МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

"Ижевский государственный технический университет имени М.Т.Калашникова"

(ФГБОУ ВО «ИжГТУ имени М.Т.Калашникова)

"Институт непрерывного профессионального образования"

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА к дипломной работе на тему:

Сайт для боксерского клуба

Дипломник К.К. Климов

студент гр. Д20-191-3

Руководитель Е.А. Зылева

Нормоконтролер В.П. Соболева

Ижевск 2024.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc165661197)

[1. ПРОЕКТИРОВАНИЕ ПРОГРАММОГО ПРОДУКТА 6](#_Toc165661198)

[1.1. Обоснование целесообразности разработки 6](#_Toc165661199)

[1.2. Аналитический обзор существующих программных аналогов 6](#_Toc165661200)

[1.3. Требования к программному продукту 8](#_Toc165661201)

[1.4. Требования к информационной и программной совместимости 9](#_Toc165661202)

[1.5. Планирование тестирования программного продукта 18](#_Toc165661203)

[2. РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА 20](#_Toc165661204)

[2.1. Разработка подсистемы взаимодействия с новостями клуба 20](#_Toc165661205)

[2.2. Разработка подсистемы взаимодействия с отзывами 21](#_Toc165661206)

[2.3. Разработка подсистемы взаимодействия с расписанием 22](#_Toc165661207)

[ЗАКЛЮЧЕНИЕ 23](#_Toc165661208)

[СПИСОК ЛИТЕРАТУРЫ 24](#_Toc165661209)

[ПРИЛОЖЕНИЯ 26](#_Toc165661210)

# ВВЕДЕНИЕ

Сайт для боксерского клуба имеет большую актуальность и необходимость в наше время. Вот несколько причин, почему сайт может быть полезен и актуален для боксерского клуба:

- Предоставление информации: Сайт позволяет предоставить всю необходимую информацию о боксерском клубе, такую как расписание тренировок, контактные данные, информацию о тренерах и спортсменах, а также о проводимых соревнованиях и мероприятиях. Это помогает потенциальным клиентам и посетителям сайта получить все необходимые сведения о клубе.

- Привлечение новых клиентов: Сайт является мощным инструментом привлечения новых клиентов. Посетители могут узнать о преимуществах клуба, увидеть фотографии тренировок и соревнований, прочитать отзывы других клиентов. Это помогает создать положительное впечатление и заинтересовать потенциальных клиентов.

- Улучшение коммуникации: Сайт предоставляет возможность легкой и удобной коммуникации с посетителями. Так же сайт предоставляет возможность обратной связи, где люди могут задавать вопросы, оставлять отзывы или записываться на тренировки. Также сайт даёт возможность использовать электронную почту и социальные сети для поддержания связи с клиентами.

- Повышение видимости в интернете: Сайт позволяет боксерскому клубу быть видимым в интернете. При правильной оптимизации для поисковых систем (SEO) сайт может появиться в результатах поиска, когда люди ищут боксерские клубы в вашем районе. Это помогает привлечь больше потенциальных клиентов и увеличить популярность клуба.

- Обновление информации: Сайт позволяет быстро обновлять информацию о расписании тренировок, ценах, новостях и других важных сведениях. На сайт всегда добавлять новые материалы, фотографии и видео, чтобы посетители всегда были в курсе актуальных событий и предложений.

В целом, сайт для боксерского клуба является важным инструментом для привлечения клиентов, предоставления информации и поддержания связи с посетителями. Он помогает клубу быть видимым в интернете и оставаться актуальным.

Практическая значимость данной работы заключается в повышения навыка программирования выпускника и возможность использования работы в профессиональной деятельности. Материалы, разработанные в ходе выполнения работы, могут служить основой разработки сайтов, связанных со спортивными клубами.

Новизна данного проекта состоит в разработке сайта для боксерского клуба, объединившим в себя функционал добавления информационного контента, а так же возможности получить обратную связь на добавленный контент виде комментариев и отзывов и оставления заявки для записи на занятие.

Цель работы

Целью работы является разработка программного продукта “Боксерский клуб” для более удобного информирования и простой записи на занятие, с использованием современных технологий для разработки сайтов.

Задачи работы:

* Проведение анализа существующих программных продуктов для управления недвижимостью.
* Формулирование требований к программному продукту (техническое задание).
* Проектирование архитектуры программного продукта "Боксерский клуб".
* Разработка функционала для управления контентом на сайте.
* Планирование тестирования программного продукта.
* Подготовка отчета о выполненной работе.

# ПРОЕКТИРОВАНИЕ ПРОГРАММОГО ПРОДУКТА

## Обоснование целесообразности разработки

* + 1. Назначение ПП

Программный продукт предназначен для удобной записи на занятие, для изучения новостей клуба и возможность их комментировать. Функциональное назначение программного продукта является более удобное получение информации о боксерском клубе, а так же более простой записи на занятия.

* + 1. Обоснование цели создания ПП

Продукт создается с целью:

- Обеспечение эффективного информирования пользователей.

- Удобная запись на занятие.

- Создание обратной связи в виде комментариев и отзывов.

## 1.2. Аналитический обзор существующих программных аналогов

В качестве аналогов были выбраны популярные сайты боксерских клубов.

1.2.1. Эверест – спортивный клуб предоставляющий занятия по боксу.

Возможности системы:

- Возможность записаться на первое занятие онлайн.

- Имеется личный кабинет.

- Возможность написать свой отзыв о клубе.

Интерфейс прост и интуитивно понятен. Навигационные элементы достаточного размера.

1.2.2. Динамо – спортивный клуб предоставляющий занятия по боксу.

Возможности системы:

- Возможность записаться на первое занятие онлайн.

- Имеется личный кабинет.

- Возможность оплатить занятие онлайн.

Интерфейс прост и интуитивно понятен. Навигационные элементы достаточного размера.

1.2.3. После уроков – организация предоставляющая дополнительные занятие для школьников, включительно занятия по боксу.

Возможности системы:

- Возможность записаться на первое занятие онлайн.

- Имеется личный кабинет.

- Возможность оплатить занятие онлайн.

Интерфейс крайне не интуитивно понятен. Навигационные элементы очень маленького размера.

1.2.4. Ударник – спортивный клуб предоставляющий занятия по боксу.

Возможности системы:

- Возможность записаться на первое занятие онлайн.

- Возможность изучить новости клуба.

Интерфейс прост и интуитивно понятен. Навигационные элементы достаточного размера.

Аналитический обзор существующих программных аналогов представлен на Таблица 1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Онлайн запись на занятие | Личный кабинет | Отзывы | Новостная страница | Онлайн оплата |
| *Боксерский клуб* | **+** | **+** | **+** | **+** | **-** |
| Эверест | **+** | **+** | **+** | **-** | **-** |
| Динамо | **+** | **+** | **-** | **-** | **+** |
| После уроков | **+** | **+** | **-** | **-** | **+** |
| Ударник | **+** | **-** | **-** | **+** | **-** |

Таблица 1- Аналоги программного продукта

## 1.3. Требования к программному продукту

1.3.1. Требования к функциональным характеристикам

- Система должна давать возможность предоставить информацию о новостях клуба.

- Система должна предоставить возможность оставить отзыв или комментарий.

- Система должна предоставить возможность оправить заявку для посещения первого занятия.

- Система должна предоставить возможность добавить новость.

1.3.2. Требования к надёжности

- Система должна быть устойчива к высоким нагрузкам и запросам.

- Система должна обеспечивать контроль целостности и безопасности пользовательских данных.

- Система должна восстанавливаться после сбоев.

1.3.3. Требования к составу и параметрам технических средств

Программный продукт должен быть совместим с следующими техническими средствами:

- Компьютеры и мобильные устройства, содержащие оперативные системы: Windows 10, Windows 11, Linux, Android, macOS, iOS.

- Браузеры: Google Chrome, Safari, Microsoft Edge, Opera GX.

1.3.4. Требования к информационной и программной совместимости

Программный продукт должен обеспечивать совместимость с различными информационными и программными системами:

- Совместимость с базой данных.

## 1.4. Требования к информационной и программной совместимости

1.4.1. Организация информационной базы

Структура базы данных программного продукта представлена на Рисунок 1.

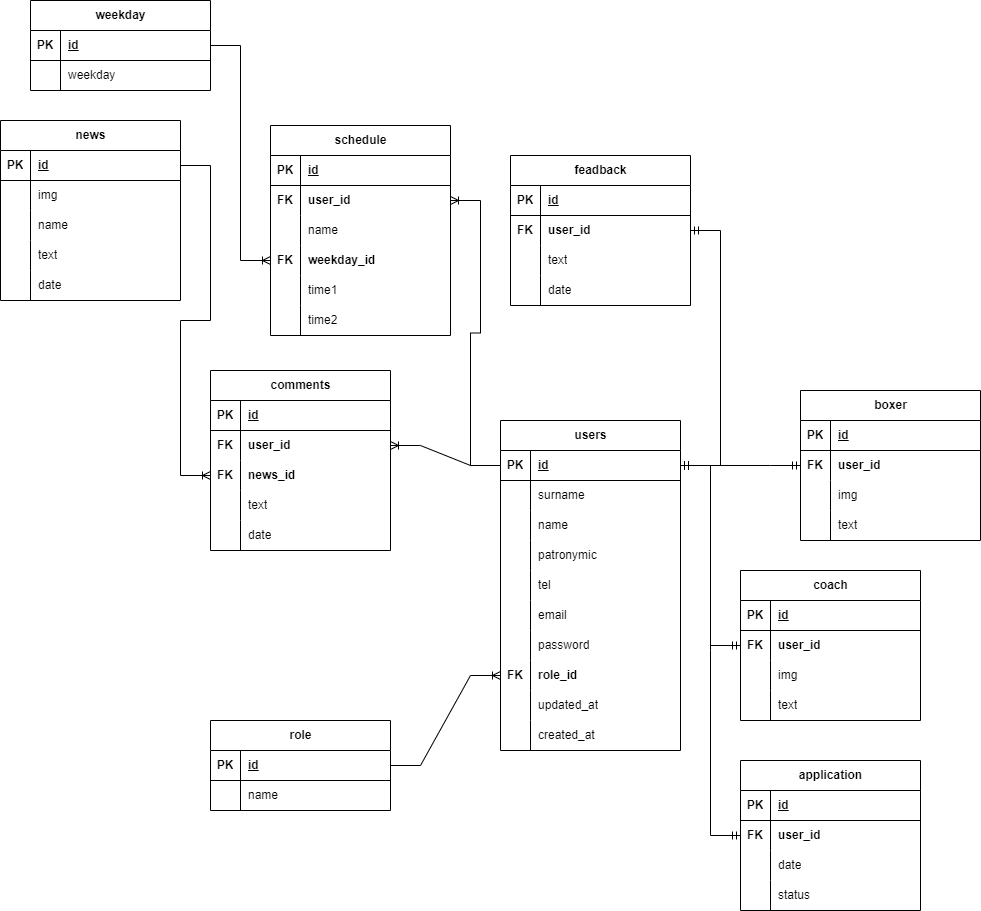


Рисунок 1 - ER-диаграмма базы данных

Сущность users содержит информацию о пользователе.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Название | Тип поля | Размер | Атрибуты | Описание |
| 1 | id | int | От -32 768 до 32 767 | PK, Auto incriment | Уникальный идентификатор |
| 2 | surname | varchar | 255 | Not null | Фамилия |
| 3 | name | varchar | 255 | Not null | Имя |
| 4 | patronymic | varchar | 255 | Not null | Отчество |
| 5 | tel | varchar | 13 | Not null | Номер телефона |
| 6 | email | varchar | 255 | Not null | Почта |
| 7 | password | varchar | 255 | Not null | Пароль |
| 8 | role\_id | int | От -32 768 до 32 767 | FK, not null | Идентификатор роли пользователя |
| 9 | updated\_at | datetime | От 01.01.1753 до 31.12.9999 | Not null | Дата последнего входа в аккаунт |
| 10 | created\_at | datetime | От 01.01.1753 до 31.12.9999 | Not null | Дата создания аккаунта пользователя |

Таблица 2 - Структура сущности users

Сущность role содержит информацию о ролях.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Название | Тип поля | Размер | Атрибуты | Описание |
| 1 | id | int | От -32 768 до 32 767 | PK, Auto incriment | Уникальный идентификатор |
| 2 | name | varchar | 255 | Not null | Название |

Таблица 3 - Структура сущности role

Сущность news содержит информацию о новостях клуба.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Название | Тип поля | Размер | Атрибуты | Описание |
| 1 | id | int | От -32 768 до 32 767 | PK, Auto incriment | Уникальный идентификатор |
| 2 | img | varchar | 255 | Not null | Ссылка на фотографию |
| 3 | name | varchar | 255 | Not null | Название новости |
| 4 | text | text |  | Not null | Описание новости |
| 5 | date | datetime | От 01.01.1753 до 31.12.9999 | Not null | Дата новости |

Таблица 4 - Структура сущности news

Сущность comments содержит информацию о комментариях к новостям.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Название | Тип поля | Размер | Атрибуты | Описание |
| 1 | id | int | От -32 768 до 32 767 | PK, Auto incriment | Уникальный идентификатор |
| 2 | user\_id | int | От -32 768 до 32 767 | FK, Not null | Идентификатор  пользователя |
| 3 | news\_id | int | От -32 768 до 32 767 | FK, Not null | Идентификатор  новости |
| 4 | text | text |  | Not null | Текст комментария |
| 5 | date | datetime | От 01.01.1753 до 31.12.9999 | Not null | Дата комментария |

Таблица 5 - Структура сущности comments

Сущность weekday содержит информацию о тренировочных днях недели.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Название | Тип поля | Размер | Атрибуты | Описание |
| 1 | id | int | От -32 768 до 32 767 | PK, Auto incriment | Уникальный идентификатор |
| 2 | weekday | varchar | 255 | Not null | Список дней недели |

Таблица 6 - Структура сущности weekday

Сущность schedule содержит информацию о расписании тренировок.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Название | Тип поля | Размер | Атрибуты | Описание |
| 1 | id | int | От -32 768 до 32 767 | PK, Auto incriment | Уникальный идентификатор |
| 2 | user\_id | int | От -32 768 до 32 767 | FK, Not null | Идентификатор  пользователя |
| 3 | name | varchar | 255 | Not null | Название группы |
| 4 | weekday\_id | int | От -32 768 до 32 767 | FK, Not null | Идентификатор  тренировочных дней |
| 5 | time1 | time | От 00:00:00 до 23:59:59 | Not null | Начало тренировки |
| 6 | time2 | time | От 00:00:00 до 23:59:59 | Not null | Конец тренировки |

Таблица 7 - Структура сущности schedule

Сущность feedback содержит информацию об отзывах пользователей.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Название | Тип поля | Размер | Атрибуты | Описание |
| 1 | id | int | От -32 768 до 32 767 | PK, Auto incriment | Уникальный идентификатор |
| 2 | user\_id | int | От -32 768 до 32 767 | FK, Not null | Идентификатор  пользователя |
| 3 | text | text |  | Not null | Текст отзыва |
| 4 | date | datetime | От 01.01.1753 до 31.12.9999 | Not null | Дата отзыва |

Таблица 8 - Структура сущности feedback

Сущность boxer содержит информацию о гордости клуба.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Название | Тип поля | Размер | Атрибуты | Описание |
| 1 | id | int | От -32 768 до 32 767 | PK, Auto incriment | Уникальный идентификатор |
| 2 | user\_id | int | От -32 768 до 32 767 | FK, Not null | Идентификатор  пользователя |
| 3 | img | varchar | 255 | Not null | Ссылка на фотографию |
| 4 | text | text |  | Not null | Информация о боксере |

Таблица 9 - Структура сущности boxer

Сущность coach содержит информацию о тренерах.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Название | Тип поля | Размер | Атрибуты | Описание |
| 1 | id | int | От -32 768 до 32 767 | PK, Auto incriment | Уникальный идентификатор |
| 2 | user\_id | int | От -32 768 до 32 767 | FK, Not null | Идентификатор  пользователя |
| 3 | img | varchar | 255 | Not null | Ссылка на фотографию |
| 4 | text | text |  | Not null | Информация о тренере |

Таблица 10 - Структура сущности coach

Сущность application содержит информацию о записях на первое занятие.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Название | Тип поля | Размер | Атрибуты | Описание |
| 1 | id | int | От -32 768 до 32 767 | PK, Auto incriment | Уникальный идентификатор |
| 2 | user\_id | int | От -32 768 до 32 767 | FK, Not null | Идентификатор  пользователя |
| 3 | date | datetime | От 01.01.1753 до 31.12.9999 | Not null | Дата отзыва |
| 4 | status | varchar | 255 | Not null | Статус заявки |

Таблица 11 - Структура сущности application

Между сущностями users и feedback существует связь один к одному по атрибутам: id в сущности users и user\_id в сущности feedback.

Между сущностями users и boxer существует связь один к одному по атрибутам: id в сущности users и user\_id в сущности boxer.

Между сущностями users и coach существует связь один к одному по атрибутам: id в сущности users и user\_id в сущности coach.

Между сущностями users и application существует связь один к одному по атрибутам: id в сущности users и user\_id в сущности application.

Между сущностями users и comments существует связь один ко много по атрибутам: id в сущности users и user\_id в сущности comments.

Между сущностями users и schedule существует связь один ко много по атрибутам: id в сущности users и user\_id в сущности schedule.

Между сущностями users и role существует связь один ко много по атрибутам: id в сущности role и role\_id в сущности users.

Между сущностями news и comments существует связь один ко много по атрибутам: id в сущности news и news\_id в сущности comments.

Между сущностями weekday и schedule существует связь один ко много по атрибутам: id в сущности weekday и weekday\_id в сущности schedule.

1.4.2. Системы программного обеспечения

Для создания и взаимодействия с базой данный будет использоваться СУБД MySQL.

СУБД будет защищена паролей и иметь ограниченный доступ к данным для обеспечения безопасности.

Для функциональности программного продукта “Боксерский клуб” необходимо следующие программное обеспечение:

Операционная система: Windows 10 последней версией.

Трансляторы: php версии 8.0.

Библиотеки программ: Библиотеки для работы с базами данных MySQL.

Специальное программное обеспечение: СУБД MySQL для взаимодействия с базой данных, программы для разработки веб-приложений PHP.

Программы ввода-вывода: Браузер с пользовательским интерфейсом.

Компоненты, не реализуемые средствами общего программного обеспечения: средства для отображения и отображения данных на веб-страницах, такие как HTML, CSS; Фреймфорк Laravel 10 для взаимодействия системы.

## 1.5. Планирование тестирования программного продукта

Процесс верификации

Системное тестирование:

- Проверка соответствия программы функциональным и нефункциональным требованием.

- Тестирование работы основных функций, обработки ошибок, производительности.

Интеграционное тестирование:

- Проверка корректной работы программы в связке с другими компонентами и системами.

- Тестирование взаимодействия с базой данных.

Модульное тестирование:

- Тестирование отдельных модулей программы для проверки их корректности.

- Проверка работы отдельных функций.

Оценка показателей качества программной системы

Описание показателей и требуемых значений атрибутов качества:

- Надёжность: стабильная работа программы без сбоев.

- Производительность: быстрый отклик программы на действия пользователя.

- Удобство использования: интуитивный интерфейс сайта.

- Безопасность: защита данных и проверка доступа пользователя.

- Масштабируемость: возможность расширения функционала программного продукта

Планирование процесса анализа и описание процедур оценки характеристик и атрибутов качества:

- Анализ результатов тестирования на каждом этапе разработки.

- Выявление проблем и улучшение качества программы.

- Оценка атрибутов качества в соответствии с жизненным циклом программы.

# РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА

## Разработка подсистемы взаимодействия с новостями клуба

Подсистема должна обеспечивать взаимодействие с новостями клуба такими, как возможность прочитать новость, возможность прочитать комментарии, а так же возможность их написать.

* + 1. Описание подсистемы взаимодействия с новостями клуба

Для возможности прочитать новость из базы данных выводится вся информация новости.

Для возможности прочитать комментарий к новости вызывается функция, которая выводит из базы данных комментарии в соответствии с новостью.

Для возможности оставить комментарий проверяется регистрация пользователя, если пользователь зарегистрирован, то он может заполнить форму и оправить комментарий.

* + 1. Описание алгоритма взаимодействия с новостями клуба.

- Получение данных о новости с базы данных.

- Получение комментариев новости из базы данных.

- Проверка пользователя на регистрацию в системе:

Если пользователь зарегистрирован в системе, то может написать комментарий.

Комментарий отправляется в базу данных.

Схема алгоритма находится в приложении 6.

* + 1. Реализация подсистемы взаимодействия с новостями клуба

Исходный код представлен в приложении 3.

* + 1. Тестирование подсистемы взаимодействия с новостями клуба

Тестирование проводилось путем ручного тестирования по средством открытия новости и написания к ней комментария.

В качестве результата комментарий добавлялся к комментариям новости.

## Разработка подсистемы взаимодействия с отзывами

Подсистема должна обеспечивать взаимодействие с отзывами: удалять, добавлять, выводить из базы данных.

* + 1. Описание подсистемы взаимодействия с отзывами.

Для вывода комментариев для их прочтения, используется функция для вывода из базы данных.

Для добавления комментария используется функция записи в базу данных.

Для удаления комментария используется функция удаления из базы данных.

Для возможности написать или удалить комментарий, идёт проверка на авторизацию пользователя.

* + 1. Описание алгоритма подсистемы взаимодействия с отзывами

- Получение отзывов из базы данных.

- Проверка авторизации пользователя.

Если пользователь авторизирован:

- Если написан отзыв пользователем, то можно его удалить и удалить из базы данных.

- Если отзыв не написан пользователем, то его можно написать и отправить в базу данных.

Схема алгоритма находится в приложении 7.

* + 1. Реализация подсистемы взаимодействия с отзывами

Исходный код представлен в приложении 4.

* + 1. Тестирование подсистемы взаимодействия с отзывами

Тестирование проводилось путем ручного тестирования по средством открытия отзывов и написания к ней своего отзыва, а потом его удаления.

В качестве результата сначала добавляется комментарий, а потом удаляется.

## Разработка подсистемы взаимодействия с расписанием

Подсистема должна обеспечивать взаимодействие с расписанием, а также должна давать возможность подать заявки на первое занятие.

* + 1. Описание подсистемы взаимодействия с расписанием

Для вывод расписания используются функции вывода тренеров и вывода спортивных групп привязанных к тренеру.

Для отправки заявки используется специальная функция.

Для удаления заявки используется функция.

* + 1. Описание алгоритма взаимодействия с расписанием

- Получение расписания из базы данных

- Отправка заявки пользователем

Если пользователь не авторизирован, то его перенесет на авторизацию.

Схема алгоритма находится в приложении 8.

* + 1. Реализация подсистемы взаимодействия с расписанием

Исходный код представлен в приложении 5.

* + 1. Тестирование подсистемы взаимодействия с расписанием

Тестирование проводилось путём ручного тестирования, нужно перейти в расписание и отправить заявку.

В качестве результата заявка добавиться в базу данных и форма измениться на “удалить заявку”.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения работы был проведен анализ аналогичных систем. После анализа систем были выбраны средства разработки. Когда средства разработки были выбраны, началось проектирование модели базы данных. После проектировки модели базы данных и импорта созданного кода sql, началась разработка пользовательского интерфейса. Когда проект был почти готов, начался этап тестирования проекта. После тестирования была проведена работа над ошибками.

В конечном этапе была разработана функциональная и эффективная система, удовлетворяющая требованиям предметной области и обеспечивающая удобство использования для конечных пользователей, с возможностью дальнейшего расширения проекта.

# СПИСОК ЛИТЕРАТУРЫ

1. ГОСТ 34.320-96 Информационные технологии. Система стандартов по базам данных. Концепции и терминология для концептуальной схемы и информационной базы.
2. ГОСТ 34.602-89. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Техническое задание на создание автоматизированной системы.
3. Лазицкас, Е.А. Базы данных и системы управления базами данных: учебное пособие: [12+] / Е.А. Лазицкас, И.Н. Загумённикова, П.Г. Гилевский. – Минск: РИПО
4. Методы и средства проектирования информационных систем и технологий: учебное пособие / авт.-сост. Е.В. Крахоткина; Ставрополь: СКФУ
5. Молдованова, О. В. Информационные системы и базы данных : учебное пособие для СПО / О. В. Молдованова. — 2-е изд. — Саратов
6. ГОСТ 7.32-81 Отчет о научно-исследовательской работе. Общие требования и правила оформления.
7. ГОСТ 19.701-90 ЕСПД. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения.
8. ГОСТ 19.401-78 ЕСПД. Текст программы. Требования к содержанию и оформлению.
9. ГОСТ 19.402-78 ЕСПД. Описание программы.
10. ГОСТ 19.404-79 ЕСПД. Пояснительная записка. Требования к содержанию и оформлению.
11. Соболева В.П. Методические указания по оформлению курсовых и курсовых работ. – Ижевск: Изд-во ИжГТУ, 2008.- 25с.
12. Сайт центра спортивной подготовки Эверест:[электронный ресурс] url: <https://udmeverest.ru> Дата обращения 06.12.2023
13. Сайт Динамо Удмуртия:[электронный ресурс] url: <https://dynamo-udm.ru> Дата обращения 06.12.2023
14. Сайт Боксерского клуба ударник:[электронный ресурс] url: <https://extrimpower.ru> Дата обращения 06.12.2023
15. Сайт организации дополнительного образования «После уроков» :[электронный ресурс] url: <https://posleurokov.ru/online> Дата обращения 06.12.2023

# ПРИЛОЖЕНИЯ

Приложение 1

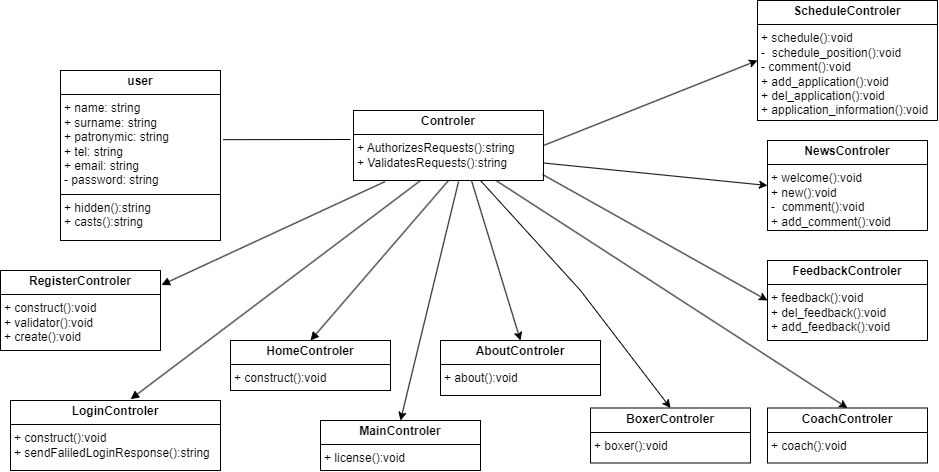


Диаграмма классов

Приложение 2



Структурная схема продукта

Приложение 3

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\DB;

use function Laravel\Prompts\table;

class NewsController extends Controller

{

// страница новости

public function welcome()

{

$news = DB::table('news')->orderBy('news.date', 'DESC')->get();

foreach ($news as $new){

$new->comments = $this->comment($new->id, 3);

}

return view('welcome', ['news' => $news]);

}

// страница отдельной новости

public function new($id){

$news = DB::table('news')->where('id', $id)->get();

foreach ($news as $new){

$new->comments = $this->comment($new->id, null);

}

return view('new', ['news' => $news]);

}

// функция вызова комментариев

private function comment($id, $limit){

$comments = DB::table('comments')->join('users', 'comments.user\_id', '=', 'users.id')->where('comments.news\_id', $id)->orderBy('comments.date', 'DESC')->limit($limit)->get();

return $comments;

}

// функция добавления комментария

public function add\_comment(Request $comment){

DB::table('comments')->insert(['news\_id' => $comment->new\_id, 'user\_id' => auth()->user()->id, 'text' => $comment->comment]);

return redirect()->back();

}

}

Приложение 4

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\DB;

use function Laravel\Prompts\table;

class FeedbackController extends Controller

{

//страница отзывов

public function feedback(){

if(Auth()->User()) {

$id = Auth()->User()->id;

}

else{

$id = 0;

}

$form = DB::table('feedback')->where('feedback.user\_id', '=', $id)->count('feedback.id');

$feedbacks = DB::table('feedback')->join('users', 'users.id', '=', 'feedback.user\_id')->orderBy('feedback.date', 'desc')->get();

return view('feedback', ['form' => $form, 'feedbacks' => $feedbacks]);

}

// функция удаления отзыва

public function del\_feedback(Request $feedback){

$id = Auth()->User()->id;

DB::table('feedback')->where('user\_id', '=', $id)->delete();

return redirect()->back();

}

// функция добавления отзыва

public function add\_feedback(Request $feedback){

$id = Auth()->User()->id;

DB::table('feedback')->insert(['user\_id' => $id, 'text' => $feedback->feedback]);

return redirect()->back();

}

Приложение 5

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\DB;

use function Laravel\Prompts\table;

class ScheduleController extends Controller

{

//страница расписание

public function schedule(){

$coach = DB::table('schedule')->select('schedule.user\_id', 'users.name', 'users.surname', 'users.patronymic')->join('users', 'users.id', '=', 'schedule.user\_id')->groupBy('user\_id')->get();

foreach ($coach as $schedule){

$schedule->position = $this->schedule\_position($schedule->user\_id);

}

if(Auth()->User()){

$id = Auth()->User()->id;

}

else{

$id = 0;

}

$form = DB::table('application')->where('application.user\_id', '=', $id)->count('application.id');

return view('schedule',['schedule' => $coach, 'form' => $form]);

}

//функция возращающая позиции в расписании

private function schedule\_position($id){

$position = DB::table('schedule')->join('weekday', 'weekday.id', '=', 'schedule.weekday\_id')->where('schedule.user\_id', '=', $id)->get();

return $position;

}

//функция отправления заявки

public function add\_application(Request $data){

DB::table('application')->insert(['user\_id' => Auth()->User()->id]);

return redirect(route('application\_information'));

}

//информация о статусе заявки

public function application\_information(){

return view('application\_information');

}

//фунция удаления заявки

public function del\_application(){

DB::table('application')->where('user\_id', '=', Auth()->User()->id)->delete();

return redirect()->back();

}

}

Приложение 6

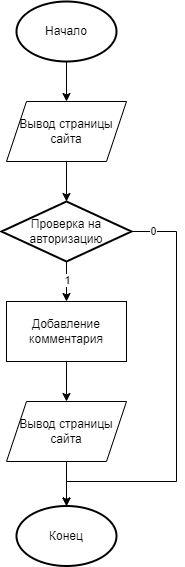


Схема алгоритма написания комментария

Приложение 7

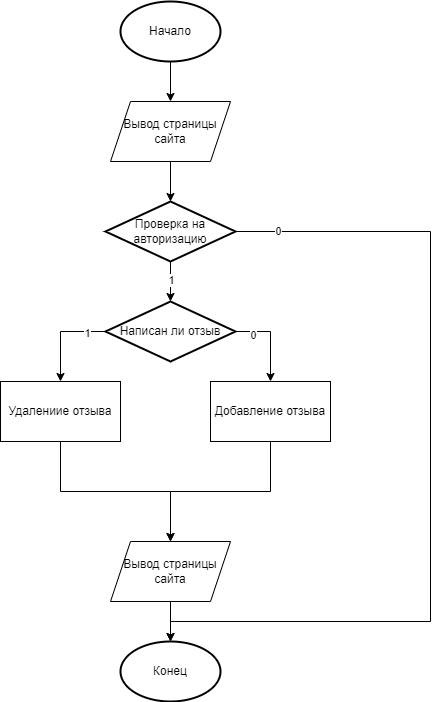


Схема алгоритма изменения отзыва

Приложение 8

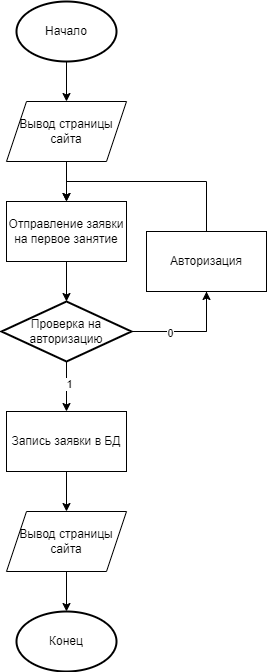


Схема алгоритма оправки заявки

Приложение 9

Руководство системного программиста

1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ
   1. Назначение программы

Программный продукт предназначен для упрощения взаимодействия между боксёрским клубом и спортсменами, которые занимаются в клубе. Так же предназначен для упрощения записи на первое занятие.

* 1. Функции программы

Программный продукт реализует следующий функционал:

- Авторизация пользователя в системе.

- Изменение прав пользователей.

- Редактирование информационного контента.

- Отправление заявки на первое занятие.

- Обработка заявок.

- Изменение расписания.

- Добавление отзывов пользователями.

1.3. Состав технических средств

В состав технических средств входит:

- Сервер.

Минимальные требования к характеристикам сервера:

- Объем дискового хранилища: 20 ГБ.

- ОЗУ 16 ГБ.

- CPU с частой 2,4 ГГц.

1.4. Состав программных средств

- PhpMyAdmin.

- Laravel 10.

- PHP 8.0.

- HTML 5.

- Bootstrap 5.

1.5. Требования к персоналу

Системный программист должен иметь средний уровень квалификации и иметь опыт добавления сайта на хостинг.

1. СТРУКТУРА ПРОГРАММЫ
   1. Сведенья о структуре программы

Программный продукт реализован на основе Framework Laravel 10.

* 1. Сведенья о составных частях программы



Рисунок 1 - Структурная схема программного продукта

1. НАСТРОЙКА ПРОГРАММЫ
   1. Установка сайта на хостинг

Для установки сайта на хостинг и корректной его работы необходимо:

- Перенести архив с сайтом в корневую папку с доменами сервера.

- Распаковать архив.

- Из архива базу данных экспортируем в базу данных сервера.

3.2. Настройка конфигурации Laravel

Для корректной работы программного продукта в файл .env необходимо внести изменения следующих переменных:

* DB\_DATABASE – название используемой базы данных.
* DB\_USERNAME – Логин для авторизации в СУБД.
* DB\_PASSWORD – Пароль для авторизации в СУБД.
  1. Настройка ролей на сайте

Выдача роли администратор происходит только в базе данных. Для этого нужно сделать:

- Создать новую учётную запись при помощи сайта

- Изменить поле role\_id на значение «4» новой учётной запись в таблице users.

1. ПРОВЕРКА ПРОГРАММЫ

Проверка работоспособности программного продукта происходит путем проверки ручного тестирования:

- Проверка главной страницы.

- Проверка авторизации и регистрации.

- Проверка всего функционала клиента.

- Проверка всего функционала модератора.

- Проверка всего функционала администратора.

5. СООБЩЕНИЯ СИСТЕМНОМУ ПРОГРАММИСТУ

Сообщения системному программисту предусмотрены в виде записей логов, расположенных в файле laravel.log расположенным по пути storage/logs/laravel.log.

Приложения 10

Руководство оператора

1. НАЗНАЧЕНИЕ ПРОГРАММЫ

Программный продукт предназначен для упрощения взаимодействия между боксёрским клубом и спортсменами, которые занимаются в клубе. Так же предназначен для упрощения записи на первое занятие.

1. УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ

Для корректного запуска, работы и взаимодействия с программным продуктом требуется устройство, удовлетворяющие следующим действиям:

- Поддержка браузера на основе google chrome, так же такие браузеры такие, как Safari и FireFox.

- Наличие дисплея, с рекомендуемым минимальным требованием 640 x 480 пикселей.

- Наличие средства управления, рекомендуемые средства управления: клавиатура и мышь, дисплей сенсорного устройства.

- Наличие стабильного интеренет соединения с минимальной скоростью 40 мб/с.

3. ВЫПОЛНЕНИЕ ПРОГРАММЫ

3.1. Запуск программного продукта

Для запуска ПП необходимо перейти по адресу http://DOMEN, где DOMEN, домен на котором расположен сайт.

3.2. Идентификация пользователя в системе.

Идентификация пользователя в системе происходит по адресу <http://DOMEN/login> или http://DOMEN/registr, где DOMEN, домен на котором расположен сайт. Если пользователь ещё не зарегистрирован в системе, то нужно заполнить форму по адресу http://DOMEN/registr, иначе заполнить форму и авторизироваться в системе по адресу <http://DOMEN/login>.

3.3. Добавление новости

Добавление новости происходит по адресу http://DOMEN, где DOMEN, домен на котором расположен сайт. Нужно заполнить форму и отправить её.

3.4. Добавления позиции расписания

Добавление позиции расписания происходит по адресу <http://DOMEN/login> или http://DOMEN, где DOMEN домен на котором расположен сайт. Если пользователь идентифицтрован в системе, как модератор или администратор, то может заполнить форму и отправить ее. Если время тренировки уже занято другим тренером, то система сообщит об этом.

3.5. Добавление карточки тренера

Добавление карточки тренера происходит по адресу http://DOMEN/coach, где DOMEN домен на котором расположен сайт. Если пользователь идентифицтрован в системе, как администратор, то может заполнить форму и отправить ее.

3.6. Добавление карточки спортсмена

Добавление карточки тренера происходит по адресу http://DOMEN/boxer, где DOMEN домен на котором расположен сайт. Если пользователь идентифицтрован в системе, как модератор или администратор, то может заполнить форму и отправить ее.

3.7. Изменение роли пользователя

Изменение роли пользователя на moder происходит по адресу http://DOMEN/admin, где DOMEN домен на котором расположен сайт. Нужно выбрать пользователя который получит роль модератора в форме и отправить пользователя.

Изменение роли пользователя на происходит по адресу http://DOMEN/admin, где DOMEN домен на user котором расположен сайт. Нужно выбрать пользователя который получит роль модераторальзователя в форме и отправить пользователя.

3.8. Изменения статуса заявки

Изменение статуса заявки происходит по адресу http://DOMEN/application, где DOMEN домен на user котором расположен сайт. После чего нужно выбрать нужную заявку и нажать кнопку «обработана».

3.9. Изменения отзыва

Добавления отзыва происходит по адресу http://DOMEN/feedback, где DOMEN домен на user котором расположен сайт. Нужно заполнить форму и отправить ее.

Удаление отзыва происходит по адресу http://DOMEN/feedback, где DOMEN домен на user котором расположен сайт. Нужно нажать кнопку «удалить отзыв», после чего форма отправиться и отзыв будет удален.

4. СООБЩЕНИЯ ОПЕРАТОРУ

В ходе использования программного продукта оператор может увидеть сообщения следующие сообщения:

- Данное время уже занято

- Ваша заявка отправлена

- Роль успешно изменена

- Ошибка 404