Tiger"></div>
      <div class="gallery-item" data-title="Dog"></div>
      <div class="gallery-item" data-title="Mysterious forest"></div>
      <div class="gallery-item" data-title="Heizou"></div>
      <div class="gallery-item" data-title="Witch cat"></div>
      <div class="gallery-item" data-title="Liard"></div>
      <div class="gallery-item" data-title="Pixel art"></div>
```

```
        <div class="gallery-item" data-title="A quiet night"></div>
    </div>
</main>

<footer>
    <ul>
        <li><a href='mailto:ja_na00@mail.ru'>Mail</a></li>
        <li><a href="https://another-site.com">Instagram</a></li>
    </ul>
</footer>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="gallery.js"></script>
<script src="fade.js"></script>
<script src="linkHover.js"></script>
</body>
</html>
```


Листинг html-кода страницы с набросками

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Art-portfolio - Yana Stepanova - Sketches</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <div class="background-container">
    
    
  </div>

  <header>
    <h1 class="h1fade">PORTFOLIO</h1>
    <p class="pfade">By Yana Stepanova</p>
  </header>

  <nav>
    <ul>
      <li><a href="index.html">Portrait</a></li>
      <li><a href="stylization.html">Stylization</a></li>
      <li><a href="sketches.html">Sketches</a></li>
      <li><a href="about.html">About me</a></li>
    </ul>
  </nav>

  <main>
    <div class="gallery">
      <div class="gallery-item"></div>
      <div class="gallery-item"></div>
      <div class="gallery-item"></div>
      <div class="gallery-item"></div>
      <div class="gallery-item"></div>
      <div class="gallery-item"></div>
      <div class="gallery-item"></div>
      <div class="gallery-item"></div>
      <div class="gallery-item"></div>
    </div>
  </main>

  <footer>
    <ul>
      <li><a href='mailto:ja_na00@mail.ru'>Mail</a></li>
      <li><a href="https://another-site.com">Instagram</a></li>
    </ul>
  </footer>

  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="fade.js"></script>
```

```
<script src="linkHover.js"></script>  
</body>  
</html>
```

Листинг html-кода страницы о себе

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Art-portfolio - Yana Stepanova - About</title>
  <link rel="stylesheet" href="stylesStylization.css">
</head>
<body>
  <div class="background-container">
    
    
  </div>

  <header>
    <h1 class="h1fade">PORTFOLIO</h1>
    <p class="pfade">By Yana Stepanova</p>
  </header>

  <nav>
    <ul>
      <li><a href="index.html">Portrait</a></li>
      <li><a href="stylization.html">Stylization</a></li>
      <li><a href="sketches.html">Sketches</a></li>
      <li><a href="about.html">About me</a></li>
    </ul>
  </nav>

  <main>
    <div class="gallery">
      <div class="gallery-item"></div>
    </div>
    <p>Hey there! I'm Yana. I'm a student who's passionate about programming, but
drawing has always been my thing.
    I've been drawing since I was a kid, and it's become a big part of my life.
    I love trying out different drawing styles and techniques. While I mostly focus on
portraits, I'm also curious about exploring other types of art.
    </p>
    <p>
      I do all my drawing digitally, mainly using Photoshop. It's cool to use graphic
tablets and computer software to bring my ideas to life.
      Each piece I create is a result of my exploration and experimentation, and I hope it
resonates with you in some way.</p>

    <p>If you want to contact me, just drop me a line:
      aaa@mail.ru</p>
    <p>
      Or you can try this number: +7 (964) 000 00 00
    </p>
```

</main>

<footer>

Mail

Instagram

</footer>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

<script src="fade.js"></script>

<script src="linkHover.js"></script>

</body>

</html>

Листинг css-кода страниц с портретами и набросками

```
body {
  display : none;
  margin : 0;
  font-family : Dongle;
  color : #ffffff;
  background-color : #181818;
}

.background-container {
  position : absolute;
  top : 0;
  left : 0;
  width : 100%;
  height : auto;
  z-index : -1;
  pointer-events : none;
}

.background-img {
  position : absolute;
  width : 100%;
  height : auto;
  opacity : 1;
}

header {
  text-align : center;
  padding : 50px 0;
}

header h1 {
  display : none;
  font-size : 7em;
  font-weight : normal;
  margin : 0;
  font-family : "Moscow Metro";
}

header p {
  display : none;
  font-size : 3em;
  margin : 10px 0;
  font-weight : normal;
  font-family : Dongle;
}

nav {
```

```
text-align : center;
margin : 20px 0;
padding : 1px;
box-shadow : 0 0 15px rgb(0, 0, 0, 1);
background-color : #28282850;
}
```

```
nav ul li a {
display : inline-block;
transition : transform 0.3s;
}
```

```
nav ul {
list-style : none;
padding : 0;
}
```

```
nav li {
display : inline;
margin : 0 20px;
}
```

```
nav a {
text-decoration : none;
color : #ffffff;
font-size : 2em;
}
```

```
main {
padding : 50px;
font-size : 2em;
```

```
}
```

```
.gallery {
display : flex;
flex-wrap : wrap;
gap : 40px;
justify-content : center;
```

```
}
```

```
.gallery-item {
max-width : calc(30% - 20px);
position : relative;
}
```

```
.gallery-item img {
width : 100%;
height : auto;
border-radius : 10px;
transition : transform 0.7s ease;
```

```
}

footer {
text-align : center;
padding : 5px 0;
background-color : hsl(0 0% 16%);

font-size : 2em;
}

footer ul {
list-style : none;
padding : 0;
}

footer li {
display : inline;
margin : 0 10px;
}

footer a {
text-decoration : none;
color : #fff;
}

@media (max-width: 800px) {
header {
font-size : 70%;
}

.gallery-item {
max-width : calc(50% - 20px);
}

nav {
text-align : center;
margin : 20px 0;
font-size : 70%;
}
}

@media (max-width: 600px) {
header {
font-size : 60%;
}
nav li {
display : block;
margin : 10px 0;
}
.gallery-item {
flex : 1 100%;
max-width : 100%;
}
```

```
}  
}  
  
@font-face {  
  font-family : "Moscow Metro";  
  src : url("fonts/MoscowMetroColor.otf");  
}  
  
@font-face {  
  font-family : "Pais Alt";  
  src : url("fonts/PaisAlt-Medium.otf");  
}  
  
@font-face {  
  font-family : "Dongle";  
  src : url("fonts/Dongle-Light.ttf");  
}  
  
.gallery-item:hover {  
  z-index : 1;  
}  
  
.gallery-item:hover img {  
  transform : scale(1.2);  
}  
  
.image-title {  
  position : absolute;  
  bottom : 0;  
  left : 0;  
  background-color : rgb(0, 0, 0, 0.1);  
  color : white;  
  padding : 5px;  
  border-radius : 10px;  
  display : none;  
}
```


Листинг CSS-кода страниц со стилизацией и о себе

```
body {
  display : none;
  margin : 0;
  font-family : Dongle;
  color : #bc4002;
  background-color : #ffffff;
}
.background-container {
  position : absolute;
  top : 0;
  left : 0;
  width : 100%;
  height : auto;
  z-index : -1;
  pointer-events : none;
}
.background-img {
  position : absolute;
  width : 100%;
  height : auto;
  opacity : 1;
}

header {
  text-align : center;
  padding : 50px 0;
}
header h1 {
  display : none;
  font-size : 7em;
  font-weight : normal;
  margin : 0;
  font-family : "Moscow Metro";
}
header p {
  display : none;
  font-size : 3em;
  margin : 10px 0;
  font-weight : normal;
  font-family : Dongle;
}
nav {
  text-align : center;
  margin : 20px 0;
  padding : 1px;
  box-shadow : 0 0 15px rgb(184, 184, 184, 0.5);
  background-color : #f6f6f672;
}
nav ul li a {
  display : inline-block;
  transition : transform 0.3s;
```

```
}
nav ul {
list-style : none;
padding : 0;
}
nav li {
display : inline;
margin : 0 20px;
}
nav a {
text-decoration : none;
color : #000000;
font-size : 2em;
}
main {
padding : 50px;
font-size : 2em;
}

.gallery {
display : flex;
flex-wrap : wrap;
gap : 40px;
justify-content : center;
}
.gallery-item {
max-width : calc(40% - 30px);
position : relative;
}
.gallery-item img {
width : 100%;
height : auto;
border-radius : 10px;
transition : transform 0.7s ease;
}
footer {
text-align : center;
padding : 5px 0;
background-color : #282828;
font-size : 2em;
}
footer ul {
list-style : none;
padding : 0;
}
footer li {
display : inline;
margin : 0 10px;
}
footer a {
text-decoration : none;
color : #fff;
```

```
}
@media (max-width: 800px) {
header {
font-size : 60%;
}
nav li {
display : block;
margin : 10px 0;
font-size : 70%;
}
.gallery-item {
flex : 1 100%;
max-width : 80%;
}
}
@font-face {
font-family : "Moscow Metro";
src : url("fonts/MoscowMetroColor.otf");
}
@font-face {
font-family : "Pais Alt";
src : url("fonts/PaisAlt-Medium.otf");
}
@font-face {
font-family : "Dongle";
src : url("fonts/Dongle-Light.ttf");
}
.gallery-item:hover {
z-index : 1;
}
.gallery-item:hover img {
transform : scale(1.1);
}
.image-title {
position : absolute;
bottom : 0;
left : 0;
background-color : rgb(0, 0, 0, 0.3);
color : white;
padding : 5px;
border-radius : 10px;
display : none;
}
```

Листинг javascript кода для пульсации заголовка и плавного появления страниц

```
$(document).ready(function() {  
    $('.h1 fade').fadeIn(4000);  
    $('.pfade').fadeIn(5000);  
    $('body').fadeIn(2000);  
  
    function pulsate() {  
        $('h1').animate({ opacity: 0.4 }, 600)  
            .animate({ opacity: 1.0 }, 800, pulsate);  
    }  
    pulsate();  
});
```

Листинг javascript кода для появления названия рисунков

```
$(document).ready(function(){  
    $(".gallery-item").mouseenter(function(){  
        var title = $(this).data("title");  
        $(this).append("<div class='image-title'>" + title + "</div>");  
        $(this).find(".image-title").fadeIn(); // Используем fadeIn для плавного появления  
    }).mouseleave(function(){  
        $(this).find(".image-title").fadeOut(); // Используем fadeOut для плавного  
исчезновения  
    });  
});
```

Листинг javascript кода для выдвижения ссылок

```
$(document).ready(function(){  
    $("nav ul li a").hover(  
        function() {  
            // Увеличение текста при наведении  
            $(this).css("transform", "scale(1.15)");  
        },  
        function() {  
            // Возврат текста к нормальному размеру  
            $(this).css("transform", "scale(1)");  
        }  
    );  
});
```