Содержание

[Введение 4](#_Toc134060750)

[1. Постановка задачи 5](#_Toc134060751)

[1.1. Обзор аналогичных решений 5](#_Toc134060752)

[1.2. Техническое задание 7](#_Toc134060753)

[1.3. Выбор средств реализации программного продукта 7](#_Toc134060754)

[1.4. Вывод 8](#_Toc134060755)

[2. Проектирование страниц веб-сайта 9](#_Toc134060756)

[2.1. Выбор способа верстки 9](#_Toc134060757)

[2.2. Выбор стилевого оформления 9](#_Toc134060758)

[2.3. Выбор шрифтового оформления 9](#_Toc134060759)

[2.4. Разработка логотипа 10](#_Toc134060760)

[2.5. Разработка пользовательских элементов 10](#_Toc134060761)

[2.6. Разработка спецэффектов 13](#_Toc134060762)

[2.7. Выводы 15](#_Toc134060763)

[3. Реализация структуры веб-сайта 16](#_Toc134060764)

[3.1. Структура HTML-документа 16](#_Toc134060765)

[3.2. Добавление таблиц стилей Sass и CSS 16](#_Toc134060766)

[3.3. Использование стандартов XML (SVG) 16](#_Toc134060767)

[3.4 Управление элементами DOM 17](#_Toc134060768)

[3.5. Выводы 17](#_Toc134060769)

[4. Тестирование веб-сайта 19](#_Toc134060770)

[4.1. Адаптивный дизайн веб-сайта 19](#_Toc134060771)

[4.2. Кроссбраузерность веб-сайта 20](#_Toc134060772)

[4.3. Руководство пользователя 22](#_Toc134060773)

[4.4. Выводы 24](#_Toc134060774)

[Заключение 25](#_Toc134060775)

[Список использованных литературных источников 25](#_Toc134060776)

[ПРИЛОЖЕНИЕ А Прототипы веб-страниц 27](#_Toc134060777)

[ПРИЛОЖЕНИЕ Б Макеты структуры веб-сайта 32](#_Toc134060778)

[ПРИЛОЖЕНИЕ В Листинг HTML-документа 37](#_Toc134060779)

[ПРИЛОЖЕНИЕ Г Листинг SCSS и CSS 51](#_Toc134060780)

[ПРИЛОЖЕНИЕ Д Листинг XML-файлов 82](#_Toc134060781)

[ПРИЛОЖЕНИЕ Е Листинг SVG 86](#_Toc134060782)

[ПРИЛОЖЕНИЕ Ж Листинг JavaScript 94](#_Toc134060783)

Введение

Электронные испарители стали популярны в последние десятилетия как альтернатива традиционным сигаретам. С ростом осведомленности об опасностях табачного курения, люди начали искать новые способы уменьшения рисков, связанных с потреблением табака. Смотря на их растущую популярность компании начинают рассматривать сервисы по продаже испарителей как стратегию для продвижения бренда.

Одним из главных преимуществ электронных испарителей является их уменьшение вреда здоровью пользователя и природе. В отличие от традиционных сигарет, они не выделяют вредных веществ, таких как тар и углекислый газ, в окружающую среду. Это значит, что люди могут использовать парогенераторы в помещениях без вреда для окружающих, и это может привести к значительному снижению пассивного курения, а это значит, что вредное влияние брендов традиционных сигарет будет с темпом снижаться благодаря компаниям, которые производят и продают свои испарители.

Цель курсового проекта: разработать веб-сайт “Вейпшоп” с использованием HTML5, а также с применением SCSS, CSS3 и JavaScript.

Задачи 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;
* разработать руководство пользователя.

Целевой аудиторией являются пользователи возрастной категории после 18 лет.

1. Постановка задачи

1.1. Обзор аналогичных решений

Перед тем как приступать к разработке своего сайта, можно изучить готовые решения и оценить варианты других разработчиков. Так мы сможем понять, что другие разработчики думают насчет реализации полезного и удобного сайта.

На рисунке 1.1 представлен внешний вид главной страницы белорусского сайта «https://www.sigaretnet.by/».

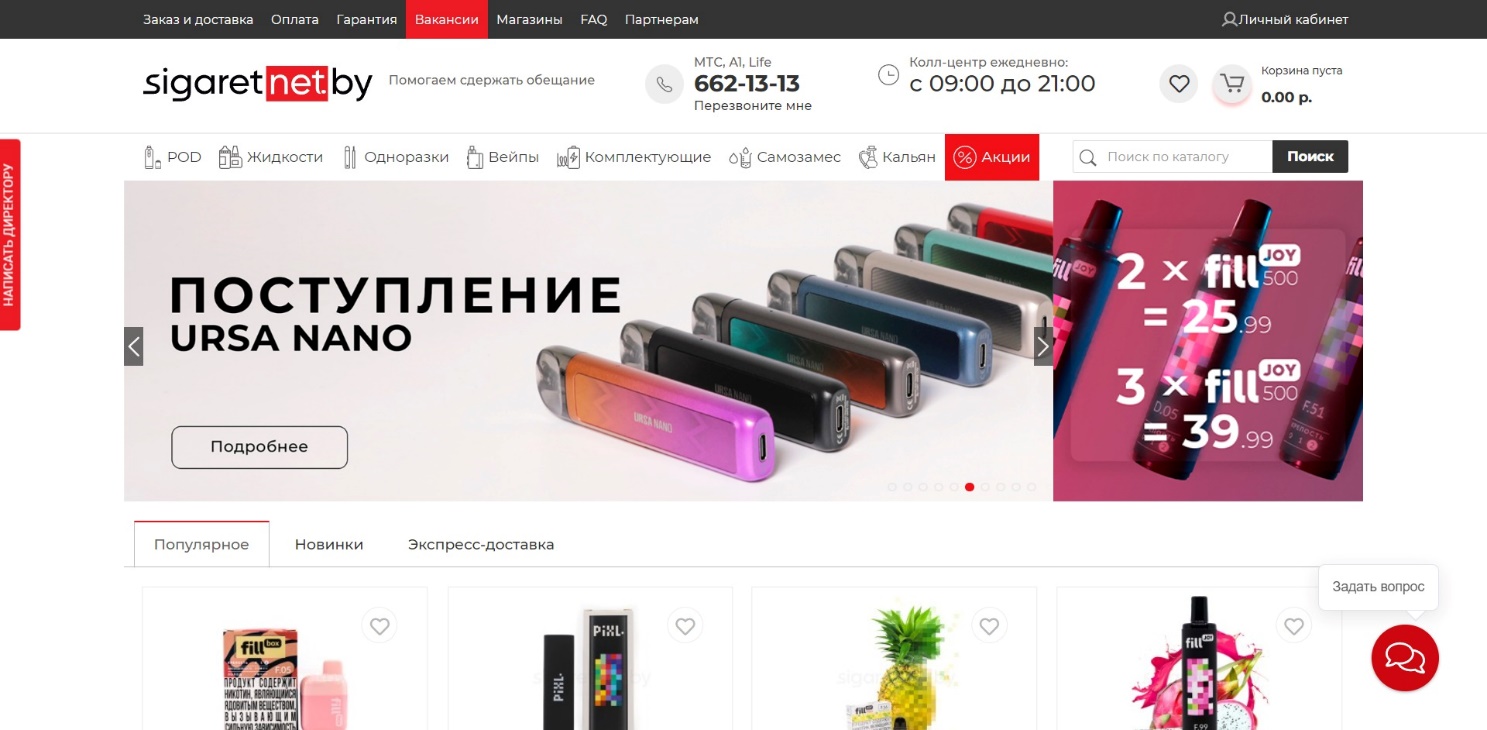


Рисунок 1.1 – Окно главной страницы сайта «https://www.sigaretnet.by»

Сайт выполнен в минималистичном дизайне, что позволяет пользователю легко перемещаться по сайту и в два клика открыть нужную ему страницу. В цветовой гамме преобладают белый, красный и черный цвета, которые подчеркивают выраженный оригинальный дизайн, однако мне такая цветовая гамма не нравится. Также на главной странице сразу в удобном формате указан контактный телефон, навигация по имеющимся видам товаров и слайдер с рекламой товаров.

Данный сайт специализируется на продаже электронных сигарет и их комплектующих. Из него можно вынести элементы, привлекающие внимание пользователя яркими названиями и акциями.

На рисунке 1.2 показано, что сайт отлично адаптирован под мобильные устройства.

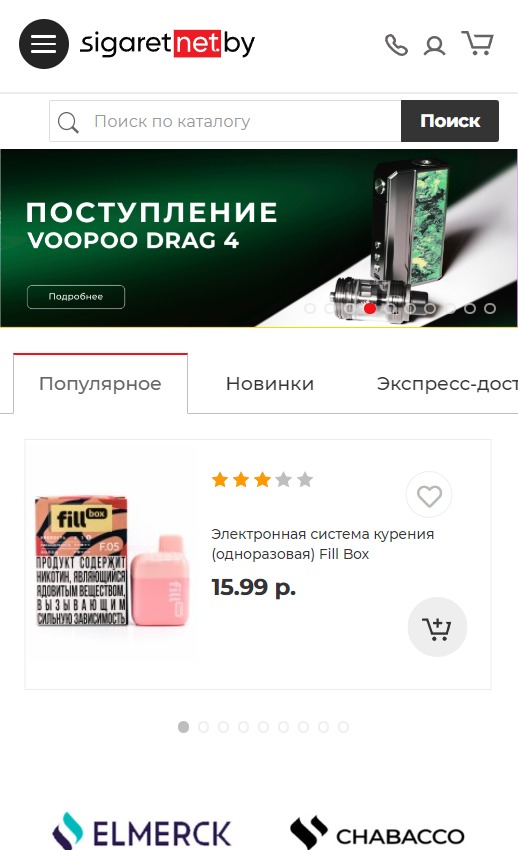


Рисунок 1.2 –Мобильная версия сайта «https://www.sigaretnet.by»

После изучения других аналогов можно сделать вывод о том, что рассмотренный сайт предоставляет необходимую информацию и содержит все основные функции.

Из минусов сайта хочу подчеркнуть цветовую гамму и излишнее количество блоков навигации.

Рассмотрим другой пример сайта.

На рисунке 1.3 представлен внешний вид главной страницы магазина VOOPOO Vape.

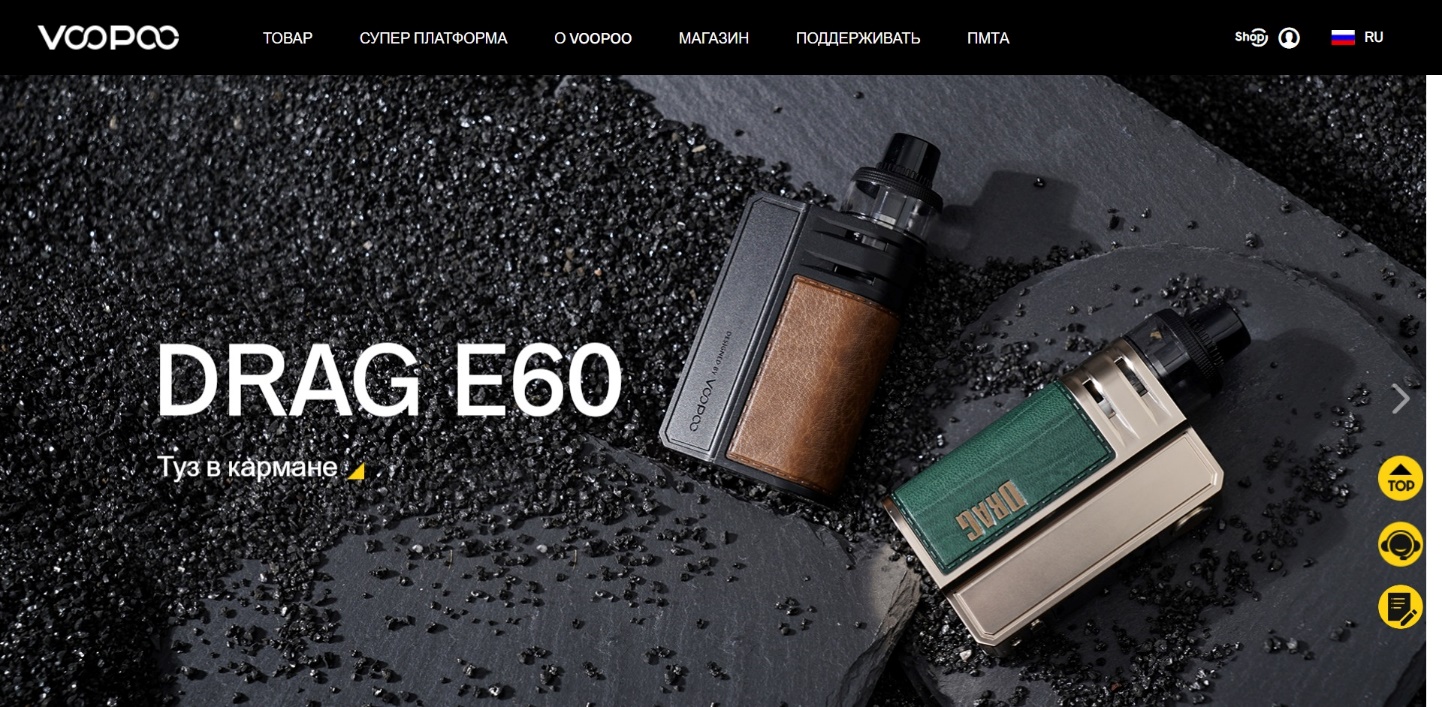


Рисунок 1.3 – Окно главной страницы сайта «https://www.voopoo.com/»

В дизайне данного сайта смешаны различные, но сочетающиеся стили сайта.

В дизайне сайта преобладают минималистичный дизайн и черно-белые цвета. Сайт старается рассказать о своей компании, что еще больше привлекает пользователя (рисунок 1.4).

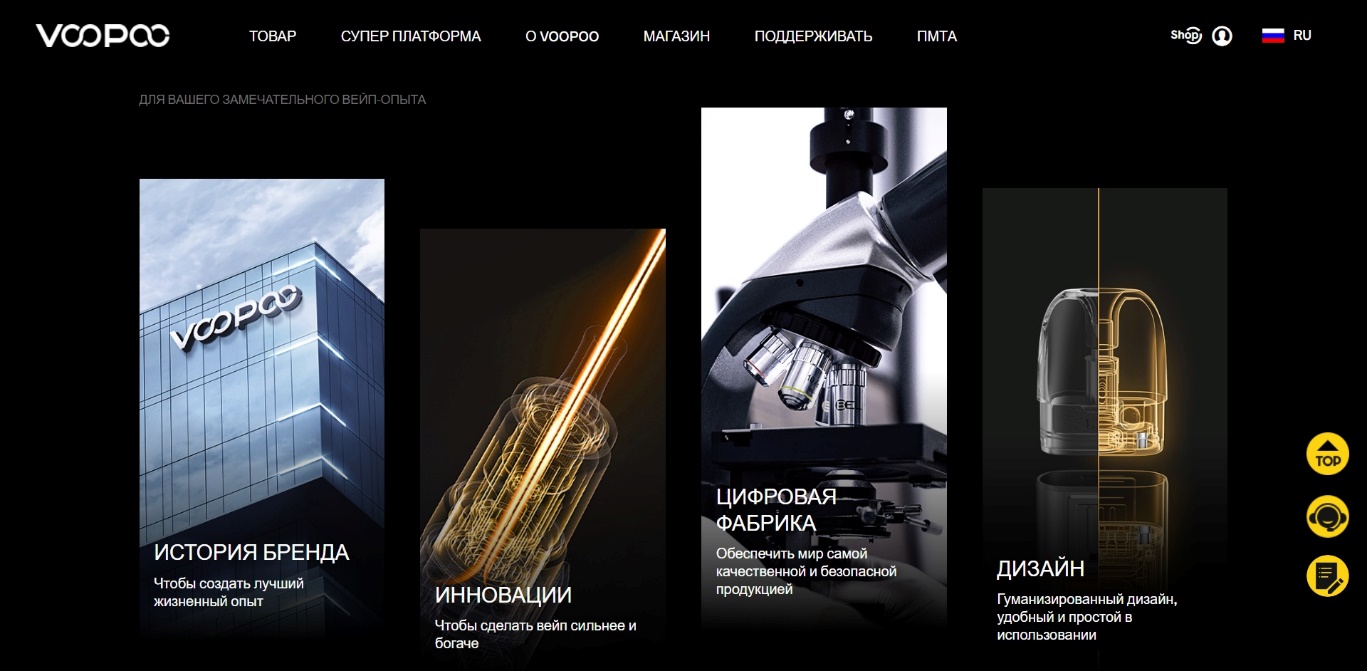


Рисунок 1.4 –Вкладка “О себе” сайта «https://www.voopoo.com/»

1.2. Техническое задание

Поставлена задача реализовать полноценный многостраничный веб-сайт на тему «Вейпшоп» (8 страниц). На сайте должна быть представлена информации об электронных сигаретах, такая как поиск, сортировка, выбор товара, а также история компании.

Для привлечения аудитории главная задача сделать интуитивно понятный сайт с информацией об испарителях и их .

Основными задачами сайта являются:

* предоставление навигационного меню;
* предоставление блоков-ссылок;
* возможность выбора испарителя;

На главной странице меню будет располагаться сверху. В меню будут предоставляться ссылки на основную информацию (продукты, поддержка, о нас).

Веб-страница должна быть кроссбраузерной и адаптивной для мобильной версии (ширина до 420px), планшетного устройства (ширина до 1000px). Начиная с ширины в 1000px должна открываться версия для персонального компьютера.

1.3. Выбор средств реализации программного продукта

Курсовой проект выполняется в редакторе кода JetBrains WebStorm.

При создании веб-страниц были использованы языки: HTML, CSS, JS, XML. Если HTML необходим для структурирования содержания страницы, то CSS необходим для того, чтобы форматировать это структурированное содержание.

JavaScript поддерживает объектно-ориентированный, императивный и функциональный стили. Во все основные браузеры встроен интерпретатор JavaScript, именно поэтому они могут выполнять скрипты на странице.

1.4. Вывод

В данном разделе были рассмотрены плюсы и минусы аналогичных решений на выбранную тему сайта для предотвращения ошибок в реализации веб-сайта. Были определены основные технические задания по созданию страницы. Также выбраны средства реализации программного продукта, а именно языки HTML, CSS, JS, XML.

2. Проектирование страниц веб-сайта

2.1. Выбор способа верстки

Требования проекта включают в себя кроссбраузерность и адаптивность. По этой причине была выбрана flex-box вёрстка.

Flexbox - это новая технология, которая значительно упрощает создание сложных макетов веб-страниц и предоставляет множество новых возможностей для веб-дизайнеров. Ее поддержка во всех современных браузерах делает ее необходимым инструментом для создания качественных и эффективных веб-сайтов.

2.2. Выбор стилевого оформления

Выбор цветовой гаммы— важный этап разработки проекта Цвета должны не только сочетаться между собой, но и передавать тему проекта. Они применяются не только к текстовым элементам, но и к блочным элементам в качестве фона. Далее представлены цвета, используемые на страницах.

Как основной цвет веб-страницы был выбран чёрный, для фона блоков использовались его оттенки.

Как первый акцентный цвет, использованный для оформления текста и некоторых иконок светло-голубой.

2.3. Выбор шрифтового оформления

Основным шрифтом на сайте является San Francisco, который создает гармоничный интерфейс сайта.



Рисунок 2.1 – Шрифт San Francisco

Размерность была использована от 16 до 24 пикселей, цвета текста #A7B7D0, #D9D9D9 и #000000 в зависимости от назначения текста.

2.4. Разработка логотипа

Логотип — это фирменный знак, который люди ассоциируют с конкретным брендом. Логотип — это не абстрактный набор символов или просто красивое изображение.

Логотип выполнен под концепцию сайта. Чтобы подчеркнуть уникальный стиль сайта, был создан логотип, который имеет отсылку к названию. Логотип, представленный на рисунке 2.2 был создан с помощью Figma.



Рисунок 2.2 – Логотип сайта «seinsdieum.github.io/EllipseVape»

2.5. Разработка пользовательских элементов

В проекте будут представлены такие элементы пользовательского интерфейса, как кликабельные баннеры, блоки ссылками, интерактивные слайдеры и навигационные элементы.



Рисунок 2.3 – Шапка для страниц сайта

Навигация – ключевой критерий удобства сайта, своеобразная «карта местности». Она представляет собой набор специальных приемов, методов и элементов, дающих возможность посетителям перемещаться между различными страницами ресурса. Навигационное меню представлено на рисунке 2.3.

В большинстве случаев данная система предопределяет успешность продвижения.



Рисунок 2.4 – Блоки-ссылки на серии электронных испарителей

Данный пользовательский элемент привлечет внимание пользователя и позволит ему быстро перемещаться по страницам сайта.

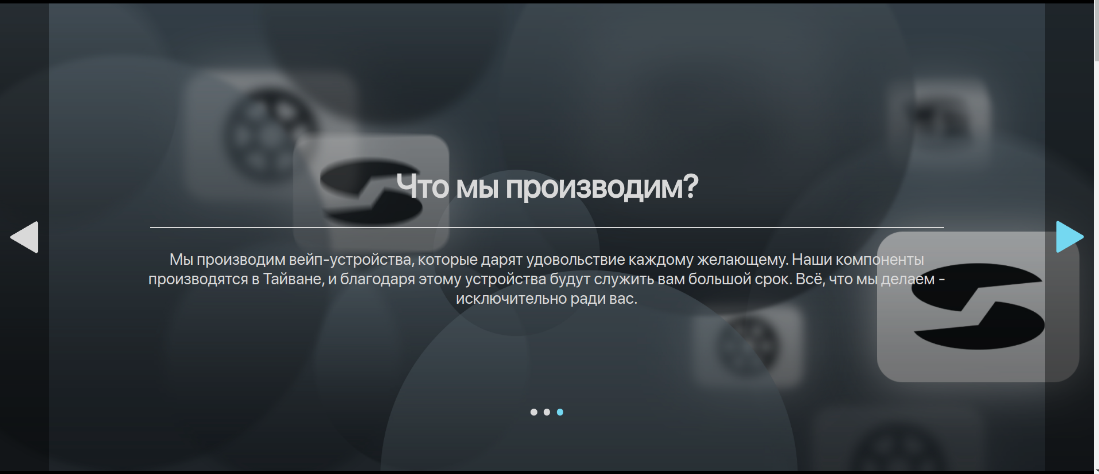


Рисунок 2.6 – Слайдер с баннерами

Слайдер с баннерами – наиболее простой вариант привлечения внимания пользователя. Его привлекает ­динамичная информация, что и предоставляют различные сайты. Он представлен на рисунке 2.6.

Для просмотра желаемого товара пользователь может сначала взглянуть на “витринное” представление товара в каталоге, где он может просмотреть фотографии испарителя, его особенности и характеристики. Со страницы представления товара он может перейти сразу на страницу покупки. Страница представления устройства представлена на рисунке 2.7.



Рисунок 2.7 – Страница представления устройства

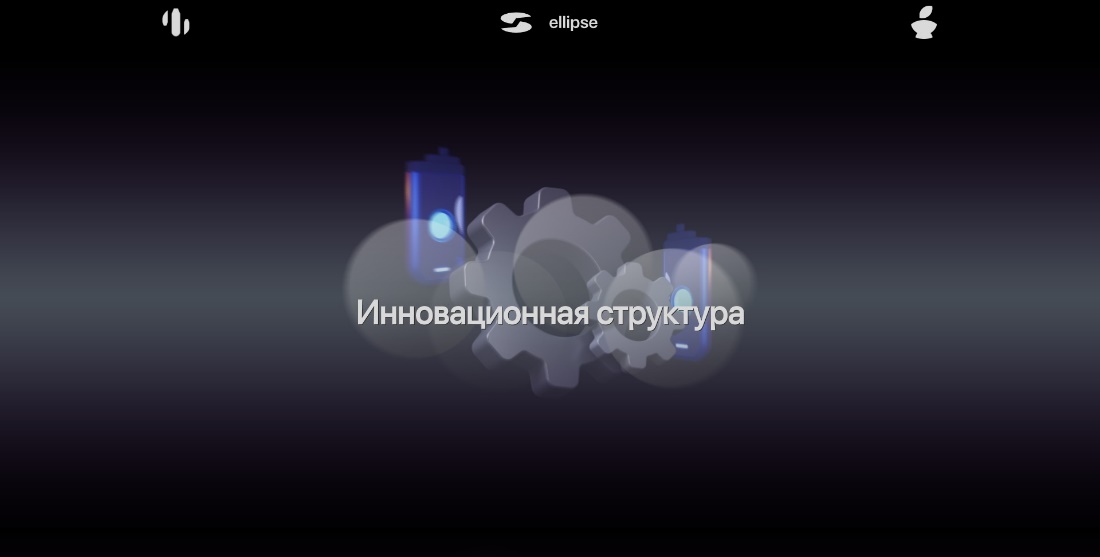


Рисунок 2.8 – Особенность товара с иллюстрацией

Для возможности заказа электронной сигареты на странице товара представлены изображения товара, просмотр его характеристик, заполнение формы для заказа товара. Форма представлена на рисунке 2.9.

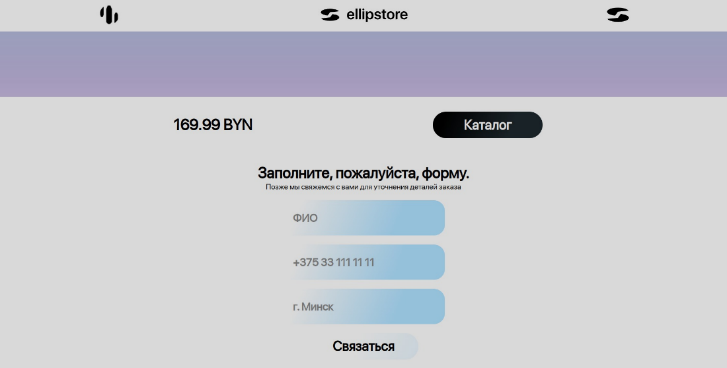


Рисунок 2.9 – Элементы навигации и форма

На странице каталога и магазина присутствуют блоки товаров и поле для их поиска. Они представлены на рисунке 2.10.



Рисунок 2.10 – Блоки товаров с поиском

Таким образом пользовательские элементы помогают увеличить удобство использования сайта клиентом и предоставить все виды информации.

2.6. Разработка спецэффектов

На сайте присутствуют динамические эффекты и анимации, которые придадут сайту интерактивность и сделают его более привлекательным для пользователя.

Для показа сообщений были созданы уведомления с плавным выпадением из верха страницы. Они могут быть использованы с целью предоставить интуитивно понятную информацию пользователю. Уведомление представлено на рисунке 2.11.

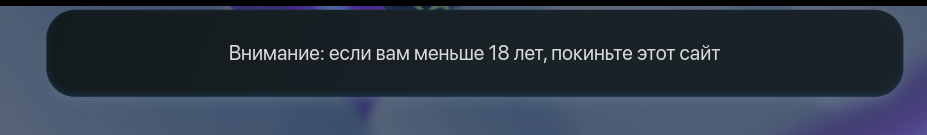


Рисунок 2.11 – Уведомление с плавной анимацией появления

Для показа сообщений были созданы уведомления с плавным выпадением из верха страницы. Они могут быть использованы с целью предоставить интуитивно понятную информацию пользователю.

Для того, чтобы получение информации не было затрудненным, элементы управления слайдером плавно появляются при наведении на сам слайдер. Пример представлен на рисунке 2.12.

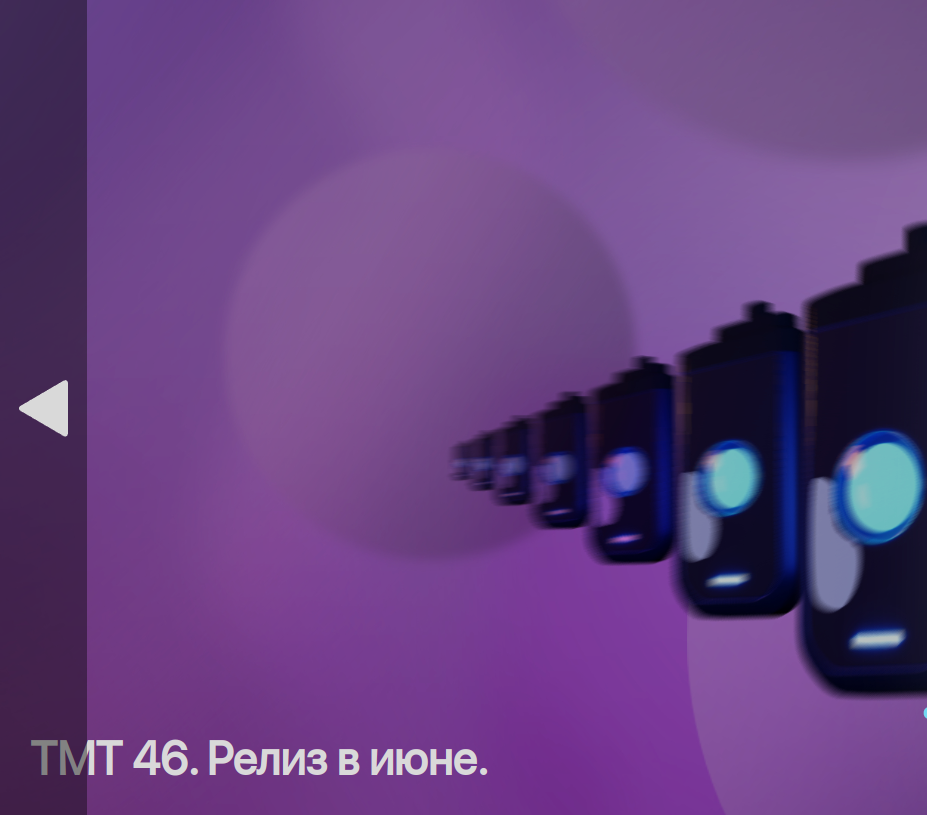


Рисунок 2.12 – Анимированные элементы управления слайдером

Скролл-анимация - это анимация элементов при прокрутке веб-страницы. Чтобы подчеркнуть динамичность информации сайта, были добавлены плавные скролл-анимации. Пример представлен на рисунке 2.13.



Рисунок 2.13 – Скролл-анимация элемента

На странице каталога присутствует градиентная анимация фона элемента при наведении на него. Это было сделано с целью подчеркнуть индивидуальность сайта. Она представлена на рисунке 2.14.

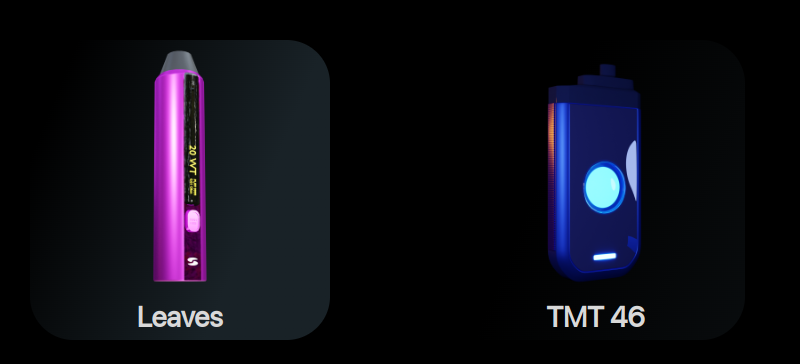


Рисунок 2.14 – Градиентная анимация при наведении

2.7. Выводы

В данном разделе был создан прототип веб-сайта, определены задачи по созданию содержания на веб-сайте. Выбрано единое цветовое и шрифтовое оформления сайта для удобства просмотра пользователем. Разработан дизайн и расположение на сайте пользовательских элементов, спецэффектов и анимации. Логотип выполнен под цветовую гамму и концепцию сайта.

Прототипы страниц сайта представлен в приложении А. Макеты страниц сайта представлен в приложении Б.

3. Реализация структуры веб-сайта

## **3.1. Структура HTML-документа**

В начале HTML-документа, внутри тега <head></head>, располагается служебная информация, такая как название веб-сайта. Тег <head> используется для хранения других элементов, которые помогают браузеру обрабатывать данные. За исключением тега <title>, который устанавливает заголовок окна веб-страницы, содержимое тега <head> не отображается на веб-странице. Для страницы с подробной информацией об устройстве, название страницы и её содержимое было задано с помощью JavaScript.

Тег <body></body>, расположенный после тега <head>, содержит все теги, в которых расположен веб-сайт. Структура тела документа состоит из семантических тегов, таких как header, main, section и footer. Эти теги предназначены для того, чтобы компьютерные программы, такие как поисковые системы, сборщики информации и речевые браузеры, могли понимать, какой тип информации содержится в данных тегах. Листинг с семантическими тегами представлен в приложении 3.

Код HTML, CSS, XML и Sass был протестирован и не требовал дополнительного тестирования, так как IDE WebStorm автоматически отображает ошибки и предупреждения. Тег <header> был использован для создания шапки сайта на всех страницах. Тег <main> был создан для размещения основной информации и разметки на страницах.

Листинг HTML представлен в приложении В.

## **3.2. Добавление таблиц стилей SCSS и CSS**

В этом проекте были использованы различные способы подключения таблиц стилей: внешнее, внутреннее и строковое. Внешнее подключение через тег link было применено для подключения основного стилевого файла, который используется на каждой странице. Файл со стилями хранится отдельно и может быть использован на любой странице веб-сайта. Внутреннее подключение было использовано для подключения глобального стилевого файла, в котором находились переменные для цветов, и стили для базовых элементов. В таблицах стилей были использованы селекторы по классу и идентификатору.

Листинг SCSS и CSS представлен в приложении Г.

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

SVG (Scalable Vector Graphics) - это формат векторной графики, который используется для создания графических изображений в веб-браузерах. SVG файлы содержат описание графических объектов, таких как линии, кривые, фигуры и текст, в виде векторных элементов, что позволяет сохранять качество изображения при масштабировании. Кроме того, SVG файлы занимают меньше места на сервере и загружаются быстрее, чем растровые изображения, такие как JPG или PNG, что улучшает производительность веб-сайтов.

Для создания логотипа сайта и иконок было выбрано использование SVG-формата, который обеспечивает сохранение качества изображения при любом размере экрана. Это позволяет создавать картинки с высоким разрешением, которые выглядят четко и детализировано на любых устройствах.

Данные хранения каталога вейпов удобно хранить отдельно в XML-документах. В проекте в XML-файлах хранятся название устройства, картинка представления, картинка устройства, описание, характеристики и т.д.

Для валидации XML-файлов была применена XML Schema.

Листинг SCSS и CSS представлен в приложении Д.

## **3.4 Управление элементами DOM**

DOM (Document Object Model) представляет собой иерархическое дерево, которое отображает структуру HTML документа и позволяет управлять его элементами. С помощью JavaScript можно манипулировать элементами DOM, изменять их свойства, добавлять новые элементы, удалять существующие и многое другое. В этом тексте мы рассмотрим основные методы работы с элементами DOM, которые позволяют взаимодействовать с HTML документом и динамически изменять его содержимое.

В этом проекте манипуляция DOM посредством JavaScript использовалась для управления классами элементов при прокрутке страницы.

Для отображения подробной информации о вейпе на странице мы передаем url параметр, который содержит ключевое имя устройства, о котором нужно отобразить информацию. После получения ключевого имени ищется информация об устройства с данным именем и выводится на страницу.

Также управление DOM было использовано для отображения уведомлений.

Листинг JavaScript представлен в приложении Ж.

## **3.5. Выводы**

На данном этапе была реализована структура на HTML, а также созданы таблицы стилей SCSS/CSS. Было описано использование стандартов XML (SVG), а также языка программирования JavaScript. Было определено стилевое оформление веб-сайта, созданы анимации, а также веб-сайт был подготовлен к следующему этапу – к тестированию.

4. Тестирование веб-сайта

## **4.1. Адаптивный дизайн веб-сайта**

Для обеспечения адаптивности веб-страницы была применена современная технология верстки flex. Кроме того, для адаптивной подстройки внешнего вида страницы под различные устройства использовались медиа-запросы. С помощью flex-верстки было задано расположение основных блоков, затем для задания размеров использовалась переменная size, которая равна calc(1vw+1vh), что позволило легко добиться нужного отображения на разных разрешениях и соотношениях сторон.

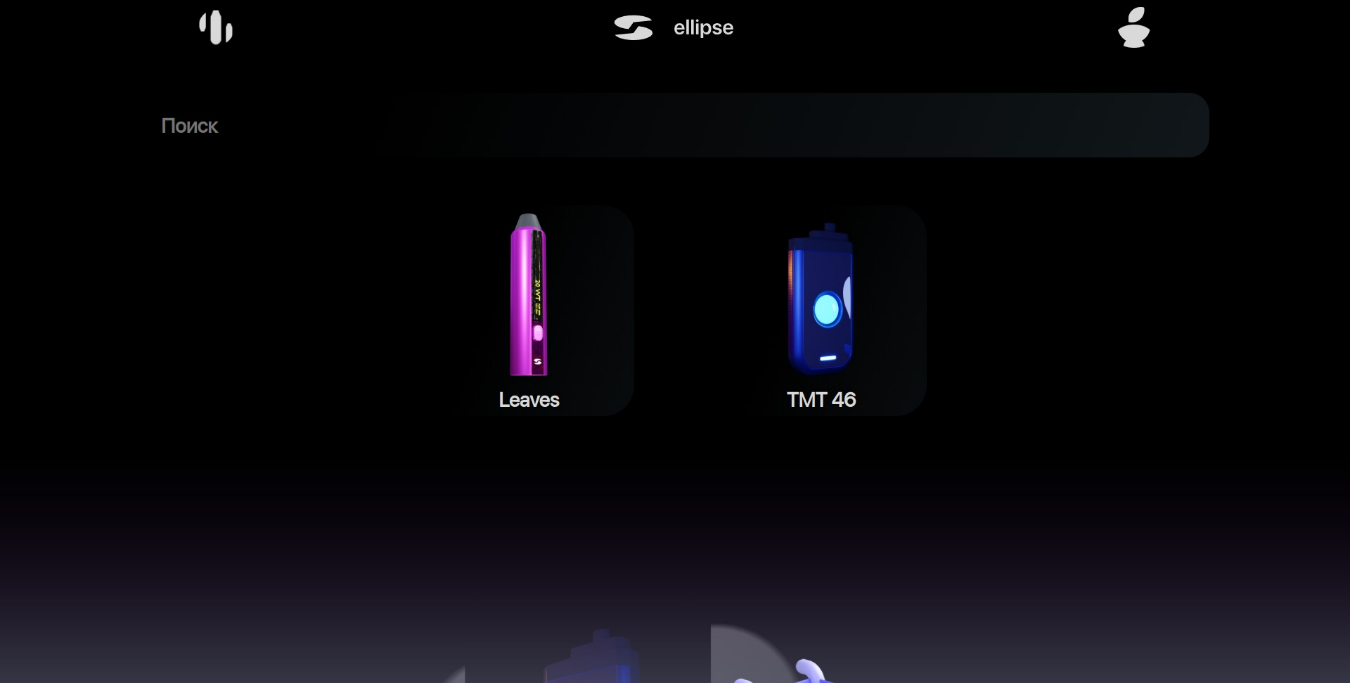


Рисунок 4.1 – Вид страницы для компьютера

Для верстки шапки сайта была использована технология flex, которая позволила легко и гибко расположить элементы в нужном порядке. Кроме того, для обеспечения корректного отображения страницы на мобильных устройствах были применены медиа-запросы, которые позволяют изменять расположение элементов в основной части страницы при маленьких размерах экрана. Для изменяющегося расположения элементов в каталоге также была применена технология flex, которая позволяет удобно смещать элементы.

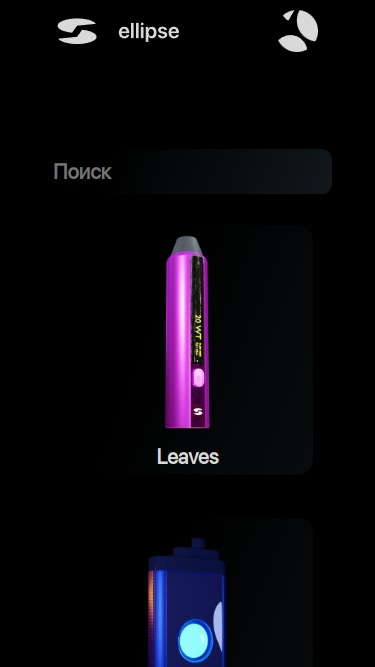


Рисунок 4.2 – Вид страницы для мобильных устройств

Адаптивность является одним из важнейших требований к современным сайтам. Веб-ресурс должен корректно отображаться на экранах компьютеров, планшетов и смартфонов. Для этого используется адаптивный дизайн и верстка, что позволяет гарантировать равномерное качество отображения на всех устройствах и обеспечивает максимальное охват аудитории.

## **4.2. Кроссбраузерность веб-сайта**

Кроссбраузерность - это способность веб-сайта одинаково хорошо работать в различных браузерах и на разных операционных системах. Она является важным фактором для создания успешного и функционального сайта. Кроссбраузерность является необходимым условием для обеспечения максимального охвата аудитории.



Рисунок 4.3 – Вид страницы в браузере Chrome

После написания основной структуры страницы на HTML и внешнего стилевого оформления появился вопрос о тестировании. Веб-сайт был открыт при помощи различных браузеров.

В ходе тестирования выяснилось, что сайт одинаково отображается и работает во всех современный браузерах таких как: Google Chrome, Microsoft Edge, Mozilla Firefox.

Лишь на этапе тестирования сайта в браузере Safari была обнаружена проблема с правильным отображением шрифта. Для решения этой проблемы были подключены интерфейсные шрифты Apple устройств, которые также входят в семейство шрифтов San Francisco и похожи на использованный на сайте шрифт.

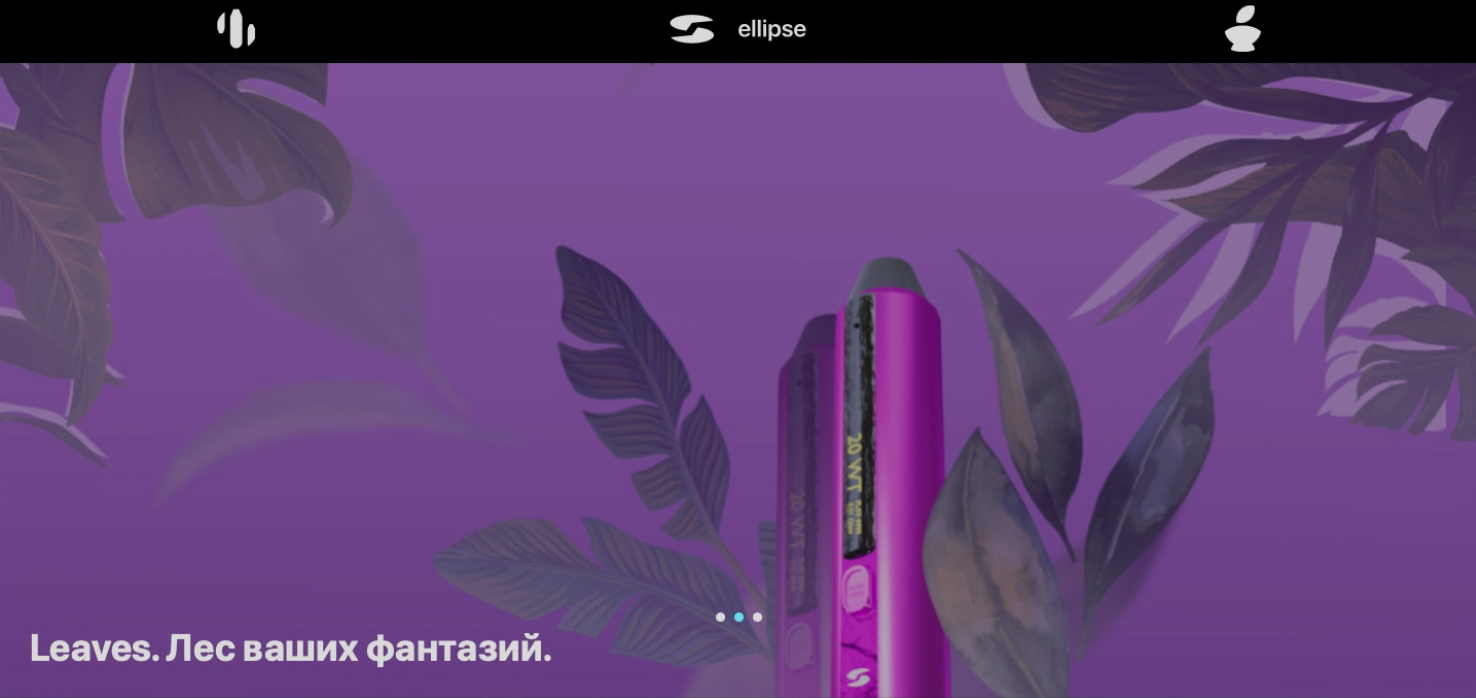


Рисунок 4.4 – Вид страницы в браузере Safari



Рисунок 4.5 – Вид страницы в браузере Firefox

## **4.3. Руководство пользователя**

Интерфейс простой и сразу понятный. Основное взаимодействие производится в каталоге, где пользователь найти нужный товар в поиске.

При вводе устройства, которого нет в списке, сайт выдает уведомление о том, что товаров с введенным именем не было найдено.

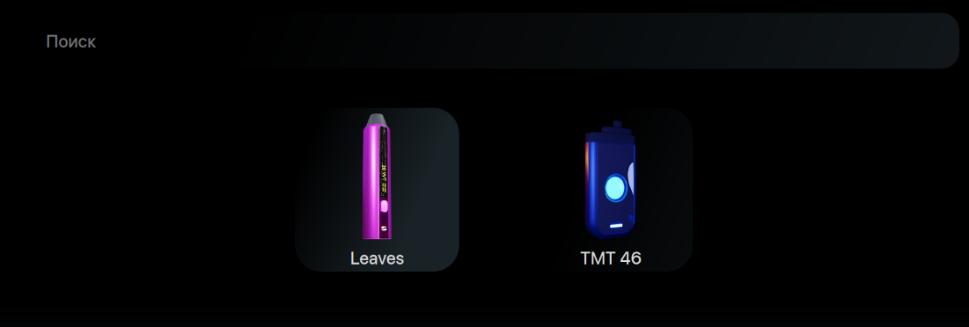


Рисунок 4.6 – Поиск

На странице устройства можно прочитать его описание, слоган, особенности и характеристики при прокрутке, пример на рисунке 4.7.

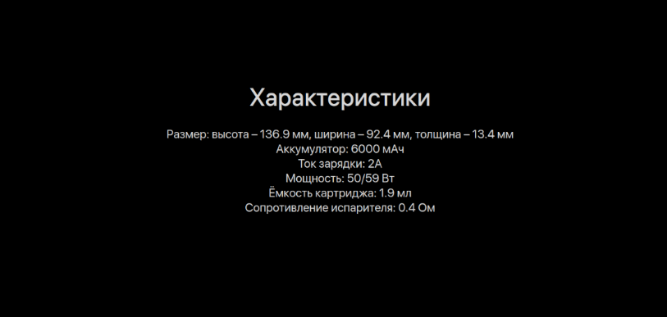


Рисунок 4.7 – Страница с подробной информацией

Для перехода на страницу с покупкой товара пользователю нужно пролистать вниз и нажать на баннер с предложением покупки товара.



Рисунок 4.8 – Ссылка-баннер на страницу покупки товара

При переходе на страницу покупки пользователь может выбрать покупку, и при нажатии на кнопку “Купить” появится форма для связи с клиентом.

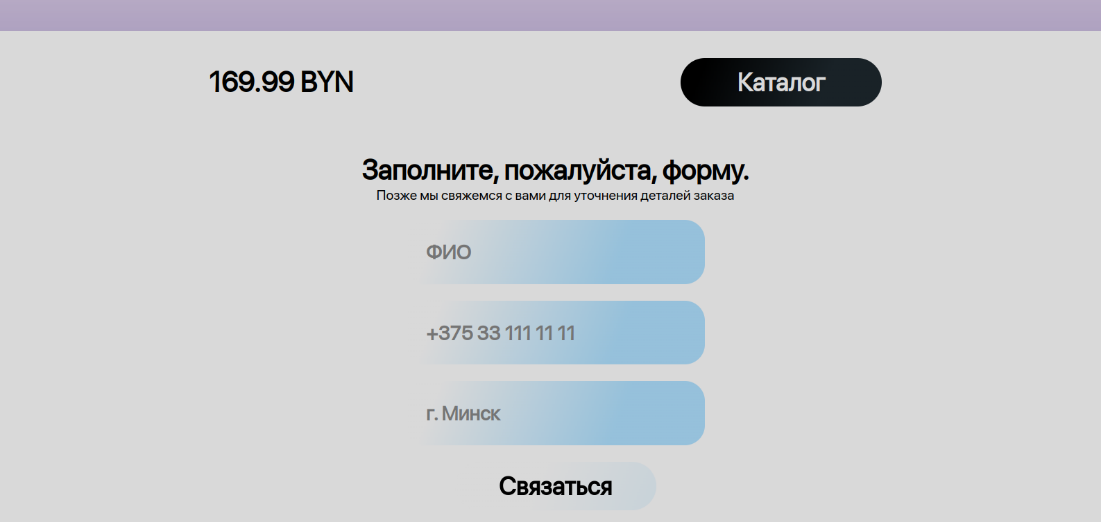


Рисунок 4.9 – Форма для связи с клиентом

При возникновении у клиентов вопросов насчет товара, в подвале сайта есть ссылка на страницу с поддержкой и наиболее частыми вопросами, где кратко расписаны ответы на них.



Рисунок 4.10 – Страница поддержки для клиентов

Если у пользователя возникают какие-либо сторонние вопросы или ему нужна поддержка, то он может оставить заявку на поддержку с помощью формы на этой странице.

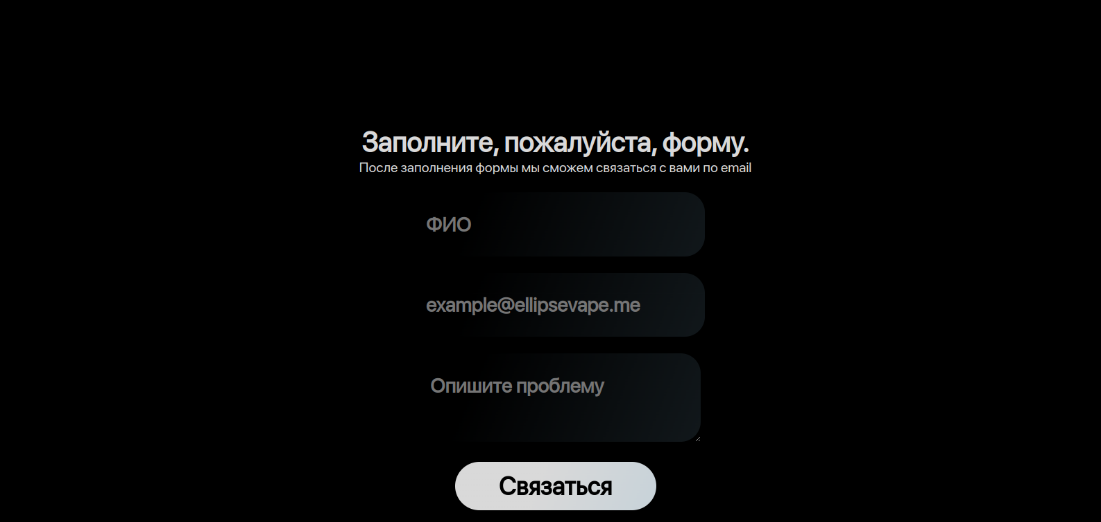


Рисунок 4.11 – Форма для обратной связи

## **4.4. Выводы**

В данном разделе было рассмотрено, как сайт отображается на различных устройствах и в разных браузерах. Посредством тестирования были выявлены некоторые недостатки, которые необходимо исправить для дальнейшего улучшения продукта. Кроме того, было создано руководство пользователя, в котором кратко описывается, как пользоваться сайтом.

Заключение

В ходе выполнения проекта был создан полноценный веб-сайт для аренды автомобилей. Была применена адаптивная верстка для корректного отображения сайта на всех типах устройств.

При разработке сайта были изучены и проанализировали различные решения, исследовали их плюсы и минусы, а также узнали о последних тенденциях в дизайне веб-сайтов. Продумано содержание основных страниц и выбрали наиболее подходящие инструменты для разработки.

Были разработаны прототипы и макеты дизайна страниц сайта в программе Figma. Были также созданы пользовательские элементы, спецэффекты и логотип для идентификации нашего сайта среди множества других ресурсов в интернете.

Для качественного отображения изображений на всех устройствах мы использовали SVG-изображения и создали XML-документ для последующей вставки на веб-страницу HTML. Веб-сайт успешно прошел тестирование на валидность, кроссбраузерность и адаптивность. Было также подготовлено руководство пользователя для удобства пользования нашим сайтом.

В процессе разработки были использованы технологии, не изучаемые на данном курсе, такие как JavaScript.

Таким образом, была достигнута поставленная цель по созданию веб-сайта для онлайн-бронирования автомобилей, учитывая все требования и задачи проекта.

Сайт курсового проекта загружен на репозиторий в GitHub и доступен по ссылке: https://github.com/seinsdieum/EllipseVape

Список использованных источников

1. Покрытие CSS атрибутов браузерами [Электронный ресурс]. — Режим доступа: https://caniuse.com/ – Дата доступа 22.04.2023.
2. Документация по JavaScript/HTML [Электронный ресурс]. — Режим доступа: <https://developer.mozilla.org> – Дата доступа 22.04.2023.

ПРИЛОЖЕНИЕ А

**Прототипы веб-страниц**

|  |  |
| --- | --- |
| Рисунок - Прототип главной страницы | Рисунок - Прототип страницы каталога |
| Рисунок - Прототип страницы представления | Рисунок - Прототип страницы магазина |
| Рисунок - Прототип страницы товара | Рисунок - Прототип страницы истории |
| Рисунок - Прототип страницы общения | Рисунок - Прототип страницы поддержки |

|  |  |
| --- | --- |
| Рисунок - Прототип главной страницы | Рисунок - Прототип страницы каталога |
| Рисунок - Прототип страницы представления | Рисунок - Прототип страницы магазина |
| Рисунок - Прототип страницы товара | Рисунок - Прототип страницы истории |
| Рисунок - Прототип страницы общения | Рисунок - Прототип страницы поддержки |

ПРИЛОЖЕНИЕ Б

**Макет структуры веб-сайта**

|  |  |
| --- | --- |
| Рисунок - Макет главной страницы | Рисунок - Макет страницы каталога |
| Рисунок - Макет страницы представления | Рисунок - Макет страницы магазина |
| Рисунок - Макет страницы товара | Рисунок - Макет страницы истории |
| Рисунок - Макет страницы общения | Рисунок - Макет страницы поддержки |

|  |  |
| --- | --- |
| Рисунок - Макет главной страницы | Рисунок - Макет страницы каталога |
| Рисунок - Макет страницы представления | Рисунок - Макет страницы магазина |
| Рисунок - Макет страницы товара | Рисунок - Макет страницы истории |
| Рисунок - Макет страницы общения | Рисунок - Макет страницы поддержки |

ПРИЛОЖЕНИЕ В

**Листинг HTML-документа**



Рисунок - Структура страницы



Рисунок - Структура footer



Рисунок - Структура header

Листинг тега main главной страницы:

|  |
| --- |
| <main>   <section class="slider-view non-interactive " ></section>  <a href="catalog.html" class="scroll-hidden">  <div class="banner-image" style="background-image: url(img/ellipse/catalog1\_21\_9.jpg)**;**">  <div class="banner-description-center"><h1>Каталог вейпов</h1></div>  </div>  </a>   <a href="store.html" class="scroll-hidden">  <div class="banner-image non-fade" style="background-image: url(img/ellipse/storeSection1.jpg)**;**">  <div class="banner-description-center"><h1>Магазин Ellipstore</h1></div>  </div>  </a>  <a href="club.html" class="scroll-hidden">  <div class="banner-image non-fade" style="background-image: url(img/ellipse/clubSection1.jpg)**;**">  <div class="banner-description-center"><h1>Ellipse Club</h1></div>  </div>  </a>  <a href="history.html" class="scroll-hidden">  <div class="banner-image non-fade" style="background-image: url(img/ellipse/historySection1.jpg)**;**">  <div class="banner-description-center"><h1>История бренда</h1></div>  </div>  </a> </main> |

Листинг тега main страницы каталога:

|  |
| --- |
| <main>     <a>  <div class="banner-image scroll-hidden back-transp dark">  <div class="banner-full-description"><h1 style="color: rgb(217**,** 217**,** 217)**;**">Каталог!</h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  Устройства представлены ниже.  </p>  </div>  </div>  </a>  <div class="dark row scroll-hidden">  <input id="Search" type="text" class="search-field" placeholder="Поиск">  </div>   <div class="items-grid scroll-hidden">  <a href="catalogitem.html?item=leaves">  <div class="grid-item">  <img src="img/Vapes/Leaves/Transparent/LeavesVapePresent21\_9.png">  <h1>Leaves</h1>  </div>  </a>  <a href="catalogitem.html?item=tmtfs">  <div class="grid-item">  <img src="img/Vapes/TMT/46/Transparent/fstransparent.png">  <h1>TMT 46</h1>  </div>  </a>   </div>   <a href="store.html" class="scroll-hidden">  <div class="banner-image back-transp" style="background-image: url('img/ellipse/storeSection1.png')">  <div class="banner-full-description"><h1>Выбрать в Ellipstore</h1></div>  </div>  </a> </main> |

Листинг тега main страницы представления:

|  |
| --- |
| <main>  <a class="scroll-hidden" >  <div id="devicePresent" class="banner-image back-transp side">  <div class="banner-description"><h1>Страница не найдена</h1></div>  </div>  </a>   <a class="scroll-hidden">  <div id="deviceDescription" class="banner-image back-transp">  <div class="banner-full-description"><h1 style="color: rgb(217**,** 217**,** 217)**;**">Страница не найдена</h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  Попробуйте перейти по другим ссылкам  </p>  </div>  </div>  </a>   <div class="caps"></div>   <div id="deviceSets" class="banner-image back-transp side">  <div class="banner-full-description">  <h1 style="color: rgb(217**,** 217**,** 217)**;**"></h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  </p>  </div>  </div>   <a id="deviceStore">  <div class="banner-image back-transp" style="background-image: url('img/ellipse/storeSection1.png')">  <div class="banner-description-center"><h1>Купить в Ellipstore</h1></div>  </div>  </a> </main> |

Листинг тега main страницы магазина:

|  |
| --- |
| <main>  <a>  <div class="banner-image scroll-hidden light back-transp">  <div class="banner-full-description"><h1>ellipstore!</h1>  <p >  Выберите товар по душе  </p>  </div>  </div>  </a>  <div class=" row scroll-hidden">  <input id="Search" type="text" class="search-field" placeholder="Поиск">  </div>   <div class="light items-grid scroll-hidden">  <a href="storeitem.html?item=leaves">  <div class="grid-item">  <img src="img/Vapes/Leaves/Transparent/LeavesVapePresent21\_9.png">  <h1>Leaves</h1>  </div>  </a>  <a href="storeitem.html?item=tmtfs">  <div class="grid-item">  <img src="img/Vapes/TMT/46/Transparent/fstransparent2.png">  <h1>TMT 46</h1>  </div>  </a>  </div>   <a href="catalog.html" class="scroll-hidden">  <div class="banner-image light back-transp" style="background-image: url('img/ellipse/storeSection1.png')">  <div class="banner-full-description"><h1>Ознакомиться в каталоге</h1></div>  </div>  </a> </main> |

Листинг тега main страницы товара:

|  |
| --- |
| <main>  <a class="scroll-hidden">  <div id="devicePresent" class="banner-image back-transp light side">  <div class="banner-description"><h1>Страница не найдена</h1></div>  </div>  </a>   <a class="scroll-hidden">  <div id="deviceDescription" class="banner-image light back-transp info">  <div class="banner-full-description"><h1>Страница не найдена</h1>  </div>  </div>  </a>    <div id="deviceSets" class="banner-image light back-transp side">  <div class="banner-full-description info" style="background: url('img/ellipse/setsBackStore.png') fixed**;**">  <h1></h1>  <p>  </p>  </div>  </div>   <div class="row around wrap">  <h1 id="deviceCost">  0.00 BYN  </h1>  <button class="light fixed-sized" id="deviceForm">Купить</button>  <button class="light fixed-sized" id="deviceCatalog">Каталог</button>  </div> </main> |

Листинг тега main страницы истории:

|  |
| --- |
| <main>  <a class="scroll-hidden">  <div class="banner-image back-transp">  <div class="banner-full-description"><h1 style="color: rgb(217, 217, 217);">Инновации в дыхании</h1>  <div style="height: 2px; width: 90%; margin: 2% auto; left: 0px; right: 0px; background-color: rgb(217, 217, 217);"></div>  <p style="color: rgb(217, 217, 217);">Компания Ellipse появилась в конце 2022 года после появления идей о собственном парогенераторе. С тех пор наша компания продолжает развиваться в этой сфере для развития нашей экосистемы.  Всё с самого начала развивается ради идеи, которая поможет значительно уменьшить потребление табачной продукции во всём мире. Наши парогенераторы работают на нашей собственной операционной системе, которая может взаимодействовать с вашим мобильным устройством.  </p>  </div>  </div>  </a>   <a href="catalogitem.html?item=tmtfs" target="\_blank" class="scroll-hidden">  <div class="banner-image back-transp" style="background-image: url(img/ellipse/firstVapeSection.png);">  <div class="banner-description-center"><h1>TMT - Наш первый вейп</h1></div>  </div>  </a>   <a href="https://goo.gl/maps/7fzHEAGPwU9kHeSo8" target="\_blank" class="scroll-hidden">  <div class="banner-image back-transp" style="background-image: url('img/ellipse/headSection.png');">  <div class="banner-description-center"><h1>г. Минск, 704 - наш первый штаб</h1></div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp">  <div class="banner-full-description"><h1 style="color: rgb(217, 217, 217);">Спасибо за поддержку.</h1>  <div style="height: 2px; width: 90%; margin: 2% auto; left: 0px; right: 0px; background-color: rgb(217, 217, 217);"></div>  <p style="color: rgb(217, 217, 217);">Каждое наше устройство создается уникальным. С 2022 года мы тестируем их производительность и удобство для того, чтобы оно было от человека и для человека. В будущем мы планируем внедрить цифровую скидочную карту для студентов, чтобы наши девайсы были как можно доступнее для вас. Мы организовали наиболее удобный сервис поддержки для того, чтобы выяснение и решение проблем не было для вас препятствием. Предлагаем вам ознакомиться с каталогом наших устройств ниже. Надеемся, что вам это понравится.</p></div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp" style="background-image: url(img/ellipse/thanksSection.png)">  <div class="banner-full-description"><h1 style="color: rgb(217, 217, 217);">Отдельная благодарность...</h1>  <p style="color: rgb(217, 217, 217);">Анатолию Поборцеву и Глебу Борздову за моделирование устройств.</p></div>  </div>  </a>  <a href="catalog.html" class="scroll-hidden" >  <div class="banner-image back-transp" style="background-image: url('img/ellipse/storeSection1.png')">  <div class="banner-description-center"><h1>Каталог вейпов</h1></div>  </div>  </a> </main> |

Листинг тега main страницы общения:

|  |
| --- |
| <main>  <a class="scroll-hidden">  <div class="banner-image back-transp">  <div class="banner-full-description">  <h1 style="color: rgb(217**,** 217**,** 217)**;**">Платформа для общения</h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  Делитесь своими мыслями и отзывами в Ellipse Club  </p>  </div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp" style="background-image: url(img/ellipse/emotionsSection.png)**;**">  <div class="banner-description-center"><h1>Первые впечатления</h1></div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp" style="background-image: url('img/ellipse/communitySection.png')**;**">  <div class="banner-description-center"><h1>Новые знакомства</h1></div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp" style="background-image: url('img/ellipse/clubSection1.png')**;**">  <div class="banner-description-center"><h1>Вместе не скучно!</h1></div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp">  <div class="banner-full-description">  <h1 style="color: rgb(217**,** 217**,** 217)**;**">Присоединяйтесь!</h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  Оформите заявку уже сегодня и пользуйтесь неограниченными возможностями!  </p>  </div>  </div>  </a>   <div class="row around wrap">  <button class="fixed-sized" id="deviceForm">Регистрация</button>  </div> </main> |

Листинг тега main страницы поддержки:

|  |
| --- |
| <main>  <a class="scroll-hidden">  <div class="banner-image back-transp" style="background-image: url('img/ellipse/supportSection.png')">  <div class="banner-full-description">  <h1 style="color: rgb(217**,** 217**,** 217)**;**">Часто задаваемые вопросы</h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  Ниже будут ответы на вопросы, которые задают чаще всего.  </div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp">  <div class="banner-full-description">  <h1 style="color: rgb(217**,** 217**,** 217)**;**">Для чего нужен bluetooth на ваших устройствах?</h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  Наши вейп-устройства поддерживают bluetooth для того, чтобы можно было настраивать его с мобильного устройства  </p>  </div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp">  <div class="banner-full-description">  <h1 style="color: rgb(217**,** 217**,** 217)**;**">Устройство нагревается во время зарядки. Что с ним случилось?</h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  С ним все в порядке, во время зарядки нагревается аккумулятор устройства. Наши девайсы  распознают слишком высокую температуру и дадут вам это знать с помощью приложения на мобильном  устройстве либо с помощью индикатора на панели устройства.  В случае перегрева девайса отключите его от зарядного устройства и положите на временное хранение  при комнатной температуре. Если это не поможет, обратитесь к нашей поддержке(листайте ниже)  </p>   </div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp">  <div class="banner-full-description">  <h1 style="color: rgb(217**,** 217**,** 217)**;**">Сколько длится гарантия устройства?</h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  Гарантийный срок устройства после его покупки составляет 1 год (365 дней)  </p>  </div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp">  <div class="banner-full-description">  <h1 style="color: rgb(217**,** 217**,** 217)**;**">Уменьшилась интенсивность пара. В чем проблема?</h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  Наиболее частая проблема уменьшения интенсивности пара - расход испарителя устройства.  Для решения этой проблемы необходимо заменить картридж на новый. Не используйте поддельные  жидкости в нашем устройстве, они могут навредить вашему здоровью.  </p>  </div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp">  <div class="banner-full-description">  <h1 style="color: rgb(217**,** 217**,** 217)**;**">Гарантирует ли ваш товар, что я брошу сигареты и/или никотиносодержащие продукты?</h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  Нет. Наш товар не гарантирует избавление от никотиновой зависимости, а также чрезмерное использование  наших продуктов может вызывать побочные эффекты или вызвать отравление никотином.  </p>  </div>  </div>  </a>   <a class="scroll-hidden">  <div class="banner-image back-transp">  <div class="banner-full-description">  <h1 style="color: rgb(217**,** 217**,** 217)**;**">Не нашли вопрос или возникла проблема?</h1>  <p style="color: rgb(217**,** 217**,** 217)**;**">  Мы всегда готовы вам помочь! Поддержка работает круглосуточно  </p>  </div>  </div>  </a>   <div class="row around wrap">  <button class="dark fixed-sized" id="deviceForm">Поддержка</button>  </div> </main> |

ПРИЛОЖЕНИЕ Г

**Листинг SCSS и CSS**

|  |  |
| --- | --- |
| @font-face {  font-family: "SF Pro Display";  -webkit-font-feature-settings: normal;  -moz-font-feature-settings: normal;  src: url("font/SFPro/SFProDisplay-Regular.ttf"); }  // colors $dark: black; $dark-grad: linear-gradient(112.1deg, rgb(0, 0, 0) 11.4%, #192227 70.2%); $dark-grad2: linear-gradient(112.1deg, rgb(20, 26, 29) 11.4%, #272d34 70.2%); /\* $accent: #7eb6e0; \*/ $accent: #74d8f1; $mid: #383838; $higher: #484848; $lighter: #686868; $light: #d9d9d9; $light-grad: linear-gradient(112.1deg, #d9d9d9 11.4%, #96c1db 70.2%);  $font: "SF Pro Display";  $size: calc(1vw + 1vh); $smallsize: calc($size \* 0.5); $smallersize: calc($size \* 0.7); $defsize: calc($size \* 0.8); $midsize: calc($size \* 1.0); $bigsize: calc($size \* 2); $largesize: calc($size \* 2.8);  $hover: 0.05s ease-in-out; $slide: 0.8s ease-in-out; $dropdown: 0.4s ease-out;  $anim-appear: 0.6s appear ease-in-out; $background-animation: 10.6s background-animation linear infinite; $background-side: 30.6s background-side-side ease-in-out infinite;  \* {  padding: 0;  margin: 0;  font-family: -apple-system, BlinkMacSystemFont,'SF Pro Display', system-ui, "Roboto Light", serif;    box-sizing: border-box;  -webkit-box-sizing: border-box;  -moz-box-sizing: border-box; }  @mixin background-anim-set {  background-size: 200%;  animation: $background-side; }  @mixin icon-set {  padding: 0;  margin: 0;  display: inline;  fill: $light;  height: calc($size \* 2);  width: calc($size \* 6); }  @mixin def-text {  padding: 0;  margin: 0;  font-weight: 500;  font-size: $defsize; }  @mixin small-text {  @include def-text;  padding: 0;  margin: 0;  font-weight: 500;  font-size: $smallsize; }  @mixin smaller-text {  @include def-text;  padding: 0;  margin: 0;  font-weight: 500;  font-size: $smallersize; }   @mixin mid-text {  @include def-text;  font-weight: 600;  font-size: $midsize; }  @mixin big-text {  @include def-text;  font-weight: 600;  font-size: $bigsize; }  @mixin large-text {  @include def-text;  font-weight: 600;  font-size: $largesize; }  .mid-text {  @include mid-text; }   @mixin short {  max-height: 10%;  min-height: 10%; }  @mixin dark-theme {  background: $dark;  color: $light; }  @mixin light-theme {  color: $dark;  background: $light; }  @mixin flex {  display: flex;  justify-content: space-between; }  @mixin flex-end {  justify-content: flex-end; }  @mixin centered {  justify-content: space-around;  align-items: center; }  @mixin evenly {  justify-content: space-evenly; }  @mixin banner-set {  aspect-ratio: 21/9;  display: flex;  padding: 30px; }  @mixin banner-corner-set {  justify-content: flex-start;  align-items: flex-end; }  @mixin banner-center-set {  justify-content: center;  align-items: center; }  @mixin column {  flex-direction: column;  justify-content: center; }   .banner-image {  @include banner-set;  background-repeat: no-repeat;  overflow: hidden;  background-size: cover;  position: relative;  z-index: -100; }  .banner-image.light, .banner-image.light.back-transp{  background-size: cover; }  .info h1, .info p {  text-align: left;  align-self: flex-start; }  .banner-image::after {  content: "";  position: absolute;  top: 0;  left: 0;  width: 100%;  height: 100%;  background-color: rgba(0, 0, 0, 0.25); */\* replace with your desired color \*/* z-index: 0; }  .items-grid .search-hidden {  display: none; }  .items-grid .search-shown {  display: flex; }  .search-field {  @include flex-end;  @include mid-text;  justify-content: flex-start;  align-items: center;  justify-self: center;  flex-direction: row;  background: $dark-grad;  background-size: 200%;  background-position: 0% 0%;  border-radius: $size;  min-width: 80%;  min-height: calc($size\*2);  padding: $size;  color: $light;  transition: $slide; }   .search-field, .search-field:focus, .search-field:active{  border: unset;  outline: unset; }  .search-field:focus {  background-position: 80% 80%; }  .light .search-field {  background: $light-grad;  color: $dark; }  .banner-image.back-transp::after, .back-transp .banner-image::after, .light.back-transp {  background: rgba(0, 0, 0, 0);  }  .light.back-transp h1, .light.back-transp p{  text-outline: 3px $light;  text-shadow: #e0e0e0 1px 1px 0; }  .light-grad {  background: $light-grad; }  .back-transp h1, .back-transp p{  text-outline: 3px $dark;  */\**  *text-shadow: 3px 4px 7px rgba(0, 0, 0, 0.63);*  *\*/* text-shadow: #2a2a2a 1px 1px 0;  }  @mixin absolute-covered {  position: absolute;  top: 0;  bottom: 0;  left: 0;  right: 0; }  .banner-description {  @include def-text;  @include banner-set;  @include banner-corner-set;  @include absolute-covered;  vertical-align: center;   z-index: 1; }  .banner-description-center {  @include mid-text;  @include banner-set;  @include centered;  @include absolute-covered;  vertical-align: center;  align-self: center;  z-index: 1; }  .banner-full-description {  @include mid-text;  @include banner-set;  @include centered;  @include absolute-covered;  text-align: center;  vertical-align: center;  @include column;   z-index: 1; }  .banner-full-description p {  max-width: 80%;  font-weight: 400; }  .slide-item, .item-slider-view a {  @include flex;  @include centered;  margin: 0;  height: 80%;  aspect-ratio: 16/9;  color: $light;  position: relative;  */\**  *background-color: white;*  *\*/* z-index: 1;  transition: $slide; }  .slide-item.highlighted {  height: 100%; }   .slide-item figure {  display: flex;  flex-direction: column;  justify-content: space-between;  @include large-text;  max-height: 100%;  max-width: 100%; }  .slide-item figcaption {  text-align: center; }  .slide-item figure img {  max-height: 100%;  max-width: 100%; }  .slide-item::after {  content: "";  position: absolute;  width: 100%;  height: 100%;  background: $dark-grad;   z-index: -1;  border-radius: $size; }  .item-slider-view {  aspect-ratio: 7/3;  */\**  *background-image: url("notfound\_21\_9.jpg");*  *\*/* background-color: $accent;   display: inline-flex;  width: 100%;  padding: 0;  margin: 0;  overflow: hidden;  position: relative;  justify-content: center;  align-items: center; }  .slider-view {  display: inline-flex;  width: 100%;  padding: 0;  margin: 0;  overflow: hidden;  position: relative; }  .slider-view.fullscreen {  display: flex;  width: 100%;  padding: 0;  margin: 0; }  .non-interactive a, .non-interactive a:hover{  color: $light; }  .slider-view a {  display: grid;  min-width: 100%;  max-width: 1000%;  transition: $slide; }  .slider-view h1, .slider-view p {  transition: unset; }  .prev, .next, .slide-pointer-container {  position: absolute;  top: 50%;  height: 50px;  width: 50px;  transform: translateY(-50%);  z-index: 1; }  .prev, .next {  top: 0;  bottom: 0;  display: flex;  height: 100%;  width: calc($size \* 3);  padding: calc($size \* 0.4);  background-color: rgba(0,0,0,0.4);  transform: translateY(0%);  align-items: center;  transition: $slide;  opacity: 0; }  .banner-image .banner-description {  min-height: 100%;  min-width: 100%; }  .slider-view:hover .prev, .slider-view:hover .next {  opacity: 1; }   .slide-pointer-container {  top: unset;  right: 50%;  bottom: 10%;  width: auto;  height: auto;  transform: translateX(50%); }  .slide-pointer {  border-radius: 100%;  min-height: calc($size \* 0.4);  min-width: calc($size \* 0.4);  background-color: $light;  transition: $slide;  margin: calc($size \* 0.2) }  .slide-pointer.highlighted {  background-color: $accent; }  .prev {  left: 0; }  .next {  right: 0; }  .padded {  padding: calc($size \* 1); }  .footer-disclaimer {  @include smaller-text;  @include centered;  @include column;  display:flex;  align-content: space-around;  padding: calc($size \* 1);  text-align: center;  background: $dark; }  .footer-disclaimer p {  font-weight: 700;  max-width: 70%; }  .footer-list-head {  @include mid-text;  text-transform: uppercase;  font-weight: 500;  padding-bottom: calc($size \* 1); } .footer-list-item a, .footer-list-item p {  @include smaller-text;  color: $lighter;  font-weight: 400; }  .footer-list-item {  padding-bottom: calc($size \* 0.15); }  .footer-list-item:hover {  */\*transform: scale(1.2) translateX(10%);\*/* *}*  *.*footer-list {  color: $light;  list-style-type: none;  transition: $hover; }  button {  width: calc(9\*$size);  height: calc(2 \* $size);  background: $light-grad;  color: $dark;  border-radius: $size;  padding: $size;  @include smaller-text;  background-size: 400%;  background-position: 0 0;  z-index: 5;  justify-self: center; }   button, button:hover, button:focus {  outline: unset;  border: unset; }  button:hover, button::selection{  opacity: 0.7; }  button.light {  background: $dark-grad;  color: $light; }  .row.around {  justify-content: space-evenly;  align-items: center; }  .around button {  margin: $size; }  .banner-full-description button:hover {  background-position: 100% 100%; }  .info input {  align-self: flex-start; }  .column {  @include column; }  body {  margin: 0;  padding: 0;  background: black;  max-width: 100%; }    body.light {  background: #d9d9d9; }  .dark {  @include dark-theme; } .light {  @include light-theme; }  .def-text {  @include def-text; }  a {  text-decoration: none;  color: $light; }  a:hover {  color: $accent;  transition: $hover; }  a:hover .banner-image.dark {  color: $light; }  /\*h1, h2, h3, h4, h5, h6, p, span {  font-family: "SF Pro Display"; }\*/  // SVG  .icon-interactive:hover path, .icon-interactive:hover circle, .icon-interactive:hover ellipse, .icon-interactive:hover rect, .icon-interactive:hover text {  fill: $accent;  transition: $hover; } .icon {  @include icon-set; }  .wide {  width: calc($size \* 2); }  .short {  width: calc($size \* 10); }  .light .icon path, .light .icon circle, .light .icon ellipse, .light .icon rect, .light .icon text, .light-theme .icon path, .light-theme .icon circle, .light-theme .icon ellipse, .light-theme .icon rect, .light-theme .icon text{  fill: $dark; }  .short {  @include short; }  // icons  .row {  @include flex;  justify-content: center;  align-items: center;  flex-direction: row;  max-width: 100%;  padding: 0.5em; }  .flex {  @include flex; }  .flex-center {  @include centered; }  @keyframes appear {  0% {  filter: blur(10px);  opacity: 0;  }  100% {  opacity: 1;  filter: unset;  } }  @keyframes background-animation {  0% {  background-position: 20% 80%;  }  50% {  background-position: 80% 80%;  }  75% {  background-position: 90% 40%;  }  100% {  background-position: 20% 80%;  } }  @keyframes background-side-side {  0% {  background-position: 0 0;  }  50% {  background-position: 30% 30%;  }  100% {  background-position: 0 0;  } }  .dropdown-set.dark {  color: $light;  background: $dark-grad;  background-size: 400% 400%;  animation: $background-animation; }  .dropdown-set.light {  color: $dark;  background: $light-grad;  background-size: 400% 400%;  animation: $background-animation; }  .dropdown-set {  position: fixed;  opacity: 0;  z-index: 10;  min-width: 90%;  left: 5%;  right: 5%;  min-height: unset;  max-height: unset;  overflow: hidden;  transition: $slide;  justify-content: space-evenly;  align-items: center;  top: -100%;  border-radius: $size;  box-shadow: rgba(50, 50, 93, 0.25)6px 0px inset;  filter: blur(5px);  background-size: 400% 400%;  animation: $background-animation; }  .dropdown-set div {  margin: 0; }  .dropdown-set div:first-child a {  transition: $slide;  @include mid-text;  padding: calc($size \* 0.4); }  .dropdown-set div:last-child {  transition: unset;  min-width: 90%;  min-height: 50%; }  .dropdown-shown {  top:0;  transform: translateY(calc($size \* 3)) scale(1);  opacity: 1;  filter: unset; }  .dropdown-set .banner-image::after {  background-color: transparent; }  .dropdown-focused {  color: $accent; }  .dropdown-focused .banner-image {  display: block;  animation: $anim-appear; }  .notification {  justify-content: center;  @include smaller-text; }  .notification .icon {  margin-right: calc($size/1.5); }  .notification {  z-index: 100;  padding: calc($size/1.5);  min-width: calc($size);  max-width: calc($size\*40);  left: 50%;  transform: translateX(-50%);  height: calc($size\*3);  justify-self: center; }  .notification.dropdown-shown:hover {  box-shadow: rgba(48, 93, 120, 0.25) 6px 0px inset; }  .notification.dropdown-shown {  transform: translateY(calc($size \* 3)) translateX(-50%); }  @mixin mobile-set() {  aspect-ratio: 9/12;  @include mid-text; }  .mobile-aspect-ratio {  @include mobile-set; }  .mobile-aspect-ratio .banner-description{  @include mobile-set; }  .mobile-aspect-ratio .banner-full-description p {  font-size: calc($size \* 1.8); }  .mobile-aspect-ratio .banner-description h1 {  padding: $size; }  .mobile {  display: none; }   .scroll-shown .grid-item {  transform: unset;  opacity: 1; }  .scroll-hidden .banner-image, nav.scroll-hidden {  transform: unset;  opacity: 0;  filter: blur(calc($size/10));  transition: $slide;  z-index: -1; }  .scroll-hidden .banner-image.non-fade {  transform: unset;  opacity: 1;  filter: unset;  transition: $slide;  z-index: -1; }  .scroll-hidden .banner-description {  transform: translateX(-25%) ;  opacity: 0;  filter: blur(calc($size/10));  transition: $slide; }  .scroll-hidden .banner-full-description h1 {  transform: translateY(calc(-1 \* $size \* 5));  opacity: 0;  filter: blur(calc($size/10));  transition: $slide; }  .scroll-shown .banner-full-description h1 {  transform: unset;  opacity: 1;  filter: unset;  transition-delay: 300ms; }  .scroll-hidden .banner-full-description p {  transform: translateY(calc($size \* 5)) ;  opacity: 0;  filter: blur(calc($size/10));  transition: $slide; }  .scroll-shown .banner-full-description p {  transform: unset;  opacity: 1;  filter: unset;  transition-delay: 600ms; }  .scroll-shown .banner-image {  transform: unset;  opacity: 1;  filter: unset; }  //semantic elements  header {  @include short;  max-width: 100%;  position: sticky;  top: 0;  z-index: 100; }  .foot-container {  flex-direction: row; }  .back-grad.dark {  background: $dark-grad; }   .back-grad.light {  background: $light-grad; }  .dark {  @include dark-theme; }  footer {  position: sticky;  z-index: 150; }  .wrap {  flex-wrap: wrap; }  button.fixed-sized {  height: 70px;  width: 290px;  font-size: 36px;  font-weight: 700;  border-radius: 50px; }  .foot-label {  z-index: 102;  top: 0;  position: sticky; } .foot-container {  position: relative;  align-items: flex-start;  z-index: 101; }  .grid-item {  width: calc(300px);  height: calc(300px);  aspect-ratio: 1/1;  background: $dark-grad;  color: $light;  margin: calc($size\*2);  overflow: hidden;  border-radius: calc($size\* 1.5);  transition: $dropdown;  background-size: 400%;  @include flex-end;   flex-direction: column; }  .grid-item:hover, .grid-item:focus {  transition-delay: 0ms;  background-size: 100%;  background-position: 50% 80%; }   form {  @include flex;  flex-direction: column;  justify-content: center;  padding: $size; }  .items-grid.light .grid-item {  transition: $dropdown;  color: $light; }  .items-grid.dark .grid-item {  background: $light-grad; }  @media (max-width: 600px) {   body {  -webkit-font-feature-settings: normal;  -webkit-text-size-adjust: 100%;  }  .foot-container {  display: block;  }   .desktop {  display: none;  }  .mobile {  display: block;  }   .footer-list {  padding: calc($size \* 1.5);  }   .footer-list-item a {  @include big-text;  font-weight: 400;  }   .footer-list-head {  @include big-text;  font-weight: 400;  padding-bottom: calc($size \* 1);   }   .notification {  height: calc($size\*6);  width: 90%;  max-height: calc($size\*100);  @include big-text;  }    .icon {  height: calc($size \* 4);  width: calc($size \* 12);  }   .wide {  width: calc($size \* 4);  }   .dropdown-shown {  transform: translateY(calc($size \* 7.5)) scale(1);  }   .footer-disclaimer {  @include big-text;  padding: $size;  }    .footer-disclaimer p {  max-width: 90%  }   .grid-item h1 {  @include big-text;  }   .item-slider-view {  aspect-ratio: 3/4;  }   .slide-item {  aspect-ratio: 14/16;  }   .grid-item {  width: calc(250px);  height: calc(250px);  }   .search-field {  @include big-text;  min-height: calc($size\*4);  padding: $size;  }  .info h1,  .info p {  text-align: center;  align-self: center;  }   .banner-image,  .banner-image .banner-description-center,  .banner-image .banner-description,  .banner-image .banner-full-description, {  background: no-repeat center center;  }   .banner-image.back-transp.light,  .banner-image.back-transp.side{  background: rgba(0,0,0,0) no-repeat -size: cover;  @include mobile-set;  }   .banner-full-description p {  font-size: calc($size \* 1.8);  }   .banner-description h1 {  padding: $size;  }  } | @font-face {  font-family: "SF Pro Display";  -webkit-font-feature-settings: normal;  -moz-font-feature-settings: normal;  src: url("font/SFPro/SFProDisplay-Regular.ttf"); }  \* {  padding: 0;  margin: 0;  font-family: -apple-system, BlinkMacSystemFont, "SF Pro Display", system-ui, "Roboto Light", serif;  box-sizing: border-box;  -webkit-box-sizing: border-box;  -moz-box-sizing: border-box; }  .mid-text {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  font-weight: 600;  font-size: calc((1vw + 1vh) \* 1); }  .banner-image {  aspect-ratio: 21/9;  display: flex;  padding: 30px;  background-repeat: no-repeat;  overflow: hidden;  background-size: cover;  position: relative;  z-index: -100; }  .banner-image.light, .banner-image.light.back-transp {  background-size: cover; }  .info h1, .info p {  text-align: left;  align-self: flex-start; }  .banner-image::after {  content: "";  position: absolute;  top: 0;  left: 0;  width: 100%;  height: 100%;  background-color: rgba(0, 0, 0, 0.25); /\* replace with your desired color \*/  z-index: 0; }  .items-grid .search-hidden {  display: none; }  .items-grid .search-shown {  display: flex; }  .search-field {  justify-content: flex-end;  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  font-weight: 600;  font-size: calc((1vw + 1vh) \* 1);  justify-content: flex-start;  align-items: center;  justify-self: center;  flex-direction: row;  background: linear-gradient(112.1deg, rgb(0, 0, 0) 11.4%, #192227 70.2%);  background-size: 200%;  background-position: 0% 0%;  border-radius: calc(1vw + 1vh);  min-width: 80%;  min-height: calc((1vw + 1vh) \* 2);  padding: calc(1vw + 1vh);  color: #d9d9d9;  transition: 0.8s ease-in-out; }  .search-field, .search-field:focus, .search-field:active {  border: unset;  outline: unset; }  .search-field:focus {  background-position: 80% 80%; }  .light .search-field {  background: linear-gradient(112.1deg, #d9d9d9 11.4%, #96c1db 70.2%);  color: black; }  .banner-image.back-transp::after, .back-transp .banner-image::after, .light.back-transp {  background: rgba(0, 0, 0, 0); }  .light.back-transp h1, .light.back-transp p {  text-outline: 3px #d9d9d9;  text-shadow: #e0e0e0 1px 1px 0; }  .light-grad {  background: linear-gradient(112.1deg, #d9d9d9 11.4%, #96c1db 70.2%); }  .back-transp h1, .back-transp p {  text-outline: 3px black;  /\*  text-shadow: 3px 4px 7px rgba(0, 0, 0, 0.63);  \*/  text-shadow: #2a2a2a 1px 1px 0; }  .banner-description {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  aspect-ratio: 21/9;  display: flex;  padding: 30px;  justify-content: flex-start;  align-items: flex-end;  position: absolute;  top: 0;  bottom: 0;  left: 0;  right: 0;  vertical-align: center;  z-index: 1; }  .banner-description-center {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  font-weight: 600;  font-size: calc((1vw + 1vh) \* 1);  aspect-ratio: 21/9;  display: flex;  padding: 30px;  justify-content: space-around;  align-items: center;  position: absolute;  top: 0;  bottom: 0;  left: 0;  right: 0;  vertical-align: center;  align-self: center;  z-index: 1; }  .banner-full-description {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  font-weight: 600;  font-size: calc((1vw + 1vh) \* 1);  aspect-ratio: 21/9;  display: flex;  padding: 30px;  justify-content: space-around;  align-items: center;  position: absolute;  top: 0;  bottom: 0;  left: 0;  right: 0;  text-align: center;  vertical-align: center;  flex-direction: column;  justify-content: center;  z-index: 1; }  .banner-full-description p {  max-width: 80%;  font-weight: 400; }  .slide-item, .item-slider-view a {  display: flex;  justify-content: space-between;  justify-content: space-around;  align-items: center;  margin: 0;  height: 80%;  aspect-ratio: 16/9;  color: #d9d9d9;  position: relative;  /\*  background-color: white;  \*/  z-index: 1;  transition: 0.8s ease-in-out; }  .slide-item.highlighted {  height: 100%; }  .slide-item figure {  display: flex;  flex-direction: column;  justify-content: space-between;  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  font-weight: 600;  font-size: calc((1vw + 1vh) \* 2.8);  max-height: 100%;  max-width: 100%; }  .slide-item figcaption {  text-align: center; }  .slide-item figure img {  max-height: 100%;  max-width: 100%; }  .slide-item::after {  content: "";  position: absolute;  width: 100%;  height: 100%;  background: linear-gradient(112.1deg, rgb(0, 0, 0) 11.4%, #192227 70.2%);  z-index: -1;  border-radius: calc(1vw + 1vh); }  .item-slider-view {  aspect-ratio: 7/3;  /\*  background-image: url("notfound\_21\_9.jpg");  \*/  background-color: #74d8f1;  display: inline-flex;  width: 100%;  padding: 0;  margin: 0;  overflow: hidden;  position: relative;  justify-content: center;  align-items: center; }  .slider-view {  display: inline-flex;  width: 100%;  padding: 0;  margin: 0;  overflow: hidden;  position: relative; }  .slider-view.fullscreen {  display: flex;  width: 100%;  padding: 0;  margin: 0; }  .non-interactive a, .non-interactive a:hover {  color: #d9d9d9; }  .slider-view a {  display: grid;  min-width: 100%;  max-width: 1000%;  transition: 0.8s ease-in-out; }  .slider-view h1, .slider-view p {  transition: unset; }  .prev, .next, .slide-pointer-container {  position: absolute;  top: 50%;  height: 50px;  width: 50px;  transform: translateY(-50%);  z-index: 1; }  .prev, .next {  top: 0;  bottom: 0;  display: flex;  height: 100%;  width: calc((1vw + 1vh) \* 3);  padding: calc((1vw + 1vh) \* 0.4);  background-color: rgba(0, 0, 0, 0.4);  transform: translateY(0%);  align-items: center;  transition: 0.8s ease-in-out;  opacity: 0; }  .banner-image .banner-description {  min-height: 100%;  min-width: 100%; }  .slider-view:hover .prev, .slider-view:hover .next {  opacity: 1; }  .slide-pointer-container {  top: unset;  right: 50%;  bottom: 10%;  width: auto;  height: auto;  transform: translateX(50%); }  .slide-pointer {  border-radius: 100%;  min-height: calc((1vw + 1vh) \* 0.4);  min-width: calc((1vw + 1vh) \* 0.4);  background-color: #d9d9d9;  transition: 0.8s ease-in-out;  margin: calc((1vw + 1vh) \* 0.2); }  .slide-pointer.highlighted {  background-color: #74d8f1; }  .prev {  left: 0; }  .next {  right: 0; }  .padded {  padding: calc((1vw + 1vh) \* 1); }  .footer-disclaimer {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.7);  justify-content: space-around;  align-items: center;  flex-direction: column;  justify-content: center;  display: flex;  align-content: space-around;  padding: calc((1vw + 1vh) \* 1);  text-align: center;  background: black; }  .footer-disclaimer p {  font-weight: 700;  max-width: 70%; }  .footer-list-head {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  font-weight: 600;  font-size: calc((1vw + 1vh) \* 1);  text-transform: uppercase;  font-weight: 500;  padding-bottom: calc((1vw + 1vh) \* 1); }  .footer-list-item a, .footer-list-item p {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.7);  color: #686868;  font-weight: 400; }  .footer-list-item {  padding-bottom: calc((1vw + 1vh) \* 0.15); }  .footer-list-item:hover {  /\*transform: scale(1.2) translateX(10%);\*/ }  .footer-list {  color: #d9d9d9;  list-style-type: none;  transition: 0.05s ease-in-out; }  button {  width: calc(9 \* (1vw + 1vh));  height: calc(2 \* (1vw + 1vh));  background: linear-gradient(112.1deg, #d9d9d9 11.4%, #96c1db 70.2%);  color: black;  border-radius: calc(1vw + 1vh);  padding: calc(1vw + 1vh);  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.7);  background-size: 400%;  background-position: 0 0;  z-index: 5;  justify-self: center; }  button, button:hover, button:focus {  outline: unset;  border: unset; }  button:hover, button::selection {  opacity: 0.7; }  button.light {  background: linear-gradient(112.1deg, rgb(0, 0, 0) 11.4%, #192227 70.2%);  color: #d9d9d9; }  .row.around {  justify-content: space-evenly;  align-items: center; }  .around button {  margin: calc(1vw + 1vh); }  .banner-full-description button:hover {  background-position: 100% 100%; }  .info input {  align-self: flex-start; }  .column {  flex-direction: column;  justify-content: center; }  body {  margin: 0;  padding: 0;  background: black;  max-width: 100%; }  body.light {  background: #d9d9d9; }  .dark {  background: black;  color: #d9d9d9; }  .light {  color: black;  background: #d9d9d9; }  .def-text {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8); }  a {  text-decoration: none;  color: #d9d9d9; }  a:hover {  color: #74d8f1;  transition: 0.05s ease-in-out; }  a:hover .banner-image.dark {  color: #d9d9d9; }  /\*h1, h2, h3, h4, h5, h6, p, span {  font-family: "SF Pro Display"; }\*/ .icon-interactive:hover path, .icon-interactive:hover circle, .icon-interactive:hover ellipse, .icon-interactive:hover rect, .icon-interactive:hover text {  fill: #74d8f1;  transition: 0.05s ease-in-out; }  .icon {  padding: 0;  margin: 0;  display: inline;  fill: #d9d9d9;  height: calc((1vw + 1vh) \* 2);  width: calc((1vw + 1vh) \* 6); }  .wide {  width: calc((1vw + 1vh) \* 2); }  .short {  width: calc((1vw + 1vh) \* 10); }  .light .icon path, .light .icon circle, .light .icon ellipse, .light .icon rect, .light .icon text, .light-theme .icon path, .light-theme .icon circle, .light-theme .icon ellipse, .light-theme .icon rect, .light-theme .icon text {  fill: black; }  .short {  max-height: 10%;  min-height: 10%; }  .row {  display: flex;  justify-content: space-between;  justify-content: center;  align-items: center;  flex-direction: row;  max-width: 100%;  padding: 0.5em; }  .flex {  display: flex;  justify-content: space-between; }  .flex-center {  justify-content: space-around;  align-items: center; }  @keyframes appear {  0% {  filter: blur(10px);  opacity: 0;  }  100% {  opacity: 1;  filter: unset;  } } @keyframes background-animation {  0% {  background-position: 20% 80%;  }  50% {  background-position: 80% 80%;  }  75% {  background-position: 90% 40%;  }  100% {  background-position: 20% 80%;  } } @keyframes background-side-side {  0% {  background-position: 0 0;  }  50% {  background-position: 30% 30%;  }  100% {  background-position: 0 0;  } } .dropdown-set.dark {  color: #d9d9d9;  background: linear-gradient(112.1deg, rgb(0, 0, 0) 11.4%, #192227 70.2%);  background-size: 400% 400%;  animation: 10.6s background-animation linear infinite; }  .dropdown-set.light {  color: black;  background: linear-gradient(112.1deg, #d9d9d9 11.4%, #96c1db 70.2%);  background-size: 400% 400%;  animation: 10.6s background-animation linear infinite; }  .dropdown-set {  position: fixed;  opacity: 0;  z-index: 10;  min-width: 90%;  left: 5%;  right: 5%;  min-height: unset;  max-height: unset;  overflow: hidden;  transition: 0.8s ease-in-out;  justify-content: space-evenly;  align-items: center;  top: -100%;  border-radius: calc(1vw + 1vh);  box-shadow: rgba(50, 50, 93, 0.25) 0 calc((1vw + 1vh) \* 0.5) calc(1vw + 1vh) calc((1vw + 1vh) \* -0.2), rgba(0, 0, 0, 0.3) 0 calc((1vw + 1vh) \* 0.3) calc((1vw + 1vh) \* 0.6) calc((1vw + 1vh) \* -0.3), rgba(10, 37, 64, 0.35) 0px -2px 6px 0px inset;  filter: blur(5px);  background-size: 400% 400%;  animation: 10.6s background-animation linear infinite; }  .dropdown-set div {  margin: 0; }  .dropdown-set div:first-child a {  transition: 0.8s ease-in-out;  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  font-weight: 600;  font-size: calc((1vw + 1vh) \* 1);  padding: calc((1vw + 1vh) \* 0.4); }  .dropdown-set div:last-child {  transition: unset;  min-width: 90%;  min-height: 50%; }  .dropdown-set div:last-child a {  transition: 0.8s ease-in-out;  border: none; }  .dropdown-set .banner-image {  display: none;  transition: unset;  animation: unset;  border-radius: calc(1vw + 1vh); }  .dropdown-shown {  top: 0;  transform: translateY(calc((1vw + 1vh) \* 3)) scale(1);  opacity: 1;  filter: unset; }  .dropdown-set .banner-image::after {  background-color: transparent; }  .dropdown-focused {  color: #74d8f1; }  .dropdown-focused .banner-image {  display: block;  animation: 0.6s appear ease-in-out; }  .notification {  justify-content: center;  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8); }  .notification .icon {  margin-right: calc((1vw + 1vh) / 1.5); }  .notification {  z-index: 100;  padding: calc((1vw + 1vh) / 1.5);  min-width: calc(1vw + 1vh);  max-width: calc((1vw + 1vh) \* 40);  left: 50%;  transform: translateX(-50%);  height: calc((1vw + 1vh) \* 3);  justify-self: center; }  .notification.dropdown-shown {  transform: translateY(calc((1vw + 1vh) \* 3)) translateX(-50%); }  .mobile-aspect-ratio {  aspect-ratio: 9/12;  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8); }  .mobile-aspect-ratio .banner-description, .mobile-aspect-ratio .banner-description-center, .mobile-aspect-ratio .banner-full-description {  aspect-ratio: 9/12;  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8); }  .mobile-aspect-ratio .banner-full-description p {  font-size: calc((1vw + 1vh) \* 1.8); }  .mobile-aspect-ratio .banner-description h1 {  padding: calc(1vw + 1vh); }  .mobile {  display: none; }  .scroll-hidden .grid-item {  transform: translateX(-25%);  opacity: 0; }  .scroll-shown .grid-item {  transform: unset;  opacity: 1; }  .scroll-hidden .banner-image.non-fade {  transform: unset;  opacity: 1;  filter: unset;  transition: 0.8s ease-in-out;  z-index: -1; }  .scroll-hidden .banner-full-description h1 {  transform: translateY(calc(-1 \* (1vw + 1vh) \* 5));  opacity: 0;  filter: blur(calc((1vw + 1vh) / 10));  transition: 0.8s ease-in-out; }  .scroll-shown .banner-full-description h1 {  transform: unset;  opacity: 1;  filter: unset;  transition-delay: 300ms; }  .scroll-hidden .banner-full-description p {  transform: translateY(calc((1vw + 1vh) \* 5));  opacity: 0;  filter: blur(calc((1vw + 1vh) / 10));  transition: 0.8s ease-in-out; }  .scroll-shown .banner-full-description p {  transform: unset;  opacity: 1;  filter: unset;  transition-delay: 600ms; }  .scroll-shown .banner-description, .scroll-shown .banner-description-center, nav.scroll-shown {  transform: unset;  opacity: 1;  filter: unset;  transition-delay: 400ms; }  .scroll-shown .banner-image {  transform: unset;  opacity: 1;  filter: unset; }  header {  max-height: 10%;  min-height: 10%;  max-width: 100%;  position: sticky;  top: 0;  z-index: 100; }  .back-grad.light {  background: linear-gradient(112.1deg, #d9d9d9 11.4%, #96c1db 70.2%); }  .dark {  background: black;  color: #d9d9d9; }  footer {  position: sticky;  z-index: 150; }  .wrap {  flex-wrap: wrap; }  button.fixed-sized {  height: 70px;  width: 290px;  font-size: 36px;  font-weight: 700;  border-radius: 50px; }  .foot-label {  z-index: 102;  top: 0;  position: sticky; }  .foot-container {  position: relative;  align-items: flex-start;  z-index: 101; }  .grid-item {  width: 300px;  height: 300px;  aspect-ratio: 1/1;  background: linear-gradient(112.1deg, rgb(0, 0, 0) 11.4%, #192227 70.2%); }  .grid-item:hover, .grid-item:focus {  transition-delay: 0ms;  background-size: 100%;  background-position: 50% 80%; }  .grid-item img, .grid-item.light img {  max-height: 80%; }  form input[type=text]:focus {  outline: unset;  border: unset; }  @media (max-width: 600px) {  body {  -webkit-font-feature-settings: normal;  -webkit-text-size-adjust: 100%;  }  .foot-container {  display: block;  }  .desktop {  display: none;  }  .mobile {  display: block;  }  .footer-list {  padding: calc((1vw + 1vh) \* 1.5);  }  .footer-list-item a {  padding: 0;  margin: 0;  .footer-list-head {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  padding-bottom: calc((1vw + 1vh) \* 1);  }  .notification {  height: calc((1vw + 1vh) \* 6);  width: 90%;  max-height: calc((1vw + 1vh) \* 100);  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);    }  .icon {  height: calc((1vw + 1vh) \* 4);  width: calc((1vw + 1vh) \* 12);  }  .wide {  width: calc((1vw + 1vh) \* 4);  }  .dropdown-shown {  transform: translateY(calc((1vw + 1vh) \* 7.5)) scale(1);  }  .footer-disclaimer {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  padding: calc(1vw + 1vh);  }  .dropdown-set div:first-child a {  text-align: center;  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  }  .footer-disclaimer p {  max-width: 90%;  }  .grid-item h1 {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  font-weight: 600;  font-size: calc((1vw + 1vh) \* 2);  }  .item-slider-view {  aspect-ratio: 3/4;  }  .slide-item {  aspect-ratio: 14/16;  }  .banner-full-description {  font-size: unset;  }  .grid-item {  width: 250px;  height: 250px;  }  .search-field {  padding: 0;  margin: 0;  font-weight: 500;  font-size: calc((1vw + 1vh) \* 0.8);  font-weight: 600;  font-size: calc((1vw + 1vh) \* 2);  min-height: calc((1vw + 1vh) \* 4);  padding: calc(1vw + 1vh);  }  .info h1,  .info p {  text-align: center;  align-self: center;  }  .banner-image,  .banner-image .banner-   .banner-image.back-transp.light,  .banner-image.back-transp.side {  background-size: cover;  aspect-ratio: 9/12;  padding: 0;  margin: 0; 1vh) \* 0.8);  font-weight: 600;  font-size: calc((1vw + 1vh) \* 1);  }  .banner-full-description p {  font-size: calc((1vw + 1vh) \* 1.8);  }  .banner-description h1 {  padding: calc(1vw + 1vh);  } } |

ПРИЛОЖЕНИЕ Д

**Листинг XML-файлов**

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" standalone="yes" ?>  <items>  <item name="leaves">  <name>Leaves</name>  <catalogImage>  img/Vapes/Leaves/Present/1\_21\_9.jpg  </catalogImage>  <shopImage>  img/Vapes/Leaves/Present/1\_21\_9.jpg  </shopImage>  <tagLine>  Лес ваших фантазий  </tagLine>  <description>  Эволюционный скачок паровых устройств. Он поможет тем, кто хочет отказаться от никотиновой зависимости. Мягкий и стильный дизайн и внутреннее противостояние всему искусственному.  </description>  <sets>  Размер: высота – 103.9 мм, ширина – 23.4 мм, толщина – 23.4 мм  Аккумулятор: 3000 мАч  Ток зарядки: 1А  Мощность: 20/26 Вт  Ёмкость картриджа: 1.5 мл  Сопротивление испарителя: 0.4 Ом  </sets>  <secondImage>  img/Vapes/Leaves/Transparent/LeavesVapePresent2\_21\_9.png  </secondImage>  <capabilities>  <capability>  <capocaption>Лёгкий парогенератор</capocaption>  <capoimage>img/Vapes/Leaves/Capabilities/leavescap1.png</capoimage>  </capability>  <capability>  <capocaption>Натуральные компоненты</capocaption>  <capoimage>img/Vapes/Leaves/Capabilities/leavescap2.png</capoimage>  </capability>  <capability>  <capocaption>Следите за потреблением</capocaption>  <capoimage>img/Vapes/Leaves/Capabilities/leavescap3.png</capoimage>  </capability>  </capabilities>  <cost>  79.99  </cost>  <sidePhoto>  img/Vapes/Leaves/Transparent/LeavesVapePresent2\_21\_9.png  </sidePhoto>  </item>  <item name="tmtfs">  <name>TMT 46</name>  <catalogImage>  img/Vapes/TMT/46/Present/fspresent.jpg  </catalogImage>  <shopImage>  img/Vapes/TMT/46/Present/fspresent.jpg  </shopImage>  <tagLine>  Создай волну  </tagLine>  <description>  Самый навороченный на рынке. Технология VapeTouch со встроенным Taptic Engine 5 подарит вам невероятный опыт использования и массу положительных впечатлений, а технология TransMediate позволит вам ощутить всю палитру вкусов. Внимание: на данный момент предоставляется предзаказ устройства. Купить устройство сразу можно будет 16 июня.  </description>  <sets>  Размер: высота – 136.9 мм, ширина – 92.4 мм, толщина – 13.4 мм  Аккумулятор: 6000 мАч  Ток зарядки: 2А  Мощность: 50/59 Вт  Ёмкость картриджа: 1.9 мл  Сопротивление испарителя: 0.4 Ом  </sets>  <secondImage>  img/Vapes/TMT/46/Transparent/fstransparent2.png  </secondImage>  <capabilities>  <capability>  <capocaption>  Инновационная структура  </capocaption>  <capoimage>  img/Vapes/TMT/46/Capabilities/fscapSection1.png  </capoimage>  </capability>  <capability>  <capocaption>  Эргономичный дизайн  </capocaption>  <capoimage>  img/Vapes/TMT/46/Capabilities/fscapSection2.png   </capoimage>  </capability>  <capability>  <capocaption>  Непокоримая мощность  </capocaption>  <capoimage>  img/Vapes/TMT/46/Capabilities/fscapSection3.png  </capoimage>  </capability>  </capabilities>  <cost>  169.99  </cost>  <sidePhoto>  img/Vapes/TMT/46/Transparent/fstransparent2.png  </sidePhoto>  </item> </items> |

ПРИЛОЖЕНИЕ Е

**Листинг SVG**

Листинг иконки компании:

|  |
| --- |
| <svg viewBox="0 0 1000 250" fill="none" xmlns="http://www.w3.org/2000/svg">  <g clip-path="url(#clip0\_1\_23)">  <path d="M312.087 68.0598C304.967 58.2567 290.979 49.3177 271.527 42.1413C252.075 34.9649 227.846 29.804 201.274 27.1773C174.702 24.5505 146.725 24.5505 120.153 27.1773C93.5815 29.804 69.3522 34.9649 49.9004 42.1413C30.4486 49.3177 16.4597 58.2567 9.33989 68.0598C2.22004 77.8629 2.22004 88.1848 9.33988 97.9879C16.4597 107.791 30.4485 116.73 49.9003 123.906C69.3521 131.083 93.5815 136.244 120.153 138.87L160.714 83.0238L312.087 68.0598Z" fill="#D9D9D9"/>  <path d="M9.33989 185.859C16.4597 195.663 30.4486 204.602 49.9004 211.778C69.3522 218.954 93.5815 224.115 120.153 226.742C146.725 229.369 174.702 229.369 201.274 226.742C227.846 224.115 252.075 218.954 271.527 211.778C290.979 204.602 304.967 195.663 312.087 185.859C319.207 176.056 319.207 165.735 312.087 155.931C304.967 146.128 290.979 137.189 271.527 130.013C252.075 122.837 227.846 117.676 201.274 115.049L160.714 170.895L9.33989 185.859Z" fill="#D9D9D9"/>  <path d="M536.958 106.904C531.101 106.904 526.202 108.931 522.26 112.986C518.374 116.984 516.206 122.137 515.755 128.444H557.569C557.344 122.081 555.345 116.9 551.572 112.901C547.799 108.903 542.928 106.904 536.958 106.904ZM557.654 155.222H577.336C575.984 163.726 571.676 170.624 564.412 175.918C557.147 181.155 548.249 183.774 537.718 183.774C524.315 183.774 513.784 179.663 506.126 171.441C498.467 163.163 494.637 151.871 494.637 137.567C494.637 123.376 498.439 112 506.041 103.44C513.7 94.8805 523.977 90.6006 536.874 90.6006C549.545 90.6006 559.597 94.6834 567.03 102.849C574.464 110.958 578.181 121.827 578.181 135.456V142.298H515.587V143.565C515.587 150.773 517.614 156.574 521.669 160.966C525.78 165.359 531.27 167.555 538.141 167.555C542.984 167.555 547.151 166.457 550.643 164.261C554.191 162.008 556.528 158.995 557.654 155.222ZM592.372 182V60.106H613.406V182H592.372ZM631.905 182V60.106H652.939V182H631.905ZM671.354 182V92.459H692.388V182H671.354ZM690.107 77.4229C687.854 79.6191 685.095 80.7173 681.829 80.7173C678.562 80.7173 675.803 79.6191 673.55 77.4229C671.298 75.1702 670.172 72.4671 670.172 69.3135C670.172 66.1035 671.298 63.4004 673.55 61.2041C675.803 58.9515 678.562 57.8252 681.829 57.8252C685.095 57.8252 687.854 58.9515 690.107 61.2041C692.416 63.4004 693.57 66.1035 693.57 69.3135C693.57 72.4671 692.416 75.1702 690.107 77.4229ZM759.206 90.9385C770.581 90.9385 779.62 95.0776 786.321 103.356C793.023 111.634 796.374 122.925 796.374 137.229C796.374 151.477 793.051 162.74 786.406 171.019C779.761 179.297 770.807 183.436 759.544 183.436C753.124 183.436 747.464 182 742.565 179.128C737.665 176.2 733.977 172.201 731.499 167.133H731.076V211.481H710.043V92.459H730.569V107.917H730.992C733.582 102.624 737.327 98.4847 742.227 95.5C747.126 92.459 752.786 90.9385 759.206 90.9385ZM752.786 166.204C759.6 166.204 764.978 163.613 768.92 158.432C772.862 153.195 774.833 146.127 774.833 137.229C774.833 128.388 772.834 121.349 768.836 116.111C764.894 110.874 759.544 108.255 752.786 108.255C746.253 108.255 740.988 110.902 736.989 116.196C732.991 121.489 730.964 128.501 730.907 137.229C730.964 146.015 732.991 153.054 736.989 158.348C740.988 163.585 746.253 166.204 752.786 166.204ZM808.284 118.139C808.284 109.917 811.607 103.271 818.252 98.2031C824.897 93.1348 833.654 90.6006 844.523 90.6006C854.998 90.6006 863.501 93.1911 870.034 98.3721C876.623 103.497 880.058 110.198 880.339 118.477H860.742C860.404 114.591 858.771 111.55 855.842 109.354C852.914 107.157 849 106.059 844.101 106.059C839.483 106.059 835.766 107.073 832.95 109.1C830.135 111.071 828.727 113.69 828.727 116.956C828.727 121.912 833.007 125.431 841.567 127.515L857.532 131.063C866.486 133.147 872.878 136.075 876.707 139.848C880.537 143.621 882.451 148.83 882.451 155.476C882.451 163.923 878.903 170.765 871.808 176.002C864.712 181.183 855.505 183.774 844.185 183.774C833.035 183.774 824.053 181.24 817.238 176.171C810.481 171.047 806.736 164.204 806.004 155.645H826.699C827.319 159.756 829.177 162.909 832.275 165.105C835.428 167.302 839.652 168.4 844.945 168.4C850.07 168.4 854.097 167.443 857.025 165.528C860.01 163.557 861.502 160.91 861.502 157.587C861.502 154.997 860.573 152.941 858.714 151.421C856.912 149.844 853.843 148.493 849.507 147.366L833.204 143.565C816.591 139.679 808.284 131.204 808.284 118.139ZM934.486 106.904C928.63 106.904 923.73 108.931 919.788 112.986C915.902 116.984 913.734 122.137 913.284 128.444H955.098C954.873 122.081 952.873 116.9 949.1 112.901C945.327 108.903 940.456 106.904 934.486 106.904ZM955.182 155.222H974.864C973.513 163.726 969.205 170.624 961.94 175.918C954.675 181.155 945.778 183.774 935.247 183.774C921.844 183.774 911.313 179.663 903.654 171.441C895.995 163.163 892.166 151.871 892.166 137.567C892.166 123.376 895.967 112 903.569 103.44C911.228 94.8805 921.506 90.6006 934.402 90.6006C947.073 90.6006 957.125 94.6834 964.559 102.849C971.992 110.958 975.709 121.827 975.709 135.456V142.298H913.115V143.565C913.115 150.773 915.142 156.574 919.197 160.966C923.308 165.359 928.799 167.555 935.669 167.555C940.512 167.555 944.679 166.457 948.171 164.261C951.719 162.008 954.056 158.995 955.182 155.222Z" fill="#D9D9D9"/>  </g>  <defs>  <clipPath id="clip0\_1\_23">  <rect width="1000" height="250" fill="white"/>  </clipPath>  </defs> </svg> |

Листинг иконки страницы общения:

|  |
| --- |
| <svg viewBox="0 0 1000 1000" fill="none" xmlns="http://www.w3.org/2000/svg">  <g clip-path="url(#clip0\_58\_7)">  <mask id="mask0\_58\_7" style="mask-type:alpha" maskUnits="userSpaceOnUse" x="354" y="-2" width="607" height="608">  <circle cx="657.65" cy="302.298" r="303.287" transform="rotate(-30 657.65 302.298)" fill="#D9D9D9"/>  </mask>  <g mask="url(#mask0\_58\_7)">  <circle cx="446.697" cy="73.8851" r="303.287" transform="rotate(-30 446.697 73.8851)" fill="#D9D9D9"/>  </g>  <mask id="mask1\_58\_7" style="mask-type:alpha" maskUnits="userSpaceOnUse" x="32" y="430" width="930" height="570">  <ellipse cx="496.734" cy="715.395" rx="464.734" ry="284.605" fill="#D9D9D9"/>  </mask>  <g mask="url(#mask1\_58\_7)">  <circle cx="496.735" cy="998.799" r="262.989" fill="#D9D9D9"/>  <circle cx="495.534" cy="471.62" r="401.089" fill="#D9D9D9"/>  </g>  </g>  <defs>  <clipPath id="clip0\_58\_7">  <rect width="1000" height="1000" fill="white"/>  </clipPath>  </defs> </svg> |

Листинг иконки каталога

|  |
| --- |
| <svg viewBox="0 0 500 500" fill="none" xmlns="http://www.w3.org/2000/svg">  <mask id="mask0\_138\_40" style="mask-type:alpha" maskUnits="userSpaceOnUse" x="34" y="28" width="444" height="444">  <circle cx="256" cy="250" r="222" fill="#D9D9D9"/>  </mask>  <g mask="url(#mask0\_138\_40)">  <mask id="mask1\_138\_40" style="mask-type:alpha" maskUnits="userSpaceOnUse" x="152" y="67" width="196" height="394">  <ellipse cx="250.122" cy="264.191" rx="97.2414" ry="196.809" fill="#D9D9D9"/>  </mask>  <g mask="url(#mask1\_138\_40)">  <rect x="185.454" y="67.3821" width="129.336" height="393.618" fill="#D9D9D9"/>  </g>  <mask id="mask2\_138\_40" style="mask-type:alpha" maskUnits="userSpaceOnUse" x="144" y="40" width="212" height="272">  <rect x="144" y="40" width="211.356" height="271.539" fill="#D9D9D9"/>  </mask>  <g mask="url(#mask2\_138\_40)">  <path d="M249.678 -41.0054L309.666 114.73H189.69L249.678 -41.0054Z" fill="#D9D9D9"/>  </g>  <mask id="mask3\_138\_40" style="mask-type:alpha" maskUnits="userSpaceOnUse" x="26" y="40" width="122" height="244">  <ellipse cx="86.9899" cy="161.545" rx="60.0541" ry="121.545" transform="rotate(180 86.9899 161.545)" fill="#D9D9D9"/>  </mask>  <g mask="url(#mask3\_138\_40)">  <rect x="126.927" y="283.089" width="79.8748" height="243.089" transform="rotate(180 126.927 283.089)" fill="#D9D9D9"/>  </g>  <mask id="mask4\_138\_40" style="mask-type:alpha" maskUnits="userSpaceOnUse" x="22" y="132" width="131" height="168">  <rect x="152.528" y="300" width="130.528" height="167.696" transform="rotate(180 152.528 300)" fill="#D9D9D9"/>  </mask>  <g mask="url(#mask4\_138\_40)">  <path d="M87.2643 350.027L50.2172 253.848L124.311 253.848L87.2643 350.027Z" fill="#D9D9D9"/>  </g>  <mask id="mask5\_138\_40" style="mask-type:alpha" maskUnits="userSpaceOnUse" x="352" y="217" width="121" height="244">  <ellipse cx="412.538" cy="339.455" rx="60.0541" ry="121.545" fill="#D9D9D9"/>  </mask>  <g mask="url(#mask5\_138\_40)">  <rect x="372.601" y="217.911" width="79.8748" height="243.089" fill="#D9D9D9"/>  </g>  <mask id="mask6\_138\_40" style="mask-type:alpha" maskUnits="userSpaceOnUse" x="347" y="201" width="131" height="168">  <rect x="347" y="201" width="130.528" height="167.696" fill="#D9D9D9"/>  </mask>  <g mask="url(#mask6\_138\_40)">  <path d="M412.264 150.973L449.311 247.152H375.217L412.264 150.973Z" fill="#D9D9D9"/>  </g>  </g> </svg> |

Листинг бургер-иконки:

|  |
| --- |
| <svg viewBox="0 0 500 500" fill="none" xmlns="http://www.w3.org/2000/svg">  <g clip-path="url(#clip0\_138\_20)">  <mask id="mask0\_138\_20" style="mask-type:alpha" maskUnits="userSpaceOnUse" x="0" y="0" width="500" height="500">  <circle cx="250" cy="250" r="250" fill="#D9D9D9"/>  </mask>  <g mask="url(#mask0\_138\_20)">  <circle cx="498.503" cy="125.749" r="250" fill="#D9D9D9"/>  <path d="M367.365 552.306C377.711 513.697 376.375 472.888 363.526 435.038C350.678 397.189 326.895 363.999 295.184 339.667C263.473 315.334 225.259 300.951 185.374 298.337C145.488 295.723 105.724 304.995 71.108 324.98C36.4922 344.965 8.58026 374.767 -9.09836 410.615C-26.777 446.464 -33.4282 486.75 -28.211 526.379C-22.9937 566.008 -6.14236 603.199 20.2122 633.251C46.5668 663.303 81.2408 684.864 119.85 695.21L172.156 500L367.365 552.306Z" fill="#D9D9D9"/>  <path d="M375.472 -101.078C362.571 -123.424 344.702 -142.502 323.248 -156.837C301.794 -171.173 277.329 -180.38 251.747 -183.748C226.166 -187.116 200.152 -184.554 175.719 -176.26C151.285 -167.966 129.088 -154.163 110.842 -135.917C92.5971 -117.672 78.7936 -95.4743 70.4997 -71.041C62.2057 -46.6078 59.6436 -20.594 63.0115 4.98786C66.3794 30.5697 75.5871 55.0339 89.9223 76.488C104.257 97.9421 123.336 115.811 145.682 128.712L229.791 -16.9691L375.472 -101.078Z" fill="#D9D9D9"/>  <path d="M110.842 -66.239C129.088 -47.9938 151.285 -34.1903 175.719 -25.8963C200.152 -17.6024 226.166 -15.0402 251.748 -18.4082C277.329 -21.7761 301.794 -30.9838 323.248 -45.3189C344.702 -59.6541 362.571 -78.7325 375.472 -101.078C388.373 -123.424 395.961 -148.438 397.649 -174.185C399.336 -199.933 395.078 -225.723 385.204 -249.562C375.33 -273.4 360.104 -294.648 340.705 -311.661C321.305 -328.673 298.252 -340.995 273.329 -347.674L229.791 -185.187L110.842 -66.239Z" fill="#D9D9D9"/>  </g>  </g>  <defs>  <clipPath id="clip0\_138\_20">  <rect width="500" height="500" fill="white"/>  </clipPath>  </defs> </svg> |

Листинг стрелочки вправо:

|  |
| --- |
| <svg viewBox="0 0 73 83" fill="none" xmlns="http://www.w3.org/2000/svg">  <g clip-path="url(#clip0\_176\_14)">  <path d="M71 37.5359C73.6667 39.0755 73.6667 42.9245 71 44.4641L7.99999 80.8372C5.33333 82.3768 2 80.4523 2 77.3731L2 4.62693C2 1.54773 5.33333 -0.376772 8 1.16283L71 37.5359Z" fill="#D9D9D9"/>  </g>  <defs>  <clipPath id="clip0\_176\_14">  <rect width="73" height="83" fill="white"/>  </clipPath>  </defs> </svg> |

Листинг стрелочки влево:

|  |
| --- |
| <svg viewBox="0 0 73 83" fill="none" xmlns="http://www.w3.org/2000/svg">  <g clip-path="url(#clip0\_173\_2)">  <path d="M3 45.4641C0.333333 43.9245 0.333334 40.0755 3 38.5359L66 2.16283C68.6667 0.623231 72 2.54773 72 5.62693L72 78.3731C72 81.4523 68.6667 83.3768 66 81.8372L3 45.4641Z" fill="#D9D9D9"/>  </g>  <defs>  <clipPath id="clip0\_173\_2">  <rect width="73" height="83" fill="white"/>  </clipPath>  </defs> </svg> |

ПРИЛОЖЕНИЕ Ж

**Листинг JavaScript**

Листинг Objects:

|  |
| --- |
| const *vh* = *visualViewport*.height \* 0.01 const *vw* = *visualViewport*.width \* 0.01  const *size* = 1.4\**vw* + 0.5\**vh*;  const *defIconWidth* = 6\**size*; const *defIconHeight* = 3\**size*;  const *dark* = "black" const *light* = "#d9d9d9" const *accent* = "#74d8f1"  const *body* = *document*.querySelector("body")   function Line(dark, noPad) {  const line = *document*.createElement("div")  line.style.height = "2px"  line.style.width = "75%"  line.style.margin = "0 auto"  if (!noPad) {  line.style.marginTop = "2%"  line.style.marginBottom = "2%"  }  line.style.left = "0"  line.style.right = "0"  if (dark) line.style.backgroundColor = "#d9d9d9"  else line.style.backgroundColor = "black"  return line }  function Icon(svgFilePath, width, height, interactive, square) {  let svgElement = *document*.createElementNS("http://www.w3.org/2000/svg", "svg");   svgElement.classList.add("icon");  if (interactive) {  svgElement.classList.add("icon-interactive")  }  /\* if (width) svgElement.setAttribute("width", width)  else svgElement.setAttribute("width", "auto")   if (height) svgElement.setAttribute("height", height)  else svgElement.setAttribute("height", "auto")\*/  let xhr = new XMLHttpRequest();  xhr.open("GET", svgFilePath, true);  xhr.onreadystatechange = function () {  if (xhr.readyState === 4 && xhr.status === 200) {  // Inject the SVG data into the SVG element  svgElement.innerHTML = xhr.responseText;  }  };  xhr.send();  svgElement.innerHTML  if(square) svgElement.classList.add("wide")  // Return the SVG element  return svgElement; }  function IconObj(svgFilePath, interactive, square) {  const icon = *document*.createElement("div")  const iconWidth = square ? *defIconHeight* : *defIconWidth* */\* icon.style.width = `${iconWidth}`*  *icon.style.height = `${defIconHeight}`\*/* icon.appendChild(Icon(svgFilePath, iconWidth, *defIconHeight*, interactive, square))  return icon; }  function IconLink(svgFilePath, link, interactive, square) {  const linkObj = *document*.createElement("a")  if(link){  linkObj.href = link;  }  const iconWidth = square ? *defIconHeight* : *defIconWidth*;  linkObj.appendChild(Icon(svgFilePath, iconWidth, *defIconHeight*, interactive, square))  return linkObj }  function defText(centered, dark, text) {  const txt = *document*.createElement("h1")  txt.innerText = text  txt.classList.add("def-text")  return txt }  function textLink(centered, dark, text, href) {  const link = *document*.createElement("a")  link.appendChild(defText(centered, dark, text))  link.href = href;  return link; }  function MidText(centered, dark, text) {  const txt = *document*.createElement("h1")  txt.innerText = text  txt.classList.add("mid-text")  return txt }  function Label(dark, text) {  const labelObj = *document*.createElement("div")  if (dark) labelObj.classList.add("dark")  else labelObj.classList.add("light")  labelObj.classList.add("row")  labelObj.classList.add("centered")   const txt = MidText(true, true, text)   labelObj.appendChild(txt)   return labelObj }  function Header(dark) {  const headLabel = *document*.createElement("header")  if (dark) headLabel.classList.add("dark")  else headLabel.classList.add("light")  headLabel.classList.add("row")  headLabel.classList.add("flex-center")   return headLabel; }  function Footer(logoElement, Lists, disclaimer) {  const footer = *document*.createElement("footer")  footer.classList.add("dark")  footer.classList.add("column")    const footLabel = *document*.createElement("div") // append to footer  footLabel.classList.add("dark")  footLabel.classList.add("row")  footLabel.classList.add("flex-center")  footLabel.classList.add("foot-label")  footLabel.style.borderBottom = "2px solid #d9d9d9"   const footContainer = *document*.createElement("div")  footContainer.classList.add("dark")  footContainer.classList.add("row")  footContainer.classList.add("flex-center")  footContainer.classList.add("foot-container")  footContainer.style.padding = "2%"  for (let i = 0; i < Lists.length; i++) {  footContainer.appendChild(Lists[i])  }   const footDisclaimerBlock = *document*.createElement("div")  const footDisclaimer = *document*.createElement("p")  if (disclaimer) {  footDisclaimer.innerText = disclaimer  }  footDisclaimerBlock.classList.add("footer-disclaimer")   //appends   footLabel.appendChild(logoElement)   footer.appendChild(footLabel)  footer.appendChild(footContainer)   if (disclaimer) {  footer.appendChild(Line(true, true))  footDisclaimerBlock.appendChild(footDisclaimer)  footer.appendChild(footDisclaimerBlock)  }   return footer; }  function FooterList(header, items) {  const list = *document*.createElement("ul")  list.classList.add("footer-list")   const headItem = *document*.createElement("li")  headItem.classList.add("footer-list-head")  headItem.innerText = header   list.appendChild(headItem)   for (let i = 0; i < items.length; i++) {  let item = *document*.createElement("li")  item.classList.add("footer-list-item")  item.appendChild(items[i])  list.appendChild(item)  }  return list; }  function LinkText(link, text, lght) {  const href = *document*.createElement("a")  if(link) href.href = link;  if(lght) href.style.color = *dark*;  href.innerText = text;  return href; }  function Banner(link, source, info, centered, dark, transparentBack) {  const linkObject = *document*.createElement("a")  if(link) linkObject.href = link;   const image = *document*.createElement("div")  image.classList.add("banner-image")  image.style.backgroundImage = `url(${source})`   const description = *document*.createElement("div")  if (centered) description.classList.add("banner-description-center")  else description.classList.add("banner-description")    const text = *document*.createElement("h1")  text.innerText = info  if (!dark) {  text.style.color = "black"  }   description.appendChild(text)  image.appendChild(description)  linkObject.appendChild(image)  return linkObject; }  function InformationBanner(source, info, info2, dark) {  const containerHref = *document*.createElement("a")  const image = *document*.createElement("div")  image.classList.add("banner-image")  image.style.backgroundImage = `url(${source})`   const description = *document*.createElement("div")  description.classList.add("banner-full-description")   const text = *document*.createElement("h1")  text.innerHTML = info  if (dark) {  text.style.color = "#d9d9d9"  } else text.style.color = "black"   const text2 = *document*.createElement("p")  text2.innerHTML = info2  if (dark) {  text2.style.color = "#d9d9d9"  } else text2.style.color = "black"    description.appendChild(text)  description.appendChild(Line(dark))  description.appendChild(text2)   image.appendChild(description)  containerHref.appendChild(image)  return containerHref; }  function Timer(f, t) {  let timer = setInterval(f, t)  this.stop = function () {  if (timer) {  clearInterval(timer)  timer = null  }  }  this.start = function () {  if (!timer) {  this.stop()  timer = setInterval(f, t)  }  }  this.reset = function (newT = t) {  t = newT  this.stop()  this.start()  } }    function DropdownMenu(pairList, dark) {  this.menu = *document*.createElement("div")  this.menu.classList.add("row")  this.menu.classList.add("dropdown-set")  if(dark) this.menu.classList.add("dark")  else this.menu.classList.add("light")  this.menu.classList.add("desktop")  const headerContainer = *document*.createElement("div")  const InformationContainer = *document*.createElement("div")   let current = 0;   this.DropDown = function() {  this.menu.classList.add("dropdown-shown")  }   this.DropOut = function() {  this.menu.classList.remove("dropdown-shown")  }  let informationPairs = []   this.SwitchTo = function(i) {  informationPairs[i][0].classList.add("dropdown-focused")  informationPairs[i][1].classList.add("dropdown-focused")  }   this.SwitchOut = function(i) {  informationPairs[i][0].classList.remove("dropdown-focused")  informationPairs[i][1].classList.remove("dropdown-focused")  }   for(let i = 0; i < pairList.length; i++) {  headerContainer.appendChild(pairList[i][0])   pairList[i][0].addEventListener("mouseover", () => {  this.SwitchOut(current)  current = i;  this.SwitchTo(current);  })   InformationContainer.appendChild(pairList[i][1])    informationPairs.push(pairList[i])  }   this.menu.appendChild(headerContainer)  this.menu.appendChild(InformationContainer)  headerContainer.classList.add("flex")   headerContainer.classList.add("column")  InformationContainer.classList.add("column")   this.hovered = false    this.menu.addEventListener("mouseleave", () => {  this.DropOut();  })    this.SwitchTo(current) }  function DropdownList(linkList, dark) {  this.menu = *document*.createElement("div")  this.menu.classList.add("row")  this.menu.classList.add("dropdown-set")  if(dark) this.menu.classList.add("dark")  else this.menu.classList.add("light")  this.menu.classList.add("desktop")  const headerContainer = *document*.createElement("div")    this.DropDown = function() {  this.menu.classList.add("dropdown-shown")  }   this.DropOut = function() {  this.menu.classList.remove("dropdown-shown")  }    for(let i = 0; i < linkList.length; i++) {  linkList[i].classList.add("centered")  headerContainer.appendChild(linkList[i])  }   this.menu.appendChild(headerContainer)  headerContainer.classList.add("flex")   headerContainer.classList.add("column")  this.hovered = false    this.menu.addEventListener("mouseleave", () => {  this.DropOut();  }) }  function Notification(icon, message, dark) {  this.msg = *document*.createElement("div")  this.msg.classList.add("row")  this.msg.classList.add("dropdown-set")  this.msg.classList.add("notification")  if(dark) this.msg.classList.add("dark")  else this.msg.classList.add("light")  this.text = *document*.createElement("p")  this.text.innerText = message;  if(icon) {  this.msg.appendChild(IconObj(icon, false, true))  }  this.msg.appendChild(this.text)  this.hidden = true;   let notDelay = new Timer(() => {  this.Hide()  notDelay.stop()  }, 5000)    this.Hide = function() {  this.hidden = true;  this.msg.classList.remove("dropdown-shown")  }  this.Show = function() {  this.hidden = false;  notDelay.start()  this.msg.classList.add("dropdown-shown")  }  this.msg.addEventListener("mouseenter", () => {  notDelay.stop();  })  this.msg.addEventListener("mouseleave", () => {  this.Hide()  }) }  function Slider(banners, scrollInteractive) {  const count = banners.length  const slider = *document*.createElement("div")  slider.classList.add("slider-view")  slider.classList.add("non-interactive")  const left = IconObj(*iconPath* + "left.svg", true)  const right = IconObj(*iconPath* + "right.svg", true)   left.className = "prev"  right.className = "next"   slider.append(right)  slider.append(left)  const pointerContainer = *document*.createElement("div")  pointerContainer.classList.add("slide-pointer-container")  pointerContainer.classList.add("row")  pointerContainer.classList.add("centered")  slider.append(pointerContainer)   let pointers = []  let pointedSlide = 0;  for (let i = 0; i < banners.length; i++) {  let pointer = *document*.createElement("div")  pointer.classList.add("slide-pointer")  pointerContainer.append(pointer)  pointers.push(pointer)  pointer.addEventListener('mousedown', () => {  this.SlideTo(i);  })  if(scrollInteractive) banners[i].classList.add("scroll-hidden")  slider.append(banners[i])  }     this.Point = function (pointer) {  pointer.classList.add("highlighted")  }  this.PointOut = function (pointer) {  pointer.classList.remove("highlighted")  }   this.ApplyTranslation = function () {  for (let i of slider.children) {  if (i !== right && i !== left && i !== pointerContainer) {  i.style.transform = `translateX(${translation}%)`  }  }  }   let translation = 0;  this.SlideLeft = function () {  if (translation < 0) {  this.PointOut(pointers[pointedSlide])  pointedSlide--;  this.Point(pointers[pointedSlide])  translation += 100  this.ApplyTranslation()  } else this.ToEnd()  }  this.SlideRight = function () {  if (translation > -1 \* (count - 1) \* 100) {  this.PointOut(pointers[pointedSlide])  pointedSlide++;  this.Point(pointers[pointedSlide])  translation -= 100  this.ApplyTranslation()  } else this.GetBack()  }   this.SlideTo = function (index) {  translation = -1\*index\*100;  this.PointOut(pointers[pointedSlide])  pointedSlide = index;  this.Point(pointers[pointedSlide])  this.ApplyTranslation()  }   this.GetBack = function () {  this.PointOut(pointers[pointedSlide])  pointedSlide = 0;  this.Point(pointers[pointedSlide])  translation = 0;  this.ApplyTranslation()  }   this.ToEnd = function () {  this.PointOut(pointers[pointedSlide])  pointedSlide = count-1;  this.Point(pointers[pointedSlide])  translation = -1 \* (count - 1) \* 100;  this.ApplyTranslation()  }   let SlideTimer = new Timer(() => {  this.SlideRight()  }, 15000)   left.addEventListener('mousedown', () => {  this.SlideLeft()  SlideTimer.reset()  })  right.addEventListener('mousedown', () => {  this.SlideRight()  SlideTimer.reset()  })   this.Point(pointers[pointedSlide]);  SlideTimer.start()   return slider; }  function Item(imgsource, caption, link) {  const href = *document*.createElement("a")  if(link) href.href = link;   const container = *document*.createElement("div")  container.classList.add("slide-item")   const item = *document*.createElement("figure")    const img = *document*.createElement("img")  if(imgsource) img.src = imgsource  const figcaption = *document*.createElement("figcaption")  if(caption) figcaption.innerText = caption   container.appendChild(item)  href.appendChild(container)  item.appendChild(img)  item.appendChild(figcaption)   return href }  function ItemSlider(ItemsList, dark) {  this.slider = *document*.createElement("div")  this.slider.classList.add("item-slider-view")  /\*this.slider.classList.add(dark ? "dark" : "light")\*/  const slidecounter = ItemsList.length;  this.sliderList = []  for(let i = 0; i < slidecounter; i++) {  this.sliderList.push(ItemsList[i])  this.slider.appendChild(ItemsList[i])  /\*ItemsList[i].style.transform = `translateX(${i\*50}%)`\*/  }   this.prev = IconObj("left.svg", true)  this.next = IconObj("right.svg", true)   this.prev.className = "prev"  this.next.className = "next"   this.slider.appendChild(this.prev)  this.slider.appendChild(this.next)   let translation = 0;  let current = 5  current = parseInt(`${slidecounter/2}`)   this.ApplyTranslation = function() {  for (let i = 0; i < slidecounter; i++) {  this.sliderList[i].style.transform = `translateX(${translation}%)`  }  }   this.Delight = function() {  this.sliderList[current].classList.remove("highlighted")  }   this.Highlight = function() {  this.sliderList[current].classList.add("highlighted")  }   this.SlideLeft = function() {  *console*.log(100\*(slidecounter/2))  if(translation < 100\*slidecounter/2 - 50){  translation += 100;  this.Delight()  current -= 1;  this.Highlight()  for (let i = 0; i < slidecounter; i++) {  this.sliderList[i].style.transform = `translateX(${translation}%)`  }  this.ApplyTranslation()  }  }   this.SlideRight = function() {  if(translation > -1\*100\*slidecounter/2 + 50){  translation -= 100;  this.Delight()  current+=1;  this.Highlight()   for (let i = 0; i < slidecounter; i++) {  this.sliderList[i].style.transform = `translateX(${translation}%)`  }  this.ApplyTranslation()  }  }    this.SlideTo = function(n) {  translation = -1\*100\*n;  for (let i = 0; i < slidecounter; i++) {  this.sliderList[i].style.transform = `translateX(${translation}%)`  }  current = n;  this.ApplyTranslation()  }   this.prev.addEventListener("mousedown", () => {  this.SlideLeft()  })   this.next.addEventListener("mousedown", () => {  this.SlideRight()  })   /\*this.SlideTo(0)\*/ }  function GridItem(text, img, link) {  const item = *document*.createElement("div")  item.classList.add("grid-item")   const itemHref = *document*.createElement("a")  const itemImg = *document*.createElement("img")  const itemText = *document*.createElement("h1")  if(link) itemHref.href = link  if(img) itemImg.src = img  if(text) itemText.innerText = text   itemHref.appendChild(item)  item.appendChild(itemImg)  item.appendChild(itemText)  return itemHref } function ItemGrid(dark, itemlist) {  this.container = *document*.createElement("div")  this.container.classList.add("items-grid")  for(let i = 0; i < itemlist.length; i++) {  this.container.appendChild(itemlist[i])  } }  function loadXMLDoc(source, func) {  var xmlhttp = new XMLHttpRequest();  xmlhttp.onreadystatechange = function() {  if (this.readyState === 4 && this.status === 200) {  func(this);  }  };  xmlhttp.open("GET", source, true);  xmlhttp.send(); } |

Листинг Sources:

|  |
| --- |
| //folders const *mobileWidth* = 420 const *iconPath* = "img/icons/" const *vapePath* = "img/Vapes/" const *ellipsePath* = "img/ellipse/"  //icons  const *ellipseIcon* = *iconPath* + "ellipse.svg" const *clubIcon* = *iconPath* + "club.svg" const *goodsIcon* = *iconPath* + "goods.svg" const *leftIcon* = *iconPath* + "left.svg" const *rightIcon* = *iconPath* + "right.svg" const *MoreIcon* = *iconPath* + "more.svg"  //photos  const *NotFound* = "notfound\_21\_9.jpg"  //ready-to-use solution functions function AddDefaultMobileDropdown(dark, back) {  //dropdown mobile  const closeNavigationDropdown = (LinkText("", "Закрыть", !dark))  const backLink = LinkText("", "Назад", !dark)  const navigationDropdown =  back ? new DropdownList  (  [  LinkText("store.html", "Ellipstore", !dark),  LinkText("club.html", "Ellipse Club", !dark),  LinkText("catalog.html", "Каталог", !dark),  LinkText("support.html", "Поддержка", !dark),  backLink,  closeNavigationDropdown  ],  dark  ) : new DropdownList  (  [  LinkText("store.html", "Ellipstore", !dark),  LinkText("club.html", "Ellipse Club", !dark),  LinkText("catalog.html", "Каталог", !dark),  LinkText("support.html", "Поддержка", !dark),  closeNavigationDropdown  ],  dark  )   closeNavigationDropdown.addEventListener("mousedown", () => {  navigationDropdown.DropOut()  })  if(back) backLink.addEventListener("mousedown", () => {  *history*.back()  })  *body*.appendChild(navigationDropdown.menu)  navigationDropdown.menu.classList.add("mobile")  *More*.addEventListener("mousedown", () => {  navigationDropdown.DropDown()  }) //   *body*.appendChild(*header*) } |

Листинг animationHandler:

|  |
| --- |
| const *allSideObserver* = new IntersectionObserver((entries) => {  entries.forEach((entry) => {  if(entry.isIntersecting) {  entry.target.classList.add("scroll-shown")  }  else entry.target.classList.remove("scroll-shown")  }) })  let *observed* = false   const *oneSideObserver* = new IntersectionObserver((entries) => {  entries.forEach((entry) => {  if(entry.isIntersecting) {   entry.target.classList.add(".scroll-shown")  }  else entry.target.classList.remove(".scroll-shown")  }) }) const *scrollHiddenElems* = *document*.querySelectorAll(".scroll-hidden") if(*window*.innerWidth >= *mobileWidth*) {  *window*.addEventListener("load", () => {   *scrollHiddenElems*.forEach((el) => *allSideObserver*.observe(el))  }) } else {  *scrollHiddenElems*.forEach((el) => el.classList.remove("scroll-hidden")) } |

Листинг Notifications:

|  |
| --- |
| const *InvalidDataDark* = new Notification(false,"Неверный ввод", true) const *InvalidDataLight* = new Notification(false,"Неверный ввод", false)  let *ClubInviter* = new Notification(false,"Ellipse Club - Платформа для пользователей устройствами Ellipse", true)  const *FormSubmitDark* = new Notification(false,"Ваша заявка отправлена", true) const *FormSubmitLight* = new Notification(false,"Ваша заявка отправлена", false)  const *NotFoundDark* = new Notification(false, "Товаров не найдено", true) const *NotFoundLight* = new Notification(false, "Товаров не найдено", false)  const *AdultAlertDark* = new Notification(false, "Внимание: если вам меньше 18 лет, покиньте этот сайт", true) |

Листинг IndexConstructor:

|  |
| --- |
| //notifications body.appendChild(*ClubInviter*.msg) body.appendChild(*AdultAlertDark*.msg) //header const *header* = *document*.querySelector("header")  const *Goods* = *document*.querySelector("a[id='catalogIcon']") const *Club* = *document*.querySelector("a[id='clubIcon']") const *More* = *document*.querySelector("a[id='mobileIcon']")  Club.addEventListener("mouseenter", () => {  *ClubInviter*.Show() })  //dropdown  const *tmtBanner* = Banner("", *vapePath* + "TMT/46/Transparent/fstransparent.png", "TMT 46", false, true) const *tmtPair* = [LinkText("catalogitem.html?item=tmtfs", "TMT"), *tmtBanner*]  const *leavesBanner* = Banner("", *vapePath* + "Leaves/Transparent/LeavesVapePresent21\_9.png", "Leaves", false, true) const *leavesPair* = [LinkText("catalogitem.html?item=leaves", "Leaves"), *leavesBanner*]  const *surfingemBanner* = Banner("", *vapePath* + "Leaves/Transparent/LeavesVapePresent21\_9.png", "SurfinGem Wise", false, true) const *surfingemPair* = [LinkText("catalogitem.html?item=wise", "SurfinGem"), *surfingemBanner*]  const *kaliBanner* = Banner("", *vapePath* + "Leaves/Transparent/LeavesVapePresent21\_9.png", "Byteshard A", false, true) const *kaliPair* = [LinkText("catalogitem.html?item=bytesharda", "Byteshard"), *kaliBanner*]   const *goodsDropdown* = new DropdownMenu([*tmtPair*, *leavesPair*, *surfingemPair*, *kaliPair*], true) Goods.addEventListener("mouseenter", () => {  const height = *header*.getBoundingClientRect().top  if(height === 0){  *goodsDropdown*.DropDown()  } }) goodsDropdown.menu.classList.add("desktop") body.appendChild(*goodsDropdown*.menu)    //dropdown burger AddDefaultMobileDropdown(true)  //slider  const *slideBanner1* = Banner("catalogitem.html?item=tmtfs", "img/Vapes/TMT/46/Present/fspresent.jpg",  "TMT 46. Релиз в июне.", false, true) const *slideBanner2* = Banner("catalogitem.html?item=leaves", *vapePath* + "Leaves/Present/1\_21\_9.jpg",  "Leaves. Лес ваших фантазий.", false, true) const *slideBanner3* = InformationBanner(*ellipsePath* + "weDo\_21\_9.jpg", "Что мы производим?" ,  "Мы производим вейп-устройства, которые дарят удовольствие каждому желающему. " +  "Наши компоненты производятся в Тайване, и благодаря этому устройства будут служить вам большой срок. " +  "Всё, что мы делаем - исключительно ради вас.", true)   const *tempSl* = *document*.querySelector(".slider-view") const *slider* = new Slider([*slideBanner1*, *slideBanner2*, *slideBanner3*], true) tempSl.replaceWith(*slider*)  body.prepend(*header*) AdultAlertDark.Show() |

Листинг catalogConstructor:

|  |
| --- |
| //notifications body.appendChild(*ClubInviter*.msg) body.appendChild(*AdultAlertDark*.msg) //header const *header* = *document*.querySelector("header")  const *Goods* = *document*.querySelector("a[id='catalogIcon']") const *Club* = *document*.querySelector("a[id='clubIcon']") const *More* = *document*.querySelector("a[id='mobileIcon']")  Club.addEventListener("mouseenter", () => {  *ClubInviter*.Show() })  //dropdown  const *tmtBanner* = Banner("", *vapePath* + "TMT/46/Transparent/fstransparent.png", "TMT 46", false, true) const *tmtPair* = [LinkText("catalogitem.html?item=tmtfs", "TMT"), *tmtBanner*]  const *leavesBanner* = Banner("", *vapePath* + "Leaves/Transparent/LeavesVapePresent21\_9.png", "Leaves", false, true) const *leavesPair* = [LinkText("catalogitem.html?item=leaves", "Leaves"), *leavesBanner*]  const *surfingemBanner* = Banner("", *vapePath* + "Leaves/Transparent/LeavesVapePresent21\_9.png", "SurfinGem Wise", false, true) const *surfingemPair* = [LinkText("catalogitem.html?item=wise", "SurfinGem"), *surfingemBanner*]  const *kaliBanner* = Banner("", *vapePath* + "Leaves/Transparent/LeavesVapePresent21\_9.png", "Byteshard A", false, true) const *kaliPair* = [LinkText("catalogitem.html?item=bytesharda", "Byteshard"), *kaliBanner*]   const *goodsDropdown* = new DropdownMenu([*tmtPair*, *leavesPair*, *surfingemPair*, *kaliPair*], true) Goods.addEventListener("mouseenter", () => {  const height = *header*.getBoundingClientRect().top  if(height === 0){  *goodsDropdown*.DropDown()  } }) goodsDropdown.menu.classList.add("desktop") body.appendChild(*goodsDropdown*.menu)    //dropdown burger AddDefaultMobileDropdown(true)  //slider  const *slideBanner1* = Banner("catalogitem.html?item=tmtfs", "img/Vapes/TMT/46/Present/fspresent.jpg",  "TMT 46. Релиз в июне.", false, true) const *slideBanner2* = Banner("catalogitem.html?item=leaves", *vapePath* + "Leaves/Present/1\_21\_9.jpg",  "Leaves. Лес ваших фантазий.", false, true) const *slideBanner3* = InformationBanner(*ellipsePath* + "weDo\_21\_9.jpg", "Что мы производим?" ,  "Мы производим вейп-устройства, которые дарят удовольствие каждому желающему. " +  "Наши компоненты производятся в Тайване, и благодаря этому устройства будут служить вам большой срок. " +  "Всё, что мы делаем - исключительно ради вас.", true)   const *tempSl* = *document*.querySelector(".slider-view") const *slider* = new Slider([*slideBanner1*, *slideBanner2*, *slideBanner3*], true) tempSl.replaceWith(*slider*)  body.prepend(*header*) AdultAlertDark.Show() |

Листинг itemConstructor:

|  |
| --- |
| //notifications body.appendChild(*ClubInviter*.msg)  //header const *header* = *document*.querySelector("header")  const *Club* = *document*.querySelector("a[id='clubIcon']") const *More* = *document*.querySelector("a[id='mobileIcon']")  Club.addEventListener("mouseenter", () => {  *ClubInviter*.Show() })  //item elements const *main* = *document*.querySelector("main") const *deviceParam* = new URLSearchParams(*window*.location.search).get('item') if(!*deviceParam*) {  *window*.location.replace("notFound.html") } const *devicePresent* = *document*.querySelector("div[id='devicePresent']") const *deviceDescription* = *document*.querySelector("div[id='deviceDescription']") const *deviceSets* = *document*.querySelector("div[id='deviceSets']") const *deviceStore* = *document*.querySelector("a[id='deviceStore']") function setDevice(xml) {   let xmlDoc = xml.responseXML;  const element = xmlDoc.querySelector(`item[name=${*deviceParam*}]`)  if(element) {  const capsContainer = *document*.querySelector(".caps")  *document*.title = `${element.children[0].innerHTML} - Ellipse`  *devicePresent*.querySelector("h1").innerText = element.children[0].innerHTML  *devicePresent*.style.backgroundImage = `url(${element.children[1].innerHTML})`  *deviceDescription*.querySelector("h1").innerText = element.children[3].innerHTML  *deviceDescription*.querySelector("p").innerText = element.children[4].innerHTML  *deviceStore*.href = `storeitem.html?item=${*deviceParam*}`  /\*deviceStore.querySelector(".banner-image").style.backgroundImage = `url(${element.children[6].innerHTML})`\*/  *deviceSets*.querySelector("h1").innerText = "Характеристики"  *deviceSets*.querySelector("p").innerText = element.children[5].innerHTML  const caps = element.querySelectorAll("capability")  for(let i of caps) {  let b = Banner(false, i.querySelector("capoimage").innerHTML, i.querySelector("capocaption").innerHTML, true, true)  if(*window*.innerWidth > *mobileWidth*) {  b.classList.add("scroll-hidden")  *allSideObserver*.observe(b)  }   b.classList.add("back-transp")  *main*.insertBefore(b, capsContainer)  }  }  else {  *window*.location.replace("notFound.html")  } } loadXMLDoc("items.xml", setDevice) //dropdown burger AddDefaultMobileDropdown(true, true)  //slider  body.prepend(*header*) |

Листинг storeConstructor:

|  |
| --- |
| //notifications body.appendChild(*NotFoundLight*.msg)  //header const *header* = *document*.querySelector("header")  const *More* = *document*.querySelector("a[id='mobileIcon']")     //dropdown  //dropdown burger AddDefaultMobileDropdown(false, true)  //slider  //Grid operations  const *gridItems* = *document*.querySelectorAll("a .grid-item"); const *textSearch* = *document*.querySelector("input[id='Search']")  for(let i of *gridItems*) {  i.classList.add("search-hidden")  i.classList.add("search-shown") } function hideGridElements() {  for(let i of *gridItems*) {  i.classList.remove("search-shown")  } }  function showNecessaryGridElements(search) {  if(!search) {  for(let i of *gridItems*) i.classList.add("search-shown")   }  else {  let f = false;  for(let i of *gridItems*) {  if(i.querySelector("h1").innerText.includes(search)) {  i.classList.add("search-shown")  f = true  }  }  if(!f) *NotFoundDark*.Show()  } }  textSearch.addEventListener("change", () => {  hideGridElements()  showNecessaryGridElements(*textSearch*.value) })     body.prepend(*header*) handleResize() |

Листинг storeItemConstructor:

|  |
| --- |
| //notifications body.appendChild(*ClubInviter*.msg) body.appendChild(*InvalidDataLight*.msg) body.appendChild(*FormSubmitLight*.msg) //header const *header* = *document*.querySelector("header")   const *More* = *document*.querySelector("a[id='mobileIcon']")   //item elements  function CustomerForm() {  this.form = *document*.createElement("form")  this.form.classList.add("flex-center")  this.form.id = "customer";  const pls = *document*.createElement("h1")  const pls2 = *document*.createElement("p")  pls.innerText = "Заполните, пожалуйста, форму."  pls2.innerText = "Позже мы свяжемся с вами для уточнения деталей заказа"   this.nameLabel = *document*.createElement("label")  this.name = *document*.createElement("input")  this.name.type = "text"  this.name.classList.add("search-field")  this.name.name = "name"  this.name.placeholder = "ФИО"  this.nameLabel.appendChild(this.name)   this.phoneLabel = *document*.createElement("label")  this.phone = *document*.createElement("input")  this.phone.type = "text"  this.phone.classList.add("search-field")  this.phone.name = "phone"  this.phone.placeholder = "+375 33 111 11 11"  this.phoneLabel.appendChild(this.phone)   this.townLabel = *document*.createElement("label")  this.town = *document*.createElement("input")  this.town.type = "text"  this.town.classList.add("search-field")  this.town.name = "town"  this.town.placeholder = "г. Минск"  this.townLabel.appendChild(this.town)   this.button = *document*.createElement("button")  this.button.classList.add("fixed-sized")  this.button.formMethod = ""  this.button.innerText = "Связаться"  this.button.type = "button"  this.form.appendChild(pls)  this.form.appendChild(pls2)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.nameLabel)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.phoneLabel)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.townLabel)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.button)   this.realButton = this.form.querySelector("button")  this.realButton.addEventListener("mousedown", () => {   let c = 0;  let complete = true;   for(let i of this.name.value) {  if(i === ' ') c++  }  if (c < 1) complete = false;  for(let i in this.phone.value) {  if(!(i === '+' || (i >= '0' && i <= '9') || i === '(' || i === ')' || i === ' ')) {  complete = false;  }  }   if(complete && this.town.value.length > 0) {  *FormSubmitLight*.Show()  this.name.value = ""  this.phone.value = ""  this.town.value = ""  this.form.remove()  }  else *InvalidDataLight*.Show()  }) }  const *deviceParam* = new URLSearchParams(*window*.location.search).get('item') if(!*deviceParam*) {  *window*.location.replace("notFound.html") } const *devicePresent* = *document*.querySelector("div[id='devicePresent']") const *deviceDescription* = *document*.querySelector("div[id='deviceDescription']") const *deviceSets* = *document*.querySelector("div[id='deviceSets']") const *deviceCost* = *document*.querySelector("h1[id='deviceCost']") const *deviceForm* = *document*.querySelector("button[id='deviceForm']") const *deviceCatalog* = *document*.querySelector("button[id='deviceCatalog']") function setDevice(xml) {   let xmlDoc = xml.responseXML;  const element = xmlDoc.querySelector(`item[name=${*deviceParam*}]`)  if(element) {  *document*.title = `${element.children[0].innerHTML} - Ellipstore`  *devicePresent*.querySelector("h1").innerText = element.children[0].innerHTML  *devicePresent*.style.backgroundImage = `url(${element.children[1].innerHTML})`  *deviceDescription*.querySelector("h1").innerText = element.children[3].innerHTML  *deviceDescription*.style.backgroundImage = `url(${element.children[6].innerHTML})`  *deviceSets*.querySelector("h1").innerText = "Характеристики"  *deviceSets*.querySelector("p").innerText = element.children[5].innerHTML  *deviceCost*.innerHTML = `${element.querySelector("cost").innerHTML} BYN`   const formm = new CustomerForm()  *deviceForm*.addEventListener("mousedown", () => {  *body*.append(formm.form)  *window*.scrollTo({top: 10000, behavior: "smooth"})  *deviceForm*.remove()  })  *deviceCatalog*.addEventListener("mousedown", () => {  *window*.location.href = `catalogitem.html?item=${*deviceParam*}`  })  }  else {  *window*.location.replace("notFound.html")  } } loadXMLDoc("items.xml", setDevice)  //dropdown burger AddDefaultMobileDropdown(false, true)  //slider  body.prepend(*header*) |

Листинг clubConstructor

|  |
| --- |
| //notifications body.appendChild(*FormSubmitDark*.msg) body.appendChild(*InvalidDataDark*.msg)  //header const *header* = *document*.querySelector("header")  const *More* = *document*.querySelector("a[id='mobileIcon']")  //dropdown  function CustomerForm() {  this.form = *document*.createElement("form")  this.form.classList.add("flex-center")  this.form.id = "customer";  const pls = *document*.createElement("h1")  const pls2 = *document*.createElement("p")  pls.style.color = "#d9d9d9"  pls2.style.color = "#d9d9d9"  pls.innerText = "Заполните, пожалуйста, форму."  pls2.innerText = "На ваш email придёт инвайт-код, после чего вы сможете войти в систему"   this.nameLabel = *document*.createElement("label")  this.name = *document*.createElement("input")  this.name.type = "text"  this.name.classList.add("search-field")  this.name.name = "name"  this.name.placeholder = "ФИО"  this.nameLabel.appendChild(this.name)   this.phoneLabel = *document*.createElement("label")  this.phone = *document*.createElement("input")  this.phone.type = "text"  this.phone.classList.add("search-field")  this.phone.name = "phone"  this.phone.placeholder = "+375 33 111 11 11"  this.phoneLabel.appendChild(this.phone)   this.emailLabel = *document*.createElement("label")  this.email = *document*.createElement("input")  this.email.type = "email"  this.email.classList.add("search-field")  this.email.name = "email"  this.email.placeholder = "[example@ellipsevape.me](mailto:example@ellipsevape.me)"  this.emailLabel.appendChild(this.email)   this.townLabel = *document*.createElement("label")  this.town = *document*.createElement("input")  this.town.type = "text"  this.town.classList.add("search-field")  this.town.name = "town"  this.town.placeholder = "г. Минск"  this.townLabel.appendChild(this.town)   this.button = *document*.createElement("button")  this.button.classList.add("fixed-sized")  this.button.formMethod = ""  this.button.innerText = "Связаться"  this.button.type = "button"  this.form.appendChild(pls)  this.form.appendChild(pls2)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.nameLabel)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.phoneLabel)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.emailLabel)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.townLabel)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.button)   this.realButton = this.form.querySelector("button")  this.realButton.addEventListener("mousedown", () => {   /\* let c = 0;  let complete = true;   for(let i of this.name.value) {  if(i === ' ') c++  }  if (c < 1) complete = false;  for(let i in this.phone.value) {  if(!(i === '+' || (i >= '0' && i <= '9') || i === '(' || i === ')' || i === ' ')) {  complete = false;  }  }\*/   if(this.form.reportValidity()  && this.town.value.length > 0  && this.phone.value.length > 0  && this.name.value.length > 0  && this.email.value.length > 0) {  const thanks = *document*.createElement("h1")  const thanks2 = *document*.createElement("p")  thanks.style.color = "#d9d9d9"  thanks2.style.color = "#d9d9d9"  thanks.innerText = `Спасибо, ${this.name.value}.`  thanks2.innerText = `Мы отправим вам инвайт код на ${this.email.value}`  this.form.appendChild(thanks)  this.form.appendChild(thanks2)  *FormSubmitDark*.Show()  this.name.value = ""  this.phone.value = ""  this.email.value = ""  this.town.value = ""   this.name.remove()  this.phone.remove()  this.email.remove()  this.town.remove()  this.realButton.remove()  pls.remove()  pls2.remove()  }  else *InvalidDataDark*.Show()  }) }  const *deviceForm* = *document*.querySelector("button[id='deviceForm']") const *formm* = new CustomerForm() deviceForm.addEventListener("mousedown", () => {  *deviceForm*.remove()  *body*.append(*formm*.form)  *window*.scrollTo({top: 10000, behavior: "smooth"}) }) //dropdown burger AddDefaultMobileDropdown(true, true)  //slider body.prepend(*header*) |

Листинг historyConstructor:

|  |
| --- |
| //notifications body.appendChild(*ClubInviter*.msg)  //header const *header* = *document*.querySelector("header")  const *Club* = *document*.querySelector("a[id='clubIcon']") const *More* = *document*.querySelector("a[id='mobileIcon']")  Club.addEventListener("mouseenter", () => {  *ClubInviter*.Show() })  //dropdown  //dropdown burger AddDefaultMobileDropdown(true, true)  //slider body.prepend(*header*) |

Листинг supportConstructor:

|  |
| --- |
| //notifications body.appendChild(*FormSubmitDark*.msg) body.appendChild(*InvalidDataDark*.msg)  //header const *header* = *document*.querySelector("header")  const *More* = *document*.querySelector("a[id='mobileIcon']")   //dropdown  function CustomerForm() {  this.form = *document*.createElement("form")  this.form.classList.add("flex-center")  this.form.id = "customer";  const pls = *document*.createElement("h1")  const pls2 = *document*.createElement("p")  pls.style.color = "#d9d9d9"  pls2.style.color = "#d9d9d9"  pls.innerText = "Заполните, пожалуйста, форму."  pls2.innerText = "После заполнения формы мы сможем связаться с вами по email"   this.nameLabel = *document*.createElement("label")  this.name = *document*.createElement("input")  this.name.type = "text"  this.name.classList.add("search-field")  this.name.name = "name"  this.name.placeholder = "ФИО"  this.nameLabel.appendChild(this.name)   this.emailLabel = *document*.createElement("label")  this.email = *document*.createElement("input")  this.email.type = "email"  this.email.classList.add("search-field")  this.email.name = "email"  this.email.placeholder = "[example@ellipsevape.me](mailto:example@ellipsevape.me)"  this.emailLabel.appendChild(this.email)   this.descriptionLabel = *document*.createElement("label")  this.description = *document*.createElement("textarea")  this.description.type = "textarea"  this.description.classList.add("search-field")  this.description.name = "textarea"  this.description.placeholder = "Опишите проблему"  this.descriptionLabel.appendChild(this.description)    this.button = *document*.createElement("button")  this.button.classList.add("fixed-sized")  this.button.formMethod = ""  this.button.innerText = "Связаться"  this.button.type = "button"  this.form.appendChild(pls)  this.form.appendChild(pls2)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.nameLabel)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.emailLabel)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.descriptionLabel)  this.form.appendChild(*document*.createElement("br"))  this.form.appendChild(this.button)   this.realButton = this.form.querySelector("button")  this.realButton.addEventListener("mousedown", () => {   /\* let c = 0;  let complete = true;   for(let i of this.name.value) {  if(i === ' ') c++  }  if (c < 1) complete = false;  for(let i in this.phone.value) {  if(!(i === '+' || (i >= '0' && i <= '9') || i === '(' || i === ')' || i === ' ')) {  complete = false;  }  }\*/   if(this.form.reportValidity()  && this.name.value.length > 0  && this.email.value.length > 0  && this.description.value.length > 0) {  *FormSubmitDark*.Show()  this.name.value = ""  this.email.value = ""  this.description.value = ""  this.form.remove()  }  else *InvalidDataDark*.Show()  }) }  const *deviceForm* = *document*.querySelector("button[id='deviceForm']") const *formm* = new CustomerForm() deviceForm.addEventListener("mousedown", () => {  *deviceForm*.remove()  *body*.append(*formm*.form)  *window*.scrollTo({top: 10000, behavior: "smooth"}) }) //dropdown burger AddDefaultMobileDropdown(true, true)  //slider body.prepend(*header*) |