Учреждение образования

«Гродненский государственный политехнический колледж»

**ОТЧЁТ  
ПО УЧЕБНОЙ ПРАКТИКЕ ПО WEB-РАЗРАБОТКЕ**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Учащегося |  | 3 | | | | |  | курса, группы |  | ПЗТ-40 |
| специальности | | |  | 2 - 40 01 01 «Программное обеспечение информационных | | | | | | |
|  | | |  | технологий» | | | | | | |
| Место прохождения практики | | | | |  | УО «Гродненский государственный | | | | |
|  | | | | |  | политехнический колледж» | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Тема проекта: |  | «Разработка front-end части главной страницы сайта «Магазин двер- | | |
|  |  | »ных замков» | | |
| Ссылка на проект: | | |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Выполнил |  |  |  | Н.И. Хартанович |
|  |  |  |  | (инициалы, фамилия) |
| Руководитель практики от колледжа |  |  |  | О.И.Кизер |
|  |  |  | (инициалы, фамилия) |

Содержание

[Введение 3](#_Toc194324354)

[1 Анализ задачи 5](#_Toc194324355)

[1.1 Постановка задачи 5](#_Toc194324356)

[1.2 Инструменты разработки 6](#_Toc194324357)

[2 Проектирование 8](#_Toc194324358)

[2.1 Стайлгайд 8](#_Toc194324359)

[2.2 Описание тестов 11](#_Toc194324359)

[3 Реализация 18](#_Toc194324362)

[3.1 Руководство программиста 18](#_Toc194324359)

[3.2 Спецификация файлов проекта 24](#_Toc194324359)

[4 Тестирование 25](#_Toc194324363)

[5 Применение 27](#_Toc194324364)

[Заключение 28](#_Toc194324368)

[Список использованных источников 29](#_Toc194324369)

[Приложение A Модульная сетка 30](#_Toc194324370)

Введение

На учебной практике была поставлена задача разработать front – end части главной страницы сайта «Магазин дверных замков».

Цель проекта – разработать front – end части главной страницы сайта, которое позволит пользователям удобно находить актуальную информацию о современных дверных замках, добавлять товары в корзину, оставлять заявку на звонок. Данное программное обеспечение будет иметь удобный и интуитивно понятный интерфейс, который позволит пользователям развивать критическое мышление, память и внимание.

Далее краткое описание разделов пояснительной записки.

Первый раздел носит название «Анализ задачи». В нем можно ознакомиться с постановкой задачи и инструментами разработки. Также в этом разделе можно узнать о том, как данная задача решается в настоящее время.

В разделе «Проектирование» будут рассмотрены основные аспекты разработки программного продукта. Здесь можно будет узнать об организации данных в контексте среды разработки. В данном разделе будет описан пользовательский интерфейс, стайлгайд, который содержит в себе логотип и сетку, цветовую палитру сайта, типографику, навигацию и меню, кнопки и поля ввода.

«Реализация» – это третий раздел отчёта по практике, в котором описываются все элементы и объекты, которые будут использованы при реализации данного веб-ресурса. В этом разделе будут описаны функции пользователя и их структура. Здесь можно будет найти таблицу, в которой будет представлена полная аннотация файлов, используемых в данном проекте. Этот раздел так же включает в себя подразделы, такие как: руководство программиста и спецификацию файлов проекта.

Четвёртый раздел – «Тестирование». В нем будет описано функциональное тестирование данного программы, которая выполняется веб-ресурсом. Будут смоделированы возможные действия клиента при работе с данным приложением.

В разделе «Применение» будет описано назначение, область применения, среда функционирования данного ПП.

«Заключение» будет содержать краткую формулировку задачи, результаты проделанной работы, описание использованных методов и средств на различных этапах разработки.

В разделе «Список использованных источников» будет приведён список используемых при разработке источников.

В приложение А к пояснительной записке будут приведены модульные сетки веб-ресурса.

1 Анализ задачи

1.1 Постановка задачи

На учебную практику по WEB ­– программированию была поставлена задача по верстке шаблона по макету из Figma на тему: «Магазин дверных замков».

Требуется сверстать главную страницу сайта по макету, представленному в виде figma – ссылки:

<https://www.figma.com/design/SXRi6her3g4Ebo6yUCojnj/Магазин-замков--Copy-?node-id=0-1&p=f>

Задача: Разработка интернет – ресурса «Магазин дверных замков» HTML, CSS и JS.

Главная страница должна быть сверстана с учётом просмотра её на различных устройствах с разными разрешениями экрана, то есть иметь адаптивную верстку для просмотра на компьютере при различных разрешениях (ширина по умолчанию - 1440рх, макет соответствует ширине 1440рх), а также на мобильном телефоне (ширина экрана менее 470рх) и планшете (ширина экрана от 481рх до 800рх, ширина по умолчанию 768рх).

Также должен быть составлен документ, который содержит в себе список элементов правил, подходящих под стилистику конкретного сайта – стайлгайд. Стайлгайд – это справка набором стандартов и требований, обязательных к соблюдению. Их необходимо использовать процессе оформления сайта. Ключевое назначение - создание единого стилистического оформительского однообразия. На основании данных из Figma стайлгайд должен включать себя следующие разделы:

* логотип;
* цветовая палитра;
* типографика;
* сетка и отступы;
* навигация, меню;
* иконки;
* кнопки, поля ввода;
* компоненты;
* пользовательские формы;
* модальные окна, алерты и т.д.

После завершения составления стайлгайда и вёрстки макета необходимо выполнить помощью CSS отдельных элементов макета в соответствии со следующими стилизацию с критериями:

* hover-эффекты;
* оформление гиперссылок;
* тени;
* градиенты;
* оформление изображений;
* оформление(стилизация) пользовательских форм;
* трансформации;
* анимация и переходы;
* декоративные элементы;
* любые другие эффекты.

Далее необходимо добавить в сверстанный шаблон главной страницы интерактивные, динамические элементы. При их разработке необходимо писать код на JavaScript и использовать библиотеки Jquery, Jquery UI в соответствии со следующими критериями:

* работа с окнами сообщений;
* создание динамического меню и организация навигации;
* слайдеры;
* работа с изображениями через JS;
* использование библиотеки Jquery;
* использование библиотеки Jquery UI;
* использование любых возможных способов JS+библиотеки.

1.2 Инструменты разработки

Для разработки проекта будет выбрана среда разработки Visual Studio Code, которая является наиболее актуальной средой для создания программных продуктов данного типа.

Разработка будет производиться на языках программирования HTML, CSS, JavaScript.

Иные инструменты, используемые при разработке и написании сопутствующей документации:

* Figma – будет использоваться для создания UX – UI макетов проекта;
* Microsoft Office Word 2021 – для написания документации к программному продукту;

– Opera GX – это игровой браузер, который предоставляет специализированные функции. Используется для поиска дополнительной информации;

GitHub – крупнейший веб-сервис для хостинга IT – проектов и их совместной разработки.

Разработка ведется на ноутбуке MACHENIKE S16S16. У данного ноутбука следующие параметры:

* процессор 12th Gen Intel(R) Core(TM) i7-12700H 2.30 GHz;
* объем ОЗУ – 16 Гб;
* объем места на SSD – 512 Гб;
* видеоподсистема 1920х1080 точек с глубиной цвета 16 Bit;
* ОС Windows 11.

Данное разрабатываемый интернет–ресурс не очень требовательно к аппаратным ресурсам, что является большим плюсом.

2 Проектирование

2.1 Стайлгайд

Логотип – графический знак, эмблема или символ, используемый территориальными образованиями, коммерческими предприятиями, организации и частными лицами для повышения узнаваемости и распознаваемости в социуме. Логотип представляет собой название сущности, которую он идентифицирует, в виде стилизованных букв и/или идеограммы. Логотип представлен на рисунке 1.



Рисунок 1 – Логотип

Модульная сетка сайта для персонального компьютера (ширина 1440рх) и для телефона (ширина 360px) представлена в приложении А.

Цветовая палитра:

* основной цвет: темно-синий (#2c3e50);
* вторичный цвет: серо-голубой (#3498db);
* акцентный цвет: голубой (#4295E4);
* фоновый цвет: #F1F1F1 (белый).

Типографика – сильнейший инструмент для выражения посыла в веб дизайне. С его помощью вы можете объединить текстовую и визуальную составляющие, что поможет вам достучаться до посетителя. Типографика для шаблона макета «GoldenSoft» представлена ниже:

* основный шрифт в проекте является Arial;
* размер шрифта составляет 12 – 26px;
* интервал между строками в среднем составляет 15 – 30px;
* используемые цвета текста в проекте #FFFFFF (белый), #454F5B (серый), #EA5D4A (красный), #000001 (черный), #24C56E (зеленый)

Важнейшей составляющей любого сайта является навигация и меню. Частая ошибка юзабилити сайта – это сложная или запутанная навигация, которая отталкивает клиента.

Простое правило, чем проще ориентироваться на сайте, тем быстрее пользователь найдёт необходимую информацию.

В шапке сайта пользователь без трудностей может перейти к различным разделам сайта, увидеть специальное предложение и контакты. Шапка представлена на рисунке 2.

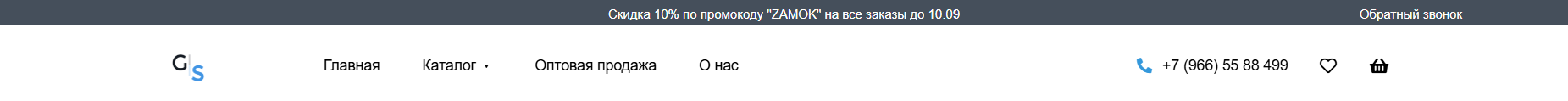


Рисунок 2 – Шапка сайта

Подвал данного макета схож со структурой шапки сайта. Подвал представлен на рисунке 3.

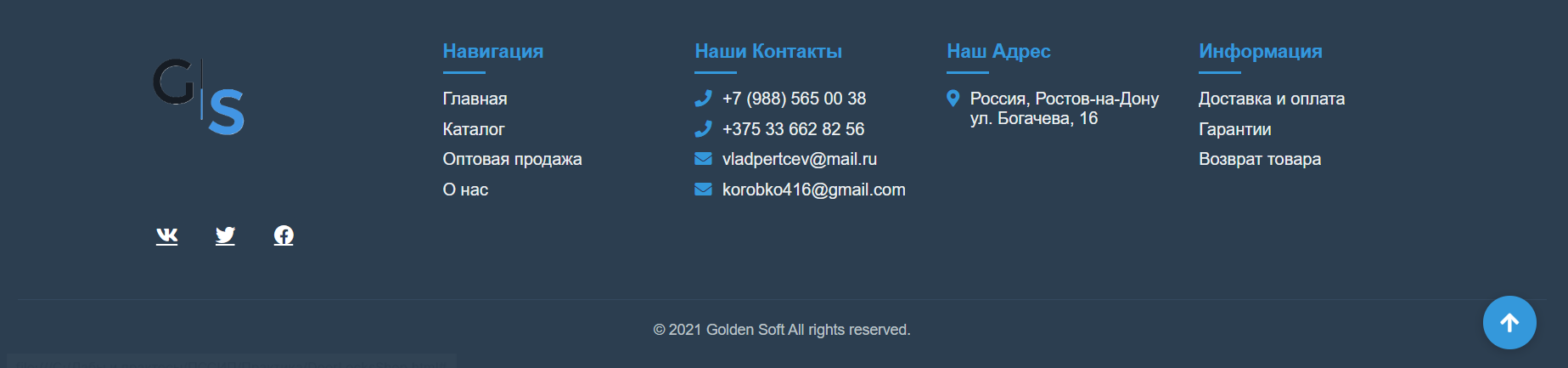
****

Рисунок 3 – Подвал

Интерактивные элементы пользовательского интерфейса создают основу мощных и удобных цифровых продуктов. Внимательный подход к деталям помогает создать положительный опыт взаимодействия с продуктом в целом. Кнопка является одним из самых распространенных элементов любого интерфейса.

Кнопка «Корзина», кнопка «Избранное» представлены на рисунке 4.

Рисунок 4 – Кнопки «Записаться на экскурсию», «Социальные сети»

Слайдер, на котором находятся карточки товаров и кнопка «Добавить в корзину» представлен на рисунке 5.

Рисунок 5 – Слайдер

Категории товаров разделены на отдельные блоки с кнопками «Перейти», которые представлены на рисунке 6.

Рисунок 6 – Категория товара

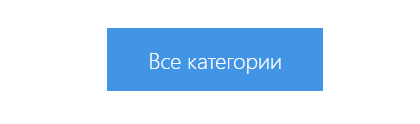
Кнопка «Все категории» представлена на рисунке 7.

Рисунок 7 – Кнопка «Все категории»

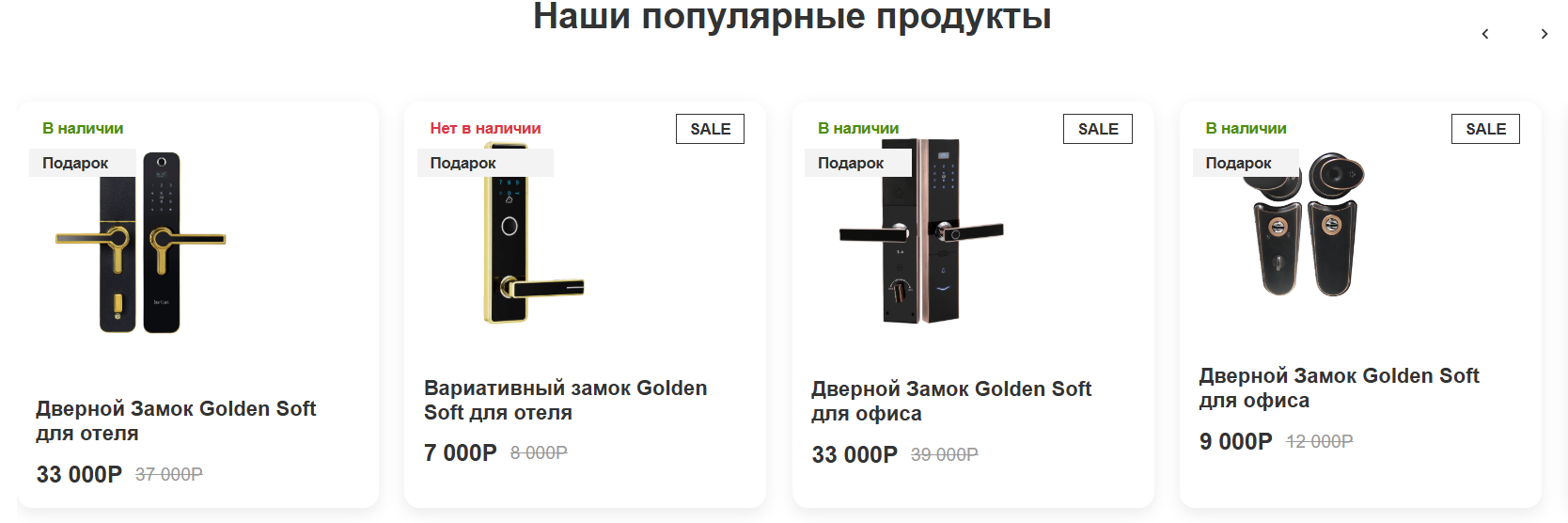
Слайдер с карточками популярных продуктов представлен на рисунке 8.

Рисунок 8 – Слайдер с популярными товарами

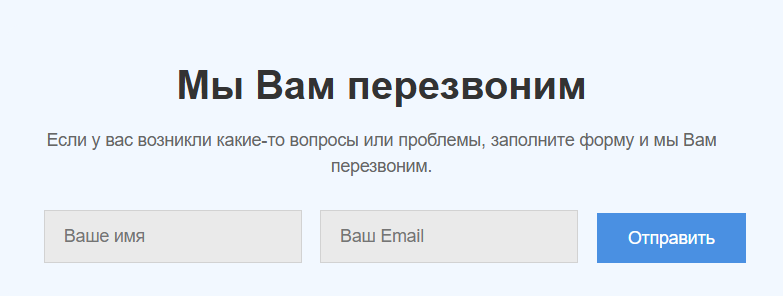
Поля для ввода в форме «Мы Вам перезвоним» и кнопка «Отправить» представлены на рисунке 9.

Рисунок 9 – Поля и кнопка

**2.2 Описание тестов**

При разработке проекта необходимо будет провести некоторые функциональные и нефункциональные тесты. Они должны охватывать различные аспекты функциональности, такие как кнопки, поля ввода, ссылки.

Результаты тестов на использование представлены в таблице 1.

Таблица 1 – Тесты на использование

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор | Приоритет | Заглавие и шаги выполнения | Ожидаемый результат |
| 1 | 2 | 3 | 4 |
| T\_01 | высокий | **Слайдер, правая кнопка прокрутки**  1. Открыть страницу «Магазин дверных замков».  2. Нажать на правую кнопку прокрутки на слайдере. | 1. Загружается страница «Магазин дверных замков».  2. Карусель листается вправо, появляется новое изображение. |
| T\_02 | высокий | **Слайдер, левая кнопка прокрутки**  1. Открыть страницу «Магазин дверных замков».  2. Нажать на левую кнопку прокрутки на слайдере. | 1. Загружается страница «Магазин дверных замков».  2. Карусель листается влево, появляется новое изображение. |
| T\_03 | низкий | **Анимация карточек товара**  1. Открыть страницу «Магазин дверных замков».  2. Навести курсор на карточку. | 1. Загружается страница «Магазин дверных замков».  2. Изображение поднимается вверх. |
| Т\_04 | низкий | **Hover-эффект при наведении на кнопку**  1. Открыть страницу «Магазин дверных замков».  2. Навести курсор на кнопку «Добавить в корзину». | 1. Загружается страница «Магазин дверных замков».  2. Кнопка изменяет свой цвет. |
| Т\_05 | средний | **Кнопка «Перейти»**  1. Открыть страницу «Магазин дверных замков».  2. Нажать на кнопку «Перейти». | 1. Загружается страница «Магазин дверных замков».  2. Переход вверх на главную страницу. |
|  |  |  |  |
| Продолжение Таблицы 1 | | | |
| Т\_06 | высокий | **Форма «Мы Вам перезвоним»**  1. Открыть страницу «Магазин дверных замков».  2. В поле «Ваше имя» ввести «Никита».  3. В поле «Ваш Email» ввести «nikalatosh@gmail.com».  4. Нажать на кнопку «Отправить» | 1. Загружается страница «Магазин дверных замков».  2. Введённое значение отображается в поле «Ваше имя».  3. Введённое значение отображается в поле «Ваш Email».  4. Появляется сообщение «Спасибо! Мы скоро с вами свяжемся». |
| T\_07 | средний | **Кнопка «Логотип» в футере**  1. Открыть страницу «Магазин дверных замков».  2. Нажать на кнопку «Логотип». | 1. Загружается страница «Магазин дверных замков».  2. Переход вверх на главную страницу. |
| T\_08 | средний | **Копирование при нажатии на информацию в футере**  1. Открыть страницу «Магазин дверных замков».  2. Нажать на номер «+7 (988) 565 00 38». | 1. Загружается страница «Магазин дверных замков».  2. Вместо «+7 (988) 565 00 38» отображается «Скопировано!» |
| T\_09 | средний | **Кнопка «Наверх»**  1. Открыть страницу «Магазин дверных замков».  2. Нажать на кнопку «Наверх». | 1. Загружается страница «Магазин дверных замков».  2. Переход вверх на главную страницу. |

**3 Реализация**

**3.1 Руководство программиста**

Общие требования:

Сайт должен быть разработан с помощью HTML, CSS и JS.

Сайт должен быть совместим с популярными браузерами и устройствами.

Сайт должен обеспечивать безопасность и защищенность данных клиентов.

Структура проекта:

Использовать HTML5 для создания структуры страницы.

Использовать семантические теги для улучшения доступности и поисковой оптимизации. Пример кода HTML:

<body>

<div class="top-banner">

<div class="banner-content">

Скидка 10% по промокоду "ZAMOK" на все заказы до 10.09

</div>

<a href="#" class="banner-link">Обратный звонок</a>

</div>

<header>

<div class="header-container">

<img src="IMG/logo.png" width="35" alt="Логотип">

<nav>

<ul>

<li><a href="#">Главная</a></li>

<li class="has-dropdown">

<a href="#">Каталог</a>

<ul class="dropdown-menu">

<li><a href="#">Цилиндровые замки</a></li>

<li><a href="#">Сувальдные замки</a></li>

<li><a href="#">Электронные замки</a></li>

<li><a href="#">Врезные замки</a></li>

<li><a href="#">Накладные замки</a></li>

</ul>

</li>

<li><a href="#">Оптовая продажа</a></li>

<li><a href="#">О нас</a></li>

</ul>

</nav>

<div class="header-contacts">

<div class="phone">

<i class="fas fa-phone-alt"></i>

<span>+7 (966) 55 88 499</span>

</div>

<div class="icon-buttons">

<div class="icon-btn" title="Избранное">

<i class="far fa-heart"></i>

</div>

<div class="icon-btn" title="Корзина">

<i class="fas fa-shopping-basket"></i>

</div>

</div>

</div>

</div>

<div class="burger" id="burger">

<span></span>

<span></span>

<span></span>

</div>

</div>

<!-- Мобильное меню -->

<nav class="mobile-nav" id="mobileNav">

<ul>

<li><a href="#">Главная</a></li>

<li><a href="#">Каталог</a></li>

<li><a href="#">Оптовая продажа</a></li>

<li><a href="#">О нас</a></li>

</ul>

</nav>

</header>

<div class="slider-container">

<div class="product-slider" id="productSlider">

<!-- Карточка товара 1 -->

<div class="product-card">

<div class="product-image">

<img src="IMG/lock1.png" alt="Golden Soft GS-200Z-5">

</div>

<div class="product-info">

<div class="brand">Golden Soft</div>

<div class="model">GS-200Z-5 для офиса</div>

<p class="description">

Замок дверной электронный Golden Soft GS-200Z-5 имеет роскошный глянцевый блеск, четкие линии, красивые формы.

</p>

<p class="features">

Подходит для установки на деревянную/межкомнатную дверь

</p>

<p class="price-text">Цена</p>

<div class="price-container">

<span class="current-price">33 000₽</span>

<span class="old-price">37 000₽</span>

</div>

<button class="add-to-cart">Добавить в корзину</button>

</div>

</div>

Использовать CSS3 для создания стилей и дизайна сайта. Пример кода CSS:

        nav ul li a:hover {

            color: #3498db;/\*Изменение цвета ссылки в шапке при наведении\*/

        }

        .add-to-cart {

            background-color: #4295e4;

            color: white;

            border: none;

            padding: 12px 30px;

            font-size: 16px;

            font-weight: lighter;

            cursor: pointer;

            transition: background-color 0.3s;

            font-family: system-ui, -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica Neue', sans-serif;

        }

        .add-to-cart:hover {

            background-color: #3370a9; /\*Изменение цвета при наведении на кнопку\*/

        }

        .icon-btn {

            width: 36px;

            height: 36px;

            border-radius: 50%;

            background: none;

            display: flex;

            align-items: center;

            justify-content: center;

            cursor: pointer;

            transition: background-color 0.3s;

        }

        .icon-btn:hover {

            background-color: #f0f0f0; /\*Hover-эффект для кнопок в шапке\*/

        }

        .product-cardS {

            flex: 0 0 280px;

            background-color: rgb(255, 255, 255);

            border-radius: 12px; /\*Скругление углов\*/

            box-shadow: 0 4px 12px rgba(0, 0, 0, 0.08); /\*Тени\*/

            overflow: hidden;

            position: relative;

            transition: transform 0.3s;

        }

        .product-cardS:hover {

            transform: translateY(-5px); /\*Анимация карточки товара при наведении\*/

        }

        .social-icon::before {

                content: '';

                position: absolute;

                top: 0;

                left: 0;

                width: 100%;

                height: 100%;

                background: linear-gradient(45deg, /\*Градиентная заливка при наведении\*/

                    #3498db,

                    #9b59b6,

                    #e74c3c,

                    #3498db);

                background-size: 300% 300%;

                opacity: 0;

                transition: opacity 0.3s ease;

                z-index: -1;

        }

        .back-to-top i {

            transition: transform 0.3s ease; /\*Анимация стрелки «Наверх» при наведении\*/

        }

Использовать медиа-запросы в CSS3 для создания адаптивного сайта. При экране меньше, чем 768px – адаптивный сайт для телефона, при экране больше 1000px и меньше 768px – адаптивный сайт для планшета. Пример кода с медиа-запросами:

@media (max-width: 768px) {

            .menu-container{

                padding: 0 150px;

                justify-content: space-between;

                flex-direction: column;

            }

            .product-imageR{

                width: 60%;

            }

            .product-titleR{

                font-size: 11px;

            }

            .product-buttonR{

                height: 20px;

                width: 40px;

                font-size: 8px;

            }

            .burger {

                display: flex;

            }

            .mobile-nav.active {

                display: block;

            }

            .footer-container {

                grid-template-columns: 1fr;

            }

            .footer-column {

                margin-bottom: 30px;

            }

            .advantages-container{

                flex-direction: column;

                padding: 10px;

                margin-bottom: 350px;

            }

            .stats-container {

                flex-direction: column;

                height: auto;

                padding: 20px;

                margin-bottom: 80px;

            }

            .stat-item {

                padding: 15px 0;

                width: 100%;

            }

            .stat-item:not(:last-child)::after {

                display: none;

            }

            .stat-item:not(:last-child) {

                border-bottom: 1px solid #ddd;

            }

            .header-container {

                display: none;

            }

            nav ul {

                margin-left: 0;

                order: 3;

                width: 100%;

                margin-top: 15px;

                justify-content: space-around;

            }

            .header-contacts {

                margin-left: auto;

                order: 2;

            }

            .dropdown-menu {

                position: static;

                box-shadow: none;

                border-top: 1px solid #eee;

                display: none;

            }

            .has-dropdown.active .dropdown-menu {

                display: block;

            }

            .slider-container {

                padding: 0 15px;

            }

            .product-card {

                flex-direction: column;

            }

            .product-image, .product-info {

                width: 100%;

            }

            .product-cardS {

                flex: 0 0 240px;

            }

            .section-titleS {

                font-size: 24px;

            }

            .slider-nav {

                top: -45px;

            }

            .callback-form {

                grid-template-columns: 1fr;

            }

            .submit-btn {

                width: 100%;

                margin-top: 10px;

            }

            .advantages-item:not(:last-child)::after {

                display: none;

            }

            .banner-link{

                display: none;

            }

Использовать JavaScript для создания интерактивных элементов и функциональности сайта.

Использовать библиотеку jQuery для упрощения написания JS-кода. Пример кода JavaScript:

**//кнопка «наверх»**

        document.addEventListener('DOMContentLoaded', function() {

            // Получаем кнопку

            const backToTopButton = document.querySelector('.back-to-top');

            // Обработчик скролла

            window.addEventListener('scroll', function() {

                // Показываем/скрываем кнопку

                if (window.pageYOffset > 300) {

                    backToTopButton.classList.add('active');

                } else {

                    backToTopButton.classList.remove('active');

                }

            });

            // Обработчик клика

            backToTopButton.addEventListener('click', function(e) {

                e.preventDefault();

                // Плавная прокрутка вверх

                window.scrollTo({

                    top: 0,

                    behavior: 'smooth'

                });

            });

        });

//**клик по логотипу в футере**

            document.addEventListener('DOMContentLoaded', function() {

            // Находим логотип в футере

            const footerLogo = document.querySelector('.footer-logo');

            // Добавляем обработчик клика

            footerLogo.addEventListener('click', function(e) {

                e.preventDefault(); // Предотвращаем стандартное поведение

                // Плавный скролл наверх страницы

                window.scrollTo({

                    top: 0,

                    behavior: 'smooth'

                });

            });

            // Делаем логотип кликабельным (добавляем курсор)

            footerLogo.style.cursor = 'pointer';

        });

**//бургер**

        document.getElementById('burger').addEventListener('click', function() {

            this.classList.toggle('active');

            document.getElementById('mobileNav').classList.toggle('active');

        });

**//анимация при скролле к футеру**

        document.addEventListener('DOMContentLoaded', function() {

            // Анимация при скролле к футеру

            const footer = document.querySelector('footer');

            function checkFooterVisibility() {

                const footerPosition = footer.getBoundingClientRect().top;

                const screenPosition = window.innerHeight / 1.2;

                if (footerPosition < screenPosition) {

                    footer.style.opacity = '1';

                    footer.style.transform = 'translateY(0)';

                }

            }

            // Инициализация стилей для анимации

            footer.style.opacity = '0';

            footer.style.transform = 'translateY(20px)';

            footer.style.transition = 'opacity 0.5s ease, transform 0.5s ease';

            // Проверка при загрузке и скролле

            checkFooterVisibility();

            window.addEventListener('scroll', checkFooterVisibility);

**// скрипт для копирования контактов**

            const contactItems = document.querySelectorAll('.contact-item');

            contactItems.forEach(item => {

                item.addEventListener('click', function() {

                    const text = this.innerText.trim();

                    navigator.clipboard.writeText(text).then(() => {

                        const originalText = this.innerHTML;

                        this.innerHTML = '<span class="contact-icon"><i class="fas fa-check"></i></span> <span>Скопировано!</span>';

                        setTimeout(() => {

                            this.innerHTML = originalText;

                        }, 2000);

                    });

                });

            });

        });

**//валидация имени**

        document.addEventListener('DOMContentLoaded', function() {

            const nameInput = document.getElementById('name');

            const nameError = document.getElementById('nameError');

            // Валидация имени

            function validateName() {

                if (nameInput.value.trim() === '') {

                    nameError.style.display = 'block';

                    return false;

                } else {

                    nameError.style.display = 'none';

                    return true;

                }

            }

            // Валидация в реальном времени

            nameInput.addEventListener('input', function() {

                validateName();

            });

            // Валидация при потере фокуса

            nameInput.addEventListener('blur', function() {

                validateName();

            });

        });

**//валидация Email**

        document.addEventListener('DOMContentLoaded', function() {

            const emailInput = document.getElementById('email');

            const emailError = document.getElementById('emailError');

            // Валидация email

            function validateEmail() {

                const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

                if (!emailRegex.test(emailInput.value.trim())) {

                    emailError.style.display = 'block';

                    return false;

                } else {

                    emailError.style.display = 'none';

                    return true;

                }

            }

            // Валидация в реальном времени

            emailInput.addEventListener('input', function() {

                validateEmail();

            });

            // Валидация при потере фокуса

            emailInput.addEventListener('blur', function() {

                validateEmail();

            });

        });

**//обработчик отправки формы**

        document.addEventListener('DOMContentLoaded', function() {

            const form = document.getElementById('callbackForm');

            const submitBtn = document.getElementById('submitBtn');

            const successMessage = document.getElementById('successMessage');

            // Обработчик отправки формы

            form.addEventListener('submit', function(e) {

                e.preventDefault();

                // Вызываем валидации для обоих полей

                const isNameValid = validateName();

                const isEmailValid = validateEmail();

                if (isNameValid && isEmailValid) {

                    // Здесь можно добавить AJAX запрос к серверу

                    submitBtn.disabled = true;

                    submitBtn.textContent = 'Отправка...';

                    // Имитация отправки данных

                    setTimeout(function() {

                        form.style.display = 'none';

                        successMessage.style.display = 'block';

                        // Очистка формы (для демонстрации)

                        document.getElementById('name').value = '';

                        document.getElementById('email').value = '';

                        submitBtn.disabled = false;

                        submitBtn.textContent = 'Отправить';

                    }, 1500);

                }

            });

            // Объявляем функции валидации глобально для доступа из других скриптов

            window.validateName = function() {

                const nameInput = document.getElementById('name');

                const nameError = document.getElementById('nameError');

                if (nameInput.value.trim() === '') {

                    nameError.style.display = 'block';

                    return false;

                } else {

                    nameError.style.display = 'none';

                    return true;

                }

            };

            window.validateEmail = function() {

                const emailInput = document.getElementById('email');

                const emailError = document.getElementById('emailError');

                const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

                if (!emailRegex.test(emailInput.value.trim())) {

                    emailError.style.display = 'block';

                    return false;

                } else {

                    emailError.style.display = 'none';

                    return true;

                }

            };

        });

**//слайдер с точками**

        document.addEventListener('DOMContentLoaded', function() {

            const slider = document.getElementById('slider');

            const prevBtn = document.getElementById('prevnBtn');

            const nextBtn = document.getElementById('nextnBtn');

            const cardWidth = 280 + 20; // Ширина карточки + gap

            let scrollPosition = 0;

            const maxScroll = slider.scrollWidth - slider.clientWidth;

            // Проверяем, есть ли куда скроллить

            function updateButtons() {

                prevBtn.disabled = scrollPosition <= 0;

                nextBtn.disabled = scrollPosition >= maxScroll;

            }

            // Обработчики для кнопок

            prevBtn.addEventListener('click', function() {

                scrollPosition = Math.max(scrollPosition - cardWidth \* 2, 0);

                slider.scrollTo({

                    left: scrollPosition,

                    behavior: 'smooth'

                });

                setTimeout(updateButtons, 300);

            });

            nextBtn.addEventListener('click', function() {

                scrollPosition = Math.min(scrollPosition + cardWidth \* 2, maxScroll);

                slider.scrollTo({

                    left: scrollPosition,

                    behavior: 'smooth'

                });

                setTimeout(updateButtons, 300);

            });

            // Инициализация кнопок

            updateButtons();

            // Обработчик для скролла мышью

            slider.addEventListener('scroll', function() {

                scrollPosition = slider.scrollLeft;

                updateButtons();

            });

            // Адаптация к изменению размера окна

            window.addEventListener('resize', function() {

                maxScroll = slider.scrollWidth - slider.clientWidth;

                updateButtons();

            });

        });

        // Слайдер товаров

        const slider = document.getElementById('productSlider');

        const prevBtn = document.getElementById('prevBtn');

        const nextBtn = document.getElementById('nextBtn');

        const dotsContainer = document.getElementById('sliderDots');

        const cards = document.querySelectorAll('.product-card');

        const cardWidth = document.querySelector('.product-card').offsetWidth + 20;

        // Создаем точки для навигации

        cards.forEach((\_, index) => {

            const dot = document.createElement('div');

            dot.classList.add('dot');

            if (index === 0) dot.classList.add('active');

            dot.addEventListener('click', () => {

                scrollToCard(index);

            });

            dotsContainer.appendChild(dot);

        });

        const dots = document.querySelectorAll('.dot');

        function scrollToCard(index) {

            slider.scrollTo({

                left: index \* cardWidth,

                behavior: 'smooth'

            });

            updateActiveDot(index);

        }

        function updateActiveDot(index) {

            dots.forEach(dot => dot.classList.remove('active'));

            dots[index].classList.add('active');

        }

        nextBtn.addEventListener('click', () => {

            const scrollPosition = slider.scrollLeft;

            const maxScroll = slider.scrollWidth - slider.clientWidth;

            if (scrollPosition < maxScroll) {

                const nextCard = Math.floor((scrollPosition + cardWidth / 2) / cardWidth) + 1;

                scrollToCard(nextCard);

            } else {

                scrollToCard(0);

            }

        });

        prevBtn.addEventListener('click', () => {

            const scrollPosition = slider.scrollLeft;

            if (scrollPosition > 0) {

                const prevCard = Math.floor((scrollPosition - cardWidth / 2) / cardWidth);

                scrollToCard(prevCard);

            } else {

                scrollToCard(cards.length - 1);

            }

        });

        slider.addEventListener('scroll', () => {

            const scrollPosition = slider.scrollLeft;

            const activeCard = Math.round(scrollPosition / cardWidth);

            updateActiveDot(activeCard);

        });

**// Выпадающее меню для мобильных устройств**

        document.addEventListener('DOMContentLoaded', function() {

            const dropdownToggles = document.querySelectorAll('.has-dropdown > a');

            dropdownToggles.forEach(toggle => {

                toggle.addEventListener('click', function(e) {

                    if (window.innerWidth <= 768) {

                        e.preventDefault();

                        const parent = this.parentElement;

                        const isActive = parent.classList.contains('active');

                        // Закрываем все открытые меню

                        document.querySelectorAll('.has-dropdown').forEach(item => {

                            item.classList.remove('active');

                        });

                        // Открываем текущее, если было закрыто

                        if (!isActive) {

                            parent.classList.add('active');

                        }

                    }

                });

            });

**// Закрытие меню при клике вне его**

            document.addEventListener('click', function(e) {

                if (window.innerWidth <= 768 && !e.target.closest('.has-dropdown')) {

                    document.querySelectorAll('.has-dropdown').forEach(item => {

                        item.classList.remove('active');

                    });

                }

            });

        });

**3.2 Спецификация файлов проекта**

Расположение файлов проекта важная часть любого проекта. Благодаря этому можно легко ориентироваться в проекте, особенно другим участникам проекта.

Спецификация файлов проекта представлена в таблице 2.

Таблица 2 – Спецификация файлов проекта

|  |  |  |
| --- | --- | --- |
| Папка | Название файла | Назначение |
| Практика | DoorLocksShop.html | Главный файл |
| CSS | advantages.css  Categories.css  footer.css  form.css  slider.css  statstyle.css  top.css  header.css | Стили сайта |
| JS | script.js | Скрипты |
| IMG | \*.png | Изображения |

**4 Тестирование**

При тестировании программного продукта, в первую очередь нужно обратить внимание на правильную работу страницы. Показывает ли текст, видны ли картинки, нет ли неравномерных отступов, все ли работают пункты меню. Нужно проверить верную работу всех подключённых скриптов для правильной работы сайта. Также необходимо проверить имеется ли адаптация под разные устройства.

Таким образом, основная часть ошибок и недоработок была выявлена и исправлена на этапе реализации проекта. После завершения этапа реализации, было проведено тщательное тестирование.

Расписание проведения и время, затраченное на тестирование, описано в таблице 3.

Таблица 3 – Расписание работ над проектом

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Дата | Деятельность | Продолжительность, ч |
| Хартанович Никита | 19.05.2025 | Разработка тестов | 2 |
| Хартанович Никита | 19.05.2025 | Тестирование главной страницы сайта | 3 |
| Хартанович Никита | 20.05.2025 | Составление отчетов о найденных дефектах | 3 |
| Хартанович Никита | 20.05.2025 | Исправление найденных ошибок | 5 |
| Хартанович Никита | 21.05.2025 | Проведение регрессионного тестирования | 2 |
| Хартанович Никита | 22.05.2025 | Составление отчета о результатах тестирования | 3 |

Элементы сайта были проверены, и было установлено, что все они работают правильно и выполняют задачи, указанные в процедурах.

Статистика по всем дефектам представлена в таблице 4.

Таблица 4 – Статистика по всем дефектам

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Статус | Количество | Важность | | | |
| Низкая | Средняя | Высокая | Критическая |
| Найдено | 0 | 0 | 0 | 0 | 0 |
| Исправлено | 0 | 0 | 0 | 0 | 0 |
| Проверено | 0 | 0 | 0 | 0 | 0 |
| Открыто заново | 0 | 0 | 0 | 0 | 0 |
| Отклонено | 0 | 0 | 0 | 0 | 0 |

Таким образом, после проведения тестирования, не были выявлены критические ошибки.

Опираясь на вышеперечисленное, можно сделать вывод о стабильности работы программного продукта и его основного функционала. Соответственно, тестирование прошло успешно и программный продукт характеризуется как реализованный и готовый к работе.

5 Применение

Цель данного проекта заключается в создании инновационного онлайн-ресурса, ориентированного на продажу дверных замков.

Быстродействие любой программы во многом зависит от характеристик выбранного персонального компьютера: рабочей частоты процессора, объема оперативной памяти и т.д.

При верстке данного шаблона использовался персональный компьютер со следующими характеристиками:

* процессор 12th Gen Intel(R) Core(TM) i7-12700H 2.30 GHz;
* объем ОЗУ – 16 Гб;
* объем места на SSD – 512 Гб;
* видеоподсистема 1920х1080 точек с глубиной цвета 16 Bit;
* ОС Windows 10.

Сайт будет размещен на платформе GitHub, где пользователи смогут просматривать и скачивать код сайта. Ссылка на GitHub: https://github.com/shkamillochka/PRACTICA-PSSIP.

Главная страница сайта изображена на рисунке 10.

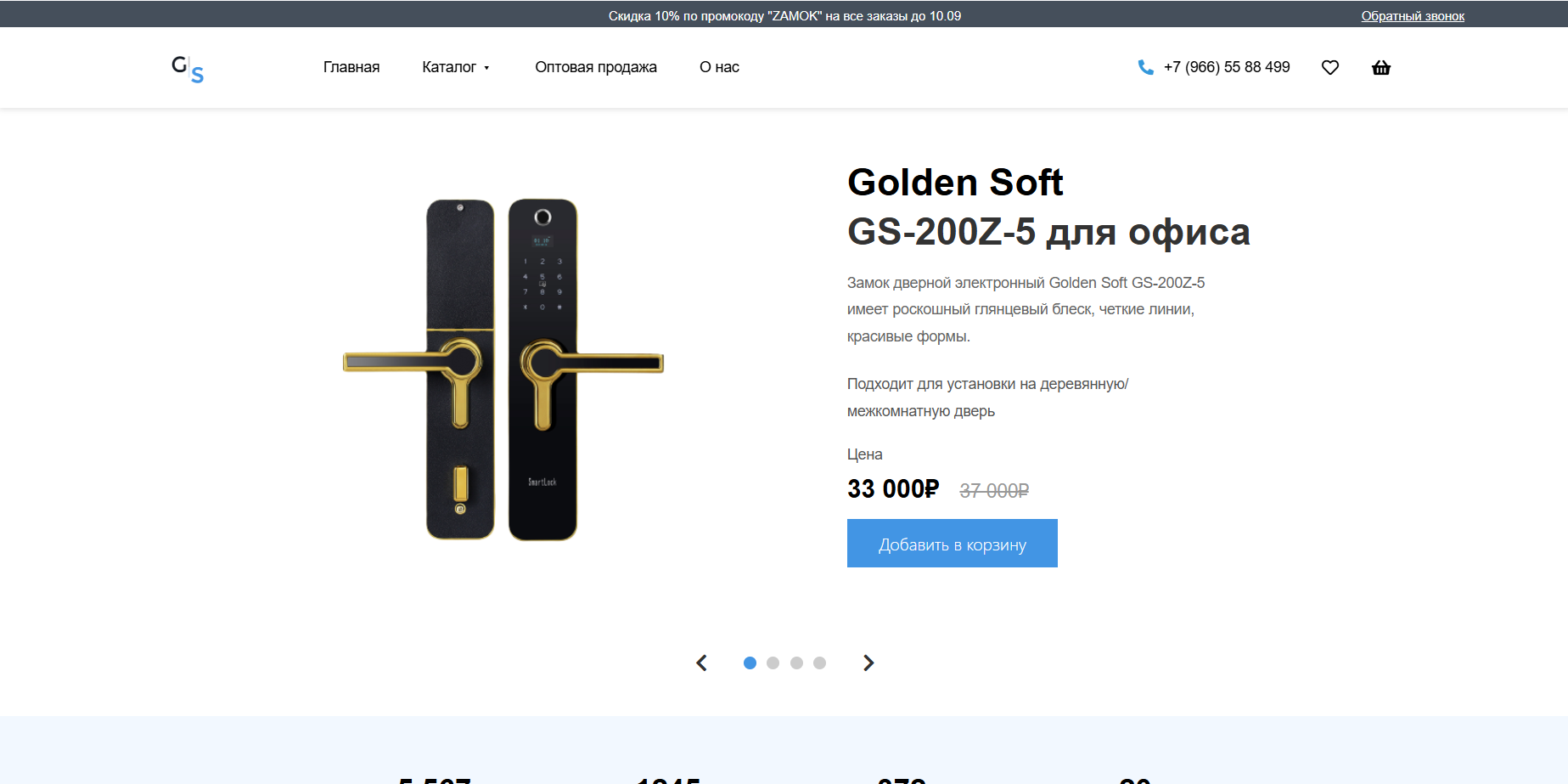


Рисунок 10 – Главная страница сайта

**Заключение**

Разработка современного и функционального интернет-магазина «Golden Soft», специализирующегося на продаже дверных замков для различных нужд (офисы, гостиницы, дома). Проект направлен на предоставление пользователям удобного способа выбора и покупки качественных дверных замков, а также на повышение узнаваемости бренда.

В ходе тестирования интернет-магазина не было выявлено критических ошибок или исключительных ситуаций. Все функции, включая каталог товаров, корзину покупок и систему фильтрации, работают стабильно и без сбоев. Проект полностью готов к использованию.

В процессе разработки были усовершенствованы навыки работы с HTML, CSS и JavaScript, включая создание динамических элементов интерфейса, таких как слайдеры товаров, фильтры и интерактивные кнопки. Особое внимание уделялось обработке исключительных ситуаций, что обеспечило высокую надежность ресурса. Также были закреплены умения по оптимизации производительности и адаптации интерфейса под различные устройства.

Интерфейс магазина разработан с учетом удобства и простоты использования. Интуитивно понятная навигация, четкая структура каталога и минималистичный дизайн позволяют пользователям быстро находить нужные товары и оформлять заказы. Особое внимание уделено адаптивности, благодаря чему сайт корректно отображается как на компьютерах, так и на мобильных устройствах.

Список использованных источников

1. Сайт с информацией о работе с html [Электронный ресурс]. – Режим доступа: https://htmlacademy.ru/ – Дата доступа: 12.05.2025.

2. Знакомство с CSS [Электронный ресурс]. – Режим доступа: https://htmlacademy.ru/courses/41 – Дата доступа: 13.05.2025.

3. Разработка с помощью JS [Электронный ресурс]. – Режим доступа: https://htmlacademy.ru/intensive/javascript – Дата доступа: 14.05.2025.

4. Введение в jQuery [Электронный ресурс]. – Режим доступа: https://metanit.com/web/jquery/8.1.php – Дата доступа: 15.05.2025.

Приложение A  
Модульная сетка

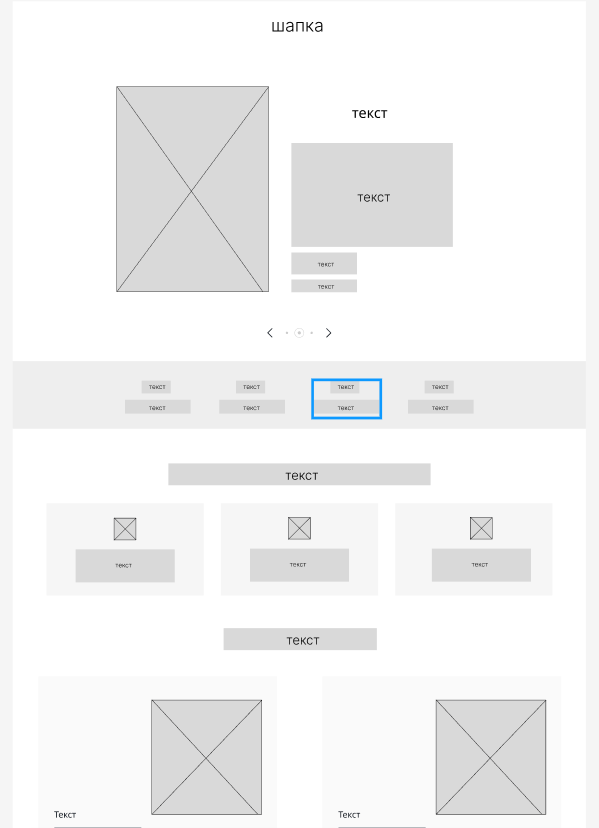
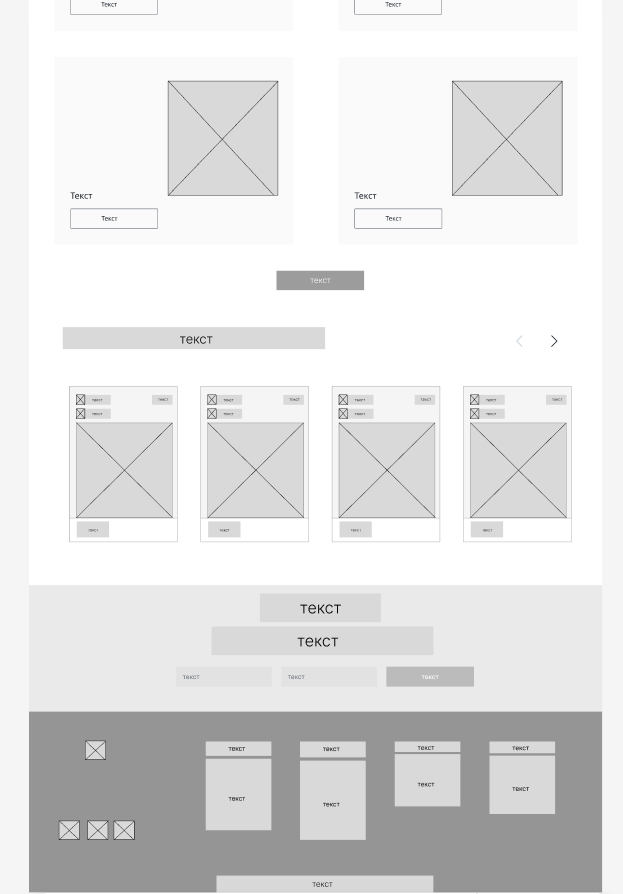
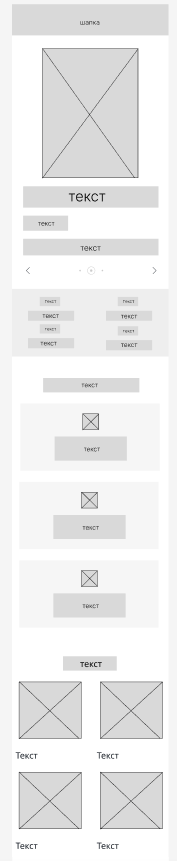
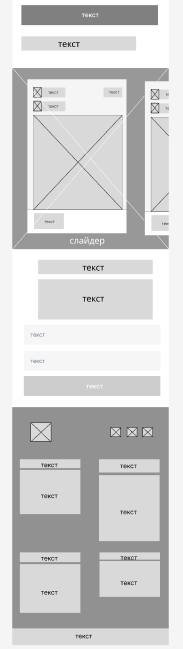


Рисунок А.1 – ПК-версия

Рисунок А.2 – ПК-версия

Рисунок А.3 – Мобильная версия