

УДК 004.65

UDC 004.65

5.2.2. – Математические, статистические и инструментальные методы в экономике

5.2.2 – Mathematical, statistical and instrumental methods in economics

**РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ «ОБРАБОТКА И АНАЛИЗ ДАННЫХ ФИТНЕС-БРАСЛЕТА»**

**DEVELOPMENT OF AN INFORMATION SYSTEM "PROCESSING AND ANALYSIS OF FITNESS BRACELET DATA"**

Оксанич Елена Анатольевна  
Кандидат экономических наук, доцент  
РИНЦ SPIN-код: 3138-8291  
email: [oksanich.elena@rambler.ru](mailto:oksanich.elena@rambler.ru)  
*ФГБОУ «Кубанский государственный аграрный университет», 350044, Россия, г. Краснодар, ул. Калинина 13*

Oksanich Elena Anatolievna  
Candidate of Economic Sciences associate professor  
RSCI SPIN-code: 3138-8291  
email: [oksanich.elena@rambler.ru](mailto:oksanich.elena@rambler.ru)  
*Kuban State Agricultural university, 350044, Russia, Krasnodar, Kalinina, 13*

Рыбьянцева Мария Сергеевна  
Кандидат экономических наук, доцент  
РИНЦ SPIN-код: 7874-8981  
email: [Riban1@mail.ru](mailto:Riban1@mail.ru)  
*ФГБОУ «Кубанский государственный аграрный университет», 350044, Россия, г. Краснодар, ул. Калинина 13*

Rybyantseva Maria Sergeevna  
Candidate of Economic Sciences associate professor  
RSCI SPIN-code: 7874-8981  
email: [Riban1@mail.ru](mailto:Riban1@mail.ru)  
*Kuban State Agricultural university, 350044, Russia, Krasnodar, Kalinina, 13*

Мозговая Анастасия Александровна  
студент группы ИТ2302  
email: [anastasia.m2085@gmail.com](mailto:anastasia.m2085@gmail.com)  
*ФГБОУ «Кубанский государственный аграрный университет», 350044, Россия, г. Краснодар, ул. Калинина 13*

Mozgovaya Anastasya Aleksandrovna  
student of group IT2302  
email: [anastasia.m2085@gmail.com](mailto:anastasia.m2085@gmail.com)  
*Kuban State Agricultural university, 350044, Russia, Krasnodar, Kalinina, 13*

В современном мире здоровый образ жизни и мониторинг физической активности становятся все более популярными. Фитнес-браслеты и умные часы собирают огромное количество данных о физической активности, сне, пульсе и других показателях здоровья. Однако без удобного инструмента для анализа и обработки эта информация остается неструктурированной и сложной для интерпретации. Разработка базы данных для анализа и обработки данных фитнес-браслета позволит пользователям систематизировать, визуализировать и эффективно использовать собранные данные для улучшения физической формы, контроля здоровья и повышения мотивации. Разрабатываемая БД создается для компании, производящей фитнес-браслеты «VitaBand». Разрабатываемая БД для фитнес-приложения компании "VitaBand" позволит пользователям эффективно анализировать свою активность, улучшать физическую форму и контролировать здоровье.

In the modern world, a healthy lifestyle and monitoring of physical activity are becoming increasingly popular. Fitness bracelets and smart watches collect a huge amount of data on physical activity, sleep, heart rate and other health indicators. However, without a convenient tool for analysis and processing, this information remains unstructured and difficult to interpret. Development of a database for analyzing and processing fitness bracelet data will allow users to systematize, visualize and effectively use the collected data to improve physical fitness, monitor health and increase motivation. The developed database is created for the company that produces fitness bracelets "VitaBand". The developed database for the fitness application of the company "VitaBand" will allow users to effectively analyze their activity, improve physical fitness and monitor health

Ключевые слова: ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ, БАЗА ДАННЫХ, ФИТНЕС-БРАСЛЕТ, ЗАПРОСЫ К БАЗЕ ДАННЫХ

Keywords: INFORMATION SYSTEMS DESIGN, DATABASE, FITNESS BRACELET, DATABASE QUERIES

<http://dx.doi.org/10.21515/1990-4665-212-025>

При разработке информационной системы предметной областью является фитнес-аналитика и мониторинг здоровья, включая:

- отслеживание физической активности (шаги, км, продолжительность);
- учет тренировок (тип, время активности);
- анализ сна (продолжительность, время начала и конца сна);
- контроль пульса;
- назначение разработки базы данных.

Разрабатываемая база данных является ключевым компонентом фитнес-приложения, обеспечивающим:

- взаимодействие между пользовательскими устройствами и серверной частью;
- формирование отчетов и аналитики для пользователей;
- хранение истории активности для долгосрочного анализа.

При оценке бизнес-процессов интерес вызывает исследование подпроцессов углубленного анализа и обработки конкретных данных (рисунок 1), включающий семь блоков.

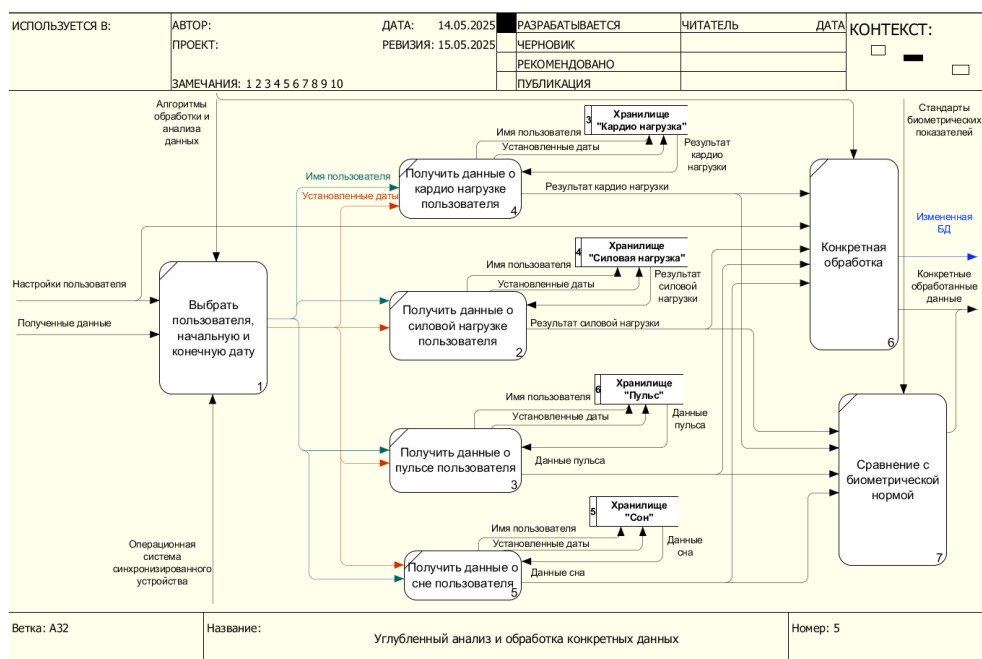


Рисунок 1 – Диаграмма 4 уровня

В результате модель бизнес-процессов помогла определить ключевые сценарии взаимодействия пользователя с системой, выявить роли, данные и операции, критичные для работы приложения, заложить основу для проектирования реляционной модели базы данных.

На основе функционального подхода и модели бизнес-процессов было выделено несколько основных сущностей:

1. Пользователи. Пользователь – центральный объект системы, так как все данные привязаны к нему. Без этой сущности невозможно персонализировать данные и обеспечить разделение доступа. Примеры атрибутов: ID, имя, вес, рост.

2. Активность разделяется на два основных типа: кардионагрузки и силовые тренировки. Такое разделение обусловлено принципиальными различиями в методиках расчета и анализе этих нагрузок. Кардионагрузки (бег, ходьба) характеризуются атрибутами продолжительности, дистанции и пульса, тогда как силовые тренировки учитывают, количество подходов, время. Это разделение позволяет более точно оценивать эффективность разных видов тренировок.

3. Сон - важнейшая сущность, отражающая качество восстановления организма. Она хранит данные о продолжительности, времени пробуждения и засыпания. Эти показатели критичны для анализа общего состояния здоровья, оценки влияния физических нагрузок на организм и формирования рекомендаций по режиму дня.

4. Пульс представляет собой ключевой медицинский показатель, который измеряется как во время активности, так и в состоянии покоя. Эта сущность особенно важна для контроля интенсивности тренировок, выявления потенциальных проблем со здоровьем и анализа восстановления после нагрузок.

5. Дата - необходимая сущность, которая обеспечивает временную привязку всех записей. Она позволяет анализировать динамику

показателей, выявлять закономерности и сравнивать результаты за разные периоды. Правильная организация данных по времени критична для формирования статистики и долгосрочного мониторинга состояния пользователя.

Каждая из перечисленных сущностей играет важную роль в системе, обеспечивая комплексный подход к фитнес-трекингу. Их тщательная проработка позволяет создать гибкую и масштабируемую структуру базы данных, способную учитывать все аспекты физической активности и состояния пользователя. Взаимосвязи между этими сущностями образуют целостную систему мониторинга здоровья и физической формы.

Модель «сущность-связь» представлена на Рисунке 2.

Приложение запускается через среду разработки Visual Studio.

Целевая платформа NET 8.0.

Для использования приложения необходимо установить загрузочный файл Visual Studio.

В папке «Загрузки» дважды щелкнуть загрузчик с именем VisualStudioSetup, чтобы начать установку.

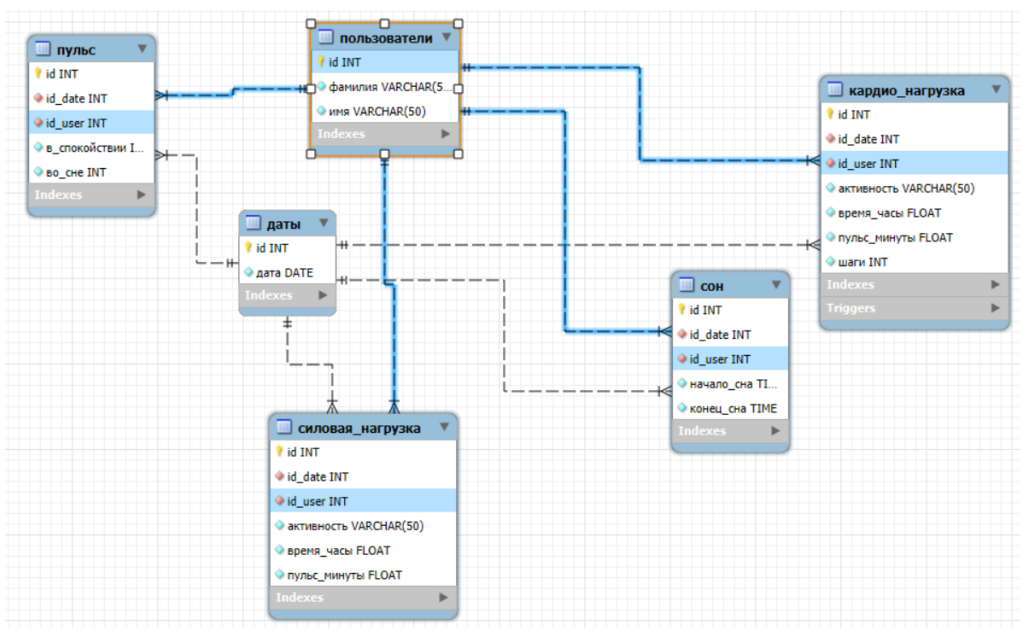


Рисунок 2 – ER-диаграмма

После установки установщика Visual Studio его можно использовать для настройки установки, выбрав нужные наборы компонентов или рабочие нагрузки. Когда установка Visual Studio завершится, необходимо нажать на кнопку «Запустить», чтобы приступить к работе в Visual Studio. Запуск программы можно осуществлять как непосредственно через сам исполняемый файл (Fitness\_Bracelet.exe), так и через ярлык (если был создан) (Рисунок 3).

runtimes	11.05.2025 17:34	Папка с файлами	
BouncyCastle.Cryptography.dll	07.02.2025 9:15	Расширение при...	4 757 КБ
Fitness_Bracelet.deps.json	11.05.2025 17:34	Исходный файл J...	17 КБ
Fitness_Bracelet.dll	17.05.2025 1:00	Расширение при...	52 КБ
Fitness_Bracelet.exe	17.05.2025 1:00	Приложение	136 КБ
Fitness_Bracelet.pdb	17.05.2025 1:00	Program Debug D...	23 КБ
Fitness_Bracelet.runtimeconfig.json	11.05.2025 17:34	Исходный файл J...	1 КБ
Google.Protobuf.dll	05.03.2025 1:16	Расширение при...	479 КБ
K4os.Compression.LZ4.dll	18.03.2024 0:56	Расширение при...	69 КБ
K4os.Compression.LZ4.Streams.dll	18.03.2024 0:56	Расширение при...	83 КБ
K4os.Hash.xxHash.dll	08.11.2022 21:08	Расширение при...	13 КБ
MySQL.Data.dll	24.03.2025 13:46	Расширение при...	1 171 КБ
System.IO.Pipelines.dll	29.10.2024 5:46	Расширение при...	76 КБ
System.Text.Encodings.Web.dll	29.10.2024 5:40	Расширение при...	70 КБ
System.Text.Json.dll	29.10.2024 5:47	Расширение при...	629 КБ
ZstdSharp.dll	26.02.2025 21:10	Расширение при...	386 КБ

Рисунок 3 - Fitness\_Bracelet.exe

Приложение состоит из двух форм.

Первая Form1. Она открывается при запуске приложения и пока находится незаполненной (рисунок 4).

Рисунок 4 - Form1

На ней представлены dataGridView1 – dataGridView4, в эти элементы будут заноситься отсортированные таблицы из базы данных. Пустые поля по всей форме это labels, которые тоже будут заполняться данными.

Пока пользователь и даты начала/окончания не определены. Чтобы выбрать пользователя и даты нужно нажать на button1 – кнопку «Настроить» (верхний правый угол) (рисунок 5).

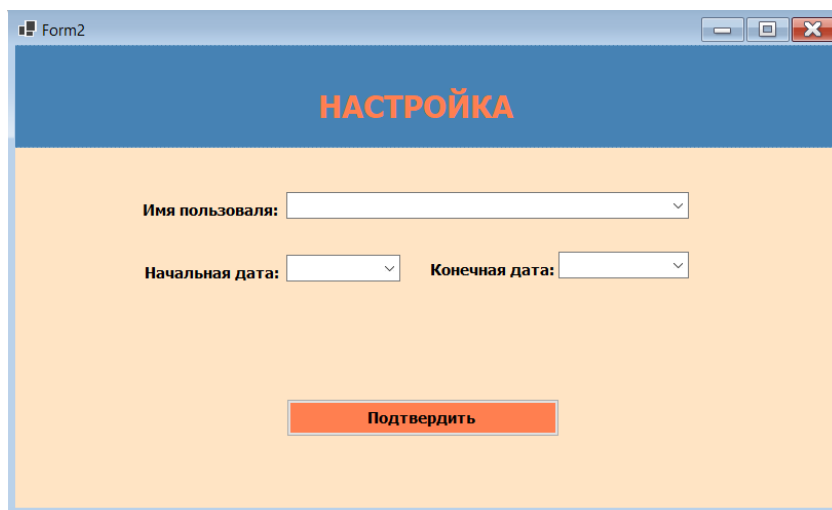


Рисунок 5 - Form2

При нажатии на кнопку «Настроить» появилась Form2. Нам предложено выбрать имя пользователя, начальную и конечную дату. Когда данные установлены, нажимается кнопка «Подтвердить». После подтверждения Form2 закрывается и открывается уже заполненная Form1 (рисунок 6).

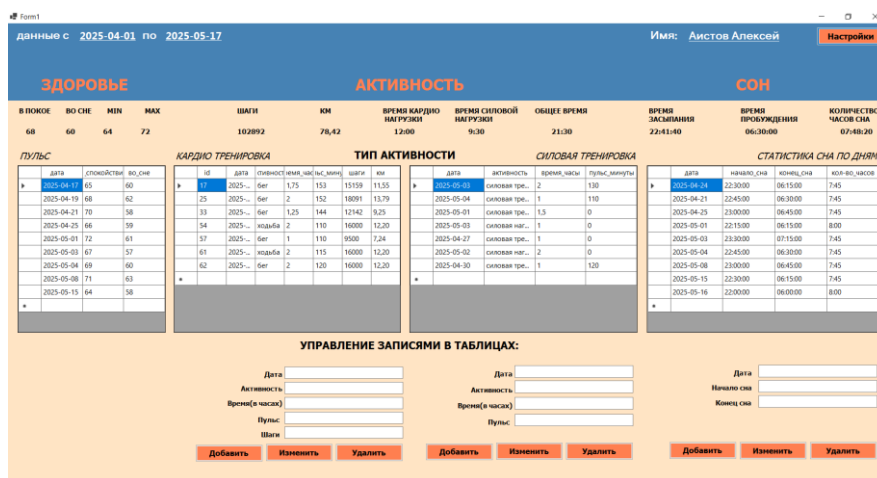


Рисунок 6 – заполнение Form1

Исходя из указанных данных на Form 2 были заполнены элементы Form1. Появились таблицы: пульс, кардио\_тренировка, силовая\_тренировка, сон. Эти таблицы показывают данные в данном случае Аистова Алексея за месяц. Из данных выведенных таблиц подсчитываются разные параметры: шаги, км, время кардио нагрузки, время силовой нагрузки и тд, все эти данные заносятся в labels.

Запросы об информации здоровья:

1. Запрос находит минимальное значение пульса в состоянии покоя для конкретного пользователя за указанный период времени.

```
SELECT MIN(B_спокойствии)
FROM пульс
WHERE id_user = 1
AND id_date IN (SELECT id FROM даты
WHERE дата BETWEEN "2025-04-17" AND "2025-05-17")
```

2. Запрос находит максимальное значение пульса в состоянии покоя для конкретного пользователя за указанный период времени.

```
SELECT MAX(B_спокойствии)
FROM пульс
WHERE id_user = 1
AND id_date IN (SELECT id FROM даты
WHERE дата BETWEEN "2025-04-17" AND "2025-05-17")
```

3. Запрос находит среднее арифметическое значение пульса в состоянии покоя для конкретного пользователя за указанный период времени.

```
SELECT AVG(B_спокойствии)
FROM пульс
WHERE id_user = 1
AND id_date IN (SELECT id FROM даты
```

WHERE дата BETWEEN "2025-04-17" AND "2025-05-17")

4. Запрос находит среднее арифметическое значение пульса во сне для конкретного пользователя за указанный период времени.

```
SELECT AVG(Во_сне)
```

```
FROM пульс
```

```
WHERE id_user = 1
```

```
AND id_date IN (SELECT id FROM даты
```

```
WHERE дата BETWEEN "2025-04-17" AND "2025-05-17")
```

Итоги подсчетов данных о здоровье выводятся в отведенных полях (рисунок 7).

<b>ЗДОРОВЬЕ</b>			
<b>В ПОКОЕ</b>	<b>ВО СНЕ</b>	<b>MIN</b>	<b>MAX</b>
<b>68</b>	<b>60</b>	<b>64</b>	<b>72</b>

Рисунок 7 – Данные о здоровье

Запросы об информации активности:

1. Запрос вычисляет общее количество шагов для конкретного пользователя за указанный период времени, используя оконную функцию SUM() с PARTITION BY. Сначала выбираются все записи о кардионагрузках указанного пользователя за заданный диапазон дат, затем оконная функция SUM() с группировкой по id\_user вычисляет общее количество шагов для этого пользователя.

```
SELECT DISTINCT
```

```
SUM(шаги) OVER (PARTITION BY id_user) AS total_steps
```

```
FROM кардио_нагрузка
```

```
WHERE id_user = 1 AND id_date IN (SELECT id FROM даты
```

```
WHERE дата BETWEEN "2025-04-17" AND "2025-05-17" )
```

2. Подсчет общего количества км с помощью функции суммы.

```
SELECT SUM(шаги / 1312) FROM кардио_нагрузка
```



WHERE id\_user = 1 AND id\_date IN (SELECT id FROM даты  
WHERE дата BETWEEN "2025-04-17" AND "2025-05-17")

3. Общее время кардионагрузок, посчитали через сумму.

SELECT SUM(время\_часы) FROM кардио\_нагрузка

WHERE id\_user = 1 AND id\_date IN (SELECT id FROM даты  
WHERE дата BETWEEN "2025-04-17" AND "2025-05-17")

4. Общее время силовых нагрузок, посчитали через сумму.

SELECT SUM(время\_часы) FROM силовая\_нагрузка

WHERE id\_user =2 AND id\_date IN (SELECT id FROM даты  
WHERE дата BETWEEN "2025-04-17" AND "2025-05-17")

Итоги подсчетов данных об активности выводятся в отведенных полях (рисунок 8).

АКТИВНОСТЬ				
ШАГИ	КМ	ВРЕМЯ КАРДИО НАГРУЗКИ	ВРЕМЯ СИЛОВОЙ НАГРУЗКИ	ОБЩЕЕ ВРЕМЯ
102892	78,42	12:00	9:30	21:30

Рисунок 8 –данные об активности

Запросы об информации сон:

1. Запрос вычисляет среднее время начала сна для конкретного пользователя за указанный период. Сначала он выбирает все записи о сне из таблицы "сон" (вероятно, опечатка, должно быть "сон") для заданного пользователя (@userId) в указанном диапазоне дат (между @startDate и @endDate). Для вычисления среднего значения времени используется несколько вложенных функций.

```
SELECT
    TIME_FORMAT(
        SEC_TO_TIME(AVG(TIME_TO_SEC(начало_сна))), '%H:%i:%s')
    FROM сон
    WHERE id_user = 1 AND id_date IN (SELECT id FROM даты
```

WHERE дата BETWEEN "2025-04-17" AND "2025-05-17")

2. Запрос берет данные из таблицы "сон", выбирая все записи для указанного пользователя в заданном диапазоне дат. Для каждого случая он берет время окончания сна из поля "конец\_сна", преобразует это время в секунды с помощью функции TIME\_TO\_SEC, затем вычисляет среднее арифметическое всех этих значений и снова преобразует результат из секунд обратно в формат времени SEC\_TO\_TIME. Финальная функция TIME\_FORMAT приводит результат к стандартному формату "часы:минусы:секунды".

```
SELECT
    TIME_FORMAT(
    SEC_TO_TIME(AVG(TIME_TO_SEC(конец_сна))), '%H:%i:%s')
    FROM сон
    WHERE id_user = 1 AND id_date IN (SELECT id FROM даты
    WHERE дата BETWEEN "2025-04-17" AND "2025-04-17")
```

3. Запрос сначала вычисляется разница в секундах между временем начала "начало\_сна" и окончания сна "конец\_сна", используя функцию TIMESTAMPDIFF. Для всех записей вычисляется средняя продолжительность сна в секундах (AVG), которая затем округляется в меньшую сторону (FLOOR) для устранения дробных секунд. Полученное среднее значение в секундах преобразуется обратно в формат времени (SEC\_TO\_TIME) и форматируется в удобочитаемый вид "ЧЧ:ММ:СС" (TIME\_FORMAT).

```
SELECT TIME_FORMAT(SEC_TO_TIME(
    FLOOR(AVG(TIMESTAMPDIFF(SECOND, начало_сна,
    CASE
    WHEN конец_сна < начало_сна
    THEN ADDTIME(конец_сна, '24:00:00')
    ELSE конец_сна END))))), '%H:%i:%s')
```

FROM сон

WHERE id\_user = 1 AND id\_date IN (SELECT id FROM даты

WHERE дата BETWEEN "2025-04-17" AND "2025-05-17")

Итоги подсчетов данных о сне выводятся в отведенных полях (рисунок 9).

СОН		
ВРЕМЯ ЗАСЫПАНИЯ	ВРЕМЯ ПРОБУЖДЕНИЯ	КОЛИЧЕСТВО ЧАСОВ СНА
22:41:40	06:30:00	07:48:20

Рисунок 9 – заполнение данных о сне

Есть возможность добавить, изменить и удалить данные выведенных таблиц (рисунок 10).

Дата

Активность

Время(в часах)

Пульс

Шаги

Добавить
Изменить
Удалить

Рисунок 10 – Пример функционала «Добавить, Изменить, Удалить»

Под каждой таблицей расположены textBoxs – элементы, соответствующие данным из таблицы выше. Под ними 3 buttons: Добавить, Изменить, Удалить.

Чтобы добавить строку нужно заполнить textBoxs над кнопкой и нажать на кнопку, новая строка добавится в таблицу и запишется в базу данных.

Чтобы изменить строку, нужно два раза кликнуть по ячейке, которую необходимо изменить, после этого ее данные автоматически занесутся в нужные textBoxs. После этого в элементах необходимо провести правку , а

затем нажать кнопку «Изменить», старая строка изменится в таблице и в базе данных.

Чтобы удалить строку, нужно выбрать ее и нажать на кнопку «Удалить». Она удалится из таблицы и из базы данных.

Разработанная база данных для приложения «Анализ и обработка данных фитнес-браслета» представляет собой комплексное решение, позволяющее пользователям эффективно управлять и анализировать данные о своей физической активности, сне и показателях здоровья. Проект создан для компании, которая специализируется на производстве умных фитнес-браслетов, и направлен на повышение удобства пользователей за счет автоматизированного сбора, хранения и визуализации данных.

Значение проекта для пользователей и компании.

– Для пользователей приложение становится персональным помощником в ведении здорового образа жизни, позволяя: контролировать уровень физической активности, анализировать качество сна и корректировать режим, отслеживать динамику пульса и выявлять возможные отклонения.

– Для компании база данных является ключевым элементом экосистемы, который: повышает лояльность клиентов за счет удобного функционала, позволяет собирать анонимизированную статистику для улучшения продукции, создает основу для будущих функций (например, интеграция с медицинскими сервисами).

Разработанная база данных полностью соответствует требованиям заказчика и предоставляет пользователям фитнес-браслета мощный инструмент для мониторинга здоровья, физической активности и сна. Благодаря удобному интерфейсу, гибкости и надежности хранения данных, проект способствует популяризации здорового образа жизни и повышает ценность продуктов компании. В перспективе система может

быть масштабирована и дополнена новыми функциями, что сделает ее еще более востребованной на рынке фитнес-технологий.

**Список использованной литературы:**

1. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. : учебник / Д. Рихтер - 4-е изд. – СПб.: Питер, 2024 – 896 с.
2. MySQL по максимуму : учебное пособие / С. Ботрос, Д. Тинли — 4-е изд. — СПб : Питер, 2023. — 432 с.
3. Грокаем Алгоритмы : учебное пособие / А. Бхаргава, Е.П. Матвеев – 2-е изд. – СПб: Питер, 2025. – 352 с.
4. SQL. Pocket Guide : учебник / Э. Жао – 4-е изд. – Казахстан: Sprint Book, 2025. – 320 с.
5. C#. Основы программирования : учебное пособие / С. Наков – Москва : АСТ, 2024. – 498 с.

**References:**

1. CLR via C#. Programmirovaniie na platforme Microsoft .NET Framework 4.5 na jazyke S#. : uchebnik / D. Rihter - 4-e izd. – SPb.: Piter, 2024 – 896 с.
2. MySQL po maksimumu : uchebnoe posobie / S. Botros, D. Tinli — 4-e izd. — SPb : Piter, 2023. — 432 с.
3. Grokaem Algoritmy : uchebnoe posobie / A. Bhargava, E.P. Matveev – 2-e izd. – SPb: Piter, 2025. – 352 с.
4. SQL. Pocket Guide : uchebnik / Je. Zhao – 4-e izd. – Kazahstan: Sprint Book, 2025. – 320 с.
5. C#. Osnovy programmirovaniija : uchebnoe posobie / S. Nakov – Moskva : AST, 2024. – 498 с.