**Оглавление**

[Введение 2](#_Toc187323499)

[1 Постановка задачи на проектирование интерфейса 3](#_Toc187323500)

[2 Предметная область 4](#_Toc187323501)

[3 Обоснование проектных решений по видам обеспечения 5](#_Toc187323502)

[3.1 По информационному обеспечению 5](#_Toc187323503)

[3.1.1 Технология доступа к данным 7](#_Toc187323504)

[3.1.2 IDEF0 7](#_Toc187323505)

[4 Интерфейс пользователя 10](#_Toc187323506)

[Заключение 15](#_Toc187323507)

[Библиографический список 16](#_Toc187323508)

[Приложения 17](#_Toc187323509)

[Приложение А HTML разметка и PHP код 17](#_Toc187323510)

[Приложение Б CSS стили 19](#_Toc187323511)

[Приложение В JavaScript код 24](#_Toc187323512)

# **Введение**

62,5% мирового населению используют интернет, люди в сети проводят около 12,5 триллионов часов каждый год и с каждый годом эта цифра закономерно увеличивается. Каждый день люди через интернет быстро узнают информацию о том или ином товаре и покупают его, используя веб-сайты. Веб-сайт — это одна страница или набор связанных по смыслу страниц, которые находятся в сети Интернет. На них размещается информация, текстовая или графическая: изображения, видео, анимации.

В современном мире цифровых технологий наличие профессионального и функционального веб-сайта стало неотъемлемой частью успешного бизнеса. Веб-студии играют ключевую роль в создании и поддержке таких сайтов, предоставляя широкий спектр услуг, начиная от дизайна и разработки, заканчивая маркетингом и SEO-оптимизацией.

# **1 Постановка задачи на проектирование интерфейса**

Разрабатываемое веб-приложение предназначено для демонстрации услуг веб-студии, портфолио, и способом взаимосвязи.

Целью является создание веб-приложения с интуитивно понятным интерфейсом, позволяющим быстро найти нужную информацию о веб-студии, рассчитать стоимость предпочитаемых работ и с возможностью просмотра оставленных заявок клиентов для работников веб-студии.

Также целями является:

* Создать, настроить и подключить базы данных для веб-приложения;
* Спроектировать контекстную и детализированную диаграмму IDEF0;
* Выполнить SEO-оптимизацию;
* Сделать отчёт о проделанной работе.

# **2 Предметная область**

Предметная область для веб-приложения веб-студии включает в себя следующие элементы:

* Веб-студия — это компания, специализирующаяся на создании, разработке и поддержке веб-сайтов и веб-приложений. Она предоставляет услуги по веб-дизайну, программированию, SEO-оптимизации и маркетингу, помогая клиентам создать эффективное онлайн-присутствие.
* SEO-оптимизация (Search Engine Optimization) — это процесс улучшения видимости и позиций веб-сайта в результатах поисковых систем, таких как Google. Основная цель SEO — привлечь органический трафик на сайт, улучшая его контент, структуру и технические аспекты.
* Хостинг — это услуга, предоставляющая место для хранения и доступа к веб-сайтам и приложениям на серверах в интернете. Хостинг-провайдеры обеспечивают необходимые ресурсы и инфраструктуру для работы сайтов, включая серверное пространство, базы данных и техническую поддержку.
* SMM (Social Media Marketing) ресурс — это платформа или инструмент, используемый для продвижения и управления присутствием бренда в социальных сетях. SMM ресурсы помогают создавать контент, планировать публикации, анализировать эффективность кампаний и взаимодействовать с аудиторией.
* SPA (Single Page Application) приложение — это тип веб-приложения, которое загружает одну веб-страницу и динамически обновляет её содержимое в ответ на действия пользователя, без необходимости перезагрузки страницы. В отличие от традиционных веб-сайтов, где каждая новая страница требует полной перезагрузки, SPA использует JavaScript для асинхронного обновления контента. Это позволяет создавать более плавный и быстрый пользовательский опыт, так как данные обновляются в реальном времени через API.

# **3 Обоснование проектных решений по видам обеспечения**

## **3.1 По информационному обеспечению**

Информационное обеспечение – важнейший элемент веб-сайта, который сочетает в себе единую систему классификации и кодирования информации.

В качестве входных данных используется:

* имя, электронная почта и сообщение клиента;

Выходной информацией является:

* набор данных об оставленной заявке;

Рассмотрим подробнее, какие реальные объекты предметной области отражают выделенные сущности и как отношения между сущностями соответствуют взаимосвязям объектов на практике.

Структура связанных сущностей приведена в таблице 1.

Таблица 1 – Описание атрибутов базы данных

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Сущность | Тип  сущности | Атрибут | Тип  данных | Ключевое поле |
| 1 | tickets | Главная | id | Числовой | Первичный ключ |
| name | Текстовой |  |
| email | Текстовой |  |
| text | Текстовой |  |
| 2 | admins | Главная | id | Числовой | Первичный ключ |
| login | Текстовой |  |
| password | Текстовой |  |

### **3.1.1 Технология доступа к данным**

Технологии доступа к данным являются прослойкой между специальным интерфейсом программы/приложения, с помощью которого одна программа/приложение может взаимодействовать с другой (application programming interface – API), конкретного сервера и приложением пользователя, предоставляя программисту простой унифицированный механизм работы с данными.

На сегодняшний день существует множество технологий доступа к данным, таких как BDE, OLE, ODBC, DАО, АDО, и до сих пор разрабатываются новые, более надежные, удобные в работе и более быстродействующие технологии.

В этой работе использовалась система управления базами данным phpMyAdmin. Интерфейс СУБД phpMyAdmin представлен на рисунке 1.

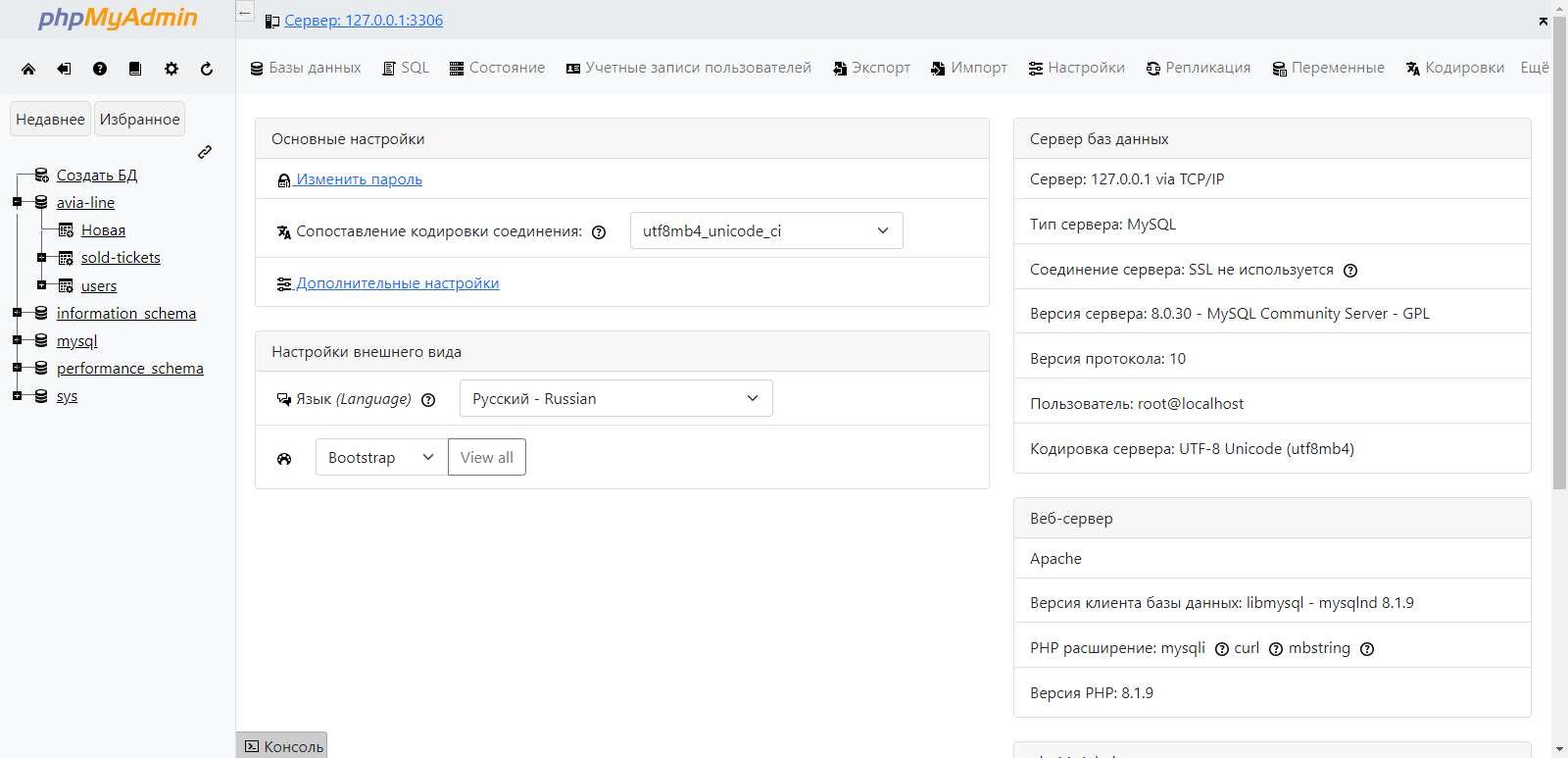


Рисунок 1 – Интерфейс СУБД phpMyAdmin

### **3.1.2 IDEF0**

IDEF0 — методология функционального моделирования (англ. function modeling) и графическая нотация, предназначенная для формализации и описания бизнес-процессов. Отличительной особенностью IDEF0 является ее акцент на соподчиненность объектов. В IDEF0 рассматриваются логические отношения между работами, а не их временная последовательность (поток работ).

Стандарт IDEF0 был разработан в 1981 году в США департаментом Военно-воздушных сил для автоматизации промышленных предприятий. В процессе разработки программного обеспечения разработчики столкнулись с необходимостью разработки новых методов анализа бизнес-процессов. В результате появилась методология функционального моделирования IDEF0, в которой для анализа применяются специальные нотации IDEF0.

Функциональная модель IDEF0 представляет собой набор блоков, каждый из которых представляет собой «черный ящик» со входами и выходами, управлением и механизмами, которые детализируются (декомпозируются) до необходимого уровня. Наиболее важная функция расположена в верхнем левом углу. А соединяются функции между собой при помощи стрелок и описаний функциональных блоков. При этом каждый вид стрелки или активности имеет собственное значение. Данная модель позволяет описать все основные виды процессов, как административные, так и организационные. Стрелки могут быть:

* Входящие – вводные, которые ставят определенную задачу.
* Исходящие – выводящие результат деятельности.
* Управляющие (сверху вниз) – механизмы управления (положения, инструкции и прочее).
* Механизмы (снизу вверх) – что используется для того, чтобы произвести необходимую работу.

Контекстная и детализированная диаграммы для разрабатываемого веб-сайта представлены на рисунках 2 и 3 соответственно.

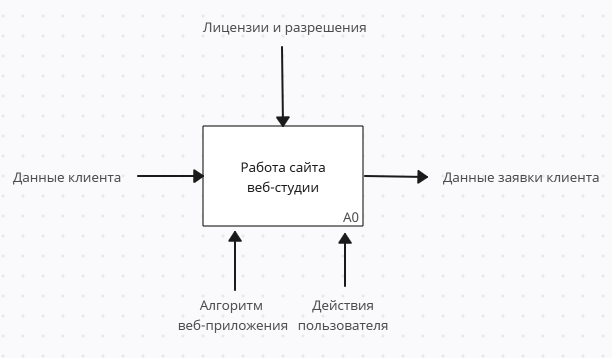


Рисунок 2 – Контекстная диаграмма IDEF0

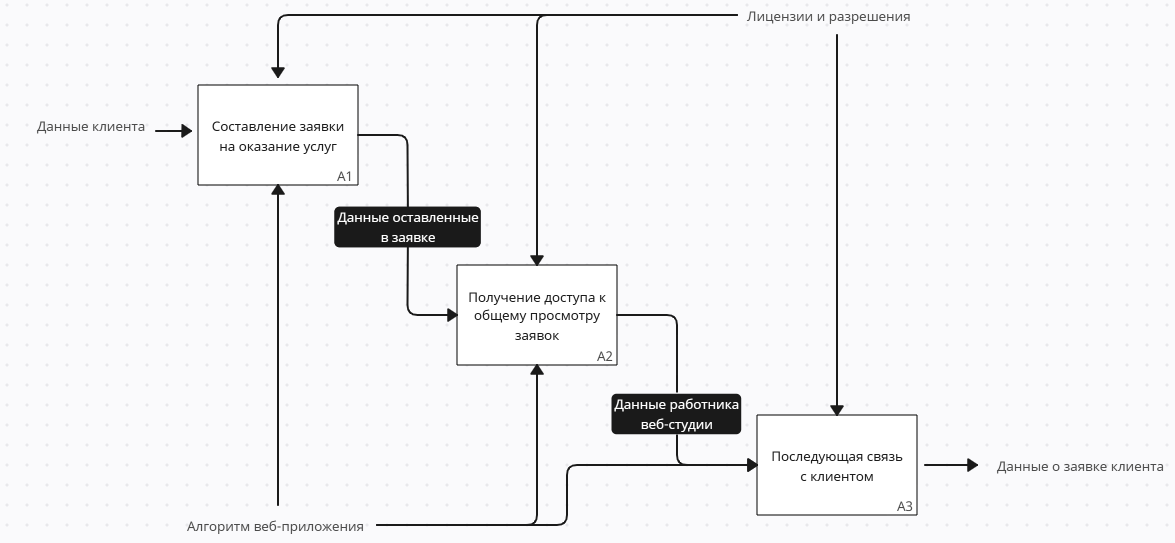


Рисунок 3 – Детализированная диаграмма IDEF0

Для создания диаграмм IDEF0 в этой работе использовался веб-сайт для построения диаграмм – app.creately.com.

# **4 Интерфейс пользователя**

Веб-сайт разрабатывался в среде разработки Visual Studio Code с помощью языка гипертекстовой разметки HTML, языка стилей CSS и языков программирования JavaScript и PHP.

При открытии веб-сайта пользователь попадает на главную страницу где может увидеть навигационную панель для быстрого и удобного перемещения между страницами, также на главной странице находится краткое описание деятельности веб-студии и кнопка «Узнать больше» для перехода на страницу «Портфолио», главная страница представлена на рисунке 4.

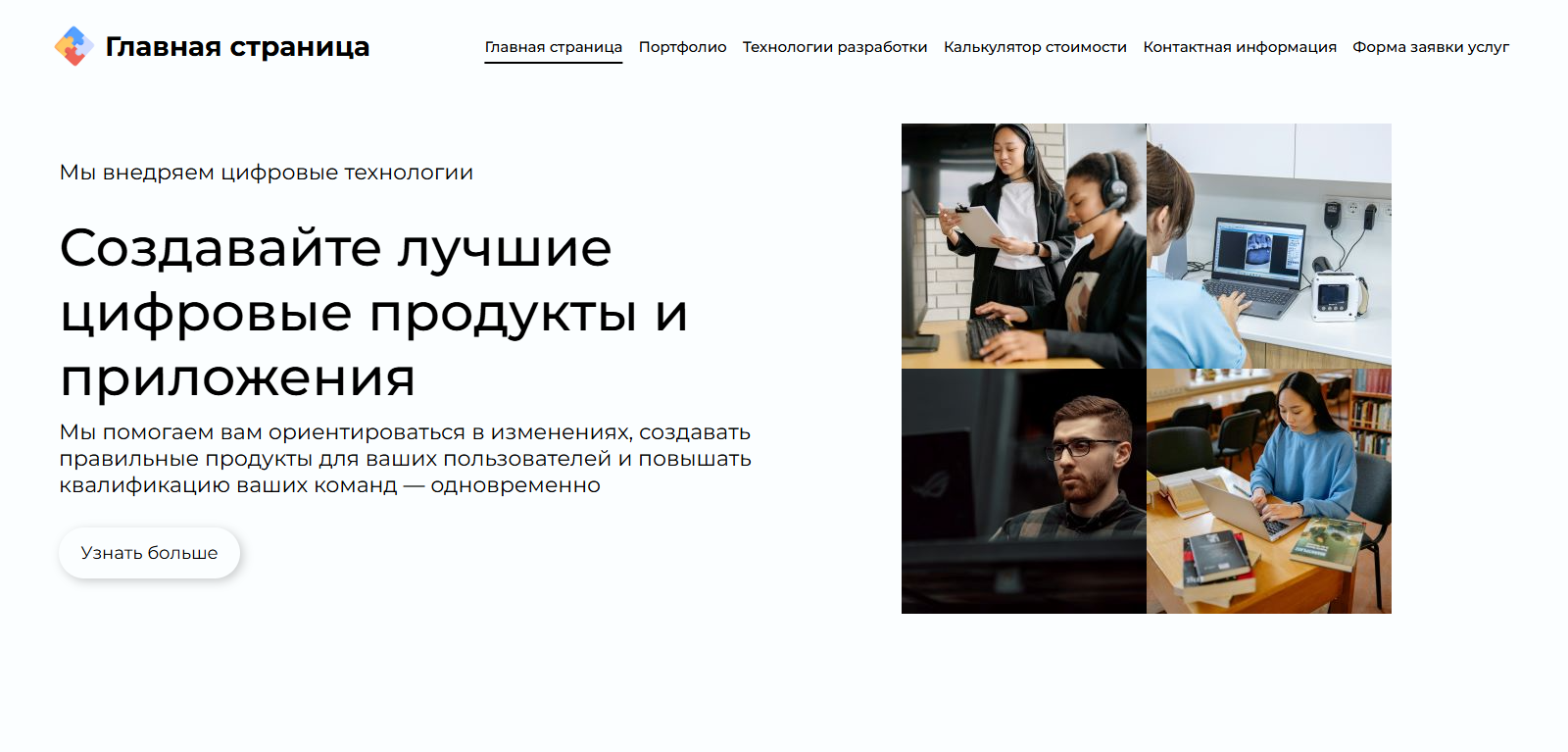


Рисунок 4 – главная страница

При нажатии на кнопку «Узнать больше» или на ссылку «Портфолио» на панели навигации пользователь будет перемещён на страницу «Портфолио», представленная на рисунке 5.

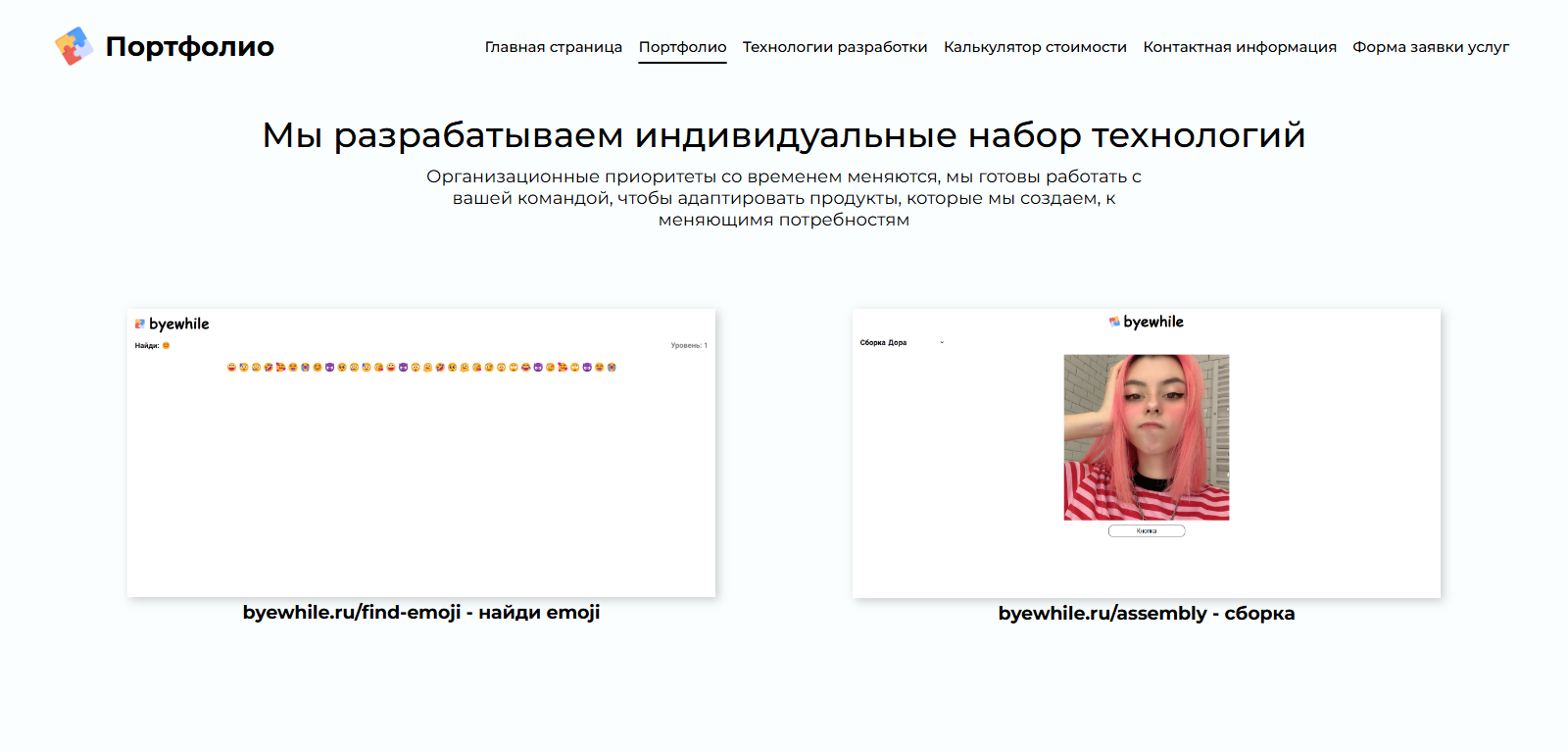


Рисунок 5 – страница «Портфолио»

На странице «Портфолио» пользователь может ознакомиться с примерами работ веб-студии в том числе и перейти на них непосредственно нажат на сам пример.

При нажатии на ссылку «Технологии разработки» в навигационной панели пользователь попадёт на соответствующую страницу, где будут представленные соответствующие технологии, которые веб-студия использует для выполнения задач, страница «Технологии разработки» представлена на рисунке 6.

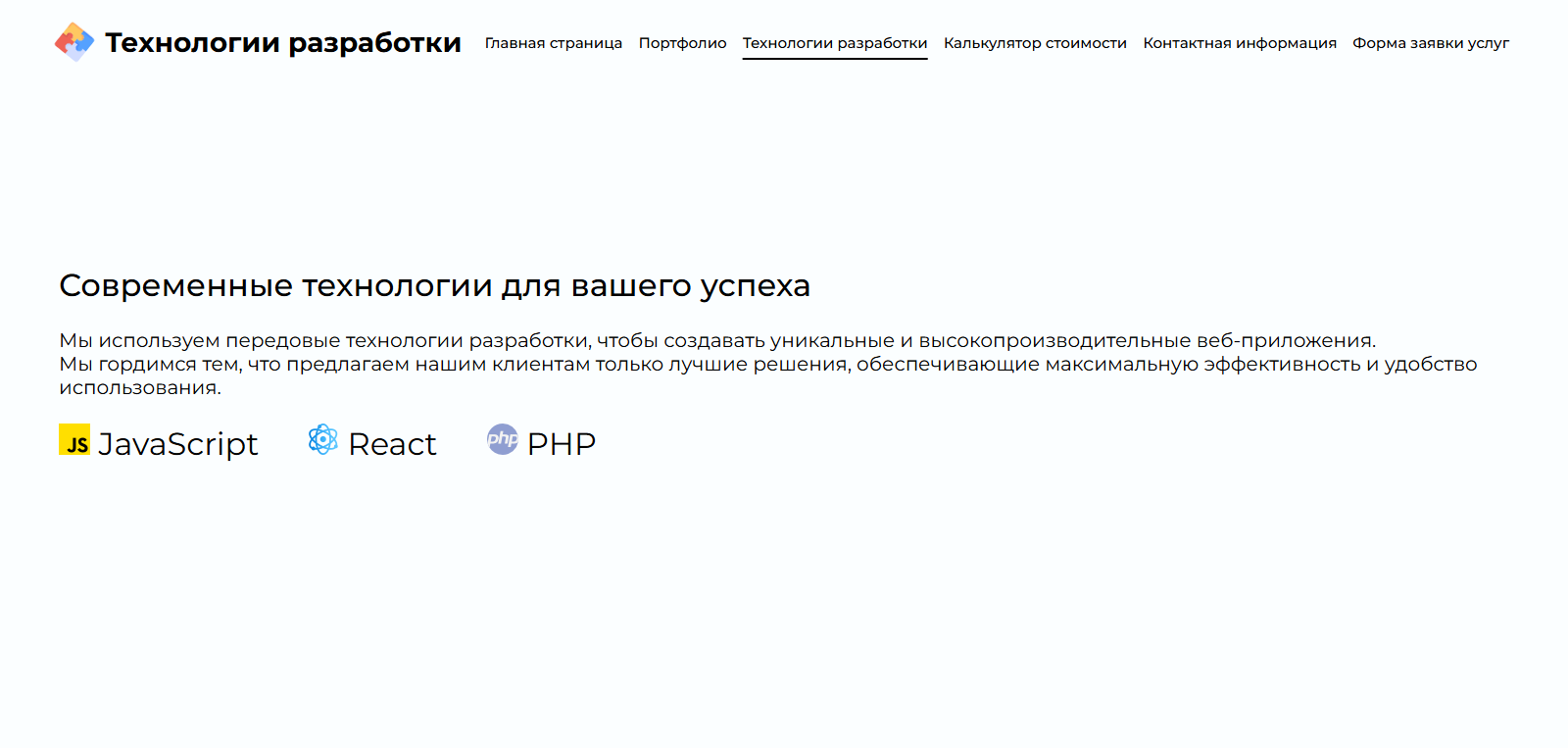


Рисунок 6 – страница «Технологии разработки»

При нажатие на ссылку «Калькулятор стоимости» пользователь будет перемещён на соответствующую страницу, где он может быть ознакомлен с ценовой политикой на услуги веб-студии, выбирая определённую услугу, калькулятор автоматически добавляет стоимость этой услуги к итогу и показывает итоговую цену, данная страница представлена на рисунке 7.

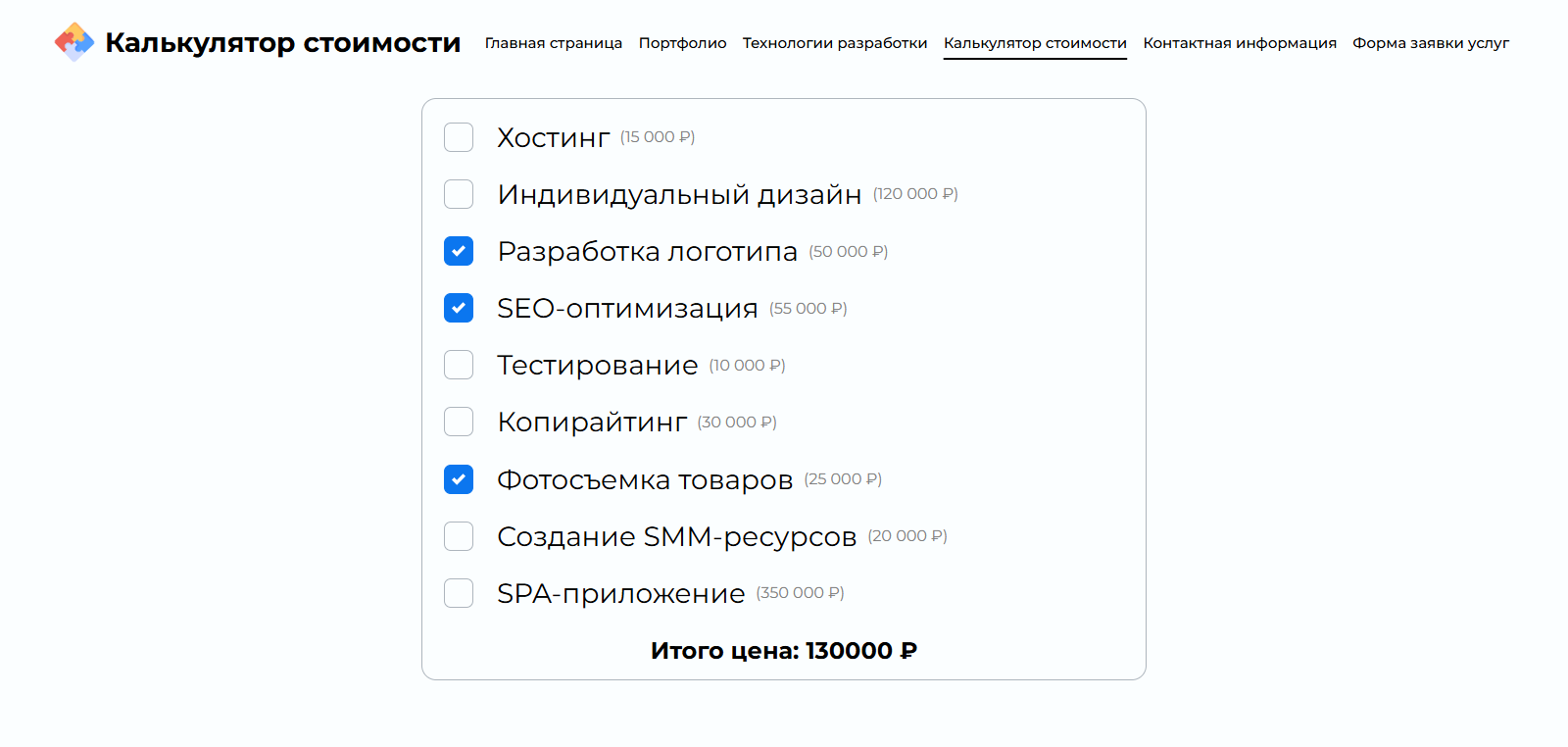


Рисунок 7 – страница «Калькулятор стоимости»

При последующем перемещение по навигационной панели пользователь может попасть на страницу «Контактная информация», представленная на рисунке 8.

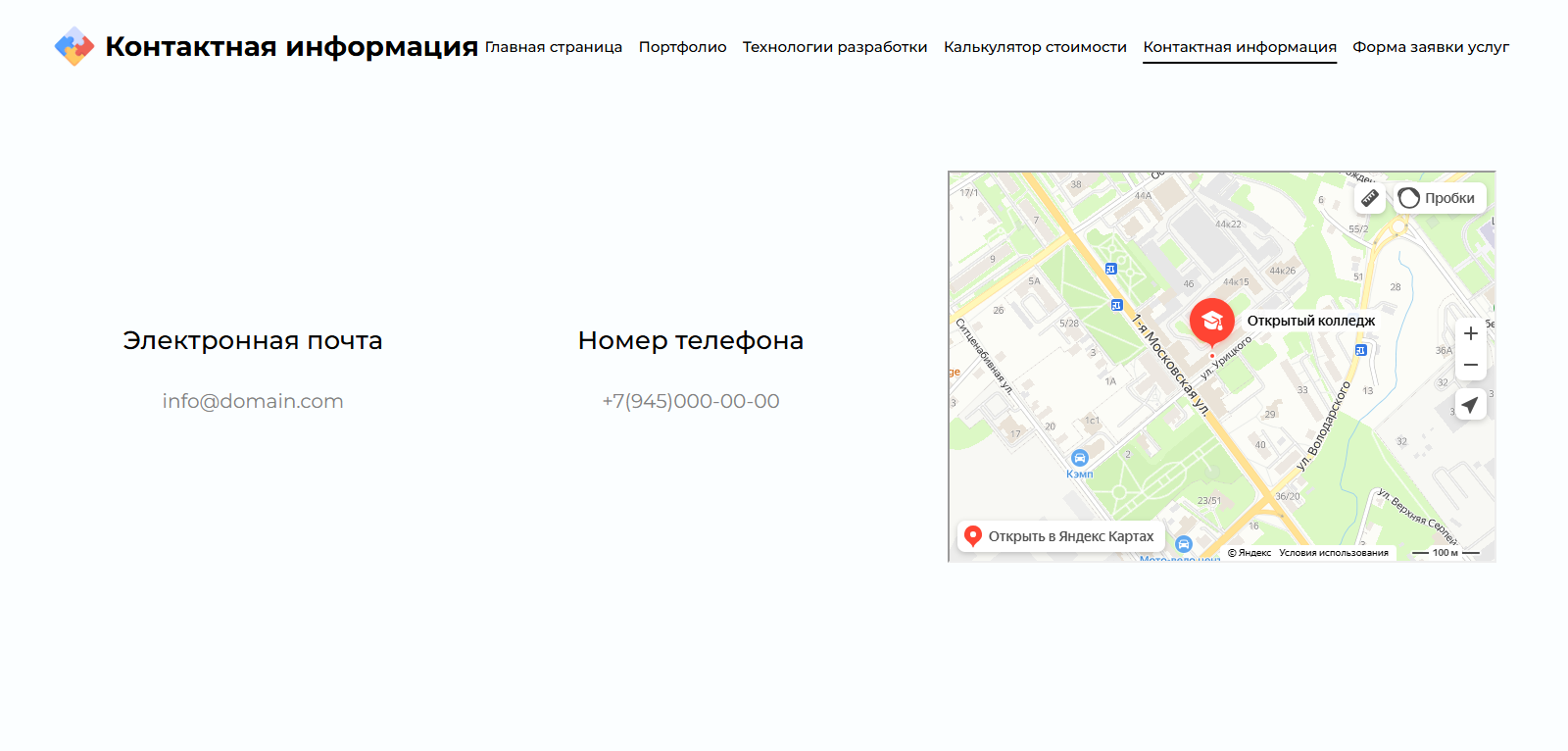


Рисунок 8 – страница «Контактная информация»

На данной странице пользователю представляется информация для связи с веб-студией (электронная почта, номер телефона, адрес на интерактивной карте).

На последней странице доступная обычному пользователя «Форма обратной связи» расположена соответствующая форма, где требуется ввести имя, электронную почту и сообщение. Данные с формы будут обработаны и отправлены в базу данных, в будущем эти данные могут быть просмотрены в качестве заявки работниками веб-студии. Страница «Форма обратной связи» представлена на рисунке 9.

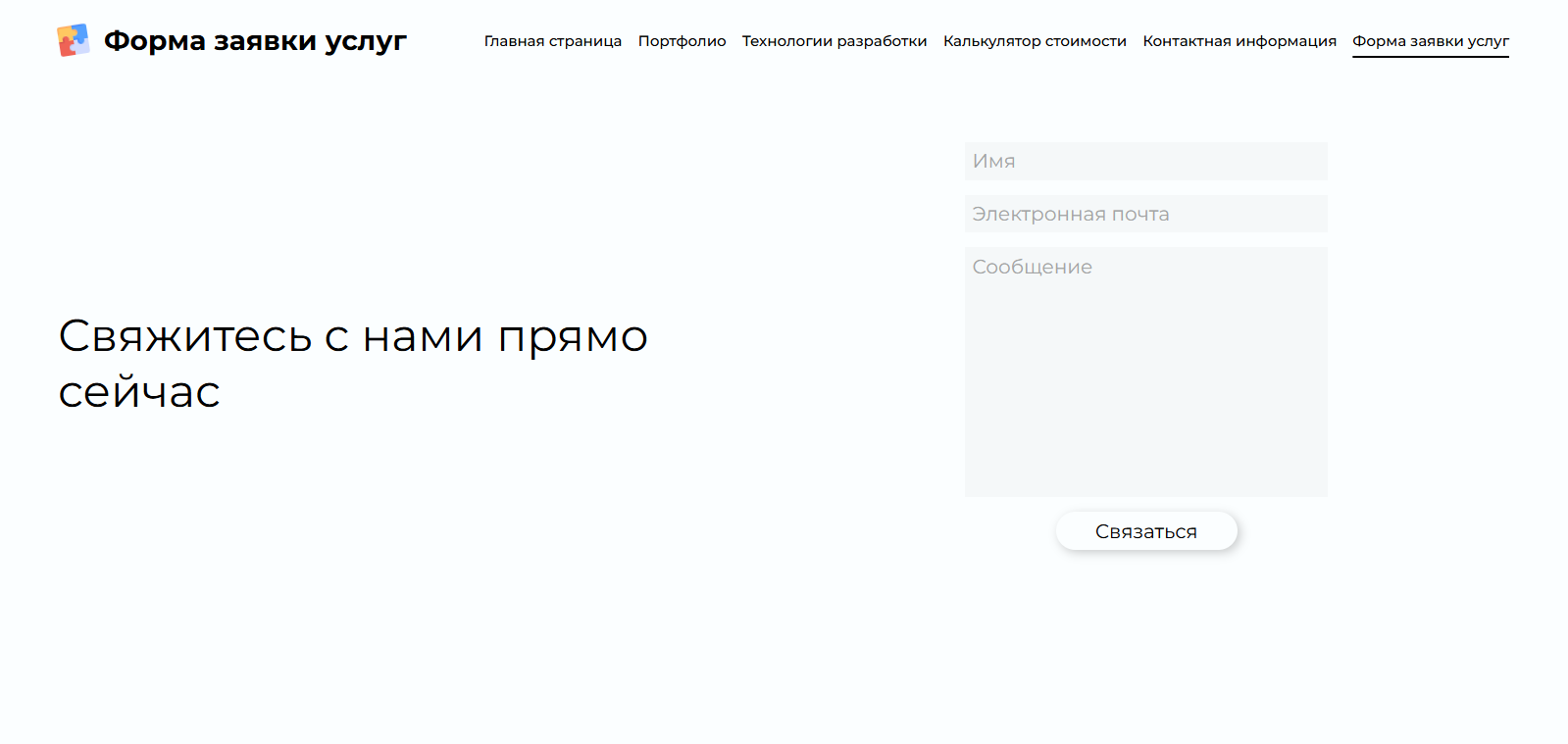


Рисунок 9 – страница «Форма обратной связи»

Для просмотра оставленных заявок существует отдельная страница, которой нет в навигационной панели, для перехода на неё необходимо заменить адрес страницы на admin.php. Пользователь будет перемещён на страницу авторизации если он не был авторизирован, представлено на рисунке 10.

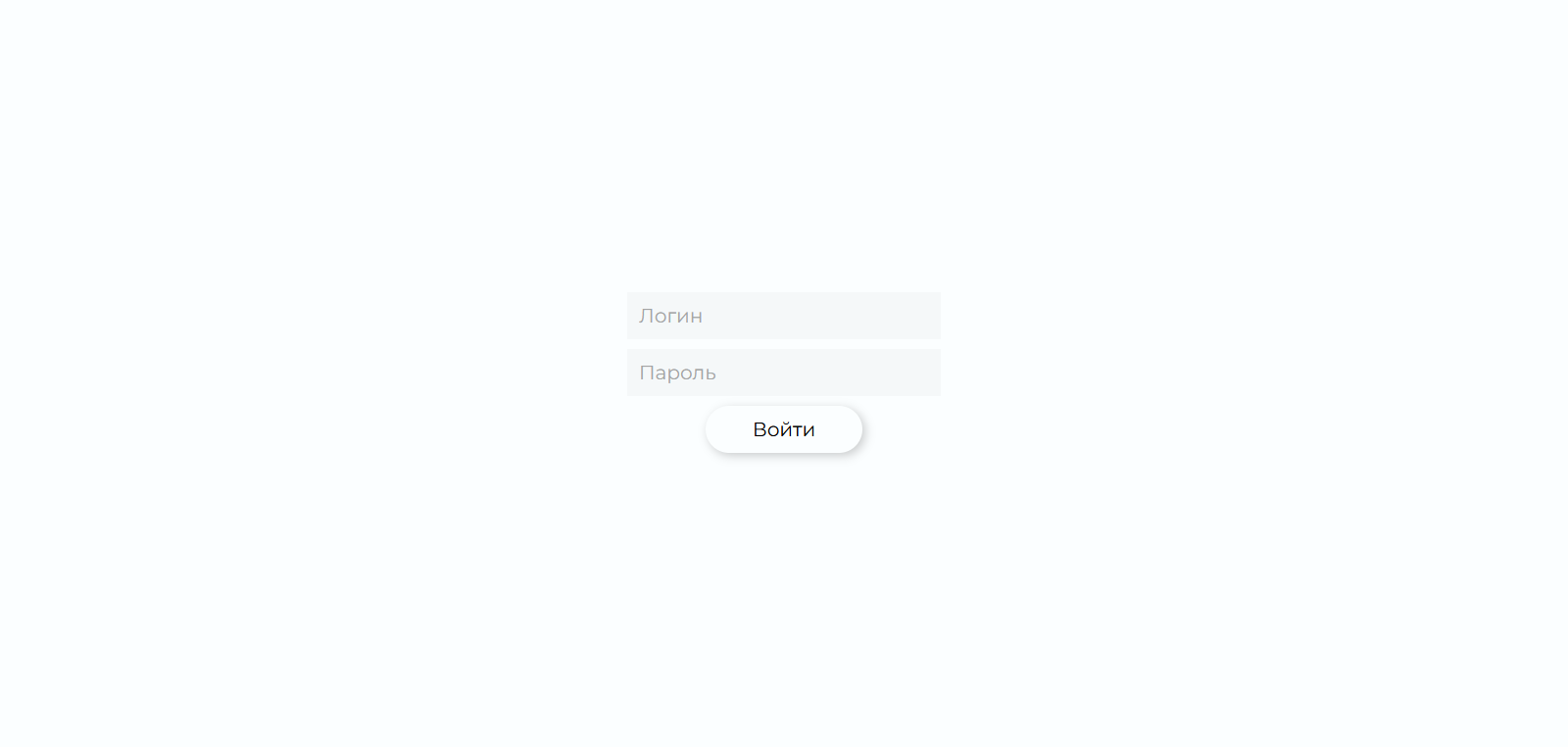


Рисунок 10 – страница авторизации в админ-панель

Все пользователи, которые имеют доступ к админ-панели расписаны в базе данных в таблице admins. Если требуется добавить нового пользователя следует добавить соответствующие значения в данную таблицу. При успешной авторизации пользователь попадёт на страницу оставленных заявок клиентов для последующего ознакомления с ними.

Страница заявок представлена на рисунке 11.

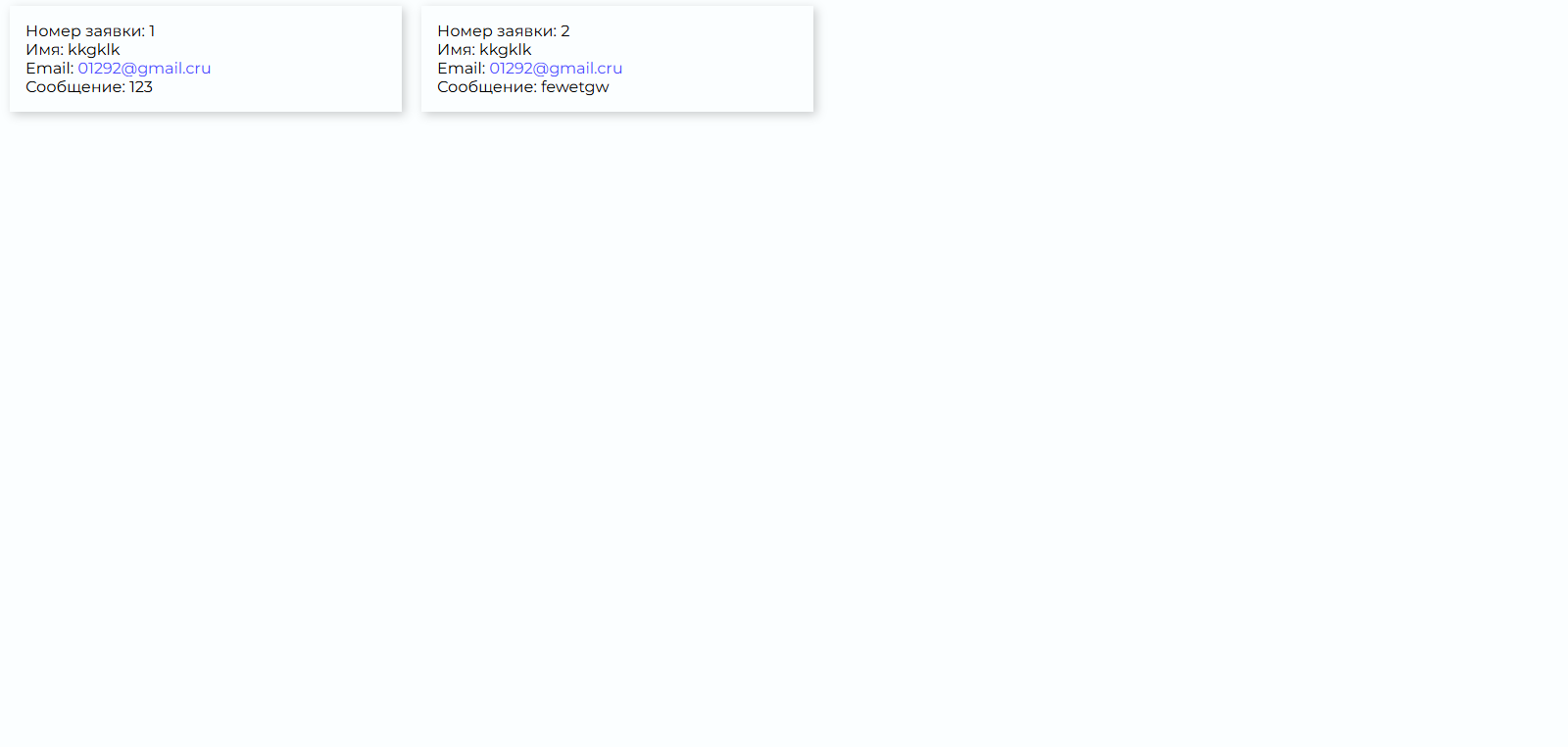


Рисунок 11 – страница оставленных заявок (админ-панель)

# **Заключение**

В ходе данной работы были выполненный все поставленные цели. Было разработано веб-приложение для веб-студии с интуитивно понятным интерфейсом, позволяющий быстро найти нужную информацию о веб-студии, рассчитать стоимость предпочитаемых работ и с возможностью просмотра оставленных заявок клиентов для работников веб-студии.

Разработка проводилась на языке гипертекстовой разметки HTML, на языке стилей CSS и на языках программирования JavaScript и PHP с использованием среды разработки Visual Studio Code.

Также были выполнены и другие цели:

* Созданы, настроены и подключены базы данных для веб-приложения;
* Спроектирована контекстная и детализированная диаграмма IDEF0;
* Выполнена SEO-оптимизация;
* Сделан отчёт о проделанной работе.

# **Библиографический список**

1. Джон Дакетт. HTML и CSS. Разработка и дизайн веб-сайтов.
2. С. Стефанов. JavaScript.
3. Загмайстер и Уолш. О Красоте.
4. Дженнифер Нидерст Роббинс. Web-дизайн. Справочник.
5. Джеф Раскин. Интерфейс: новые направления в проектировании компьютерных систем.
6. Ноубл, Андерсон. Flex 3. Сборник рецептов.
7. Дмитрий Котеров и Игорь Симдянов. PHP 7.
8. Шилд Уолтер. SQL: быстрое погружения.
9. Лора Томсон и Люк Веллинг. PHP & MySQL.
10. Владимир Дронов. HTML и CSS.

# **Приложения**

## **Приложение А HTML разметка и PHP код**

**index.php**

<?php require\_once "components/head.php" ?>

<title>Главная страница</title>

<link rel="stylesheet" href="styles/home.css">

</head>

<body>

<div class="wrapper">

<?php require\_once "components/header.php" ?>

<main>

<div class="homeContainerText">

<p>Мы внедряем цифровые технологии</p>

<div>

<h2>Создавайте лучшие цифровые продукты и приложения</h2>

<p>Мы помогаем вам ориентироваться в изменениях, создавать правильные продукты для ваших пользователей и повышать квалификацию ваших команд — одновременно</p>

</div>

<a href="portfolio.php">Узнать больше</a>

</div>

<div class="homeContainerImg">

<div class="imgsContainer">

<img src="public/homeImg1.jpg" alt="Home img1">

<img src="public/homeImg2.jpg" alt="Home img2">

<img src="public/homeImg3.jpg" alt="Home img3">

<img src="public/homeImg4.jpg" alt="Home img4">

</div>

</div>

</main>

</div>

</body>

</html>

**form.php**

<?php require\_once "components/head.php" ?>

<title>Форма заявки услуг</title>

<link rel="stylesheet" href="styles/form.css">

<script src="js/sendText.js" defer></script>

</head>

<body>

<div class="wrapper">

<?php require\_once "components/header.php" ?>

<main>

<div class="formContainerText">

<span>Свяжитесь с нами прямо сейчас</span>

</div>

<div class="formContainerForm">

<form>

<input type="text" name="name" maxlength="64" placeholder="Имя" autocomplete="on" required>

<input type="email" name="email" maxlength="255" placeholder="Электронная почта" autocomplete="on" required>

<textarea name="text" maxlength="300" placeholder="Сообщение" cols="30" rows="10" required></textarea>

<button>Связаться</button>

<span>Сообщение отправлено!</span>

</form>

</div>

</main>

</div>

</body>

</html>

**cost.php**

<?php require\_once "components/head.php" ?>

<title>Калькулятор стоимости</title>

<link rel="stylesheet" href="styles/cost.css">

<script src="js/calculator.js" defer></script>

</head>

<body>

<div class="wrapper">

<?php require\_once "components/header.php" ?>

<main>

<div class="calculatorContainer">

<div>

<input type="checkbox" id="hosting">

<label for="hosting">Хостинг <span>(15 000 ₽)</span></label>

</div>

<div>

<input type="checkbox" id="design">

<label for="design">Индивидуальный дизайн <span>(120 000 ₽)</span></label>

</div>

<div>

<input type="checkbox" id="logo">

<label for="logo">Разработка логотипа <span>(50 000 ₽)</span></label>

</div>

<div>

<input type="checkbox" id="seo">

<label for="seo">SEO-оптимизация <span>(55 000 ₽)</span></label>

</div>

<div>

<input type="checkbox" id="testing">

<label for="testing">Тестирование <span>(10 000 ₽)</span></label>

</div>

<div>

<input type="checkbox" id="copyright">

<label for="copyright">Копирайтинг <span>(30 000 ₽)</span></label>

</div>

<div>

<input type="checkbox" id="photo">

<label for="photo">Фотосъемка товаров <span>(25 000 ₽)</span></label>

</div>

<div>

<input type="checkbox" id="smm">

<label for="smm">Создание SMM-ресурсов <span>(20 000 ₽)</span></label>

</div>

<div>

<input type="checkbox" id="spa">

<label for="spa">SPA-приложение <span>(350 000 ₽)</span></label>

</div>

<h2>Итого цена: <span>0</span> ₽</h2>

</div>

</main>

</div>

</body>

</html>

## **Приложение Б CSS стили**

**style.css**

@import url('https://fonts.googleapis.com/css2?family=Montserrat:ital,wght@0,100..900;1,100..900&display=swap');

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: "Montserrat", sans-serif;

}

\*::placeholder {

color: rgba(128, 128, 128, 0.7);

}

body {

background-color: #fbfeff;

}

a {

text-decoration: none;

color: black;

}

ul {

list-style-type: none;

}

.wrapper {

padding: 30px 60px;

}

.activeLink {

pointer-events: none;

text-decoration: underline;

text-underline-offset: 10px;

text-decoration-thickness: 2px;

}

header {

display: flex;

justify-content: space-between;

flex-wrap: wrap;

align-items: center;

height: 5vh;

}

header h1 {

font-size: 28px

}

header div {

display: flex;

align-items: center;

gap: 15px;

}

header div img {

animation: spin 5s linear infinite;

}

@keyframes spin {

0% {

transform: rotate(0deg);

}

100% {

transform: rotate(360deg);

}

}

.hamburger-menu, .menu-icon {

display: none;

background-color: white;

z-index: 1;

}

.hamburger-menu {

width: 100vw;

}

.menu-icon {

font-size: 22px;

}

.hamburger-menu ul {

display: flex;

flex-direction: column;

margin-top: 10px;

}

.show-menu {

display: block;

}

@media screen and (max-width: 768px) {

.wrapper {

padding: 15px 20px;

}

header {

height: 2vh;

}

header h1 {

font-size: 22px;

}

.menu-icon {

display: block;

}

.nav-links {

display: none;

}

}

header nav ul {

display: flex;

gap: 16px;

font-weight: 500;

font-size: 15px;

}

main {

display: flex;

height: 80vh;

}

textarea {

resize: none;

}

**home.css**

.homeContainerText {

display: flex;

flex-direction: column;

justify-content: center;

gap: 30px;

width: 50%;

}

.homeContainerText p {

font-size: 22px;

}

.homeContainerText div {

display: flex;

flex-direction: column;

gap: 10px;

}

.homeContainerText h2 {

font-size: 54px;

font-weight: 500;

}

.homeContainerText a {

width: 25%;

font-size: 18px;

text-align: center;

padding: 2% 0;

box-shadow: 3px 3px 10px rgba(0, 0, 0, 0.2);

border-radius: 30px;

}

.homeContainerImg {

width: 50%;

display: flex;

align-items: center;

justify-content: center;

}

.imgsContainer {

display: flex;

flex-wrap: wrap;

width: 500px;

height: 500px;

}

@media screen and (max-width: 768px) {

main {

flex-direction: column;

}

.homeContainerText {

margin-top: 50px;

gap: 0;

width: 100%;

gap: 15px;

}

.homeContainerText p {

font-size: 18px;

}

.homeContainerText div {

gap: 5px;

}

.homeContainerText h2 {

font-size: 32px;

}

.homeContainerText a {

width: 100%;

font-size: 16px;

}

.homeContainerImg {

width: 100%;

margin-top: 15px;

}

.imgsContainer {

width: 300px;

height: 300px;

}

.imgsContainer img {

width: 150px;

height: 150px;

}

}

## **Приложение В JavaScript код**

**index.js**

const links = document.querySelectorAll("nav a");

const title = document.title;

const titleText = document.querySelector("h1");

if (titleText != null) {

titleText.innerText = title;

links.forEach((link) => {

if (link.innerText == title) {

link.classList.add("activeLink");

link.setAttribute("tabindex", "-1");

}

});

}

const menu\_icon = document.querySelector(".menu-icon");

const hamburger\_menu = document.querySelector(".hamburger-menu");

menu\_icon.addEventListener("click", () => {

hamburger\_menu.classList.toggle("show-menu");

});

**calculator.js**

const inputs = document.querySelectorAll("input");

const pricesText = document.querySelectorAll("label span");

const totalPriceText = document.querySelector("h2 span");

let prices = [];

let totalPrice = 0;

for (let i = 0; i < pricesText.length; i++) {

prices[i] = Number(pricesText[i].innerText.replace(/[^0-9,]/g,""));

}

inputs.forEach((input, index) => {

input.addEventListener("change", (e) => {

if (e.target.checked) {

totalPrice += prices[index];

} else {

totalPrice -= prices[index];

}

totalPriceText.innerText = totalPrice;

});

});

**sendText.js**

$('form').submit(function(e) {

e.preventDefault();

const data = $(this).serialize();

$.ajax({

method: 'POST',

url: './sendText.php',

data: data,

success: function() {

$('input').val('');

$('textarea').val('');

$('form span').css('color', 'black');

}

});

});