**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc184464574)

[1 Описание предметной области 5](#_Toc184464575)

[1.1 Введение в предметную область 5](#_Toc184464576)

[1.2 Анализ готовых решений 7](#_Toc184464577)

[2 Проектирование системы 9](#_Toc184464578)

[2.1 Общее назначение системы 9](#_Toc184464579)

[2.2 Требования к функциям выполняемых системой 9](#_Toc184464580)

[2.3 План тестирования 13](#_Toc184464581)

[3 Реализация проекта системы 15](#_Toc184464582)

[3.1 Описание среды разработки 15](#_Toc184464583)

[3.1.1 Выбор и описание программных инструментов 16](#_Toc184464584)

[3.1.2 Обоснование выбора инструментария по разработке 17](#_Toc184464585)

[3.2 Разработка программного модуля 18](#_Toc184464586)

[3.2.1 Реализация пользовательского интерфейса программы 18](#_Toc184464587)

[3.2.2 Описание кодом функциональных узлов модуля 19](#_Toc184464588)

[3.2.3 Результат работы и тестирования 21](#_Toc184464589)

[ЗАКЛЮЧЕНИЕ 23](#_Toc184464590)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 24](#_Toc184464591)

[ПРИЛОЖЕНИЕ 1: ДИАГРАММЫ ПРОЕКТИРОВАНИЯ БАЗЫ ДАННЫХ 26](#_Toc184464592)

[ПРИЛОЖЕНИЕ 2: ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ 27](#_Toc184464593)

[ПРИЛОЖЕНИЕ 3: ПЛАН ТЕСТИРОВАНИЯ 28](#_Toc184464594)

[ПРИЛОЖЕНИЕ 4: ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС 36](#_Toc184464595)

[ПРИЛОЖЕНИЕ 5: ФУНКЦИИ КНОПОК 48](#_Toc184464596)

[ПРИЛОЖЕНИЕ 6: ВЫПОЛНЕНИЕ ТЕСТИРОВАНИЯ 57](#_Toc184464597)

[ПРИЛОЖЕНИЕ 7: СТРУКТУРА ПРОЕКТА 61](#_Toc184464598)

ВВЕДЕНИЕ

Современные информационные технологии играют ключевую роль в автоматизации управления [8] различных процессов, также, как и в спортивной индустрии. Современная тенденция к здоровому образу жизни и повышению физической активности требует создания удобных инструментов для управления тренировочным процессом. Большое количество данных, связанных с тренировками, включая планирование, учет результатов и анализ прогресса, делает процесс управления сложным и трудоемким. Это особенно актуально для спортсменов-любителей, которые не всегда имеют доступ к профессиональным тренерам или специализированным системам учета. Система учета тренировок будет позволять упорядочивать планирование, контроль и анализ тренировок, обеспечивая улучшение результатов и оптимизацию тренировок.

Целью данного курсового проекта является разработка программного модуля «Система учета спортивных тренировок». Система будет автоматизировать процесс планирования, проведения и анализа тренировок. Помимо этого, использование такой системы улучшит эффективность процесса тренировок за счет возможности быстрого доступа к данным и анализа прогресса спортсменов.

В рамках данного курсового проекта будут решены следующие задачи:

* Проанализировать предметную область
* Проанализировать готовые решения
* Подготовить техническое задание
* Подготовить план тестирования
* Обосновать выбор инструментов и средств разработки
* Описать реализацию технического задания, выполнить тестирование

Объектом исследования в данной работе является процесс автоматизации учета спортивных тренировок.

Предметом исследования выступают методы и технологии, используемые для учета, хранения и обработки данных о тренировках.

Актуальность данного курсового проекта обусловлена потребностью в повышении эффективности управления тренировками в основном, на индивидуальном уровне. С улучшением технологий и увеличением количества данных, которые необходимо учитывать для повышения качества тренировок, автоматизация становится необходимым инструментом для качественного и быстрого контроля тренировок. Использование подобной системы позволит людям сокращать время на подсчете и повысить точность учета данных.

Кроме того, разработка подобных систем способствует популяризации здорового образа жизни и увеличению вовлеченности населения в спортивные активности. Удобство автоматизации позволяет сократить рутинные процессы и сосредоточиться на достижении спортивных целей.

Таким образом, создание программного модуля системы учета спортивных тренировок представляет собой ключевой шаг в направлении автоматизации процесса учета тренировочного процесса. Ожидается, что внедрение этого модуля усовершенствует работу спортсменов-любителей, что, в свою очередь, приведет к улучшению их физических достижений.

1. Описание предметной области
   1. Введение в предметную область

Система учета спортивных тренировок представляет собой значимый механизм в спортивной индустрии. На протяжении последних десятилетий рос интерес населения к здоровому образу жизни, к примеру с 2021 года он ежегодно растет в среднем на 20%, а эксперты в области ЗОЖ становятся все более востребованными [9], но этот процесс включал объемную работу с данными. Это приводило к ошибкам и брало внушительное количество времени. Впрочем, с формированием информационных технологий появилась потребность в автоматизации предоставленного процесса. Введение электронных систем учета позволяет значительно сжать время на управление тренировками и увеличить достоверность ведения данных.

Информатизация спортивной сферы обладает рядом неоспоримых преимуществ. Во-первых, автоматизация учетных процессов сокращает число человеческих ошибок, что критично для хранения информации спортсменов-любителей и их результатов. Во-вторых, система позволяет более эффективно исследовать тренировочные процессы, включая создание отчетов, графиков прогресса, анализа нагрузок и периодов отдыха. Это способствует наилучшему контролю за состоянием спортсменов и увеличению их физических достижений. Внедрение подобных систем вдобавок упрощает взаимодействие между инструктором и любителями за счет создания практичных интерфейсов для наблюдения динамики тренировок.

Внедрение различных технологических решений в процесс спортивных тренировок предоставляет экспертам возможность производить комплексный и прозрачный сбор данных, а впоследствии анализировать их автоматически [5], что стало возможно вследствие формирования таких инструментов, как базы данных, клиент-серверные приложения и пользовательские интерфейсы. Эти технологии позволяют сохранять огромные объемы данных, проводить трудоемкие вычисления и предоставлять пользователям комфортные методы доступа к информации. Например, базы данных позволяют хранить всю вводимую пользователями информацию и предоставлять быстрый доступ к ней [10], что помогает тренерам более эффективно исследовать тренировки за продолжительные периоды времени.

Современные электронные системы учета обязаны соответствовать ряду требований, таких как доступность, надежность, защищенность данных, масштабируемость и практичность использования. исключительно важно гарантировать защиту данных спортсменов, так как они могут включать в себя персональные данные, которые должны быть надежно защищены в соответствии с законодательством. В современных условиях требования к безопасности таких систем ужесточаются, так как утечки данных могут нанести вред как спортсменам, так и спортивным организациям.

Что касается законодательных аспектов, то при разработке и внедрении системы учета спортивных тренировок нужно учитывать запросы к защите персональных данных, такие как Федеральный закон №152-ФЗ "О персональных данных". Этот закон регулирует сбор, хранение и обработку персональных данных, и, согласно статье 5, Обработка персональных данных должна осуществляться на законной и справедливой основе» и она «должна ограничиваться достижением конкретных, заранее определенных и законных целей [1]. Это особенно важно в контексте системы учета тренировок, где хранятся данные о здоровье и физическом состоянии спортсменов. Также необходимо учитывать Федеральный закон №149-ФЗ "Об информации, информационных технологиях и о защите информации", который в статье 16 обязывает операторов информационных систем обязует обеспечить предотвращение несанкционированного доступа к информации и (или) передачи ее лицам, не имеющим права на доступ к информации и постоянный контроль за обеспечением уровня защищенности информации [2]. В соответствии с Федеральным законом №187-ФЗ "О безопасности критической информационной инфраструктуры Российской Федерации", статья 5 обязывает операторов обеспечивать безопасность таких объектов, а также разрабатывать и реализовывать меры по предупреждению, обнаружению и ликвидации последствий компьютерных атак. [4]. В части разработки программного обеспечения важно следовать стандарту ГОСТ ИСО 12207, который определяет процессы жизненного цикла программных продуктов, включая процессы приобретения, поставки, разработки, эксплуатации и сопровождения [3].

Для разработки программного модуля системы учета спортивных тренировок будет использована каскадная методика, также известная как водопадная модель. Она включает последовательные этапы разработки: от анализа требований и проектирования до программирования, тестирования и внедрения. Преимущество каскадного подхода заключается в том, что весь процесс работы над программой идёт строго поэтапно, и нельзя перейти от одного этапа к другому, пока текущий полностью не закончен [11]. Тем не менее, каскадный подход обладает своими недостатками, такие как трудности во внесении изменений на поздних стадиях разработки, отчего важно точно проработать техническое задание на ранних этапах проекта.

* 1. Анализ готовых решений

В настоящее время существует несколько информационных систем, которые уже активно используются для учета и анализа тренировочного процесса. Эти системы помогают тренерам и спортсменам структурировать данные о тренировках, отслеживать прогресс и улучшать результаты. Рассмотрим некоторые из наиболее популярных решений.

Одной из наиболее известных систем является TrainHeroic. Это платформа, которая позволяет тренерам создавать и управлять программами тренировок, а также отслеживать результаты своих спортсменов в режиме реального времени. Однако TrainHeroic ориентирована в основном на профессиональный спорт и может быть сложной для использования начинающими тренерами и спортсменами.

Ещё одной популярной системой является iGym. У неё большой функционал: база упражнений, возможность сделать свою программу, две стандартные программы, тренировка с таймером отдыха, графики прогресса, календарь тренировок, можно создавать суперсеты из упражнений. Интерфейс не интуитивно понятный. IGym очень хорошая программа, она хорошо подходит профессионалам в спортивном зале.

Старая программа «Тренировки» является своеобразным безбумажным дневником тренировок. Она автоматизирует сбор и подсчет информации о различных тренировках и их параметров. Есть возможность учета тренировок в циклических видах спорта: спортивное ориентирование, бег, лыжи, плавание и так далее. Можно проводить анализ по собственным настройкам.

Также есть My PT (Personal Training) Hub, еще одну популярную платформу для управления тренировочными программами. Программа имеет гибкие возможности для персонализации и интеграции с другими приложениями, что делает её удобной для использования как тренерами, так и спортсменами.

Таким образом, на рынке уже существуют различные решения для учета спортивных тренировок, но каждое из них имеет свои особенности и ограничения. Основными недостатками существующих систем являются высокая стоимость, сложность использования для новичков и отсутствие гибкости в настройках. Это делает разработку нового программного модуля актуальной, так как он может удовлетворить потребности небольших спортивных организаций, спортсменов, которые ищут более доступные и удобные решения для управления тренировками.

1. Проектирование системы
   1. Общее назначение системы

Основным назначением разрабатываемого программного модуля является автоматизация учёта спортивных тренировок для спортсменов-любителей. Внедрение данной системы предполагает улучшение организации тренировочного процесса и повышение удобства взаимодействия между инструктором и спортсменом. Экономический эффект от внедрения данного программного обеспечения заключается в оптимизации процесса управления тренировками и снижении затрат, связанных с ручным учётом и ведением отчётности.

Спортсменам-любителям, часто не хватает систематизации и контроля над тренировками, что может приводить к неправильному распределению нагрузок и снижению мотивации. Программный модуль позволит структурировать информацию о тренировках, делать её более наглядной и доступной, что повысит эффективность тренировочного процесса и снизит вероятность перетренированности или травм. Для спортивных клубов это может означать увеличение удержания клиентов и снижение затрат на организацию тренировочного процесса.

* 1. Требования к функциям выполняемых системой

Исходя из анализа предметной области, функциональные задачи программы включают в себя: улучшение взаимодействия с клиентом через интуитивно понятный пользовательский интерфейс, ведение учетных данных о тренировках: даты, типы тренировок, продолжительность, возможность анализа прогресса пользователя с отображением динамики на графиках, возможность хранения и экспорта данных в формате Excel или других удобных форматах.

Для создания программы важно тщательно выбрать подходящую среду разработки. В качестве основной платформы был выбран Visual Studio Code (VSCode), из-за своей популярности, простоты использования, широких возможностей для работы с Python и поддержки интеграции сторонних инструментов.

Диаграмма классов UML, представленная на рисунке 1 – «Диаграмма вариантов использования» описывает, какой функционал разрабатываемой программной системы доступен каждой группе пользователей [12].

Изображение выглядит как текст, диаграмма, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 1 - Диаграмма вариантов использования

Далее приведено краткое описание каждого элемента диаграммы классов и их возможности: Спортсмен-любитель: Просмотр прошедших тренировок, что включает в себя экспорт данных; Авторизация; Регистрация; Добавление новых тренировок. Администратор базы данных: Управление данными пользователей, что включает в себя удаление пользователей

Разработка пользовательского интерфейса основывается на принципах простоты и доступности. Интерфейс должен быть интуитивно понятным для спортсменов-любителей, с минимальным количеством сложных элементов и подробной информацией о тренировках. Всего будет 6 окон: окно авторизации, где можно зайти в аккаунт, в том числе и админу. Окно регистрации. Здесь создать аккаунт может только обычный пользователь Главное окно выбора действия, или же приветствие пользователя. Включает в себя отображение даты последней тренировки, если она есть, и 3 кнопки выбора взаимодействия, которые открывают соответствующие окна. Окно просмотра тренировок, вроде последних тренировок, их вид и комментариев к ним, а также кнопки для взаимодействия: добавить тренировку и просмотреть анализ тренировок. Окно прогресса с графиком тренировок и возможностью экспорта. Окно добавления тренировки: возможность выбрать дату, написать или выбрать тип тренировки, её продолжительность и добавить комментарий к ней.

Основные объекты системы представлены в Приложении 2, Таблице 1 «Основные объекты».

После определения объектов предметной области был проведен отбор атрибутов для каждого объекта. Атрибуты сущности «Пользователь» представлены в Приложении 2, Таблице 2 «Атрибуты пользователя»

Атрибуты сущности «Тренировка» представлены в Приложении 2, Таблице 3 «Атрибуты тренировки».

Атрибуты сущности «Администратор базы данных» представлены в Таблице 4 – «Атрибуты администратора ниже»:

Таблица 4 - Атрибуты администратора

|  |  |  |
| --- | --- | --- |
| Атрибут | Тип данных | Описание |
| Id\_admin | Int Primary key | Уникальный идентификатор |
| Name | Text (30) | Имя администратора |
| last\_name | Text (30) | Фамилия администратора |

ER-диаграмма для программного модуля системы «Система учета спортивных тренировок» включает в себя сущности, их атрибуты и связи между ними. Она служит важным инструментом для проектирования и понимания логики хранения данных, необходимых для эффективного управления бизнес-процессами. Всё это представлено в Приложении 1, на рисунке 2 «Диаграмма классов».

На данной диаграмме отображены функции сущностей. Администратор: может управлять данными пользователей. Спортсмен-любитель: может изменять или добавлять тренировки, может производить экспорт данных о тренировках.

Также на диаграмме показаны следующие связи: Связь один ко многим между администратором и спортсменами любителями: один админ может управлять данными многих пользователей. Связь один ко многим между Спортсменом-любителем и тренировками: Один спортсмен может изменять или добавлять много тренировок

Это описание ER-диаграммы отражает структуру для программного модуля системы «Система учета спортивных тренировок», включая сущности, их атрибуты и связи между ними. Данная ER-диаграмма дала чёткое представление об атрибутах и их взаимосвязях, нужных для улучшения управления операционными процессами.

Диаграмма базы данных в нотации IDEF1X для системы «Система учета спортивных тренировок» показывает основные сущности, атрибуты, типы данных и связи между ними, необходимые для учета данных о тренировках спортсменов-любителей. Она помогает структурировать информацию и определить логические взаимосвязи между объектами системы. Всё это представлено в Приложении 1, на рисунке 3 «Диаграмма базы данных».

Основные сущности и описание их атрибутов. Основные атрибуты сущности amateur\_athlete (спортсмен-любитель): id\_sportsman — первичный ключ (PK), уникальный идентификатор спортсмена. name и last\_name — строковые атрибуты, представляющие имя и фамилию спортсмена. sex — булевый атрибут, определяющий пол спортсмена. id\_login — внешний ключ (FK), ссылающийся на идентификатор в таблице login\_password, что позволяет спортсмену иметь учетные данные для авторизации. Связь: amateur\_athlete находится в отношении один ко многим с таблицей training, что позволяет одному спортсмену иметь несколько тренировок.

Основные атрибуты сущности training: id\_training — первичный ключ, уникальный идентификатор тренировки. type — строковый атрибут, указывающий тип тренировки. date — атрибут типа datetime, определяющий дату. comment — логический атрибут, позволяющий добавить заметки к тренировке. duration — атрибут типа time, обозначающий продолжительность тренировки. id\_sportsman — внешний ключ, связанный с amateur\_athlete, указывающий, какому спортсмену принадлежит данная тренировка. Связь сущности: каждый тренировочный сеанс относится к одному спортсмену, но один спортсмен может иметь несколько тренировок.

Основные атрибуты сущности login\_password (учетные данные): id\_login — первичный ключ, уникальный идентификатор учетной записи. login и password — строковые атрибуты, содержащие логин и пароль пользователя. Id\_role – внешний ключ, связанный с role, указывает чьи это данные, админа или пользователя. Связь сущности: login\_password находится в отношении 1:1 с amateur\_athlete и admin, что обеспечивает авторизацию как для спортсменов-любителей, так и для администраторов системы.

Основные атрибуты сущности admin (администратор): id\_admin — первичный ключ, уникальный идентификатор администратора. name и last\_name — строковые атрибуты, представляющие имя и фамилию администратора. id\_login — внешний ключ, связанный с login\_password, для доступа к системе. Связь сущности: admin также использует учетные данные в таблице login\_password для авторизации, что позволяет администратору управлять системой и контролировать данные спортсменов.

Основные атрибуты сущности role (Роли): Id\_role - первичный ключ, уникальный идентификатор роли. Role – текстовое поле с название роли – «admin» или «user» Связь сущности: role находится в связи один ко многим с login\_password, что также обеспечивает верную авторизацию для админов и для пользователей.

Эта модель базы данных, выполненная в нотации IDEF1X, подчеркивает четкие отношения между таблицами и их атрибутами, обеспечивая структурированное и понятное представление системы хранения данных для учета спортивных тренировок.

* 1. План тестирования

Целью тестирования является проверка соответствия модуля системы учета спортивных тренировок его качества и соответствия заранее определенным функциональным и нефункциональным требованиям. Это необходимо для того, чтобы убедиться, что система работает стабильно и выполняет все заявленные функции, а также соответствует требованиям удобства и безопасности.

Функциональное тестирование направлено на выявление ошибок и проверку правильности выполнения всех задач, которые система должна выполнять. Важно убедиться, что все функции работают в соответствии с техническим заданием, а система корректно обрабатывает данные. В рамках функционального тестирования будут проверяться следующие аспекты включают: Регистрация и авторизация пользователей, ведение учета тренировок: создание, просмотр записей о тренировках, просмотр и анализ истории тренировок с возможностью экспорта данных, вычисление процента Жировой массы тела.

Отдельным блоком в рамках тестирования системы проводится обеспечение безопасности. Для этого будут выполнены следующие задачи: Проверка аутентификации и авторизации пользователей: правильность ввода данных, защита от несанкционированного доступа.

Объекты тестирования включают: Регистрация/авторизация пользователя, операции с тренировками, просмотр данных о тренировках, функции экспорта данных, интерфейс и взаимодействие с пользователем. Описание каждого объекта тестирования приведено в Приложении 3, Таблице 5 «План тестирования».

Таким образом, тестирование системы учета спортивных тренировок направлено на обеспечение соответствия функциональным требованиям, стабильности и удобства работы системы, а также безопасности данных пользователей. Это позволит убедиться в том, что система является надежным инструментом для спортсменов-любителей и полностью соответствует заявленным ожиданиям.

1. Реализация проекта системы
   1. Описание среды разработки

Для разработки программного модуля системы учёта спортивных тренировок можно выбрать различные среды разработки и инструменты, в зависимости от потребностей проекта. Существует ряд вариантов, которые можно рассмотреть.

Языки программирования: среди них выделяется Python благодаря его простоте и универсальности. Python подходит как для начинающих, так и для опытных разработчиков. Альтернативы включают JavaScript, используемый для веб-разработки, Java, популярный в корпоративной среде, и C++, который часто применяется для высокопроизводительных приложений

Среды разработки: PyCharm - Отличается широким набором функций для Python-разработки, включая поддержку тестирования, отладки. VS Code — бесплатный и очень популярный редактор кода от Microsoft, который поддерживает большое количество языков программирования. В VS Code встроены много возможностей, которые интересны опытным разработчикам [12].

Системы управления базами данных: PostgreSQL - имеет большую функциональность и высокую производительность. SQLite - очень компактная СУБД, которая не использует серверы и другие утилиты. Все данные хранятся на одном устройстве.

Библиотека для графического интерфейса: PyQt5 — это библиотека для разработки графического интерфейса на языке Python, которая обеспечивает возможность создания удобных интерфейсов и поддерживает множество современных элементов управления. Tkinter — это кроссплатформенный графический интерфейс Python, позволяющий создавать различные приложения.

При разработке модуля программист должен обладать знаниями в нескольких областях. Во-первых, это базовые концепции программирования, такие как работа с переменными, условными операторами и циклами, а также знакомство с такими структурами данных, как списки, множества и словари. Важным навыком является умение использовать различные инструменты разработки, включая итераторы и обработку ошибок. Также программист должен уметь работать с внешними библиотеками и API, а для более сложных задач - с объектно-ориентированным программированием

В зависимости от конкретных потребностей и сложности проекта, будет выбрана одна или несколько технологий и инструментов для работы. Эти средства обеспечивают гибкость, удобство и позволяют ускорить процесс разработки, тестирования и внедрения программного модуля.

* + 1. Выбор и описание программных инструментов

Для разработки данного программного обеспечения были выбраны основные инструменты, предоставленные далее.

PyQt5 — это библиотека для разработки графического интерфейса на языке Python, которая обеспечивает возможность создания удобных интерфейсов и поддерживает множество современных элементов управления. PyQt5 предоставляет гибкие и мощные инструменты для построения сложных интерфейсов.

SQLite - это программное обеспечение, которое позволяет пользователям взаимодействовать с реляционной базой данных. В SQLite база данных хранится в одном файле — черта, которая отличает ее от других движков баз данных [6]. SQLite является отличным решением для проектов небольшого и среднего масштаба, которые не требуют работы с внешними базами данных.

Python — высокоуровневый язык программирования, который используется в разных областях. Он не только позволяет создавать веб и мобильные приложения, но и разрабатывать программное обеспечение для ПК [14]. Python имеет мощную экосистему библиотек и модулей, что позволяет интегрировать его с различными инструментами для разработки приложений.

Кроме перечисленных инструментов, для реализации проекта использовались дополнительные библиотеки Python: «pandas» — используется для экспорта данных в формате Excel, что удобно для аналитики и сохранения данных. sys — помогает управлять запуском приложения и работать с системными аргументами. «matplotlib» и «openpyxl» — используются для создания и отображения графиков, что позволяет наглядно представлять данные. «re» — применяется для установки ограничений на ввод данных.

* + 1. Обоснование выбора инструментария по разработке

Выбор вышеописанных инструментов обусловлен их многочисленными преимуществами и соответствием поставленным задачам проекта. Среда Visual Studio Code предоставляет необходимые возможности для отладки и тестирования кода, что особенно важно при разработке сложных модулей. PyQt5 кроссплатформенный фреймворк, используемый в разработке графических интерфейсов пользователя. Используя его, разработчик может создавать приложения намного быстрее, не жертвуя при этом значительной частью производительности [15].

База данных SQLite выбрана из-за её компактности и встроенности, что позволяет хранить данные локально, без необходимости развертывания серверных решений. Это также упрощает процесс развертывания системы на различных устройствах.

Использование языка Python обусловлено его популярностью и гибкостью в программировании. Так как, Python — это мощный и универсальный язык программирования, он используется в самых различных областях, таких как веб-разработка, наука, искусственный интеллект и автоматизация, что делает его идеальным выбором для множества проектов [16].

Таким образом, выбор инструментов для разработки был обоснован с учетом требований к производительности, кроссплатформенности и лёгкости интеграции, что способствует созданию устойчивой и функциональной системы учёта спортивных тренировок.

Дополнительные библиотеки Python выбраны из-за удобности в использовании, простоты и популярности, благодаря чему есть много руководств к ним.

* 1. Разработка программного модуля

В соответствии с техническим заданием, разрабатываемая система предназначена для учета спортивных тренировок для любителей. Модуль будет включать несколько ключевых элементов, таких как учет и создание пользователей, управление тренировками и их просмотр, экспорт в виде таблицы Excel, просмотр прогресса пользователя в виде линейного графика и его экспорт, а также администрирование системы, включающее в себя просмотр, удаление пользователей, и их тренировок. Основной упор будет сделан на удобство и простоту использования интерфейса, с интеграцией базы данных, состоящей из связанных таблиц, для хранения информации о пользователях и тренировках.

* + 1. Реализация пользовательского интерфейса программы

Для реализации интерфейса будет использована библиотека PyQt, которая позволяет создать интуитивно понятные и функциональные окна для взаимодействия с пользователем. В рамках проекта предусмотрены следующие 7 основных окон и 4 уведомления для обычных пользователей:

1. Окно авторизации предоставлено ниже на рисунке 16 – «Окно авторизации»

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 16 - Окно авторизации

Заголовок у окна - «Авторизация». Доступны 2 поля для ввода – Имя пользователя и Пароль. Также можно задействовать 2 кнопки: «Войти» – срабатывает только если пароль и логин верны и «Нет аккаунта» – открывает окно регистрации.

1. Окно ошибки авторизации предоставлено ниже на рисунке 17 – «Окно ошибка авторизации».

Изображение выглядит как текст, снимок экрана, мультимедиа

Автоматически созданное описание

Рисунок 17 - Окно ошибка авторизации

Название у данного окна – «Ошибка авторизации». Оно возникает при введении неправильный данных при входе в аккаунт. Выбрать можно только одну кнопку – «ОК», которая закроет сообщение

Все остальные окна, и функции к кнопкам пользователя и админа представлены в Приложении 4 – Пользовательский интерфейс.

* + 1. Описание кодом функциональных узлов модуля

Для реализации функциональности системы, основанной на диаграмме прецедентов, можно выделить следующие основные узлы:

Авторизация пользователя. Пример кода для авторизации предоставлен на рисунке 37 – «Функция авторизации».

Изображение выглядит как текст, снимок экрана, документ, Шрифт

Автоматически созданное описание

Рисунок 37 - Функция авторизации

Эту функцию активирует кнопка «Войти» в окне авторизации. Кнопка активирует функцию «check\_password», которая активирует функцию «get\_user\_role». Вторая функция обращается к базе данных в поиске пользователя с заданной информацией. Потом первая функция проверяет, вернула ли что-то предыдущая, и если ответ – да, то активирует функцию «open\_window», которая сравнивает роль и открывает соответствующее окно, а если нет – показывает ошибку.

Функция кнопки «Экспорт данных». Пример кода предоставлен ниже на Рисунке 38 – «Экспорт данных»:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 38 - Экспорт данных

Данная функция выполняет запрос к базе данных, извлекая информация и записывая в файл Excel.

Все остальные функциональные узлы модуля представлены в Приложении 5 – Функции кнопок.

* + 1. Результат работы и тестирования

В ходе работы разработан программный модуль и выполнено его тестирование в соответствии с тестовым планом. Тестирование включало проверку ключевых функциональности системы. Ниже приведены тест-кейсы с описанием результатов:

Кейс 1: Создание пользователя. Тестируемая функциональность: Регистрация нового пользователя. Тестовый набор: Ввод данных в поле "Фамилия Имя".

Ввод данных кириллицей - ожидаемый результат: переход на следующее поле. Ввод данных латиницей - ожидаемый результат: ошибка «Имя должно содержать только русские буквы». Результат представлен ниже на рисунке 55 – «Ошибка в имени».

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 55 - Ошибка в имени

Ввод корректных данных – ожидаемый результат: «Новый пользователя успешно зарегистрирован». Результат представлен ниже на рисунке 56 – «Корректные данные для нового пользователя».

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 56 - Корректные данные для нового пользователя

Остальные тест-кейсы представлены в Приложении 6 - Выполнение тестирования, а также структура проекта находится в Приложении 7 – Структура проекта.

Таким образом, тестирование подтвердило соответствие системы функциональным требованиям. Все выявленные недочеты были исправлены. Подробный листинг кода всех модулей системы представлен на сайте «github»[7].

ЗАКЛЮЧЕНИЕ

В ходе выполнения курсового проекта была достигнута поставленная цель – разработан программный модуль «Система учета спортивных тренировок», автоматизирующий процессы планирования, проведения и анализа тренировок. Система, предназначенная для спортсменов-любителей, помогает упрощать управление тренировочным процессом и способствует улучшению их спортивных результатов. Для достижения цели курсового проекта были последовательно выполнены следующие задачи:

* Проанализирована предметная область, выявлены требования к системе учета тренировок и рассмотрены существующие готовые решения.
* Составлено техническое задание, включающее описание функциональных и нефункциональных требований к системе.
* Разработан и реализован план тестирования для проверки работоспособности программного модуля.
* Обоснован выбор инструментов разработки, включая современные программные технологии и средства автоматизации.
* Выполнено тестирование программного модуля, подтверждающее его соответствие требованиям и заявленной функциональности.

Таким образом, разработанный программный модуль успешно выполняет свои задачи, упрощает процесс управления тренировками и является надежным инструментом для спортсменов-любителей. Внедрение системы может стать важным шагом в повышении эффективности индивидуальных тренировок, а также в улучшении организации тренировочного процесса в небольших спортивных клубах.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

Стандарты и законодательные материалы

1. Федеральный закон "О персональных данных" от 27.07.2006 N 152-ФЗ
2. Федеральный закон "Об информации, информационных технологиях и о защите информации" от 27.07.2006 N 149-ФЗ
3. ГОСТ Р ИСО/МЭК12207 – 2010 «Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств»
4. Федеральный закон "О безопасности критической информационной инфраструктуры Российской Федерации" от 26.07.2017 N 187-ФЗ

Монографии

1. Белякова М.Ю., Дьяконов А.Д. Применение цифровых и информационных технологий в сфере физической культуры и спорта // Экономика и управление в спорте. – 2021. – Том 1. – № 3. – С. 133-148. – doi: [10.18334/sport.1.3.119785](https://1economic.ru/lib/119785)
2. Светлана Солтык. 2023. SQLite – КОМПАКТНАЯ ВСТРАИВАЕМАЯ СИСТЕМА УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ. - Департамент Программной Инженерии и Автоматики, TI-196, Факультет Вычислительной Техники, Информатики и Микроэлектроники, Технический Университет Молдовы, Кишинев, Республика Молдова
3. Романов А.А. 2024. Курсовой проект Разработка программного модуля «Система учета тренировок» <https://github.com/Floom1/Kursovoy_project_Razrabotka_programmnogo_modulya_Sistema_ucheta_trenirovok>

Интернет-источники

1. Информационные технологии автоматизации управления бизнес процессами - <https://apptask.ru/blog/informacionnye-texnologii-avtomatizacii-upravleniia-biznes-processami>
2. Исследование показало ежегодный рост интереса россиян к теме ЗОЖ - <https://iz.ru/1637881/2024-01-23/issledovanie-pokazalo-ezhegodnyi-rost-interesa-rossiian-k-teme-zozh>
3. Что такое база данных - <https://gb.ru/blog/chto-takoe-baza-dannyh/>
4. Что такое каскадная разработка - <https://thecode.media/waterfall/>
5. Сырцев Вадим Использование диаграммы вариантов использования - <https://habr.com/ru/articles/566218/>
6. Евгений Шкляр Visual Studio Code // <https://htmlacademy.ru/blog/soft/vscode>
7. Python: что это за язык программирования, его преимущества и где он используется? - <https://vc.ru/dev/835543-python-chto-eto-za-yazyk-programmirovaniya-ego-preimushestva-i-gde-on-ispolzuetsya#anchor1>
8. PyQt - <https://blog.skillfactory.ru/glossary/pyqt/>
9. Язык программирования Python: особенности и перспективы - <https://blog.geekbrains.by/jazyk-programmirovanija-python-osobennosti-i-perspektivy/#4>

ПРИЛОЖЕНИЕ 1: ДИАГРАММЫ ПРОЕКТИРОВАНИЯ БАЗЫ ДАННЫХ

Изображение выглядит как текст, диаграмма, снимок экрана, зарисовка

Автоматически созданное описание

Рисунок 2 - Диаграмма классов

Изображение выглядит как текст, диаграмма, План, Технический чертеж

Автоматически созданное описание

Рисунок 3 - Диаграмма базы данных

ПРИЛОЖЕНИЕ 2: ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ

Таблица 1 – Основные объекты

|  |  |  |
| --- | --- | --- |
| № | Название объекта | Краткое описание |
| 1 | Пользователь | Спортсмен-любитель, который регистрируется в системе |
| 2 | Тренировка | Информация о конкретной тренировке |
| 3 | Администратор | Управляет данными о пользователях |

Таблица 2 – Атрибуты пользователя

|  |  |  |
| --- | --- | --- |
| Атрибут | Тип данных | Описание |
| id\_sportsman | Int Primary key | Уникальный идентификатор |
| Name | Text | Имя пользователя |
| last\_name | text | Фамилия пользователя |
| Sex | Boolean | Пол пользователя |

Таблица 3 - Атрибуты тренировки

|  |  |  |
| --- | --- | --- |
| Атрибут | Тип данных | Описание |
| Id\_train | Int Primary key | Уникальный идентификатор |
| Date\_training | Date | Дата проведения тренировки |
| Type | Text (100) | Тип тренировки |
| Duration | Time | Длительность тренировки |
| Comment | Text (255) | Комментарий к тренировке от пользователя |

ПРИЛОЖЕНИЕ 3: ПЛАН ТЕСТИРОВАНИЯ

Таблица 5 - План тестирования

| № | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат | Результат тестирования |
| --- | --- | --- | --- | --- | --- |
| 1 | Создание пользователя | Фамилия Имя пользователя | Фамилия Романов, Имя Александр | Переход на следующее поле Пол | Переход на следующее поле Пол |
|  |  |  | Фамилия Romanov Имя Aleksandr | Ошибка Имя должно содержать только русские буквы | Ошибка Имя должно содержать только русские буквы |
|  |  |  | Пустое поле | Ошибка. Поле не должно быть пустым | Ошибка. Поле не должно быть пустым |
|  | Рисунок 4 - Ошибка регистрации | | | | |
|  |  | Поле Пол | Выбор мужчина или женщина | Переход в следующее окно | Переход в следующее окно |
|  |  | Поле Логин | Логин Alex0 | Переход следующее поле Пароль | Переход следующее поле Пароль |
|  |  |  | Логин Алекс0 | Переход следующее поле Пароль | Переход следующее поле Пароль |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Продолжение таблицы 5 | | | | | |
| № | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат | Результат тестирования |
|  |  |  | Пустой логин | Ошибка. Поле не должно быть пустым | Ошибка. Поле не должно быть пустым |
| Рисунок 5 - Ошибка авторизации | | | | | |
|  |  | Поле Пароль | Логин Alex0 | Пользователь зарегистрирован | Пользователь зарегистрирован |
|  |  |  | Пароль Романов1 | Ошибка: поле не должно содержать русские буквы | Ошибка: поле не должно содержать русские буквы. |
| Рисунок 6 - Ошибка пароля | | | | | |
|  |  |  | Пустое поле Пароль | Ошибка. Поле не должно быть пустым | Ошибка. Поле не должно быть пустым. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Продолжение таблицы 5 | | | | | | | | |
| № | | Наименование функциональности | Наименование поля | | Тестовый набор | Ожидаемый результат | | Результат тестирования |
| Рисунок 7 - Ошибка длины пароля | | | | | | | | |
| Рисунок 8 - Ввод данных в регистрации | | | | | | | | |
| 2 | Авторизация пользователя | | Поле Имя пользователя | Ввод неверного логина Alex2 | | Ошибка неверное имя пользователя | Ошибка неверное имя пользователя | |
|  |  | |  | Верный логин Alex1 | | Переход к полю Пароль | Переход к полю Пароль | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Продолжение таблицы 5 | | | | | |
| № | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат | Результат тестирования |
|  |  | Поле Пароль | Неверный пароль Romanov123 | Ошибка неверный пароль | Ошибка неверный пароль |
| Рисунок 9 - Ошибка авторизации | | | | | |
|  |  |  | Правильный пароль Romanov0 | Открытие следующего окна | Открытие следующего окна |
| Рисунок 10 - Ввод данных в авторизации | | | | | |
| 3 | Добавление новой тренировки | Поле Дата | Ввод завтрашней даты – 05.12.2024 | Ошибка дата не может быть позже сегодняшнего дня | Ошибка дата не может быть позже сегодняшнего дня |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Продолжение таблицы 5 | | | | | | | | |
| № | | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат | | Результат тестирования | |
| Рисунок 11 - Ошибка даты тренировки | | | | | | | | |
|  |  | |  | Ввод даты меньшей чем сегодняшняя – 03.12.2024 | | Переход на поле комментарий | | Переход на поле комментарий |
|  |  | | Поле Длительность | Ввод больше 3,5 часов | | Ошибка: слишком долгая тренировка | | На поле стоит ограничение, не дающее поставить больше 3,5 часов |
|  |  | |  | Ввод менее 3,5 часов | | Добавление тренировки | | Добавление тренировки |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Продолжение таблицы 5 | | | | | | | | |
| № | Наименование функциональности | | Наименование поля | Тестовый набор | Ожидаемый результат | | Результат тестирования | |
| Рисунок 12 - Ввод данных в добавлении тренировки | | | | | | | | |
| 4 | | Вычисление %ЖМТ (Жировой массы тела) | Поле Обхват талии | Ввод слишком большого числа - 300 | | Ошибка: слишком большое число | | Ошибка: Обхват талии должен быть не более 225 см. |
|  | |  |  | Ввод слишком маленького числа - 25 | | Ошибка: Слишком маленькое число | | Ошибка: Обхват талии должен быть не менее 50 см. (Ошибки почти одинаковые, прилагаю только 1 скриншот) |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Продолжение таблицы 5 | | | | | | | | | |
| № | | Наименование функциональности | Наименование поля | | Тестовый набор | Ожидаемый результат | | Результат тестирования | |
| Рисунок 13 - Ошибка охвата талии | | | | | | | | | |
|  |  | |  | Ввод текста – «Двадцать пять» | | | Ошибка: нельзя вводить текст | | Ошибка: Программа не может преобразовать ваш текст в число |
|  |  | |  | Ввод правильного числа - 85 | | | Переход на поле Вес | | Переход на поле Вес |
|  |  | | Поле Вес | Ввод слишком большого числа - 300 | | | Ошибка: слишком большое число | | Ошибка: Вес должен быть не более 165 кг. |
| Рисунок 14 - Ошибка веса | | | | | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Продолжение таблицы 5 | | | | | |
| № | Наименование функциональности | Наименование поля | Тестовый набор | Ожидаемый результат | Результат тестирования |
|  |  |  | Ввод слишком маленького числа - 25 | Ошибка: слишком маленькое число | Ошибка: Вес должен быть не менее 30 кг. |
|  |  |  | Ввод текста «Двадцать пять» | Ошибка: нельзя вводить текст | Ошибка: Программа не может преобразовать ваш текст в число |
|  |  |  | Ввод правильного числа - 75 | Вывод результат вычисления | Вывод результат вычисления |
| Рисунок 15 - Ввод данных в вычислении процента жира | | | | | |

ПРИЛОЖЕНИЕ 4: ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС

1. Окно регистрации. Смотреть рисунок 18 – «Окно регистрации».

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 18 - Окно регистрации

Окно появляется при нажатии на кнопку «Нет аккаунта?» в окне «Авторизация». Заголовок у окна – «Регистрация нового пользователя». Доступны 4 поля для ввода: Имя, Фамилия, Логин, Пароль. Также есть выпадающий список для выбора пола и кнопка «Зарегистрироваться». Она, при вводе правильных данных, и, если они не находятся уже в базе данных создает пользователя. Иначе выходит окно с ошибкой.

1. Окно ошибка такой пользователь уже существует. Смотреть рисунок 19 – «Ошибка при создании пользователя».

Изображение выглядит как текст, снимок экрана, Шрифт, логотип

Автоматически созданное описание

Рисунок 19 - Окно ошибка при создании пользователя

Заголовок у данного окно – «Ошибка». Доступно только одно действие – нажать на кнопку «ОК». Тогда появится возможность выбрать другой логин для нового пользователя

1. Главное окно пользователя. Смотреть рисунок 20 – «Главное окно пользователя».

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 20 - Главное окно пользователя

Главное окно пользователя имеет название «Выбор использования». Оно открывается если правильно ввести все данные пользователя и нажать «Войти» в окне авторизации. В окне находится текст, который здоровается с пользователем по имени, указанному при регистрации. Также снизу написана дата последней тренировки, которая не отображается при отсутствии таковых. Доступны 3 кнопки: «Просмотреть тренировки» - открывает окно с таблицей, «Отследить прогресс» - открывает окно с графиком, «Вычислить %ЖМТ»(Жировую массу тела) - открывает окно опроса, для вычислений.

1. Окно просмотра тренировок. Смотреть рисунок 21 – «Окно просмотра тренировок».

Изображение выглядит как текст, снимок экрана, программное обеспечение, дизайн

Автоматически созданное описание

Рисунок 21 - Окно просмотра тренировок

Окно просмотра тренировок имеет название «Тренировки». Оно открывается если нажать «Просмотреть тренировки» в главном окне пользователя. Доступны 3 кнопки: «Добавить тренировку» - откроет окно для заполнения данных, «Экспорт данных» - откроет системное окно для сохранения табличного файла, кнопка «Назад» - откроет предыдущее окно.

1. Окно добавления тренировки. Смотреть рисунок 22 – «Окно добавления тренировки».

Изображение выглядит как текст, снимок экрана, программное обеспечение, Операционная система

Автоматически созданное описание

Рисунок 22 - Окно добавления тренировки

Это окно является опросом, с помощью которого программа узнает тип тренировки, дату, комментарий и длительность тренировки. Тип тренировки представляет собой выпадающий список, в котором можно выбрать либо указанный тип, либо ввести свой. Дату можно ввести самостоятельно или выбрать в календаре. Комментарий нужно писать самому, а Длительность можно регулировать кнопками со стрелками. Кнопка «Ок» проверит данный и занесёт их в базу данных, кнопка «Отмена» отменить добавление тренировки. После нажатия любой из этих кнопок, данное окно закроется и откроется предыдущее.

1. Окно отслеживания прогресса. Смотреть рисунок 23 – «Окно Прогресс пользователя».

Изображение выглядит как текст, снимок экрана, График, линия

Автоматически созданное описание

Рисунок 23 - Окно Прогресс пользователя

Окно просмотра тренировок имеет название «Прогресс пользователя». Оно открывается если нажать «Отследить прогресс» в главном окне пользователя. Доступны 2 кнопки: «Экспорт данных» - откроет системное окно для сохранения табличного файла с графиком, смотреть рисунок 24 – «Результат экспорта», кнопка «Назад» - откроет предыдущее окно.

Изображение выглядит как текст, снимок экрана, График, линия

Автоматически созданное описание

Рисунок 24 - Результат экспорта

Результат экспорта данных – табличный файл с графиком

1. Окно расчета процента жиров. Смотреть рисунок 25 – «Окно расчета процента жиров».

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 25 - Окно расчета процента жиров

Данные окно называется «Расчёт процента жира», оно открывается при нажатии кнопки «Вычислить %ЖМТ» в главном окне. Это окно является опросом, с помощью которого программа узнает обхват живота спортсмена-любителя и его вес. Доступно 2 кнопки: «Назад» - открывает предыдущее окно, «Вычислить» - программа проверяет данные на правдивость и типы данных, затем выполняет запрос к базе данных для того, чтобы узнать пол пользователя. Затем вся информация подставляется в формулу и вычисляется процент жировой массы тела, который выводится в диалоговом окне.

1. Окно ошибка ввода данных в %ЖМТ. Смотреть рисунок 26 – «Окно ошибок в %ЖМТ».

Изображение выглядит как текст, снимок экрана, Шрифт, логотип

Автоматически созданное описание

Рисунок 26 - Окно ошибок в %ЖМТ

Данное окно имеет название «Ошибка». Оно уведомляет что мы ввели некорректные данные. На выбор только нажать кнопку «ОК», которая закроет ошибку с подсказкой. Таких окон ошибок в процессе вычисления %ЖМТ 5 штук. Ошибки появляются если ввести: Обхват живот меньше 50 или больше 225 сантиметров, вес меньше 30 или больше 165 килограммов или ввести текст, заместо числа. В каждом таком окно будет подсказка, что пользователя сделал не так.

1. Окно с результатом %ЖМТ. Смотреть рисунок 27 – «Окно результата вычисления %ЖМТ».

Изображение выглядит как текст, снимок экрана, Шрифт, мультимедиа

Автоматически созданное описание

Рисунок 27 - Окно результата вычисления %ЖМТ

Окно имеет название «Результат», в нем написан результат вычисления жировой массы тела. Можно только нажать кнопку «ОК», которая вернет к заполнению формы.

Также в рамках проекта предусмотрены следующие окна для администраторов:

1. Окно просмотра пользователей. Смотреть рисунок 28 – «Окно просмотра пользователей».

Изображение выглядит как текст, снимок экрана, программное обеспечение, дизайн

Автоматически созданное описание

Рисунок 28 - Окно просмотра пользователей

В нем администратор может выбрать пользователя и удалить его кнопкой «Удалить пользователя» или выбрать его для дальнейших действий кнопкой «Выбрать пользователя».

1. Окно подтверждения удаления пользователя. Смотреть рисунок 29 – «Окно подтверждения удаления пользователя».

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 29 - Окно подтверждения удаления пользователя

Данное сообщения появляется после нажатия «Удалить пользователя». Оно предупреждает что будут удалены все данные связанные с пользователем

1. Окно редактирования пользователя. Смотреть рисунок 30 – «Окно редактирования пользователя».

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 30 - Окно редактирования пользователя

Это окно откроется если нажать кнопку «Выбрать пользователя» в предыдущем окне. В данном окне можно перейти на предыдущее окно кнопкой «Назад», ввести новый логин для пользователя и сохранить его кнопкой «Обновить логин». Можно удалить данные кнопкой «Удалить пользователя» и можно просмотреть тренировки спортсмена кнопкой «Просмотреть тренировки».

1. Окно записей тренировок пользователя. Смотреть рисунок 31 – «Окно записей тренировок».

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Рисунок 31 - Окно записей тренировок

Это окно откроется если нажать на кнопку «Просмотреть тренировки» в предыдущем окне. Здесь можно просмотреть тренировки, выбрать одну и кнопкой «Удалить выбранную тренировку» удалить её. Кнопкой «Удалить все тренировки» можно удалить все тренировки. Нажав на «Экспорт в Excel» данные импортируются в файл Excel.

1. Окно подтверждения удаления тренировки. Смотреть рисунок 32 – «Диалог подтверждения удаления тренировки».

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 32 - Диалог подтверждения удаления тренировки

Данный диалог возникает при нажатии «Удалить выбранную тренировку». Нужно оно для подтверждения удаления.

1. Окно подтверждения удаления всех тренировок. Смотреть рисунок 33 – «Диалог удаление всех тренировок».

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 33 - Диалог удаление всех тренировок

Данный диалог возникает при нажатии «Удалить все тренировки». Нужно оно для подтверждения удаления всех тренировок.

1. Окно, уведомляющее об удалении. Смотреть рисунок 34 – «Окно уведомление об удалении».

Изображение выглядит как текст, снимок экрана, мультимедиа, Шрифт

Автоматически созданное описание

Рисунок 34 - Окно уведомление об удалении

Данный диалог возникает при успешном удалении одной или всех тренировок. В зависимости от количества удаленных тренировок появиться нужно сообщение.

1. Сообщение об успешном экспорте. Смотреть рисунок 35 – «Сообщение об экспорте».

Изображение выглядит как текст, снимок экрана, Шрифт, мультимедиа

Автоматически созданное описание

Рисунок 35 - Сообщение об экспорте

Данное окно появится после успешного экспорта данных в Excel, после нажатия на «Экспорт в Excel».

1. Окно календаря, смотреть рисунок 36 – «Календарь».

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 6 - Календарь

Данное окно появляется после того, как нажать кнопку открытия календаря в окне добавления тренировки. Здесь можно выбрать дату тренировки. Если выбрать дату, позже сегодняшней, то выйдет ошибка.

ПРИЛОЖЕНИЕ 5: ФУНКЦИИ КНОПОК

1. Регистрация пользователя — включает создание нового аккаунта и сохранение данных в базу данных. Эту функцию активирует кнопка «Нет аккаунта?» на окно входа. Пример кода для регистрации, смотреть рисунок 39 – «Функция регистрация» и рисунок 40 «Функция добавления пользователя».

Изображение выглядит как текст, снимок экрана, документ, Шрифт

Автоматически созданное описание

Рисунок 39 - Функция регистрация

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 40 - Функция добавления пользователя

Функция «register\_user» делает множественные проверки, при прохождении которых активируется функция «add\_user», которая обращается к базе данных и заносит данные нового пользователя. Если проверки не пройдены, то появляется соответственная ошибка.

1. Функция кнопки «Просмотреть тренировки» , смотреть рисунок 41 – «Функция кнопки Просмотреть тренировки».

Изображение выглядит как текст, Шрифт, линия, снимок экрана

Автоматически созданное описание

Рисунок 41 - Функция кнопки Просмотреть тренировки

Данная функция открывает окно для просмотра тренировок — отображает список всех тренировок в таблице, фильтруя их по дате.

1. Функция кнопки «Отследить прогресс» , смотреть рисунок 42 – «Функция кнопки Отследить прогресс».

Изображение выглядит как текст, Шрифт, линия, снимок экрана

Автоматически созданное описание

Рисунок 42 - Функция кнопки Отследить прогресс

Данная функция открывает окно для просмотра прогресса тренировок.

1. Функция кнопки «Вычислить %ЖМТ» , смотреть рисунок 43 – «Вычислить %ЖМТ».

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 43 - Функция кнопки Вычислить %ЖМТ

1. Функция кнопки «Назад» , смотреть рисунок 44 – «функция кнопки Назад».

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 44 - Функция кнопки Назад

Данная функция открывает соответствующее предыдущее окно. В других файлах меняется только 216 строка на имя класса необходимого окна.

1. Добавление тренировки — позволяет пользователю добавить новую тренировку в систему. Функция активируется нажатием кнопки «Добавить тренировку» в окне просмотра тренировок. Пример кода для добавления тренировки, смотреть рисунок 45 – «Добавление тренировки».

Изображение выглядит как текст, снимок экрана, документ, Шрифт

Автоматически созданное описание

Рисунок 45 - Добавление тренировки

Функция «add\_training» Открывает диалог для заполнения данных и берет оттуда данные. Далее производится попытка заполнения данных в базу данных и обновления таблицы тренировок функцией «load\_trainings». Если попытка неудачная, то выходит уведомление об ошибке.

1. Функция кнопки для отображения календаря находится в окне добавления тренировки, смотреть рисунок 46 – «Проверка календаря».

|  |
| --- |
|  |

Рисунок 46 - Проверка календаря

Данные из календаря проверяются, чтобы нельзя было записать тренировку позже сегодняшней даты

1. Функция кнопки «Экспорт данных», расположенной на окне с отображением графика прогресса, смотреть рисунок 47 – «Экспорт графика».

Изображение выглядит как текст, снимок экрана, документ, меню

Автоматически созданное описание

Рисунок 47 - Экспорт графика

Данная функция открывает диалог для сохранения файла в виде Excel, в котором находятся данные и график. Также сохраняется отдельно скриншот графика.

1. Функция кнопки «Вычислить», находится в окне расчёта процента жировой массы тела. , смотреть рисунок 48 – «Вычисление %ЖМТ».

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 48 - Вычисление %ЖМТ

Функция «calculate\_fat\_percentage» пытается преобразовать данные из текстовых полей в числа. Если это не получается, то выводится ошибка. Далее производится вычисление функцией «fat\_percentage» и выводится диалог с результатом.

Также необходимо описать функции кнопок в окнах админа.

1. Функция кнопки «Выбрать пользователя», которая находится в окне выбора пользователей, смотреть рисунок 49 – «Выбор поьзователя».

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 49 - Выбор пользователя

Она осуществляет открытие окна взаимодействия с этим пользователем.

1. Функция кнопки «Удалить пользователя», находящаяся в там же окне выбора пользователей, смотреть рисунок 50 – «Удаление пользователя».

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 50 - Удаление пользователя

Она, если выбрана какай-то колонка, выполняет запрос к базе данных и удаляет все связанные с пользователем данные. Если колонка не выбрана, то выведется ошибка

1. Функция кнопки «Обновить логин», которая расположена в окне изменения пользователя, смотреть рисунок 51 – «Изменение логина».

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 51 - Изменение логина

Она выполняет запрос к базе данных, где изменяет логин выбранного пользователя

1. Кнопка «Удалить пользователя» в окне редактирования пользователя выполняет аналогичную функцию, как на рисунке 50 - «Удаление пользователя».
2. Функция кнопки «Просмотреть тренировки», находящейся в том же окне, предоставлена на рисунке 52 – «Просмотр тренировок».

Изображение выглядит как текст, Шрифт, линия, число

Автоматически созданное описание

Рисунок 52 - Просмотр тренировок

Она открывает окно с просмотром тренировок данного пользователя.

1. Функция кнопки «Удалить выбранную тренировку», расположенной в окне записи тренировок пользователя представлена на рисунке 53 – «Удаление одной тренировки».

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 53 - Удаление одной тренировки

Она проверяет выбрана ли строка, если нет – выводится ошибка. Если строка выбрана, то появляется вопрос «Вы уверены что хотите удалить эту тренировку?», затем выполняется запрос к базе данных, где удаляется данная тренировка.

1. Функция кнопки «Удалить все тренировки», находящейся в том же окне представлена на рисунке 54 – «Удаление всех тренировок».

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 54 - Удаление всех тренировок

Она проверяет выбрана ли строка, если нет – выводится ошибка. Если строка выбрана, то появляется вопрос «Вы уверены что хотите удалить ВСЕ тренировки?», затем выполняется запрос к базе данных, где удаляются все тренировки пользователя.

1. Функция кнопки «Экспорт в Excel» аналогична кнопке Отследить прогресс», смотреть рисунок 55 – «Функция кнопки Отследить прогресс».

ПРИЛОЖЕНИЕ 6: ВЫПОЛНЕНИЕ ТЕСТИРОВАНИЯ

Кейс 2: Авторизация пользователя

Тестируемая функциональность: Авторизация с проверкой введенных данных.

Тестовый набор: Проверка полей "Имя пользователя" и "Пароль".

Неправильный логин «Alex2» - ожидаемый результат: ошибка «Неверное имя пользователя.» Результат – Ошибка «Неверное имя пользователя или пароль». Смотреть рисунок 57 – «Ошибка неверное имя или пароль».

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рисунок 57 - Ошибка неверное имя или пароль

Правильный логин «Alex1» и пароль «Romanov0» - ожидаемый результат: открытие главного окна. Результат – Вход в аккаунт и открытие главного окна пользователя. Смотреть рисунок 58 – «Ввод правильных данных» и рисунок 59 – «Главное окно при корректных данных».

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 58 - Ввод правильных данных

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 59 - Главное окно при корректных данных

Кейс 3: Добавление новой тренировки

Тестируемая функциональность: Добавление новой записи о тренировке.

Тестовый набор: Проверка поля "Дата" и "Длительность".

* Дата больше сегодняшнего дня «05.12.2024» — ожидаемый результат: ошибка "Дата не может быть позже сегодняшнего дня." Результат: ошибка "Дата не может быть позже сегодняшнего дня." Смотреть рисунок 60 – «Ошибка дата позже сегодняшнего дня».

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 60 - Ошибка дата позже сегодняшнего дня

Кейс 4: Вычисление %ЖМТ

Тестируемая функциональность: Расчет показателей на основе введенных данных.

Тестовый набор: Проверка полей "Обхват живота" и "Вес".

* Ввод слишком большого числа «300» — ожидаемый результат: ошибка "Слишком большое число." Смотреть рисунок 61 – «Ошибка слишком большое число».

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 61 - Ошибка слишком большое число

* Ввод в обхват живота «85», в Вес «75». Ожидаемый результат: примерно 20%. Результат: 21,86%. Смотреть рисунок 62 – «Ввод корректных данных в %ЖМТ».

Изображение выглядит как текст, снимок экрана, программное обеспечение, мультимедиа

Автоматически созданное описание

Рисунок 62 - Ввод корректных данных в %ЖМТ

ПРИЛОЖЕНИЕ 7: СТРУКТУРА ПРОЕКТА

Изображение выглядит как текст, снимок экрана, Шрифт, документ

Автоматически созданное описание

Рисунок 63 - Структура проекта