Министерство цифрового развития, связи и массовых коммуникаций

Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

|  |  |
| --- | --- |
| Кафедра | ТС и ВС |
| Допустить к защите | |
| Зав.каф., к.т.н. доцент  *Дроздова В.Г.* | |

|  |
| --- |
| **ВЫПУСКНАЯ**  **КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА** |
| Разработка серверной части сайта по обработке пользовательских данных |
|  |

Пояснительная записка

|  |  |  |  |
| --- | --- | --- | --- |
| Студент | Морозов А.Р. | /…………………/ | |
| Факультет | ИВТ | Группа | ИА-932 |
| Руководитель | Моренкова О.И. | / …………..……/ | |

Министерство цифрового развития, связи и массовых коммуникаций

Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

**КАФЕДРА**

**ТЕЛЕКОММУНИКАЦИОННЫХ СЕТЕЙ И ВЫЧИСЛИТЕЛЬНЫХ СРЕДСТВ**

**ЗАДАНИЕ**

**НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ БАКАЛАВРА**

|  |  |
| --- | --- |
| СТУДЕНТУ Морозову А.Р. | ГРУППЫ ИА-932 |

УТВЕРЖДАЮ

зав. кафедрой ТС и ВС

**/** \_\_Дроздова В.Г\_\_/

Новосибирск 2023г.

**1. Тема выпускной квалификационной работы бакалавра**

Разработка серверной части сайта по обработке пользовательских данных.

утверждена приказом СибГУТИ от «25» января 2023 г. №4/105о-23

**2. Срок сдачи студентом законченной работы** «\_20\_» июня 2023 г.

**3. Исходные данные к работе**

1. *Робсон Э., Фримен Э.* Изучаем HTML XHTML и CSS / Э. Робсон, Э. Фримен // O’Reilly. – 2-е изд., стер. – Санкт-Петербург: Питер, 2019. – 720 с.

*2. Олищук А.В*. Разработка Web-приложений на PHP 5. / А. В. Олищук // Профессиональная работа. — М.: «Вильямс», 2006. — С. 352. — ISBN 5-8459-0944-9.

|  |  |
| --- | --- |
| **4.Содержание пояснительной записки**  **(перечень подлежащих разработке вопросов)** | **Сроки выполнения**  **по разделам** |
| Введение | 25.04.2023 |
| 1 Хаpактepиcтика пpeдмeтной облаcти | 25.04.2023 |
| 1.1 Актyальноcть веб-сайтов | 26.04.2023 |
| 1.2 Актyальноcть разработки серверной части сайта | 28.04.2023 |
| 2 Обзоp и обоcнованиe выбоpа cpeдcтв peализации пpоeкта | 04.05.2023 |
| 2.1 Языки пpогpаммиpования | 05.05.2023 |
| 2.2 Cpeды pазpаботки | 05.05.2023 |
| 2.3 Обоcнованиe выбоpа cpeдcтв peализации | 05.05.2023 |
| 3 Пpактичecкая чаcть | 09.06.2023 |
| 3.1 Общая cтpyктypа веб-сайта | 09.06.2023 |
| 3.2 Создание сервера | 10.06.2023 |
| 3.3 Проектирование и создание БД | 11.06.2023 |
| 3.4 Пpогpаммный код пpоeкта | 13.06.2023 |
| 3.5 Результат работы веб-сайта | 15.06.2023 |
| Заключение | 16.06.2023 |
| Список использованных источников | 16.06.2023 |

Дата выдачи задания «27» января 2023 г.

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*подпись*

Задание принял к исполнению «27» января 2023 г.

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*подпись*

**АННОТАЦИЯ**

Выпускной квалификационной работы студента Морозов А.Р.

по теме «Разработка серверной части сайта по обработке пользовательских данных»

Объём работы - 77 страниц, на которых размещены 25рисунков. При написании работы использовалось 5 источников.

Ключевые слова: веб-приложение, веб-сайт, сервер, база данных, среда разработки, язык программирования, PHP, SQL, HTML, CSS.

Работа выполнена на кафедре ТС и ВС СибГУТИ.

(название предприятия, подразделения)

Руководитель: Доцент, кандидат технических наук Моренкова О.И.

(должность, уч.степень, звание, Фамилия Имя Отчество*)*

Целью работы являются: освоение основных принципов разработки веб-приложений и веб-сайтов, выявление характеристик предметной области, выбор средств реализации проекта. В итоге разработка серверной части сайта по обработке пользовательских данных.

Министерство цифрового развития, связи и массовых коммуникаций

Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

##### **ОТЗЫВ**

на бакалаврскую работу студентаМорозова Андрея Романовича

по теме «Разработка серверной части сайта по обработке пользовательских данных»

В данной дипломной работе был реализован проект серверной части сайта базы отдыха АлтайДача22. Сайт реализован для персональных компьютеров и ноутбуков.

Актуальность темы не вызывает сомнений. Интернет оказал большое влияние на экономику и бизнес, поэтому практически все компании имеют свои официальные сайты и представительства в социальных сетях. Хороший сайт является дополнительным ключом к успеху для любой компании

Сформулированная в теме выпускной работы задача была реализована. Во время работы Морозов А.Р. проявил себя как ответственный и самостоятельный специалист.

Считаю, что выпускная квалификационная работа заслуживает оценки «отлично», а автор, Морозов Андрей Романович, присуждения квалификации «бакалавр» по направлению 09.03.01 «Информатика и вычислительная техника», профиль «Распределенные автоматизированные системы».

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Компетенции | | Уровень сформированности  компетенций | | |
| Высокий | Средний | Низкий |
| Универсальные | УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач | х |  |  |
| УК-3 Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде | х |  |  |
| УК-4 Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(ых) языке(ах) | х |  |  |
| УК-7 Способен поддерживать должный уровень физической подготовленности для обеспечения полноценной социальной и профессиональной деятельности | х |  |  |
| УК 8 Способен создавать и поддерживать в повседневной жизни и в профессиональной деятельности безопасные условия жизнедеятельности для сохранения природной среды, обеспечения устойчивого развития общества, в том числе при угрозе и возникновении чрезвычайных ситуаций и военных конфликтов | х |  |  |
| УК-9 Способен принимать обоснованные экономические решения в различных областях жизнедеятельности |  | х |  |
| УК-10 Способен формировать нетерпимое отношение к коррупционному поведению | х |  |  |
| Общепрофессиональные | ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решения задач профессиональной деятельности | х |  |  |
| ОПК-4 Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью |  | х |  |
| ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | х |  |  |
| ОПК-6 Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием |  | х |  |
| ОПК-7 Способен участвовать в настройке и наладке программно-аппаратных комплексов | х |  |  |
| ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения | х |  |  |
| Профессиональные | ПК-4 Способен разрабатывать компоненты системных программных продуктов | х |  |  |
| ПК-32 Способен осуществлять администрирование процесса контроля производительности сетевых устройств и программного обеспечения |  | х |  |
| ПК-33 Способен проводить концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности |  | х |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Работа имеет практическую ценность | х | Тема предложена предприятием |  |
| Работа внедрена |  | Тема предложена студентом | х |
| Рекомендую работу к внедрению | х | Тема является фундаментальной |  |
| Рекомендую работу к опубликованию |  | Рекомендую студента в магистратуру | х |
| Работа выполнена с применением ЭВМ | х | Рекомендую студента в аспирантуру |  |
|  |  |  |  |

Проверка по программе «Антиплагиат» показала 81.62% оригинальности текста.

Руководитель бакалаврской работы

к.т.н., доцент каф ТС и ВС Моренкова Ольга Ильинична

Оглавление

[ВВЕДЕНИЕ 7](#_Toc138013814)

[1 ХАРАКТЕРИСТИКА ПРЕДМЕТНОЙ ОБЛАСТИ 9](#_Toc138013815)

[1.1 Актуальность веб-сайтов 9](#_Toc138013816)

[2 ОБЗОР И ОБОСНОВАНИЕ ВЫБОРА СРЕДСТВ РЕАЛИЗАЦИИ ПРОЕКТА 10](#_Toc138013817)

[2.1 Языки программирования 10](#_Toc138013819)

[2.1.1 Язык программирования PHP 10](#_Toc138013820)

[2.1.2 Язык гипертекстовой разметки HTML 12](#_Toc138013821)

[2.2.3 Язык программирования SQL 12](#_Toc138013822)

[2.2.4 Язык таблицы стилей CSS 14](#_Toc138013823)

[2.2 Среды разработки 14](#_Toc138013824)

[2.2.1 Visual Studio Code 14](#_Toc138013825)

[2.2.2 phpMyAdmin 14](#_Toc138013826)

[2.2.3 Open Server Panel 14](#_Toc138013827)

[2.3 Обоснование выбора средств реализации 14](#_Toc138013828)

[2.3.1 Обоснование выбора языка программирования 14](#_Toc138013829)

[2.3.2 Обоснование выбора среды разработки 14](#_Toc138013830)

[3 ПРАКТИЧЕСКАЯ ЧАСТЬ 14](#_Toc138013831)

[3.1 Общая структура серверной части сайта 14](#_Toc138013832)

[3.2 Создание сервера 15](#_Toc138013833)

[3.3 Проектирование и создание БД 18](#_Toc138013834)

[3.4 Программный код проекта 20](#_Toc138013835)

[3.5 Результат работы сайта 58](#_Toc138013836)

[3.5.1 Страница авторизации 58](#_Toc138013837)

[3.5.2 Страница регистрации 59](#_Toc138013838)

[3.5.3 Страница профиля 62](#_Toc138013839)

[3.5.4 Страница изменения аватара 65](#_Toc138013840)

[3.5.5 Страница бронирований (профиль) 65](#_Toc138013841)

[3.5.6 Страница (календарь) бронирования 66](#_Toc138013842)

[3.5.7 Страница (форма) бронирования 67](#_Toc138013843)

[ЗАКЛЮЧЕНИЕ 68](#_Toc138013844)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 69](#_Toc138013845)

# ВВЕДЕНИЕ

Разработка веб-сайта или веб-приложения является процессом, включающим веб-дизайн, вёрстку страниц, программирование на стороне клиента и сервера, а также настройку веб-сервера. С самого момента запуска первой веб-страницы в 1990 году, интернет стал неотъемлемой частью жизни многих людей. Он позволяет зарабатывать и тратить деньги, знакомиться с людьми и смотреть фильмы, проходить онлайн-курсы и исследовать теории заговора.

За последние тридцать лет веб-разработка продвинулась от простых страниц до многофункциональных веб-приложений. Интернет оказал большое влияние на экономику и бизнес, и поэтому практически все компании имеют свои официальные сайты и представительства в социальных сетях. Хороший сайт является ключевым для любой компании, чтобы привлечь новых клиентов, предложить свою продукцию и заработать прибыль. Но если сайт неудобен для клиентов, долго загружается или не содержит необходимую информацию, он не сможет привлечь потенциальных потребителей и не будет эффективным инструментом для бизнеса. Цель данной работы заключается в разработке веб-сайта компании по обработке пользовательских данных.

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Осуществить теоретический анализ научной и методической литературы по актуальности веб-сайтов;
2. Произвести анализ средств реализации проекта;
3. Освоить основные компоненты и средства разработки на языке PHP текстового редактора Visual Studio Code;
4. Спроектировать логику веб-сайта и порядок его взаимодействия с пользователем;
5. Разработать веб-сайт, который позволит пользователю легко ориентироваться и пользоваться системами авторизации и бронирования.

# ХАРАКТЕРИСТИКА ПРЕДМЕТНОЙ ОБЛАСТИ

## Актуальность веб-сайтов

В наше время, когда информационные технологии развиваются стремительно, интернет-пространство становится все более популярным и востребованным. Сотни тысяч серверов, содержащих различные данные, помогают увеличивать популярность и улучшать доступ к различным ресурсам. Веб-сайты создаются в настоящее время в основном для работы в бизнесе и имеют разнообразные цели. Тем не менее, для успешной и эффективной рекламы товаров и услуг, иметь свой сайт является просто необходимостью.

Создать сайт можно как для своих личных нужд, так и с целью развития бизнеса. Онлайн продвижение коммерческой деятельности и бизнеса в целом является главным залогом успеха. Клиентам гораздо удобнее сотрудничать с компаниями, имеющими свой сайт. Более того, посещение магазина или офиса уже не является обязательным. Пользователям гораздо больше нравится оформлять заказы из дома через интернет. При этом, ресурс предоставляет всю необходимую информацию о компании и ее продуктах.

Сайт может быть создан в качестве блога, интернет-магазина, информационного портала или онлайн-энциклопедии. Любой выбранный формат сайта имеет право на существование. Нельзя недооценивать мощь интернет-пространства. Современное общество занято компьютерной эрой и использованием гаджетов, поэтому интерес к интернет-порталам очень высок.

Грамотное наполнение сайта и повышение релевантности для поисковых систем – это основные условия для успешного сайта. Визуальное оформление играет тоже важную роль. Необходимо использовать инструменты работы с платформой так, чтобы интерфейс оставался привлекательным, меню удобным, а загрузка страниц была быстрой.

Одна из проблем русского малого и среднего бизнеса состоит в том, что он не может себя правильно рекламировать. Например, у многих малых предприятий нет собственного сайта, хотя это довольно дешёвый и в тоже время действенный способ самопиара. Так, например, множество малых баз отдыха в Алтайском крае не имеют свои собственные сайты, возможность с ними связаться имеется только по телефону, если он у вас появится или на сайтах со списком этих самых баз, что обычно очень неудобны. Возможно, благодаря созданию такой важной вещи, как сайт компании туристический бизнес Алтая заиграет новыми красками и не будет ничем отличаться от курортов Турции, Греции и Египта.

* 1. Актуальность разработки серверной части сайта

Разработка серверной части сайта является одним из важнейших этапов создания любого веб-проекта. Это отвечает за функциональность сайта, его безопасность и многое другое.

Актуальность разработки серверной части сайта заключается в том, что сегодня сложно представить себе сайт, который бы не имел серверной части. Большинство сайтов работают на основе серверной технологии, позволяющей обеспечивать высокую скорость работы, надежность, безопасность и функциональность сайта.

Разработчики сайтов используют серверную часть для хранения данных и обработки запросов пользователей. Она позволяет обрабатывать большое количество информации, решать сложные задачи и интегрироваться с другими системами.

Важным аспектом разработки серверной части является безопасность сайта. Серверная часть отвечает за защиту данных пользователей, а также за то, чтобы злоумышленники не могли взломать сайт и получить доступ к конфиденциальной информации.

Таким образом, разработка серверной части сайта имеет большую актуальность в современном мире. Это позволяет создавать высококачественные, надежные и безопасные веб-проекты, которые способны удовлетворить потребности пользователей и поддерживать конкурентоспособность бизнеса.

# 2 ОБЗОР И ОБОСНОВАНИЕ ВЫБОРА СРЕДСТВ РЕАЛИЗАЦИИ ПРОЕКТА



## Языки программирования

### Язык программирования PHP

PHP - это язык программирования для создания веб-приложений, который используется для разработки динамических сайтов. Он поддерживается большинством хостинг-провайдеров и является одним из самых популярных инструментов для этой цели (занимает 3 место в рейтинге Tiobe). Некоторые известные проекты, использующие PHP, включают в себя Википедию, Вконтакте, Facebook и другие.

В 1994 году датский программист Лердорф написал скрипты на Perl/CGI, которые собирали статистику о посетителях его онлайн-резюме, и назвал их Personal Home Page. Когда функциональности Perl стало недостаточно, Лердорф написал интерпретатор шаблонов PHP/FI на языке C. Вторая версия языка была выпущена после длительного тестирования в 1997 году.

PHP 3.0 была переписана израильскими программистами Сураски и Гутманс годом позже. За следующие 6 лет группой энтузиастов в язык были внесены многочисленные изменения. На сегодняшний день последняя версия языка PHP 5.0. PHP 6.0 находится в разработке с 2006 года.

Использование

PHP - скриптовый язык, который получил огромную популярность за счет своей высокой скорости обработки, простоты, кроссплатформенности и широкого распространения исходного кода под собственной лицензией. Для создания и продвижения сайтов используется множество встроенных возможностей:

• автоматическое извлечение переменных окружения сервера, а также параметров GET и POST в заранее определенные массивы;

• взаимодействие с различными системами управления базами данных, такими как SQLite, MySQL, DBX, Microsoft SQL Server, dBase, DB+++, и многими другими;

• автоматическая отправка HTTP-заголовков для упрощения поисковой оптимизации сайта;

• загрузка файлов на сервер и их обработка;

• работа с удаленными и локальными файлами, сессиями, сокетами, HTTP-авторизацией, cookies, XForms.

Также различные возможности PHP используются для создания пользовательских интерфейсов и скриптов для сбора статистики в процессе продвижения сайтов.

Особенности языка PHP связаны с синтаксисом, схожим с языком Си. Из языка Perl заимствованы некоторые элементы, например, ассоциативные массивы. При работе с программой в PHP можно начинать с оператора без необходимости описания переменных. Код языка выполняется внутри ограничителей <?php и ?>. Данные, выводимые вне этих ограничителей, по умолчанию изображаются в виде стандартной разметки HTML документа. В PHP имя переменной должно начинаться с символа $ и оно должно быть чувствительным к регистру, как и имена классов, функций и констант. Переменные могут обрабатываться, как при заключении в двойные кавычки или апострофы в строках, так и создаваться с помощью оператора <<<. Для обеспечения кроссплатформенности приложения скрипты обрабатываются интерпретатором в требуемом порядке.

### Язык гипертекстовой разметки HTML

Язык разметки веб-страниц называется HTML. Браузер интерпретирует результат и показывает пользователю текст в заданном виде разработчика на экране компьютера или смартфона. В прошлом первые сайты создавались практически только на коде HTML с небольшим использованием скриптов на JavaScript. Одна из особенностей HTML заключается в преобразовании простых команд в визуальные объекты. Например, тег <img> используется для отображения изображений. Для этого необходимо указать ссылку на файл. Изображение обычно находится на удаленном сервере, на котором располагается сам ресурс или внешний сервис.

Существует ряд основных элементов HTML, такие, как форматирование текста (курсив, жирный шрифт, размер шрифта, нумерованные/маркированные списки), текстовые блоки (заголовки разных уровней, абзацы, перенос строк), таблицы (любое количество строк, столбцов, заданная высота и ширина, заголовки), вставка объектов (изображения, звук, текст, видео и т.д.), гиперссылки (на файлы изображений, прайс-листы, страницы или анкоры в тексте, открытие документа в текущем или новом окне).

Также есть базовые возможности для создания форм, например, ввод текстовой информации или выбор пункта из списка. Более сложные объекты, если необходимы, создаются на JavaScript или PHP, так как эти языки имеют больший функционал. HTML часто используется копирайтерами и контент-менеджерами для публикации контента.

### 2.2.3 Язык программирования SQL

Structured Query Language (SQL) - это язык запросов, который используется для работы с базами данных, организованными в специальном формате. Основные задачи SQL заключаются в составлении запросов для поиска информации, необходимой для конкретных целей в большом объеме данных, и ее дальнейшей сортировке, структурировании и представлении в простой и понятной форме.

Для понимания необходимости языка SQL можно представить, что женщина ищет летнее желтое платье в интернет-магазине и хочет уложиться в 5 тысяч рублей. В магазине есть 10 тысяч платьев. Просто перебирать их через каталог займет много времени, но с помощью фильтров по категории, сезону, цвету и цене можно найти нужные модели. Эти фильтры работают благодаря языку запросов SQL.

Первый прототип языка SQL был представлен в 1979 году компанией-разработчиком Oracle. Сначала он использовался только для извлечения необходимых данных, как фильтры в таблицах Excel. С течением времени он стал более сложным и сейчас используется как один из основных инструментов для обработки данных. С помощью SQL можно создавать и хранить данные в виде таблиц, изменять их содержимое и структуру, объединять данные и выполнять расчеты, а также обеспечивать защиту и распределять доступ к данным.

Допустим, в компании работает 500 работников, среди которых 100 человек занимаются продажами и используют CRM для регистрации информации о клиентах: новые контракты, суммы, скидки и контакты. Также, в компании имеются 15 IT-специалистов, которые обслуживают, обновляют и модифицируют структуру CRM при необходимости. 20 работников бухгалтерии регулярно выгружают данные об оплате, выставленные счета и подписываемые акты из системы. С помощью SQL можно дать им доступ только к частям CRM, необходимым для работы, что снижает риск повреждения элементов кода или ценных данных.

Особенности языка SQL включают:

Язык запросов, а не программа. Он используется в сочетании с Python, JavaScript или C++, но только для баз данных. Создание полноценного сайта или приложения на нем невозможно.

Ясная и понятная структура, что делает язык SQL довольно простым для начинающих, когда дело приходит к работе с данными.

Универсальность. Существуют единые стандарты построения запросов для всех баз данных и браузеров, позволяющие обрабатывать и анализировать огромные объемы информации.

Совместный доступ. SQL позволяет создавать интерактивные запросы, поэтому нужные данные можно получить в режиме онлайн и принять решения на основе этих данных.

Управление доступом. С помощью SQL можно выделять, закрывать или ограничивать доступ к данным различным группам пользователей, а также определять набор функций: чтение, изменение, создание, удаление, копирование. Это сохраняет базы данных от хакерских атак и некорректных изменений.

### 2.2.4 Язык таблицы стилей CSS

CSS - это язык, который описывает внешний вид HTML-документа и является одной из основных технологий в интернете. HTML и CSS работают вместе, где HTML структурирует документ и упорядочивает информацию, а CSS служит для придания документу стиля. С помощью CSS можно написать правила оформления для разных элементов документа. Каждое правило состоит из селектора и блока объявлений, где селектор определяет, на какие части документа применяется правило, а блок объявлений содержит одно или более объявлений, разделенных точкой с запятой. Эти правила пишутся в каскадной таблице стилей, которая взаимодействует с браузером для придания документу желаемого вида. Короче говоря, CSS необходим для создания красивых и индивидуальных веб-страниц.

## Среды разработки

### Visual Studio Code

VS Code - это редактор кода для разных языков программирования, который отличается относительно небольшим размером, гибкостью и удобством использования. Средство разработки позволяет писать, форматировать и редактировать код на разных языках программирования.

Не следует путать VS Code с Visual Studio, поскольку эта среда разработки является IDE, мощной и масштабной, но одновременно тяжеловесной. Название похожи, потому что обе созданы одним разработчиком, однако, это два разных продукта. VS Code является легким и бесплатным, в отличие от большинства версий Visual Studio. Редактор кода доступен для использования на всех популярных операционных системах, таких как Windows, Linux и macOS.

Программисты, работающие с различными языками программирования, используют VS Code. Например, веб-разработчики используют HTML/CSS, JavaScript, PHP, но редактор также поддерживает множество других языков, включая Python, Go, Ruby, C#, TypeScript и многие другие. Он также работает с расширениями и фреймворками для популярных языков, например, React JS и Vue.js, а также языками стилей, такими как SCSS и LESS, которые дополняют CSS. Редактор продвинутый и удобный, и может применяться как для начинающих разработчиков, так и для экспертов.

Разработчик в теории может использовать VS Code практически на любом из современных языков, но на практике он чаще всего применяется там, где не требуются мощности полноценных сред разработки. С помощью VS Code легко создать, форматировать и редактировать код на разных языках. Редактор подсвечивает синтаксис кода и помогает автоматически исправлять ошибки, а также предоставляет возможности для отладки и запуска кода на некоторых языках.

Кроме того, VS Code легко расширяется: новые функции могут быть легко добавлены, загрузив соответствующее расширение из официального каталога. Дополнения также распространяются бесплатно.

Рабочая среда VS Code состоит из документа, в котором программист пишет свой код, и различных меню, которые предоставляют доступ к различным функциям и возможностям. Боковое меню позволяет контролировать версии, устанавливать дополнения и запускать и отлаживать код. В нижней части страницы находится консоль, где можно просмотреть результаты выполнения кода, а также различные сообщения об ошибках и сведения о сборке. Консоль также поддерживает ввод команд, для которых есть соответствующие горячие клавиши.

VS Code поддерживает открытие нескольких файлов в разных вкладках и разделение экрана на несколько частей для параллельной работы с несколькими файлами, как в операционной системе, когда открыто несколько окон программ.

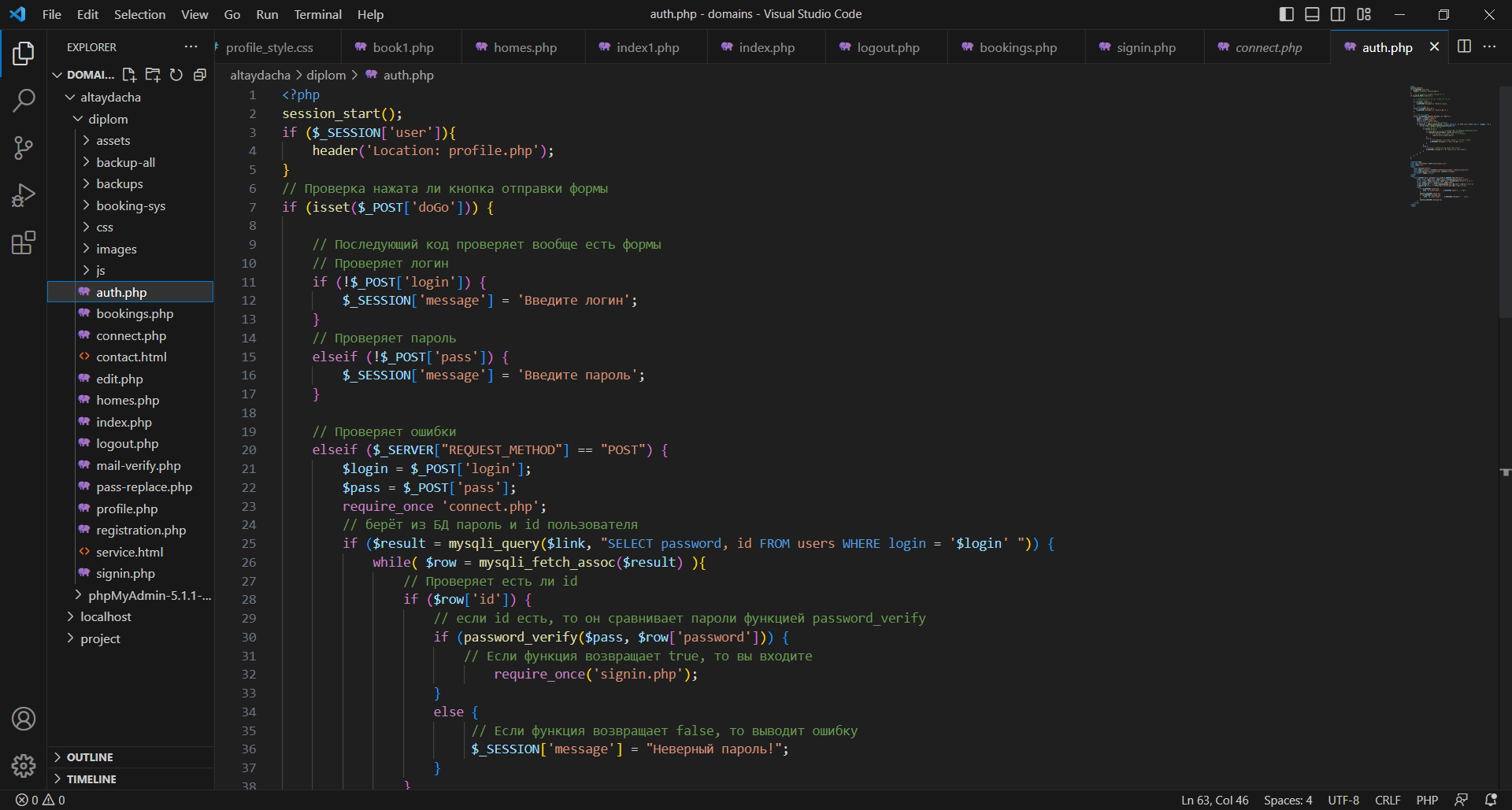


Рис. 2.1 – Интерфейс Visual Studio Code

### phpMyAdmin

PhpMyAdmin – это веб-приложение, которое позволяет администрировать базы данных через удобный графический интерфейс. Его устанавливают на сервер, на котором располагается сайт, а затем запускают в браузере для удаленного управления СУБД. PhpMyAdmin предназначен для работы с проектами, основанными на веб-сервере Apache, языке PHP и системах MySQL или MariaDB. Установка и настройка приложения немного отличаются, в зависимости от операционной системы. PhpMyAdmin обновляется и развивается, выходят новые версии, хотя панель управления остается почти неизменной. PhpMyAdmin распространяется бесплатно и имеет открытый исходный код. Это в сочетании с многофункциональностью и удобством интерфейса является основой его популярности.

Основная цель использования приложения – упрощение процесса взаимодействия с БД за счет интуитивно понятного интерфейса. Благодаря ему, не нужно знать специальных команд, чтобы обращаться к данным. Вместо написания SQL-запросов достаточно сделать несколько кликов, и приложение выполнит работу за вас.

PhpMyAdmin позволяет создавать, изменять, удалять и загружать/скачивать таблицы с информацией на сервере. В приложении можно просматривать содержимое баз, добавлять новые, изменять или вносить в них данные, импортировать/экспортировать целые базы или отдельные таблицы. Если же нужно выполнить SQL-запрос, в приложении есть командная строка для их ввода. С помощью приложения возможно администрировать сразу несколько серверов. PhpMyAdmin поддерживает многопользовательский режим и дает возможность контролировать уровни доступа для подключенных пользователей.

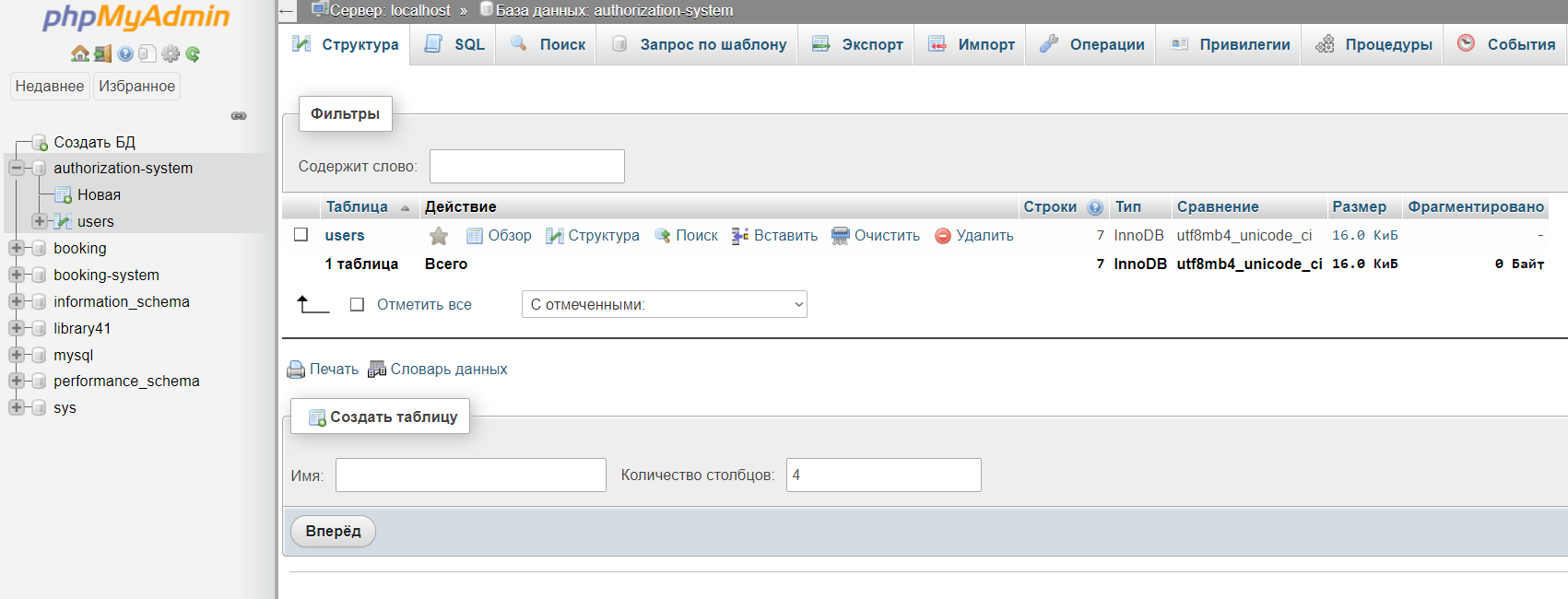


Рис. 2.2 – Интерфейс phpMyAdmin

### Open Server Panel

Open Server - это портативный локальный сервер WAMP/WNMP с управляющей программой и большим выбором компонентов. Данный пакет не является любительской сборкой, а является полноценным профессиональным инструментом, созданным специально для разработчиков с учетом их рекомендаций и пожеланий.

Open Server предоставляет выбор между двумя HTTP-серверами, различными версиями PHP и модулей СУБД для отладки скриптов в различных средах. Важно отметить, что Open Server является полностью портативным сервером без системных сервисов и мусора в реестре и system32. Его можно запускать на любой рабочей или домашней машине без опасения о нестабильной работе.

В случае отсутствия необходимых системных компонентов на компьютере, Open Server автоматически устанавливает их самостоятельно при первом запуске сервера на компьютере.

Open Server имеет множество опций для управления сервером и доменами, такие как поддержка SSL и кириллических доменов, создание локальных поддоменов, поддержка алиасов или других доменных указателей, быстрый доступ к шаблонам конфигурации модулей и многое другое.

Программа регулярно обновляется с учетом пожеланий пользователей. Open Server - это не просто стандартный сервер, он обладает специфическими возможностями, которые делают его уникальным и особенным.

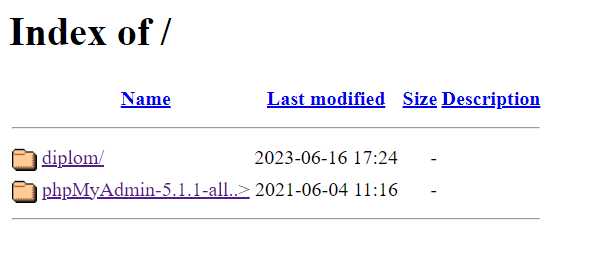


Рис. 2.3 – Интерфейс Open Server Panel

## Обоснование выбора средств реализации

### Обоснование выбора языка программирования

Одной из самых привлекательных черт PHP для разработчиков и ИТ-менеджеров является открытый доступ. Нет необходимости выплачивать лицензионные сборы. Кроме явной экономии, это также означает большое и развитое сообщество, которое постоянно улучшает и дополняет функциональность. Сотни готовых решений на PHP уже доказали свою жизнеспособность, так что можно использовать их, не изобретая велосипедов.

Однако, стоит заметить, что это основное преимущество PHP и одновременно его главная угроза для проектов. Легкость и доступность существенно снижают порог вхождения в профессию, что неизбежно приводит к снижению качества. Поэтому правильный выбор профессиональной команды является критически важным для PHP-проектов.

PHP - язык, частично ориентированный на объекты. Это обеспечивает возможность повторного использования кода, что экономит время и усилия в процессе разработки. Существует множество фреймворков PHP, таких как Symfony, CodeIgniter, Laravel, Joomla, WordPress и другие. Каждый из них обладает своей функциональностью и заточен под определенные задачи.

PHP был оптимизирован для быстрого создания веб-приложений. Он имеет встроенные функции, такие как использование запросов GET и POST, работу с HTML и URL. Для бизнеса это означает, что время и деньги, затраченные на разработку, могут быть сокращены, а инвестиции начинают окупаться быстрее.

Одним из значимых преимуществ PHP является его гибкость и универсальность. Он совместим со всеми основными платформами - от Windows, Unix и Linux до MacOS. PHP поддерживает большинство серверов, в том числе Apache, Microsoft IIS, Netscape, iPlanet, Caudium, Xitami и Tornado, а также более 20 баз данных, включая MySQL, MongoDB, PostgreSQL и другие. Именно поэтому PHP часто используется для создания кросс-платформенных приложений, что позволяет бизнесу максимально задействовать существующую инфраструктуру с минимальными затратами.

Хотя традиционно рекомендуется использовать PHP в связке с HTML, он легко интегрируется с JavaScript, WML, XML и другими языками программирования. У браузера также не возникает проблем, потому что все сценарии компилируются на стороне сервера.

Любой проект в какой-то момент времени подвергается изменениям в функциональной составляющей. Благодаря динамическому характеру PHP, внесение изменений возможно на любом этапе разработки без потерь во времени. Благодаря хорошей читабельности, изменение и модификация кода на PHP не составляет труда. Как следствие, меньше проблем с поддержкой и обновлением. Проекты на PHP легко адаптируются под новые приложения, которые появляются на рынке, и новые бизнес-требования. Благодаря вкладу открытого сообщества, новый функционал, который соответствует потребностям бизнеса, появляется регулярно и ничего не стоит. Поскольку код легко читается, поддержка сервисов на PHP может осуществляться любой командой, а не обязательно той, которая создавала проект.

Универсальность PHP делает его популярным среди хостинг-провайдеров. Любой надежный провайдер предоставляет поддержку PHP и хостинговые услуги для сайтов на нем. Как правило, хостинг-пакеты услуг предоставляют поддержку PHP без дополнительной оплаты, включая бесплатные тарифные планы, а также дешевые с неограниченным объемом и бесплатным доменным именем.

Рост популярности PHP и его фреймворков в последние годы был значительным, и эта тенденция продолжается. Основные бизнес-преимущества PHP заключаются в его экономичности, масштабируемости, простоте, совместимости и удобстве. PHP сам по себе или в комбинации с другими технологиями подходит для создания ИТ-проектов разных типов – от простых сайтов до CRM-решений и крупных онлайн-магазинов.

### Обоснование выбора среды разработки

Одним из самых распространенных среди программистов редакторов исходного кода является Visual Studio Code. Он сочетает в себе скорость, легкость и мощь, благодаря чему был разработан компанией Microsoft в качестве кросс-платформенного редактора кода для написания веб- и облачных приложений. Объявление о его создании было сделано на конференции Build 2015 в Сан-Франциско 29 апреля 2015 года. Это привело к свободному доступу к исходному коду на GitHub и выпуску VSC под лицензией MIT 18 ноября 2015 года. Он был выпущен в Интернете 14 апреля 2016 года.

Основная причина, по которой VSC стал настолько популярным, заключается в том, что он обладает всеми необходимыми функциями, ожидаемыми программистами от редактора кода, а также дополнительными и полезными опциями. Он прост в использовании, быстр и имеет открытый исходный код, а также другие удобные функции, что предоставляет ему дополнительное преимущество перед другими редакторами. Вот 10 причин, почему он стал популярен среди сообщества разработчиков, по мнению автора текста.

Также из плюсов:

1. Кроссплатформенность
2. Поддержка множества языков программирования
3. Изменение языка для выбранного файла
4. Языковая документация
5. Отладка
6. Встроенная интеграция Git
7. Функция управления кодом
8. Кастомизация

# 3 ПРАКТИЧЕСКАЯ ЧАСТЬ

Разработка серверной части веб-сайта происходила в три этапа:

1. Разработка структуры и определение функций сайта;

2. Создание сервера;

3. Проектирование и создание баз данных;

4. Тестирование всех функций и регистрация результатов их работы.

## 3.1 Общая структура серверной части сайта

При разработке серверной части сайта необходимо заранее продумать его структуру, функционал, что определит объем работы, временные затраты, а также количество страниц и необходимых скриптов.

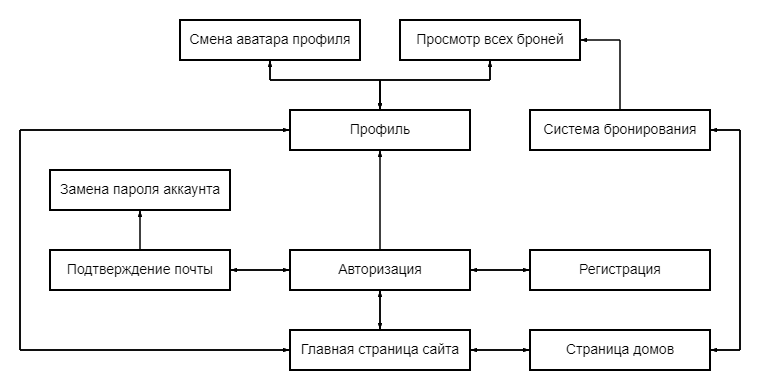


Рис. 3.1 - Структура приложения.

Структурная схема позволяет наглядно представить конечную работу, что в свою очередь позволяет удобно и понятно разбить процесс разработки на несколько этапов, и выделить необходимые задачи.

## 3.2 Создание сервера

Первым делом необходима настройка «Open Server Panel». Нужно зайти в опцию «Настройки» выпадающего меню. После того как откроется окно настроек OSP, понадобится вкладка «Домены». В этом окне необходимо создать локальный домен – «altaydacha22/»:

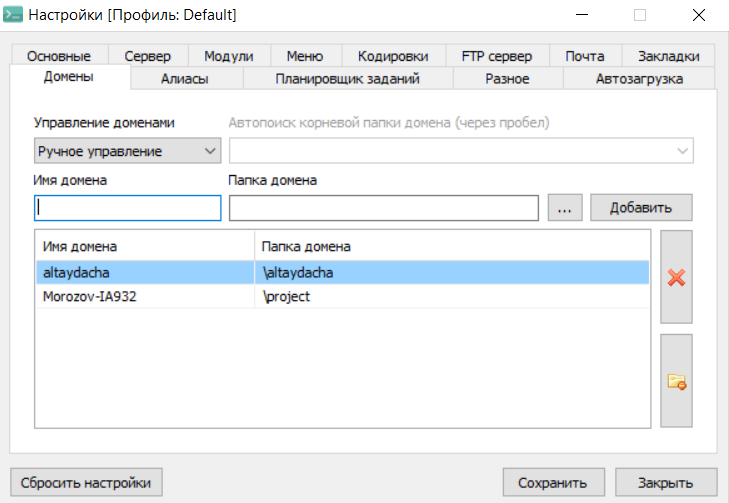


Рис. 3.2 – Настройки: Создание домена

После настройки домена нужно в созданную папку домена поместить файлы «phpMyAdmin» и в меню настроек во вкладке «Модули» выбрать MySQL нужной версии. Далее нужна проверка работоспособности созданного домена и веб-интерфейса для MySQL:

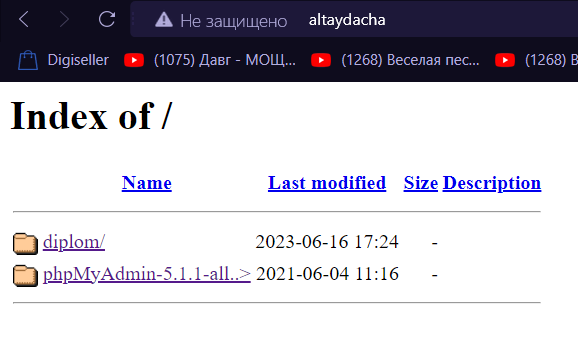


Рис. 3.3 – Работающий домен

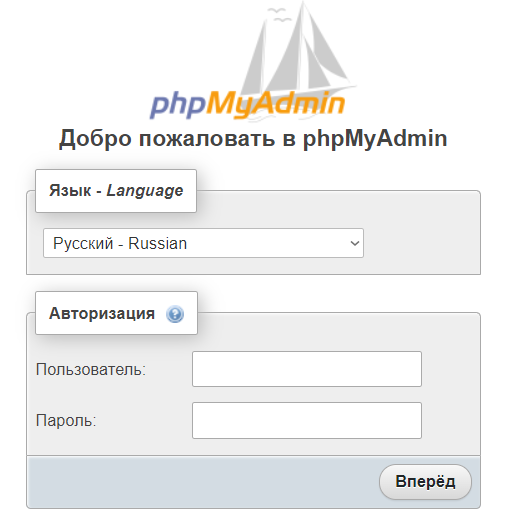


Рис. 3.4 – Окно авторизации «phpMyAdmin»

После проверки на правильность создания домена и веб-интерфейса нужно заранее подключить удаленный SMTP сервер Google, для этого необходимо попасть во вкладку «Почта»:

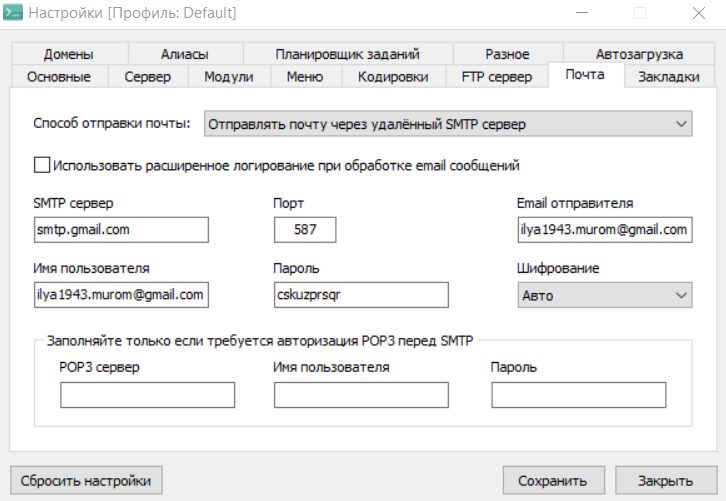


Рис. 3.5 – Настройка SMTP сервера

Убедивишись, что всё настроено и подключено, можно приступать к дальнейшим этапам работы.

## 3.3 Проектирование и создание БД

Для хранения и обработки всей пользовательской информации понадобится две Базы данных и компактные таблицы внутри. Сначала нужно спроектировать реляционную модель обеих БД:

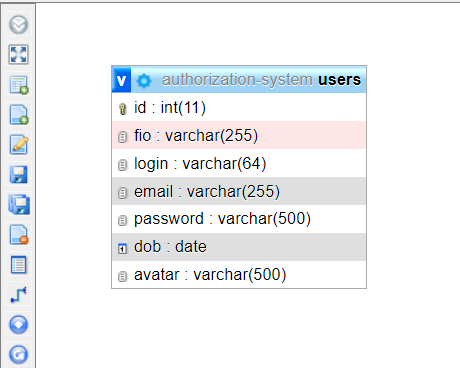


Рис. 3.6 – Реляционная модель БД authorization-system

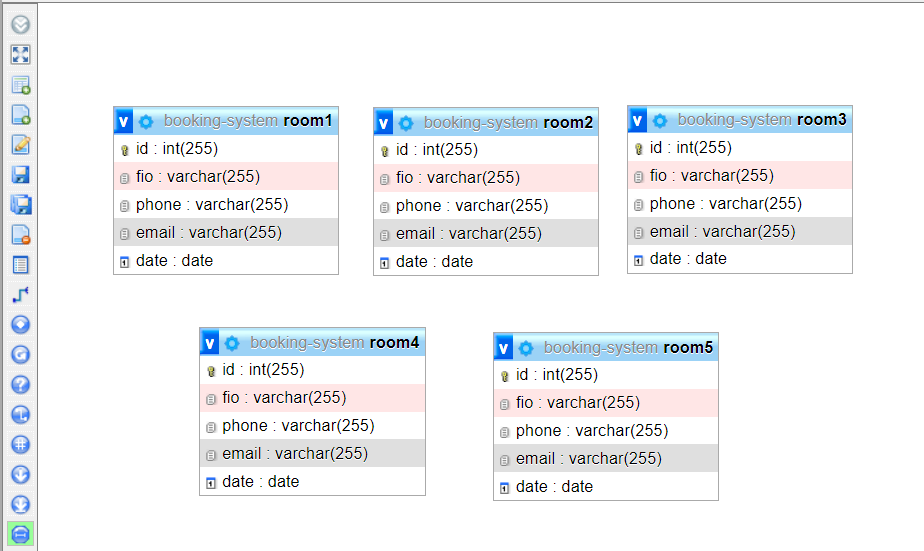


Рис. 3.7 – Реляционная модель БД booking-system

Теперь нужно воплотить реляционные модели в реальность и создать две БД и 5 таблиц в phpMyAdmin, соблюдая все типы данных. Также нельзя забыть про ключевые поля.

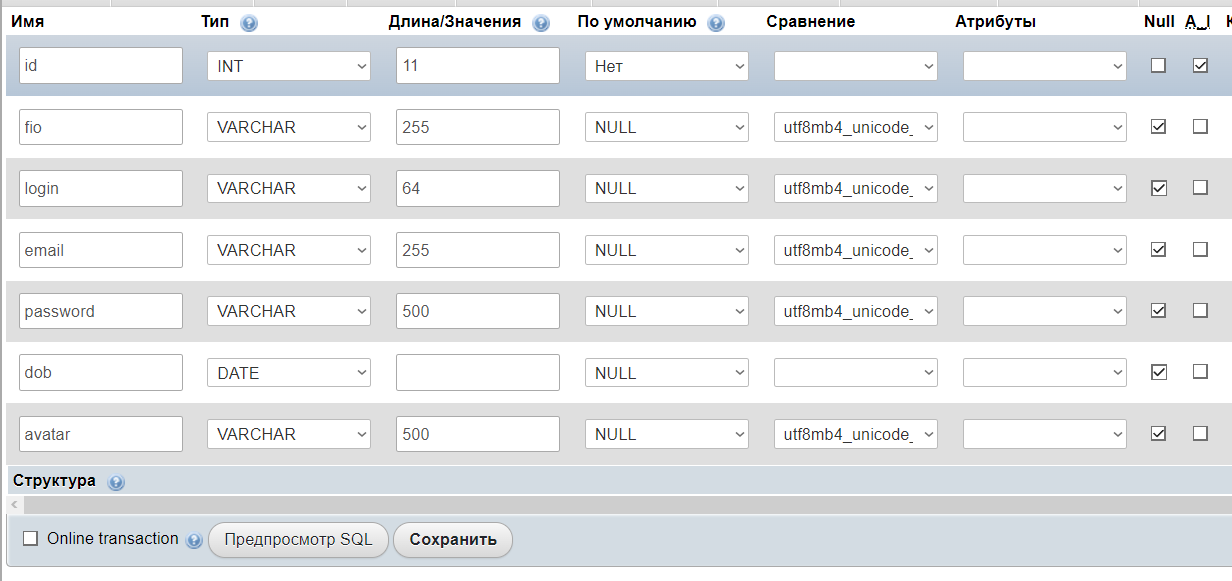


Рис. 3.8 – Создание таблицы users



Рис. 3.9 – Структура таблицы users

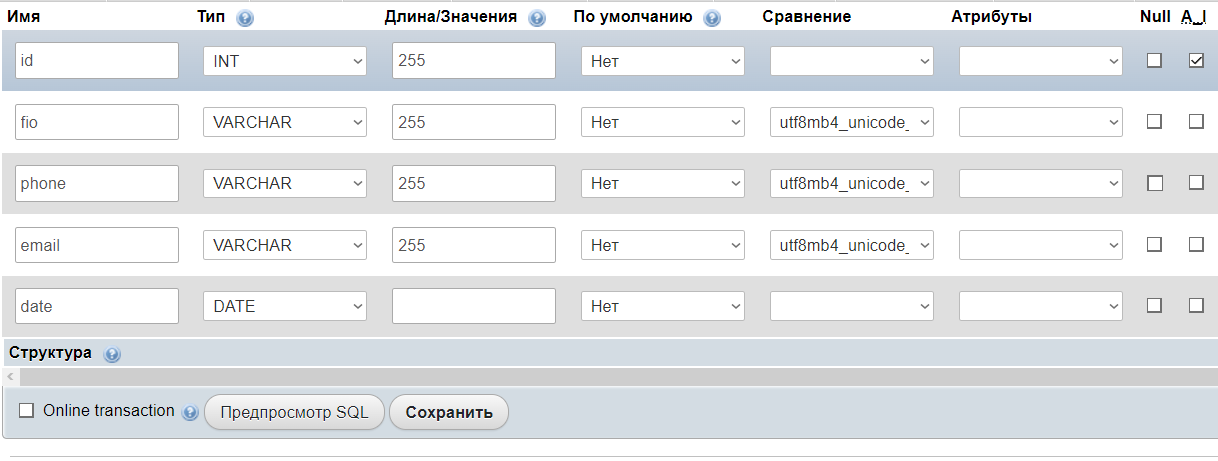


Рис. 3.10 – Создание таблицы room1

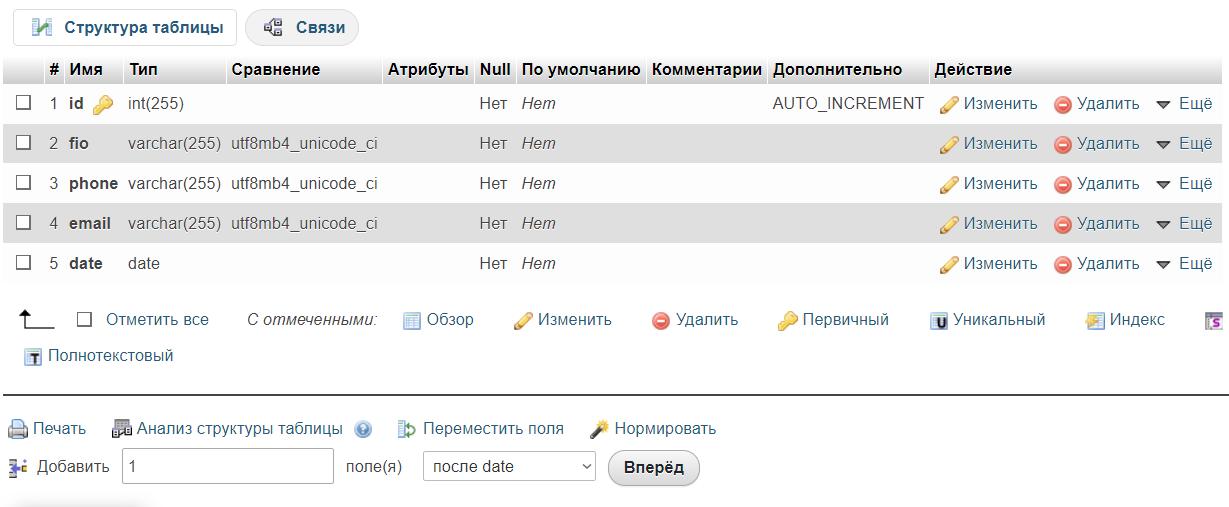


Рис. 3.11 – Структура таблицы room1

Стоит понимать, что таблицы room2, room3, room4 и room5 абсолютно идентичны структуре таблицы room1, поэтому создаются по одному принципу (см. Рис. 3.6).

## 3.4 Программный код проекта

Разработав структуру сайта, имеется четкое представление о количестве страниц. В данном проекте их будет 17:

1. Auth.php – первая страница системы авторизации, которую видит пользователь содержит 1 кнопку для инициализации скрипта, две ссылки для перехода на страницу регистрации и страницу подтверждения почты, а также форму, необходимую для сбора данных пользователя.
2. Registration.php – страница системы авторизации, содержащая форму, которую пользователь заполняет для дальнейшего занесения всей информации в базу данных и создания профиля пользователя.
3. Mail-verify.php – страница системы авторизации, которая имеет поле ввода почты пользователя, для дальнейшего использования удаленного SMTP сервера и отправки автоматического письма со ссылкой на замену пароля от профиля пользователя.
4. Pass-replace.php – страница системы авторизации, которая осуществляет замену пароля профиля пользователя.
5. Profile.php – сам профиль, а также страница системы авторизации, которая содержит функциональные кнопки для просмотра всез броней, замены аватара профиля пользователя, выхода из профиля пользователя, а также ссылку на главную страницу.
6. Edit.php - страница системы авторизации, имеющая форму для загрузки аватара профиля.
7. Bookings.php – страница системы авторизации, содержащая все брони совершенные пользователем.
8. Index1 … index5 – страницы системы бронирования, содержащие календарь через который осуществляется посуточное бронирование.
9. Book1.php … book5.php – страницы системы бронирования с формой для заполнения, на которые пользователь попадает после выбора даты в календаре.

Для начала нужно создать форму для ввода логина и пароля, а также написать скрипт сравнивающий введенный пароль в форме с паролем находящимся в базе данных. Для этого можно использовать функцию password\_verify. Для передачи данных из формы на сервер используется метод пост-запросов, который позволяет использовать глобальную переменную *$\_POST*. Эта переменная будет записывать введенные пользователем данные из соответствующих полей с указанным идентификатором. Данные полученные пост-запросом можно обрабатывать и/или хранить в БД.

Листинг 3.1 – auth.php

|  |
| --- |
| *<?php*  *session\_start();*  *if ($\_SESSION['user']){*  *header('Location: profile.php');*  *}*  *// Проверка нажата ли кнопка отправки формы*  *if (isset($\_POST['doGo'])) {*    *// Последующий код проверяет вообще есть формы*  *// Проверяет логин*  *if (!$\_POST['login']) {*  *$\_SESSION['message'] = 'Введите логин';*  *}*  *// Проверяет пароль*  *elseif (!$\_POST['pass']) {*  *$\_SESSION['message'] = 'Введите пароль';*  *}*    *// Проверяет ошибки*  *elseif ($\_SERVER["REQUEST\_METHOD"] == "POST") {*  *$login = $\_POST['login'];*  *$pass = $\_POST['pass'];*  *require\_once 'connect.php';*  *// берёт из БД пароль и id пользователя*  *if ($result = mysqli\_query($link, "SELECT password, id FROM users WHERE login = '$login' ")) {*  *while( $row = mysqli\_fetch\_assoc($result) ){*  *// Проверяет есть ли id*  *if ($row['id']) {*  *// если id есть, то он сравнивает пароли функцией password\_verify*  *if (password\_verify($pass, $row['password'])) {*  *// Если функция возвращает true, то вы входите*  *require\_once('signin.php');*  *}*  *else {*  *// Если функция возвращает false, то выводит ошибку*  *$\_SESSION['message'] = "Неверный пароль!";*  *}*  *}*  *else {*  *// Выводит ошибку если не нашли такой логин*  *$\_SESSION['message'] = "Вы ввели не верный логин!";*  *}*  *}*  *}*  *}*  *}*  *?>*  *<!DOCTYPE html>*  *<link rel="stylesheet" href="assets/main.css">*  *<html lang="ru">*  *<head>*  *<meta charset="UTF-8">*  *<meta name="viewport" content="width=device-width, initial-scale=1.0">*  *<meta http-equiv="X-UA-Compatible" content="ie=edge">*  *<title>Авторизация</title>*  *</head>*  *<body>*  *<form method="post" action="<?php echo $\_SERVER['PHP\_SELF'];?>">*  *Логин <input type="text" name="login" placeholder="Введите логин">*  *Пароль <input type="password" name="pass" placeholder="Введите Пароль">*  *<input type="submit" value="Войти" name="doGo">*  *<p>Нет аккаунта? - <a href="registration.php">Зарегистрируйтесь!</a></p>*  *<p>Забыли пароль? - <a href="mail-verify.php">Вам сюда!</a></p>*  *<?php*  *if ($\_SESSION['upset']){*  *echo '<p class="msg"> ' . $\_SESSION['upset'] . ' </p>';*  *}*  *unset($\_SESSION['upset']);*  *if ($\_SESSION['message']){*  *echo '<p class="msg"> ' . $\_SESSION['message'] . ' </p>';*  *}*  *unset($\_SESSION['message']);*  *?>*  *</form>*  *</body>*  *</html>* |

Также, сразу нужно создать main.css файл и привязать его ко всем дальнейшим формам системы авторизации.

Листинг 3.2 – main.css

|  |
| --- |
| \* {  margin: 0;  padding: 0;  box-sizing: border-box;  }  body {  height: 100vh;  display: flex;  align-items: center;  justify-content: center;  font-family: Montserrat, sans-serif;  }  a {  color: #7c9ab7;  font-weight: bold;  text-decoration: none;  }  p {  margin: 10px 0;  }  form {  display: flex;  flex-direction: column;  width: 400px;  }  input {  margin: 10px 0;  padding: 10px;  border: unset;  border-bottom: 2px solid #e3e3e3;  outline: none;  }  button {  padding: 10px;  background: #e3e3e3;  border: unset;  cursor: pointer;  }  .msg {  border: 2px solid #ffa908;  border-radius: 3px;  padding: 10px;  text-align: center;  font-weight: bold;  }  .logout {  color: #b7625c;  } |

После этого нужно создать небольшой скрипт, при обращении к которому будет осуществляться подключение к нужной БД. Для этого переходим в файл connect.php.

Листинг 3.3 – connect.php

|  |
| --- |
| <?php  $link = mysqli\_connect("localhost", "root", "", "authorization-system") or die ("Невозможно подключиться к серверу");  ?> |

Далее переходим в файл registration.php. Нужна форма регистрации и скрипт, который будет заносить всю информацию введенную пользователем в базу данных. Нельзя забыть и о хешировании пароля, ведь это обеспечит безопасность данных пользователя. Требуется проверка на идентичность введенной информации с информацией в базе данных, для избежания создания повторяющихся аккаунтов.

Листинг 3.4 – registration.php

|  |
| --- |
| *<?php*  *session\_start();*  *if ($\_SESSION['user']){*  *header('Location: profile.php');*  *}*  *// Проверяем нажата ли кнопка отправки формы*  *if (isset($\_POST['doGo'])) {*    *// Все последующие проверки, проверяют форму и выводят ошибку*  *// Проверка на совпадение паролей*  *if ($\_POST['pass'] !== $\_POST['pass\_rep']) {*  *$\_SESSION['message'] = 'Пароли не совпадают';*  *}*    *// Проверка есть ли вообще повторный пароль*  *elseif (!$\_POST['pass\_rep']) {*  *$\_SESSION['message'] = 'Введите повторный пароль!';*  *}*    *// Проверка есть ли пароль*  *elseif (!$\_POST['pass']) {*  *$\_SESSION['message'] = 'Введите пароль!';*  *}*    *// Проверка есть ли email*  *elseif (!$\_POST['email']) {*  *$\_SESSION['message'] = 'Введите почту!';*  *}*    *// Проверка есть ли логин*  *elseif (!$\_POST['login']) {*  *$\_SESSION['message'] = 'Введите логин!';*  *}*  *elseif (!$\_POST['fio']) {*  *$\_SESSION['message'] = 'Введите ФИО!';*  *}*    *// Если ошибок нет, то происходит регистрация*  *elseif ($\_SERVER["REQUEST\_METHOD"] == "POST") {*  *$fio = $\_POST['fio'];*  *$login = $\_POST['login'];*  *$email = $\_POST['email'];*  *// Пароль хешируется*  *$pass = password\_hash($\_POST['pass'], PASSWORD\_DEFAULT);*  *// Если день рождения не был указан, то будет самый последний год из доступных*  *$dob = $\_POST['year\_of\_birth'];*  *// Привязка*  *require\_once 'connect.php';*  *//mysqli\_set\_charset($link, "utf8");*  *//выборка по логину*  *$result = mysqli\_query($link, "SELECT \* FROM users WHERE login = '$login' OR email = '$email'");*  *if(mysqli\_num\_rows($result) > 0){*  *$\_SESSION['message'] = 'Такой логин или Email уже есть!';*  *}*  *else {*  *$db = mysqli\_query($link, "INSERT INTO users (fio, login, email, password, dob) VALUES ('$fio', '$login', '$email', '$pass', '$dob')");*  *$\_SESSION['upset'] = 'Регистрация прошла успешно!';*  *header('Location: auth.php');*  *}*  *}*  *}*  *?>*  *<!DOCTYPE html>*  *<link rel="stylesheet" href="assets/main.css">*  *<html lang="ru">*  *<head>*  *<meta charset="UTF-8">*  *<meta name="viewport" content="width=device-width, initial-scale=1.0">*  *<meta http-equiv="X-UA-Compatible" content="ie=edge">*  *<title>Регистрация</title>*  *</head>*  *<body>*  *<form method="post" action="<?php echo $\_SERVER['PHP\_SELF'];?>">*  *ФИО <input type="text" name="fio" placeholder="Введите ФИО полностью"> <samp style="color:red">\*</samp>*  *Логин <input type="text" name="login" placeholder="Введите логин"> <samp style="color:red">\*</samp>*  *EMail <input type="email" name="email" placeholder="Введите почту"><samp style="color:red">\*</samp>*  *Пароль <input type="password" name="pass" placeholder="Введите пароль"><samp style="color:red">\*</samp>*  *Повторите пароль <input type="password" name="pass\_rep" placeholder="Введите пароль"><samp style="color:red">\*</samp>*  *Дата рождения <input type="date" name="year\_of\_birth" placeholder="Введите дату рождения">*  *<input type="submit" value="Зарегистрироваться" name="doGo">*  *<p>Уже зарегестрированы? - <a href="auth.php">Авторизируйтесь!</a></p>*  *<?php*  *if ($\_SESSION['message']){*  *echo '<p class="msg"> ' . $\_SESSION['message'] . ' </p>';*  *}*  *unset($\_SESSION['message']);*  *?>*  *</form>*  *</body>*  *</html>* |

Далее нужно создать страницу профиля авторизованного пользователя. Для этого переходим в файл profile.php.

Листинг 3.5 – profile.php

|  |
| --- |
| <!DOCTYPE html>  <link rel="stylesheet" type="text/css" href="assets/profile\_style.css">  <html lang="en">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>Профиль</title>  </head>  <body>  <?php  session\_start();  if (!$\_SESSION['user']){  header('Location: auth.php');  }  include\_once 'connect.php';  ?>  <div class="card">  <div class="card\_background\_img"></div>  <div>  <a href="edit.php">  <?php  $log = $\_SESSION['user']['login'];  $ava = mysqli\_query($link, "SELECT avatar FROM users WHERE login = '$log'");  $pic = mysqli\_fetch\_assoc($ava);  $chk = $pic['avatar'];  if ($chk == NULL || file\_exists($chk) == false){  echo '<img src="images/user.png" class="card\_profile\_img">';  }  else {  echo '<img src="'. $pic['avatar'] .'" class="card\_profile\_img">';  }  ?>  </a>  </div>  <div class="user\_details">  <h3><?php echo $\_SESSION['user']['login'];?></h3>  <p>Рады вас видеть!</p>  </div>  <div class="card\_count">  <div class="count2">  <div>  <h3><?php echo $\_SESSION['user']['fio'];?></h3><br>  <p>ФИО</p>  </div>  </div>  <div class="count">  <div>  <h3><?php echo $\_SESSION['user']['email'];?></h3><br>  <p>Почта</p>  </div>  <div>  <h3><?php echo $\_SESSION['user']['dob'];?></h3><br>  <p>Дата рождения</p>  </div>  </div>  <?php  if ($\_SESSION['user']['login'] == 'admin'){  echo '<div class="btn"><a href="">Админ панель</a></div><br>';  }  if ($\_SESSION['user']['login'] !== 'admin'){  echo '<div class="btn"><a href="bookings.php">Мои брони</a></div><br>';  }  ?>  <div class="infobtn"><a href="index.php">На главную</a></div><br>  <div class="exitbtn"><a href="logout.php">Выход</a></div>  </div>  </div>  </body>  </html> |

Далее нужно создать profile\_style.css документ для того, чтобы использовать стили на каждой странице профиля (авторизованного пользователя).

Листинг 3.6 – profile\_style.css

|  |
| --- |
| \* {  margin: 0;  padding: 0;  box-sizing: border-box;  }  body {  background-color: #e1e7ed;  text-align: center;  }  h3 {  font-size: 18px;  font-weight: 500;  color: #33363b;  }  p{  font-size: 14px;  color: #868d9b;  }  .card {  width: 440px;  height: auto;  background-color: #ffffff;  margin: 0 auto;  margin-top: 80px;  box-shadow: 0px 1px 3px 0px rgba(0, 0, 0, 0.15);  }  .cardbooking {  width: 600px;  height: auto;  background-color: #ffffff;  margin: 0 auto;  margin-top: 80px;  box-shadow: 0px 1px 3px 0px rgba(0, 0, 0, 0.15);  }  .card\_profile\_img {  width: 120px;  height: 120px;  background-color: #868d9b;  background-repeat: no-repeat;  background-size: cover;  background-position: center;  border: 2px solid #ffffff;  border-radius: 120px;  margin: 0 auto;  margin-top: -60px;  }  .card\_background\_img {  width: 100%;  height: 180px;  background-color: #e1e7ed;  background: url("../images/profile-back.jpg");  background-repeat: no-repeat;  background-size: cover;  background-position: center;  }  .user\_details p {  margin-bottom: 20px;  margin-top: -5px;  }  .user\_details h3 {  margin-top: 10px;  }  .card\_count {  padding: 30px;  border-top: 1px solid #dde1e7;  }  .count {  display: flex;  align-items: center;  justify-content: space-between;  margin-bottom: 28px;  }  .count2 {  display: inline-block;  align-items: center;  justify-content: space-between;  margin-bottom: 28px;  }  .count p {  margin-top: -10px;  }  .btn {  padding: 16px;  width: 100%;  border: none;  outline: none;  border-radius: 4px;  background-color: #93c5e4;  color: #000000;  cursor: pointer;  text-align: center;  font-size: 14px;  font-weight: 600;  text-transform: uppercase;  position: relative;  left: 50%;  transform: translate(-50%, 0);  }  .infobtn {  padding: 16px;  width: 100%;  border: none;  outline: none;  border-radius: 4px;  background-color: #5e8da8;  color: #ffffff;  cursor: pointer;  text-align: center;  font-size: 14px;  font-weight: 600;  text-transform: uppercase;  position: relative;  left: 50%;  transform: translate(-50%, 0);  }  .smallbtn {  padding: 16px;  width: 300px;  border: none;  outline: none;  border-radius: 4px;  background-color: #3c596a;  color: #ffffff;  cursor: pointer;  text-align: center;  font-size: 14px;  font-weight: 600;  text-transform: uppercase;  position: relative;  left: 50%;  transform: translate(-50%, 0);  }  .exitbtn {  padding: 16px;  width: 100%;  border: none;  outline: none;  border-radius: 4px;  background-color: #3c596a;  color: #ffffff;  cursor: pointer;  text-align: center;  font-size: 14px;  font-weight: 600;  text-transform: uppercase;  position: relative;  left: 50%;  transform: translate(-50%, 0);  }  a:link {  color: #ffffff;  text-decoration: none;  }  a:hover {  color: #ffffff;  text-decoration: none;  }  a:visited {  color: #ffffff;  text-decoration: none;  }  a:active {  color: #ffffff;  text-decoration: none;  } |

На каждой странице системы авторазции нужно подключить сессию, для этого нужно использовать функцию *session\_start()*. Эта функция в дальнейшем позволит использовать суперглобальную переменную *$\_SESSION*, благодаря которой сервер сможет запоминать авторизованного пользователя. Потребуется скрипт, который будет создавать запоминающую переменную, поэтому переходим в файл signin.php:

Листинг 3.7 – signin.php

|  |
| --- |
| <?php  session\_start();  require\_once 'connect.php';  $login = $\_POST['login'];  $check\_user = mysqli\_query($link, "SELECT \* FROM users WHERE login = '$login'");  if (mysqli\_num\_rows($check\_user) > 0) {  $user = mysqli\_fetch\_assoc($check\_user);  $\_SESSION['user'] = [  "id" => $user['id'],  "fio" => $user['fio'],  "login" => $user['login'],  "email" => $user['email'],  "dob" => $user['dob'],  "avatar" => $user['avatar']  ];  header('Location: profile.php');  } else {  header('Location: auth.php');  }  ?> |

И для того чтобы пользователь в любой момент мог выйти из профиля, нужен небольшой скрипт для удаления авторизованного пользователя из памяти. Для этого переходим в файл logout.php.

Листинг 3.8 – logout.php

|  |
| --- |
| <?php  session\_start();  unset($\_SESSION['user']);  header('Location: index.php');  ?> |

Теперь создается форма для смены аватара профиля. Для этого переходим в файл edit.php.

Листинг 3.9 – edit.php

|  |
| --- |
| *<?php*  *session\_start();*  *if (!$\_SESSION){*  *header('Location: auth.php');*  *}*  *include\_once 'connect.php';*    *if (isset($\_POST['doGo'])){*  *$path = 'images/' . time() . $\_FILES['avatar']['name'];*  *if (!move\_uploaded\_file($\_FILES['avatar']['tmp\_name'], $path)) echo 'Ошибка при загрузке сообщения';*  *else{*  *$log = $\_SESSION['user']['login'];*  *mysqli\_query($link, "UPDATE users SET avatar = '$path' WHERE login = '$log'");*  *header('Location: profile.php');*  *}*  *}*  *?>*  *<!DOCTYPE html>*  *<link rel="stylesheet" href="assets/profile\_style.css">*  *<html lang="en">*  *<head>*  *<meta charset="UTF-8">*  *<meta name="viewport" content="width=device-width, initial-scale=1.0">*  *<title>Профиль</title>*  *</head>*  *<body>*  *<div class="card">*  *<div class="user\_details">*  *<br>*  *<h3>Изменить аватар</h3><br>*  *<p><?php echo $\_SESSION['user']['login'];?></p>*  *</div>*  *<div class="card\_count">*  *<div class="count2">*  *<div>*  *<form method="post" action="<?php echo $\_SERVER['PHP\_SELF'];?>" enctype="multipart/form-data">*  *<label>Выберите изображение профиля</label><br>*  *</div>*  *<br>*  *<div>*  *<input type="file" name="avatar">*  *</div>*  *<br>*  *<div>*  *<input type="submit" name="doGo" class="btn" value="Изменить">*  *</form>*  *</div>*  *</div>*  *</div>*  *</div>*  *</body>*  *</html>* |

Далее нужно написать страницу, которая будет отображать все брони пользователя. Нужно осуществить проверку для кажого поля БД на соответствие почты или логина. Для этого переходим в файл bookings.php.

Листинг 3.10 – bookings.php

|  |
| --- |
| <!DOCTYPE html>  <link rel="stylesheet" type="text/css" href="assets/profile\_style.css">  <html lang="en">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>Профиль</title>  </head>  <body>  <?php  session\_start();  if (!$\_SESSION['user']){  header('Location: auth.php');  }  include\_once 'connect.php';  ?>  <div class="cardbooking">  <div class="card\_background\_img"></div>  <div>  <?php  $log = $\_SESSION['user']['login'];  $ava = mysqli\_query($link, "SELECT avatar FROM users WHERE login = '$log'");  $pic = mysqli\_fetch\_assoc($ava);  $chk = $pic['avatar'];  if ($chk == NULL || file\_exists($chk) == false){  echo '<img src="images/user.png" class="card\_profile\_img">';  }  else {  echo '<img src="'. $pic['avatar'] .'" class="card\_profile\_img">';  }    $email = $\_SESSION['user']['email'];  $bd = mysqli\_connect("localhost", "root", "", "booking-system") or die ("Невозможно подключиться к серверу");  $home1 = mysqli\_query($bd, "SELECT date FROM room1 WHERE email = '$email'");  $i=0;  $items1 = array();  while(($datas1 = mysqli\_fetch\_assoc($home1)) !== NULL){  $items1[$i] = $datas1['date'];  $i++;  }  $home2 = mysqli\_query($bd, "SELECT date FROM room2 WHERE email = '$email'");  $i=0;  $items2 = array();  while(($datas2 = mysqli\_fetch\_assoc($home2)) !== NULL){  $items2[$i] = $datas2['date'];  $i++;  }  $home3 = mysqli\_query($bd, "SELECT date FROM room3 WHERE email = '$email'");  $i=0;  $items3 = array();  while(($datas3 = mysqli\_fetch\_assoc($home3)) !== NULL){  $items3[$i] = $datas3['date'];  $i++;  }  $home4 = mysqli\_query($bd, "SELECT date FROM room4 WHERE email = '$email'");  $i=0;  $items4 = array();  while(($datas4 = mysqli\_fetch\_assoc($home4)) !== NULL){  $items4[$i] = $datas4['date'];  $i++;  }  $home5 = mysqli\_query($bd, "SELECT date FROM room5 WHERE email = '$email'");  $i=0;  $items5 = array();  while(($datas5 = mysqli\_fetch\_assoc($home5)) !== NULL){  $items5[$i] = $datas5['date'];  $i++;  }  ?>  </a>  </div>  <div class="user\_details">  <h3><?php echo $\_SESSION['user']['login'];?></h3>  <p>Ваши брони</p>  </div>  <div class="card\_count">  <div class="count">  <?php  if ($items1 != NULL){  $i=1;  echo '<div>Домик №1 на даты</div><div>';  foreach ($items1 as $value){  echo '(', $i, ') ', $value, ' | ';  $i++;  }  echo '<p></p></div>';  }  ?>  </div>  <div class="count">  <?php  if ($items2 != NULL){  $i=1;  echo '<div>Домик №2 на даты</div><div>';  foreach ($items2 as $value){  echo '(', $i, ') ', $value, ' | ';  $i++;  }  echo '<p></p></div>';  }  ?>  </div>  <div class="count">  <?php  if ($items3 != NULL){  $i=1;  echo '<div><p>Домик №3 на даты</p></div><div>';  foreach ($items3 as $value){  echo '(', $i, ') ', $value, ' | ';  $i++;  }  echo '<p></p></div>';  }  ?>  </div>  <div class="count">  <?php  if ($items4 != NULL){  $i=1;  echo '<div><p>Домик №4 на даты</p></div><div>';  foreach ($items4 as $value){  echo '(', $i, ') ', $value, ' | ';  $i++;  }  echo '<p></p></div>';  }  ?>  </div>  <div class="count">  <?php  if ($items5 != NULL){  $i=1;  echo '<div><p>Домик №5 на даты</p></div><div>';  foreach ($items5 as $value){  echo '(', $i, ') ', $value, ' | ';  $i++;  }  echo '<p></p></div>';  }  ?>  </div>  <div class="smallbtn">  <a href="profile.php">Назад</a>  </div>  </div>  </div>  </body>  </html> |

Далее создается страница обработки почты неавторизованного пользователя. Это нужно для того, чтобы узнать существует ли профиль с такой почтой и, в случае если пользователь с такой почтой существует, сервер отправит автоматическое письмо со ссылкой для смены пароля, используя удаленный SMTP сервер Google. Для этого переходим в файл mail-verify.php.

Листинг 3.11 – mail-verify.php

|  |
| --- |
| *<?php*  *session\_start();*  *if ($\_SESSION['user']){*  *header('Location: profile.php');*  *}*  *include\_once 'connect.php';*    *if (isset($\_POST['doGo'])){*  *$to = $\_POST['email'];*  *$check = mysqli\_query($link, "SELECT login, email FROM users WHERE email = '$to'");*  *$log = mysqli\_fetch\_assoc($check);*  *$login = $log['login'];*  *$email = $log['email'];*  *if ($to == $email){*  *$\_SESSION['passworded'] = [ "login" => $log['login'] ];*  *$subject = "Здравствуйте, это автоматическое письмо для смены пароля!";*  *$message = 'Ссылка на смену пароля: http://altaydacha/diplom/pass-replace.php';*  *mail($to, $subject, $message);*  *$\_SESSION['upset'] = 'Письмо отправлено на вашу почту';*  *header('Location: auth.php');*  *}*  *else{*  *$\_SESSION['upset'] = 'Пользователя с таким email нет';*  *//header('Location: auth.php');*  *var\_dump($to);*  *var\_dump($login);*  *}*    *}*  *?>*  *<!DOCTYPE html>*  *<link rel="stylesheet" href="assets/main.css">*  *<html lang="en">*  *<head>*  *<meta charset="UTF-8">*  *<meta name="viewport" content="width=device-width, initial-scale=1.0">*  *<title>Замена пароля</title>*  *</head>*  *<body>*  *<form method="post" action="<?php echo $\_SERVER['PHP\_SELF'];?>">*  *<h3>Введите почту указанную при регистрации</h3><br>*  *Почта <input type="email" name="email" placeholder="Введите почту">*  *<input type="submit" value="Отправить" name="doGo">*  *<p>Вам придет письмо со ссылкой на замену пароля</p>*  *</form>*  *</body>*  *</html>* |

Теперь нужно создать страницу для смены пароля профиля, ссылка на которую будет приходить пользователю по почте. Заходим в файл pass-replace.php.

Листинг 3.12 – pass-replace.php

|  |
| --- |
| *<?php*  *session\_start();*  *include\_once 'connect.php';*  *if (isset($\_POST['doGo'])) {*  *if ($\_POST['newpass'] !== $\_POST['newpass\_rep']) {*  *$\_SESSION['mispass'] = 'Пароли не совпадают';*  *}*  *else{*  *$newpass = password\_hash($\_POST['newpass'], PASSWORD\_DEFAULT);*  *$log = $\_SESSION['passworded']['login'];*  *$db = mysqli\_query($link, "UPDATE users SET password = '$newpass' WHERE login = '$log'");*    *if(mysqli\_affected\_rows($link) > 0) {*  *$\_SESSION['upset'] = 'Пароль успешно изменен';*  *unset($\_SESSION['passworded']);*  *header('Location: auth.php');*  *}*  *else {*  *$\_SESSION['mispass'] = 'Ошибка кода';*  *}*  *}*  *}*  *?>*  *<!DOCTYPE html>*  *<link rel="stylesheet" href="assets/main.css">*  *<html lang="en">*  *<head>*  *<meta charset="UTF-8">*  *<meta name="viewport" content="width=device-width, initial-scale=1.0">*  *<title>Замена пароля</title>*  *</head>*  *<body>*  *<form method="post" action="<?php echo $\_SERVER['PHP\_SELF'];?>">*  *Новый Пароль <input type="password" name="newpass" placeholder="Введите новый пароль">*  *Повторите пароль <input type="password" name="newpass\_rep" placeholder="Повторите новый пароль">*  *<input type="submit" value="Изменить" name="doGo">*  *<?php*  *if ($\_SESSION['mispass']){*  *echo '<p class="msg"> ' . $\_SESSION['mispass'] . ' </p>';*  *}*  *unset($\_SESSION['upset']);*  *?>*  *</form>*  *</body>*  *</html>* |

Теперь нужно создать страницу, которая будет выполнять функцию построения календаря для последующего бронирования. Страница будет отображать текущий месяц и день для удобства. Пользователю будет доступно:

1. Смена месяца по нажатию кнопки.
2. Выбор любого дня и дальнейшее заполнение формы для посуточного бронирования.
3. Кнопка возврата на главную страницу и/или страницу каждого дома доступного к бронированию

Соответственно нужна БД и страница для каждого дома. Для этого перейдем в файлы index1.php … index5.php.

Листинг 3.13 – index1.php

|  |
| --- |
| <?php  if(isset($\_POST['bit'])){  header('Location: ../../homes.php');  }  function build\_calendar($month, $year) {  $mysqli = new mysqli('localhost', 'root', '', 'booking-system');  $stmt = $mysqli->prepare("SELECT \* FROM room1 WHERE MONTH(date) = ? AND YEAR(date) = ?");  $stmt->bind\_param('ss', $month, $year);  $bookings = array();  if($stmt->execute()){  $result = $stmt->get\_result();  if($result->num\_rows>0){  while($row = $result->fetch\_assoc()){  $bookings[] = $row['date'];  }    $stmt->close();  }  }  $daysOfWeek = array('Воскресенье','Понедельник','Вторник','Среда','Четверг','Пятница','Суббота');  $firstDayOfMonth = mktime(0,0,0,$month,1,$year);  $numberDays = date('t',$firstDayOfMonth);  $dateComponents = getdate($firstDayOfMonth);  $monthName = $dateComponents['month'];  $dayOfWeek = $dateComponents['wday'];  $datetoday = date('Y-m-d');    $calendar = "<table class='table table-bordered'>";  $calendar .= "<center><h2>$month - $year</h2>";  $calendar.= "<a class='btn btn-xs btn-primary' href='?month=".date('m', mktime(0, 0, 0, $month-1, 1, $year))."&year=".date('Y', mktime(0, 0, 0, $month-1, 1, $year))."'><=</a> ";    $calendar.= " <a class='btn btn-xs btn-primary' href='?month=".date('m')."&year=".date('Y')."'>Текущий месяц</a> ";    $calendar.= "<a class='btn btn-xs btn-primary' href='?month=".date('m', mktime(0, 0, 0, $month+1, 1, $year))."&year=".date('Y', mktime(0, 0, 0, $month+1, 1, $year))."'>=></a></center><br>";    $calendar .= "<tr>";  foreach($daysOfWeek as $day) {  $calendar .= "<th class='header'>$day</th>";  }  $currentDay = 1;  $calendar .= "</tr><tr>";  if ($dayOfWeek > 0) {  for($k=0;$k<$dayOfWeek;$k++){  $calendar .= "<td class='empty'></td>";  }  }    $month = str\_pad($month, 2, "0", STR\_PAD\_LEFT);    while ($currentDay <= $numberDays) {  if ($dayOfWeek == 7) {  $dayOfWeek = 0;  $calendar .= "</tr><tr>";  }    $currentDayRel = str\_pad($currentDay, 2, "0", STR\_PAD\_LEFT);  $date = "$year-$month-$currentDayRel";    $dayname = strtolower(date('l', strtotime($date)));  $eventNum = 0;  $today = $date==date('Y-m-d')? "today" : "";  if($date<date('Y-m-d')){  $calendar.="<td><h4>$currentDay</h4> <button class='btn btn-danger btn-xs'>Н/Д</button>";  }  elseif(in\_array($date, $bookings)){  $calendar.="<td class='$today'><h4>$currentDay</h4> <button class='btn btn-danger btn-xs'>Забронировано</button>";  }  else{  $calendar.="<td class='$today'><h4>$currentDay</h4> <a href='book1.php?date=".$date."' class='btn btn-success btn-xs'>Забронировать</a>";  }    $calendar .="</td>";    $currentDay++;  $dayOfWeek++;  }    if ($dayOfWeek != 7) {    $remainingDays = 7 - $dayOfWeek;  for($l=0;$l<$remainingDays;$l++){  $calendar .= "<td class='empty'></td>";  }  }    $calendar .= "</tr>";  $calendar .= "</table>";  echo $calendar;  }  ?>  <html>  <head>  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css">  <link rel="stylesheet" href="../assets/main.css">  <style>  @media only screen and (max-width: 760px), (min-device-width: 802px) and (max-device-width: 1020px) {  table, thead, tbody, th, td, tr {  display: block;  }  .empty {  display: none;  }  th {  position: absolute;  top: -9999px;  left: -9999px;  }  tr {  border: 1px solid #ccc;  }  td {  border: none;  border-bottom: 1px solid #eee;  position: relative;  padding-left: 50%;  }  td:nth-of-type(1):before {  content: "Sunday";  }  td:nth-of-type(2):before {  content: "Monday";  }  td:nth-of-type(3):before {  content: "Tuesday";  }  td:nth-of-type(4):before {  content: "Wednesday";  }  td:nth-of-type(5):before {  content: "Thursday";  }  td:nth-of-type(6):before {  content: "Friday";  }  td:nth-of-type(7):before {  content: "Saturday";  }  }  h1{  text-align: center;  }  /\* Под смартфоны \*/  @media only screen and (min-device-width: 320px) and (max-device-width: 480px) {  body {  padding: 0;  margin: 0;  }  }  /\* Под айпады \*/  @media only screen and (min-device-width: 802px) and (max-device-width: 1020px) {  body {  width: 495px;  }  }  @media (min-width:641px) {  table {  table-layout: fixed;  }  td {  width: 33%;  }  }    .row{  margin-top: 20px;  }    .today{  background:yellow;  }  </style>  </head>  <body>  <div class="container">  <div class="row">  <div class="col-md-12">  <h1>Бронирование для Дома №1, на:</h1>  <?php  $dateComponents = getdate();  if(isset($\_GET['month']) && isset($\_GET['year'])){  $month = $\_GET['month'];  $year = $\_GET['year'];  }else{  $month = $dateComponents['mon'];  $year = $dateComponents['year'];  }  echo build\_calendar($month,$year);  ?>  <form action="" method="post" autocomplete="off">  <button class="btn btn-primary" type="submit" name="bit">Назад</button>  </form>  </div>  </div>  </div>  </body>  </html> |

Листинг 3.14 – index2.php

|  |
| --- |
| *<?php*  *if(isset($\_POST['bit'])){*  *header('Location: ../../homes3.html');*  *}*  *function build\_calendar($month, $year) {*  *$mysqli = new mysqli('localhost', 'root', '', 'booking-system');*  *$stmt = $mysqli->prepare("SELECT \* FROM room2 WHERE MONTH(date) = ? AND YEAR(date) = ?");*  *$stmt->bind\_param('ss', $month, $year);*  *$bookings = array();*  *if($stmt->execute()){*  *$result = $stmt->get\_result();*  *if($result->num\_rows>0){*  *while($row = $result->fetch\_assoc()){*  *$bookings[] = $row['date'];*  *}*    *$stmt->close();*  *}*  *}*  *$daysOfWeek = array('Воскресенье','Понедельник','Вторник','Среда','Четверг','Пятница','Суббота');*  *$firstDayOfMonth = mktime(0,0,0,$month,1,$year);*  *$numberDays = date('t',$firstDayOfMonth);*  *$dateComponents = getdate($firstDayOfMonth);*  *$monthName = $dateComponents['month'];*  *$dayOfWeek = $dateComponents['wday'];*  *$datetoday = date('Y-m-d');*    *$calendar = "<table class='table table-bordered'>";*  *$calendar .= "<center><h2>$month - $year</h2>";*  *$calendar.= "<a class='btn btn-xs btn-primary' href='?month=".date('m', mktime(0, 0, 0, $month-1, 1, $year))."&year=".date('Y', mktime(0, 0, 0, $month-1, 1, $year))."'><=</a> ";*    *$calendar.= " <a class='btn btn-xs btn-primary' href='?month=".date('m')."&year=".date('Y')."'>Текущий месяц</a> ";*    *$calendar.= "<a class='btn btn-xs btn-primary' href='?month=".date('m', mktime(0, 0, 0, $month+1, 1, $year))."&year=".date('Y', mktime(0, 0, 0, $month+1, 1, $year))."'>=></a></center><br>";*    *$calendar .= "<tr>";*  *foreach($daysOfWeek as $day) {*  *$calendar .= "<th class='header'>$day</th>";*  *}*  *$currentDay = 1;*  *$calendar .= "</tr><tr>";*  *if ($dayOfWeek > 0) {*  *for($k=0;$k<$dayOfWeek;$k++){*  *$calendar .= "<td class='empty'></td>";*  *}*  *}*    *$month = str\_pad($month, 2, "0", STR\_PAD\_LEFT);*    *while ($currentDay <= $numberDays) {*  *if ($dayOfWeek == 7) {*  *$dayOfWeek = 0;*  *$calendar .= "</tr><tr>";*  *}*    *$currentDayRel = str\_pad($currentDay, 2, "0", STR\_PAD\_LEFT);*  *$date = "$year-$month-$currentDayRel";*    *$dayname = strtolower(date('l', strtotime($date)));*  *$eventNum = 0;*  *$today = $date==date('Y-m-d')? "today" : "";*  *if($date<date('Y-m-d')){*  *$calendar.="<td><h4>$currentDay</h4> <button class='btn btn-danger btn-xs'>Н/Д</button>";*  *}*  *elseif(in\_array($date, $bookings)){*  *$calendar.="<td class='$today'><h4>$currentDay</h4> <button class='btn btn-danger btn-xs'>Забронировано</button>";*  *}*  *else{*  *$calendar.="<td class='$today'><h4>$currentDay</h4> <a href='book2.php?date=".$date."' class='btn btn-success btn-xs'>Забронировать</a>";*  *}*    *$calendar .="</td>";*    *$currentDay++;*  *$dayOfWeek++;*  *}*    *if ($dayOfWeek != 7) {*    *$remainingDays = 7 - $dayOfWeek;*  *for($l=0;$l<$remainingDays;$l++){*  *$calendar .= "<td class='empty'></td>";*  *}*  *}*    *$calendar .= "</tr>";*  *$calendar .= "</table>";*  *echo $calendar;*  *}*  *?>*  *<html>*  *<head>*  *<meta name="viewport" content="width=device-width, initial-scale=1.0">*  *<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css">*  *<link rel="stylesheet" href="../assets/main.css">*  *<style>*  *@media only screen and (max-width: 760px), (min-device-width: 802px) and (max-device-width: 1020px) {*  *table, thead, tbody, th, td, tr {*  *display: block;*  *}*  *.empty {*  *display: none;*  *}*  *th {*  *position: absolute;*  *top: -9999px;*  *left: -9999px;*  *}*  *tr {*  *border: 1px solid #ccc;*  *}*  *td {*  *border: none;*  *border-bottom: 1px solid #eee;*  *position: relative;*  *padding-left: 50%;*  *}*  *td:nth-of-type(1):before {*  *content: "Sunday";*  *}*  *td:nth-of-type(2):before {*  *content: "Monday";*  *}*  *td:nth-of-type(3):before {*  *content: "Tuesday";*  *}*  *td:nth-of-type(4):before {*  *content: "Wednesday";*  *}*  *td:nth-of-type(5):before {*  *content: "Thursday";*  *}*  *td:nth-of-type(6):before {*  *content: "Friday";*  *}*  *td:nth-of-type(7):before {*  *content: "Saturday";*  *}*  *}*  *h1{*  *text-align: center;*  *}*  */\* Под смартфоны \*/*  *@media only screen and (min-device-width: 320px) and (max-device-width: 480px) {*  *body {*  *padding: 0;*  *margin: 0;*  *}*  *}*  */\* Под айпады \*/*  *@media only screen and (min-device-width: 802px) and (max-device-width: 1020px) {*  *body {*  *width: 495px;*  *}*  *}*  *@media (min-width:641px) {*  *table {*  *table-layout: fixed;*  *}*  *td {*  *width: 33%;*  *}*  *}*    *.row{*  *margin-top: 20px;*  *}*    *.today{*  *background:yellow;*  *}*  *</style>*  *</head>*  *<body>*  *<div class="container">*  *<div class="row">*  *<div class="col-md-12">*  *<h1>Бронирование для Дома №2, на:</h1>*  *<?php*  *$dateComponents = getdate();*  *if(isset($\_GET['month']) && isset($\_GET['year'])){*  *$month = $\_GET['month'];*  *$year = $\_GET['year'];*  *}else{*  *$month = $dateComponents['mon'];*  *$year = $dateComponents['year'];*  *}*  *echo build\_calendar($month,$year);*  *?>*  *<form action="" method="post" autocomplete="off">*  *<button class="btn btn-primary" type="submit" name="bit">Назад</button>*  *</form>*  *</div>*  *</div>*  *</div>*  *</body>*  *</html>* |

Листинг 3.15 – index3.php

|  |
| --- |
| <?php  if(isset($\_POST['bit'])){  header('Location: ../../homes3.html');  }  function build\_calendar($month, $year) {  $mysqli = new mysqli('localhost', 'root', '', 'booking-system');  $stmt = $mysqli->prepare("SELECT \* FROM room3 WHERE MONTH(date) = ? AND YEAR(date) = ?");  $stmt->bind\_param('ss', $month, $year);  $bookings = array();  if($stmt->execute()){  $result = $stmt->get\_result();  if($result->num\_rows>0){  while($row = $result->fetch\_assoc()){  $bookings[] = $row['date'];  }    $stmt->close();  }  }  $daysOfWeek = array('Воскресенье','Понедельник','Вторник','Среда','Четверг','Пятница','Суббота');  $firstDayOfMonth = mktime(0,0,0,$month,1,$year);  $numberDays = date('t',$firstDayOfMonth);  $dateComponents = getdate($firstDayOfMonth);  $monthName = $dateComponents['month'];  $dayOfWeek = $dateComponents['wday'];  $datetoday = date('Y-m-d');    $calendar = "<table class='table table-bordered'>";  $calendar .= "<center><h2>$month - $year</h2>";  $calendar.= "<a class='btn btn-xs btn-primary' href='?month=".date('m', mktime(0, 0, 0, $month-1, 1, $year))."&year=".date('Y', mktime(0, 0, 0, $month-1, 1, $year))."'><=</a> ";    $calendar.= " <a class='btn btn-xs btn-primary' href='?month=".date('m')."&year=".date('Y')."'>Текущий месяц</a> ";    $calendar.= "<a class='btn btn-xs btn-primary' href='?month=".date('m', mktime(0, 0, 0, $month+1, 1, $year))."&year=".date('Y', mktime(0, 0, 0, $month+1, 1, $year))."'>=></a></center><br>";    $calendar .= "<tr>";  foreach($daysOfWeek as $day) {  $calendar .= "<th class='header'>$day</th>";  }  $currentDay = 1;  $calendar .= "</tr><tr>";  if ($dayOfWeek > 0) {  for($k=0;$k<$dayOfWeek;$k++){  $calendar .= "<td class='empty'></td>";  }  }    $month = str\_pad($month, 2, "0", STR\_PAD\_LEFT);    while ($currentDay <= $numberDays) {  if ($dayOfWeek == 7) {  $dayOfWeek = 0;  $calendar .= "</tr><tr>";  }    $currentDayRel = str\_pad($currentDay, 2, "0", STR\_PAD\_LEFT);  $date = "$year-$month-$currentDayRel";    $dayname = strtolower(date('l', strtotime($date)));  $eventNum = 0;  $today = $date==date('Y-m-d')? "today" : "";  if($date<date('Y-m-d')){  $calendar.="<td><h4>$currentDay</h4> <button class='btn btn-danger btn-xs'>Н/Д</button>";  }  elseif(in\_array($date, $bookings)){  $calendar.="<td class='$today'><h4>$currentDay</h4> <button class='btn btn-danger btn-xs'>Забронировано</button>";  }  else{  $calendar.="<td class='$today'><h4>$currentDay</h4> <a href='book3.php?date=".$date."' class='btn btn-success btn-xs'>Забронировать</a>";  }    $calendar .="</td>";    $currentDay++;  $dayOfWeek++;  }    if ($dayOfWeek != 7) {    $remainingDays = 7 - $dayOfWeek;  for($l=0;$l<$remainingDays;$l++){  $calendar .= "<td class='empty'></td>";  }  }    $calendar .= "</tr>";  $calendar .= "</table>";  echo $calendar;  }  ?>  <html>  <head>  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css">  <link rel="stylesheet" href="../assets/main.css">  <style>  @media only screen and (max-width: 760px), (min-device-width: 802px) and (max-device-width: 1020px) {  table, thead, tbody, th, td, tr {  display: block;  }  .empty {  display: none;  }  th {  position: absolute;  top: -9999px;  left: -9999px;  }  tr {  border: 1px solid #ccc;  }  td {  border: none;  border-bottom: 1px solid #eee;  position: relative;  padding-left: 50%;  }  td:nth-of-type(1):before {  content: "Sunday";  }  td:nth-of-type(2):before {  content: "Monday";  }  td:nth-of-type(3):before {  content: "Tuesday";  }  td:nth-of-type(4):before {  content: "Wednesday";  }  td:nth-of-type(5):before {  content: "Thursday";  }  td:nth-of-type(6):before {  content: "Friday";  }  td:nth-of-type(7):before {  content: "Saturday";  }  }  h1{  text-align: center;  }  /\* Под смартфоны \*/  @media only screen and (min-device-width: 320px) and (max-device-width: 480px) {  body {  padding: 0;  margin: 0;  }  }  /\* Под айпады \*/  @media only screen and (min-device-width: 802px) and (max-device-width: 1020px) {  body {  width: 495px;  }  }  @media (min-width:641px) {  table {  table-layout: fixed;  }  td {  width: 33%;  }  }    .row{  margin-top: 20px;  }    .today{  background:yellow;  }  </style>  </head>  <body>  <div class="container">  <div class="row">  <div class="col-md-12">  <h1>Бронирование для Дома №3, на:</h1>  <?php  $dateComponents = getdate();  if(isset($\_GET['month']) && isset($\_GET['year'])){  $month = $\_GET['month'];  $year = $\_GET['year'];  }else{  $month = $dateComponents['mon'];  $year = $dateComponents['year'];  }  echo build\_calendar($month,$year);  ?>  <form action="" method="post" autocomplete="off">  <button class="btn btn-primary" type="submit" name="bit">Назад</button>  </form>  </div>  </div>  </div>  </body>  </html> |

Листинг 3.16 – index4.php

|  |
| --- |
| *<?php*  *if(isset($\_POST['bit'])){*  *header('Location: ../../homes3.html');*  *}*  *function build\_calendar($month, $year) {*  *$mysqli = new mysqli('localhost', 'root', '', 'booking-system');*  *$stmt = $mysqli->prepare("SELECT \* FROM room4 WHERE MONTH(date) = ? AND YEAR(date) = ?");*  *$stmt->bind\_param('ss', $month, $year);*  *$bookings = array();*  *if($stmt->execute()){*  *$result = $stmt->get\_result();*  *if($result->num\_rows>0){*  *while($row = $result->fetch\_assoc()){*  *$bookings[] = $row['date'];*  *}*    *$stmt->close();*  *}*  *}*  *$daysOfWeek = array('Воскресенье','Понедельник','Вторник','Среда','Четверг','Пятница','Суббота');*  *$firstDayOfMonth = mktime(0,0,0,$month,1,$year);*  *$numberDays = date('t',$firstDayOfMonth);*  *$dateComponents = getdate($firstDayOfMonth);*  *$monthName = $dateComponents['month'];*  *$dayOfWeek = $dateComponents['wday'];*  *$datetoday = date('Y-m-d');*    *$calendar = "<table class='table table-bordered'>";*  *$calendar .= "<center><h2>$month - $year</h2>";*  *$calendar.= "<a class='btn btn-xs btn-primary' href='?month=".date('m', mktime(0, 0, 0, $month-1, 1, $year))."&year=".date('Y', mktime(0, 0, 0, $month-1, 1, $year))."'><=</a> ";*    *$calendar.= " <a class='btn btn-xs btn-primary' href='?month=".date('m')."&year=".date('Y')."'>Текущий месяц</a> ";*    *$calendar.= "<a class='btn btn-xs btn-primary' href='?month=".date('m', mktime(0, 0, 0, $month+1, 1, $year))."&year=".date('Y', mktime(0, 0, 0, $month+1, 1, $year))."'>=></a></center><br>";*    *$calendar .= "<tr>";*  *foreach($daysOfWeek as $day) {*  *$calendar .= "<th class='header'>$day</th>";*  *}*  *$currentDay = 1;*  *$calendar .= "</tr><tr>";*  *if ($dayOfWeek > 0) {*  *for($k=0;$k<$dayOfWeek;$k++){*  *$calendar .= "<td class='empty'></td>";*  *}*  *}*    *$month = str\_pad($month, 2, "0", STR\_PAD\_LEFT);*    *while ($currentDay <= $numberDays) {*  *if ($dayOfWeek == 7) {*  *$dayOfWeek = 0;*  *$calendar .= "</tr><tr>";*  *}*    *$currentDayRel = str\_pad($currentDay, 2, "0", STR\_PAD\_LEFT);*  *$date = "$year-$month-$currentDayRel";*    *$dayname = strtolower(date('l', strtotime($date)));*  *$eventNum = 0;*  *$today = $date==date('Y-m-d')? "today" : "";*  *if($date<date('Y-m-d')){*  *$calendar.="<td><h4>$currentDay</h4> <button class='btn btn-danger btn-xs'>Н/Д</button>";*  *}*  *elseif(in\_array($date, $bookings)){*  *$calendar.="<td class='$today'><h4>$currentDay</h4> <button class='btn btn-danger btn-xs'>Забронировано</button>";*  *}*  *else{*  *$calendar.="<td class='$today'><h4>$currentDay</h4> <a href='book4.php?date=".$date."' class='btn btn-success btn-xs'>Забронировать</a>";*  *}*    *$calendar .="</td>";*    *$currentDay++;*  *$dayOfWeek++;*  *}*    *if ($dayOfWeek != 7) {*    *$remainingDays = 7 - $dayOfWeek;*  *for($l=0;$l<$remainingDays;$l++){*  *$calendar .= "<td class='empty'></td>";*  *}*  *}*    *$calendar .= "</tr>";*  *$calendar .= "</table>";*  *echo $calendar;*  *}*  *?>*  *<html>*  *<head>*  *<meta name="viewport" content="width=device-width, initial-scale=1.0">*  *<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css">*  *<link rel="stylesheet" href="../assets/main.css">*  *<style>*  *@media only screen and (max-width: 760px), (min-device-width: 802px) and (max-device-width: 1020px) {*  *table, thead, tbody, th, td, tr {*  *display: block;*  *}*  *.empty {*  *display: none;*  *}*  *th {*  *position: absolute;*  *top: -9999px;*  *left: -9999px;*  *}*  *tr {*  *border: 1px solid #ccc;*  *}*  *td {*  *border: none;*  *border-bottom: 1px solid #eee;*  *position: relative;*  *padding-left: 50%;*  *}*  *td:nth-of-type(1):before {*  *content: "Sunday";*  *}*  *td:nth-of-type(2):before {*  *content: "Monday";*  *}*  *td:nth-of-type(3):before {*  *content: "Tuesday";*  *}*  *td:nth-of-type(4):before {*  *content: "Wednesday";*  *}*  *td:nth-of-type(5):before {*  *content: "Thursday";*  *}*  *td:nth-of-type(6):before {*  *content: "Friday";*  *}*  *td:nth-of-type(7):before {*  *content: "Saturday";*  *}*  *}*  *h1{*  *text-align: center;*  *}*  */\* Под смартфоны \*/*  *@media only screen and (min-device-width: 320px) and (max-device-width: 480px) {*  *body {*  *padding: 0;*  *margin: 0;*  *}*  *}*  */\* Под айпады \*/*  *@media only screen and (min-device-width: 802px) and (max-device-width: 1020px) {*  *body {*  *width: 495px;*  *}*  *}*  *@media (min-width:641px) {*  *table {*  *table-layout: fixed;*  *}*  *td {*  *width: 33%;*  *}*  *}*    *.row{*  *margin-top: 20px;*  *}*    *.today{*  *background:yellow;*  *}*  *</style>*  *</head>*  *<body>*  *<div class="container">*  *<div class="row">*  *<div class="col-md-12">*  *<h1>Бронирование для Дома №4, на:</h1>*  *<?php*  *$dateComponents = getdate();*  *if(isset($\_GET['month']) && isset($\_GET['year'])){*  *$month = $\_GET['month'];*  *$year = $\_GET['year'];*  *}else{*  *$month = $dateComponents['mon'];*  *$year = $dateComponents['year'];*  *}*  *echo build\_calendar($month,$year);*  *?>*  *<form action="" method="post" autocomplete="off">*  *<button class="btn btn-primary" type="submit" name="bit">Назад</button>*  *</form>*  *</div>*  *</div>*  *</div>*  *</body>*  *</html>* |

Листинг 3.17 – index5.php

|  |
| --- |
| <?php  if(isset($\_POST['bit'])){  header('Location: ../../homes3.html');  }  function build\_calendar($month, $year) {  $mysqli = new mysqli('localhost', 'root', '', 'booking-system');  $stmt = $mysqli->prepare("SELECT \* FROM room5 WHERE MONTH(date) = ? AND YEAR(date) = ?");  $stmt->bind\_param('ss', $month, $year);  $bookings = array();  if($stmt->execute()){  $result = $stmt->get\_result();  if($result->num\_rows>0){  while($row = $result->fetch\_assoc()){  $bookings[] = $row['date'];  }    $stmt->close();  }  }  $daysOfWeek = array('Воскресенье','Понедельник','Вторник','Среда','Четверг','Пятница','Суббота');  $firstDayOfMonth = mktime(0,0,0,$month,1,$year);  $numberDays = date('t',$firstDayOfMonth);  $dateComponents = getdate($firstDayOfMonth);  $monthName = $dateComponents['month'];  $dayOfWeek = $dateComponents['wday'];  $datetoday = date('Y-m-d');    $calendar = "<table class='table table-bordered'>";  $calendar .= "<center><h2>$month - $year</h2>";  $calendar.= "<a class='btn btn-xs btn-primary' href='?month=".date('m', mktime(0, 0, 0, $month-1, 1, $year))."&year=".date('Y', mktime(0, 0, 0, $month-1, 1, $year))."'><=</a> ";    $calendar.= " <a class='btn btn-xs btn-primary' href='?month=".date('m')."&year=".date('Y')."'>Текущий месяц</a> ";    $calendar.= "<a class='btn btn-xs btn-primary' href='?month=".date('m', mktime(0, 0, 0, $month+1, 1, $year))."&year=".date('Y', mktime(0, 0, 0, $month+1, 1, $year))."'>=></a></center><br>";    $calendar .= "<tr>";  foreach($daysOfWeek as $day) {  $calendar .= "<th class='header'>$day</th>";  }  $currentDay = 1;  $calendar .= "</tr><tr>";  if ($dayOfWeek > 0) {  for($k=0;$k<$dayOfWeek;$k++){  $calendar .= "<td class='empty'></td>";  }  }    $month = str\_pad($month, 2, "0", STR\_PAD\_LEFT);    while ($currentDay <= $numberDays) {  if ($dayOfWeek == 7) {  $dayOfWeek = 0;  $calendar .= "</tr><tr>";  }    $currentDayRel = str\_pad($currentDay, 2, "0", STR\_PAD\_LEFT);  $date = "$year-$month-$currentDayRel";    $dayname = strtolower(date('l', strtotime($date)));  $eventNum = 0;  $today = $date==date('Y-m-d')? "today" : "";  if($date<date('Y-m-d')){  $calendar.="<td><h4>$currentDay</h4> <button class='btn btn-danger btn-xs'>Н/Д</button>";  }  elseif(in\_array($date, $bookings)){  $calendar.="<td class='$today'><h4>$currentDay</h4> <button class='btn btn-danger btn-xs'>Забронировано</button>";  }  else{  $calendar.="<td class='$today'><h4>$currentDay</h4> <a href='book5.php?date=".$date."' class='btn btn-success btn-xs'>Забронировать</a>";  }    $calendar .="</td>";    $currentDay++;  $dayOfWeek++;  }    if ($dayOfWeek != 7) {    $remainingDays = 7 - $dayOfWeek;  for($l=0;$l<$remainingDays;$l++){  $calendar .= "<td class='empty'></td>";  }  }    $calendar .= "</tr>";  $calendar .= "</table>";  echo $calendar;  }  ?>  <html>  <head>  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css">  <link rel="stylesheet" href="../assets/main.css">  <style>  @media only screen and (max-width: 760px), (min-device-width: 802px) and (max-device-width: 1020px) {  table, thead, tbody, th, td, tr {  display: block;  }  .empty {  display: none;  }  th {  position: absolute;  top: -9999px;  left: -9999px;  }  tr {  border: 1px solid #ccc;  }  td {  border: none;  border-bottom: 1px solid #eee;  position: relative;  padding-left: 50%;  }  td:nth-of-type(1):before {  content: "Sunday";  }  td:nth-of-type(2):before {  content: "Monday";  }  td:nth-of-type(3):before {  content: "Tuesday";  }  td:nth-of-type(4):before {  content: "Wednesday";  }  td:nth-of-type(5):before {  content: "Thursday";  }  td:nth-of-type(6):before {  content: "Friday";  }  td:nth-of-type(7):before {  content: "Saturday";  }  }  h1{  text-align: center;  }  /\* Под смартфоны \*/  @media only screen and (min-device-width: 320px) and (max-device-width: 480px) {  body {  padding: 0;  margin: 0;  }  }  /\* Под айпады \*/  @media only screen and (min-device-width: 802px) and (max-device-width: 1020px) {  body {  width: 495px;  }  }  @media (min-width:641px) {  table {  table-layout: fixed;  }  td {  width: 33%;  }  }    .row{  margin-top: 20px;  }    .today{  background:yellow;  }  </style>  </head>  <body>  <div class="container">  <div class="row">  <div class="col-md-12">  <h1>Бронирование для Дома №5, на:</h1>  <?php  $dateComponents = getdate();  if(isset($\_GET['month']) && isset($\_GET['year'])){  $month = $\_GET['month'];  $year = $\_GET['year'];  }else{  $month = $dateComponents['mon'];  $year = $dateComponents['year'];  }  echo build\_calendar($month,$year);  ?>  <form action="" method="post" autocomplete="off">  <button class="btn btn-primary" type="submit" name="bit">Назад</button>  </form>  </div>  </div>  </div>  </body>  </html> |

Также для каждой страницы с календарем нужна форма которая будет собирать данные в соответствующие таблицы БД, для дальнейшего отображения бронирований в профиле. Для этого перейдем в файлы book1.php …book5.php.

Листинг 3.18 – book1.php.

|  |
| --- |
| *<?php*  *if(isset($\_GET['date'])){*  *$date = $\_GET['date'];*  *}*  *if(isset($\_POST['submit'])){*  *$name = $\_POST['name'];*  *$phone = $\_POST['phone'];*  *$email = $\_POST['email'];*  *$mysqli = new mysqli('localhost', 'root', '', 'booking-system');*  *$stmt = $mysqli->prepare("INSERT INTO room1 (fio, phone, email, date) VALUES (?,?,?,?)");*  *$stmt->bind\_param('ssss', $name, $phone, $email, $date);*  *$stmt->execute();*  *$msg = "<div class='alert alert-success'>Бронирование успешно!</div>";*  *$stmt->close();*  *$mysqli->close();*  *}*  *if(isset($\_POST['exitsub'])){*  *header('Location: index1.php');*  *}*  *?>*  *<!doctype html>*  *<html lang="en">*  *<head>*  *<meta charset="UTF-8">*  *<meta http-equiv="X-UA-Compatible" content="ie=edge">*  *<meta name="viewport" content="width=device-width, initial-scale=1.0">*  *<title></title>*  *<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" crossorigin="anonymous">*  *<link rel="stylesheet" href="../assets/main.css">*  *</head>*  *<body>*  *<div class="container">*  *<h1 class="text-center">Забронировать дату: <?php echo date('d/m/Y', strtotime($date)); ?></h1><hr>*  *<div class="row">*  *<div class="col-md-6 col-md-offset-3">*  *<?php echo isset($msg)?$msg:''; ?>*  *<form action="" method="post" autocomplete="off">*  *<div class="form-group">*  *<label for="">Имя</label>*  *<input type="text" class="form-control" name="name">*  *</div>*  *<div class="form-group">*  *<label for="">Телефон</label>*  *<input type="text" class="form-control" name="phone">*  *</div>*  *<div class="form-group">*  *<label for="">Почта</label>*  *<input type="email" class="form-control" name="email">*  *</div>*  *<button class="btn btn-primary" type="submit" name="submit">Забронировать</button><br>*  *<button class="btn btn-primary" type="submit" name="exitsub">Назад</button>*  *</form>*  *</div>*  *</div>*  *</div>*  *<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>*  *<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-Tc5IQib027qvyjSMfHjOMaLkfuWVxZxUPnCJA7l2mCWNIpG9mGCD8wGNIcPD7Txa" crossorigin="anonymous"></script>*  *</body>*  *</html>* |

Листинг 3.19 – book2.php.

|  |
| --- |
| <?php  if(isset($\_GET['date'])){  $date = $\_GET['date'];  }  if(isset($\_POST['submit'])){  $name = $\_POST['name'];  $phone = $\_POST['phone'];  $email = $\_POST['email'];  $mysqli = new mysqli('localhost', 'root', '', 'booking-system');  $stmt = $mysqli->prepare("INSERT INTO room2 (fio, phone, email, date) VALUES (?,?,?,?)");  $stmt->bind\_param('ssss', $name, $phone, $email, $date);  $stmt->execute();  $msg = "<div class='alert alert-success'>Бронирование успешно!</div>";  $stmt->close();  $mysqli->close();  }  if(isset($\_POST['exitsub'])){  header('Location: index2.php');  }  ?>  <!doctype html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta http-equiv="X-UA-Compatible" content="ie=edge">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title></title>  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" crossorigin="anonymous">  <link rel="stylesheet" href="../assets/main.css">  </head>  <body>  <div class="container">  <h1 class="text-center">Забронировать дату: <?php echo date('d/m/Y', strtotime($date)); ?></h1><hr>  <div class="row">  <div class="col-md-6 col-md-offset-3">  <?php echo isset($msg)?$msg:''; ?>  <form action="" method="post" autocomplete="off">  <div class="form-group">  <label for="">Имя</label>  <input type="text" class="form-control" name="name">  </div>  <div class="form-group">  <label for="">Телефон</label>  <input type="text" class="form-control" name="phone">  </div>  <div class="form-group">  <label for="">Почта</label>  <input type="email" class="form-control" name="email">  </div>  <button class="btn btn-primary" type="submit" name="submit">Забронировать</button><br>  <button class="btn btn-primary" type="submit" name="exitsub">Назад</button>  </form>  </div>  </div>  </div>  <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-Tc5IQib027qvyjSMfHjOMaLkfuWVxZxUPnCJA7l2mCWNIpG9mGCD8wGNIcPD7Txa" crossorigin="anonymous"></script>  </body>  </html> |

Листинг 3.20 – book3.php.

|  |
| --- |
| *<?php*  *if(isset($\_GET['date'])){*  *$date = $\_GET['date'];*  *}*  *if(isset($\_POST['submit'])){*  *$name = $\_POST['name'];*  *$phone = $\_POST['phone'];*  *$email = $\_POST['email'];*  *$mysqli = new mysqli('localhost', 'root', '', 'booking-system');*  *$stmt = $mysqli->prepare("INSERT INTO room3 (fio, phone, email, date) VALUES (?,?,?,?)");*  *$stmt->bind\_param('ssss', $name, $phone, $email, $date);*  *$stmt->execute();*  *$msg = "<div class='alert alert-success'>Бронирование успешно!</div>";*  *$stmt->close();*  *$mysqli->close();*  *}*  *if(isset($\_POST['exitsub'])){*  *header('Location: index3.php');*  *}*  *?>*  *<!doctype html>*  *<html lang="en">*  *<head>*  *<meta charset="UTF-8">*  *<meta http-equiv="X-UA-Compatible" content="ie=edge">*  *<meta name="viewport" content="width=device-width, initial-scale=1.0">*  *<title></title>*  *<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" crossorigin="anonymous">*  *<link rel="stylesheet" href="../assets/main.css">*  *</head>*  *<body>*  *<div class="container">*  *<h1 class="text-center">Забронировать дату: <?php echo date('d/m/Y', strtotime($date)); ?></h1><hr>*  *<div class="row">*  *<div class="col-md-6 col-md-offset-3">*  *<?php echo isset($msg)?$msg:''; ?>*  *<form action="" method="post" autocomplete="off">*  *<div class="form-group">*  *<label for="">Имя</label>*  *<input type="text" class="form-control" name="name">*  *</div>*  *<div class="form-group">*  *<label for="">Телефон</label>*  *<input type="text" class="form-control" name="phone">*  *</div>*  *<div class="form-group">*  *<label for="">Почта</label>*  *<input type="email" class="form-control" name="email">*  *</div>*  *<button class="btn btn-primary" type="submit" name="submit">Забронировать</button><br>*  *<button class="btn btn-primary" type="submit" name="exitsub">Назад</button>*  *</form>*  *</div>*  *</div>*  *</div>*  *<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>*  *<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-Tc5IQib027qvyjSMfHjOMaLkfuWVxZxUPnCJA7l2mCWNIpG9mGCD8wGNIcPD7Txa" crossorigin="anonymous"></script>*  *</body>*  *</html>* |

Листинг 3.21 – book4.php.

|  |
| --- |
| <?php  if(isset($\_GET['date'])){  $date = $\_GET['date'];  }  if(isset($\_POST['submit'])){  $name = $\_POST['name'];  $phone = $\_POST['phone'];  $email = $\_POST['email'];  $mysqli = new mysqli('localhost', 'root', '', 'booking-system');  $stmt = $mysqli->prepare("INSERT INTO room4 (fio, phone, email, date) VALUES (?,?,?,?)");  $stmt->bind\_param('ssss', $name, $phone, $email, $date);  $stmt->execute();  $msg = "<div class='alert alert-success'>Бронирование успешно!</div>";  $stmt->close();  $mysqli->close();  }  if(isset($\_POST['exitsub'])){  header('Location: index4.php');  }  ?>  <!doctype html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta http-equiv="X-UA-Compatible" content="ie=edge">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title></title>  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" crossorigin="anonymous">  <link rel="stylesheet" href="../assets/main.css">  </head>  <body>  <div class="container">  <h1 class="text-center">Забронировать дату: <?php echo date('d/m/Y', strtotime($date)); ?></h1><hr>  <div class="row">  <div class="col-md-6 col-md-offset-3">  <?php echo isset($msg)?$msg:''; ?>  <form action="" method="post" autocomplete="off">  <div class="form-group">  <label for="">Имя</label>  <input type="text" class="form-control" name="name">  </div>  <div class="form-group">  <label for="">Телефон</label>  <input type="text" class="form-control" name="phone">  </div>  <div class="form-group">  <label for="">Почта</label>  <input type="email" class="form-control" name="email">  </div>  <button class="btn btn-primary" type="submit" name="submit">Забронировать</button><br>  <button class="btn btn-primary" type="submit" name="exitsub">Назад</button>  </form>  </div>  </div>  </div>  <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-Tc5IQib027qvyjSMfHjOMaLkfuWVxZxUPnCJA7l2mCWNIpG9mGCD8wGNIcPD7Txa" crossorigin="anonymous"></script>  </body>  </html> |

Листинг 3.22 – book5.php.

|  |
| --- |
| *<?php*  *if(isset($\_GET['date'])){*  *$date = $\_GET['date'];*  *}*  *if(isset($\_POST['submit'])){*  *$name = $\_POST['name'];*  *$phone = $\_POST['phone'];*  *$email = $\_POST['email'];*  *$mysqli = new mysqli('localhost', 'root', '', 'booking-system');*  *$stmt = $mysqli->prepare("INSERT INTO room5 (fio, phone, email, date) VALUES (?,?,?,?)");*  *$stmt->bind\_param('ssss', $name, $phone, $email, $date);*  *$stmt->execute();*  *$msg = "<div class='alert alert-success'>Бронирование успешно!</div>";*  *$stmt->close();*  *$mysqli->close();*  *}*  *if(isset($\_POST['exitsub'])){*  *header('Location: index5.php');*  *}*  *?>*  *<!doctype html>*  *<html lang="en">*  *<head>*  *<meta charset="UTF-8">*  *<meta http-equiv="X-UA-Compatible" content="ie=edge">*  *<meta name="viewport" content="width=device-width, initial-scale=1.0">*  *<title></title>*  *<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" crossorigin="anonymous">*  *<link rel="stylesheet" href="../assets/main.css">*  *</head>*  *<body>*  *<div class="container">*  *<h1 class="text-center">Забронировать дату: <?php echo date('d/m/Y', strtotime($date)); ?></h1><hr>*  *<div class="row">*  *<div class="col-md-6 col-md-offset-3">*  *<?php echo isset($msg)?$msg:''; ?>*  *<form action="" method="post" autocomplete="off">*  *<div class="form-group">*  *<label for="">Имя</label>*  *<input type="text" class="form-control" name="name">*  *</div>*  *<div class="form-group">*  *<label for="">Телефон</label>*  *<input type="text" class="form-control" name="phone">*  *</div>*  *<div class="form-group">*  *<label for="">Почта</label>*  *<input type="email" class="form-control" name="email">*  *</div>*  *<button class="btn btn-primary" type="submit" name="submit">Забронировать</button><br>*  *<button class="btn btn-primary" type="submit" name="exitsub">Назад</button>*  *</form>*  *</div>*  *</div>*  *</div>*  *<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>*  *<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-Tc5IQib027qvyjSMfHjOMaLkfuWVxZxUPnCJA7l2mCWNIpG9mGCD8wGNIcPD7Txa" crossorigin="anonymous"></script>*  *</body>*  *</html>* |

## 3.5 Результат работы сайта

### 3.5.1 Страница авторизации

При нажатии на кнопку «Войти» на главной странице сайта, пользователя встречает форма авторизации. На данной странице размещены два поля для ввода: «Логин», «Пароль», и кнопка «Войти», а также две ссылки на прилегающие страницы с формами.

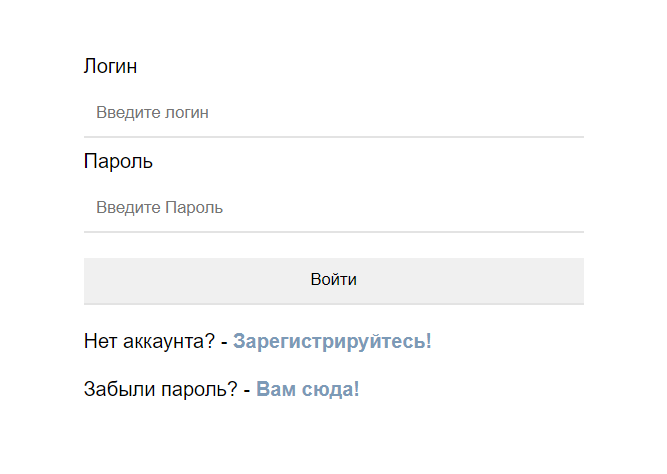


Рис. 3.12 – Форма авторизации.

При нажатии кнопки «Войти» если логин и пароль были введены правильно, то пользователь попадет на страницу профиля. В ином случае сервер выведет соответствующую ошибку: «Неверный логин» или «Неверный пароль».

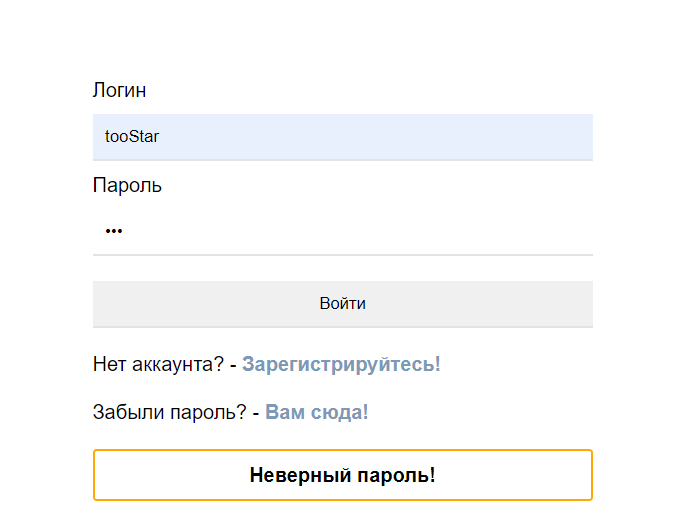


Рис. 3.13 – Сообщение об ошибке

При нажатии на ссылку «Зарегестрируйтесь!» проиходит переадресация на страницу регистрации.

При нажатии на ссылку «Вам сюда!» происходит переадресация на страницу подтверждению почты.

Пользователь не сможет попасть на эту страницу, будучи уже авторизованным. Скрипт автоматически переадресует его на страницу профиля.

### 3.5.2 Страница регистрации

Когда пользователь попадает на страницу регистрации, его встречает форма, которая состоит из шести полей для ввода: «ФИО», «Логин», «Email», «Пароль», «Повторите пароль», «Дата рождения» и кнопка «Зарегистрироваться», а также ссылка на прилегающую страницу авторизации с формой.

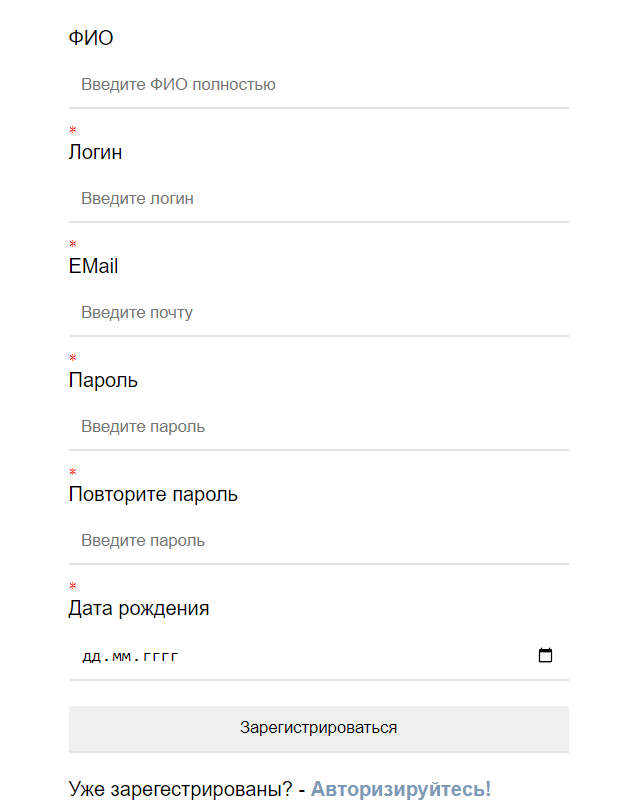


Рис. 3.14 – Форма регистрации

После нажатия на единственную кнопку скрипт обработает данные введенные пользователем и выведет сообщение об удачной или неудачной регистрации.

Условия для удачной регистрации:

1. Логин пользователя уникален;
2. Email пользователя уникален;
3. Первый и повторно введенный пароли совпадают;
4. Все поля формы заполнены.

Если пользователь ввел все данные верно, то он попадет на страницу авторизации и получит сообщение об удачной регистрации. Скрипт же в свою очередь занесет все данные пользователя в БД в соответствующую таблицу «users»:

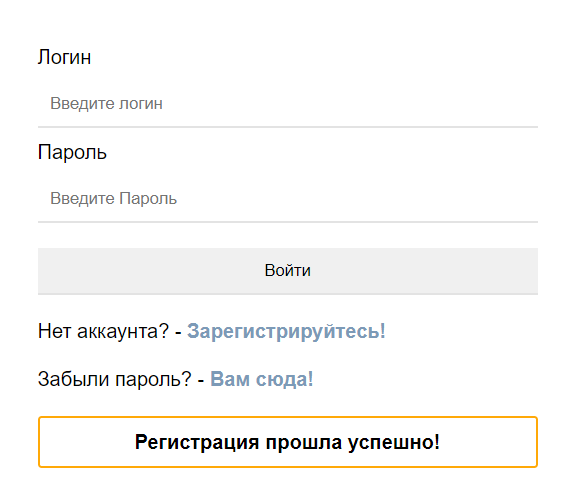


Рис. 3.15 – Успешная регистрация

В случае, если пользователь введет данные не верно, то он останется на странице регистрации и получит соответствующее сообщение об ошибке:

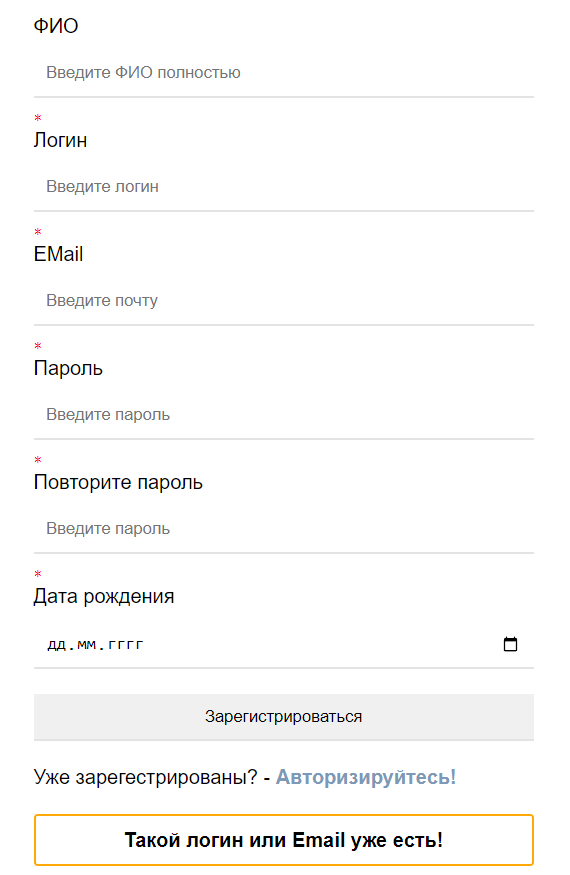


Рис. 3.16 – Ошибка регистрации

При удачной регистрации пользователь получает возможность авторизироваться используя только что созданный аккаунт и попасть на страницу своего профиля.

Пользователь не сможет попасть на эту страницу, будучи уже авторизованным, скрипт автоматически переадресует его на страницу профиля.

### 3.5.3 Страница профиля

Как только пользователь попадает на страницу своего профиля он может увидеть часть информации которую он вводил при регистрации. Скрипт достал данные от соответствующего профиля из БД и вывел в соответствии с заданными параметрами стиля.

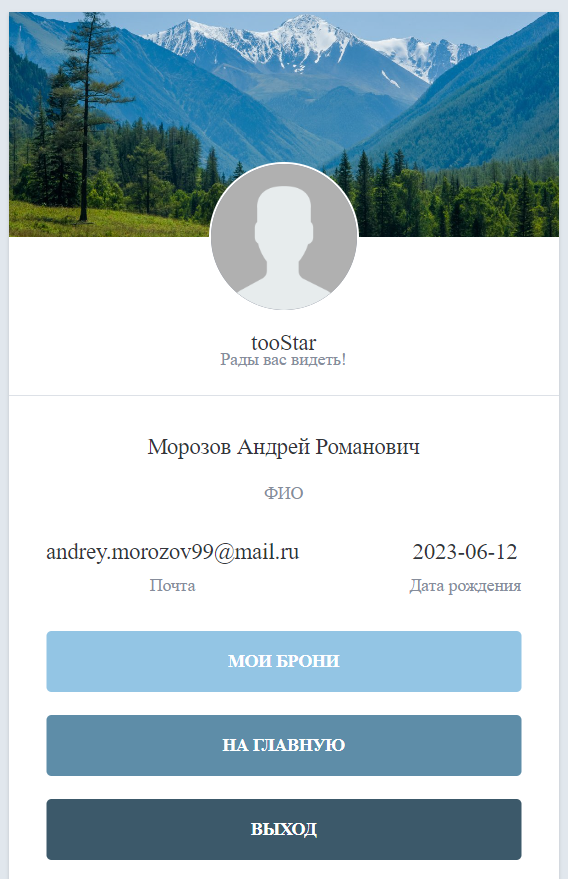


Рис. 3.17 – Страница профиля («заглушка»)

В профиле присутствуют три кнопки: «Мои брони», «На главную» и «Выход».

Кнопка «Мои брони» отсылает пользователя на страницу со всеми забронированными датами.

Кнопка «На главную» отсылает пользователя на главную страницу сайта.

Кнопка «Выход» выполняет скрипт удаляющий из памяти пользователя и отсылающий его на главную страницу сайта, благодаря чему он сможет вновь попасть на страницу авторизации.

Также аватар профиля является ссылкой на страницу загрузки пользовательского аватара. До тех пор пока пользователь не заменит аватар, будет использоваться стандартная «заглушка».

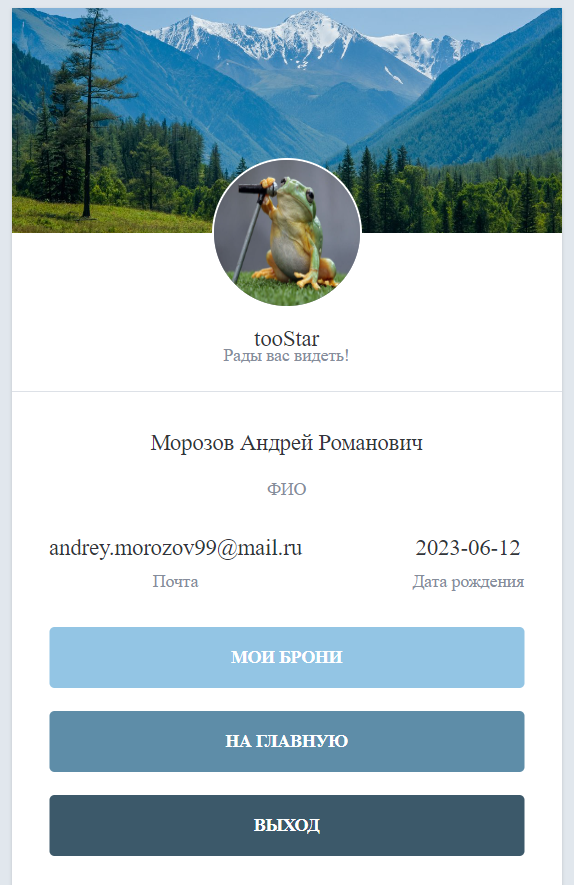


Рис. 3.18 – Страница профиля (пользовательский аватар)

Пользователь не сможет попасть на эту страницу, будучи не авторизованным, скрипт автоматически переадресует его на страницу авторизации.

### 3.5.4 Страница изменения аватара

Пользователь попадает на эту страницу из своего профиля нажав по аватарке.

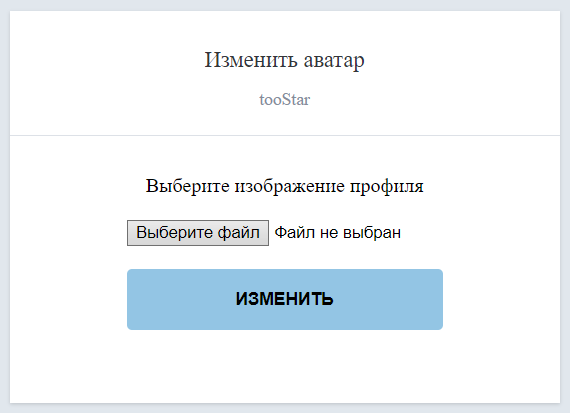


Рис. 3.19 – Страница изменения аватара

Пользователь может загрузить любую фотографию через соответствующую кнопку «Выберите файл» скрипт автоматически подстроит фотографию под место в профиле и сохранит её на сервере. После выбора фото и нажатия на кнопку «Изменить» пользователя переадресует на страницу профиля с уже измененной аватаркой.

Пользователь не сможет попасть на эту страницу будучи не авторизованным, скрипт автоматически переадресует его на страницу авторизации.

### 3.5.5 Страница бронирований (профиль)

Данная страница встречает пользователя информацией о том на какие даты он совершил бронирования. Скрипт проверяет каждую таблицу в БД на наличие совпадений и выводит соответствующие даты.

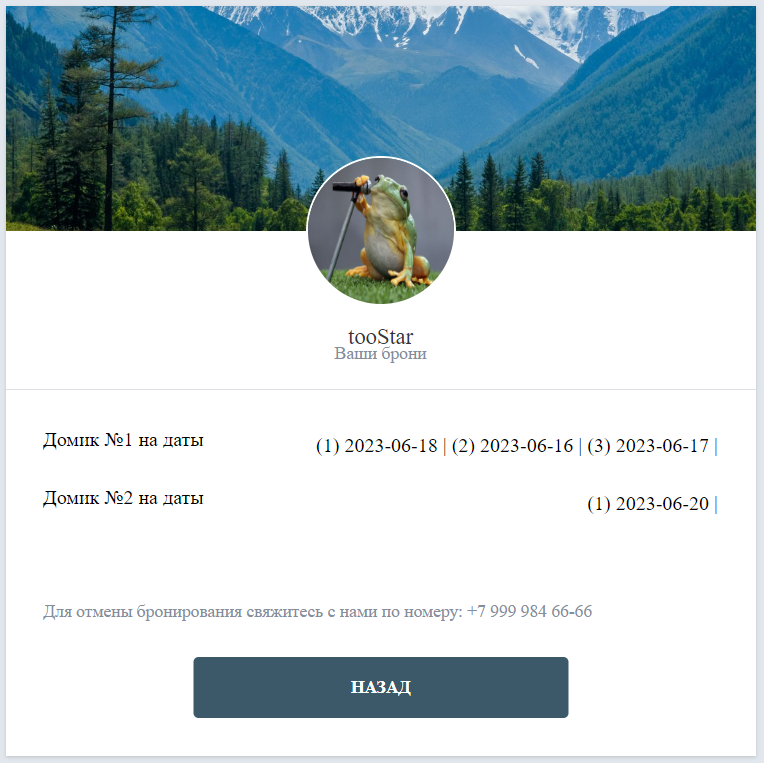


Рис. 3.20 – Страница бронирований

Пользователь не сможет попасть на эту страницу будучи не авторизованным, скрипт автоматически переадресует его на страницу авторизации.

### 3.5.6 Страница (календарь) бронирования

После того как пользователь выбрал понравившийся ему дом, он может нажать на кнопку «Забронировать» и попадет в календарь бронирования для соответствующего дома:



Рис. 3.21 – Календарь бронирования

В этом календаре пользователь может перемещаться по ближайшим месяцам, присутствует кнопка для возврата на текущий месяц, также здесь имеется вся информация по уже забронированным датам всех пользователей. На каждую дату имеется кнопка «Забронировать» если эта дата свободна, а также заблокированная кнопка «Забронировано» если эта дата занята. Кнопка «Назад» вернет пользователя на страницу выбора домов.

### 3.5.7 Страница (форма) бронирования

На эту страницу пользователь попадет если нажмет кнопку «Забронировать» в календаре бронирований. В данной форме присутствует три поля: «Имя», «Телефон» и «Почта». При нажатии на кнопку на кнопку «Назад» пользователя переадресует на календарь. Скрипт заносит введенную в форме информацию в соответсвующую таблицу БД для дальнейшего использования.

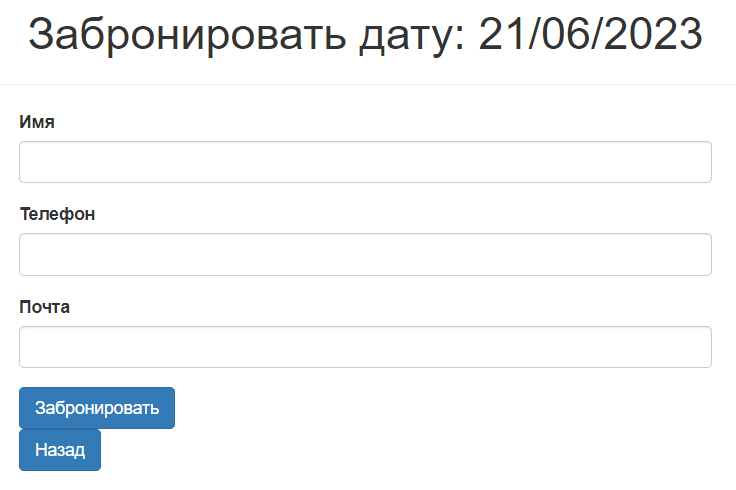


Рис. 3.22 – Форма бронирования

# ЗАКЛЮЧЕНИЕ

В данной дипломной работе был реализован проект серверной части сайта базы отдыха АлтайДача22. Сайт реализован для персональных компьютеров и ноутбуков. Основные результаты зафиксированы и приведены в тексте дипломной работы. Для выполнения работы использовались современные инфокоммуникационные средства: персональный компьютер, языки программирования и разметки, современное программное обеспечение, а также работа в сети Интернет. Характеристика предметной области описываются в первой главе проекта, обзор и выбор средств разработки во второй главе, а структура проекта и его реализация в третьей главе. Подводя итоги проведенной работы, можно сказать, что разработка Веб-сайтов является весьма важным направлением, в особенности для частного бизнеса. Каждой уважающей себя компании нужен веб-сайт. Даже если у вас высоко конкурентная ниша, благодаря сайту вы сможете быть доступнее. Создание сайтов для малых компаний не только поможет им получить дешёвую саморекламу, но и увеличит интерес потребителей.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. *Робсон Э., Фримен Э.* Изучаем HTML XHTML и CSS / Э. Робсон, Э. Фримен // O’Reilly. – 2-е изд., стер. – Санкт-Петербург: Питер, 2019. – 720 с.
2. *Олищук А.В*. Разработка Web-приложений на PHP 5. / А. В. Олищук // Профессиональная работа. — М.: «Вильямс», 2006. — С. 352. — ISBN 5-8459-0944-9.
3. *Робин Никсон.* Создаем динамические веб-сайты, с помощью PHP, MySQL, JavaScript, CSS и HTML5 / Робин Никсон // O’Reilly. – 5-е изд., стер. – Санкт-Петербург: Питер, 2019. – 816с (дата обращения: 21.05.2023).
4. *К. В. Ахтырченко, Т. П. Сорокваша.* Методы и технологии реинжиниринга ИС / К. В. Ахтырченко, Т. П. Сорокваша — Текст : электронный // citforum : [сайт]. — URL: <http://citforum.ru/SE/project/isr/>, (дата обращения: 13.05.2023).
5. Средства и методологии проектирования, разработки и сопровождения клиент-серверных приложений. — Текст : электронный // citforum : [сайт]. — URL: [http://citforum.ru/cfin/prcorpsys/infsistpr\_05.shtml](https://vk.com/away.php?utf=1&to=http%3A%2F%2Fcitforum.ru%2Fcfin%2Fprcorpsys%2Finfsistpr_05.shtml), (дата обращения: 29.05.2023).