

Министерство образования и молодежной политики  
Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного  
строительства»

Специальность 09.02.07: «Информационные системы и  
программирование»

Разработка веб-сайта для «Фитнес-клуба»

**Пояснительная записка**

к курсовому проекту

КР-ПР-41-02-2026-ПЗ

Разработала:

Студентка гр. ПР-41

\_\_\_\_\_ / Э.А. Горелова

Руководитель

\_\_\_\_\_ / Д.О. Гарiev

2026

Министерство образования и молодежной политики  
Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного  
строительства»

Специальность 09.02.07: «Информационные системы и  
программирование»

Разработка веб-сайта для «Фитнес-клуба»

**Курсовой проект**

КП-ПР-41-02-2026-ПЗ

2026

## Оглавление

ВВЕДЕНИЕ.....	4
1 Постановка задачи.....	5
1.1    Описание предметной области .....	5
1.1.1    Сущность и структура предметной области .....	6
1.1.2    Бизнес-процессы предметной области .....	7
1.1.3    Проблемы в предметной области.....	8
1.2    Требования к программному продукту .....	9
1.2.1    Требования к функциональным характеристикам .....	9
1.2.2    Состав выполняемых функций.....	10
1.2.3    Организация входных и выходных данных .....	11
1.2.4    Анализ аналогов.....	11
2 Проектирование .....	14
2.1 Проектирование логической структуры .....	14
2.2 Проектирование физической структуры .....	15
2.2.2 Описание примерной файловой структуры проекта .....	17
3 Разработка и тестирование .....	19
3.1    Реализация модулей.....	19
3.2 Тестирование программного продукта .....	21
Заключение .....	27
Список использованной литературы.....	28

Инв. № подп	Подп. и дата	Лит	Изм.	№ докум.	Подп.	Дата	КП-ПР-41-02-2026-П3		
							Лит	Лист	Листов
		Разраб.	.						
		Пров.	Гареев Д.О.					3	32
		Н. контр.	Гареев Д.О.						
		Утв.	Гареев Д.О.						
							Разработка веб сайта для «Фитнес-клуба»		

## ВВЕДЕНИЕ

Веб-сайт для фитнес-клуба — это не просто витрина услуг, а комплексное решение, которое объединяет множество функциональных модулей для обеспечения эффективного взаимодействия между клубом и его клиентами. Разработка такого проекта требует глубокого понимания принципов интеграции, что как раз и является основной задачей курсового проектирования.

Целью данного курсового проекта является закрепление и углубление знаний интеграции программных модулей посредством разработки полнофункционального веб-приложения для фитнес-клуба. В ходе выполнения проекта формируется комплексное представление об этапах, начиная от проектирования архитектуры до финальной синхронизации всех компонентов.

Для достижения поставленной цели необходимо решить следующие задачи:

- проанализировать предметную область и существующие решения;
- разработать требования к программному продукту;
- создать базу данных;
- реализовать основные функциональные модули.

Актуальность данного проекта заключается в том, что разработка интегрированного веб-приложения позволяет овладеть навыками, необходимыми для работы в профессиональной сфере, где интеграция модулей — один из ключевых этапов создания сложных информационных систем. Веб-сайт фитнес-клуба служит отличным примером реального бизнес-приложения, требующего синхронизации и слаженной работы множества компонентов для обеспечения благоприятного пользовательского опыта.

Инв. № подп	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата	Лист	4
					КП-ПР-41-02-2026-ПЗ	

# 1 Постановка задачи

Фитнес-клубы сегодня активно внедряют цифровые технологии для автоматизации бизнес-процессов и улучшения клиентского опыта. Однако во многих клубах отсутствует удобный и функциональный онлайн-сервис, позволяющий пользователям быстро получать информацию, записываться на тренировки и управлять своими абонементами. В связи с этим возникла задача создания веб-сайта, который не только демонстрировал бы услуги клуба, но и обеспечивал бы интерактивное взаимодействие с клиентами и сотрудниками.

## 1.1 Описание предметной области

Предметная область — это сфера услуг фитнес-клуба, которая включает в себя несколько ключевых бизнес-процессов:

- регистрация и управление клиентами: сбор данных пользователя (ФИО, контакты, предпочтения), хранение информации об активности и статусе абонемента;
- расписание занятий: формирование и отображение актуального расписания с указанием видов тренировок, времени, тренеров и доступных мест;
- запись на тренировки: пользователи должны иметь возможность онлайн-бронировать места на групповые и персональные занятия с подтверждением или уведомлениями о статусе заявки;
- административный контроль: интерфейсы для администраторов и тренеров, позволяющие обновлять данные о расписании, клиентах и отчетности, автоматизация шаблонной работы с договорами.

Подп. и дата	
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	

Ли						Лист
Изм.						
№ докум.		Подп.	Дата			5

### 1.1.1 Сущность и структура предметной области

Предметная область проекта охватывает деятельность современного фитнес-клуба, ориентированного на предоставление услуг по организации тренировок и управлению клиентским опытом. Ключевая проблема заключается в отсутствии единого информационного пространства, связывающего клиентов, тренеров и администраторов, что приводит к неэффективному использованию ресурсов и снижению удовлетворенности клиентов.

В работе фитнес-клуба участвуют несколько категорий пользователей, у каждого из которых свои задачи и функции:

- клиент — посетитель клуба, который хочет заниматься спортом, посещать тренировки и получать фитнес-услуги;
- администратор — сотрудник, управляющий расписанием, записью клиентов, контролем оплаты и организацией работы персонала;
- тренер — специалист, проводящий групповые или персональные тренировки, ведёт учет посещаемости и оценивает успехи клиентов;
- система оплаты — отвечает за обработку транзакций и управление платежными данными клиентов.

Ключевые сущности:

- профиль клиента — содержит персональные данные, историю посещений, детали абонемента и результаты тренировок;
- абонемент — документ, дающий право посещать услуги клуба в течение определённого времени или по количеству посещений;
- расписание занятий — актуальный календарь тренировок с указанием времени, тренера и типа занятия;
- заявка на тренировку — запись клиента на конкретное занятие с указанием статуса (подтверждена, ожидает, отменена);
- платёжная транзакция — информация о внесённой оплате, дате, сумме и способе оплаты;

Подп. и дата	
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	
Инв. № подп	

Ли	Изм.	№ докум.	Подп.	Дата	Лист	6
					КП-ПР-41-02-2026-ПЗ	

- уведомления — сообщения для клиентов о подтверждении записи, сменах расписания и акциях клуба.

### 1.1.2 Бизнес-процессы предметной области

Регистрация нового клиента: начинается с создания аккаунта на сайте, где пользователь вводит свои личные данные. Система проводит автоматическую проверку корректности предоставленной информации и создает профиль клиента. После успешной регистрации пользователь получает возможность выбрать подходящий тип абонемента и произвести оплату выбранных услуг через платежную систему.

Управление расписанием: осуществляется администратором клуба, который добавляет новые занятия и вносит корректировки в существующее расписание. Тренера также имеют доступ к системе, где они могут отмечать присутствие клиентов и вносить определенные изменения в свое рабочее расписание.

Онлайн-запись на тренировки: предоставляет клиентам возможность просматривать актуальное расписание и выбирать подходящие занятия, система проверяет доступность свободных мест и подтверждает.

Оплата услуг: способы расчетов, включают банковские карты и электронные кошельки, после успешной обработки платежа система автоматически обновляет статус абонемента клиента.

Управление абонементами и посещаемостью: включает ведение учета посещений клиентов, который осуществляется тренерами и администраторами. Система регулярно информирует пользователей о текущем статусе их абонемента и сроке его действия, по окончании срока действия абонемента доступ к услугам клуба автоматически блокируется до момента продления.

Подп. и дата	
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	
Инв. № подп	

Ли	Изм.	№ докум.	Подп.	Дата	Лист
					7

Обратная связь и информирование клиентов: обеспечивает регулярную рассылку уведомлений о новых курсах, специальных акциях и изменениях в расписании тренировок.

### 1.1.3 Проблемы в предметной области

Системы онлайн-бронирования сопровождается рядом операционных сложностей, которые негативно влияют на ключевые бизнес-показатели:

Ручное управление записью: администраторы тратят значительное время на прием телефонных звонков и обработку запросов на бронирование;

Низкая прозрачность расписания: клиенты не имеют доступа к актуальной информации о свободных местах, часто записываются на уже заполненные занятия, что приводит к конфликтным ситуациям и недовольству;

Ошибки при ручном учете: при ведении записей в журналах или Excel-таблицах высока вероятность человеческих ошибок: двойное бронирование, путаница во времени, потеря информации;

Неэффективное использование ресурсов: без системы аналитики сложно отследить популярность направлений и тренеров, что приводит к неоптимальному формированию расписания и недогрузке ресурсов клуба;

Отсутствие автоматизированных уведомлений: клиенты не получают напоминаний о предстоящих занятиях, что увеличивает процент неявок, и администраторам приходится вручную обзванивать записавшихся;

Сложности с контролем платежей и абонементов: отслеживание статусов абонементов, сроков их действия и задолженностей ведется вручную, что приводит к финансовым потерям и неконтролируемому доступу в клуб.

Подп. и дата	
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	

Ли				
Изм.				
№ докум.				

## **1.2 Требования к программному продукту**

Программный продукт должен представлять собой систему для автоматизации работы фитнес-клуба, обеспечивающую удобное взаимодействие клиентов, тренеров и администраторов через личные кабинеты с ролевым доступом. Система обязана поддерживать функционал онлайн-записи на тренировки с проверкой доступности мест и абонементов, управление расписанием, учёт посещаемости, ведение абонементов.

### **1.2.1 Требования к функциональным характеристикам**

#### **Управление пользователями и ролями**

В системе будут три типа пользователей: клиенты, тренеры и администраторы. У каждого своя страница входа и свои возможности. Все смогут зарегистрироваться, указать свои данные, войти по email и паролю, а потом менять профиль, например, обновить телефон или загрузить фото. Администратор сможет управлять всеми аккаунтами: добавлять новых пользователей, править их данные или временно отключать.

#### **Управление тренировками и расписанием**

Тренеры смогут создавать тренировки: указывать название, описание, дату, время и сколько человек может записаться. Они будут видеть своё расписание в удобном виде со списком, кто уже записался. Клиенты увидят актуальное расписание на сайте. Если что-то поменяется сразу отобразится у всех.

#### **Система бронирования и записи**

Клиенты смогут записываться на тренировки прямо с сайта, если есть свободные места и у них действующий абонемент. Программа сама проверит, доступна ли тренировка и спишет одно посещение с абонемента. Если клиент передумает, он сможет отменить запись, и место снова станет свободным. После записи ему придёт подтверждение.

Подп. и дата
Взам. инв. №
Инв. № дубл.
Подп. и дата
Инв. № подп

Ли	Изм.	№ докум.	Подп.	Дата

## **Управление абонементами и посещаемостью**

В системе будут разные абонементы: на месяц, 3 месяца, 6 месяцев и год. Тренер на тренировке сможет отметить, кто пришёл, а кто нет, и даже оставить заметку по клиенту. Как только тренер отмечает посещение, клиент видит остаток.

## **Формирование отчётности и аналитика**

Тренеры увидят свою статистику, сколько тренировок провели, какая посещаемость, сколько разных клиентов пришло и какой средний рейтинг. Администраторы получат более полные отчёты, по продажам абонементов, загруженности залов, активности тренеров и общим показателям клуба.

## **Администрирование и мониторинг системы**

Администратор сможет следить за активностью. Все важные действия в системе будут записываться, чтобы при необходимости можно было понять, кто и что сделал.

В итоге моя программа покроет все основные процессы в фитнес-клубе, от записи на тренировку до отчётов для руководства. Она упростит работу сотрудников, сделает услуги понятнее для клиентов и поможет клубу работать более организованно и эффективно.

### **1.2.2 Состав выполняемых функций**

Управление пользователями и аутентификация: безопасный доступ к системе через регистрацию, авторизацию и восстановление пароля. Разделение ролей на Клиента, Тренера и Администратора позволяет дифференцировать права доступа и функциональные возможности для каждой категории пользователей.

Личный кабинет клиента: доступ к персональной информации, включая редактирование профиля, просмотр действующего абонемента, историю посещений и платежей.

Подп. и дата
Взам. инв. №
Инв. № дубл.
Подп. и дата
Инв. № подп

Ли				
Изм.				
№ докум.	Подп.	Дата		

Модуль расписания и онлайн-записи: отображение тренировок с возможностью фильтрации, функционал онлайн-записи и отмены занятий дополняется проверкой доступности мест в реальном времени и автоматическим контролем статуса абонемента.

Панель администратора: предназначена для комплексного управления всеми аспектами работы клуба, включая работу с пользовательскими данными, формирование расписания, финансовую отчетность и аналитику посещаемости.

Панель тренера: ориентирована на нужды инструкторов, предоставляя доступ к личному расписанию, отметкам посещений и обратной связи от клиентов.

Уведомления: автоматическая отправка email-уведомлений о подтверждении записи, напоминании о занятии, изменениях в расписании.

### 1.2.3 Организация входных и выходных данных

Входные данные:

- Данные форм регистрации и авторизации (логин, пароль, email);
- Параметры расписания (дата, тренер);
- Данные для создания/редактирования заявки на запись;

Выходные данные:

- расписание, личный кабинет;
- проверка мест, запись;
- Электронные письма (уведомления);
- Отчеты в панели администратора (графики, таблицы).

### 1.2.4 Анализ аналогов

WorldClass - разработан для известной сети фитнес-клубов. Его основным преимуществом является удобная система бронирования групповых занятий,

Подп. и дата	Подп. и дата
Взам. инв. №	Взам. инв. №
Инв. № дубл.	Инв. № дубл.
Подп. и дата	Подп. и дата

Ли				
Изм.				
№ докум.				
Подп.				
Дата				

однако продукт обладает ограниченным функционалом для небольших клубов и характеризуется высокой стоимостью (Рисунок 1).

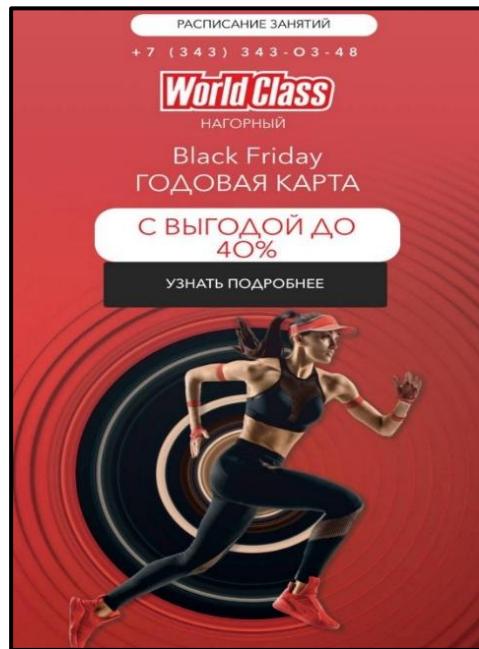


Рисунок 1. Сайт «WorldClass»

DriveFitness - привлекает пользователей простым и понятным интерфейсом, а также налаженной системой связи с администраторами. К существенным недостаткам можно отнести отсутствие встроенной системы отзывов о тренерах и возможности онлайн-записи на тренировки (Рисунок 2).



Рисунок 2. Сайт «DriveFitness»

Инв. № подп	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-02-2026-ПЗ

Лист  
12

BriteFitness - отличается современным дизайном и подробной информацией, однако в системе отсутствуют важные функции напоминаний о занятиях и отображения информации о свободных местах в зале (Рисунок 3).

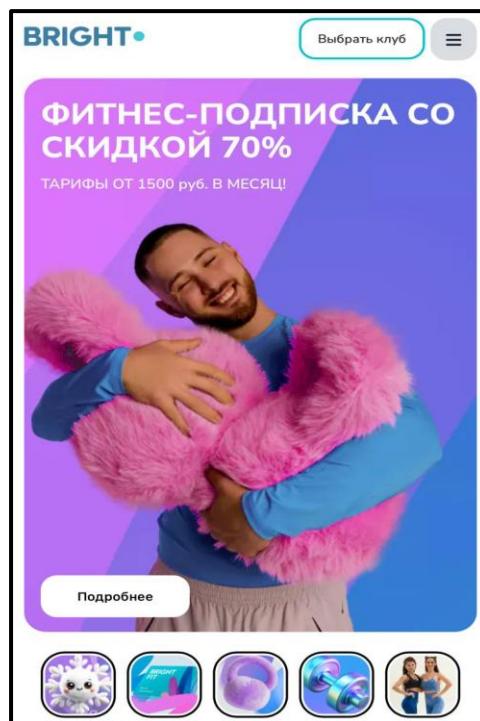


Рисунок 3. Сайт «BrightFitness»

Разрабатываемая система учитывает все выявленные недостатки существующих решений. В нашем приложении будет реализована удобная система онлайн-записи с отображением свободных мест, простой в использовании интерфейс для администраторов и клиентов, интеграция с платежными системами, автоматические уведомления о занятиях и изменениях в расписании. Будет доступное решение, которое подойдет как небольшим фитнес-студиям, так и крупным клубам, объединяя в себе лучшие функции аналогов без их недостатков.

Подп. и дата	
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	

Ли	Изм.	№ докум.	Подп.	Дата

## 2 Проектирование

Проектирование — это этап создания. Здесь определяется, как будет выглядеть сайт с точки зрения пользователя, какие технологии будут использоваться для его создания и как будут организованы данные.

### 2.1 Проектирование логической структуры

Система поддерживает три основные роли: клиент, тренер и администратор, при этом каждая роль имеет свой набор прав и интерфейс. В качестве этого была разработана диаграмма прецедентов, представленная на Рисунок 4.

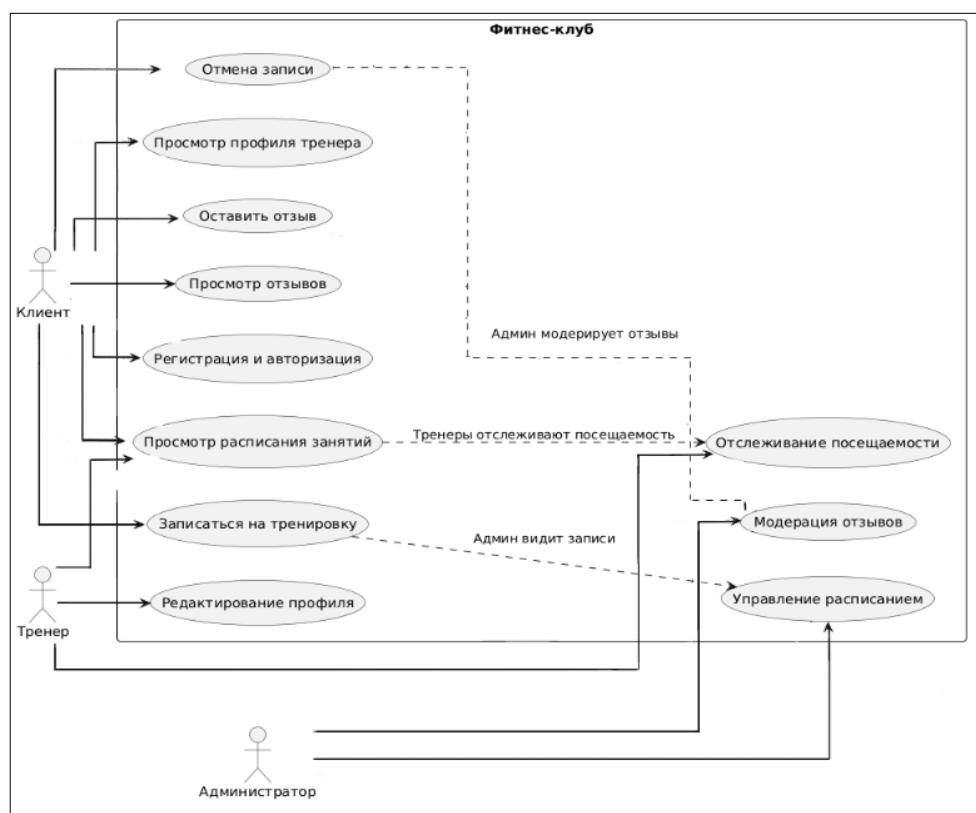


Рисунок 4- Диаграмма «Прецендентов»

На диаграмме четко видны три основных типа пользователей и их действия:

Пользователь регистрируется через форму, вводя email, пароль и личные данные. После регистрации он может просматривать доступные тренировки и

Инв. № подп	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата	Лист	14
					KП-ПР-41-02-2026-ПЗ	

актуальное расписание занятий. Система позволяет записываться на групповые тренировки, автоматически проверяя наличие свободных мест. Пользователь управляет своим абонементом и отслеживает количество оставшихся посещений. Для него доступна история посещений и прогресс тренировок, а также он получает уведомления о предстоящих занятиях и изменениях в расписании.

Тренер, как специалист, проводящий тренировки, создаёт и редактирует расписание своих занятий с указанием времени и вместимости. Он управляет параметрами тренировок, включая дату, время начала и окончания, а также максимальное количество участников. После проведения занятия тренер отмечает посещаемость клиентов и просматривает детальный список записавшихся с их контактными данными. Для персонализации подхода он ведёт индивидуальные заметки о клиентах. Тренер также получает статистику по проведённым занятиям, посещаемости и рейтингу, а также управляет своим профилем, добавляя описание, специализацию и фото.

Администратор управляет всей системой и контролирует работу клуба. Он управляет общим расписанием клуба, в его обязанности входит контроль работы тренеров и активности клиентов, генерация аналитических отчётов по посещаемости, популярности занятий и выручке. Администратор также управляет абонементами клиентов, создавая, продлевая и блокируя их, контролирует платежи и финансовые транзакции, а также при необходимости редактирует права доступа пользователей.

## 2.2 Проектирование физической структуры

Физическая структура веб-сайта фитнес-клуба основана на классической клиент-серверной модели. Для создания сайта были выбраны проверенные и широко распространённые технологии: серверная часть написана на PHP, для хранения информации используется база данных MySQL. Интерфейс для пользователей построен на стандартных технологиях - HTML, CSS и JavaScript.

Подп. и дата	
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	
Инв. № подп	

Ли	Изм.	№ докум.	Подп.	Дата	Лист	15
					КП-ПР-41-02-2026-ПЗ	

Это сделало проект быстрым, надёжным и удобным для размещения в сравнении с аналогами, например:

React хорошо работает там, где данные меняются каждый раз, но, например для записи на тренировки это не обязательно, в результате сайт бы работал с большими задержками.

WordPress подходит для рекламы. Чтобы добавить запись на тренировку и личные кабинеты, пришлось бы ставить много дополнений. От этого сайт стал бы медленным и менее безопасным.

Tilda позволяет быстро собрать красивую страницу, но нельзя сделать своё расписание или добавить базу клиентов. Можно использовать только то, что уже есть в шаблонах.

Создавать приложение через Android Studio долго и сложно, приложение нужно скачивать и обновлять.

Клиентская часть, или фронтенд, создана с помощью базовых веб-инструментов. Каждая ключевая функция, например, просмотр расписания или личный кабинет, представляет собой отдельную HTML-страницу с понятной структурой. Единый стиль для всех страниц задаётся с помощью CSS, где для цветов и размеров используются переменные — это позволяет быстро менять оформление сайта. Интерактивность, вроде отправки формы или обмена данными с сервером, реализована на JavaScript, что делает работу с сайтом плавной, без постоянных перезагрузок. Дизайн сайта адаптирован под разные устройства - от компьютеров до смартфонов - благодаря технологиям CSS.

Все данные системы - информация о пользователях, расписании, абонементах и посещениях - хранятся в реляционной базе данных MySQL. Для безопасного взаимодействия с ней серверный код использует подготовленные выражения, что надёжно защищает от внешних атак.

Основные сущности связаны между собой через внешние ключи, что обеспечивает целостность данных и эффективность выполнения запросов. Для

Подп. и дата
Взам. инв. №
Инв. № дубл.
Подп. и дата
Инв. № подп

Ли				
Изм.				
№ докум.	Подп.	Дата		

детального описания структуры данных, которая поддерживает всю эту логику, была разработана ER-диаграмма, изображённая на Рисунок 5.

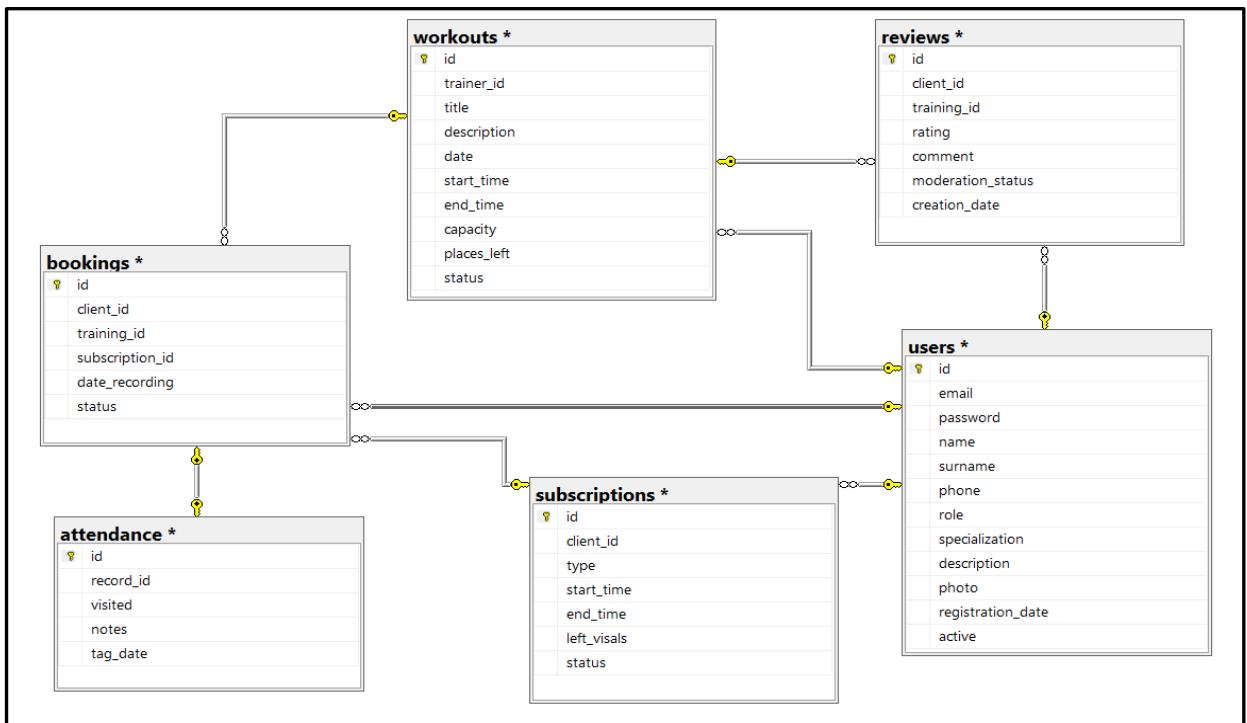


Рисунок 5-«ER-диаграмма»

Основные сущности системы включают таблицу **users**, которая хранит данные всех пользователей, таблицу **subscriptions** для описания абонементов клиентов, таблицу **workouts** для данных о тренировках, таблицу **bookings** для реализации процесса записи, таблицу **attendance** для фиксации факта посещения и таблицу **reviews** для хранения отзывов клиентов. Связи между сущностями организованы по принципу «один ко многим» и «многие ко многим» через таблицу-связку **bookings**, что позволяет эффективно отражать реальные бизнес-процессы фитнес-клуба.

## 2.2.2 Описание примерной файловой структуры проекта

Для наглядного описания организации исходных файлов, которая реализует эту логику, была разработана схема файловой структуры проекта, изображённая на Рисунок 6.

Лист						
Инв. № подп	Подп. и дата	Инв. № дубл	Подп. и дата	Взам. инв. №	Подп. и дата	Подп. и дата
Ли	Изм.	№ докум.	Подп.	Дата		

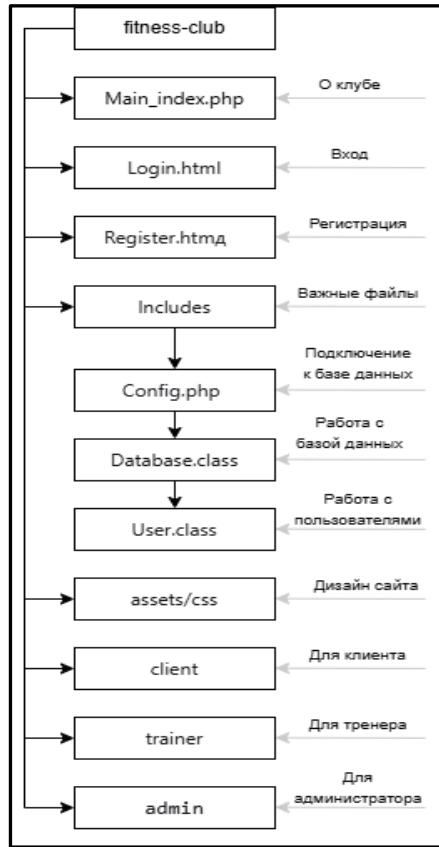


Рисунок 6 – Файловая структура

Верху расположены основные страницы сайта: главная, вход и регистрация. Ниже находится папка Includes с важными рабочими файлами для работы с базой данных и пользователями. Дизайн всего сайта прописан в папке assets/css. Внизу представлена структура с тремя отдельными папками: для клиента, тренера и администратора.

Инв. № подп	Подп. и дата
Инв. № дубл.	Взам. инв. №
	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

### 3 Разработка и тестирование

Разработка и тестирование — это этап, на котором идеи реализовываются в работающий код, а затем проверяются на ошибки.

#### 3.1 Реализация модулей

Реализация модулей — это процесс написания кода для отдельных частей системы (вход, регистрация, запись).

##### Вход в систему

Система на Рисунок 7 умеет находить токен пользователя несколькими способами: через адресную строку, сохраненные данные браузера или специальные заголовки запроса. Это удобно - можно зайти с компьютера, с мобильного приложения или по специальной ссылке.

```
public function getCurrentUser(): array|bool|null
{
    $token = null;

    if (isset($_GET['token'])) {
        $token = $_GET['token'];
    }

    if (!$token && isset($_COOKIE['fitness_token'])) {
        $token = $_COOKIE['fitness_token'];
    }

    if (!$token) {
        $headers = getallheaders();
        if (isset($headers['Authorization'])) {
            $authHeader = $headers['Authorization'];
            if (strpos(haystack: $authHeader, needle: 'Bearer ') === 0) {
                $token = substr(string: $authHeader, offset: 7);
            }
        }
    }

    if (!$token) {
        return null;
    }

    try {
        $decoded = JWT::decode(token: $token);
        if (!$decoded || !isset($decoded['user_id'])) {
            return null;
        }

        return $this->getById(id: $decoded['user_id']);
    } catch (Exception $e) {
        error_log(message: "Invalid token: " . $e->getMessage());
        return null;
    }
}
```

Рисунок 7 - Поиск токена пользователя

Инв. № подп	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

## Безопасное работа с запросами в базе данных

Система на Рисунок 8 защищает базу данных от взлома через специальный метод, который обрабатывает все запросы безопасным способом. Даже если злоумышленник попытается ввести в форму команду OR 1=1 (обход защиты), это не сработает.

```
public function executeQuery($sql, $params = []): bool|mysqli_stmt
{
    $stmt = $this->connection->prepare(query: $sql);

    if (!stmt) {
        throw new Exception(message: "Ошибка подготовки запроса: " . $this->connection->error);
    }

    if (!empty($params)) {
        $types = '';
        foreach ($params as $param) {
            if (is_int(value: $param)) {
                $types .= 'i';
            } elseif (is_float(value: $param)) {
                $types .= 'd';
            } else {
                $types .= 's';
            }
        }

        $stmt->bind_param(types: $types, ...var: &$params);
    }

    if (!$stmt->execute()) {
        throw new Exception(message: "Ошибка выполнения запроса: " . $stmt->error);
    }

    return $stmt;
}
```

Рисунок 8 - Безопасное выполнение запросов

## Безопасная работа с запросами в базе данных

При регистрации системы на Рисунок 9 тщательно проверяет все введённые данные, чтобы избежать ошибок. Процесс проходит через несколько этапов: сначала проверяется заполнение обязательных полей - email, пароль, имя и фамилия. Затем система проверяет корректность email-адреса, убеждаясь, что он соответствует формату "пример@почта.ру". Далее проверяется, не зарегистрирован ли уже такой email в системе, чтобы исключить дублирование учётных записей.

Подп. и дата
Взам. инв. №
Инв. № дубл.
Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

Пароль пользователя автоматически шифруется, превращаясь в нечитаемый код для безопасного хранения. Система назначает роль - клиент, тренер или администратор, при этом по умолчанию устанавливается роль клиента. Все текстовые данные очищаются от лишних пробелов. В завершение создаётся учётная запись пользователя и сразу выдаётся токен для входа, что позволяет новому пользователю начать работу в системе.

```

public function register($data): array
{
    $required = ['email', 'password', 'first_name', 'last_name'];
    foreach ($required as $field) {
        if (empty($data[$field])) {
            throw new Exception(message: "Поле '$field' обязательно для заполнения");
        }
    }
    if (!filter_var(value: $data['email'], filter: FILTER_VALIDATE_EMAIL)) {
        throw new Exception(message: "Некорректный email адрес");
    }
    $existing = $this->db->fetchOne(sql: "SELECT id FROM users WHERE email = ?", params: [$data['email']]);
    if ($existing) {
        throw new Exception(message: "Пользователь с таким email уже существует");
    }
    $password_hash = password_hash(password: $data['password'], algo: PASSWORD_BCRYPT);
    $role = $data['role'] ?? 'client';
    $allowed_roles = ['client', 'trainer', 'admin'];
    if (!in_array(needle: $role, haystack: $allowed_roles)) {
        $role = 'client';
    }
    $userData = [
        'email' => trim(string: $data['email']),
        'password_hash' => $password_hash,
        'role' => $role,
        'first_name' => trim(string: $data['first_name']),
        'last_name' => trim(string: $data['last_name']),
        'phone' => isset($data['phone']) ? trim(string: $data['phone']) : null
    ];
    $user_id = $this->db->insert(table: 'users', data: $userData);
    $token = $this->generateToken(user_id: $user_id, email: $data['email'], role: $role);
    $user = $this->getById(id: $user_id);
    return [
        'user' => $user,
        'token' => $token
    ];
}

```

Рисунок 9 - "Умная" регистрация

### 3.2 Тестирование программного продукта

Тестирование проводилось для сайта фитнес-клуба, которое включает в себя работу с базой данных и взаимодействие с клиентской частью. Тестирование

Подп. и дата	
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	

нужно для того, чтобы проверить корректность работы ключевых модулей системы.

## Тестирование методом "Черного ящика" - Функциональность авторизации

Объект тестирования:

Страница авторизации (login.html)

Входные данные:

Email (строковое значение), пароль (строковое значение)

Выходные данные:

- Успешная авторизация с перенаправлением на соответствующий интерфейс (клиент/тренер/админ)

- Сообщение об ошибке: "Пользователь не найден"
- Сообщение об ошибке: "Неверный пароль"
- Сообщение об ошибке: "Email обязателен"
- Сообщение об ошибке: "Пароль обязателен"
- Сетевая ошибка: "Сетевая ошибка. Попробуйте позже"
- Сообщение при валидации полей ввода

Условия:

1. Email должен соответствовать формату email
2. Пользователь должен существовать в базе данных
3. Пароль должен совпадать с хэшем в базе данных
4. Пользователь должен быть активным (active=1)
5. Должен генерироваться JWT токен при успешной авторизации

Правильные данные:

1. Email соответствует формату email (valid@example.com)

Инв. № подп	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

2. Пользователь существует в базе данных
3. Пароль корректен
4. Пользователь активен
5. Генерация токена успешна

Неправильные данные:

1. Email не соответствует формату (not-an-email)
2. Пользователь не существует в базе данных
3. Неверный пароль
4. Пользователь заблокирован (active=0)
5. Ошибка генерации токена
6. Пустой email
7. Пустой пароль
8. SQL-инъекция в поле email

№	Входные данные	Ожидаемый результат	Проверяемые условия
1	Email: admin@fitness.ru  Пароль: (правильный)	A) Успешная авторизация, перенаправление на /admin/index.php	1,2,3,4,5
2	Email: trainer1@fitness.ru  Пароль: (правильный)	A) Успешная авторизация, перенаправление на /trainer/index.php	1,2,3,4,5

Подп. и дата
Взам. инв. №
Инв. № дубл.
Подп. и дата

3	Email: client1@fitness.ru  Пароль: (правильный)	A) Успешная авторизация, перенаправление на /client/index.php	1,2,3,4,5
4	Email: no@test.ru  Пароль: anypassword	Б) "Пользователь не найден"	2
5	Email: admin@fitness.ru  Пароль: wrongpass	В) "Неверный пароль"	3
6	Email: (пусто)  Пароль: pass1234	Г) "Email обязателен"	6
7	Email: admin@fitness.ru  Пароль: (пусто)	Д) "Пароль обязателен"	7
8	Email: (не формат email)  Пароль: anypassword	Б) "Пользователь не найден"	1
9	Email: admin@fitness.ru' OR '1'='1  Пароль: pass1234	Ж) Сообщение при валидации полей ввода	8

Результат теста на Рисунок 10: все проверки системы авторизации пройдены.

Инв. № подп	Подп. и дата	Инв. № дубл	Подп. и дата	Взам. инв. №	Подп. и дата
Ли	Изм.	№ докум.	Подп.	Дата	Лист

КП-ПР-41-02-2026-ПЗ

24

```

PS C:\Users\eivin\source\repos> cd C:\OSPanel\home\fitness-and-tests
PS C:\OSPanel\home\fitness-and-tests> php test_login_blackbox.php

=====
TESTIROVANIE СИСТЕМЫ АВТОРИЗАЦИИ
=====

1. Проверка файлов системы...
- login.html: OK
- api/login.php: OK
- classes/User.php: OK
- classes/JWT.php: OK

2. Проверка формы логина...
- Форма логина: OK
- Поле email: OK
- Поле password: OK
- Кнопка отправки: OK

3. Проверка API логина...
- JSON обработка: OK
- Класс User: OK
- Ответ JSON: OK

4. Проверка перенаправлений...
- Роль Клиент: OK
- Роль Тренер: OK
- Роль Админ: OK

=====
ПРОВЕРКА СИСТЕМЫ ЗАВЕРШЕНА
=====

```

Рисунок 10 - Тестирование методом "Черного ящика"

### Тестирование методом "Белого ящика"

№ подп.	Инв. № подп	Подп. и дата	№ подп.	Взам. инв. №	Подп. и дата	Подп. и дата
№	Тестовый сценарий	Код для проверки	Ожидаемый результат			
1	Проверка без инициализации секрета		Пытаемся создать токен без настройки ключа			
2	Декодирование с неверной подписью		Подмена signature части в токене			
3	Токен устарел		Используем токен, срок которого уже вышел			

4	Декодирование неверного формата	Даём токен неправильного формата (например, без подписи)	Ошибка: "Некорректный формат токена"
5	Пустые данные	Создаём токен без информации о пользователе	Должен создаться токен, но без личных данных

Результат теста на Рисунок 11 успешен: все пять тестов пройдены, подтверждая корректную работу, проверки и обработки токенов, включая проверку подписи, формата и срока действия.

```
PS C:\OSPanel\home\fitness-and-tests> php test_jwt.php

=====
TESTIROVANIE JWT KLASCHA
=====

1. Проверка инициализации секрета... OK
2. Кодирование и декодирование... OK
3. Проверка подписи токена... OK
4. Проверка формата токена... OK
5. Проверка срока действия... OK

-----
ИТОГИ:
Пройдено: 5 из 5
СТАТУС: ВСЕ ТЕСТЫ ПРОЙДЕНЫ
=====
```

Рисунок 11 - Тестирование методом "Белого ящика"

Инв. № подп	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

## **Заключение**

В ходе курсовой работы был разработан сайт для фитнес-клуба. Этот проект направлен на то, чтобы сделать работу клуба более современной и удобной для клиентов, тренеров и администраторов.

Сайт позволяет клиентам легко записываться на тренировки онлайн, выбирать удобное время и следить за своим абонементом. У тренеров есть возможность планирования занятий и учета посещений. Администраторы могут контролировать работу клуба, видеть отчеты и управлять данными.

При создании сайта использовались популярные и надежные технологии: PHP, MySQL, HTML, CSS и JavaScript. Все ключевые функции, такие как управление расписанием, онлайн-запись и формирование отчетов, успешно реализованы и протестированы.

В результате получился готовый сайт, который решает ключевые задачи фитнес-клуба: от записи на тренировки до учета и анализа.

Инв. № подп	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

**КП-ПР-41-02-2026-ПЗ**

## Список использованной литературы

1. Дронов В. А. PHP 8, MySQL 8 и HTML 5. Разработка современных динамических веб-сайтов / В. А. Дронов. — Санкт-Петербург, 2023г.
2. Прохоренок Н. А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера / Н. А. Прохоренок, В. В. Дронов. — Санкт-Петербург, 2020г.
3. Дакетт Д. HTML и CSS. Разработка и дизайн веб-сайтов / Д. Дакетт. — Москва, 2020г.
4. Макфарланд Д. JavaScript и jQuery. Д. Макфарланд. — Москва, 2021г.
5. Беляков С. В. Разработка веб-приложений с использованием PHP и MySQL: учебное пособие / С. В. Беляков. — Москва, 2022г.
6. Официальная документация по PHP: <https://www.php.net/manual/ru/>
7. Официальная документация MySQL: <https://dev.mysql.com/doc/>
8. [SQL-EX.RU](https://sql-ex.ru/) - тренажер по SQL. Сайт: <https://sql-ex.ru/>
9. [Metanit.com](https://metanit.com/php/): руководство по PHP. Сайт: <https://metanit.com/php/>
10. Руководство по JavaScript на [JavaScript.ru](https://JavaScript.ru). Сайт: <https://learn.javascript.ru/>
11. Официальный сайт React: <https://react.dev/>
12. Официальный сайт WordPress: <https://developer.wordpress.org/>
13. Официальный сайт Tilda Publishing: <https://tilda.cc/ru/>
14. Официальный сайт Android Studio: <https://developer.android.com/studio>
15. PHPMyAdmin Documentation: <https://docs.phpmyadmin.net/>
16. Официальный сайт фитнес-клуба WorldClass: <https://worldclass-ekb.com/>
17. Официальный сайт фитнес-клуба DriveFitness: <https://drivefitness.ru/>
18. Официальный сайт фитнес-клуба BrightFitness:  
<https://акция.брайтфит.рф>

Инв. № подп	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-02-2026-ПЗ

Лист  
28

## Приложения

Инв. № подп	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-02-2026-ПЗ

Лист  
29

**Приложение А**  
**(код программы)**

Инв. № подп	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата
-------------	--------------	--------------	--------------	--------------

Ли	Изм.	№ докум.	Подп.	Дата
----	------	----------	-------	------

**КП-ПР-41-02-2026-ПЗ**

Лист  
30

Ссылка на GitHub: <https://github.com/Elvina555/fitness-club.git>



Рисунок 12 - QR-код репозитория

Инв. № подп	Подп. и дата

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-02-2026-ПЗ

Лист  
31